MICROSOFT® BUSINESS SOLUTIONS NAVISION® 4.0

COURSE: 8404A INSTALLATION AND CONFIGURATION TRAINING

Last Revision: July 2004

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

INTRODUCTION

Welcome

We know training is a vital component of retaining the value of your Microsoft® Business Solutions investment. Our quality training from industry experts keeps you up-to-date on your solution and helps you develop the skills necessary for fully maximizing the value of your solution. Whether you choose Online Training, Classroom Training, or Training Materials, there's a type of training to meet everyone's needs. Choose the training type that best suits you so you can stay ahead of the competition.

Online Training

Online Training delivers convenient, in-depth training to you in the comfort of your own home or office. Online training provides immediate access to training 24 hours a day. It's perfect for the customer who doesn't have the time or budget to travel. Our newest online training option, eCourses, combine the efficiency of online training with the in-depth product coverage of classroom training, with at least two weeks to complete each course.

Classroom Training

Classroom Training provides serious, in-depth learning through hands-on interaction. From demonstrations to presentations to classroom activities, you'll receive hands-on experience with instruction from our certified staff of experts. Regularly scheduled throughout North America, you can be sure you'll find a class convenient for you.

Training Materials

Training Materials enable you to learn at your own pace, on your own time with information-packed training manuals. Our wide variety of training manuals feature an abundance of tips, tricks, and insights you can refer to again and again:

Microsoft Business Solutions Training Courseware: The Microsoft Business Solutions Training Courseware are very detailed training manuals, designed from a training perspective. These manuals include advanced topics as well as training objectives, exercises, interactions, and quizzes.

Look for a complete list of manuals available for purchase on the Microsoft Business Solutions website: www.microsoft.com/BusinessSolutions.

Overview

Microsoft® Business Solutions–Navision® is a collaborative business management solution for medium-sized companies. It is an integrated product that includes functionality to support financial and relationship management, distribution, and manufacturing.

This course is designed for students who are preparing to complete the installation and configuration of Microsoft Navision. The course primarily provides explanations of the databases and servers associated with Microsoft Navision, client hardware and software requirements, as well as various database administration tasks.

About the Navision Installation and Configuration Training Manual

This training manual provides a conceptual and operational description of the installation and configuration of Microsoft® Business Solutions–Navision®. The manual can be used both in the context of an instructor-led training course and as reference material for self-teaching.

Target Audience

This training manual is intended for Microsoft Certified Business Solutions Partners' employees selling, implementing and supporting Microsoft Navision.

Training Objectives

Whether used in a course setting or for self-teaching, the manual is designed to equip course participants with product knowledge, both at conceptual and functional levels, required when installing, configuring, implementing and supporting Microsoft Navision.

Training Prerequisites

To successfully participate in the Installation and Configuration course or complete the training in the self-teaching manner, participants don't need any training prerequisites. Although, participants should have a basic understanding of servers, client hardware and software, and system administration.

Training Manual Overview

This manual provides the participants with an in depth understanding of the hardware, server options, and database administration necessary to install and configure Microsoft Navision. The manual consists of the following chapters:

- Introduction
- Chapter 1: Hardware and Software Requirements
- Chapter 2: Microsoft Navision Architecture
- Chapter 3: Microsoft Navision Client Installation
- Chapter 4: Microsoft Navision Database Server
- Chapter 5: Microsoft Navision SQL Server Option
- Chapter 6: Microsoft Navision Application Server
- Chapter 7: Microsoft Navision ODBC Driver
- Chapter 8: Microsoft Navision Security
- Chapter 9: Basic Customizations
- Chapter 10: Microsoft Navision Database Administration Tools
- Chapter 11: Optimizing Microsoft Navision
- Chapter 12: Troubleshooting Microsoft Navision

Every chapter includes an overview, describes the requirements, and explains the setup procedures.

In this manual, the chapters are organized in a logical way that reflects the relationship between the installation and configuration of the program. It is therefore recommended that the course (when built on this manual) or self-teaching activity follows the suggested teaching path. If preferred, however, the users of the manual can choose alternative paths.

The chapters in this training manual typically contain the following elements:

- concepts
- scenarios
- exercises

Concepts explain at a high level how to set up and configure the system to best meet a client's business process.

Scenarios represent the typical business situations where the application is used. Scenarios provide a basis for examples that give practical step-by-step descriptions of how to use the application.

Exercises allow the course participants (manual readers) to practice using the program, where applicable.

Dates

The dates in the course manual are in the American format. In order to avoid confusion when reading the manual and following the exercises, if necessary, it is a good idea to change the date format in Windows before starting the training session.

The working date is set to be 01/25/01 (January 25, 2001). It is important that this date is used when following the examples and doing the exercises included in the manual.

Demonstration Data

All examples and exercises in this manual are based on a fictitious company, Cronus International Ltd.

License Information

To run the examples and exercises included in this training manual, you need a standard Cronus license, which is provided, on your product CD.

Suggested Course Duration

When this training manual is used to run a course, it is recommended that the course is taught over a period of three days.

Further Information

You can learn more about Microsoft Navision in the Microsoft Navision Overview course.

Student Objectives

What do you hope to learn by participating in this course?

List three main objectives below.

1.

2.

3.

CHAPTER 1: HARDWARE AND SOFTWARE REQUIREMENTS

Overview

This chapter leads you through the database, server, and client installation decisions necessary for a successful implementation.

Before you begin installing and configuring Microsoft® Business Solutions–Navision®, you will need to consider the network, database, and server needs and clarify any questions with your customer. Therefore, you must be clear on these issues:

- Which database you want to install
- Whether you need Microsoft Navision 2-tiered or n-tiered
- Whether the client installation should be shared
- Whether the client software should be installed on the server

Choosing the Installation Type and Equipment

Before installing Microsoft Navision, you should carefully consider the needs of your customers' organization as this will greatly influence the type of installation you will require and the hardware that this will involve.

Single-User and Multiuser Installations

Microsoft Navision can be used in single-user or multiuser installations.

- In single-user installations, all the work is done on one computer, and all the information (the database) is stored on this computer. Even if you purchase several single-user installations and run each on its own computer, the entire installation is still called "single-user" because the information is stored on each individual computer and not in one centrally located database.
- In a multiuser installation, meaning many users share common information that is stored in one or more databases on a server. The computers that work with the data are called clients, and the way the server and the computers work together is called a client/server installation.

What You Should Know Before Installing the Program

Regardless of whether you have chosen to work in a single-user or multiuser environment, you start by purchasing the same package. The program is installed from the same CD and with the same installation wizard for both single-users and clients.

For a multiuser installation, the CD contains a Microsoft Navision Database Server package that is installed only on the server computer. However, the demonstration database that comes with a single-user installation is not included in the server package. You need some of the data in the demonstration database (and you need a client or single-user to modify a database), so you must always install a single-user somewhere in the network in a multiuser installation. The simplest way to do this is to install a single-user on the same computer as the server. You can read about this in Chapter 4 titled "Microsoft Navision Installation." The single-user can later be converted to a client by connecting it to the server's database.

Once the program has been installed, you must customize it for your particular situation. Not all installations need the same size database. For example, a database server, which serves many users, probably needs a larger database than a single–user installation.

With Microsoft Navision, you do not need to plan for the distant future when you purchase your installation or set it up. You can change or expand the installation whenever you like. To switch from a single-user to a multiuser installation, you purchase a server package and the number of session licenses that you need. You can place the database that you already have on a server so that several people can work with it. If, after a while, you need a larger database, you can expand the existing one.

Choosing Equipment

Microsoft Navision does not require particularly sophisticated equipment, but as with all programs, the better your equipment, the better the results. You get the best solution with the optimal equipment and with the program settings optimized for that equipment.

This section contains a brief description of the equipment recommended for your Microsoft Navision installation.

Choice of Computers

In a multi-user installation, you can, in principle, use the same type of computer for both the clients and the server, but there is a difference in how much CPU power, memory, and disk space each will need. Details about required and recommended hardware and software, including operating system requirements, can be found in the next section, "Hardware and Software Requirements."

Computers for Clients and Single-User Installations

Computers for clients or a single-user installation that run Windows XP or Windows 2000 must comply with the requirements specified by Microsoft. Because the client computers process the data they retrieve, they need a relatively large amount of computing power. If the client computers are too slow, they will take a long time to finish the calculations for a transaction. A slow computer working on a transaction that must update the database can delay all the other clients until it finishes.

Computers for Servers

Computers for servers that run Windows 2000 or Windows Server 2003 must comply with the requirements specified by Microsoft.

The database is stored on the server, making it a critical area of the application since several users can access it at the same time. Therefore, it is important to select a powerful computer for the server. In addition, there are certain aspects of the server that require further consideration:

- The hard disk and controller
- The RAID system
- The memory
- The network adapter
- The CPU

The Hard Disk and Controller

The hard disk is the slowest component in a computer because it consists of mechanical parts. Access times to the hard disk are long compared to those to memory (normal access time to memory is less than 60 nanoseconds and access time to a hard disk is faster than 10 milliseconds). All of the programs and information are stored on the hard disk, so data is continuously read from and written to the disk. Since there is only one read/write head in a hard disk, only one read or write operation can be carried out at a time.

By using more than one disk in your system, you can increase performance dramatically. You must, however, use a hard disk controller that supports control of more than one hard disk at a time without increasing access time to the disks. Furthermore, it is important for the controller to have a high transfer rate so data can travel quickly between the memory and the hard disk. The use of CPU per disk transfer must also be minimized. An example of a controller with these features is the Fast Wide SCSI 2 (Small Computer System Interface) controller.

WARNING: *Do not use the write-back or lazy-write caching systems that are built into your hard disk controller unless the disk controller has a battery backup. Using a battery-supported hard disk controller will prevent loss of data that might otherwise result if the system suffered a power failure.*

You should also be aware of the write-cache facility that most of today's hard disks use. When you buy a hard disk, be sure that you can disable its write-cache (using software or a jumper on the disk). If a power failure occurs when data is still in this cache, you could lose the data.

It is also necessary to have some sort of error detection unit implemented to allow the controller to determine when a byte of data in the cache is corrupted – for example, from a single-bit error or a defective memory chip.

Furthermore, any errors that occur must be corrected, so some sort of correction scheme must be implemented in the controller. An ECC (Error Correction Code) RAM is an example of this kind of correction scheme.

Speed and Hard Disk Capacity

You can increase performance by several hundred percent by expanding your system from one to six hard disks. To avoid poor performance in your daily work, you should add more than one hard disk to your system and divide the database among these hard disks. Four relatively slow hard disks (for example of 5 GB each) perform much better together than one super hard disk (of 13 GB). Tests have shown that the four-hard-disk configuration allows more than twice the number of transactions per second than the one-hard-disk configuration. Each hard disk that you add will improve performance.

As a rule of thumb, you could say that each time you double the number of hard disks, you increase performance by 100%.

Several "intelligent" controllers exist, which can control several hard disks simultaneously, for example, RAID (Redundant Array of Independent Disks) systems.

RAID Systems

If you have a very large Microsoft Navision installation, you should consider using a RAID system. This system consists of several disks. The key feature of a RAID system is that the failure of one disk does not bring the entire system down. You can recreate the lost data from parity information or from a direct copy stored on the other disks.

Several RAID configurations exist and are described in the following table:

Item	Description
RAID 0	Called striping. The data is broken up into chunks as it is written across all the drives. This provides the highest performance, but no redundancy is provided.
RAID 1	Called mirroring. The data is written redundantly to pairs of drives and can be read independently from each drive. This is fast and provides full redundancy, but the disk capacity required is doubled. The read performance can be up to twice as fast as a single drive because both drives can process the read request simultaneously. Write performance is nearly unchanged. RAID 1 is best for transaction processing, where many small I/Os are required.
RAID 2	Called redundancy through Hamming code. The data is bit-interleaved across groups of drives with some of the drives storing error correction codes. This provides full redundancy and more error correction capability, but is often slow due to the hardware overhead.
RAID 3	Called parallel transfer disks with parity. The data is bit-interleaved across groups of drives with only one drive dedicated to storing parity. This provides full redundancy and high transfer rates when large blocks of data are transmitted.
RAID 4	Called independent access array. The data is broken up into chunks and written across the drives with one drive dedicated to parity. This provides full redundancy, but the single parity drive causes a bottleneck when the parity has to be updated.
RAID 5	Called independent access array with rotating parity. The data is broken up into chunks and written across the drives. The parity for the stripes of data is also spread across all the drives, so no one drive is dedicated to parity.

Microsoft Navision Database Server is a program that requires high security and good disk performance. RAID 1, “mirroring,” best satisfies these criteria with its combination of high performance in read processing and very high reliability resulting from a constant, online backup of the database. RAID 1 is, however, the most expensive configuration because every megabyte of disk storage requires two megabytes of actual disk space. The Microsoft Navision Database Server program can handle up to 16 disks (32 disks in a mirrored configuration).

Memory and the Server

It is important for the server to have enough memory for all its tasks. There cannot be too much memory in your server. Because a Microsoft Navision Database Server has to serve several users and must put some memory aside for each user, the amount of memory should be proportional to the number of users. Normally a user "takes up" 2 to 8 MB of Microsoft Navision cache, depending on the task, so you can estimate the requirements for your configuration. Remember to include the memory requirements of the operating system (at least 128MB for Windows 2000 or Windows 2003), the Microsoft Navision Database Server program (1 MB), 50 kilobytes per user connected to the server, and add to this the memory requirements of other programs that are running. It is also important that the memory has a parity bit or is of the ECC (Error Correction Code) type.

The Network Adapter

Communication to and from the client passes through the network. If messages are to be delivered quickly, you must have a fast network adapter. This also ensures that the CPU use per network send/receive activity is minimal, which reduces the load on the CPU. The physical connection (the cabling) between the server and the clients must also be able to support the high speed.

The CPU

The speed of the CPU is also an important performance factor. It is the CPU that performs all the calculations involved in Microsoft Navision – the faster the CPU, the more calculations per second. It is also important to have as much level 2 cache in the system as possible. This increases the speed with which the CPU gets data from and saves data to RAM.

However, it should be noted that adding more hard disks gives a greater improvement in performance than increasing the speed of the CPU.

Hardware and Software Requirements

The following tables display the System Requirements for Microsoft Navision.

Microsoft Business Solutions – Microsoft Navision Client (Microsoft Business Solutions – Microsoft Navision Database Server)

Operating System	Microsoft® Windows 2000 Microsoft® Windows® XP Microsoft® Windows Server 2003
Hardware Resources	Hard disk space: About 250 MB (full installation incl. two language modules) About 110 MB (minimal installation incl. two language modules) Memory: 25 MB free memory with default cache settings CPU: See operating system requirements.

Microsoft Business Solutions – Microsoft Navision Client (Microsoft SQL Server Option for Microsoft Navision)

Operating System	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows Server 2003
Hardware Resources	Hard disk space: About 250 MB (full installation incl. two language modules) About 110 MB (minimal installation incl. two language modules) Memory: 60 MB free memory with default cache settings CPU: See operating system requirements.

Microsoft Business Solutions – Microsoft Navision Client (Single-User Installation)

Operating System	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows Server 2003
Hardware Resources	Hard disk space: About 250 MB (full installation incl. two language modules) Memory: 25 MB free memory with default cache settings CPU: See operating system requirements

Microsoft Business Solutions – Microsoft Navision Database Server

Operating System	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows Server 2003
Hardware Resources	Hard disk space: Space for database Temporary space equivalent to the largest primary index. Memory: Min 30 MB, Max 1 GB CPU: The server can take advantage of no more than 1 CPU. Network: WinSockets-compatible TCP/IP (if you are using the tcp protocol with the Microsoft Navision Server) NetBIOS-compatible LAN (if you are using the netb protocol with the Microsoft Navision Server) Lan network connection should be 100 mbit/sec or better.

Microsoft SQL Server Option for Microsoft Navision (server)

Operating System	See requirements for Microsoft SQL Server 2000.
Other Applications	Microsoft® SQL Server® 2000, SP 3

Microsoft Business Solutions – Microsoft Navision Application Server (Microsoft Navision Database Server)

Operating System	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows Server 2003
Hardware Resources	Hard disk space: 8 MB Memory: Min 35 MB free memory CPU: The server can take advantage of no more than 1 CPU. Network: WinSockets-compatible TCP/IP (if you are using the tcp protocol with the Microsoft Navision Server) NetBIOS-compatible LAN (if you are using the netb protocol with the Microsoft Navision Server) Lan network connection should be 100 mbit/sec or better.

Microsoft Business Solutions – Microsoft Navision Application Server (Microsoft SQL Server)

Operating System	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows Server 2003
Hardware Resources	Hard disk space: 8 MB Memory: Min 65 MB free memory CPU: The server can take advantage of no more than 1 CPU. Network: WinSockets-compatible TCP/IP (if you are using the tcp protocol with the Microsoft Navision Server) NetBIOS-compatible LAN (if you are using the netb protocol with the Microsoft Navision Server) Lan network connection should be 100 mbit/sec or better.

Commerce Portal

The following is a list of software requirements for Commerce Portal.
Requirements for the database server and client: please see requirements above.

Web Server(s)

Operating System	Microsoft Windows 2000 Server, SP3*
Other Applications	Microsoft Internet Information Server* (Part of Microsoft Windows 2000 Server) Secure Internet Information Services 5 Checklist Microsoft Message Queue* (Part of Microsoft Windows 2000 Server) Microsoft SQL Server 2000*, SP3 (used by Microsoft® Commerce Server) Microsoft Commerce Server* 2000/2002 Microsoft® Business Solutions–Navision® Application Server 3.70 Microsoft XML Parser 3.0, SP2 .NET Framework - only with Commerce Server 2002 (download framework)
Hardware Resources	Microsoft components requirements

NOTE: *These Microsoft products (*) must be in non-localized versions. They have only been tested and are only supported in US-English versions.*

Commerce Portal Client

Other Applications	Microsoft® Internet Explorer 5.5, SP2

Commerce Gateway

The following is a list of software requirements for Commerce Gateway. Requirements for the database server and client: please see requirements above.

Commerce Gateway Server(s)

Operating System	Microsoft Windows 2000 Server* (Only on the computer where Microsoft BizTalk Server 2000 is installed.)
Other Applications	Microsoft® BizTalk® Server* 2000/2002 Microsoft SQL Server 2000, SP3* (used by BizTalk Server) Microsoft® Business Solutions–Navision® Database Server 4.00 or Microsoft SQL Server Option for Microsoft Navision Attain 4.00 Microsoft Navision Application Server 4.00 Microsoft XML Parser 3.0, SP2 .NET Framework (download framework)
Hardware Resources	Microsoft components requirements

Automated Data Capture System

Other Applications	Microsoft Navision Application Server 4.00 Microsoft® XML3 (available with an installation of Microsoft Internet Explorer 5.5. (or later) Telnet or Microsoft Windows HyperTerminal, available as standard Windows components

Mail Merge

Other Applications	Microsoft Word 2000/ XP /2003

Outlook Client Integration

Other Applications	Microsoft® Outlook® 2000 / XP / 2003 with CDO (Collaboration Data Object) installed

E-mail logging for Microsoft Exchange

Other Applications	Microsoft® Exchange Server 5.5, SP4 or later
Other Applications	Microsoft Outlook 2000/ XP /2003 with CDO (Collaboration Data Object) installed

Import and Export Budget to/from Excel

Other Applications	Microsoft® Excel 2000 / 2003

***NOTE**: These system requirements are subject to change.*

2 – Tier and n – Tier Solutions

Due to the very flexible architecture of Microsoft Navision, it is possible to build configurations for virtually any environment - ranging from small installations with a few dedicated 2-tier clients to large environments with remote connectivity. Migration from one architecture to another does not have to be completed all at once, but can be done over a period of time, thereby easing implementation of new architectures.

Microsoft Navision is a client-server (2-Tier) solution. The standard client (32-bit) requires a LAN speed connection to the Database server. Access to the database over a WAN requires the use of Terminal Services under Windows 2000 or 2003. Microsoft Navision implements an n-tier solution with the application server to specifically integrate with external products. See, for example, Biztalk 2002 in Commerce Gateway and Commerce Server 2002 in the Commerce Portal solution. Partners and ISVs may develop their own custom solutions using the Microsoft Navision application server.

2–Tier Solution

Microsoft Navision is a two-tier solution application. It consists of a Database Management System (DBMS) that resides on the server and a Graphical User Interface (GUI) that resides on the client. You can also configure the client to be a stand-alone installation, which means that the client functions as a server and a single client in one.

2–Tier (Terminal Services Environment)

Terminal Services in Windows 2000 and 2003 offer the low cost, centrally managed environment of the traditional mainframe with terminals, but adds the familiarity of Microsoft Windows, the ease of use, and access to the multitude of applications available for the Windows platform.

Terminal Services in Windows 2000 and 2003 works by running the application on the central server system and only lets the client control the application (similar to remote control) and receives the output displayed at the client window. This reduces the bandwidth needed between the Windows client and the corporate ERP, but does not give any means of exploiting the Windows client capacity with respect to processing. It also introduces some difficulties with respect to local printing. Also important is the amount of concurrent clients Terminal Services for Windows can handle on any given hardware.

n–Tier (Microsoft Navision Application Server Environment)

The Microsoft Navision Application Server (NAS) is a middle-tier server that supports an n-tier architecture, which executes business logic without user intervention. Microsoft Navision Application Server allows you to communicate with external services.

Microsoft Navision Application Server acts as a client towards a database server and can act as a server for other services. When you start Microsoft Navision Application Server, it opens a predefined database and executes the C/AL code in a predefined codeunit.

Microsoft Navision Application Server can communicate with both Microsoft Navision Database Server and the SQL Server Option in the same way as a Microsoft Navision client.

Test Your Knowledge – Hardware and Software Requirements

1. Microsoft Navision can be used in _____ or _____ installations.
2. Besides having a powerful computer for the server, the other server aspects that require further consideration are _____ _____ _____ _____ _____ _____.
3. The acronym RAID stands for _____ _____ _____ _____ _____.
4. Microsoft Navision is a _____ solution.
5. The _____ _____ is a middle-tier server that supports an n-tier architecture.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 2: MICROSOFT NAVISION ARCHITECTURE

Overview

In this chapter, you learn about the architecture of Microsoft® Business Solutions–Navision®, the two server options that Microsoft Navision supports, and the Microsoft Navision security system.

It describes some of the most important differences between the two server options and explains some of the advantages that the Microsoft® SQL Server® Option has over Microsoft Navision Database Server.

It explains the different layers of security that exist in Microsoft Navision and how they work. There is also a brief a description of the functional areas that exist in Microsoft Navision. The Microsoft Navision license system is also explained.

The Microsoft Navision Client/Server Environment

Microsoft Navision is a two-tier application. It consists of a Database Management System (DBMS) that resides on the server and a Graphical User Interface (GUI) that resides on each client. You can also configure the client to be a stand-alone installation, which means that the client functions as a server and a single client in one.

Microsoft Navision has two database options: the standard Microsoft Navision database and SQL Server database. You can choose to use the database engine that is built into each client to run the standard Microsoft Navision database as a stand-alone installation. The database can also be run from a server, which allows many clients to connect simultaneously to the same database. The server runs on a designated computer that all the clients communicate with.

You can also install the SQL Server Option for Microsoft Navision as a stand-alone installation. To do this, you install the Microsoft SQL Server Desktop Engine with the client. During the installation, the demonstration database is attached to this local instance of SQL Server and is opened the first time you start Microsoft Navision.

***NOTE**: The Microsoft SQL Server Desktop Engine is only for use with Microsoft Navision Standard and is a small version of SQL Server with a database size limit of 2 GB.*

If you want to run the SQL Server Option as a client/server installation, you install SQL Server on the server computer. You must then install the Microsoft Navision client on all the client computers.

In the SQL Server Option, the client usually uses the TCP/IP protocol to communicate with SQL Server although it can also use an ODBC connection. In the Microsoft Navision Database Server option, the client generally uses the TCP/IP protocol to communicate with the server.

Once the server and clients are configured, the user does not need to worry about the way the server and clients work together. It appears seamless to the user.

Microsoft Navision Application Server is a middle-tier server that supports an n-tier architecture, which executes business logic without user intervention. Microsoft Navision Application Server allows you to communicate with external services.

Microsoft Navision Application Server acts as a client towards a database server and can act as a server for other services. When you start Microsoft Navision Application Server, it opens a predefined database and executes the C/AL code in a predefined codeunit.

Microsoft Navision Application Server can communicate with both Microsoft Navision Database Server and the SQL Server Option in the same way as a Microsoft Navision client. Microsoft Navision Application Server only supports Windows Authentication and automatically reconnects to the database server if there is a problem with the network.

The Client

The Microsoft Navision client is basically responsible for the user interface, but it actually does much more. The client can connect directly to a standard database file without going through the server. This is the stand-alone setup that was mentioned earlier. The client is also responsible for executing all the business logic. The client reads objects from the database and is also responsible for running the objects and controlling their behavior. Most of the Microsoft Navision application runs on the individual clients.

The Server

If the clients do most of the work, what is left for the server to do? The standard server component of Microsoft Navision :

- Controls the number of users that can connect to the database at one time.
- Controls access to the data through locking.
- Keeps track of all the read and write transactions performed by each user.
- Sends data to each client as requests are made.
- Performs all the key-based filtering and calculates the **SumIndexFields**.
- Caches data that can be requested again.

This is not a complete list and is only designed to give you an idea of what the server does. Microsoft SQL Server also does all of these things. One thing that the standard server does that SQL Server does not is keep track of the different versions of the same record that different users have accessed. We will discuss this in later chapters.

Together, the client and the database server provide a seamless solution.

The Server Options

In this section, you learn about some of the important differences between running Microsoft Navision on Microsoft Navision Database Server and on the SQL Server Option.

The Different Server Options

As stated earlier, Microsoft Navision can run on two different servers – Microsoft Navision Database Server and Microsoft SQL Server.

To the client these two server options look and perform exactly the same. However, there are some important differences between them including:

- The way you create a database.
- The backup facilities that are available.
- The ability to access the data in the database with third party tools.
- The way that SIFT™ works.
- Performance monitoring.
- Scalability.
- Multi-processor support.

We will elaborate on these topics in the following chapters.

Backup Features

The two server options offer very different backup and restore facilities.

Microsoft Navision Database Server

If you are running on Microsoft Navision Database Server, you can choose between two kinds of backup: a client based backup and a server-based backup.

The Microsoft Navision client based backup is initiated by clicking TOOLS→BACKUP.

The advantages of using the Microsoft Navision backup function are:

- The system tests the database for errors, so incorrect information is not copied to a backup.
- The data is compressed, so it takes up as little space as possible.
- The system calculates how much space the backup will use.
- You can keep working in Microsoft Navision while you are making a backup.

HotCopy
Microsoft Navision Database Server has a server based backup program called HotCopy. HotCopy is much faster than the client based backup facility. This program is installed with Microsoft Navision Database Server and is stored in the same directory as Microsoft Navision Database Server.

HotCopy can only be run from the server location and can only create backups on hard disks. You cannot make incremental or differential backups. You can make a backup of a database while clients are using it. The backups are file copies of the database and are not compressed.

SQL Server Option

Microsoft SQL Server supports four different types of backup. You should choose the type of backup you will be using carefully in order to ensure that you get the level of security you require.

The four types of backup are:

- Database backup – This makes a backup of the entire database.
- Transaction log backup – This makes a backup of the entire transaction log.
- Differential backup – This makes a backup of all committed entries since the last database backup.
- File and filegroup backup – This makes a backup of individual files or filegroups within a database.

These can be combined to form many different types of backup and restore procedures. This allows you to design a backup and restore strategy that fits your database needs.

For more information about SQL Server backup and restore strategies, consult Microsoft's SQL Server documentation.

You can also use the Microsoft Navision client based backup/restore tool when you are running on the SQL Server Option. However, the SQL Server backup/restore system is server-based and is, therefore, considerably faster than the Microsoft Navision backup/restore tool, which is client-based.

It is possible to restore a SQL Server backup of a Microsoft Navision database directly into SQL Server without using Microsoft Navision. You can also create a database directly in SQL Server without first having to create it in Microsoft Navision and then restore a SQL Server backup of a Microsoft Navision database directly into the database on SQL Server.

SQL Server allows you to make backups when the system is in use. With SQL Server, you can also automate many of your administrative tasks, including making backups. SQL Server allows you to establish a database maintenance plan (with the help of a wizard) that includes database optimization, integrity tests, and a backup plan.

One of the great advantages that SQL Server has over Microsoft Navision Database Server is its ability to record a transaction log. Transaction logs give SQL Server a roll forward capability that you can use to recover all the committed transactions that were carried out up to the point of failure. Roll forward is achieved by restoring your last database backup and applying all subsequent transaction log backups to recreate these transactions.
In such cases, only uncommitted work (incomplete transactions) will be lost, provided the active transaction log is also backed up and applied. The active transaction log also contains details of all uncommitted transactions. When you apply the active transaction log backup, SQL Server will roll back the uncommitted transactions.

SIFT

SumIndexField Technology (SIFT) has been designed to improve performance when carrying out certain activities such as calculating customer balances. In traditional database systems, this involves performing a series of database calls and calculations before arriving at a result.

Microsoft Navision Database Server

The power and efficiency of SIFT on Microsoft Navision Database Server makes calculating sums for numeric columns in tables extremely fast, even in tables that contain thousands of records. This powerful feature is used throughout the Microsoft Navision application.

Let us say you want the sum of all the values in the **Amount** field of a table. In a conventional system, the Database Management System (DBMS) is forced to access every record and add each value in the **Amount** field, a very time-consuming operation in a database with thousands of records. In Microsoft Navision, you create a FlowField, define the calculation formula of this FlowField to sum the **Amount** field, and then the DBMS only needs to retrieve the value from the SumIndexField.

SIFT has been built into the index structure used on Microsoft Navision Database Server, and the more SumIndexFields that are added, the larger the index becomes. However, the time used to maintain the accumulated sum for SumIndexFields is negligible due to a special index structure used in the DBMS.

SQL Server Option

The way that SIFT is implemented in the SQL Server is much different than the way it is implemented in Microsoft Navision Database Server.

In the SQL Server Option, the SIFT system has been reproduced by creating extra so-called SIFT tables on SQL Server. A SIFT table is a SQL Server table that is created and maintained automatically by Microsoft Navision and used to store pre-calculated totals based on the values that are stored in SumIndexFields in standard Microsoft Navision tables. A SIFT table is created for every standard Microsoft Navision table key that has at least one SumIndexField associated with it. No matter how many SumIndexFields are associated with a key, only one SIFT table is created for that key.

This means that every time you update a key or a SumIndexField in a Microsoft Navision table, all of the SIFT tables that are associated with this Microsoft Navision table must also be updated.

If you have a very dynamic Microsoft Navision table that is constantly having records inserted, modified and deleted, the SIFT tables that are associated with it will also have to be updated constantly. The SIFT tables can get very large, both because of the new records that are entered and because the records that are deleted from the Microsoft Navision table are not removed from the SIFT tables. This can also badly affect performance, particularly when the SIFT tables are queried to calculate sums.

This means that the number of SIFT tables that you create can affect performance.

Performance Monitoring

Another area in which the two server options differ is the way in which you monitor performance.

The Client Monitor is the most important tool that you can use when you want to monitor the performance of your application. This tool can be used with both server options. When you use the Client Monitor with the SQL Server Option, it contains some extra options and fields that give you more insight into how your application is performing.

The Client Monitor is an important tool for troubleshooting performance and locking problems. You can also use it to identify the worst server calls and to identify index and filter problems in the SQL Server Option. The Client Monitor and the Code Coverage tool now work closely together allowing you to easily identify, for example, the code that generated a particular server call.

Microsoft Navision also contains a debugger that you can use to refine functions that you write in C/AL code. The debugger can also be used with both server options.

When you are using the SQL Server Option, you can supplement these tools with the SQL Server Error Log. By enabling trace flags 1204 and 3605, you generate extra diagnostic error messages in the error log. These give you information about the type of locks that are involved in a deadlock.

For a detailed description of how to use these tools and of performance troubleshooting in general, see the manual Performance Troubleshooting Guide that is available on the Microsoft Navision Tools CD. The Tools CD also contains some extra tools that you can use for troubleshooting.

Other Differences

- Scalability:
  Another one of the main differences between the two server options is scalability. The SQL Server Option can support more simultaneous users than Microsoft Navision Database Server.
- Multi-Processors:
  Microsoft Navision Database Server does not make use of multi-processors while SQL Server does.
- Accessing the Database with Third Party Tools:
  It is much easier to access data in the database with third party tools when you are running on the SQL Server Option for Microsoft Navision.

The Microsoft Navision Security Model

This section introduces you to the Microsoft Navision security system. It explains the different layers of security that exist in Microsoft Navision and how they work.

There is also a brief a description of the functional areas that exist in Microsoft Navision. The Microsoft Navision license system is also explained.

Security

The Microsoft Navision security system allows you to control which objects (tables and so on) each individual user can access within each database. You can specify the type of access that each user has to these tables and records – whether he or she is able to read, modify, or enter data.

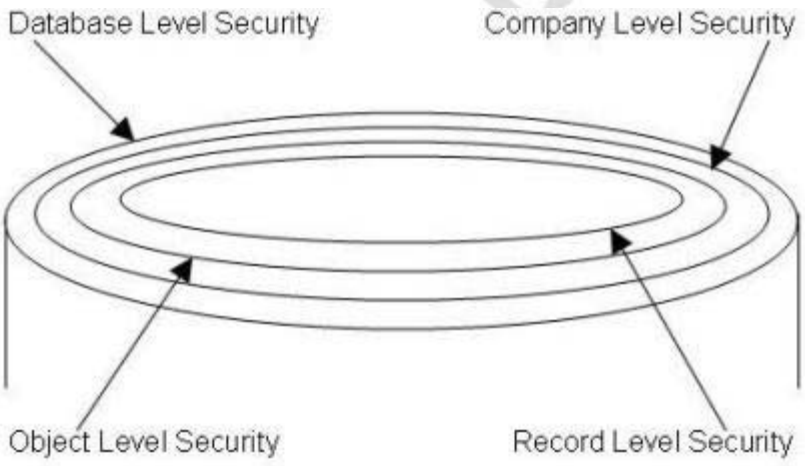
Furthermore, in the SQL Server Option you can specify which particular records that are stored in these tables each individual user is allowed to access. In other words, permissions can be allocated at both table level and at record level in the SQL Server Option for Microsoft Navision.

The Microsoft Navision security system contains information about the permissions that have been granted to each individual user who can access each particular database. This information includes the roles that the users have been assigned as well as any particular permissions that they have been granted as individual users.

The Microsoft Navision security system, even though it is a homogenous integrated system, can be said to consist of four different levels of security:

- Database Level Security
- Company Level Security
- Object Level Security
- Record Level Security

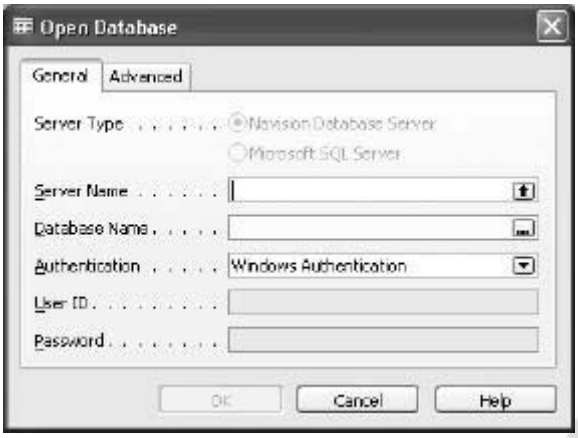
Graphically, these can be represented as the layers of an onion where the central layer is the records in the database:



Database Level Security

The first layer of security that you encounter when you open Microsoft Navision is database security. When you have started the program and try to open a database, your credentials are checked, and if you have not been granted permission to open the database you receive an error message informing you of this fact.

To open a database, click FILE→DATABASE→OPEN and the Open Database window appears:



In this window, you use the **AssistButtons** to locate the server that you want to access and the database that you want to open. If you are running on Microsoft Navision Database Server, you can select the server from a list of Microsoft Navision Database Server. If you are running on SQL Server, you can select the server from a list of SQL Servers.

In the **Authentication** field, you select the type of authentication that you want to use to verify you credentials and give you access to the database.

Microsoft Navision supports two kinds of authentication:

- Windows Authentication
- Database Authentication

The authentication that you must use depends on which kind of login you have been granted.

Windows Logins

Users are given a Windows login when you use Windows authentication to control access to Microsoft Navision. Windows logins in Microsoft Navision correspond to the Windows users and groups of the Windows domain. These are administered and listed in a separate table and window.

With Windows authentication, when a user tries to connect to a server and open a database, she does not have to supply a user ID or password. Microsoft Navision automatically asks Windows to confirm whether or not this user, who has already logged on to the network, has a valid Windows account and whether this account gives her the right to access this particular server.

If the user is allowed to access the server, Microsoft Navision checks to see if the user has been assigned a Windows login within Microsoft Navision. If the user has a Windows login, she will be granted access to the database.

The user will be granted access to Microsoft Navision and be given the permissions specified for that Windows user and those specified for any Windows groups of which she is a member.

If the user does not have a valid Windows account or if her account does not include permission to log on to the Microsoft Navision database, authentication fails, and the user receives an error.

Database Logins

Users are given a database login when they have their own user ID and password in Microsoft Navision. They must enter their user ID and password to access the database.

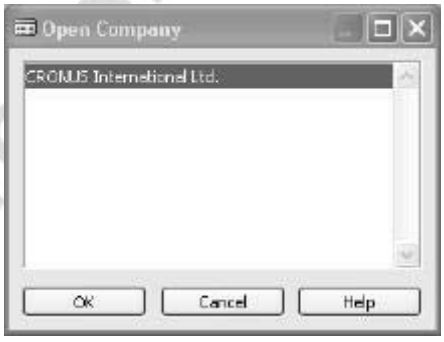
In the SQL Server Option, they must also have a login on SQL Server. SQL Server carries out its own authentication of the user's ID and password. SQL Server does this by checking whether a SQL Server login with this user's ID and password has been created. This login must first have been created by a SQL Server administrator, with a SQL Server tool. If a SQL Server login has not been set up, authentication fails and the user receives an error.

The user is granted access to the server after his login has been authenticated. Database security then validates the user's permissions by checking the database user accounts on the server. The permissions that the user has been granted to the various objects within the database, such as tables, are determined by the information contained in the user's database user account. It also contains information about any additional permissions that the user may have been granted to alter the database itself.

Company Level Security

After you have gained access to the database, you can open the company that you want to work with.

To open a company, click FILE→COMPANY→OPEN and the Open Company window appears:



This window lists all of the companies that have been created in the current database and that you have been given access to. A Microsoft Navision database can contain several companies. These companies can use their own tables, and they can also share some tables with each other.

Select the company that you want to open; click **OK** and the company opens. If there are companies in the database that you have not been given permissions to access, you will not be able to see them in this window.

Object Level Security

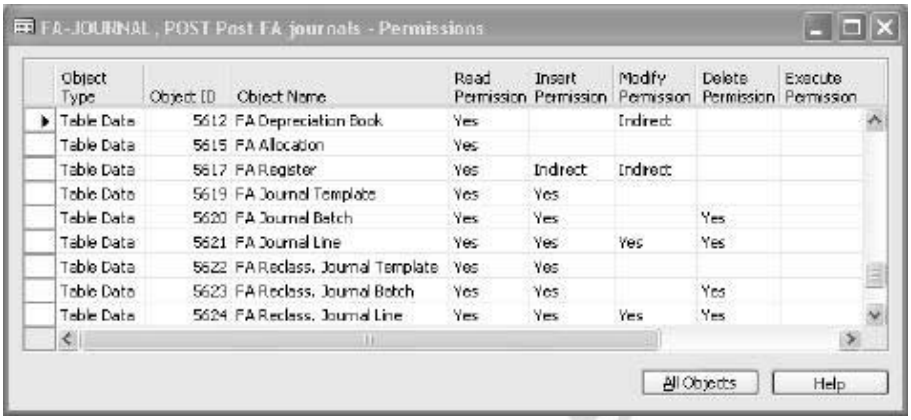
You have now opened a company, and your ability to work in it is still determined by the Microsoft Navision security system.

The Microsoft Navision security system consists of roles and permissions that you can assign to the users who have access to the company. The security roles in Microsoft Navision determine the access you have and the tasks you can perform on the objects that exist in the database.

The Microsoft Navision security system divides the database into the following objects:

Object	Description
Table Data	The actual data that is stored in the tables.
Table	The tables themselves.
Form	The forms that are used to view and enter data.
Report	The reports that are used to present the data.
Dataport	The dataports that are used to import and export data.
Codeunit	The codeunits that are used in the database.
XMLport	The XMLports that are used to import and export data in XML format.
MenuSuite	The object that contains the menus that are displayed in the Navigation Pane.
System	The system tables in the database that allow the user to make backups, change license file and so on.

The various roles that exist in Microsoft Navision determine the tasks that you can perform on these objects. The following picture illustrates how these permissions are allocated:



Object Type	Object ID	Object Name	Read Permission	Insert Permission	Modify Permission	Delete Permission	Execute Permission
Table Data	5612	FA Depreciation Book	Yes		Indirect		
Table Data	5615	FA Allocation	Yes				
Table Data	5617	FA Register	Yes	Indirect	Indirect		
Table Data	5619	FA Journal Template	Yes	Yes			
Table Data	5620	FA Journal Batch	Yes	Yes		Yes	
Table Data	5621	FA Journal Line	Yes	Yes	Yes	Yes	
Table Data	5622	FA Reclass. Journal Template	Yes	Yes			
Table Data	5623	FA Reclass. Journal Batch	Yes	Yes		Yes	
Table Data	5624	FA Reclass. Journal Line	Yes	Yes	Yes	Yes	

This picture shows some of the permissions that have been granted to the FA – Journal, Post role. As you can see, this role has been granted permission to perform various tasks on a long list of objects.

The permissions that a role can have on an object are:

Permission	Description
Read	You can read this object.
Insert	You can insert data into this object.
Modify	You can modify data in this object.
Delete	You can delete data from this object.
Execute	You can run this object.

If you have been granted permission to read a form, you can open it and view the data that it displays. If, however, you do not have write permission, you are not allowed to enter data into this form.

Sometimes, when you open a form, it displays information that is drawn from several tables. However, to access this form you must have permission to view all the data displayed by the form. You might not have permission to read directly from all the tables that the form uses. In this case, you must have what is known as indirect permission to read from the tables in question. Having indirect permission to a table means that you cannot open the table and read from it, but you can view the data it contains indirectly through another object, such as a form or report that you have direct permission to access.

Microsoft Navision comes with a number of standard predefined security roles. You can use these roles as they are or you can change them to suit your particular needs. You can also create your own security roles and give them the permissions that you want.

Record Level Security

Record level security is a system that allows you to limit the access that a user has to the data in a table by specifying that the user only has permission to access certain records in the table. Record level security is only available in the SQL Server Option for Microsoft Navision.

Record level security is implemented by applying security filters to the tables and the table data that a user has access to. You can specify, for example, that a user can only read the records that contain information about a particular customer. This also means that the user cannot access the records in this table that contains information about any other customer.

Things to Remember about the Microsoft Navision Security System

There are some important issues that you must remember about the Microsoft Navision security system:

- The Microsoft Navision security system is initiated when you create the first database login. Therefore, it is imperative that the first login you create is a database login. Furthermore, the first login you create must be that of a superuser. The superuser then owns and administers all access to this database from within Microsoft Navision.
- You can only grant permissions to other users that you already possess yourself. We, therefore, recommend that the user who administers security in Microsoft Navision should be a superuser. Until you create the first database login, anyone can carry out all the transactions they want in a Microsoft Navision database.
- One of the first things that the superuser should do is create logins for the other people who will have access to the database and grant the appropriate permissions to these users.

For a more detailed description of how to manage security in Microsoft Navision including detailed instructions on how to create logins and roles as well as how to grant and modify permissions, see the Installation manual for the server that you are using.

Business Areas and Granules

Now that you are familiar with the different levels of security that exist in Microsoft Navision, you should learn something about the business areas that Microsoft Navision contains and how your license file controls the access that you have to these areas.

Microsoft Navision is divided into a number of business areas. Each business area covers a large amount of functionality, and different customers will probably only need some of the functionality provided. To facilitate this, a Microsoft Navision license file consists of a number of granules. Each granule represents a smaller area of functionality.

This system makes it possible for each customer to purchase a license that only gives him and his employees access to exactly the functionality that they need.

Microsoft Navision contains the following business areas:

Business Areas	Functional Areas
Finance	General Ledger Receivables Payables Fixed Assets Inventory
Sales & Marketing	Sales Order Processing Marketing Inventory & Pricing Analysis & Reporting History
Purchase	Planning Order Processing Analysis & Reporting Inventory History
Warehouse	Orders & Contacts Planning & Education Goods Handling Inventory History
Manufacturing	Product Design Capacities Planning Execution Costing History
Service	Contact Management Planning & Dispatching Order Processing Periodic Activities
Human Resources	Employees Absence Registration Reports

Business Areas	Functional Areas
Resource Planning	Resources Jobs Reports Periodic Activities
IT Administrator	IT Administration Application Setup

Each of these business areas represents an important area of activity. However, even though you have a general ledger where you make all of your entries, you do not necessarily need all the functionality that is available in this area.

The following table lists the granules that are available in the General Ledger:

Granule – Name & Number	Description
Basic General Ledger (3,010)	You use this granule to set up a company and post to the general ledger. The granule provides you with the basic facilities necessary for setting up a company and posting to the general ledger: chart of accounts, general journals, VAT facilities, recurring journals, and source codes. It also includes facilities for internal and external reporting. The granule allows you to post and report in the company's base currency. If you also purchase the Multiple Currencies granule, you can post and report in an additional currency as well. The granule allows two languages from the beginning – the English US language and the native language for the particular country. The granule allows 1 instance of Microsoft Navision Application Server. This granule must always be included as part of the initial purchase of a solution because it includes one session and the first company. Requirements: The granule Microsoft Navision Version 3.XX

Granule – Name & Number	Description
Budgets (3,030)	This granule allows you to work with budgets in G/L accounts. Once you have created a budget, you can print a balance compared to the budget showing variances by percentages. You can work with several budgets. Budgets are normally entered per period for the relevant G/L accounts. You can create, copy, and work with any number of budgets at the same time. You can work with, for example, a 100% budget, a 110% budget, and so on. Requirements: Basic General Ledger
Account Schedules (3,040)	You use this granule for financials reporting. You can arrange reports based on the figures in the chart of accounts and budgets, but with a different arrangement of financial figures, texts, or details than in the chart of accounts. The Account Schedules granule is like a filter for the chart of accounts that enables you to choose the accounts that you want to include (or not include). You can also use it to change the order of the accounts or combine the figures in various ways, and you can set up which columns to print. In addition, it is possible to make simple calculations. Requirements: Basic General Ledger
Consolidation (3,050)	This granule enables you to consolidate companies in Microsoft Navision. The companies can come from one or from several different Microsoft Navision databases or from another type of file. There are facilities for imports and exports of financial information in the Consolidation granule. If the data used is retrieved from several Microsoft Navision solutions, the granule is only used in the parent company itself. Requirements: Basic General Ledger
Allocations (3,020)	This granule allows you to allocate general ledger entries to combinations of accounts, departments, and projects using allocation keys. The allocation keys can be based on amount, percentage or quantity. Allocations can be used for many purposes, for example, when allocating overhead (such as rent) to company profit centers. Requirements: Basic General Ledger

Granule – Name & Number	Description
Responsibility Centers (3,060)	With this granule, you can set up profit centers and/or cost centers. A company can sell items with specific prices and related to a responsibility center. The functionality provides the ability to tie a user to a responsibility center so that only sales and purchase documents related to the particular user are displayed. In addition, users get assistance with entering extra data, such as dimensions and location codes. Requirements: Multiple Locations
Basic XBRL (3,070)	With this granule, you can export documents from Microsoft Navision in XBRL format and import XBRL taxonomies into Microsoft Navision from the Internet, emails, or from other systems. XBRL is an XML-based specification that uses accepted financial reporting standards based upon standardized, underlying data tags. It is possible to map your general ledger to XBRL taxonomies, meaning that the same XBRL instance document can be used for various purposes, independent of the format required by the receiver of the document. This granule is an upgraded version of the existing Basic XBRL granule that we released with Microsoft Navision 3.10. The existing functionality has been revised to support the XBRL Specification 2 released by the XBRL Consortium in late 2001. Requirements: Basic General Ledger
Change Log (3,080)	This granule enables you to log user changes made to Microsoft Navision master data. It is possible to log all direct modifications a user makes to the data in the database, except changes to "working documents," such as journals and sales and purchase orders. The change log functionality makes it possible to get a chronological list of all changes to any field in any table (except the ones mentioned earlier) and to see who (which user ID) made the changes. Requirements: Basis General Ledger

The information contained in this table does not necessarily correspond with the granule definition that is currently enforced by Microsoft Business Solutions. Each of the other functional areas can be similarly broken down into areas of more specific functionality.

This means that your license file and the way that it is configured have an important influence on the functionality that is available to you in Microsoft Navision. Your license file and the granules that it contains determine which functional areas you have access to and which functions you can perform in these areas.

The advantage of this system is that the customer does not just pay for a general license to run the product, but instead only pays for the granules that she needs to run her business.

Test Your Knowledge – Microsoft Navision Architecture

1. Microsoft Navision has two database options. What are they?

2. Besides being responsible for the user interface, what else does the Microsoft Navision client do?

3. Even though the Microsoft Navision client does most of the work, what are some of the responsibilities of the server?

4. What are the different server options, and what are some of their differences?

5. Even though the Microsoft Navision security system is a homogenous integrated system, it can be said to consist of four different levels of security. What are they?

6. There are three important things to remember about the Microsoft Navision Security System. Can you remember them?

7. What are the differences among Business Areas, Functional Areas, and Granules?

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 3: MICROSOFT NAVISION CLIENT INSTALLATION

Overview

Before installing Microsoft® Business Solutions–Navision®, you should give some consideration to the kind of installation that you want and the equipment that it requires. When you have decided which type of installation you want and you understand the hardware and communication requirements, you can install the program.

Microsoft Navision comes with a standard setup that enables it to be used immediately. However, different installations may require small variations in the setup. You can easily implement these changes yourself. Once you have changed the settings, the program uses them until they are changed again. There is no simple formula that specifies the setup selections that different users need or the order in which they must be defined.

This chapter also describes the basic operations involved in working with a database, as well as license information.

Installing, Maintaining, and Removing Single-Users and Clients

You use the setup program to install, maintain, and remove Microsoft Navision. Microsoft Navision can be installed as either single-user installations or as client installations in a network.

The Microsoft Navision installation program also installs Commerce Portal automatically and allows you to determine whether or not the Commerce Gateway connectivity components are installed. These components must be installed with the Microsoft Navision client in order for it to connect with a Microsoft® BizTalk server.

For more information about installing the E-commerce products see the manual, Installation & System Management: E-Commerce Products.

Installing Microsoft Navision

Make sure that the operating system on which you will run Microsoft Navision (Microsoft® Windows® XP or Windows® 2000) is installed on the computer. If it is not, you will have to install it before you can install Microsoft Navision. If you are installing Microsoft Navision from a network drive, make sure that you are connected to the network server.

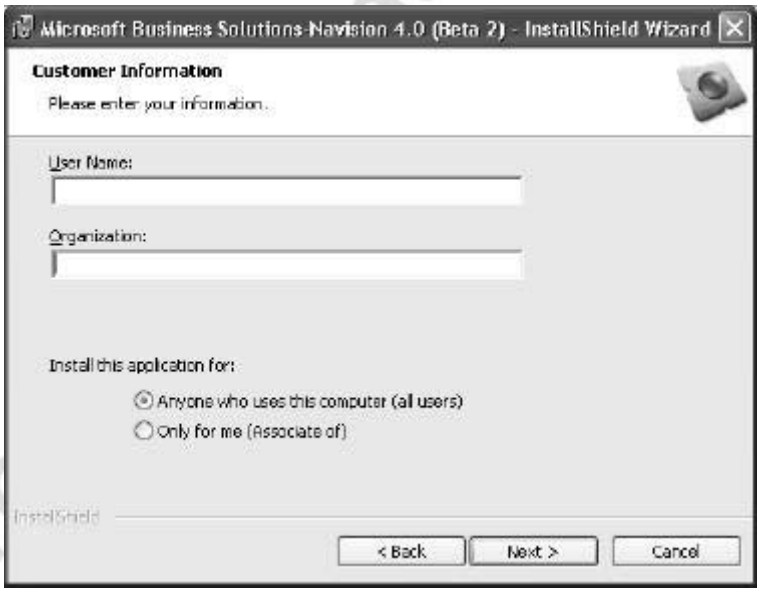
Microsoft Navision supports the Microsoft AutoPlay feature, so it is not necessary to click the **Start** button. A menu appears automatically when the CD is inserted. If you install from a network drive, type the path and name of the installation wizard on the network. Click **OK** to start the installation wizard. Windows XP and Windows 2000 allow you to use the **Add/Remove Programs** function under SETTINGS→CONTROL PANEL.

If you have an earlier version of the program installed on the computer, you can upgrade the old installation. For more information about upgrading, see the section titled "Upgrading an Old Installation."

The installation can be cancelled at any time. If you choose to cancel the installation, a dialog box appears asking you to confirm your decision. If you click **No**, the installation process will continue. If you click **Yes**, Microsoft Installer will perform a full rollback and restore the computer to the state it was in before the installation process began. This rollback functionality is a new feature provided by Microsoft Installer.

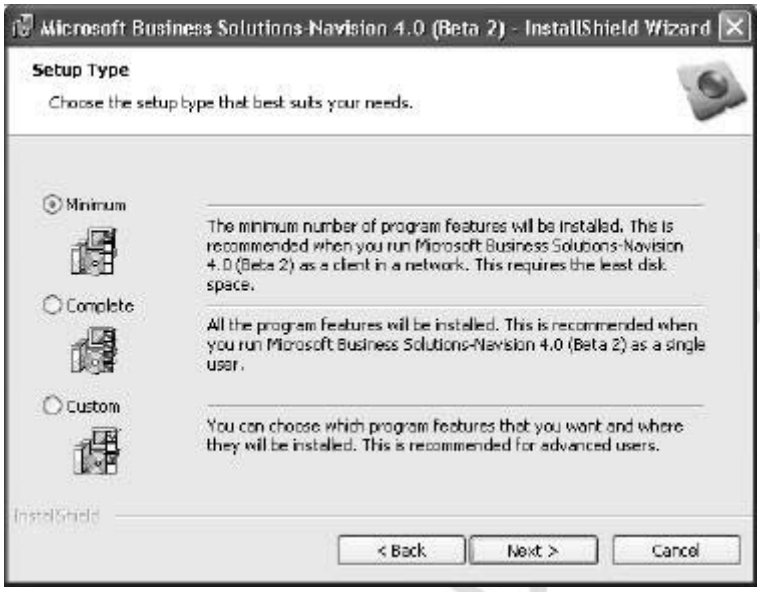
When the installation program starts the Welcome window appears informing you that you have started the installation wizard for Microsoft Navision.

If you do not want to continue with the installation, click **Cancel**. To continue, click **Next**, and the Customer Information window appears:



You can enter your name and the name of your organization in this window. You can also specify to whom this installation belongs. The installation will also continue if you leave these fields blank. You can choose between the person who installed it and any user who logs onto this computer. This determines who is allowed to see the installation and, therefore, able to modify or uninstall it. It does not determine who is able to use the program from this computer.

Click **Next** and the Setup Type window appears:



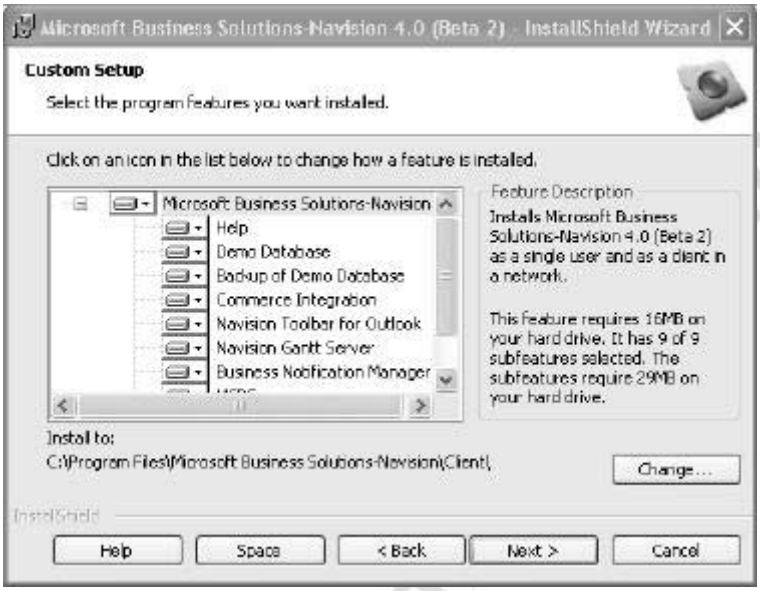
This window allows you to specify the type of installation that you want to install. You can choose between three types of installation:

- Minimum: this installation will only install a minimum of features (the demo database and the backup of the demo database will not be installed). This is the recommended installation, if you want to run Microsoft Navision as a client in a network. This installation includes the online Help. A client can be installed as a single-user, that is, with its own database, but this is not normally done because it takes up so much extra space on the client computer.
- Complete: this installation will install all of the program features except the Commerce Gateway connectivity components. This installation is recommended if you want to run Microsoft Navision in a *single-user* environment with a local database.
- Custom: this installation allows you to choose which program features will be installed. This option is only recommended for advanced users.

NOTE: *In a multi-user installation, you must always install one single-user installation in the network because the server package does not include the standard database that is part of the single-user installation. You need some of the data contained in the backup of the standard database when you create a new database. You can always turn a single-user into a client by allowing it to access the database on the server. The simplest thing to do is to install a single-user installation on the same computer as the server package. This procedure is described in "Importing a Database into Microsoft Navision Database Server".*

If you select a complete or a minimum installation, the installation runs automatically.

If you select **Custom,** the following window appears:



In the Custom Setup window, you can decide how and where each feature and subfeature will be installed. This window is divided into three sections:

- A feature selection area where you can select the individual features and specify how each feature will be installed.
- A feature description area that displays a short description of each feature as it is selected and an estimate of how much disk space the feature requires. It also gives you an estimate of the amount of space that its subfeatures require.
- An installation location area that tells you where each feature will be installed. You are able to change the location where the installation will be installed by clicking Change.

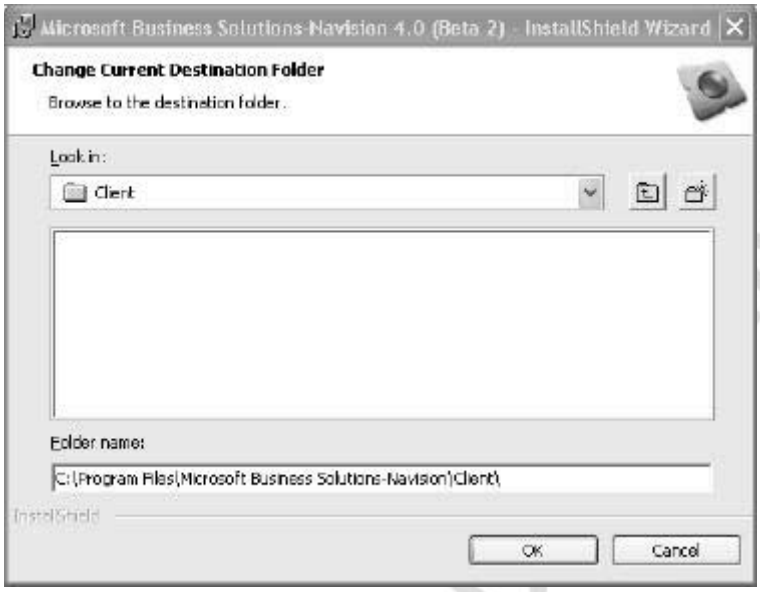
The Custom Setup window lists all the features that you can install:

- Help – the online Help for Microsoft Navision.
- Demo Database – A Microsoft Navision database that contains a demonstration company. This database will be opened automatically the first time you start Microsoft Navision.
- Backup of Demo Database – A Microsoft Navision backup of the demonstration database. You can restore this backup into a new database.

- Commerce Integration – The Commerce Gateway and Commerce Portal components. You must install these components if you want to run either Commerce Gateway or Commerce Portal. If you select this feature, the Microsoft .NET Framework is also installed. The .NET Framework is not removed when you uninstall the Microsoft Navision client. It is given an entry of its own in the Add or Remove Programs window, and you can uninstall it from there.
- Microsoft Navision Toolbar for Outlook – This feature creates a toolbar in Outlook that allows you to open a Microsoft Navision Contact or a Microsoft Navision To-do from the corresponding Outlook item.
- Microsoft Navision Gantt Server – An ActiveX component that allows production managers to plan shop floor production with the help of Gantt charts and update their schedules in Microsoft Navision.
- Business Notification Manager – This feature allows you to automatically send emails to your employees and business partners informing them of business events.
- MDAC – The Microsoft Data Access Components. These are operating system components that allow you to access data in the database with third-party tools.
- MSDE – The Microsoft SQL Server 2000 Desktop Engine. This is a small version of SQL Server and installing it allows you to run the SQL Server Option for Microsoft Navision as a stand-alone installation. This will be the local instance of SQL Server to which the demonstration database is attached.
- Furthermore, MSDE:
  - Is not installed if SQL Server is already installed on the client computer.
  - Is not removed when you uninstall the Microsoft Navision client. It is given an entry of its own in the Add or Remove Programs window, and you can uninstall it from there.
  - If you do not install MSDE with the client, you can install it later. However, if you then want to use the demonstration database, you must attach it manually.
  - All of these features are part of a complete installation.

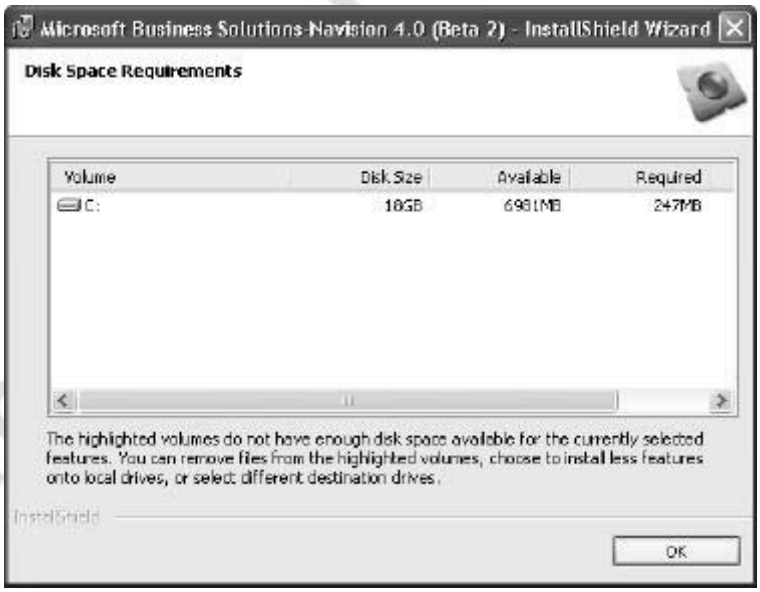
NOTE: *If you select one of the features, and the estimated space required is given as zero (or less than the original amount of space required), the file will not be copied to the target folder. This occurs because this feature or some of its subfeatures already exist in the target folder. This happens when you have had Microsoft Navision installed before and this feature was not removed when you uninstalled the program. No database files, database backup files or license files are removed when you uninstall Microsoft Navision.*

When you click **Change**, the following dialog box appears:

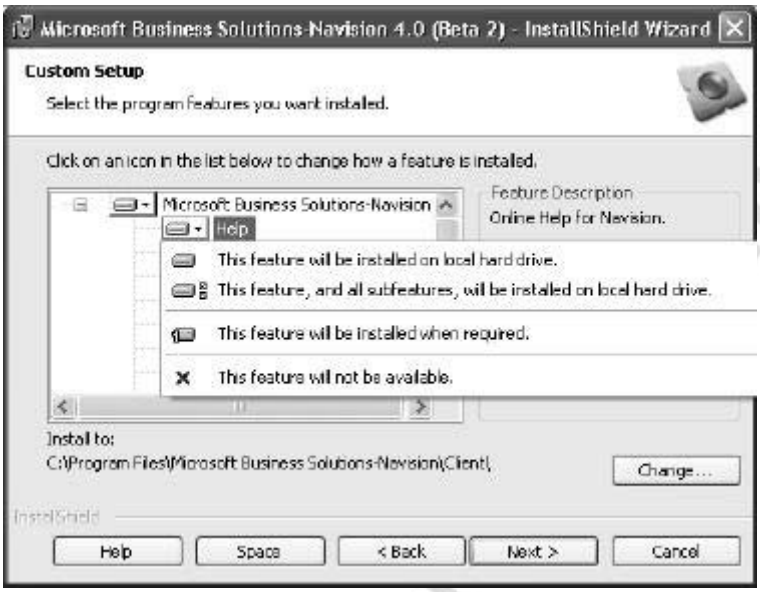


This window allows you to locate the folder you want or to create a new folder.

You can also get an overview of your disk configuration. In the Custom Setup window, click **Disk Space**, and a window appears informing you of how much space is available on your hard disk(s).



When you click one of the feature icons in the Custom Setup window, the following drop-down list appears displaying the options that are available for that feature:



The options are:

- This feature will be installed on your local hard drive.
- This feature and all of its subfeatures will be installed on your local hard drive.
- This feature and all of its subfeatures will be installed when it is required. Note that its subfeatures will automatically have the same option.
- The current feature and all of its subfeatures will not be installed.

If you click the **Help** button at the bottom of the Custom Setup window, a window containing a short explanation of the options appears.

The most common installations are Stand-alone installation and Client installation.

- A Stand-alone installation requires you to install the demo database. To do this, click the icon next to Demo Database, and select the installed (first) option in the drop-down menu. The backup of the Demo Database is optional. The online Help is also optional.
- In a normal client installation, you will install the online Help and ignore the other subfeatures.

The Ready to Install the Program window confirms that the installation wizard now has all the information that it requires to carry out the installation process.

Click **Install** to start the installation. The Installation Progress window allows you to monitor the installation process and displays an overall status message that tells you what kind of action is currently being performed. Below that is a more specific description of the particular action that is being carried out, for example, the name of the file that is currently being copied. Finally, there is a progress bar that shows you the status of the installation process.

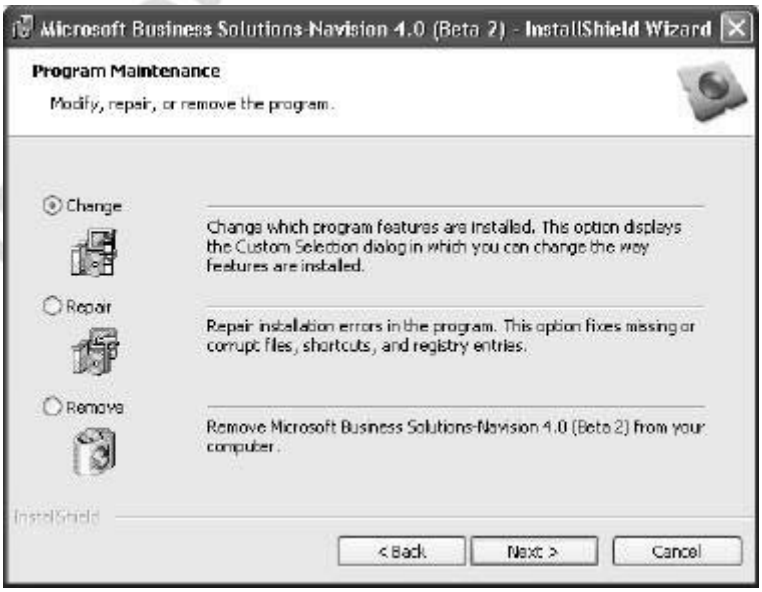
After a few minutes, the Installation Complete window appears. This window informs you that the installation has been completed successfully.

Maintaining Microsoft Navision

Microsoft Installer is also used for changing, repairing, and removing Microsoft Navision.

1. Open the Control Panel, and select **Add/Remove Programs**.
2. Select **Microsoft Navision**.
3. Click **Change** and the Installation Wizard opens.

   The Maintenance Welcome window is similar to the Installation Welcome window and informs the user that she can now change, repair, or remove the program.
4. Click **Next** in the Maintenance Welcome window, and the Program Maintenance window appears:



This window allows you to choose between changing, repairing, or removing the product installation.

- If you select **Change**, the Custom Setup window appears. You are now able to select the features and subfeatures that you want to install. You are also able to uninstall all the features and subfeatures. You are not able to change the target path of the installation. You can only change the features that have been installed.
- If you select **Repair**, the Installation Wizard updates, installs, or reinstalls any missing files, corrupt files, shortcuts, and registry entries. Windows Installer protects any license files, databases, and database backups and will, therefore, not overwrite these files.
- If you select **Remove**, a window appears prompting you to confirm that you want to uninstall the product.
  - If you choose to uninstall the product, a progress window similar to the Installation Progress window appears. You are able to cancel the un-installation at any time. If you do so, Microsoft Installer will perform a full rollback returning the computer to the state it was in before the un-installation process began.
  - When you remove the program, any databases or database backups that are stored locally will not be deleted. If you store a copy of your license file locally, it will not be deleted but the demonstration license file will be. If you want to remove these files, you will have to do it manually.
  - A progress window appears when you are changing, repairing or removing the program. When the process is finished an Installation Finished window appears informing you whether or not Microsoft Navision has been successfully changed, repaired, or removed.

NOTE: *We recommend that you make a backup of any license files, databases, and database backups that are stored locally before changing, repairing, or removing the client installation.*

The installation program creates a log file (an ASCII file called delfin.log that lists the changes made by the installation program) in the Microsoft Navision folder. If you need to uninstall Microsoft Navision manually, you can look in the log file to see what must be removed or changed.

Upgrading an Old Installation

If you have Microsoft Navision Financials 2.50 or an earlier version installed on the computer, the Installation Wizard will ask you to uninstall the old version before you can install Microsoft Navision.

If you have Microsoft Navision Financials 2.60 installed on the computer, the Installation Wizard will ask you if you want to upgrade the old installation or install Microsoft Navision without upgrading the old installation.

ATTENTION: *The upgrade program only upgrades the client installation and not the database. To upgrade your database, you must use the Upgrade Toolkit that is located on the product CD. We recommend that you do not upgrade your database without first consulting your local Microsoft Certified Business Solutions Partner.*

If you have Microsoft Navision Financials 2.60 installed on the computer, the Installation Upgrade Wizard will open. This wizard will guide you through the process of upgrading your installation. You can choose between three different types of upgrade:

- **Typical**: this upgrade will uninstall the old version of the program and install a new version. It will transfer any custom selections that you made in the earlier installation to the new installation. Any databases or database backups that are stored locally will not be deleted. If you store a copy of your license file locally, it will not be deleted.
- **Custom**: this upgrade will uninstall the old version of the program and install a new version. Any databases or database backups that are stored locally will not be deleted. If you store a copy of your license file locally, it will not be deleted. When the new version is being installed the installation program will stop at the **Custom Setup** window. This window allows you to specify which features are installed and how they are installed.
- **New**: this upgrade will install Microsoft Navision. It will not uninstall the old version of the program or make any changes to it.

For more information about upgrading to Microsoft Navision, see the Upgrade Toolkit documentation on the product CD or contact your local Microsoft Certified Business Solutions Partner.

System Setup

Microsoft Navision Installation

Microsoft Navision can be installed in several different configurations. The various program properties can then be used to determine the way the different types of installations perform. However, the program properties do not all apply to the different configurations and their effect can vary.

Single-User

In single-user installations, all the work is done on one computer and all the information (the database) is stored on this computer. Even if you purchase several single-user installations and run each on its own computer, the entire installation is still called single-user because the information is stored on each individual computer and not in one centrally located database.

Single-user installations are the easiest to set up and maintain.

The Setup File (zup file)

Microsoft Navision is able to remember the various settings that are used by different clients and single-users. This includes the server the client was connected to and the database and company the client was working with. It does this by saving these settings in a setup file. This setup file is called a zup file in Microsoft Navision. The different clients can all share the same zup file and thereby use the same settings, or they can all have individual zup files and thereby use their own customized settings.

If the network administrator has decided to implement roaming users, you will not have to utilize setup IDs. With roaming users, Windows will save the personal settings of each individual user on the server, including their zup file. This file will be called fin.zup and will contain Microsoft Navision information that is specific to that user.

If the network administrator has decided to implement local users, the zup file of each user will be saved on each user's local hard disk. Several users can also use the same computer and have each of their own zup files stored on the hard disk as long as each person has an individual window account. The zup file of each individual user is stored in C:\DOCUMENTS AND SETTINGS\USER NAME\APPLICATION DATA. All of these zup files will be called fin.zup unless you create individual setup IDs.

You only have to implement setup IDs if the users are using the same Windows account.

If you do not enter a setup ID, the program will use the setup file called fin.zup when you start the program. You will be prompted to save any modifications that you have made during the session when you close the program. These changes will be saved as fin.zup and will be recreated the next time the program is started.

When several users are using the same zup file, they will each be prompted to save the changes they have made in the fin.zup file when they close the program. The fin.zup file that was saved by the previous user who quit the program will be replaced.

If you delete the zup file on a client or single-user installation, Microsoft Navision will recreate the default zup file the next time the program is started.

You can also make a standard setup created for a particular type of user available to other users with similar needs. To do this, copy the appropriate setup file to the folder from which the user starts the program, and enter the setup ID (file name) in the **Target** field.

Alternatively, you can store the setup file on a common drive in the network (but here it can be overwritten by other users). If you choose to place the setup file in a folder other than the one containing the program files, you must remember to specify the entire path name after id=.

The setup ID is not the same as the user IDs in Microsoft Navision, but you can use the same names. In fact, it can be an advantage to do so because although you cannot see the name of the setup file in the program, you can always see the user ID on the status bar at the bottom of the program window.

The following table lists the different program properties and where they are located in this chapter:

Property	Described on
ID	Page 54
Server Name	Page 55
Database	Page 56
Company	Page 57
Windows Authentication	Page 57
Commit Cache	Page 58
Object Cache	Page 59
Net Type	Page 59
TempPath	Page 60
DB Test	Page 60
TestTarget	Page 61
Status Bar	Page 62
Close Form on Esc	Page 62
Marquee Full Selection	Page 63
Quick Find	Page 63
DB Read-Only	Page 64

Single-User and Client Setup Properties

This section describes all of the program properties that you can set for both clients and single-users.

ID – Saving the User Setup

Program Property	Purpose	Where Specified	Default Value	Value
ID (clients only)	Saves the program setup	By entering ID=alice in the Command line or the **Target** field	Fin	Name of ID (including path if ID file is not located in Microsoft Navision folder)

Each user in a Microsoft Navision multi-user installation can choose the setup of windows and program properties that he wants to use. Each user must have a unique ID in order for the program to be able to save and use the setup selections of the individual users. You can create a user setup by starting the program with an ID. The information about the users' setup will be stored under this ID. Here is an example in which the program starts with an ID called SUPER:

```
C:\PROGRAM FILES\MICROSOFT BUSINESS SOLUTIONS–MICROSOFT
NAVISION\CLIENT\FIN.EXE ID=SUPER
```

In a list of Microsoft Navision program files, you can see that each time you have started with a new setup ID, a file has been created that has the ID as the first part of the file name and .zup as the file name extension (for example, "super.zup"). This is called a setup file.

Returning to the Original Setup

You can always return to the standard setup file, fin.zup, by starting the program without specifying a setup ID.

If you have previously worked without a setup ID and made changes in the setup, the fin.zup file will contain these changes. If you do not want to use this modified fin.zup file but would prefer to return to the original starting point of the program, delete the fin.zup file, and start the program again without an ID. The program will create a new, clean setup file, named fin.zup.

For more information about zup files, see the section called The Setup File.

Server Name – Choosing the Server

Program Property	Purpose	Where Specified	Default Value	Value
Server Name	Specifies which server to connect to	Can be selected by clicking FILE→DATABASE→OPEN or by entering servername=My Server in the Command line or the **Target** field	None	Name of Server

This program property is used to specify the server that a particular client will connect to and does not apply in a single-user environment.

You can set up the connection to the server in the **Target** field by entering the name of the server after servername=.

You can connect to a server from within Microsoft Navision by clicking FILE→DATABASE→OPEN on the menu bar. For more information about connecting with a server and opening a database, see the section called Opening the Database.

Click FILE→DATABASE→INFORMATION, and click the **Database** tab to see which server you are currently connected to.

Database – Selecting a Database

Program Property	Purpose	Where Specified	Default Value	Value
Database	Specifies which database to open. A Microsoft Navision Database Server can only have one database open at a time.	FILE→DATABASE→OPEN or by entering database=My database in the Command line or the **Target** field. In a client/server installation this only works in combination with servername=	None	Name of database (including path if database is not located in Microsoft Navision folder)

The Database program property is used to make the program start with a particular database open. (The database must already exist.) In the **Target** field or on the command line that starts the program, type the name of the database immediately after database=.

To open a database from within Microsoft Navision:

1. Click FILE→DATABASE→OPEN.
2. In the window that appears, select the server and the database that you want to open.
3. Click FILE→DATABASE→INFORMATION to see which database is being used.

Microsoft Navision Database Server can only have one database open at a time. If a client wants to open a database other than the one that is currently open, she will have to wait until the first database is closed before she can open the desired database.

When you select a database for a client, you can also select the company you want to open automatically by using the Company program property.

Company – Selecting a Company

Program Property	Purpose	Where Specified	Default Value	Value
Company	Specifies which company to open.	FILE→COMPANY→OPEN or by entering company=CRONUS International Ltd. in the Command line or the **Target** field. In a client/server installation this only works in combination with servername= and database=	None	Company Name

With this program property, you can select the company that will open automatically when a client starts Microsoft Navision. You must also specify the server and the database that contain the company before you specify the company in the **Target** field.

Because Microsoft Navision can only have one database open at a time, you must connect to a server that already has the appropriate database open. If no other clients are using the system, you can select the server, database, and company that you want.

From within the program, you can select a company by clicking FILE→COMPANY→OPEN. You can also select a company from the list displayed at the bottom of the File menu. You can see the current company on the title bar of the program window.

Windows Authentication – Selecting the Authentication Mode

Program Property	Purpose	Where Specified	Default Value	Value
NTAuthentication	Specifies which type of authentication is to be used	FILE→DATABASE→OPEN or by entering ntauthentication=yes in the Command line or the **Target** field.	Yes	Yes/No

This program property is used to determine which type of authentication is to be used when logging on to a server and opening a database.

After selecting the server and the database in the Open Database window, you must:

1. Select the type of authentication that is to be used.
2. Enter your user ID and your password, if database server authentication is being used. If Windows authentication is being used, you do not have to enter a password or user ID.
3. Click **OK**.

Alternatively, you can enter "yes" or "no" after ntauthentication= in the **Target** field or on the command line that starts the program.

If you are using Windows authentication, Microsoft Navision will start, automatically connect to the server, and open the database that you have specified.

If Database authentication is being used, Microsoft Navision will start and prompt you to supply your user ID and password before connecting to the server and opening the database.

For more information about the types of authentication used in Microsoft Navision, see "Microsoft Navision Security."

Commit Cache – Writing the Cache

Program Property	Purpose	Where Specified	Default Value	Value
Commit Cache	Makes the program run faster	TOOLS→OPTIONS or by entering commitcache= in the Command line or the **Target** field.	No	Yes/No

The Commit Cache program property allows Microsoft Navision to postpone writing the information stored in cache on the server to the database until later. Storing this information in cache allows Microsoft Navision to work faster.

You must restart the program before any changes that you make to this parameter take effect.

Object Cache – Improving Response Times

Program Property	Purpose	Where Specified	Default Value	Value
Object Cache (KB) (clients only)	Makes the program run faster	TOOLS→OPTIONS or by entering objectcache=8000 in the Command line or the **Target** field.	8000 KB	More than 0 KB and less than 1,000,000 KB

The Object Cache property increases the speed of the program. Objects such as code, descriptions, and windows that will be used on the client computer are stored in the object cache. This means that the client computer only needs to retrieve these objects once from the server, and then they will be stored in the object cache. The client computer must have enough memory to store the objects while they are being used in order to benefit from the object cache.

Running out of object cache (that is, setting too small a value) does not cause any problems. The total size of all the objects used in the standard application is approximately 20 MB. If you have enough memory, set the object cache to 20 MB. The size of the most important objects, such as the table descriptions, is 1 MB. You should therefore, as a minimum, set the object cache to 1 MB. The upper limit is 1 GB.

Click FILE→DATABASE→INFORMATION to open the Database Information window and see how much space has been allocated in the **Object Cache (KB)** field. To change the amount of space allocated to the object cache, on the menu bar click TOOLS→OPTIONS, and enter the setting in the **Object Cache (KB)** field.

NetType – Selecting a Net Type

Program Property	Purpose	Where Specified	Default Value	Value
NetType	Permits choice of network protocol	TOOLS→OPTIONS or by entering NetType= in the Command line or the **Target** field.	TCP	Netb, TCP

To use Microsoft Navision in a network, you must select the network protocol that is used for communication between the server and the clients. There are two possible values:

- nettype=tcp (for TCP/IP)
- nettype=netb (for NetBIOS)

The same selection must be entered on all the client computers in the network as well as on the server. On the server, enter the net type you have selected in the **Target** field or on the command line after the start command. On the client computers, enter the selection in the **Target** field or click TOOLS→OPTIONS on the menu bar within Microsoft Navision.

To check the setting when you are using the program, click FILE→DATABASE→INFORMATION, and look at the **Connection** tab.

TempPath – Location of Temporary Working Files

Program Property	Purpose	Where Specified	Default Value	Value
TempPath	Specifies location of temporary working files created automatically	TOOLS→OPTIONS or by entering Temppath= in the Command line or the **Target** field.	Windows 2000: c:\Documents and Settings\User Name\Application Data\Local Settings\Temp Windows XP: <C:\Documents and Settings\User Name\Local Settings\Temp	Path to Temp Files

When Microsoft Navision is running, it creates a number of temporary files, which are automatically deleted when you close the program. As a default, the temporary files of each individual user are stored in the location listed in the previous table unless you specify a different working folder. If you do so, this working folder will be the default location. You can specify the working folder in the **Target** field or by clicking TOOLS→OPTIONS. You must specify the full path, including the drive and all the folders.

DB Test – Testing the Database

Program Property	Purpose	Where Specified	Default Value	Value
DB Test	Tests the database	FILE→DATABASE→ TEST or by entering dbtest=min in the Command line or the **Target** field.	None	Min., Normal or Max.

You can use this program property to test the consistency and integrity of the database. You can also run the test from within the program by clicking FILE→DATABASE→TEST. You can specify exactly what you want to test in the dialog box that appears.

When you enter the DB Test program property in the **Target** field, the database will be tested before the program opens. You can specify one of the following options: You can read about the extent of these tests, as well as how to create a customized version of the database test, in the "Microsoft Navision Database Administration Tools."

- dbtest=min
- dbtest=normal
- dbtest=max

TestTarget

Program Property	Purpose	Where Specified	Default Value	Value
TestTarget	To specify how error messages generated by the database test are managed.	FILE→DATABASE→ TEST or by entering testtarget=@screen in the Command line or the **Target** field.	@screen	@screen, @eventlog, filepath

You use this program property to specify how any error messages that are generated during a database test are managed. They can be displayed on the screen or stored in the Event Log or in a text file.

You can enter one of the following options:

- testtarget=@screen
- testtarget=@eventlog
- testtarget=filepath

You must enter the full path and the name of the text file. If you select event log, you can read the error messages that were generated during the database test in the Windows Event Viewer. If you select screen, the error messages will be displayed on the screen, and the database test will require interaction from the user if any errors are found. Selecting screen can make the database test quite time consuming.

Status Bar

Program Property	Purpose	Where Specified	Default Value	Value
TestTarget	Activated or deactivates the status bar	TOOLS→OPTIONS	Yes	Yes/No

On the menu bar, click TOOLS→OPTIONS, and in the Options window you can specify whether or not the status bar will be displayed at the bottom of the program window.

The status bar contains the following information:

- The complete name of the active field and its contents.
- The work date.
- The current user ID.
- Whether or not any filters have been placed on the data (FILTER appears).
- Whether or not you are about to create something NEW (an account, for example). Whether you are working in Insert (INS) or Overtype (OVR) mode.

When you make a visible change in the setup (such as making the status bar invisible), it is practical to use the ID program property and a setup file on your own computer. This makes the setup selections valid only for yourself. For more information, see the section called ID – Saving the User Setup. This property can only be adjusted from within Microsoft Navision.

Close Forms on Esc

Program Property	Purpose	Where Specified	Default Value	Value
Close Forms on Esc	Determines whether windows close when you press Esc	TOOLS→OPTIONS	Yes	Yes/No

Click TOOLS→OPTIONS on the menu bar, and you can choose whether or not the window you are working in will close when you press **Esc**.

It is practical to use the ID program property and have a setup file on your own computer if you change the setup. This makes the setup selections valid only for yourself. For more information, see the section called ID – Saving the User Setup.

This property can only be adjusted from within Microsoft Navision.

Marquee Full Selection

Program Property	Purpose	Where Specified	Default Value	Value
Marquee Full Selection	Determines how graphical objects are selected on the screen	TOOLS→OPTIONS	No	Yes/No

With this setting, you can choose whether graphical objects must be completely within the frame in order to be selected or whether it is sufficient for them just to touch the edges. This property is relevant for developers using the C/SIDE® development environment. To make this selection, on the menu bar click TOOLS→OPTIONS, and make your selection in the Options window.

This property can only be adjusted from within Microsoft Navision.

Quick Find

Program Property	Purpose	Where Specified	Default Value	Value
Quick Find	Quick search by letter in all windows	TOOLS→OPTIONS	Yes	Yes/No

This setting allows you to activate a quick search facility. When the Quick Find setting is enabled, you can search for an entry in any non-editable field, by typing a letter or number. You can also enter the entire name of the element you are looking for. When you enter a letter or number, the Find window opens automatically, and the first row that matches what you entered becomes the active row.

When the Quick Find property is disabled, you can open the Find window by clicking EDIT→FIND on the menu bar or by clicking **Find** on the toolbar.

This property can only be adjusted from within Microsoft Navision.

DB Read-Only

Program Property	Purpose	Where Specified	Default Value	Value
DB Read-Only	Determines that it is impossible to enter any data into the database.	By entering dbreadonly= in the Command line or the **Target** field.	No	Yes/No

This program property allows you to specify that the database has read access only. This prevents other users from entering data into the database.

Setting the Program Properties

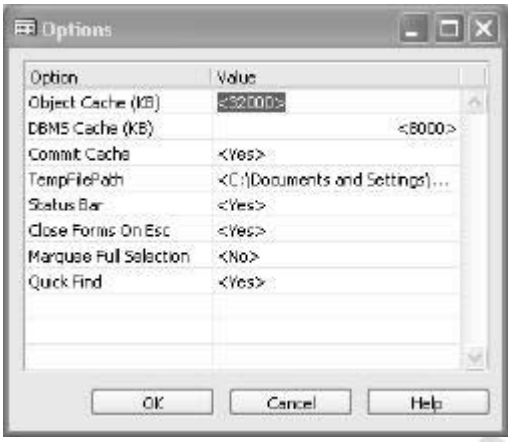
As explained in the previous sections, you can customize the system setup by changing the settings of the various program properties.

Some settings must be entered in the **Target** field of the Microsoft Navision Properties window (see the section called Connecting Automatically). You can also start a server from a command prompt. If the server has been installed as a service and the Startup Type option is set to automatic, the server will start automatically every time you start the operating system.

You can specify the program properties in any order. Enter them after the program's start command, separated by commas. The name of each program property is followed by an equal sign (=) and the value to which the property is to be set, for example:

```
D:\FIN\FIN.EXE SERVERNAME=MY SERVER, COMPANY=CRONUS INTERNATIONAL LTD., ID=ALICE
```

You can also set most of the program properties from the menu bar in the program – for example, by clicking FILE→COMPANY→OPEN (to set Company) and FILE→DATABASE→OPEN (to set Database). To see the properties that do not exist as menu items click, TOOLS→OPTIONS, and the Options window appears:



NOTE: *Any changes made in this window are saved in the setup file and will be valid the next time the program is opened. If you do not want users to be able to make "permanent" changes in these options, you can set default values in the command line of a batch file (called fin.bat, for example) with which the user starts the program.*

Connecting Automatically

Many of the properties that are described in the this chapter can be entered as command lines after the command prompt or included as program properties that are automatically set when you start Microsoft Navision.

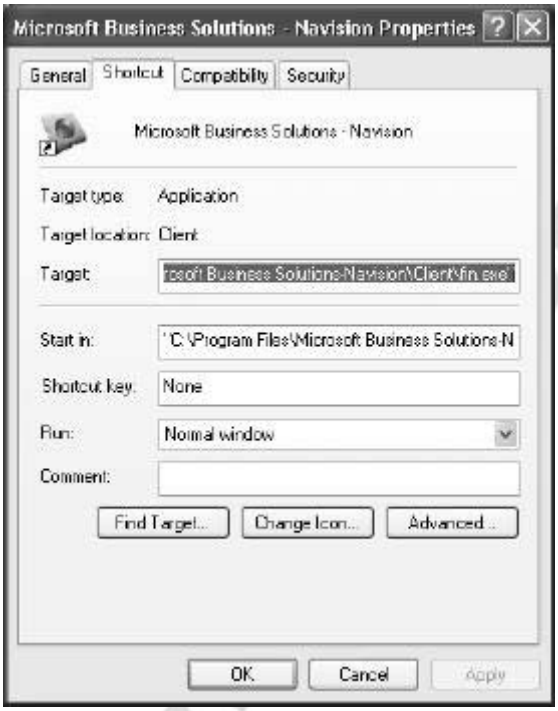
To carry out this procedure, you must have administrative rights on the computer. To set them as automatic program properties:

1. Open Windows Explorer.
2. Open the folder DOCUMENTS AND SETTINGS\ALL USERS\START MENU\ PROGRAMS\MICROSOFT BUSINESS SOLUTIONS – MICROSOFT NAVISION.
3. Notice that there are two identical Microsoft Navision icons on the right. You must select the icon for the Microsoft Navision executable that you are using.

IMPORTANT*: If you are using Windows XP and Windows 2000, you MUST delete these existing shortcuts and create new ones in order to gain access to the* ***Target*** *field. New shortcuts should be made from the Microsoft Navision Client folder and then copied into the same path noted in step 2.*

Also note that by copying the shortcut to the user specific folders, you can tailor the program properties to the individual user.

4. Click FILE→PROPERTIES, or right-click the Microsoft Navision icon and select Properties. The Microsoft Business Solutions–Navision Properties window appears:



5. Click the **Shortcut** tab. The **Target** field shows where Microsoft Navision is located. It contains the path for the start command fin.exe. After the start command, you can add other commands and settings for program properties. Here is an example:

SERVERNAME=MY SERVER, NETTYPE=TCP, COMPANY=CRONUS INTERNATIONAL LTD.

When you set the program properties in this way, the program performs certain tasks the next time it is opened. It will use TCP/IP to connect to the server called My Server and open the company called CRONUS International Ltd., if it exists in the database. If you do not enter the company parameter, you will have to open the company manually after the program starts.

You can change any of these selections while you are working. You can do this from within the program. For example, you can select a different database (provided it has already been created) or a different company, or you can create a new company. If you do not want a client to be able to do these things, you can set limits when you assign user permissions (by setting limits on the "system" object type). You can read about assigning user permissions in "Microsoft Navision Security."

Working with Databases

The Standard Database

You must have a database to be able to work with Microsoft Navision. When you install a singleuser installation, a standard database called database.fdb is automatically provided.

Using the Standard Database

You can use the standard database (database.fdb) in two ways: with a demonstration license (cronus.flf) or with your own license (fin.flf). The standard database contains a demonstration company called CRONUS International Ltd.

Using the Demonstration License File Cronus.flf

If you choose to work with the demonstration license file, cronus.flf, you have access to all the Microsoft Navision application areas and can test all the functions – including ones you have not purchased permissions for. The demonstration license file does, however, contain certain restrictions:

- Posting is only possible in the period November to February.
- You are only allowed to make 4000 write transactions in a database.
- The maximum number of companies is two.
- You can only have stand-alone installations or run Microsoft Navision Database Server on Windows 2000 and Windows XP.
- You can have a maximum of two sessions running at any one time.
- Any company name must start with CRONUS (written in capital letters). This ensures that it will be clearly identifiable as a demonstration company – and you will not accidentally create a "real" company with the wrong license file.

Using Your Own License File

If you work with your own license file (fin.flf), you can use only the functions for which you have purchased permissions. This means that you can see only the data for those functions – even in the demonstration company.

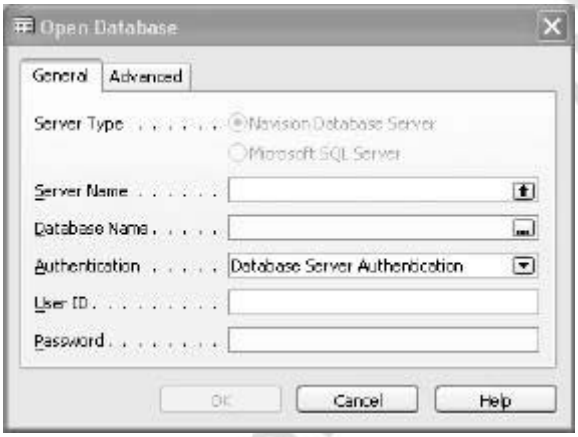
On the other hand, your own license file does not limit posting dates. You can also create as many companies in database.fdb as you have purchased permissions for. If you create more than one additional company in the database, however, you will no longer be able to use the license file cronus.flf because it allows only two companies in the database. Thus, you will lose the benefits of using cronus.flf. For example, you will not be able to see all the functions in the entire demonstration company. See the section called "License Files" for more information.

Opening the Database

On clients and single-user installations, you can open the database from within the program.

To open a database:

1. Click FILE→DATABASE→OPEN. The Open Database window appears:



2. In the **Database Name** field, enter the entire path and the name of the database manually or use the AssistButton. A standard Windows dialog box will appear, and you can use it to locate the desired database with the file extension .fdb.
3. In the **Authentication** field, specify the type of authentication you require. You can choose between Windows Authentication and Database Server Authentication. You can use the AssistButton to select the authentication type from a list.
4. You must enter a user ID and password if you are using database server authentication. If you are using Windows authentication, you do not have to enter a user ID and password.
5. Click **OK** to open the database.

You can specify the network type that will be used when you connect to the server. In the Open Database window, click the **Advanced** tab. However, it is not usually necessary to change the network type from the default setting.

When you have opened the database, you can open a company by clicking FILE→COMPANY→OPEN, or you can add a new company by clicking FILE→COMPANY→NEW. You can only have one company open at a time. However, you can open and close companies regardless of whether other users are working with them.

NOTE: *Microsoft Navision will automatically open the database and company that you were last working on when you reopen the program.*

Closing a Database

If you want to close a database (for example, before opening another), you can click FILE→DATABASE→CLOSE. However, you do not have to follow this procedure; Microsoft Navision will save all data and close everything down correctly when you select another database or quit the program. There can never be more than one database open at a time, but you can choose to close the database as an extra safety precaution if you want to delete a database or do something similar.

License Files

The Microsoft Navision installation comes with a demonstration license file, CRONUS.flf. The demonstration license file allows you to use the standard Microsoft Navision program as a stand-alone application and gives you access to the demonstration company that is part of the accompanying standard database, database.fdb. To start working with Microsoft Navision, you will need a license file that contains permissions for the desired application areas and functions. This license file will be supplied by your Microsoft Certified Business Solutions Partner.

You can use your license file to work in the demonstration company and in your own companies, but your permissions will be limited (even in the demonstration company) to those provided by the license file. Your license file is not subject to the same restrictions as the demonstration license file. These restrictions are listed in the section “Using the Demonstration License File Cronus.flf.”

Your license file is always named fin.flf and is uploaded to the server during the server installation. The clients automatically work with the same license file as the server they are connected to.

Importing, Exporting and Changing License Files

On the menu bar, click TOOLS→LICENSE INFORMATION. The License Information window appears:

The information displayed includes the license number of the current license file, the name of the owner, and the functionality that the owner has purchased (along with any expiration dates). If at any time you want information about the current license file, you can open this window. The buttons at the bottom of the window allow you to import and export license files and to temporarily change the license that you are using.

The license information that is displayed in this window will always be taken from the license information that is stored on the server, unless you have changed your license temporarily with the Change facility.

Importing a License File

In the License Information window, click **Import**, to use a different license file. The Import License File dialog box appears. Locate and select the license file that you want, and then click **Open**. The program will then import the license file into the Microsoft Navision folder on your computer, and it will be called fin.flf. The license file will automatically replace any other file called fin.flf without asking you to confirm that you want it to do so.

This new license file will be the active license file the next time you open Microsoft Navision. When you connect to a server, the license stored there will become the active license.

Exporting a License File

Click **Export** to export a copy of your license file, for example, to a disk. The Export License File appears. This is a standard windows dialog box.

Temporarily Changing the License File

If, for example, you are a Microsoft Certified Business Solutions Partner representative visiting a customer, you may want to change the license file temporarily. To do so, insert the disk containing the license file and click **Change**. The Change License File dialog box appears. Select the license file to be read into the system. The information contained in it will be transferred to the client when you click **Open**. When you access any servers, this temporary license information will continue to be used instead of the license information stored on the servers. The server license will be reinstated when Microsoft Navision is closed and opened again.

Microsoft Navision will warn you before your license expires. If you fail to notice the warning and the license on the server expires, you will not be able to access the server. However, you can use the Change facility to gain access to the server by using an alternative license file.

When you receive your new license file, you should copy it to the server.

Ensuring that You are Using a Valid Microsoft Navision License

You may want to verify that you are using a valid Microsoft Navision license that has been issued to your company.

To check the validity of your license file:

1. Click HELP→ABOUT MICROSOFT NAVISION and the About Microsoft Navision window appears.
2. Click **Check your license information**, and your web browser opens a Web page that will help you check the validity of your Microsoft Navision license.

Test Your Knowledge – Microsoft Navision Client Installation

True	False	
❑	❑	1. Microsoft Navision comes with a standard setup that enables it to be used immediately.
❑	❑	2. The three types of installations are: Minimum, Maximum, and Custom.
❑	❑	3. If you have Microsoft Navision 2.6 or an earlier version installed on the computer, the Installation Wizard will ask you to uninstall the old version before you can install Microsoft Navision.
❑	❑	4. The System Setup file, or ZUP file, remembers the various settings that are used by different clients and single-users.
❑	❑	5. You can customize the system setup by changing the settings of the various program properties.
❑	❑	6. If ntauthentication=no on the Target line, Microsoft Navision will start and automatically connect to the server and open the database specified.
❑	❑	7. Program properties must be entered in a specific order.
❑	❑	8. Microsoft Navision will automatically open the database and company that you were last working on when you reopen the program.
❑	❑	9. The Change License functionality is a permanent change to which license is used.
❑	❑	10. If the license on the server expires, you will not be able to access the server.

Quick Interaction: Lessons Learned

Take a moment to write down 3 Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 4: MICROSOFT NAVISION DATABASE SERVER

Overview

In a multiuser installation, many users share common information that is stored in one or more databases on a server. The computers that work with the data are called clients, and the way the server and the computers work together is called a client/server installation.

Chapter 4 covered how to complete the Microsoft® Business Solutions–Navision® Client installation. This chapter describes the Microsoft Navision Database Server installation, setup, and some basic operations involved in working with a database.

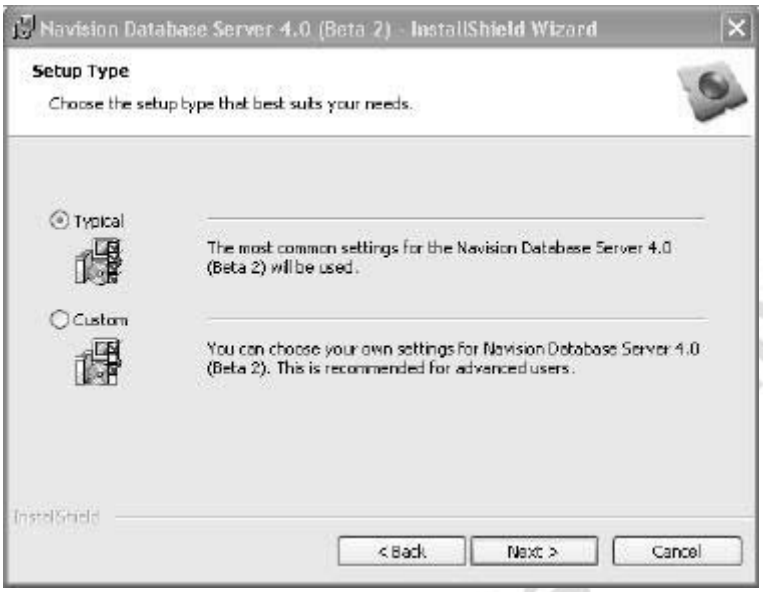
Installing Microsoft Navision Database Server

The installation wizard for Microsoft® Business Solutions–Navision® Database Server is like the installation wizard for Microsoft Navision.

The installation process starts with a window informing you that the installation wizard has been initiated. After the Welcome window, the Customer Information window appears.

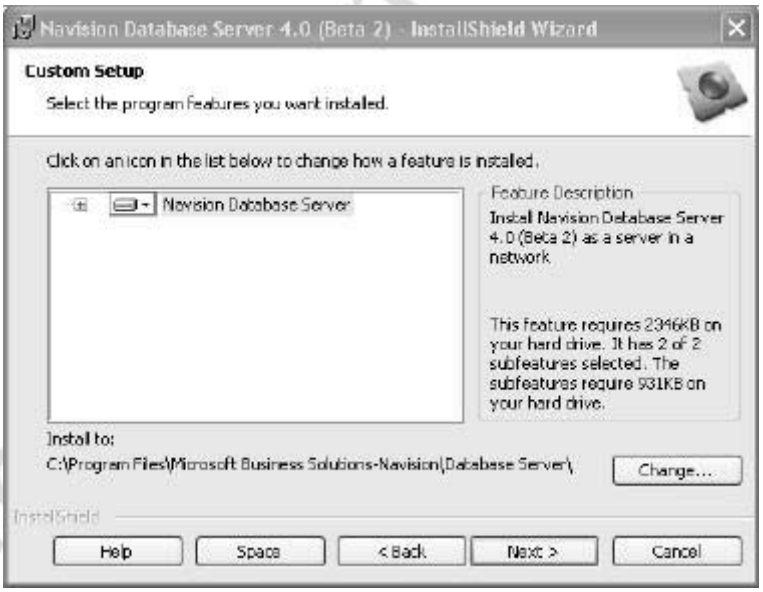
You can enter your name and the name of your organization in this window. You can also specify to whom this installation belongs. You can choose between the person who installed it and any user who logs onto this computer. This determines who is allowed to see the installation and, therefore, able to uninstall it. It does not determine who is able to log on to the server via the network.

Click **Next** and the Setup Type window appears:



In the Setup Type window, you can choose between a typical installation and a custom installation. A custom installation allows you to specify which features will be installed and how they will be installed.

If you select **Custom**, the following window appears:



In the Custom Setup window, you can select the features that you want to install. This is a standard Windows Installer dialog box where you can select to install or not to install the Microsoft Navision Database Server Snap-In for Microsoft Management Console. The snap-in allows you to supervise Microsoft Navision Database Servers across the domain.

For more information about the Snap-In for Microsoft Management Console, see the section Microsoft Navision Database Server Manager.

If you want to install Microsoft Navision Database Server in a different folder, click **Change**. The Change Current Destination Folder dialog box will appear.

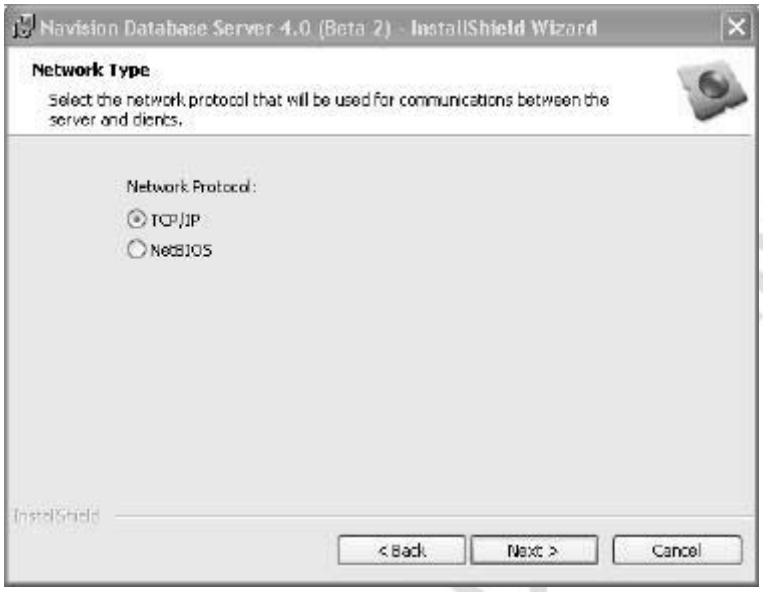
If you click **Disk Space** in the Custom Setup window, the Disk Space Requirements window will appear informing you of how much space is available on the various drives to which you have access.

Click **Next** in the Custom Setup window, and the Server Name window appears:

You can type in a name for the server. By default it will select the name that the computer has in the network. This name must not contain any spaces.

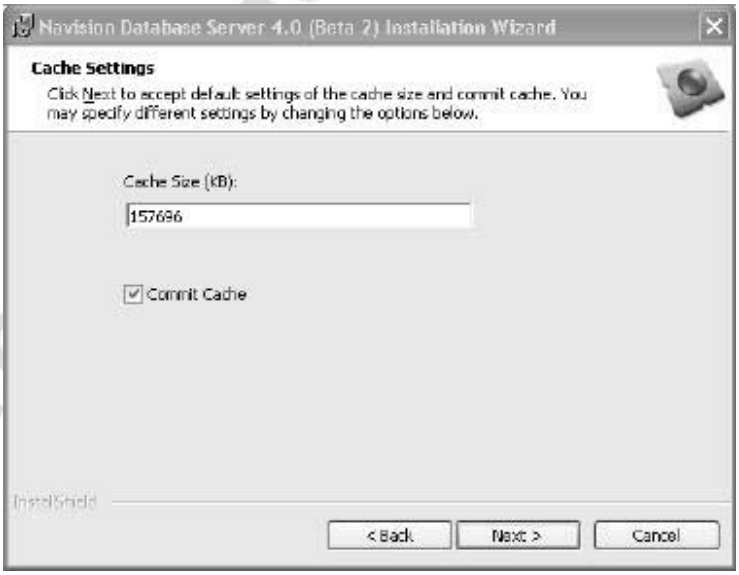
If you select Typical in the Setup Type window, the installation will use the computer name as the server name. If this name exists, it will insert #1 at the end of the computer name. If this name also exists, it will try with #2, and so on. That means if the computer name is PC0515 and this server name exists, the installation will try PC0515#1.

Click **Next** after you have named the server. The Network Type window appears:



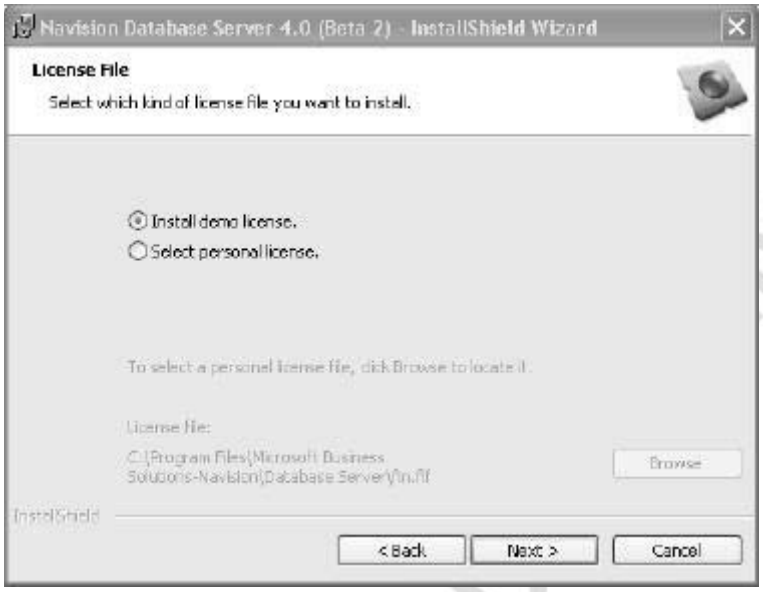
In the Network Type window, you can select which network protocol will be used for communication between the clients and the server (TCP/IP or NetBIOS). If you have chosen Typical in the Setup Type window, TCP/IP will be selected as the network protocol. TCP/IP is the default setting.

Click **Next** after selecting the network protocol. The Cache Settings window appears:



This window allows you to determine the amount of space that will be reserved on the server used for both the Cache and the Commit Cache. You can accept the default settings or adjust them to suit your needs.

Click **Next** after specifying the cache settings. The License File window appears:

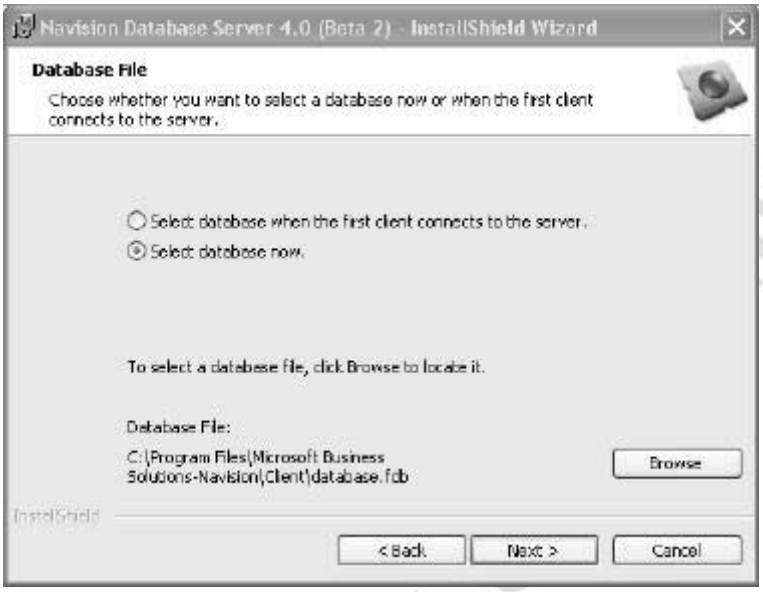


In the License File window, you specify which kind of license file you want to install. You can choose between installing the demo license file or your personal license file. Install demo license is the default value.

When you choose the Select personal license option, the **Browse** button and the **License file** field will be enabled. You are now able to use the **Browse** button to locate your license file. If you have chosen to install a Typical installation in the Setup Type window, the **Install demo license** option is automatically selected.

For more information about license files, see the section called “License Files.”

Click **Next** after you have selected the license file you want to install, and the Database File window appears:



This window allows you to specify whether you want to select a database now or wait until the first client connects to the server. If you choose to select the database now, the **Browse** button and the **Database File** field become enabled, and you can use them to locate the database that you want to be opened when Microsoft Navision is started.

Click **Next**, and the installation process will continue. It will display a progress bar that allows you to monitor the installation. The installation program will also inform you when the installation is complete and whether it has been successful or not.

Publishing Microsoft Navision Database Server in a Network

In order for other server and client computers to be able to identify and connect with your Microsoft Navision Database Server, you must publish it in the network. Before you can publish a Microsoft Navision Database Server in the network, you must add a definition of a Microsoft Navision service connection point to the Active Directory. Microsoft Navision Database Server uses this service connection point to publish itself in the network and the clients use this information to locate the server and to find out how to connect to it. Only servers that have been installed as a service can be published in the network.

To add a definition of a Microsoft Navision Database Server to the Active Directory, you must add a Schema Extension to Active Directory. This must only be done once for each forest of domains and domain trees, regardless of how many Microsoft Navision Database Servers are in that forest.

When you install Microsoft Navision Database Server, a file called schemaXt.exe is copied into the folder that contains Microsoft Navision Database Server. Run this file to add a definition of a Microsoft Navision service connection point to the Active Directory. Starting up a Microsoft Navision Database Server (running as a service) will publish a service connection point using the definition that has been added to Active Directory.

Maintaining Microsoft Navision Database Server

The Maintenance Setup Type window for Microsoft Navision Database Server differs from that of Microsoft Navision in that you are not able to modify the server installation. If you need to change the server name, for example, you must uninstall the program first and then install it again and give it the new server name.

If you install the server again in the same folder (for example, if you install an update), any existing database files, database backup files, and license files will not be overwritten by the new installation. This means that if you want a new database or a new license file to be installed when you reinstall the program, you must delete these two existing files from the Microsoft Navision folder before you start.

NOTE: *We recommend that you make a backup of any license files, databases, and database backups that are stored on the server before modifying, repairing, or removing the server installation.*

Importing a Database into Microsoft Navision Database Server

In a multiuser installation, in addition to installing one or more clients, you must install the server program and set up a database on the server. It is easier to start by setting up the database.

The server program does not come with a database. If you already have a Microsoft Navision database, copy it to the same computer that your server will be installed on. If you do not already have a database, you can use the standard database, database.fdb, or a copy of it. For more information, see the section called "Using the Standard Database."

You can also create a new database. To do this, you need some of the information in the standard database. A single-user version must, therefore, be installed somewhere in the network so that you can transfer the database that comes with it to the server.

To create a new database before you install the server:

1. Install a Microsoft Navision single-user.
   - Remember to select the Complete Installation option. (See "Installing, Maintaining and Removing Single-Users and Clients" in Chapter 3.)
   - The standard database will be installed as part of the complete installation and comes with a demonstration company. It is not a good idea to leave the demonstration company in your working database. If, however, you want to leave it in, skip to step 6.
2. Click FILE→COMPANY→OPEN to open the demonstration company.
3. Click FILE→COMPANY→DELETE.
4. Click **Yes** when you are asked (twice) whether you want to delete CRONUS International Ltd.
5. Close the single-user program.
6. Start the Microsoft Navision Database Server Setup program.

Starting the Server

When installed as a service the Microsoft Navision Database Server starts automatically every time you start the server operating system.

NOTE: *Remember to make a full backup of the database before deleting it from the hard disk. For more information, see "Microsoft Navision Database Administration Tools."*

Running more than one Server

The following sections describe the possibilities for running more than one server with the same database or with different databases.

Two or More Servers, Same Database

It is possible to run two or more Microsoft Navision Database Servers with the same database provided that:

- Each server has a different server name.
- All the servers are run from the same folder.

With this configuration you could, for example, have one server running TCP/IP and another one running NetBIOS on the same database. See NetType – Selecting a Net Type for a description of TCP/IP and NetBIOS.

Two or More Servers, Different Databases

It is possible to run two or more Microsoft Navision Database Servers with different databases provided that:

- each server has a different server name.
- the server programs are located in different folders.

This means that to use two databases on the same computer, you must install the server program twice in two different folders.

The following examples describe how to configure your installation for running two servers with different databases, using TCP/IP and using NetBIOS.

Two Servers on the Same Computer, Both Using TCP/IP

To run two servers, both running TCP/IP on the same computer, follow the procedure outlined below to configure the computer correctly.

Install the Microsoft Navision Database Server in two different folders as follows:

1. Install the first server as described previously.
2. Make a copy of the folder where Microsoft Navision Database Server was just installed and rename it. You must do this because you cannot install two versions of Microsoft Navision Database Server on the same computer.
3. Select a unique server name for the second server.

Once the two servers have been installed, proceed as follows:

1. Locate the SERVICES file. On Windows 2000 Server and on Windows Server 2003, the services file is stored in:

   C:WINDOWS\SYSTEM32\DRIVERS\ETC\

2. Edit the SERVICES file to include statements such as:

   SERVERNAME1 2407/TCP
   SERVERNAME2 2408/TCP

   The server names shown should be replaced with your actual server names.

3. Start the first server in the first folder, using the following parameters:

   SERVER SERVERNAME=SERVERNAME1, NETTYPE=TCP, CACHE=XXX, COMMITCACHE=YES, DATABASE=AAAA

4. Start the second server in the second folder, using the following parameters:

   SERVER SERVERNAME=SERVERNAME2, NETTYPE=TCP, CACHE=XXX, COMMITCACHE=YES, DATABASE=BBBB

   In these examples, xxx represents the cache size, and AAAA and BBBB are the names of the databases. Remember to write out the full path for each database file.

5. Edit the SERVICES file on the client as follows:

   SERVERNAME1 2407/TCP
   SERVERNAME2 2408/TCP

6. Start the first client, using the following parameters:

   FIN SERVERNAME=SERVERNAME1, NETTYPE=TCP

7. Start the second client by using the following parameters:

   FIN SERVERNAME=SERVERNAME2, NETTYPE=TCP

You must also edit the HOST file on the clients. It is located in the same folder as the SERVICES file. Follow the same procedure when editing the HOST file.

The system should now be up and running.

Two Servers on the Same Computer, Both Using NetBIOS

When you use NetBIOS, you do not have to configure the HOSTS and SERVICES files.

1. Install the two servers in two different folders.
2. Start the first server in the first folder using the following parameters:

   SERVER SERVERNAME=SERVERNAME1, NETTYPE=NETB, CACHE=XXX, COMMITCACHE=YES, DATABASE=AAAA
3. Start the second server in the second folder using the following parameters:

   SERVER SERVERNAME=SERVERNAME2, NETTYPE=NETB, CACHE=XXX, COMMITCACHE=YES, DATABASE=BBBB

   In these examples, xxx represents the cache size, and AAAA and BBBB are the names of the databases. Again, remember to write out the full path for each database file.
4. Now that the servers are up and running, start the first client, using the following parameters:

   FIN SERVERNAME=SERVERNAME1, NETTYPE=NETB
5. Start the second client, using the following parameters:

   FIN SERVERNAME=SERVERNAME2, NETTYPE=NETB

The system should now be up and running.

System Setup

Microsoft Navision Installation

Microsoft Navision can be installed in several different configurations. The various program properties can then be used to determine the way the different types of installations perform. However, the program properties do not all apply to the different configurations and their effect can vary.

Multiuser

In a multiuser installation, many users have common information that is stored in one or more databases on a server. The computers that work with the data are called clients, and the way the server and the computers work together is called client/server.

There are, however, two different types of multiuser installation:

- A classic client/server installation, where the server is installed on one computer, and the clients are installed on other computers in the network.

- A multiuser installation, where both the server and the client programs are installed on a file server, and the clients are run remotely from other computers in the network.

The Setup File (the zup file)

Microsoft Navision is able to remember the various settings that are used by different clients and single-users. This includes the server the client was connected to and the database and company the client was working with. It does this by saving these settings in a setup file. This setup file is called a zup file in Microsoft Navision. The different clients can all share the same zup file and thereby use the same settings, or they can all have individual zup files and thereby use their own customized settings.

If the network administrator has decided to implement roaming users, you will not have to utilize setup IDs. With roaming users, Windows will save the personal settings of each individual user on the server, including their zup file. This file will be called fin.zup and will contain Microsoft Navision information that is specific to that user.

If the network administrator has decided to implement local users, the zup file of each user will be saved on his or her local hard disk. Several users can also use the same computer and have their own zup files stored on the hard disk as long as they have individual windows accounts. The zup file of each individual user is stored in C:\DOCUMENTS AND SETTINGS\USER NAME\APPLICATION DATA. All of these zup files will be called fin.zup unless you create individual setup IDs.

You only have to implement setup IDs if the users are using the same Widows account.

If you do not enter a setup ID, the program will use the setup file called fin.zup when you start the program. You will be prompted to save any modifications that you have made during the session when you close the program. These changes will be saved as fin.zup and will be recreated the next time the program is started.

When several users are using the same zup file, they will each be prompted to save the changes they have made in the fin.zup file when they close the program. The fin.zup file that was saved by the previous user who quit the program will be replaced.

If you delete the zup file on a client or single-user installation, Microsoft Navision will recreate the default zup file the next time the program is started.

You can also make a standard setup created for a particular type of user available to other users with similar needs. To do this, copy the appropriate setup file to the folder from which the user starts the program, and enter the setup ID (file name) in the **Target** field.

Alternatively, you can store the setup file on a common drive in the network (but here it can be overwritten by other users). If you choose to place the setup file in a folder other than the one containing the program files, you must remember to specify the entire path name after id=.

The setup ID is not the same as the user IDs in Microsoft Navision, but you can use the same names. In fact, it can be an advantage to do so because although you cannot see the name of the setup file in the program, you can always see the user ID on the status bar at the bottom of the program window.

The following table lists the different program properties and the type of installation to which they apply:

Property	Client	Server	Described on
ID	X		Page 88
Server Name	X	X	Page 89
Database	X	X	Page 89
Company	X		Page 90
Windows Authentication	X		Page 91
Commit Cache	X	X	Page 92
Object Cache	X		Page 92
Net Type	X	X	Page 93
TempPath	X		Page 93
DB Test	X		Page 94
TestTarget	X		Page 94
Status Bar	X		Page 95
Close Form on Esc	X		Page 96
Marquee Full Selection	X		Page 96
Quick Find	X		Page 97
DB Read-Only	X		Page 97
DBMS Cache	X	X	Page 98
Stoptime		X	Page 99
Sessions		X	Page 100
Install as Service		X	Page 101
Uninstall as Service		X	Page 101

Single-User and Client Setup Properties

This section describes all of the program properties that you can set for both clients and single-users.

ID – Saving the User Setup

Program Property	Purpose	Where Specified	Default Value	Value
ID (clients only)	Saves the program setup	By entering ID=alice in the Command line or the **Target** field	Fin	Name of ID (including path if ID file is not located in Microsoft Navision folder)

Each user in a Microsoft Navision multiuser installation can choose the setup of windows and program properties that they want to use. Each user must have a unique ID in order for the program to be able to save and use the setup selections of the individual users. You can create a user setup by starting the program with an ID. The information about the users' setup will be stored under this ID. Here is an example in which the program starts with an ID called SUPER:

```
C:\PROGRAM FILES\MICROSOFT BUSINESS SOLUTIONS–MICROSOFT
NAVISION\CLIENT\FIN.EXE ID=SUPER
```

In a list of Microsoft Navision program files, you can see that each time you have started with a new setup ID, a file has been created that has the ID as the first part of the file name and .zup as the file name extension (for example, super.zup). This is called a setup file.

Returning to the Original Setup

You can always return to the standard setup file, fin.zup, by starting the program without specifying a setup ID.

If you have previously worked without a setup ID and made changes in the setup, the fin.zup file will contain these changes. If you do not want to use this modified fin.zup file but would prefer to return to the original starting point of the program, delete the fin.zup file and start the program again without an ID. The program will create a new, clean setup file, named fin.zup.

For more information about zup files, see the section called "The Setup File."

Server Name – Choosing the Server

Program Property	Purpose	Where Specified	Default Value	Value
Server Name	Specifies which server to connect to	Can be selected by clicking FILE→DATABASE→OPEN or by entering servername=My Server in the Command line or the **Target** field	None	Name of Server

This program property is used to specify the server that a particular client will connect to and does not apply in a single-user environment.

You can set up the connection to the server in the **Target** field by entering the name of the server after servername=.

You can connect to a server from within Microsoft Navision by clicking FILE→DATABASE→OPEN on the menu bar. For more information about connecting with a server and opening a database, see the section called Opening the Database.

Click FILE→DATABASE→INFORMATION and click the **Database** tab to see which server you are currently connected to.

Database – Selecting a Database

Program Property	Purpose	Where Specified	Default Value	Value
Database	Specifies which database to open. A Microsoft Navision Database Server can only have one database open at a time.	FILE→DATABASE→OPEN or by entering database=My database in the Command line or the **Target** field. In a client/server installation this only works in combination with servername=	None	Name of database (including path if database is not located in Microsoft Navision folder)

The Database program property is used to make the program start with a particular database open. (The database must already exist.) In the **Target** field or on the command line that starts the program, type the name of the database immediately after database=.

To open a database from within Microsoft Navision:

1. Click FILE→DATABASE→OPEN.
2. In the window that appears, select the server and the database that you want to open.

Click FILE→DATABASE→INFORMATION to see which database is being used. For more information about databases, see the section called Working with Databases.

Microsoft Navision Database Server can only have one database open at a time. If a client wants to open a database other than the one that is currently open, he will have to wait until the first database is closed.

When you select a database for a client, you can also select the company you want to open automatically by using the Company program property.

Company – Selecting a Company

Program Property	Purpose	Where Specified	Default Value	Value
Company	Specifies which company to open.	FILE→COMPANY→OPEN or by entering company=CRONUS International Ltd. in the Command line or the **Target** field. In a client/server installation, this only works in combination with servername= and database=	None	Company Name

With this program property, you can select the company that will open automatically when a client starts Microsoft Navision. You must also specify the server and the database that contain the company before you specify the company in the **Target** field.

Because Microsoft Navision can only have one database open at a time, you must connect to a server that already has the appropriate database open. If no other clients are using the system, you can select the server, database, and company that you want.

From within the program, you can select a company by clicking FILE→COMPANY→OPEN. You can also select a company from the list displayed at the bottom of the File menu. You can see the current company on the title bar of the program window.

Windows Authentication – Selecting the Authentication Mode

Program Property	Purpose	Where Specified	Default Value	Value
NTAuthentication	Specifies which type of authentication is to be used	FILE→DATABASE→OPEN or by entering ntauthentication=yes in the Command line or the **Target** field.	Yes	Yes/No

This program property is used to determine which type of authentication is to be used when logging on to a server and opening a database.

After selecting the server and the database in the Open Database window, you must:

1. Select the type of authentication that is to be used.
2. Enter your user ID and your password, if database server authentication is being used. If Windows authentication is being used, you do not have to enter a password or user ID.
3. Click **OK**.

Alternatively, you can enter "yes" or "no" after ntauthentication= in the **Target** field or on the command line that starts the program.

If you are using Windows authentication, Microsoft Navision will start, automatically connect to the server, and open the database that you have specified.

If Database authentication is being used, Microsoft Navision will start and prompt you to supply your user ID and password before connecting to the server and opening the database.

For more information about the types of authentication used in Microsoft Navision, see "Microsoft Navision Security."

Commit Cache – Writing the Cache

Program Property	Purpose	Where Specified	Default Value	Value
Commit Cache	Makes the program run faster	TOOLS→OPTIONS or by entering commitcache= in the Command line or the **Target** field.	No	Yes/No

The Commit Cache program property allows Microsoft Navision to postpone writing the information stored in cache on the server to the database until later. Storing this information in cache allows Microsoft Navision to work faster.

You must restart the program before any changes that you make to this parameter take effect.

Object Cache – Improving Response Times

Program Property	Purpose	Where Specified	Default Value	Value
Object Cache (KB) (clients only)	Makes the program run faster	TOOLS→OPTIONS or by entering objectcache=8000 in the Command line or the **Target** field.	8000 KB	More than 0 KB and less than 1,000,000 KB

The Object Cache property increases the speed of the program. Objects such as code, descriptions, and windows that will be used on the client computer are stored in the object cache. This means that the client computer only needs to retrieve these objects once from the server, and then they will be stored in the object cache. The client computer must have enough memory to store the objects while they are being used in order to benefit from the object cache.

Running out of object cache (that is, setting too small a value) does not cause any problems. The total size of all the objects used in the standard application is approximately 20 MB. If you have enough memory, set the object cache to 20 MB. The size of the most important objects, such as the table descriptions, is 1 MB. You should therefore, as a minimum, set the object cache to 1 MB. The upper limit is 1 GB.

Click FILE→DATABASE→INFORMATION to open the Database Information window and see how much space has been allocated in the **Object Cache (KB)** field. To change the amount of space allocated to the object cache, on the menu bar, click TOOLS→OPTIONS, and enter the setting in the **Object Cache (KB)** field.

NetType – Selecting a Net Type

Program Property	Purpose	Where Specified	Default Value	Value
NetType	Permits choice of network protocol	TOOLS→OPTIONS or by entering NetType= in the Command line or the **Target** field.	TCP	Netb, TCP

To use Microsoft Navision in a network, you must select the network protocol that is used for communication between the server and the clients. There are two possible values:

- nettype=tcp (for TCP/IP)
- nettype=netb (for NetBIOS)

The same selection must be entered on all the client computers in the network as well as on the server. On the server, enter the net type you have selected in the **Target** field or on the command line after the start command. On the client computers, enter the selection in the **Target** field, or click TOOLS→OPTIONS on the menu bar within Microsoft Navision.

To check the setting when you are using the program, click FILE→DATABASE→INFORMATION, and look at the **Connection** tab.

TempPath – Location of Temporary Working Files

Program Property	Purpose	Where Specified	Default Value	Value
TempPath	Specifies location of temporary working files created automatically	TOOLS→OPTIONS or by entering Temppath= in the Command line or the **Target** field.	Windows 2000: c:\Documents and Settings\User Name\Application Data\Local Settings\Temp Windows XP: <C:\Documents and Settings\User Name\Local Settings\Temp	Path to Temp Files

When Microsoft Navision is running, it creates a number of temporary files, which are automatically deleted when you close the program. As a default, the temporary files of each individual user are stored in the location listed in the previous table unless you specify a different working folder. If you do so, this working folder will be the default location. You can specify the working folder in the **Target** field or by clicking TOOLS→OPTIONS. You must specify the full path, including the drive and all the folders.

DB Test – Testing the Database

Program Property	Purpose	Where Specified	Default Value	Value
DB Test	Tests the database	FILE→DATABASE→TEST or by entering dbtest=min in the Command line or the **Target** field.	None	Min., Normal or Max.

You can use this program property to test the consistency and integrity of the database. You can also run the test from within the program by clicking FILE→DATABASE→TEST. You can specify exactly what you want to test in the dialog box that appears.

When you enter the DB Test program property in the **Target** field, the database will be tested before the program opens. (You can read about the extent of these tests, as well as how to create a customized version of the database test, in "Microsoft Navision Database Administration Tools.") You can specify one of the following options:

- dbtest=min
- dbtest=normal
- dbtest=max

TestTarget

Program Property	Purpose	Where Specified	Default Value	Value
TestTarget	To specify how error messages generated by the database test are managed.	FILE→DATABASE→ TEST or by entering testtarget=@screen in the Command line or the **Target** field.	@screen	@screen, @eventlog, filepath

You use this program property to specify how any error messages that are generated during a database test are managed. They can be displayed on the screen or stored in the Event Log or in a text file.

You can enter one of the following options:

- *testtarget=@screen*
- *testtarget=@eventlog*
- *testtarget=filepath*

You must enter the full path and the name of the text file. If you select event log, you can read the error messages that were generated during the database test in the Windows Event Viewer. If you select screen, the error messages will be displayed on the screen, and the database test will require interaction from the user if any errors are found. Selecting screen can make the database test quite time consuming.

Status Bar

Program Property	Purpose	Where Specified	Default Value	Value
TestTarget	Activated or deactivates the status bar	TOOLS→OPTIONS	Yes	Yes/No

On the menu bar, click TOOLS→OPTIONS, and in the Options window you can specify whether or not the status bar will be displayed at the bottom of the program window.

The status bar contains the following information:

- The complete name of the active field and its contents.
- The work date.
- The current user ID.
- Whether or not any filters have been placed on the data (FILTER appears).
- Whether or not you are about to create something NEW (an account, for example). Whether you are working in Insert (INS) or Overtype (OVR) mode.

When you make a visible change in the setup (such as making the status bar invisible), it is practical to use the ID program property and a setup file on your own computer. This makes the setup selections valid only for yourself. For more Information, see the section called "ID – Saving the User Setup." This property can only be adjusted from within Microsoft Navision.

Close Forms on Esc

Program Property	Purpose	Where Specified	Default Value	Value
Close Forms on Esc	Determines whether windows close when you press Esc	TOOLS→OPTIONS	Yes	Yes/No

Click TOOLS→OPTIONS on the menu bar, and you can choose whether or not the window you are working in will close when you press **Esc**.

It is practical to use the ID program property and have a setup file on your own computer if you change the setup. This makes the setup selections valid only for yourself. For more information, see the section called "ID – Saving the User Setup."

This property can only be adjusted from within Microsoft Navision.

Marquee Full Selection

Program Property	Purpose	Where Specified	Default Value	Value
Marquee Full Selection	Determines how graphical objects are selected on the screen	TOOLS→OPTIONS	No	Yes/No

With this setting, you can choose whether graphical objects must be completely within the frame in order to be selected or whether it is sufficient for them just to touch the edges. This property is relevant for developers using the C/SIDE® development environment. To make this selection, on the menu bar click TOOLS→OPTIONS, and make your selection in the Options window.

This property can only be adjusted from within Microsoft Navision.

Quick Find

Program Property	Purpose	Where Specified	Default Value	Value
Quick Find	Quick search by letter in all windows	TOOLS→OPTIONS	Yes	Yes/No

This setting allows you to activate a quick search facility. When the Quick Find setting is enabled, you can search for an entry in any non-editable field by typing a letter or number. You can also enter the entire name of the element you are looking for. When you enter a letter or number, the Find window opens, and the first row that matches what you entered becomes the active row.

When the Quick Find property is disabled, you can open the Find window by clicking EDIT→FIND on the menu bar or by clicking **Find** on the toolbar.

This property can only be adjusted from within Microsoft Navision.

DB Read-Only

Program Property	Purpose	Where Specified	Default Value	Value
DB Read-Only	Determines that it is impossible to enter any data into the database.	By entering dbreadonly= in the Command line or the **Target** field.	No	Yes/No

This program property allows you to specify that the database has read access only. This prevents other users from entering data into the database.

Microsoft Navision Database Server Setup Properties

Almost all the program properties for the server can be specified in the **Target** field or on the command line that is used to start the server.

The following four properties apply to both the server and the client and have been described in the previous section:

Program Property	Purpose	Where Specified	Default Value	Value
Commit Cache	Makes the program run faster	Command line, **Target** field	No	Yes/No
Server Name	Sets the name of the server	During installation	The name of the server computer	Server Name
Database	Selects an existing database	Command line, **Target** field	None	Name of the database (including path if not located in Microsoft Navision folder)
Net Type	Selects network protocol	During installation, Command line, **Target** field	TCP	TCP or Netb

The server name is specified during the installation, and the only way it can be changed is to uninstall the server and then install it again giving it a different name.

The remaining setup properties only apply to the server and are described in this section.

DBMS Cache

Program Property	Purpose	Where Specified	Default Value	Value
DBMS Cache (KB)	Makes the program run faster	TOOLS→OPTIONS or by *entering cache=* in the Command line or the **Target** field.	8000 KB	More than 100 KB and less than 1,000,000 KB

This property only applies to server and single-user installations.

DBMS Cache (Database Management System Cache or just Cache) is the name for a reserved space in the computer's memory – where data is stored temporarily, until it has been processed completely. After this, the commit cache transfers it to the hard disk. This system quickly frees Microsoft Navision for new work.

When the cache becomes full, old information in it is replaced with new. When you turn off the computer, everything in the cache is cleared from the memory. At that point, it has already been transferred from the cache to the database.

The cache requires a minimum of 100 KB memory. The upper limit is 1 GB. To see how much cache has been allocated, click FILE→DATABASE→INFORMATION and check the **DBMS Cache (KB)** field in the Database Information window.

It is usually advantageous to specify as large a cache as possible. If the computer does not have enough memory, however, doing so can cause the operating system to "swap." Swapping means moving part of the cache to the hard disk in order to make more memory available. When this happens, the performance will be considerably slower because reading and writing take much longer when the hard disk is involved than when these operations are carried out in memory. To avoid swapping, you can add more memory to your computer, reduce the number of programs running simultaneously, or you can make the cache smaller.

You must restart the program before any changes that you make to this parameter take effect.

Stoptime – Closing the Program Automatically

Program Property	Purpose	Where Specified	Default Value	Value
Stoptime	Stops the server automatically at a particular time	By entering *stoptime*= in the Command line or the **Target** field.	None	hhmmss

You can set the server Stoptime program property in the **Target** field or on the command line. Enter the property as follows: stoptime=hhmmss, with hhmmss replaced by a time in hours, minutes, and seconds.

The server will stop automatically when the computer's built-in clock reaches the specified time. You can use this function to make the database unavailable after a particular time. It is not necessary to stop Microsoft Navision in order to make backups. For more information see "Microsoft Navision Database Administration Tools."

Sessions – Number of Client Sessions Allowed

Program Property	Purpose	Where Specified	Default Value	Value
Sessions	Specifies the number of client sessions that can be connected to the server at the same time	By entering *sessions*= in the Command line or the **Target** field.	Maximum determined by the license file	Number of sessions allowed by the license file

A session is an active (running) copy of Microsoft Navision. The term must not be confused with client, which refers to one connected computer. One client computer can run several sessions at a time if the client program is started more than once.

Here are two different ways in which ten sessions can be running:

- Two computers, each with five sessions (each computer starts the program five times)
- Ten client computers, each started once

The Sessions program property is set on servers and specifies the number of sessions that are allowed to be connected to the server at one time. When you obtain your license file, you obtain permissions for a specific number of sessions. These are automatically assigned to the server when you start it.

To see how many sessions a license allows, click FILE→DATABASE→INFORMATION. The Database Information window appears. Look on the **Sessions** tab in the **Licensed Sessions** field.

You use this program property to distribute the sessions efficiently – either because you want to start several servers with the same license file or because you want to limit access to the current server.

If a client tries to start a session that will exceed the number specified, a message will appear.

Install As Service

Program Property	Purpose	Where Specified	Default Value	Value
Install as Service	Installs the server as a service on the Windows server	**Target** field or Command line	Has no value	Has no value

Microsoft Navision Database Server is automatically installed as a service. When the Microsoft Navision Database Server is installed as a service, it starts every time you start the server computer – without the user having to log on to the system. To manually install the Microsoft Navision Database Server as a service, enter the following after the command prompt:

```
server installasservice
```

Do not use the equal sign with this property.

NOTE: *If you install Microsoft Navision Database Server as a service and want to use Hotcopy to store the database backups on a remote computer, you must change the Log On options of the Microsoft Navision Database Server service. The Microsoft Navision Database Server service must use the credentials of a domain user that has the appropriate rights on the remote computer and not use the default value (local system account). For more information about HotCopy, see "Microsoft Navision Database Administration Tools."*

Uninstall As Service

Program Property	Purpose	Where Specified	Default Value	Value
Uninstall as Service	Uninstalls the server as a service on the Windows server	**Target** field or Command line	Has no value	Has no value

If you want to uninstall Microsoft Navision Database Server as a service, enter the following after the command prompt:

```
server uninstallasservice
```

Do not use the equal sign with this property.

Microsoft Navision Database Server Manager

If the Microsoft Navision Database Server Manager Snap-in for the Microsoft Management Console is installed and Microsoft Navision Database Server is installed as a service, you can use the Microsoft Management Console to change the properties of Microsoft Navision Database Server, using the Microsoft Navision Database Server Manager.

The Microsoft Navision Database Server Manager is a Microsoft Management Console snap-in that makes it possible for you to manage Microsoft Navision Database Servers across a domain. With the Microsoft Navision Database Server Manager, you can see and reconfigure a number of Microsoft Navision Database Server properties. The Microsoft Navision Database Server can be started immediately after the installation of Microsoft Navision Database Server is complete. For more information about the installation, see the section "Installing Microsoft Navision Database Server."

The Microsoft Navision Database Server Manager interacts with Microsoft Navision Database Server the same way that the SQL Server Enterprise Manager does with SQL Server and requires that Microsoft Navision Database Server is installed. The Microsoft Navision Database Server Manager accesses the Microsoft Navision Database Server properties through the registry. If any changes are made to the properties, the Microsoft Navision Database Server Manager notifies Microsoft Navision Database Server about the changes in order for it to respond accordingly.

You can change the properties at run-time, and they will be effective immediately. If Microsoft Navision Database Server is not running, the properties will take effect the next time you start Microsoft Navision Database Server.

Adding Snap-In

To open the Microsoft Navision Database Server Manager for the first time:

1. Open Microsoft Management Console. The Microsoft Management Console consists of two panes.
   - The left-hand pane displays the actual contents of the Manager, that is, an overview of the nodes that have been added to the view. The right-hand pane contains HTML pages that represent the server managers.
2. Click FILE→ADD/REMOVE SNAP-IN
3. In the Add/Remove Snap-in window, click **Add**.
4. In the Add Standalone Snap-in window, select Microsoft Navision Database Server Manager and click **Add**.
5. Click **Close** and **OK** to return to the console.

You can now add Microsoft Navision Database Server to the server manager. In the following diagram, one Microsoft Navision Database Server node has been added to the Microsoft Navision Database Server Manager node.

Adding Microsoft Navision Database Servers

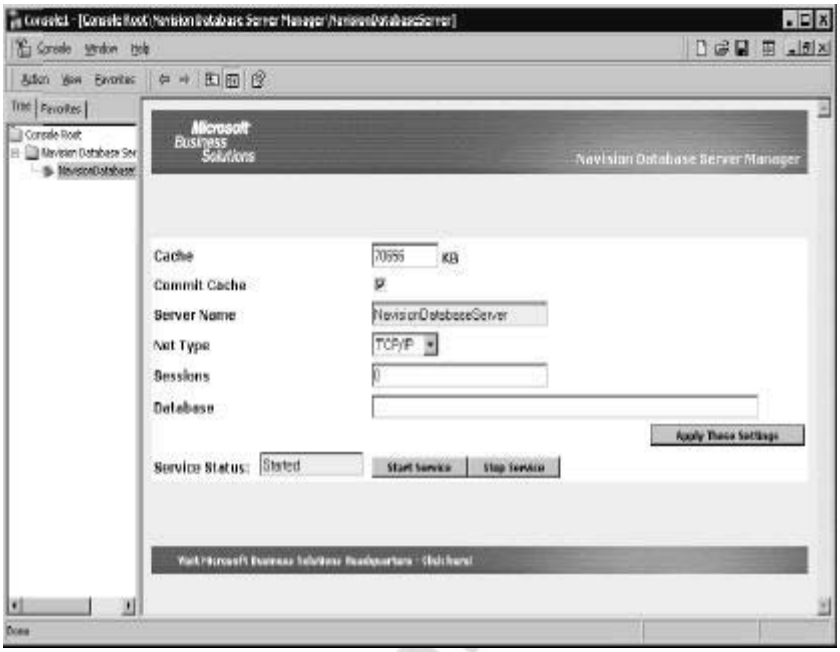
You can add as many Microsoft Navision Database Servers to the Microsoft Navision Database Server Manager as you want. They will be listed under the Microsoft Navision Database Server Manager node in the left-hand pane of the console.

To add a Microsoft Navision Database Server to the server manager:

1. Right-click the **Microsoft Navision Database Server Manager** node.
2. Click NEW→DATABASE SERVER MANAGER, and the New Microsoft Navision Database Server Manager window opens:

3. The default value that you should enter in the **Host Machine** field is LocalHost, but you have access to all computers in the domain.
4. Enter the name of the Microsoft Navision Database Server that you want to add, for example Microsoft NavisionDatabaseServer. This is the name that you gave the server when you installed it and must not contain any spaces.
5. Click **OK** to return to the console.

Microsoft Navision Database Server Manager now contains one node, that is, one Microsoft Navision Database Server:



In the right-hand pane of the Microsoft Navision Database Server Manager window, you can define or change the properties of Microsoft Navision Database Server nodes. To see the configuration settings for a specific Microsoft Navision Database Server, click the Microsoft Navision Database Server node that you want to see the properties for. In the right-hand pane of the window, the properties are then displayed.

You can enter new values for each of the shown properties for each Microsoft Navision Database Server node.

You also have the option to start or stop Microsoft Navision Database Server as a service. In the **Service Status** field, you can see the current status of the server, that is, whether it is running or not.

The field has the following three options:

- Stopped
- Starting
- Started

When you have made your changes to the configuration (by editing the values) and confirmed the changes by clicking **Apply These Settings**, the Microsoft Navision Database Server Manager does the following:

1. Changes the properties for Microsoft Navision Database Server and places them in the registry.

2. If Microsoft Navision Database Server is running, the Microsoft Navision Database Server Manager notifies Microsoft Navision Database Server of the changes that have been made to the registry. The changes take effect immediately. If Microsoft Navision Database Server is not running, the changes will take effect when you restart Microsoft Navision Database Server.

If the configuration change fails, Microsoft Navision Database Server shuts down, and the program reacts in the following way:

1. The value in the **Service Status** field changes to Stopped.

2. An error message is then logged in the event log of the Windows server.

Setting the Program Properties

As explained in the previous sections, you can customize the system setup by changing the settings of the various program properties.

Some settings must be entered in the **Target** field of the Microsoft Navision Properties window (see the section called Connecting Automatically). You can also start a server from a command prompt. If the server has been installed as a service and the Startup Type option is set to automatic, the server will start automatically every time you start the operating system.

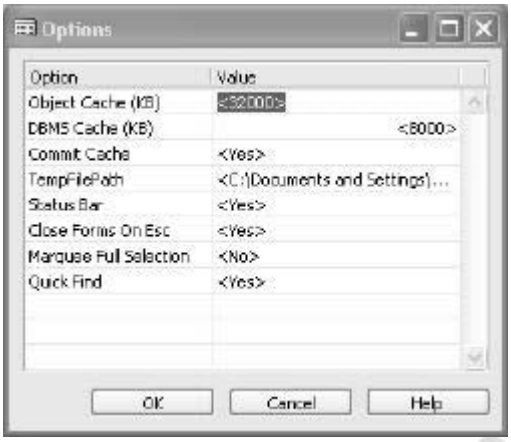
You can specify the program properties in any order. Enter them after the program's start command, separated by commas. The name of each program property is followed by an equal sign (=) and the value to which the property is to be set, for example:

```
D:\FIN\FIN.EXE SERVERNAME=MY SERVER, COMPANY=CRONUS
INTERNATIONAL LTD., ID=ALICE
```

This does not apply, however, to the two server properties: "installasservice" and "uninstallasservice." For these properties, enter the parameter on the command line as follows:

```
D:\FIN\SERVER.EXE INSTALLASSERVICE
```

You can also set most of the program properties from the menu bar in the program – for example, by clicking FILE→COMPANY→OPEN (to set Company) and FILE→DATABASE→OPEN (to set Database). To see the properties that do not exist as menu items click TOOLS→OPTIONS, and the Options window appears:



***NOTE**: Any changes made in this window are saved in the setup file and will be valid the next time the program is opened. If you do not want users to be able to make "permanent" changes in these options, you can set default values in the command line of a batch file (called fin.bat, for example) with which the user starts the program.*

The program properties that you can set depend on whether you are setting up a client computer, or a server.

Connecting Automatically

Many of the properties that are described in the this chapter can be entered as command lines after the command prompt or included as program properties that are automatically set when you start Microsoft Navision.

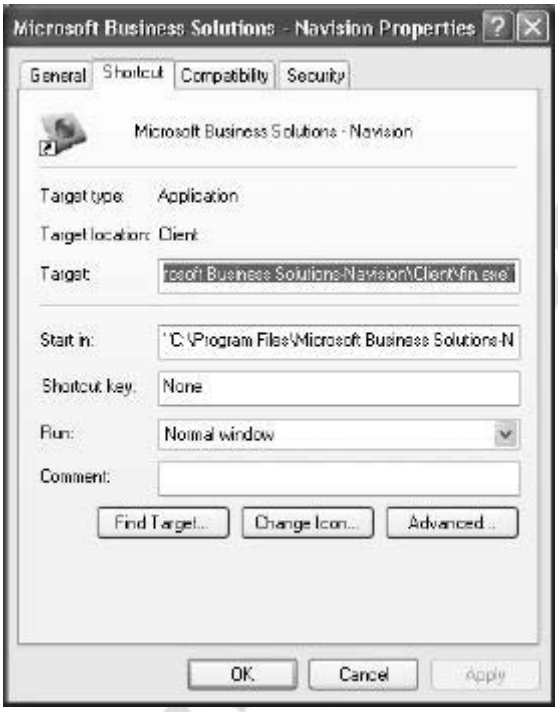
To carry out this procedure, you must have administrative rights on the computer. To set them as automatic program properties:

1. Open Windows® Explorer.
2. Open the folder DOCUMENTS AND SETTINGS\ALL USERS\START MENU\ PROGRAMS\MICROSOFT BUSINESS SOLUTIONS – MICROSOFT NAVISION.
3. Notice that there are two identical Microsoft Navision icons on the right. You must select the icon for the Microsoft Navision executable that you are using.

***IMPORTANT**: If you are using Windows XP and Windows 2000 you MUST delete these existing shortcuts and create new ones in order to gain access to the **Target** field. New shortcuts should be made from the Microsoft Navision Client folder and then copied into the same path noted in step 2.*

Also note that by copying the shortcut to the user specific folders, you can tailor the program properties to the individual user.

4. Click FILE→PROPERTIES, or right-click the Microsoft Navision icon and select **Properties**. The Microsoft Business Solutions–Microsoft Navision Properties window appears:



5. Click the **Shortcut** tab. The **Target** field shows where Microsoft Navision is located. It contains the path for the start command fin.exe. After the start command, you can add other commands and settings for program properties. Here is an example:

   SERVERNAME=MY SERVER, NETTYPE=TCP, COMPANY=CRONUS INTERNATIONAL LTD.

When you set the program properties in this way, the program performs certain tasks the next time it is opened. It will use TCP/IP to connect to the server called My Server and open the company called CRONUS International Ltd., if it exists in the database. If you do not enter the company parameter, you will have to open the company manually after the program starts.

You can change any of these selections while you are working. You can do this from within the program. For example, you can select a different database (provided it has already been created) or a different company, or you can create a new company. If you do not want a client to be able to do these things, you can set limits when you assign user permissions (by setting limits on the "system" object type). You can read about assigning user permissions in "Microsoft Navision Security."

Working with Databases

The Standard Database

You must have a database to be able to work with Microsoft Navision. When you install a singleuser installation, a standard database called database.fdb is automatically provided.

Using the Standard Database

You can use the standard database (database.fdb) in two ways: with a demonstration license (CRONUS.flf) or with your own license (fin.flf). The standard database contains a demonstration company called CRONUS International Ltd.

Using the Demonstration License File Cronus.flf

If you choose to work with the demonstration license file, cronus.flf, you have access to all the Microsoft Navision application areas and can test all the functions – including ones you have not purchased permissions for. The demonstration license file does, however, contain certain restrictions:

- Posting is only possible in the period November to February.
- You are only allowed to make 4000 write transactions in a database.
- The maximum number of companies is two.
- You can only have stand-alone installations or run Microsoft Navision Database Server on Windows 2000 and Windows XP.
- You can have a maximum of two sessions running at any one time.
- Any company name must start with CRONUS (written in capital letters). This ensures that it will be clearly identifiable as a demonstration company – and you will not accidentally create a "real" company with the wrong license file.

Using Your Own License File

If you work with your own license file (fin.flf), you can use only the functions for which you have purchased permissions. This means that you can see only the data for those functions – even in the demonstration company.

On the other hand, your own license file does not limit posting dates. You can also create as many companies in database.fdb as you have purchased permissions for. If you create more than one additional company in the database, however, you will no longer be able to use the license file CRONUS.flf because it allows only two companies in the database. Thus, you will lose the benefits of using cronus.flf, for example, you will not be able to see all the functions in the entire demonstration company.

Creating and Maintaining Databases

The demonstration database contains many limitations, which can be difficult to keep track of. Therefore, we recommend that you create a separate database for your own companies. Your license file specifies the maximum size for your database, and you can create as many new companies as you like within the allowed space.

To create a new database for your companies, you must:

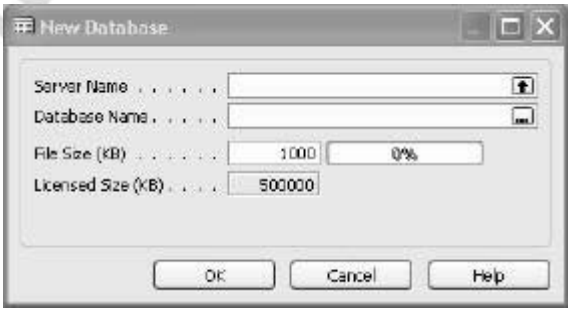
1. Create the new database.
2. Restore the backup of the original standard database (database.fbk) into the new database. The backup must include at least "Data Common to All Companies" and "Application Objects." "Data Common to All Companies" includes the program's report list and permissions groups. Restoring Application Objects transfers the accounting application to the database. For more information about making backups, see "Microsoft Navision Database Administration Tools."

NOTE: *Regardless of whether you choose to continue to work in the standard database or create a new one, make sure that you always have at least one copy of the database you are working with stored in a safe location. If you accidentally delete the folder containing Microsoft Navision, the database file will disappear, which means that all your data will be lost.*

Creating a Database

To create a new database from a client or from a single-user installation:

1. Click FILE→DATABASE→NEW. The New Database window appears:



2. In the **Server Name** field, enter the name of the server. You can use the **AssistButton** and select the server from a list of the Microsoft Navision Database Servers that are available within the current domain.

3. When you enter something in the **Server Name** field, the **AssistButton** in the **Database Name** field will disappear, indicating that you must enter the name of the database manually. If you want to create a database that will not be stored on the server, you must enter the entire path and the name of the database manually.

   If you want to create a database that is stored locally, you do not need to enter anything in the **Server Name** field and can use the **AssistButton** to open a standard Windows dialog box where you can create the database in the desired location.

4. Enter the desired size (KB) of the database in the **File Size (KB)** field. The minimum size (1000 KB) is already entered in the field. To the right of this field, you can see what percentage of the maximum size is being taken up by this database.

5. The **Licensed Size** field tells you the maximum size of your database. This information is contained in your license file.

***WARNING**: Never create the database on compressed drives (this also includes the Compress property under the NTFS file system). A database located on a compressed drive can be corrupted by a power failure.*

Database Size

The size depends on how many bookkeeping transactions will take place each day, week, and so on. Your Microsoft Certified Business Solutions Partner can help you choose a suitable database size. Do not make the database too large to begin with; you can always expand it, but you can never make it smaller (and the allocated space cannot be used for anything else on your computer). The maximum size of a database is 128 GB.

The number of KB you have permissions for is displayed in the **Licensed Size (KB)** field.

Database in Several Files

When you create a database, it consists of only one file. After the database has been created, you can divide it into several files and place them on different hard drives. This process is described in the next section called "Expanding the Database."

Dividing your database into several files on several (physical) disks lets you utilize disk space optimally. It also reduces the access time to the database, which improves the program's performance. You can read more about performance issues in "Microsoft Navision Architecture."

Using Commit Cache

If you have divided your database into several files on several disks, it will be utilized most efficiently if you start the program with the program property Commit Cache set to Yes. You can set this up under TOOLS→OPTIONS on the menu bar or in the **Target** field.

Multiuser Installations

In multiuser installations, where there is a network server available, it is possible to start the Microsoft Navision Database Server program from a computer in the network and at the same time have the database files located on the network server instead of on Microsoft Navision Database Server. This is not recommended, however, because it places a heavy load on the network, which will decrease performance. Normally, Microsoft Navision sessions communicate directly with Microsoft Navision Database Server, but if the database files are on a network server, all the database transactions must be sent from the server program via the network to the database files on the network server.

Another reason for not having database files on the network server is that this increases the risk of introducing transmission errors into the data that is being sent back and forth. Data sent between the server program and the database in this way is not protected by the Microsoft Navision checksum system. Transmission errors in a network are much more common than disk errors or other computer errors, so the risk of introducing errors is sharply increased by putting the database files on the network server.

Do not store database files on drives that are "shared" through the network program. If database files are located on shared drives, the programs performance declines drastically.

Expanding the Database

You use the expand database function to divide the database into several database files and spread them over separate disks. You can also use it to increase the size of these files if the database is beginning to run out of space and to increase the size of a single file database.

You do not need to expand your database until you are about to run out of available space. You can check this by clicking File, Database, Information and by looking in the **Database Used (KB)** field in the Database Information window. The value in this field should not exceed 80% (general rule to follow). As soon as it reaches this level, it is a good idea to expand the database.

The amount of free space in the database must reflect the amount of space required for the maximum number of updates that can be carried out by the other concurrent users, when one user is performing the lengthiest task that your installation contains. This is approximately 16 MB per user (general rule to follow).

WARNING: *Microsoft Navision does not give you any warning that you are about to run out of space in the database. You must keep track of the status of the database, including how much space is left and how much database space you have purchased permission for.*

However, Microsoft Navision displays an error message if there is not sufficient free space to perform the current task. It also displays an error message if you are performing a time-consuming task, and all the available space in the database that contains your snapshot has been used by other concurrent activities.

Performance

The performance of the program depends only slightly on the size of the database. Unsatisfactory performance may be caused by problems with your equipment, by storing the database in an impractical location, or by setting the Cache and Commit Cache program properties without giving due consideration to your needs.

Requirements

To expand your database, you must have adequate space available on your hard disk(s), and you must have purchased permission for (at least) a database of the size you want to expand to. The **Licensed Size (KB)** field in the Database Information window shows the size database you have purchased permission for. If you do not have permission to expand the database further, an error message will appear when you try to do so.

Remember that it is not always a good idea to simply expand your database to the maximum size possible; once space has been allocated to the database, it cannot be used for anything else on your computer.

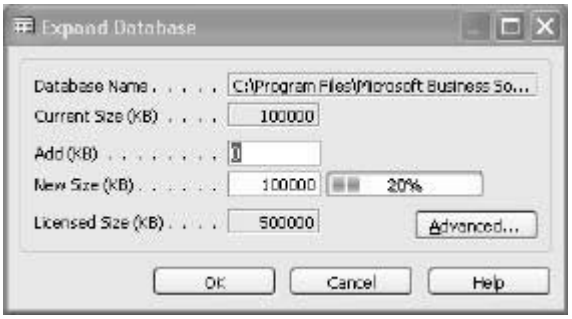
NOTE*: Before you expand your database, you should back up your database as described in "Microsoft Navision Database Administration Tools."*

Ways to Expand the Database

There are two ways to expand the database:

- Expand the existing database file.
- Create one or more new database files, possibly in different locations. This can be advantageous if there is more space on another disk.

No matter which method you choose, start by clicking FILE→DATABASE→EXPAND. The Expand Database window appears:

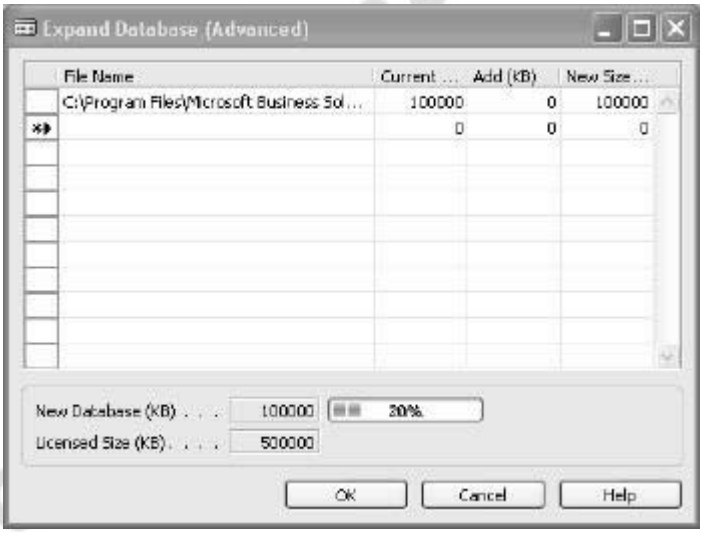


Expanding an Existing Database

To expand the database, enter the number of kilobytes you want to add in the **Add (KB)** field, or enter the new total number of kilobytes in the **New Size (KB)** field. Click **OK** to expand the database.

Creating an Extra Database File

To create an extra database file in order to expand the existing database, click **Advanced** in the Expand Database window. The Expand Database (Advanced) window appears:



Here you enter information about the new database file or files you want to create.

Each new database file must be set up on its own line. Enter the name (including the full path) in the **File Name** field and the size of the database file in the **Add (KB)** field. If the actual database name is db.fdb, for example, you could call one database file "dbpart1.fdb" and another "dbpart2.fdb" and place them in different folders on different drives.

When you are expanding the database it is important to remember that:

- All of the database files must be approximately the same size. This improves performance.
- The database can consist of up to 16 physical files.
- Each database file can be between 1 KB and 128 GB.
- All of the files together cannot exceed 128 GB.

Name of a Database with Multiple Files

The proper database name, that is, the one that will be referred to elsewhere in the program, appears on the first data entry line. This is the primary file and it points to the other files. As far as the user is concerned, the database is one logical file. If you need to use the database name (to open the database or make a Microsoft Navision backup, for example), use this name and ignore the fact that the database consists of several files.

At the bottom of the window, you can see the size of the complete database increase as you create each file.

When you click **OK**, the information is saved, and you can continue to work in the database.

Moving a Company from one Database to Another

If you happen to create a company in the wrong database, you can move it to the correct database by making a backup copy of just the company. (Select the appropriate company before you begin the backup.) When the backup is complete, you open the correct database and restore the company backup into it. Before you can use this database, you must restore a backup of the standard database.

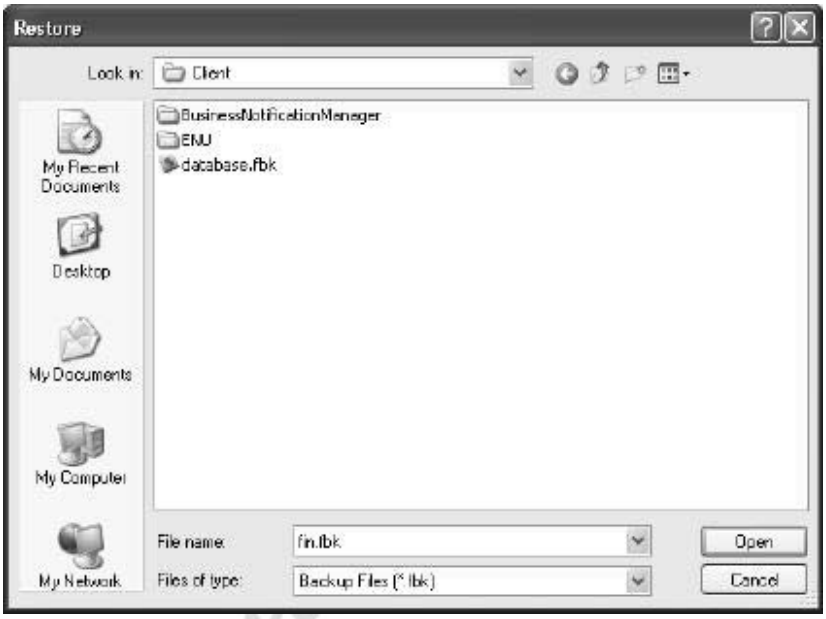
Restoring the Standard Database

The database that you have just created contains only a few basic tables and is not yet ready for use in Microsoft Navision. Before it can be used, you must restore the backup of the original standard database (database.fbk) into the new database.

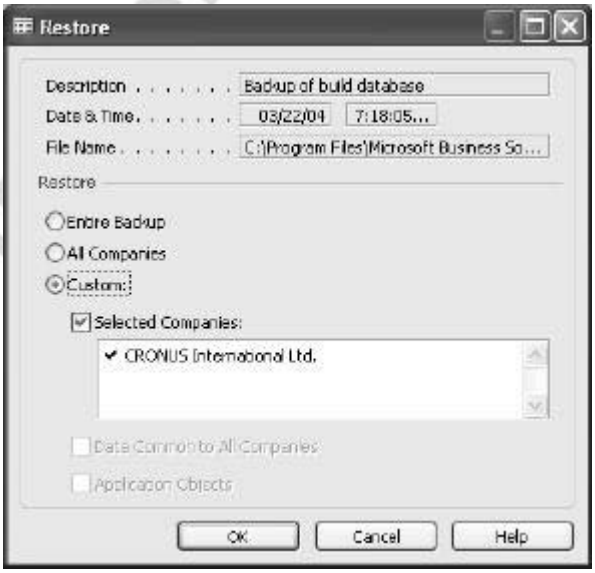
This backup comes with Microsoft Navision and is stored in the Microsoft Navision folder on the client computer if you have carried out a complete installation. The backup contains all the information necessary for using the database, including “Data Common to All Companies” and “Application Objects.” “Data Common to All Companies” includes the program’s report list and permissions groups. When you restore the “Application Objects,” the accounting application is transferred to the database.

To restore the standard database:

1. Open the new database, and on the menu bar click TOOLS→RESTORE. The following window appears:



2. Select database.fbk, and click **Open**. The Restore window appears:



3. Make sure that **Custom** is selected and that the **Data Common to All Companies** and **Application Objects** check boxes are selected.

4. Click **OK** to start restoring the database.

The restore procedure will take a few minutes, and this window allows you to monitor its progress.

For more information about making backups and restoring databases, see "Microsoft Navision Database Administration Tools."

When the restore process has been completed, your database will be ready for use in Microsoft Navision.

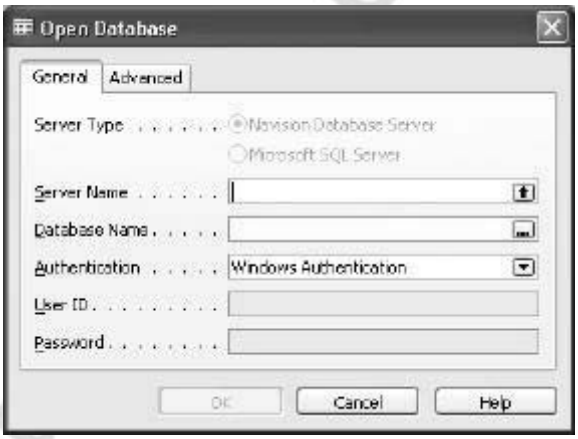
You can now create your own companies in the database.

Opening the Database

On clients and single-user installations, you can open the database from within the program.

To open a database:

1. Click FILE→DATABASE→OPEN. The Open Database window appears:



2. In the **Server Name** field, enter the name of the server. You can use the **AssistButton** and select the server from a list of the Microsoft Navision Database Servers that are available within the current domain. You can only use this function if you are running in a Windows network.
3. If you have selected a server, you will not have to enter any information in the **Database Name** field.
   a. If you want to open a database that is not stored on the server, you must enter the entire path and the name of the database manually.

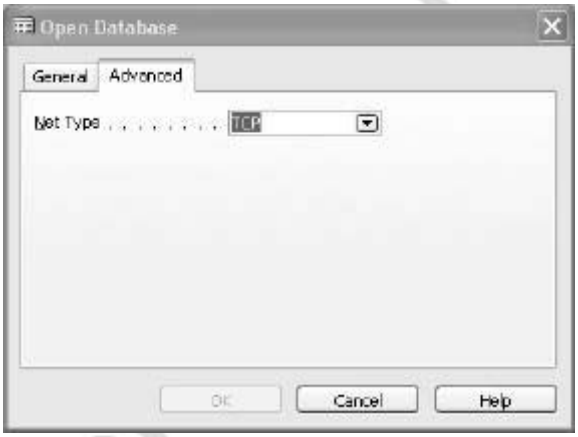
b. If the database is stored locally on your hard drive, you must enter the name of the database in the **Database Name** field. If you do not know the name of the database, you can use the **AssistButton**. A standard Windows dialog box will appear, and you can use it to locate the desired database with the file extension .fdb.

4. In the **Authentication** field, specify the type of authentication you require. You can choose between Windows Authentication and Database Server Authentication. You can use the **AssistButton** to select the authentication type from a list. You can read more about authentication in the Chapter 9, "Microsoft Navision Security."

5. You must enter a user ID and password if you are using database server authentication. If you are using Windows authentication, you do not have to enter a user ID and password.

6. Click **OK** to open the database.

You can specify the network type that will be used when you connect to the server. In the Open Database window, click the **Advanced** tab. However, it is not usually necessary to change the network type from the default setting.



You can use the **AssistButton** to select the net type from a list, and then click **OK** to accept your selection. You can choose between TCP/IP, and NetBIOS.TCP/IP is the default network protocol.

If there is already a database open, it will be closed when you open the new one. However, if you are working in a multiuser setup, you will not be able to close a database as long as one of the other clients is working with it. Microsoft Navision Database Server can only have one database open at a time.

When you have opened the database, you can open a company by clicking FILE→COMPANY→OPEN, or you can add a new company by clicking FILE→COMPANY→NEW. You can only have one company open at a time. However, you can open and close companies regardless of whether other users are working with them.

Information about the database and the company that are open when you close Microsoft Navision is saved in the .zup file, and that database and company will be used as a default database and company when you start the program again.

You can also use the Database program property to specify the database you want when starting a server or changing to another (existing) database. You can set the Database property in the **Target** field or on the command line that starts the program, like this:

```
D:\FINSERV\SERVER.EXE DATABASE=DB.FDB
```

***NOTE**: Microsoft Navision will automatically open the database and company that you were last working on when you reopen the program.*

Automatic Reconnection

Microsoft Navision can automatically detect if the connection to the server has been broken. It will do this the next time the client tries to access the server after having been inactive for at least ten minutes. If the connection has been broken, for example, because the server has been stopped and then restarted, Microsoft Navision will attempt to reestablish the connection to the server and open the database with the same settings that were used when the database was last opened. The user can then continue to work with the database and will not notice that the server has been unavailable.

This allows you to shut down the server temporarily, for example, for hardware upgrades or modifications to server properties and then restart it, without causing any serious inconvenience.

Closing a Database

If you want to close a database (for example, before opening another), you can click FILE→DATABASE→CLOSE. However, you do not have to follow this procedure; Microsoft Navision will save all data and close everything down correctly when you select another database or quit the program. There can never be more than one database open at a time, but you can choose to close the database as an extra safety precaution if you want to delete a database or do something similar.

On servers, you can use the Stoptime program property to stop the server at a specific time of day. After this, the clients will no longer be able to access the database.

License Files

The Microsoft Navision installation comes with a demonstration license file, CRONUS.flf. The demonstration license file allows you to use the standard Microsoft Navision program as a stand-alone application and gives you access to the demonstration company that is part of the accompanying standard database, database.fdb. To start working with Microsoft Navision, you will need a license file that contains permissions for the desired application areas and functions. This license file will be supplied by your Microsoft Certified Business Solutions Partner.

You can use your license file to work in the demonstration company and in your own companies, but your permissions will be limited (even in the demonstration company) to those provided by the license file. Your license file is not subject to the same restrictions as the demonstration license file. These restrictions are listed in the section entitled Using the Demonstration License File Cronus.flf.

Your license file is always named fin.flf and is uploaded to the server during the server installation. The clients automatically work with the same license file as the server they are connected to.

Importing, Exporting and Changing License Files

On the menu bar, click TOOLS→LICENSE INFORMATION The License Information window appears:

The information displayed includes the license number of the current license file, the name of the owner, and the functionality that the owner has purchased (along with any expiration dates). If at any time you want information about the current license file, you can open this window. The buttons at the bottom of the window allow you to import and export license files and to temporarily change the license that you are using.

The license information that is displayed in this window will always be taken from the license information that is stored on the server, unless you have changed your license temporarily with the Change facility.

Importing a License File

In the License Information window, click **Import**, to use a different license file. The Import License File dialog box appears. Locate and select the license file that you want, and then click **Open**. The program will then import the license file into the Microsoft Navision folder on your computer and it will be called fin.flf. The license file will automatically replace any other file called fin.flf without asking you to confirm that you want it to do so.

This new license file will be the active license file the next time you open Microsoft Navision. When you connect to a server, the license stored there will become the active license.

Exporting a License File

Click **Export** to export a copy of your license file, for example, to a disk. The Export License File appears. This is a standard windows dialog box.

Temporarily Changing the License File

If, for example, you are a Microsoft Certified Business Solutions Partner representative visiting a customer, you may want to change the license file temporarily. To do so, insert the disk containing the license file and click **Change**. The Change License File dialog box appears. Select the license file to be read into the system. The information contained in it will be transferred to the client when you click **Open**. When you access any servers, this temporary license information will continue to be used instead of the license information stored on the servers. The server license will be reinstated when Microsoft Navision is closed and opened again.

Microsoft Navision will warn you before your license expires. If you fail to notice the warning and the license on the server expires, you will not be able to access the server. However, you can use the Change facility to gain access to the server by using an alternative license file.

When you receive your new license file, you should copy it to the server.

Ensuring that You are Using a Valid Microsoft Navision License

You may want to verify that you are using a valid Microsoft Navision license that has been issued to your company.

To check the validity of your license file:

1. Click HELP→ABOUT MICROSOFT NAVISION and the About Microsoft Navision window appears.

2. Click **Check your license information** and your Web browser opens a Web page that will help you check the validity of your Microsoft Navision license.

Test Your Knowledge – Microsoft Navision Database Server

1. What are the two different types of multiuser installations, and how are they setup?

2. There are four ways to run Microsoft Navision database servers with the same or different databases. What are they?

3. There are setup properties that apply to both the client and server. What are the five setup properties that apply only to the server?

4. What is the Microsoft Navision Database Server Manager?

5. What is the maximum size of a database? Can a database be broken into several smaller files? Why would this be done?

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 5: MICROSOFT NAVISION SQL SERVER OPTION

Overview

In order to run the Microsoft® SQL Server® Option for Microsoft® Business Solutions–Navision®, you must first install SQL Server on the server and then install the SQL Server Option for Microsoft Navision on the client computers.

Microsoft Navision comes with a default setup that enables it to be used immediately. Different installations may require small variations in the setup and can be easily implemented. Once changed, the settings will be saved and used by the program until they are altered.

The database is the heart of Microsoft Navision. All the information, companies, modifications, reports, and so on are stored here. Therefore, it is important that you know how to manage the database and that you are familiar with the tools that the SQL Server Option for Microsoft Navision provides.

This chapter describes the security system of the Microsoft SQL Server Option for Microsoft Navision. It also explains how to control the access that each user has to the information contained in the program.

Server Considerations

Microsoft Navision Setups

The two typical Microsoft Navision setups are:

- Client/server installation
- Single-user installation

The client/server installation is the most common. Microsoft Navision is installed on client computers and the database(s) are located on Microsoft SQL Server on another computer.

In this manual, we focus on a client/server setup. For more information, see the section called Single-User Installations in “Microsoft Navision Client Installation.”

SQL Server Considerations

When you install Microsoft SQL Server in a network, you can choose to install it on a file server or on a dedicated server. Installing the program on a dedicated server improves performance considerably because it provides SQL Server with exclusive access to the computer’s resources.

Tools

You do not need to install the Microsoft SQL Server Client Tools on the client computers. The appropriate ODBC driver and net libraries will be installed when you install Microsoft Navision on the client computers. These will enable the Microsoft Navision clients to communicate with SQL Server. You can also install the SQL Server Client Tools from the SQL Server setup program if you need them.

Windows Authentication Requirement

With the SQL Server Option for Microsoft Navision, you can use Windows authentication to connect to SQL Server. This is only possible when the server's operating system is Microsoft® Windows® XP, Microsoft® Windows® 2000 Server, or Microsoft® Windows Server™ 2003.

You must add an extended stored procedure to each instance of SQL Server that you want to access using Windows authentication. However, if you select the Complete Installation option of Microsoft Navision and are going to access the default SQL Server instance on your local computer with Windows authentication, the installation program will add the procedure for you.

To add the extended stored procedure from your server computer:

1. Access the Microsoft Navision Installation CD-ROM.
2. In the folder $:\SQL_ESP, where $ is the CD-ROM drive, click the file xp_ndo.exe.
3. When prompted, enter the path to the BINN subfolder of the SQL Server installation folder. The unzipped xp_ndo.dll file must be in this folder on SQL Server.

   The first time Microsoft Navision connects to the server, the program will attempt to add the extended stored procedure automatically. If this is the case, you need not go through the next step.
4. However, if you have previously connected to the server from Microsoft Navision and, therefore, bypassed the automatic feature, you have to add the extended stored procedure manually. Using a Microsoft tool such as Enterprise Manager, add the file to the extended stored procedures already installed on SQL Server. The name of the extended stored procedure must be xp_ndo_enumusersids.
5. Grant the public SQL Server database role execute permission for xp_ndo_enumusersids.

NOTE: *If you want to use Windows authentication, you must not use Windows fibres.*

Server Collation

A collation specifies the rules by which character data is sorted and compared.

When you install an instance of SQL Server, you should select the collation that matches your geographic location/language needs as closely as possible as your default server collation. We recommend that you use a Windows collation.

For more information about collations, see the section called "Collation."

Installing Clients

You use the SQL Server Option for Microsoft Navision setup program to install the program on the client computers. Microsoft Navision supports Microsoft Installer. You use Microsoft Installer to install, maintain, and remove Microsoft Navision.

Installing Microsoft Navision

Make sure that the operating system on which you will run Microsoft Navision (Microsoft Windows XP or Windows 2000) is installed on the computer. If it is not, you must install it before you can install Microsoft Navision. If you are installing Microsoft Navision from a network drive, make sure that you are connected to the network server. For further information on client installation, see "Microsoft Navision Client Installation."

Completing the Installation

To complete the installation, the license file fin.flf that you receive from your Microsoft Certified Business Solutions Partner must be uploaded to the server. With the SQL Server Option for Microsoft Navision, clients receive their permissions from the server. For more information, see the section "Uploading a License File."

Single-User Installations

Even though the SQL Server Option for Microsoft Navision is a client/server application, it is possible to use it in a single-user setup. Running a single-user installation means that you must install both SQL Server and the client application on the same computer. It must be impossible for other computers to connect to this computer. The server installation you choose is, of course, dependent on the operating system you are using.

Single-user installations can be useful for small companies who want to store their databases locally or need to be able to take their databases into the field.

We do not recommend single-user installations because they make very serious demands on your hardware and give you none of the benefits that can be gained from running a proper client/server installation.

System Setup

Program Properties

You can customize the system setup by changing the settings of the program properties.

Some settings may be entered in the **Target** field of the Microsoft Navision Properties window or at the command prompt. For more information, see the section called "Connecting Automatically."

The program properties can be specified in any order – just enter them after the programs start command and separate them by commas. The name of each program property is followed by an equal sign (=) and the value to which the property is set, for example:

```
C:\PROGRAM FILES\MICROSOFT BUSINESS SOLUTIONS–MICROSOFT
NAVISION\CLIENT\FINSQL.EXE NTAUTHENTICATION=YES, SERVERNAME=MY SERVER,
DATABASE=MY DATABASE, COMPANY=CRONUS INTERNATIONAL LTD., ID=ALICE
```

You can also set most program properties using the menu bar in the program window, for example, by clicking FILE→COMPANY→OPEN (to set Company) or FILE→DATABASE→OPEN (to set Database). The properties that do not exist as menu commands are shown in the Options window on the Tools menu:

NOTE: *Changes made in this window as well as changes made to the other parameters via the* ***Target*** *field or command prompt are saved in the setup file (the zup file) and will be used the next time the program is opened. However, the zup file does not store information about the company that you were working with. If you do not want users to be able to make permanent changes to these options, you can set default values in the command line of a batch file (for example, fin.bat) that they use to start the program.*

The following table lists the different program properties:

Property	Described on
Server Name	Page 128
Database	Page 128
Company	Page 129
Windows Authentication	Page 130
ID – Saving the User Setup	Page 131
Object Cache	Page 132
Net Type	Page 133
TempPath	Page 133

Property	Described on
DB Test	Page 134
TestTarget	Page 134
Status Bar	Page 135
Close Form on Esc	Page 135
Marquee Full Selection	Page 136
Quick Find	Page 136

Server Name – Choosing the Server

Program Property	Purpose	Where Specified	Default Value	Value
Server Name	Specifies which server to connect to.	Can be selected by clicking FILE→DATABASE→OPEN or entering servername=My Server in the Command line or the **Target** field. Only works in combination with database=	None	Name of server

This program property is used to specify the server that a particular client will connect to.

You can set up the connection to the server in the **Target** field by writing the name of the server after servername=.

You can connect to a server from within Microsoft Navision by clicking FILE→DATABASE→OPEN on the menu bar. For more information on connecting to a server, see the section called "Opening Databases."

To see which server you are currently connected to, click FILE→DATABASE→INFORMATION and click the **Database** tab.

Database – Selecting a Database

Program Property	Purpose	Where Specified	Default Value	Value
Database	Specifies which database to open.	FILE→DATABASE→OPEN or by entering database=My Database in the Command line or the **Target** field. Only works in combination with servername=	None	Name of database

The Database program property is used to make the program start with a particular database open. (The database must already exist.) In the **Target** field or on the command line that starts the program, enter the name of the database immediately after database=.

To open a database from within Microsoft Navision:

1. Click FILE→DATABASE→OPEN.
2. In the window that appears, select the server and the database that you want to open.
3. Click FILE→DATABASE→INFORMATION to see which database is being used.

When you select a standard database for a client, you can also select which company will open automatically by using the Company program property.

Company – Selecting a Company

Program Property	Purpose	Where Specified	Default Value	Value
Company	Specifies which company to open.	FILE→DATABASE→OPEN or by entering company=CRONUS International Ltd. in the Command line or the **Target** field. Only works in combination with servername= and database=	None	Company name

With this program property, you can select the company that will open automatically when a client starts the SQL Server Option for Microsoft Navision. Before you specify the company for any client, you must specify the server and the database containing the company in the **Target** field. If you do not specify the server or database, the program will use the server and database that were last opened and have been saved in the zup file.

From within the program, you can select a company by clicking FILE→COMPANY→OPEN. You can also select a company from the list displayed at the bottom of the File menu. You can see the current company on the title bar of the program window.

Windows Authentication – Selecting the Authentication Mode

Program Property	Purpose	Where Specified	Default Value	Value
NTAuthentication	Specifies which type of authentication to be used.	FILE→DATABASE →OPEN or by entering ntauthentication= yes in the Command line or the **Target** field.	Yes	Yes/No

This program property is used to determine which type of authentication is to be used when logging on to a server and opening a database.

After selecting the server and the database in the Open Database window, you must:

1. Select the type of authentication that is to be used.
2. Enter your user ID and your password, if database server authentication is being used. If Windows authentication is being used, you do not have to enter a password or user ID.
3. Click **OK**.
4. In the **Target** field or on the command line that starts the program, enter "yes" or "no" after ntauthentication=.

If you are using Windows authentication, the SQL Server Option for Microsoft Navision will start, automatically connect to the server, and open the database that you have specified.

If Database authentication is being used, the SQL Server Option for Microsoft Navision will start and prompt you to supply your user ID and password before connecting to the server and opening the database.

For more information about the types of authentication used in the SQL Server Option for Microsoft Navision, see the section called Security and User Setup.

ID – Saving the User Setup

Program Property	Purpose	Where Specified	Default Value	Value
ID	Saves individual setup	Command line or the **Target** field. ID=alice	Fin	Name of ID (including path if ID file is not located in Microsoft Navision folder.)

Each user in a Microsoft Navision multiuser installation can choose the setup of windows and program properties that he or she wants to use. In order for the program to be able to save and use the setup selections of the various users, each user must have a unique identification code. You can create a user setup by starting the program with an ID. The information about the users setup will be stored under this ID. Here is an example in which the program starts with an ID called SUPER:

```
C:\PROGRAM FILES\MICROSOFT BUSINESS SOLUTIONS–MICROSOFT
NAVISION\CLIENT\FINSQL.EXE ID=SUPER
```

In a list of Microsoft Navision program files, you can see that each time you have started with a new setup ID, a file has been created that has the ID as the first part of the file name and .zup as the file name extension (for example, super.zup). This is called a setup file.

You can make a standard setup created for a particular type of user available to other users with similar needs. To do this, copy the appropriate setup file to the folder from which the user starts the program, and enter its ID (file name) in the **Target** field. Alternatively, you can put the setup file on a common drive in the network (but here it can be overwritten by other users). If you choose to place the setup file in a folder other than the one containing the program files, you must remember to specify the entire path name after id=.

The setup ID is not the same as the user IDs in Microsoft Navision, but you can make them look alike by entering the same characters. In fact, it can be an advantage to do so because although you cannot see the name of the setup file in the program, you can always see the user ID on the status bar at the bottom of the program window.

If you do not enter a setup ID, the program will use a setup file called fin.zup when the clients start the program. Because several users can use the fin.zup file at the same time, you risk running into conflicts when you save the fin.zup file to the disk (by closing Microsoft Navision). When each user closes the program, the setup changes they have made will be saved in the fin.zup file. The fin.zup file that was saved by the previous user who closed the program will be replaced. The program will notify you of this with a message in the window when you log off. You will be able to choose whether or not you want to save your latest modifications.

Returning to the Original Setup

You can always return to the standard setup file fin.zup, by starting the program without specifying a setup ID.

If you have previously worked without a setup ID and made changes in the setup, the fin.zup file will contain these changes. If you do not want to use this modified fin.zup file but would prefer to return to the original starting point of the program, delete the fin.zup file and start the program again without an ID. The program will create a new, clean setup file, named fin.zup.

Object Cache – Improving Response Times

Program Property	Purpose	Where Specified	Default Value	Value
Object Cache (KB)	Makes the program run faster	TOOLS→OPTIONS or by entering objectcache=10,000 in the Comment line or the **Target** field.	10,000 (KB)	More than 0 KB and less than 1,000,000 KB

The Object Cache property increases the speed of the program. Objects such as code, descriptions, and windows that will be used on the client computer are stored in the object cache. This means that the client computer only needs to retrieve these objects once from the server and then they will be stored in the object cache. The client computer must have enough memory to store the objects while they are being used in order to benefit from the object cache.

Running out of object cache (that is, setting too small a value) does not cause any problems. The total size of all the objects used in the standard application is around 20 MB. If you have enough memory, set the object cache to 20 MB. The size of the most important objects, for example, the table descriptions, is 1 MB. You should therefore, as a minimum, set the object cache to 1 MB. The upper limit is 1 GB.

Click FILE→DATABASE→INFORMATION to see how much space the object cache has been allocated in the **Object Cache (KB)** field. To change the amount of space allocated to the object cache, on the menu bar, click TOOLS→OPTIONS, and enter the setting in the **Object Cache (KB)** field.

NetType – Selecting a Net Type

Program Property	Purpose	Where Specified	Default Value	Value
Net Type	Permits choice of network protocol	FILE→DATABASE→OPEN→ADVANCED or FILE→DATABASE→NEW→ADVANCED or by entering NetType= in the Command line or the **Target** field.	Default. The default net type can be specified with the SQL Server Client Network Utility.	Default, Named Pipes, TCP/IP Sockets, Multiprotocol

With the SQL Server Option for Microsoft Navision, you can select the net type that the server and the clients use to communicate with each other. The net type that is used for the default setting can be set from the client by using the Client Network Utility, which is a part of the SQL Server Client Utilities, if these have been installed.

On the menu bar, click FILE→DATABASE→OPEN. Then click **Advanced** on the window that appears, and select the net type in the **Net Type** field. Click **OK**. To check this setting when you are using the program, click FILE→DATABASE→INFORMATION, and look on the **Database** tab.

TempPath – Location of Temporary Working Files

Program Property	Purpose	Where Specified	Default Value	Value
TempPath	Specifies the location of temporary working files created automatically.	TOOLS→OPTIONS or by entering Temppath= in the Command line or the **Target** field.	Windows 2000: C:\Documents and Settings\User Name\Application Data\Local Settings\Temp Windows XP: C:\Documents and Settings\User Name\Local Settings\Temp	Name of ID (including path if ID file is not located in Microsoft Navision folder.)

When Microsoft Navision is running, it creates a number of temporary files, which are automatically deleted when the program is closed. As a default, the temporary files of each individual user are stored in C:\DOCUMENTS AND SETTINGS\USER NAME\LOCAL SETTINGS\TEMP, unless you specify a different working folder. If you do so, this working folder will be the default location. You can specify the working folder in the **Target** field or by clicking TOOLS→OPTIONS. You must specify the full path, including the drive and all folders.

DB Test – Testing the Database

Program Property	Purpose	Where Specified	Default Value	Value
DB Test	Tests the database	FILE→DATABASE→ TEST or by entering dbtest=min in the Command line or the **Target** field.	None	Min, Normal or Max

You can use this program property to test the consistency and integrity of the database. You can also run the test from within the program by clicking FILE→DATABASE→TEST. If you do this, you will be able to use a detailed dialog box in which you can specify exactly what you want to test. When you enter the DB Test program property in the **Target** field, the database will be tested before the program opens. You can specify one of the following options:

- dbtest=min (Run this before any backup that includes database files.)
- dbtest=normal (Use daily.)
- dbtest=max (Use about once a month – very time-consuming.)

For more information about the extent of these tests and how to create a customized version of the database test, see "Microsoft Navision Database Administration Tools."

TestTarget

Program Property	Purpose	Where Specified	Default Value	Value
TestTarget	To specify how error messages generated by the database test are managed.	FILE→DATABASE→ TEST or by entering testtarget=@screen in the Command line or the **Target** field.	@screen	@screen, @eventlog, filepath

You use this program property to specify how any error messages that are generated during a database test are managed. They can be displayed on the screen or stored in the Event Log or in a text file. You can enter one of the following options:

- testtarget=@screen
- testtarget=@eventlog
- testtarget=filepath

You must enter the full path and the name of the text file. If you select event log, you can read the error messages that were generated during the database test in the Windows Event Viewer. If you select screen, the error messages will be displayed on the screen, and the database test will require interaction from the user if any errors are found. Selecting screen can make the database test quite time consuming.

Status Bar

Program Property	Purpose	Where Specified	Default Value	Value
Status Bar	Activates or deactivates the status bar.	TOOLS→OPTIONS	Yes	Yes/No

By clicking TOOLS→OPTIONS on the menu bar, each user can choose whether or not the status bar will be displayed at the bottom of the program window.

The status bar shows the complete name of the active field and its contents, the work date, the current user ID, whether or not any filters have been placed on the data (FILTER appears), whether or not you are about to create something NEW (an account, for example) and finally, whether you are working in Insert (INS) or Overtype (OVR) mode. When you make a visible change in the setup (such as changing whether the status bar is visible), it is practical to use the ID program property and a setup file on your own computer. This makes the setup selections valid only for yourself. For more information, see the section called "ID – Saving the User Setup."

This property can only be adjusted from within Microsoft Navision.

Close Forms on Esc

Program Property	Purpose	Where Specified	Default Value	Value
Close Forms on Esc	Determines whether windows close when you press Esc.	TOOLS→OPTIONS	Yes	Yes/No

By clicking TOOLS→OPTIONS on the menu bar, you can choose whether or not the window you are working in will close when you press **ESC**.

It is practical to use the ID program property and have a setup file on your own computer if you change the appearance of the setup (such as changing the setting of this program property). This makes the setup selections valid only for yourself.

For more information, see the section "ID – Saving the User Setup."

This property can only be adjusted from within Microsoft Navision.

Marquee Full Selection

Program Property	Purpose	Where Specified	Default Value	Value
Marquee Full Selection	Determines how graphical objects are selected on the screen.	TOOLS→OPTIONS	No	Yes/No

With this setting, you can choose whether graphical objects must be completely within the rectangle in order to be selected or whether it is sufficient for them just to touch the edges.

This property is relevant for developers using the C/SIDE® development environment. You make this selection by clicking TOOLS→OPTIONS on the menu bar.

This property can only be adjusted from within Microsoft Navision.

Quick Find

Program Property	Purpose	Where Specified	Default Value	Value
Quick Find	Quick search by letter in all windows.	TOOLS→OPTIONS	Yes	Yes/No

This setting allows you to activate a quick search facility.

When the Quick Find setting is enabled, you can search for an entry in any non-editable field, by typing a letter or number. You can also enter the entire name of the element you are looking for. When you enter a letter or number, the Find window opens automatically, and the first row that matches what you entered becomes the active row.

When the Quick Find property is disabled, you can open the Find window by clicking EDIT→FIND on the menu bar or by clicking Find on the toolbar.

The Quick Find property can only be adjusted from within Microsoft Navision.

Connecting Automatically

Many of the properties that are described in the this chapter can be entered as command lines after the command prompt or included as program properties that are automatically set when you start Microsoft Navision.

To carry out this procedure, you must have administrative rights on the computer. To set them as automatic program properties:

1. Open Windows Explorer.
2. Open the folder DOCUMENTS AND SETTINGS\ALL USERS\START MENU\ PROGRAMS\MICROSOFT BUSINESS SOLUTIONS – MICROSOFT NAVISION.
3. Notice that there are two identical Microsoft Navision icons on the right. You must select the icon for the Microsoft Navision executable that you are using.

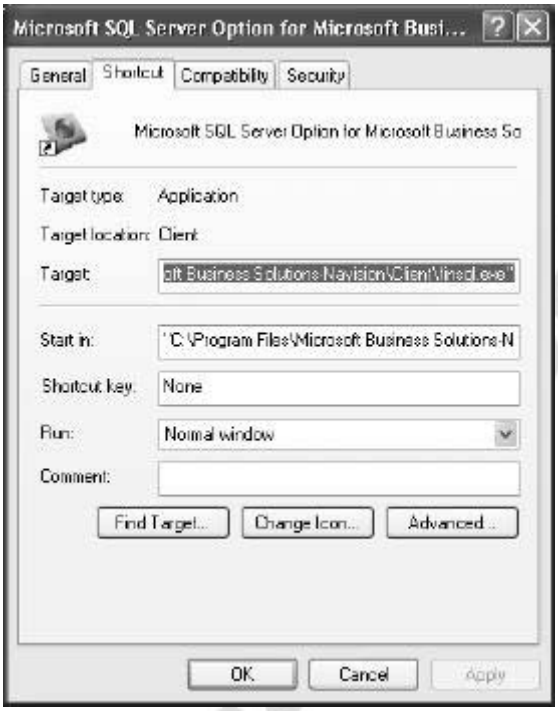
***IMPORTANT**: If you are using Windows XP and Windows 2000 you MUST delete these existing shortcuts and create new ones in order to gain access to the **Target** field. New shortcuts should be made from the Microsoft Navision Client folder and then copied into the same path noted in step 2.*

Also note that by copying the shortcut to the user specific folders, you can tailor the program properties to the individual user.

---

4. Click FILE→PROPERTIES, or right-click the Microsoft Navision icon and select **Properties**.

The Microsoft SQL Server Option for Microsoft Business Solutions – Navision window appears:



5. Click the **Shortcut** tab. The **Target** field shows where Microsoft Navision is located. It contains the path for the start command fin.exe. After the start command, you can add other commands and settings for program properties. Here is an example:

SERVERNAME=MY SERVER, NETTYPE=TCP, COMPANY=CRONUS INTERNATIONAL LTD.

When you set the program properties in this way, the program performs certain tasks the next time it is opened. It will use TCP/IP to connect to the server called My Server and open the company called CRONUS International Ltd., if it exists in the database. If you do not enter the company parameter, you will have to open the company manually after the program starts.

You can change any of these selections while you are working. You can do this from within the program. For example, you can select a different database (provided it has already been created) or a different company, or you can create a new company. If you do not want a client to be able to do these things, you can set limits when you assign user permissions (by setting limits on the "system" object type). You can read about assigning user permissions in "Microsoft Navision Security."

Working with Databases

License Files

The Microsoft Navision installation comes with a demonstration license file, CRONUS.flf. The demonstration license file allows you to use the standard Microsoft Navision program as a stand-alone application and gives you access to the demonstration company that is part of the accompanying standard database, database.fdb. The demonstration license file also allows you to work with the SQL Server Option for Microsoft Navision. The demonstration license file contains the following restrictions:

Restrictions of CRONUS.flf:

- Posting is only possible in the period November to February.
- The maximum number of companies is two.
- You can have a maximum of two sessions running at any time.
- Any company name must start with CRONUS (written in capital letters). This ensures that it will be clearly identifiable as a demonstration company – and you will not accidentally create a "real" company with the wrong license file.

You can use your own license file to work in the demonstration company and in your own companies, but your permissions will be limited (even in the demonstration company) to those provided by the license file. Your license file does not contain the restrictions of the demonstration license file. Your license file is always named fin.flf.

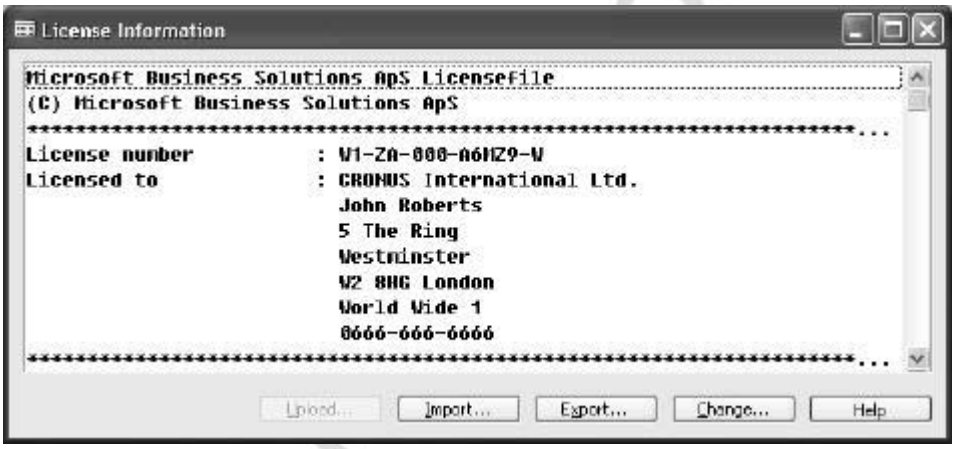
In order to work with the SQL Server Option for Microsoft Navision, you must upload the license file you want to use to all the SQL Servers you want to access, instead of copying it to the individual clients. The clients automatically work under the same license file as the server they are connected to. By default, the demonstration license file CRONUS.flf is uploaded to a SQL Server when the server is used for the first time. SQL Server 2000 allows you to run multiple server instances on the same computer. The Microsoft Navision license information must be uploaded to each server instance you want to access.

The SQL Server Option for Microsoft Navision allows you to specify that every database that is stored on an instance of SQL Server must use the same license file or that the individual databases can have their own license file. You specify that a database must use its own license file by clicking the **Save license** in **Database** field when you create or alter the database. This license file must contain the per-database granule or it cannot be stored in the database. The database will use the granules (including the sessions) specified in the per-database license file.

In addition to the license files stored in the individual databases on the server, a common license file must also be stored on the server. The common license file can be identical to one of the license files stored in the databases or it can be a different license file. Any databases that you create on this server that do not contain a per database license file will use this license file. These databases will use the granules (including the sessions) specified in the server's license file. For more information about creating a database, see the section "Creating a Database."

Uploading, Importing, Exporting and Changing License Files

On the menu bar, click TOOLS→LICENSE INFORMATION. The License Information window appears:



The information displayed includes the license number of the current license file, the name of the owner and the functionality that the owner has purchased (along with any expiration dates). If at any time you want information about the current license file, you can open this window. The buttons at the bottom of the window allow you to upload, import, and export license files and to temporarily change the license that you are using.

The license information that is displayed in this window will always come from the license information that is stored on the server, unless you have changed your license temporarily. For more information, see the section called "Changing the License File."

Uploading a License File

To upload a license file to SQL Server, in the License Information window click **Upload**. In the Upload License File window that appears, you can browse to the license file provided by your Microsoft Certified Business Solutions Partner. Select the license file, and click **Open** to upload it to the server. This license information will be used by all the clients connecting to this server, unless they change their license information temporarily with the Change function or unless a database license file is being used.

After you have uploaded the license file to the server, store a copy of the license file in a safe place until you need to use it again.

If you receive a new license file that you want all your Microsoft Navision clients to use, you must upload it to SQL Server. Click **Upload** in the License Information window. The Upload License File window appears. Locate and select the license file, and then click **Open** to upload it. The license file is uploaded to the server or to the database depending on the license option that you chose when you created the database.

Importing a License File

On the License Information window, click **Import** to use a different license file. The Import License File dialog box appears. Locate and select the license file that you want, and then click **Open**. The program will then import the license file into the Microsoft Navision folder on your computer, and it will be called fin.flf. The license file will automatically replace any other file called fin.flf without asking you to confirm that you want it to do so.

This new license file will be the active license file the next time you open Microsoft Navision. When you connect to a SQL Server, the license stored there will become the active license.

Exporting a License File

Click **Export** to export a copy of your license file, for example, to a disk. A standard Windows dialog box appears. Locate and select the license file, and then click **Save** to export it.

Changing the License File

If, for example, you are a Microsoft Certified Business Solutions Partner representative visiting a customer, you may want to change the license file temporarily. To do so, insert the disk containing the license file and click **Change**. The Change License File dialog box appears. Select the license file to be read into the system. The information contained in it will be transferred to the client when you click **Open**. When you access any servers, this temporary license information will continue to be used instead of the license information stored on the servers or in the databases. The original licenses will be reinstated when Microsoft Navision is closed and opened again.

Replacing an Expired License

Microsoft Navision warns you before your license expires. If you fail to notice the warning and the license on the server expires, you will not be able to access the server. However, you can use the Change facility to gain access to the server by using an alternative license file.

When you receive your new license file, you should use the Change facility to temporarily change to the new license. You can then connect to the server and use the Upload function to upload your new license to the server.

Ensuring that you are using a valid Microsoft Navision License

You may want to verify that you are using a valid Microsoft Navision license that has been issued to your company.

To check the validity of your license file:

1. Click **Help**, **About Microsoft Navision**, and the About Microsoft Navision window appears.
2. Click **Check your license information**, and your Web browser opens a Web page that will help you check the validity of your Microsoft Navision license.

Creating and Maintaining Databases

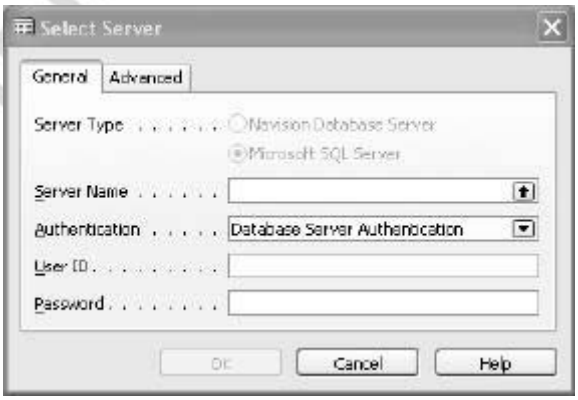
You cannot work with Microsoft Navision without a database. When you have installed your client application, there is no database available. You must first create a database on the server from within Microsoft Navision.

***NOTE**: To create or alter a database, the administrator must be a member of either the sysadmin or the dbcreator SQL Server server roles. To create a database, the administrator must also have public access to the model database. To alter a database, the administrator can alternatively be a member of the db_ddladmin or db_owner database roles for this database. For more information about security and SQL Server roles, see the "Security and User Setup" section.*

Creating a Database

To create a new database:

1. Click FILE→DATABASE→NEW and the Select Server window appears:



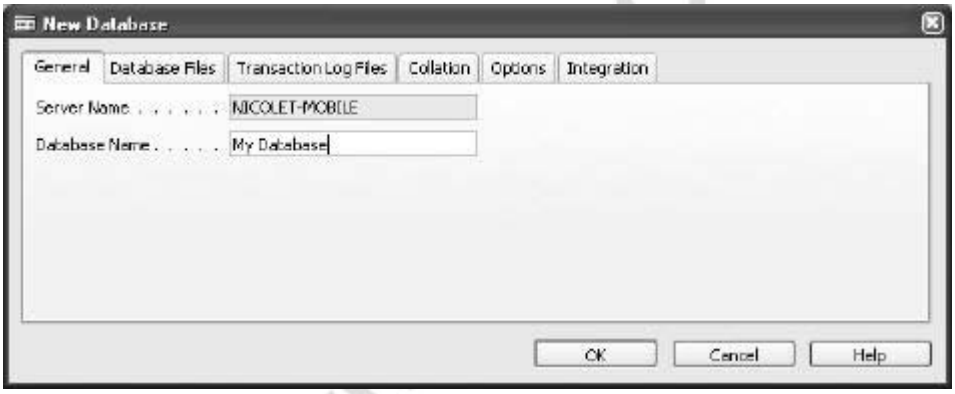
2. In the **Server Name** field, enter the name of the server. You can use the **AssistButton** to browse a list of available SQL Servers, or you can type the server name.

3. In the **Authentication** field, use the **AssistButton** to select the type of authentication that you will use. If you select database server authentication, you must enter your User ID and password.

4. Click **OK** to connect to the server and proceed with the creation of the database.

The **Advanced** tab contains information about the net type that is used to communicate with the server. This is set to the default net type, and does not generally need to be changed. For more information about any of the fields contained in this window, see the section called "Opening Databases."

The New Database window appears:

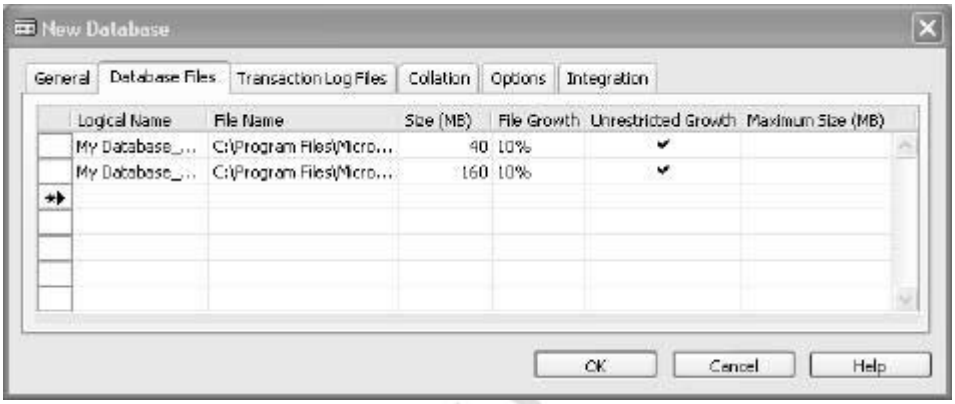


General

On the **General** tab, type the name of the database in the **Database Name** field. Clicking **OK** creates the new database with standard settings.

NOTE: *You can move from tab to tab by using the mouse or by using CTRL + PGUP and CTRL + PGDN.*

Default Database Configuration
Microsoft Navision will configure two data files and one transaction log file for new databases. The first data file is created as the primary file in the PRIMARY file group. The second data file and every subsequent file that is added will be created in an additional file group. The primary data file is always the first file listed on the **Database Files** tab and must always be specified.



As a default, the size of the primary data file will be set at 40 MB or at the size of the primary data file in the model database, whichever is the greater. The size of the secondary data file will be set at 160 MB. The size of the transaction log file will be set at 50% of the sum of the primary and secondary data files.

These settings allow you to restore the standard database backup that comes with the program to a new database without causing the data or transaction log files to grow. You can change all of these sizes and all of the other file properties so that they meet your requirements before you create the database.

Database Files

The **Database Files** tab lets you control the location, size, growth, and maximum size of the data files. We recommend that you split the database into multiple files by specifying multiple names. For more information about, see the section called "Database Configuration Guidelines." The following table contains guidelines for using the fields in the **Database Files** tab:

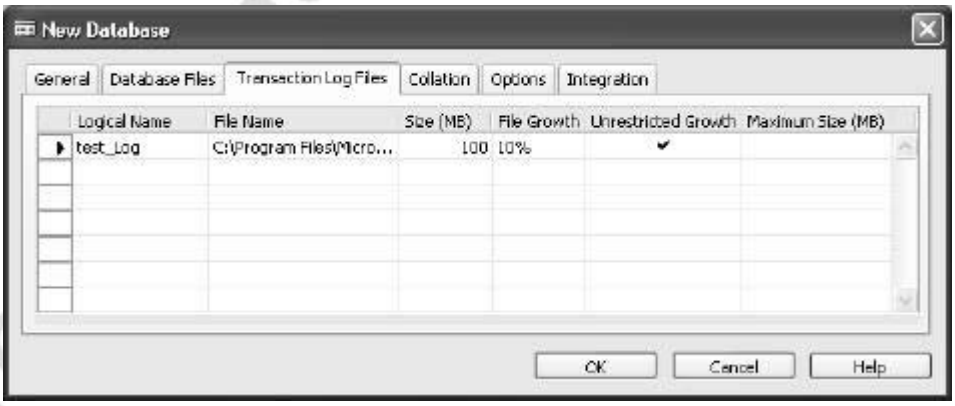
Field	Comment
Logical Name	Specify a logical name for each data file. Microsoft SQL Server uses logical names to facilitate easier management of these files.
File Name	Specify the full path and name of each physical file that makes up the database on the server. We recommend that the first file, which is the primary file, has the extension .mdf and that all other files have the extension .ndf. The path must be a valid folder on a server hard disk.
Size (MB)	Specify the size of each data file in megabytes.

Field	Comment
File Growth	Specify the amount by which the data file will increase in size each time it expands. You can express this in kilobytes (by using the suffix KB), megabytes (by using the suffix MB), or as a percentage (by using the suffix %) of its size at the time it increases. If you enter zero, the file will not be able to grow.
Unrestricted Growth	Specify whether SQL Server is able to expand the data file. This expansion is only limited by the available disk space.
Maximum Size (MB)	Specify the maximum size of the data file in megabytes if the **Unrestricted Growth** field has not been selected. You must allow the data file to grow without restriction or set a maximum size.

We recommend that you give your database needs careful consideration before determining the size of your database. Expanding your database can take up considerable time and resources.

Transaction Log Files

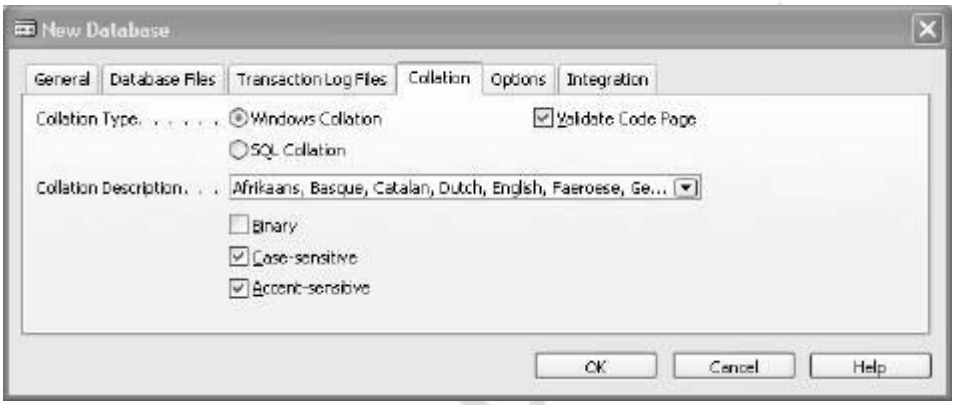
The **Transaction Log Files** tab lets you control the location, size, growth, and maximum size of all the transaction log files. The transaction log is used to track the changes that are made to the database and for database recovery. For more information on this topic, see the section called "Database Configuration Guidelines."



The fields listed on this tab have a similar purpose to those on the **Database Files** tab. We recommend that the transaction log files have the extension .ldf. For further information about the transaction log, consult Microsoft's SQL Server documentation.

Collation

The **Collation** tab lets you control the type of collation that will be used by the new database. A collation specifies the rules by which character data is sorted and compared. SQL Server 2000 allows you to specify the collation at both database and column level. The SQL Server Option for Microsoft Navision only allows you to specify the collation at database level.



The SQL Server Option for Microsoft Navision allows you to choose between Windows collations and SQL collations:

- A Windows collation corresponds to the collations supported by the Windows operating systems, where they are known as Regional and Language Options.
- SQL collations are the original collations introduced in SQL Server 7.0 and are still supported for backwards compatibility.

We recommend that you use a Windows collation when you create a database. This type of collation closely follows the collation rules of the operating system.

When you are creating a new database, the **Collation** tab displays the default server collation settings. If the server collation is a Windows collation, this collation will be used as the default collation for the database. If the server collation is a SQL collation, then a case and accent sensitive Windows collation for the English language will be used as the default collation for the database.

You should always choose the collation settings that match your requirements when you create a database.

Before you create the database, you can modify the collation to suit your requirements. If you have selected the Validate Code Page option, the **Collation** tab only displays the collation descriptions that are supported by the operating system that is installed on the client computer that is being used to create the database. That is to say it displays those collations that match either the OEM or ANSI code pages that are used by the client computer. If you have not selected the Validate Code Page option, the **Collation** tab displays all the available collations.

If you select Windows Collation, you should choose a collation that matches your geographic location/language needs as closely as possible. The binary, case sensitive and accent-sensitive properties of the collation that you select can be modified. Selecting Binary will clear both the Case-sensitive and Accent-sensitive properties. Selecting either the Case-sensitive or Accent-sensitive properties will clear the Binary property. Use the **AssistButton** to see the Windows collation descriptions that are available.

When you select SQL Collation, the binary, case-sensitive, and accent-sensitive properties are inactive because they are included in the collation description, and you should choose a collation description that matches your code page and the required binary/dictionary and case-/accent-sensitive characteristics. Use the **AssistButton** to see the SQL collation descriptions that are available.

Each new database you create can use a different collation. After the database has been created, you can change the collation that it uses by clicking FILE→DATABASE→ALTER. For more information about altering the database and changing the collation, see "Altering the Database."

The **Collation** tab also contains an option called Validate Code Page. This option is selected by default. If this option is selected, every time a client connects to the database, the OEM or ANSI code page that is used by the client computer is checked to make sure that it is compatible with the code page used by the database. If this option is not selected, the code page that is used by the client computers is not validated.

You can disable this option if you are sure that every character is converted correctly between all the clients and the database. Disabling this setting allows clients that are using different regional settings (code pages) to use the same database even though some characters entered by one client may not be interpreted correctly by another client or by the server.

NOTE: *An "incompatible" client is a client that uses a different code page than the database.*

Other problems that can be caused by not validating the code page are:

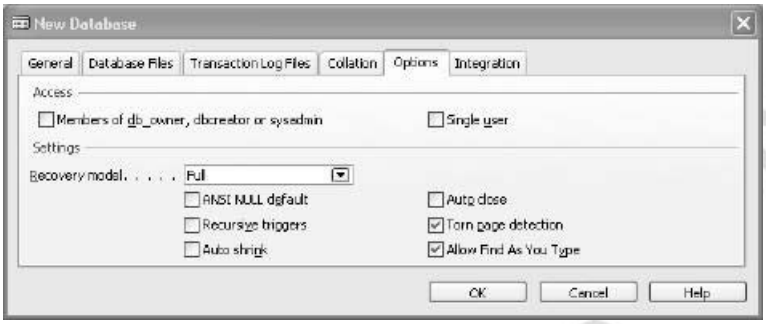
- The database server governs the sorting of textual data, and this means that the data may not be sorted according to the rules specified on the "incompatible" client computers. This problem will be more acute if there is some C/AL code that only works correctly when a particular sort order is selected.
- If you are accessing SQL Server with external tools, these tools may not be able to read the data that has been entered by the "incompatible" clients correctly.

Therefore, we recommend that you use the default setting and validate code pages because this will avoid all these problems.

For more information about collations, see Microsoft's documentation.

Options

The **Options** tab contains advanced settings for the database. These options can be changed later. For more information about changing the database, see the section "Altering the Database."



The first two settings are used for limiting the users who have access to the database:

Field	Comment
Members of db_owner, dbcreator or sysadmin	If you select this option, only members of the db_owner fixed dbcreator or sysadmin database role, the dbcreator fixed server role, or the sysadmin fixed server role have access to the database.
Single User	If you select this option, only one user can access the database at a time. This setting should be used when you are carrying out administrative functions such as testing the database, restoring a backup, altering the database, and renaming a company. Remember to clear this check box when you are finished.

The settings are used for determining the characteristics of the database:

Field	Comment	
Recovery model	This setting determines the kind of information that is written to the transaction log and, therefore, the kind of recovery model that you want to use in this database. The options are:	
	Bulk-Logged	If you select Bulk-Logged, the transaction log will only contain limited information about certain large-scale or bulk copy operations. The Bulk-Logged recovery model provides protection against media failure combined with the best performance and the minimal use of log space for certain large-scale or bulk copy operations. The backup strategy for Bulk-Logged recovery consists of: Database backups. Differential backups (optional).
	Full	If you select Full, the details of every transaction are stored in the transaction log, and this information can be used when you apply transaction log backups. The Full recovery model uses database backups and transaction log backups to provide complete protection against media failure. If one or more data files are damaged, media recovery can restore all the committed transactions. Incomplete transactions are rolled back. Full recovery allows you to recover the database to the point of failure or to a specific point in time. All the operations, including bulk operations such as SELECT INTO, CREATE INDEX and bulk loading data, are fully logged to guarantee that the database is completely recoverable. The backup strategy for Full recovery consists of: Database backups. Differential backups (optional). Transaction log backups.
	Simple	If you select Simple, the database can be recovered to the point at which the last backup was made. However, you cannot restore the database to the point of failure or to a specific point in time. To do that, choose either the Full or Bulk-Logged recovery model. The backup strategy for Simple recovery consists of: Database backups. Differential backups (optional).

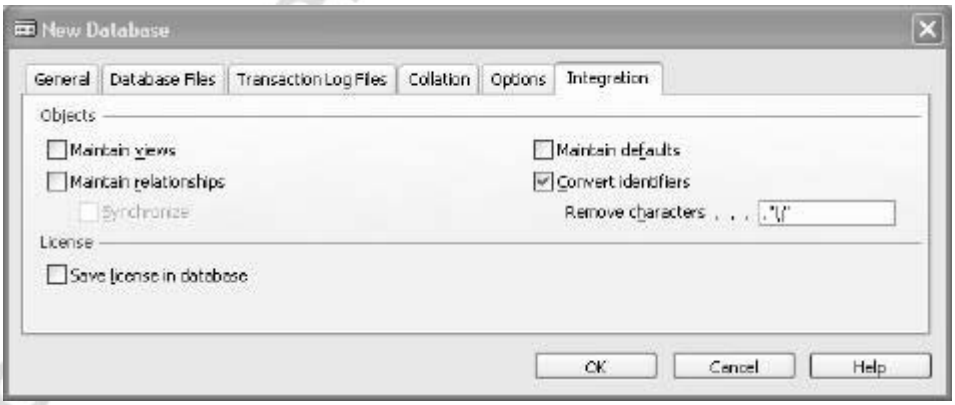
Field	Comment
ANSI NULL default	This setting controls the database default NULL settings for column definitions and user-defined data types. When you select this option, all user-defined data types or columns that have not been explicitly defined as NOT NULL default to allow NULL entries. Columns that have been defined with constraints follow the constraint rules regardless of this setting.
Recursive triggers	When you select this option, triggers fire recursively. Triggers can have two different types of recursion: Direct recursion, which occurs when a trigger fires and performs an action that causes the same trigger to be fired again. Indirect recursion, which occurs when a trigger fires and performs an action that causes a trigger on another table to fire. This second trigger updates the original table, causing the first trigger to fire again.
Auto close	This setting is used for determining whether or not the database is closed and shut down properly when all processes in the database are complete and the last user exits the database, thereby freeing resources. The auto close option is useful for databases using the SQL Server Desktop Edition because it allows database files to be managed as normal files. They can be moved, copied to make backups, or even e-mailed. The auto close option should not be used for databases that are accessed from an application that continuously makes and breaks connections to SQL Server. Closing and reopening the database between each connection will impair performance.
Torn page detection	When you select this option, SQL Server can detect incomplete Input/Output operations that have been caused by power failures or other system outages. Torn pages are usually detected during recovery because any page that was written incorrectly is likely to be read by recovery. If a torn page is detected, an I/O error is raised, and the connection is terminated. If a torn page is detected during recovery, the database is marked suspect. The database backup should then be restored, and any transaction log backups should be applied.

Field	Comment
Auto shrink	This setting determines whether or not database files are subject to periodic shrinking. Both data files and transaction log files can be shrunk automatically by SQL Server. When you select this option, SQL Server will automatically shrink a data file or transaction log file when more than 25 percent of the file is taken up by unused space. The file is shrunk until only 25 percent of the file consists of unused space, or it shrinks to the size of the file when it was created, whichever is greater. Microsoft Navision performs slightly better when this setting is not selected.
Allow Find As You Type	This setting determines whether or not you can use the Find As You Type option when using the Find function to find an entry in a table or form. Using the Find As You Type facility can affect performance because requests are sent to the server for each character that is typed.

For more information about all of these options except Allow Find As You Type, see "Microsoft's SQL Server documentation."

Integration

The **Integration** tab contains database settings that affect the way Microsoft Navision integrates with SQL Server and external tools. These options can be changed later.



The **Integration** tab is divided into two sections and contains the following settings:

Field	Comment
Maintain Views	This setting determines whether or not SQL Server will create and maintain a view for each language ID that is added to a table or field in Microsoft Navision. If you select this option, external tools can use the SQL views to gain access to the Caption ML property of the object in the required languages rather than the name supplied in the table.
Maintain Relationships	This setting determines whether or not SQL Server will create and maintain foreign key constraints for each TableRelation property of a Microsoft Navision table. If you select this option, external tools will have access to the table relationships (foreign key constraints) that exist between the Microsoft Navision tables. These relationships are disabled and are not used to enforce data integrity but are intended for modeling purposes only. For more information about table relationships in the SQL Server Option, see the manual Application Designer's Guide.
Synchronize	This setting is linked to the Maintain Relationships setting and is only active if you have already decided to create and maintain the table relationships on SQL Server. This option is only available after table relation errors have occurred while you were restoring a database or importing some objects into a database. The table relation problems can be resolved by, for example, deleting the table relations that refer to non-existent tables and then using this option to synchronize the table relations with SQL Server. If table relation errors occur when you are restoring a database, the restore will not be aborted. For more information about table relationships in the SQL Server Option, see the manual Application Designer's Guide.
Maintain Defaults	This setting determines whether or not SQL Server will create and maintain default constraints for each field of a Microsoft Navision table. If you select this option, external tools can use the defaults when inserting data into or modifying data in Microsoft Navision tables.

Field	Comment
Convert Identifiers	This setting allows you to select the invalid characters in the names of all the SQL Server objects (tables, columns, constraints) in the database and map them to the underscore character. The **Remove characters** field contains a list of the characters that are converted to underscores. You can modify this list. When the conversion is completed, the database must be closed and reopened before you can use the new identifiers. For more information about identifiers and SQL Server, see section 6.4 of the manual Application Designer's Guide.
Save license in database	This setting allows you to specify that the license file is uploaded and stored in the database instead of on the server. This is useful if you are hosting several databases with separate license files on the same server. If you select this option when you are creating or altering a database, you will be prompted to upload the license file to the database.

NOTE: *If you have converted a database from an earlier version of Microsoft Navision and you now want to convert these identifiers, you will not be able to revert to the current identifier mapping. Turning this option off again will not allow you to revert to the current identifier mapping.*

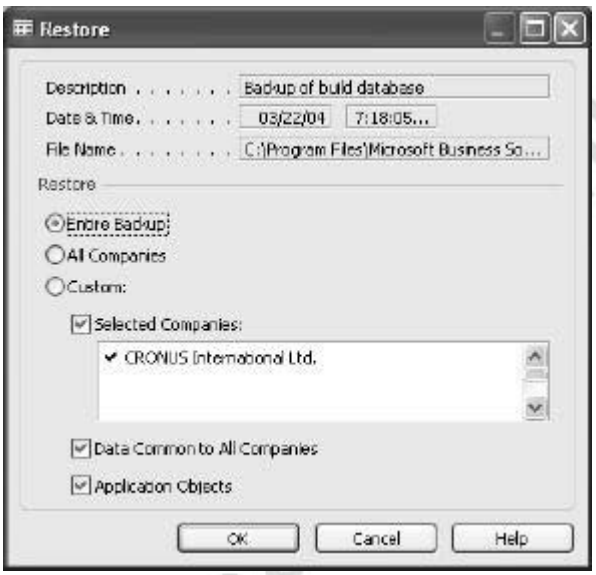
Restoring the Standard Database

The database that you have just created contains only a few basic tables and is not yet ready for use in Microsoft Navision. Before it can be used, you must restore a backup of the original standard database (database.fbk) into the new database.

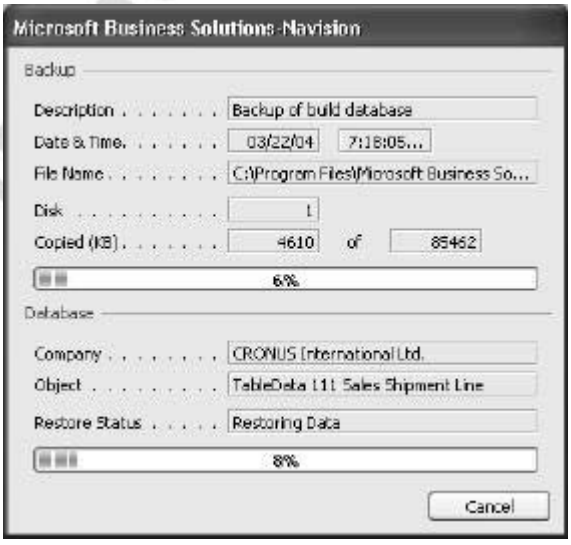
This backup comes with the SQL Server Option for Microsoft Navision and is stored in the Microsoft Navision folder on the client computer if you have carried out a complete installation. The backup contains all the information necessary for using the database, including "Data Common to All Companies" and "Application Objects." "Data Common to All Companies" includes the report list and permissions groups of the program. When you restore the "Application Objects," the accounting application is transferred to the database.

To restore the standard database:

1. Open the new database and click TOOLS→RESTORE. A standard Windows dialog box appears. In this Restore window, locate database.fbk, select it and click **Open**. The following window appears:



2. Make sure that Entire Backup is selected, and click **OK** to start restoring the database. The following window appears:



The restore procedure will take a few minutes, and this window allows you to monitor its progress. For more information, see "Microsoft Navision Database Administration Tools."

When the restore process has been completed, your database will be ready for use in Microsoft Navision. It will contain a copy of the demonstration company CRONUS International Ltd.

You can now create your own companies in the database. You can delete the demonstration company if you do not need it or if you want to use the space it takes up for storing other data.

Using the Standard Database

You can use the standard database in two ways: with a demonstration license (CRONUS.flf) or with your own license (fin.flf). The license file you want to use must first be uploaded to the server. For more information, see the section called "Uploading a License File."

Using the Demonstration License File Cronus.flf

If you choose to work under the demonstration license file, CRONUS.flf, you have access to all the Microsoft Navision application areas, and you can test all the functions – including ones you have not purchased permissions for. The demonstration license file does, however, contain certain restrictions.

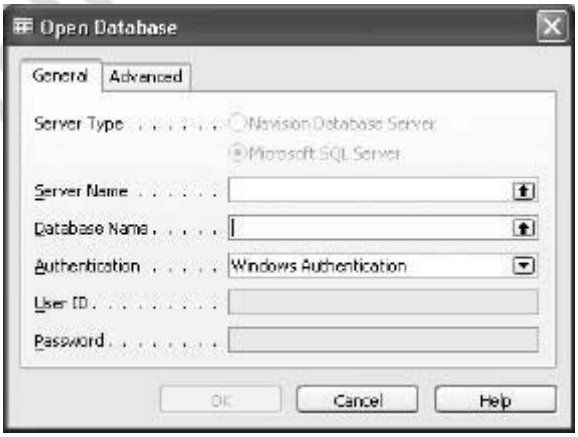
Using Your Own License File

If you work under your own license file (fin.flf), you can use only the functions for which you have purchased permissions. This means that you can only see the data for those functions – even in the demonstration company. On the other hand, your own license file does not limit posting dates. You can also create as many companies in the database as you have purchased permissions for.

Opening Databases

To open a database:

1. Click FILE→DATABASE→OPEN . The Open Database window appears:



2. In the **Server Name** field, enter the name of the server.

If the client is running on Windows XP or Windows 2000, you can see a list of servers in the active domain. You must remember the names of servers in other domains.

SQL Server 2000 allows you to run multiple server instances on the same computer. Each server instance that is available on a computer is included in the list that is shown in Available Microsoft SQL Servers window, along with the other servers that are available.

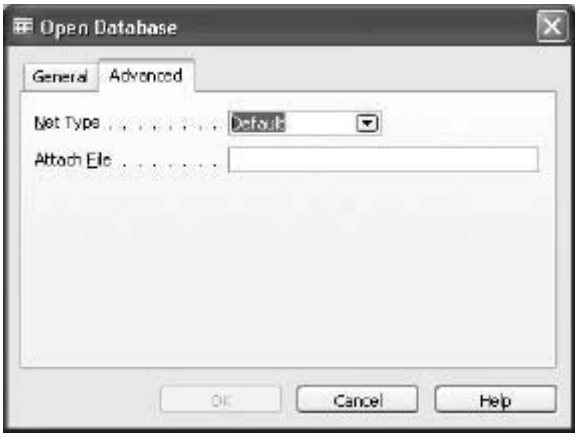
3. In the **Database Name** field, enter the name of the database that you want to open. You can browse a list of the Microsoft Navision databases that are available on the selected server by using the **AssistButton**. The Available Databases window will appear.

   This window lists all of the available databases and tells you who created them as well as when they were created. It also tells you the kind of replication that is used by the database if it is participating in SQL Server replication.

4. In the **Authentication** field, specify the type of authentication you require. You can choose between database server authentication and Windows authentication. You can use the **AssistButton** to select the authentication type from a list. For more information about authentication, see the section called "Security and User Setup."

5. You must enter a User ID and password if you are using database server authentication. If you are using Windows authentication, you do not have to enter a User ID and password.

6. You can specify the network type that will be used when you connect to the server by clicking the **Advanced** tab in the Open Database window. However, it is not usually necessary to change the network type from the default setting. The Default Net Type setting allows Microsoft Navision to connect to a server using the default client network type assigned by SQL Server. You can change the net type with the Client Network Utility, which is part of the SQL Server Client Utilities, if they have been installed on the client computer.



7. Use the **AssistButton** to select the net type from a list and click **OK**.
8. If you would like to attach and open a detached database that consists of a single data file (excluding the transaction log file), you must enter the complete path and file name of the database in the **Attach File** field. This file will typically have the .mbf extension. You can give this single file database any name by entering it in the **Database Name** field on the **General** tab. If no transaction log file exists alongside the data file, it will be created automatically when you attach the data file. The database file is attached when you open it.
9. Click **OK** to open the database.
10. When you have opened the database, you can open a company by clicking FILE→COMPANY→OPEN, or you can add a new company by clicking FILE→COMPANY→NEW.

NOTE: *Microsoft Navision will automatically open the database and company that you were last working on when you reopen the program.*

Automatic Reconnection

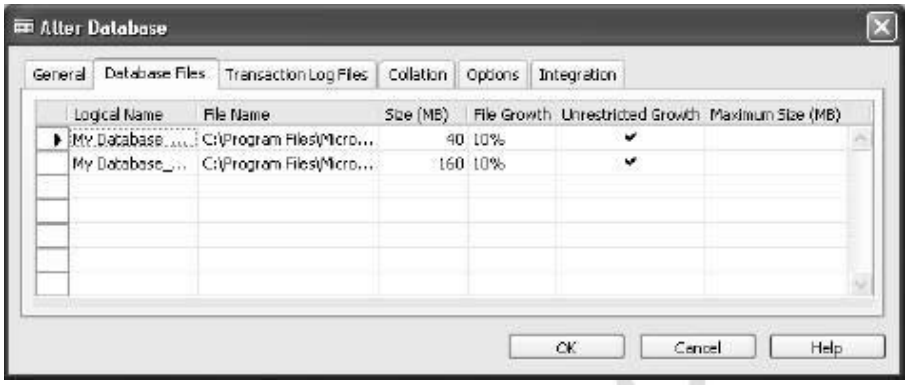
Microsoft Navision can automatically detect if the connection to the server has been broken. It will do this the next time the client tries to access the server after having been inactive for at least ten minutes. If the connection has been broken, for example, because the server has been stopped and then restarted, Microsoft Navision will attempt to reestablish the connection to the server and open the database with the same settings that were used when the database was last opened. The user can then continue to work with the database and will not notice that the server has been unavailable.

This allows you to shut down the server temporarily, for example, for hardware upgrades or modifications to server properties and then restart it, without causing any serious inconvenience.

Altering the Database

The SQL Server Option for Microsoft Navision allows you to change the database properties whenever you think it is necessary.

Click FILE→DATABASE→ALTER, and the Alter Database window appears:



The Alter Database window is similar to the New Database window and allows you to perform the following actions on the current database:

- Change the properties of the existing data or transaction log files.
- Delete existing data files or transaction log files, provided they are empty. The first data file or transaction log file that is listed is the primary file. The primary file cannot be deleted. In the SQL Server Option for Microsoft Navision, the primary data file is only used for storing SQL Server catalog information and system tables for the database, provided that secondary data files have been created.
- Add new data files or transaction log files. Any new data files that are added will be included with the secondary data file(s) in a separate filegroup from the PRIMARY filegroup. The secondary data files are used for storing Microsoft Navision data. New data files cannot be added to the PRIMARY filegroup from within Microsoft Navision.
- Change the collation that the database is using. For more information about changing the collation, see "Changing the Collation."
- Change the database options. All the properties listed on the **Options** tab can be changed.
- Create and maintain views and table relationships in SQL Server for integration purposes.
- Synchronize the table relationships that exist in Microsoft Navision with those that are maintained on SQL Server.
- Change the identifier/character mappings for SQL Server identifiers.
- Change the database license file setting. If you decide that the license file should be stored in the database instead of on the server, you will be prompted to upload the license file to the database. If you decide that you no longer want the license file to be stored in database, the database will use the license file that is stored on the server.

NOTE: *You must set the database Single user option before you alter the database.*

You cannot change the logical name or the file name of the data files or the transaction log files.

For more information about storing the license file in the database, see the section titled "License Files." For more information about table relationships, see the manual Application Designer's Guide.

Expanding the Database

You can expand the database by:

- Increasing the size of the existing data files or adding new data files in order to allow more data to be stored in the database. If secondary data files are present, you will only need to increase the size of the primary data file when the catalog it contains has become too large. When this occurs, new SQL Server objects such as tables cannot be created until you increase the size of the primary data file. When you are using secondary data files, you cannot create more space for storing Microsoft Navision data by simply increasing the size of the primary data file. You can create more space for storing data by increasing the size of the secondary data files, which contain Microsoft Navision information. You can also add new secondary data files in order to store more data.
- Increasing the size of the existing transaction log files or adding new transaction log files in order to allow more transactions to be performed in the database. The transaction log will continue to grow as new transactions are performed in the database. SQL Server truncates the log after performing a successful database or transaction log backup.

To expand your database, you must have enough space available on the server hard disk(s) where you store the files that you are expanding.

Remember that once space has been allocated to the database, it cannot be used by any other applications on the server computer until the database has been deleted.

NOTE*: You should always make a SQL Server backup of your database before expanding it. To learn more about making backups, see the "Microsoft Navision Database Administration Tools."*

Deleting Database Files

You can only delete data files or transaction log files from a SQL Server database if the files are empty.

You cannot delete the primary data file or the primary transaction log file.

Changing the Collation

You can also change the collation that is used by the database. This will change the collation of all the objects in the database except tables that have the LinkedObject property set to Yes. These objects must be recreated manually by, for example, scripting them in SQL Server's Enterprise Manager.

You must set the database Single user option before you change the collation.

If you change the collation from a case-sensitive to a case-insensitive collation or from an accent-sensitive to an accent-insensitive collation, duplicates can occur in the primary keys of the tables. Duplicates can be caused by the values of the character data stored in the primary keys. If duplicates occur, you will get an error message and the alteration of the database collation will be aborted. Therefore, we recommend that you do not change these attributes of a collation.

***NOTE**: Changing the collation can be a lengthy process, depending on the size of the database and the number of companies in the database because the system tables and all the user table indexes that contain character data must be rebuilt. Furthermore, rebuilding the database will increase the size of the transaction log. You must ensure that there is a considerable amount of free space before changing the collation. If there is not enough space for the transaction log file, you will receive an error message and the collation change will be rolled back.*

Changing Database Options

Microsoft Navision also allows you to change any of the database options that you set when you created the database. You should, for example, place a check mark in the **Single user** field on the **Options** tab before carrying out any database tests. This check box should be cleared when the tests are completed. For more information about the fields contained in the Alter Database window, see the section called "Creating a Database."

When you click **OK**, the information is saved, and you can continue to work in the database.

Database Configuration Guidelines

The first data file listed in the **Database Files** tab of the New Database and Alter Database windows is created as the primary data file, and, therefore, it is placed in the PRIMARY filegroup.

All other data files are secondary files and are placed in a separate filegroup that is set as the default filegroup for the database. All Microsoft Navision objects that are created in the database are distributed among the files in this filegroup if secondary data files have been created.

The primary data file cannot be deleted. You cannot add new data files to the PRIMARY filegroup from within Microsoft Navision. However, you can change the properties of the primary data file, such as the size, the growth, and so on in the same way as you can for the secondary data files.

The placement of data and transaction log files is important in determining the efficiency and integrity of a database. The minimum server configuration should consist of at least four disks. You should use one disk for storing the primary data file for each of your databases. You should the second disk for storing the operating system software and any installed applications. You should use the third disk for storing the secondary data files (which contain the Microsoft Navision data) if secondary data files have been created. You should use the fourth disk for storing the transaction log files.

Keeping the primary data file and the transaction log files on separate disks from the secondary data files enables you to use a more robust backup and restore strategy. If there is a media failure on the disk containing the secondary data files, it is still possible to make a backup of the active transaction log, restore the database from a previous database backup, and apply all the transaction log backups up until the point of failure.

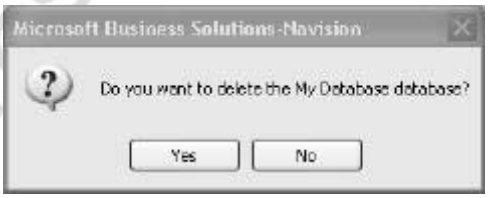
We recommend that you use as many disks as possible for the distribution of data and transaction log files. A RAID configuration allows such a distribution by providing striping to optimize performance or mirroring to support fault tolerance. Preferably, both striping and mirroring should be employed. For more information about backups and using RAID with SQL Server, consult Microsoft's SQL Server documentation, or your Microsoft Certified Business Solution Partner.

***NOTE**: Data files and transaction log files cannot be created on compressed file systems or remote network drives.*

Deleting Databases

Never delete a database without making a backup. Save one or more copies of the backup in a secure place. For more information about making backups, see the "Microsoft Navision Database Administration Tools."

After you have made the backup, you can click FILE→DATABASE→DELETE to remove the original database that will no longer be used. Before the database is deleted, you will have to answer two messages like this one:



When you delete a database, everything in the database is deleted, including any customizations you have made. You are not able to delete a database that you cannot open from within Microsoft Navision. You cannot delete a database if other connections are using the database.

Deleting Part of a Database

If you do not want to do anything quite so drastic as deleting the entire database, there are various other ways to delete part of the information:

- To remove old information, use the Date Compress batch jobs. On the main menu, click ADMINISTRATION→IT ADMINISTRATION→DATA DELETION→DATE COMPRESSION for the relevant application area.
- To remove individual records, open the relevant application and on the menu bar click EDIT→DELETE. There must be no open entries or non-zero balances for the records you want to delete.
- To remove a company, click FILE→COMPANY→DELETE.
- If you have access to the development environment for Microsoft Navision, you can delete individual objects. You can read about the development environment in the *Application Designer's Guide*.
- If you need to delete almost everything except a couple of objects, such as some reports, you can save the objects by exporting them before you delete the database. You can then click FILE→DATABASE→DELETE to delete the database. Finally, you can import the old objects to a new database.

Moving a Company from One Database to Another

If you happen to create your own company in the wrong database, you can move it to the correct database by making a backup copy of just the company. (Select the appropriate company before you begin the backup.) When the backup is complete, open the correct database and restore the company backup into it.

Renaming a Company

You can also rename a company after it has been created in Microsoft Navision.

Before you rename a company you must make sure that you are working in the company that you want to rename and that you are the only user who has access to the company.

To make sure that you are the only user with access to the company, click FILE→DATABASE→ALTER, and the Alter Database window appears. Click the **Options** tab, and enter a check mark in the **Single user option** field and click **OK**.

To rename the company, click FILE→COMPANY→RENAME, and the Rename Company window appears. Enter the new company name in the **New Company Name** field, and click **OK**.

Renaming a company that has been used and already contains data can take some time.

Security and User Setup

Security in the SQL Server Option for Microsoft Navision

An enterprise business solution must have a built-in security system that protects your database and the information that it contains from being accessed by unauthorized people. It must also allow you to specify what the authorized users are allowed to do in the database – whether they can read, enter, or modify data.

The minimum acceptable level of security requires that each user is assigned an ID and a password. This ensures that only authorized personnel can gain access to your database. This is database level security.

A medium level of security requires that you can limit the user's access so that they can only access certain types of information stored in the database. In other words, they can only gain access to particular tables in the database. This is table level security.

A high level of security requires that you can limit the access that users have to the information that is stored in the tables – that they can only gain access to specific records in the tables. This is record level security.

The SQL Server Option for Microsoft Navision satisfies these requirements by integrating its own security system (which includes record level security) with the Microsoft SQL Server security system and with the Windows security system. This allows Microsoft Navision to use the unified login system provided by Windows. If your domain is running on Windows 2000 or Windows 2003, Microsoft Navision makes use of both the Active Directory Services and the single sign-on system. For more information on Active Directory, see "Microsoft Navision Security."

Security Overview

In order to understand how security is managed in the SQL Server Option for Microsoft Navision, you must understand the SQL Server security system, Active Directory, the Microsoft Navision security system and how they interact.

This section explains how the various parts of the security system work together. With several security systems interacting, the terminology can be confusing, so before explaining how the Microsoft Navision security system works it is necessary to clarify two key concepts:

- **Authentication:** The process by which the system validates the users identity. This can be done by having the user enter an ID and password when they log on.
  - Microsoft Navision supports two kind of authentication: Windows authentication and database server authentication.
- **Login:** When a user has identified themselves and been recognized by the system they are granted access to the parts of the system for which they have permission.

If the user has used Windows authentication to log on to the system, then they have been assigned a Windows login.

If the user has used database server authentication to log on to the system, then they have been assigned a database login.

The following table shows what the different authentication modes require from the user before granting access to databases.

Authentication	Windows requires	SQL Server requires
Windows Authentication	Windows account (user ID and Windows login password)	Windows login
Database Server Authentication		SQL Server login (user ID and password)

Database Server Authentication

In the SQL Server Option for Microsoft Navision the database server authentication is based on Microsoft's SQL Server authentication. For more information, see the section called "SQL Server Authentication."

Windows Authentication

The Windows single sign-on and the unified login supported by Windows are the same. In this manual, we will refer to both of these systems as Windows authentication.

Authentication and Login

With Windows authentication, when a user tries to connect with SQL Server to open a database, he will not have to supply a user ID or password. Microsoft Navision will automatically ask Windows to confirm whether or not this user, who has already logged on to the network, has a valid Windows account and whether this account gives him the right to access this particular server.

If the user is allowed to access the server, then Microsoft Navision will check to see if the user has been assigned a Windows login within Microsoft Navision. If the user has a Windows login, he will be granted access to the database.

The user will be granted access to Microsoft Navision and be given the permissions specified for that Windows user and those specified for any Windows groups of which he is a member.

If the user does not have a valid Windows account or if his account does not include permission to log on to the Microsoft Navision database, authentication fails and the user receives an error.

Advantages of Windows Security

The Windows authentication system includes the following security features:

- Secure validation and encryption of passwords
- A time limit on passwords
- Minimum password length
- Account lockout after an invalid password is entered

The SQL Server Security System

Microsoft SQL Server has two levels of security: server security and database security. The SQL Server Option for Microsoft Navision has embraced both levels of security and interacts with them by means of an automatic synchronization process.

Server security consists of server-wide security accounts (known as logins), which are used to authenticate users before granting them access to the server.

Database security consists of database-specific security accounts that control the level of access and the permissions granted to individual users for the databases on the server.

Security Server

The SQL Server security system authenticates users by validating their logins before granting them access to any of the resources contained in the system.

SQL Server employs two types of authentication. These correspond to the two types of logins that can be created in SQL Server: Windows logins and SQL Server logins.

Windows Authentication

The Windows authentication used by SQL Server corresponds to the Windows authentication used by Microsoft Navision.

SQL Server Authentication

The database server authentication used by Microsoft Navision refers to SQL Server authentication, as mentioned in the section called Security Overview. It is used when the network administrator has decided not to support Windows authentication, or the SQL Server administrator has chosen not to use Windows authentication.

With this method, SQL Server carries out its own authentication of the user's ID and password. SQL Server does this by checking whether a SQL Server login with this user's ID and password has been created. This login must first have been created by a SQL Server administrator, with a SQL Server tool. If an SQL Server login has not been set up, authentication fails and the user receives an error.

Database Security

In the SQL Server security system, access to individual databases on the server is controlled by the database user accounts in each database.

The user is granted access to the server after the login has been authenticated. Database security then validates permissions by checking the database user accounts on the server. The permissions that the user has been granted to the various objects within the database, such as tables, are determined by the information contained in the user's database user account. It also contains information about any additional permissions that the user may have been granted to alter the database itself.

Users who have valid SQL Server logins, but no database user accounts, will be granted default permissions. The default setting grants such users access to the master database as guests. Guests have very limited rights. This means that a valid SQL Server login always gives access to at least one database.

Microsoft Navision and the SQL Server Security System

The two previous sections of this chapter have been devoted to explaining the SQL Server security system and the Microsoft Navision security system. This section explains how these two systems interact.

Note that Microsoft Navision now has two login tables. Windows logins are listed in the Windows Login table. Database logins are listed in the User table.

The heart of the security system for the SQL Server Option for Microsoft Navision is the synchronization process. The synchronization process ensures that the information contained in the Microsoft Navision User table and Windows Login table corresponds with the information contained in the SQL Server security system.

SQL Server database user accounts contain information about the permissions that the users have to the objects contained in the database. The information for managing permissions to Microsoft Navision objects is contained and administered within Microsoft Navision.

Synchronization of User Accounts

Every time a user is added, deleted, or renamed in the Windows Login table or User table, a synchronization process is initiated. The synchronization process compares the Microsoft Navision login tables with the security system in SQL Server. It modifies the SQL Server security system to reflect the changes made in the Microsoft Navision Windows Login table or User table.

This means that every time a Microsoft Navision database administrator alters the information about a login in one of the Microsoft Navision login tables for a particular database, the synchronization process automatically updates the information contained in the SQL Server database user accounts for this database.

Microsoft Navision cannot create or delete a SQL Server login. The SQL Server login must first be created by a SQL Server administrator. Microsoft Navision can only verify or reject the validity of a login before updating the database user account. However, Microsoft Navision can create and delete Windows logins in SQL Server through the synchronization process.

Adding Users

For both kinds of logins, the synchronization process creates a database user account for the login in the corresponding database if such an account does not already exist.

Windows Logins

If a new Windows login is added to the Windows Login table of a Microsoft Navision database, the synchronization process matches this login to that in SQL Server. This is done by comparing the security identifiers (SIDs) of the two logins. If the synchronization process does not find a match, the system creates a new Windows login in SQL Server.

Database Logins

If a new database login is added to the User table of a Microsoft Navision database, the synchronization process checks whether this user ID has a valid SQL Server login in SQL Server. This SQL Server login must have the same name (user ID) as the Microsoft Navision login that is being added.

Deleting Users

For both kinds of logins, if you delete a login from one of the Microsoft Navision login tables, the synchronization process deletes the SQL Server database user account for that login. Note that if you delete SQL Server database user accounts from outside Microsoft Navision without deleting the login in Microsoft Navision, synchronization will create new database user accounts for these users.

Windows Logins

When a Windows login is deleted from the Microsoft Navision Windows Login table, you are asked if you want to delete this user's Windows login on SQL Server. Microsoft Navision does not delete the Windows login on SQL Server automatically.

Database Logins

When a database login is deleted from the Microsoft Navision User table, the synchronization process will not delete the SQL Server login. It can only be deleted by a SQL Server administrator using a SQL Server tool, such as Microsoft Enterprise Manager.

When Microsoft Navision tries to match user IDs in the User table with SQL Server logins, the uppercase user ID in the User table is matched with the uppercase representation of the logins in SQL Server, regardless of case.

Synchronization

The synchronization process can be initiated from within Microsoft Navision. To start the synchronization process, click TOOLS→SECURITY→SYNCHRONIZE.

You may need to initiate the synchronization process after having restored a Microsoft Navision backup. If the logins in the backup do not match the SQL Server logins or the Windows users and groups, the necessary changes must be made to the Microsoft Navision logins, Windows users and groups, or SQL Server logins after the backup has been fully restored. You should re-initiate the synchronization process after these changes have been made.

WARNING: *Never use SQL Server tools to add or delete information stored in the Microsoft Navision Windows Login table or User table because this information is used during the synchronization process.*

SQL Server Database Roles and Server Roles

The ability to perform certain activities within SQL Server requires that the users have the appropriate server or database permissions or that the members of certain server or database roles. Membership of these roles is not assigned automatically during the synchronization process. They must be assigned by a SQL Server administrator.

The minimum requirements for carrying out these activities are listed in the following table:

Microsoft Navision Activity	Requires Membership of Server or Database Roles
Invoking the synchronization process or modifying the User table	sysadmin server role. Alternatively both a member of the securityadmin server role and a member of the db_owner database role for this database.
Creating a database	sysadmin or dbcreator server role. Alternatively, the user must have been granted the create database permission. The user must also have public access to the model database.
Altering a database	sysadmin or dbcreator server role. Alternatively a member of the db_owner or db_ddladmin database role for this database.
Creating tables within a database	sysadmin server role or be a member of the db_owner database role for this database.

In order for a user to create or modify table definitions in Microsoft Navision, they must be a member of the db_owner database role (the database creator is automatically a member of this role). Membership of this database role must be assigned outside Microsoft Navision, after the user has been created in one of the Microsoft Navision login tables. These are the Windows Login table (accessed through the Windows Logins window) and the User table (accessed through the Database Logins window). Adding a user to a database role is done with a tool such as Microsoft Enterprise Manager. Note that this user will then have all permissions within this database.

Any permissions required on individual tables or views in the database for users who are not db_owner users must be manually granted to the users if the use of external tools such as report writers is required outside Microsoft Navision.

Direct Access to Microsoft Navision Tables on SQL Server

Because of the security risk involved in using an open database such as SQL Server, all permissions to access Microsoft Navision tables in SQL Server directly, using various SQL Server tools, must be granted permissions by a SQL Server administrator from outside Microsoft Navision.

Access to SQL Server Objects

A login created in Microsoft Navision will have a corresponding login and database user account in SQL Server. This database user account does not grant the user any permissions on any SQL Server object stored in the database, regardless of whether the user has been granted permissions in Microsoft Navision or not. Therefore, if the user logs on to the server from outside Microsoft Navision, with a tool such as Microsoft Enterprise Manager, she will not be granted access to any SQL Server object stored in the Microsoft Navision part of the database.

Test Your Knowledge – Microsoft Navision SQL Server Option

Answer the following questions about the Microsoft Navision SQL Server Option:

1. You must add an _____ _____ _____ to each instance of SQL Server that you want to access using Windows authentication. However, if you select the _____ _____ option of Microsoft Navision and are going to access the default SQL Server instance on your local computer with Windows authentication, the installation program will add the above for you.
2. In order to work with the SQL Server Option for Microsoft Navision, you must _____ the license file you want to use to all the SQL Servers you wish to access, instead of copying it to the individual clients.
3. The six tabs on the New Database window are: _____, _____, _____, _____, _____ and _____.
4. The placement of data and transaction log files is important in determining the _____ and _____ of a database.
5. The _____ _____ specifies the rules by which character data is stored and compared.
6. Microsoft SQL Server has two levels of security: _____ and _____.
7. The _____ _____ function allows you to change the database properties whenever you think it is necessary.
8. SQL Server employs two types of authentication. These correspond to the two types of logins that can be created in SQL Server: _____ and _____.
9. The _____ _____ compares the Microsoft Navision login tables with the security system in SQL Server.
10. To create or alter a database, the administrator must be a member of either the _____ or _____ SQL server roles.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 6: MICROSOFT NAVISION APPLICATION SERVER

Overview

This chapter describes how to install and maintain Microsoft® Business Solutions–Navision® Application Server, which handles the exchange of documents between Microsoft Navision and external services, such as a CRM or eCommerce product.

This chapter also introduces the Microsoft Navision Automated Data Capture System (ADCS). ADCS is aimed at companies that need to use handheld devices in their warehouse processes. This is covered in this chapter because the ADCS solution is designed around the Microsoft Navision Application Server.

Getting Started with Microsoft Navision Application Server

Microsoft Navision Application Server

Microsoft Navision Application Server is a middle-tier server, which executes business logic without user intervention. With Microsoft Navision Application Server, it is possible to communicate with external services.

Microsoft Navision Application Server acts as a client towards a database server and can act as a server for other services.

When you start Microsoft Navision Application Server, it opens the pre-existing install database and executes C/AL™ code in a predefined codeunit.

To understand more about how Microsoft Navision Application Server communicates with external services, see the Development Guide for Communication Components (online Help).

No User Interface

Microsoft Navision Application Server runs without displaying anything on-screen and therefore requires no user interaction. This means, of course, that the user will not get any error messages. Instead, these are logged in the Event Viewer. However, the user can interact with Microsoft Navision Application Server through the Microsoft Navision Application Server Manager.

For more information on message logging, see the section titled “Message Logging.”

For more information on Microsoft Navision Application Server Manager, see the section titled “Microsoft Navision Application Server Manager.”

Installing Microsoft Navision Application Server

We provide a Windows Installer package. You use the setup program to install, maintain, and remove a Microsoft Navision Application Server. Windows Installer registers Microsoft Navision Application Server as a service as well as an event source. When the installation has finished, you have the option to start up Microsoft Navision Application Server through the Microsoft Navision Application Server Manager.

Canceling the Installation
You can cancel the installation at any time. Windows Installer will then restore your computer to the state it was in before the installation process began.

Preconditions and System Requirements

License Files
Before you start the installation process, you need to be aware of certain preconditions and system requirements. Once these are met, the installation can start.

A precondition for running a Microsoft Navision Application Server is that you have a valid license file installed on your computer, which contains the Microsoft Navision Application Server granule. For information about license files, see "Microsoft Navision Installation."

Make sure that Microsoft® Windows® XP or Windows® 2000 is installed on the computer on which you will be running Microsoft Navision Application Server. If this is not the case, you must install the correct operating system before installing Microsoft Navision Application Server. The computer must have at least 64 MB RAM.

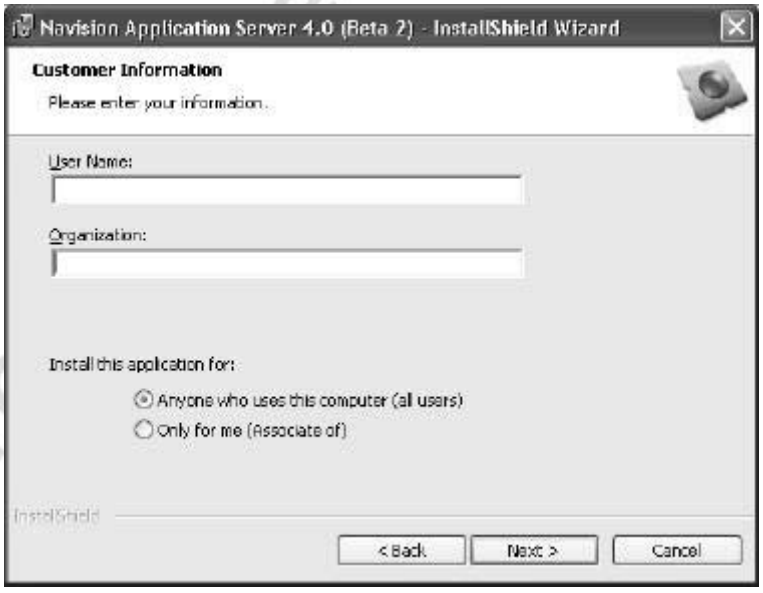
Starting the Installation

When you start the setup program, Windows Installer checks whether a Microsoft Navision Application Server is already installed. If it finds an existing installation, the Program Maintenance window appears.

If a Microsoft Navision Application Server is not already installed, the setup program will continue, and the installation wizard starts. The following window appears:

Customer Information

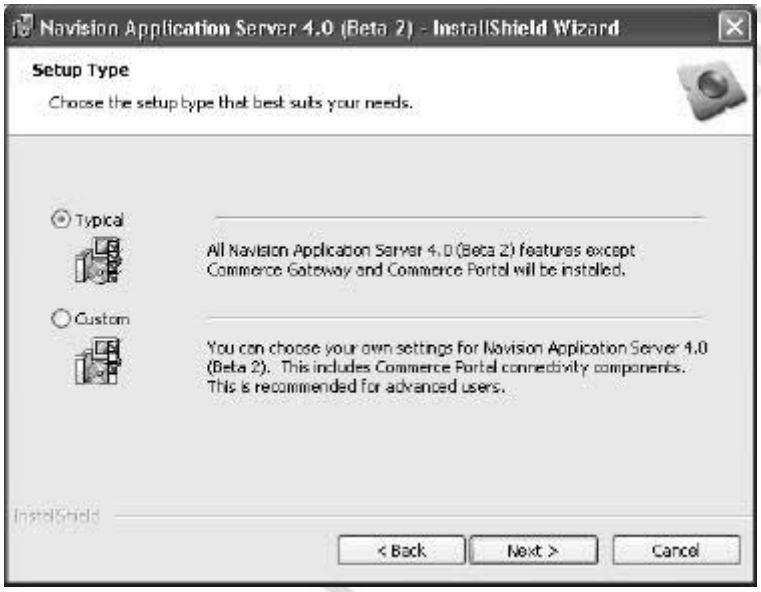
To continue the installation, click **Next**. The Customer Information window appears:



Your name and the name of your organization appears automatically in this window. You can also specify to whom this installation belongs. You can select the person who installed it or any user who logs onto this computer. This determines who is allowed to modify or uninstall it. It does not determine who is able to use the program from this computer.

Selecting the Installation Type

Click **Next**. The Setup Type window appears:

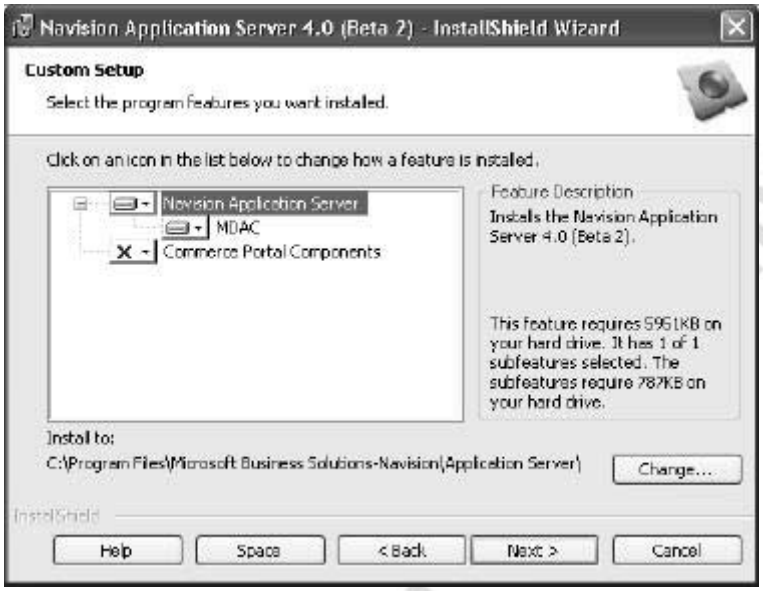


In the Setup Type window, you can select a typical installation or a custom installation.

- Select Typical to install all Microsoft Navision Application Server settings.
- Select Custom to choose your own settings. Furthermore, you can change the default destination for the Microsoft Navision Application Server program files. The custom setup type also enables you to select Commerce Portal connectivity components. For information about these components, see the manual *Installation & System Management: E-Commerce Solutions*.

Customized Installation

If you select *Custom*, the Custom Setup window appears:

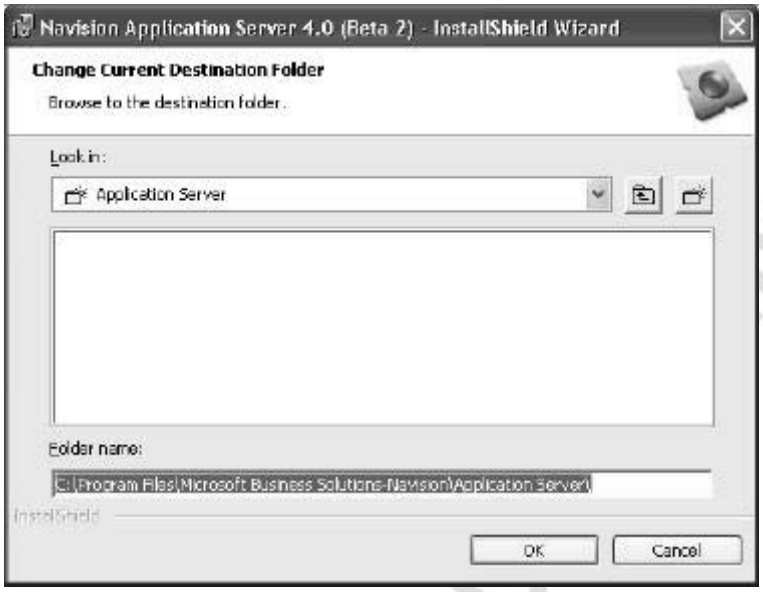


This is where you select the features that you want to install. The window is divided into three sections:

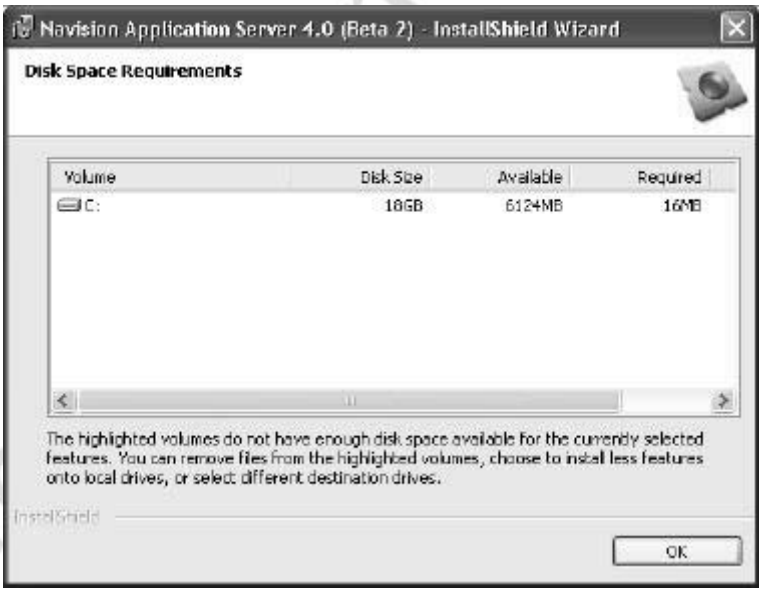
- A feature selection area.
- A feature description area that displays a short description of a feature when you select it. It also displays an estimate of how much disk space the feature requires.
- An installation destination folder that tells you where the feature will be installed. If you want to install, for example, Microsoft Navision Application Server in a different folder, click **Change**.

NOTE: *Microsoft Data Access Components (MDAC) are installed by default. These operating system components facilitate access to data in a Microsoft Navision database with third-party tools. The components are primarily used with the Microsoft Business Solutions–Microsoft Navision SQL Server Option.*

If you click **Change**, the Change Current Destination Folder window appears:

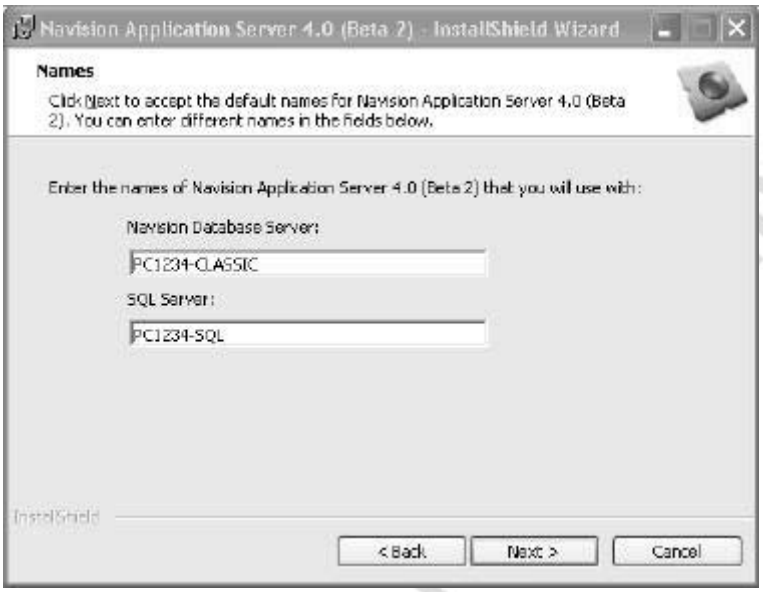


If you click **Space** in the Custom Setup window, the following window appears, which informs you of how much space is available on the various drives to which you have access:



Naming the Microsoft Navision Application Server

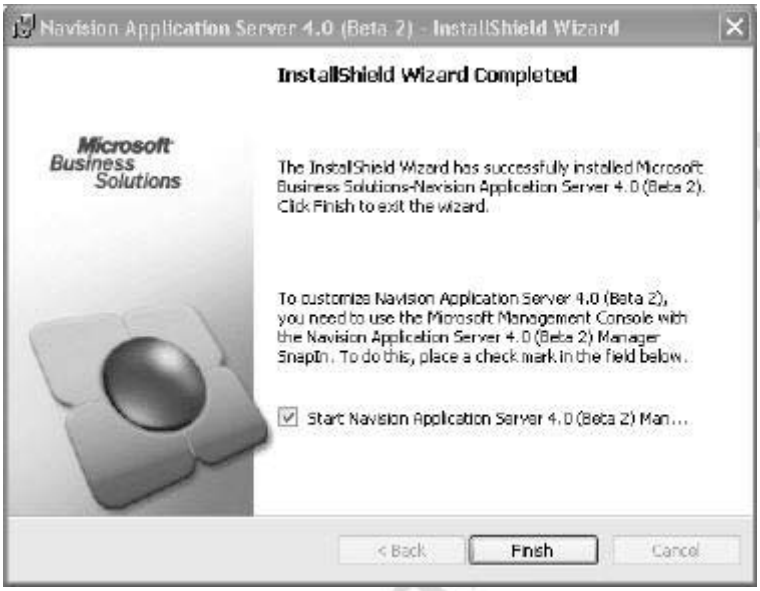
Click **Next** in the Custom Setup window. The Names window now appears:



You can either accept the default names shown for Microsoft Navision Application Server – for use with Microsoft Navision Database Server and SQL Server – or you can enter different names. By default, the installation program will select the name that your computer has in the network. If the name you select already exists, a window appears asking you to rename the server.

If you selected Typical in the Setup Type window, the installation uses your computer name as the server name. If this name already exists, it will add #1 at the end of the computer name. If this name also exists, it will try with #2, and so on. For example, if the computer name is MyComputer515 and this server name already exists, the installation will try MyComputer515#1.

When you click **Next**, the installation is ready to begin. Click **Install** in the Ready to Install the Program window. When the installation is finished, the following window appears to inform you that the installation has been completed successfully:



Installation Complete

In this window, you can choose to finish the installation or finish the installation and continue with the customization of Microsoft Navision Application Server. To customize Microsoft Navision Application Server, place a check mark in the **Start Microsoft Navision Application Server 4.00 Manager** field and click **Finish**.

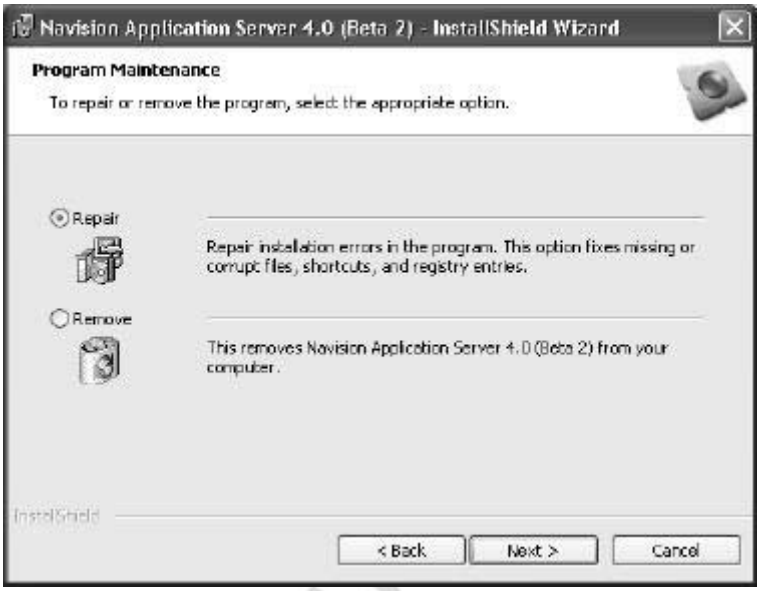
Maintaining Microsoft Navision Application Server

***ATTENTION**: We recommend that you make a backup of any license files, databases, and database backups that are stored locally before repairing or removing the server installation.*

You can use Windows Installer to repair and remove a Microsoft Navision Application Server. To do so, follow this procedure:

1. Open the Control Panel and double-click the **Add/Remove Programs** icon.
2. Select Microsoft Navision Application Server.
3. Click **Change**. The Installation Wizard opens.

4. Click **Next**. The Program Maintenance window appears:



In this window, you can choose to repair or remove Microsoft Navision Application Server.

Repairing the Installation

If you select **Repair**, the Ready to Repair the Program window opens. Click **Install** to fix any installation errors such as missing or corrupt files. Windows Installer will protect any license files and database backups. Therefore, it will not overwrite these files if you have to repair the installation. When Microsoft Navision Application Server has been repaired, the Installation Complete window appears.

Removing the Installation

If you select **Remove**, a Ready to Remove the Program window appears. Click **Remove** to uninstall the product.

You can cancel the removal process at any time. If you do so, Windows Installer returns the computer to the state it was in before you started the removal process.

When you remove Microsoft Navision Application Server, everything will be removed except any license files (.flf except for cronus.flf), database files (.fdb) and database backup files (.fbk) that are stored locally. If you want to remove these files, you will have to do so manually.

When Microsoft Navision Application Server has been removed, a window appears to inform you that it has been removed successfully.

Losing the Server Connection

If Microsoft Navision Application Server for some reason loses its connection to Microsoft Navision Database Server, either accidentally (for instance, due to network problems) or due to maintenance (for example, the shutting down of a database server), the lost connection will not cause Microsoft Navision Application Server to stop. Instead, Microsoft Navision Application Server will attempt to connect to the server at regular intervals.

Running More Than One Microsoft Navision Application Server

You can run more than one Microsoft Navision Application Server with the same database or with different databases. However, Windows Installer can only install one Microsoft Navision Application Server on each computer. This is because Windows Installer checks for other Microsoft Navision Application Server installations when installing. In other words, only one installation can be completed with the Windows Installer.

To install more than one Microsoft Navision Application Server, you must copy all files that belong to the previously installed Microsoft Navision Application Server to another target folder. The files that you have to copy depends on whether you are using Microsoft Navision Database Server or Microsoft® SQL Server®.

- When running Microsoft Navision Application Server with Microsoft Navision Database Server, you must copy the following files: nas.exe, cronus.flf, fin.etx, fin.stx, dbm.dll and slave.exe.
- When running Microsoft Navision Application Server with SQL Server, you must copy the following files: nassql.exe, cronus.flf, fin.etx and fin.stx.

After copying the necessary files, you must use the Microsoft DOS command line to install Microsoft Navision Application Server as a service and set the necessary parameters.

Microsoft Navision Application Server Authentication

The user cannot use database authentication when logging on to Microsoft Navision Application Server. The user will always be logged on to Microsoft Navision Application Server as the current Windows user, using Windows authentication. When Microsoft Navision Application Server is running as a service, the user's account can be specified in the Services window, which you access by clicking CONTROL PANEL→ADMINISTRATIVE TOOLS→SERVICES.

To be allowed to gain access to or modify a database, the user has to be added to the Windows Logins window in the Microsoft Navision client. To do this, click TOOLS→SECURITY→WNDOWS LOGINS.

For more information about security issues, see the "Microsoft Navision Security."

Microsoft Navision Application Server Setup

This section describes how to set up and manage Microsoft Navision Application Server. There is also a description of how to log messages.

Microsoft Navision Application Server Setup Properties

When starting up Microsoft Navision Application Server, you must provide start up parameters to control how you want Microsoft Navision Application Server to work. The following table lists the properties that you can set at start up:

Property	Described on
Application Server Name	Page 184
Server Name	Page 185
Database	Page 185
Company	Page 186
Startup Parameter	Page 186
NAS Debug	Page 187
Breakpoints	Page 187
Object Cache	Page 188
Net Type	Page 188
Install as Service	Page 189
Uninstall as Service	Page 189

Application Server Name – Name of Microsoft Navision Application Server

Program Property	Purpose	Where Specified	Default Value	Value
appservername	Specifies which Microsoft Navision Application Server is starting up.	Command line: appservername= <appservername>	AppServer	Microsoft Navision Application Server name

You use the appservername program property to specify the name of a Microsoft Navision Application Server. You only have to set the properties for a Microsoft Navision Application Server once. The program remembers the information the next time you run it.

Example

Run a Microsoft Navision Application Server from the command line:

```
>nas appservername=nas1, servername=srv1, company="Cronus
International Ltd.", startupparameter=p
```

Microsoft Navision Application Server with the name nas1 starts running. When stopped and restarted, you only need to type in the following:

```
>nas appservername=nas1
```

Server Name – Choosing the Database Server

Program Property	Purpose	Where Specified	Default Value	Value
servername	Specifies which server to connect to.	Command line: servername= <servername>	None	Database server name

You use this program property to specify the database server that Microsoft Navision Application Server will connect to. A Microsoft Navision Application Server that uses Microsoft Navision Database Server works without a specified database server name (meaning, it can run directly on the database). However, if a Microsoft Navision Application Server uses SQL Server, you must specify the server name.

You can set up the connection to the database server on the command line by entering the name of the server after servername=.

Database – Selecting a Database File

Program Property	Purpose	Where Specified	Default Value	Value
servername	Specifies which database to open. A Microsoft Navision Database Server can only have one database open at a time.	Command line: database= <databasename> In a client/user installation, this only works in combination with servername=.	None	Name of database (including path if database is not located in Microsoft Business Solutions-Microsoft Navision folder.)

You use the database program property to make Microsoft Navision Application Server start up with a particular database open.

If Microsoft Navision Application Server is running with Microsoft Navision Database Server and the database server name is specified, then this property is not used, or it will be overruled by the server name property. If the database server name is unspecified, then you must specify the database name in order for Microsoft Navision Application Server to start up. Microsoft Navision Database Server can only work with one database at a time. If you specify another database, then the program generates an error message.

When running Microsoft Navision Application Server with SQL Server, you must specify both properties.

On the command line that starts Microsoft Navision Application Server, type the name of the database immediately after database=.

Company – Selecting a Company

Program Property	Purpose	Where Specified	Default Value	Value
Company	Specifies which company to open.	Command line: company= <companyname>	CRONUS International Ltd.	Company name

You use this program property to specify which company will automatically open when Microsoft Navision Application Server starts running. You must specify this property to enable Microsoft Navision Application Server to start up.

Startup Parameter – Starting the Application Server

Program Property	Purpose	Where Specified	Default Value	Value
startupparameter	Starts up Microsoft Navision Application Server.	Command line: startupparameter = <parameter>	Has no value	Text selected by the user.

This parameter is passed to the codeunit trigger, which is always executed when Microsoft Navision Application Server starts up (codeunit 1, trigger 99). The parameter is in the form of a string that has a maximum length of 1024 characters. This trigger is not defined in the standard application. Interpretation of the string is the responsibility of the C/AL programmer. If you do not supply the parameter, the Microsoft Navision Application Server startup procedure fails.

NAS Debug – Activating Microsoft Navision Debugger

Program Property	Purpose	Where Specified	Default Value	Value
nas debug	Activates the Microsoft Navision Debugger.	Command line: nas debug	Has no value.	Has no value.

To activate the debugger from Microsoft Navision Application Server, you include the debug parameter at start-up:

Example

```
nas debug,startupparameter="test",servername=PC0123
```

If you deactivate the debugger, you cannot activate it again unless you terminate Microsoft Navision Application Server and then start it up with the debug parameter.

***NOTE**: To be able to activate the debugger, you must ensure that there is a developer license file in the Microsoft Navision Application Server installation folder.*

Breakpoints – Using Another Breakpoint File

Program Property	Purpose	Where Specified	Default Value	Value
Breakpoints	Breakpoints can be saved to a file or loaded from a file.	Command line: breakpoints= C:\filename.xml	Has no value.	Has no value.

You can start Microsoft Navision Application Server with a breakpoints parameter to enable you to specify a particular file for saving and loading breakpoints.

Example

```
NAS.EXE breakpoints=C:\MyBreakpoints.xml
```

Object Cache – Improving Response Time

Program Property	Purpose	Where Specified	Default Value	Value
objectcache (KB) (clients only)	Makes the program run faster.	Command line: objectcache=<cachesize>	8000 (KB)	More than 0 KB and less than 1,000,000 KB

The Object Cache property increases the speed of Microsoft Navision Application Server. Objects, such as codeunits and tables that are used on the Microsoft Navision Application Server computer, are stored in the object cache. This means that the Microsoft Navision Application Server computer only needs to retrieve these objects once from the server, and then they will be stored in the object cache. Therefore, the Microsoft Navision Application Server computer must have enough memory to store the objects while Microsoft Navision Application Server is using them.

NetType – Selecting a Net Type

Program Property	Purpose	Where Specified	Default Value	Value
nettype	Permits choice of network protocol.	Command line: nettype=<nettype>	TCP	Netb, TCP

To use Microsoft Navision Application Server in a network, you must select the network protocol that is used for communication among Microsoft Navision Database Server, Microsoft Navision Application Server, and the clients. There are two possible values, nettype=tcp (for TCP/IP) or nettype=netb (for NetBIOS).

You must enter the same value on all the client computers and Microsoft Navision Application Servers in the network as well as on Microsoft Navision Database Server. On Microsoft Navision Application Server, enter the net type you have selected on the command line after the start command. When running Microsoft Navision Application Server with SQL Server, you must use the network protocol TCP.

Install as Service

Program Property	Purpose	Where Specified	Default Value	Value
installasservice	Installs the server as a service on your PC.	Command line: installasservice	Has no value.	Has no value.

You can install Microsoft Navision Application Server as a service on your computer. When Microsoft Navision Application Server is installed as a service, it starts every time you start Windows – without the user having to log on to the system. To install Microsoft Navision Application Server as a service, enter the following after the command prompt:

```
nas installasservice
```

Do not use the equal sign with this property.

Microsoft Navision Application Server will not start up automatically after the installation. You must start it using the Microsoft Navision Application Server Manager.

Uninstall as Service

Program Property	Purpose	Where Specified	Default Value	Value
uninstallasservice	Installs the server as a service on your PC.	Command line: uninstallasservice	Has no value.	Has no value.

If you want to uninstall Microsoft Navision Application Server as a service, enter the following after the command prompt:

```
nas uninstallasservice
```

Do not use the equal sign with this property.

Setting Microsoft Navision Application Server Properties

As explained in the previous section, you can customize the system setup by changing the settings of the various Microsoft Navision Application Server properties.

When you run Microsoft Navision Application Server from the command line, you can specify the program properties in any order on the command line. Enter the properties after Microsoft Navision Application Server's start command, separated by commas. The name of each property is followed by an equal sign (=) and the value to which the property is to be set, for example:

```
database=database.fdb,company=cronus
```

This does not apply, however, to the two server properties:

`installasservice` and `uninstallasservice`.

For these properties, enter the parameter on the command line as follows:

```
installasservice
```

When you run Microsoft Navision Application Server as a service, you can set properties in the Microsoft Navision Application Server Manager.

Startup

If you installed Microsoft Navision Application Server as a service and you set the Startup Type option to Automatic, Microsoft Navision Application Server will automatically start every time you start the computer.

Managing Microsoft Navision Application Server

When you set up Microsoft Navision Application Server to run as a service, you can use the Microsoft Management Console to change its properties.

NOTE: *If you run Microsoft Navision Application Server from a command line, you must reconfigure it from the command line.*

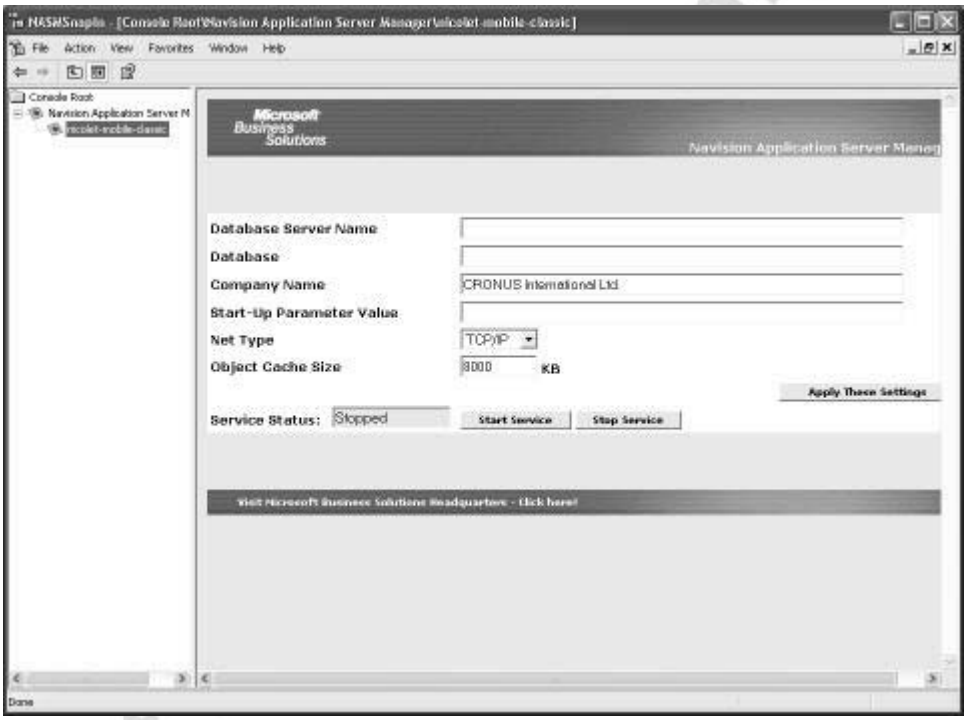
Microsoft Navision Application Server Manager

The Microsoft Navision Application Server Manager is a Microsoft Management Console snapin that does the actual managing of Microsoft Navision Application Server. With the Microsoft Navision Application Server Manager, you can see which Microsoft Navision Application Server properties you have set on the command line. You can also add and reconfigure Microsoft Navision Application Server.

The Microsoft Navision Application Server Manager interacts with Microsoft Navision Application Server and requires that the latter is installed. The Microsoft Navision Application Server Manager accesses Microsoft Navision Application Server properties through the Registry. If any changes are made to the properties, the Microsoft Navision Application Server Manager notifies Microsoft Navision Application Server so that it can respond accordingly.

You can change the properties at runtime, and they will be effective immediately. If Microsoft Navision Application Server is not running, the properties will take effect the next time you start up Microsoft Navision Application Server.

You can start the Microsoft Navision Application Server Manager immediately after the installation of Microsoft Navision Application Server is complete. Otherwise, to view the Server Manager window, click START→PROGRAMS→MICROSOFT BUSINESS SOLUTIONS-MICROSOFT NAVISION→MICROSOFT BUSINESS SOLUTIONS-MICROSOFT NAVISION APPLICATION SERVER MANAGER. The NASMSnapIn window appears.



The Microsoft Navision Application Server Manager window consists of two panes - the left-hand and right-hand panes. The left-hand pane displays the actual contents of the Manager, which means it shows an overview of the Microsoft Navision Application Servers that have been added to the view. On the right-hand pane of the window, the properties are displayed.

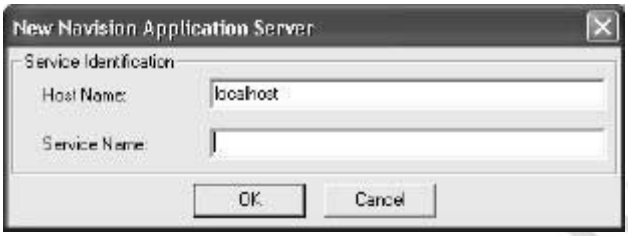
Adding Microsoft Navision Application Servers

You can add as many Microsoft Navision Application Servers to the Microsoft Navision Application Server Manager as you want. They will be listed under the Microsoft Navision Application Server node as shown in the previous picture.

To add a Microsoft Navision Application Server, follow this procedure:

1. Right-click the Microsoft Navision Application Server node.

2. Click NEW→APPLICATION SERVER, and the following window appears:



3. Enter the name of the Microsoft Navision Application Server that you want to add and the name of its host machine. The default value that is entered in the **Host Name** field is localhost.

Properties

In the right-hand pane of the Microsoft Navision Application Server Manager window, you can define or change the properties of Microsoft Navision Application Server. To see the configuration settings for a specific Microsoft Navision Application Server, you click the Microsoft Navision Application Server that you want to see the properties for on the left-hand pane of the Microsoft Navision Application Server Manager. On the right-hand pane of the window, the properties are displayed.

Field	Purpose
Database Server Name	If you are connecting to a database on the server, enter the database server name in the Database **Server Name** field.
Database	If you are connecting to a database not on a server, enter the name of the database (and the path if it is not located in the Microsoft Navision folder). Note that if Microsoft Navision Application Server is running with Microsoft Navision Database Server and the server name is specified in the Database **Server Name** field, this field cannot be populated.
Company Name	Here you specify which company will automatically open when Microsoft Navision Application Server starts running.
Start-Up Parameter Value	This parameter is passed to the codeunit trigger, which is always executed when Microsoft Navision Application Server starts up. For instance, entering MAILLOG in the **Start-Up Parameter Value** field prompts the Microsoft Navision Email Logging process to run from the codeunit trigger.

Field	Purpose
NetType	Enter the same value in this field as was entered in the Microsoft Navision Database Server. You can choose either TCP/IP or NetBIOS. When running Microsoft Navision Application Server with SQL Server, you must use the network protocol TCP.
Object Cache Size	This property increases the speed of the Microsoft Navision Application Server. The default setting is 8000KB, but this can be changed.

You can also see the status of Microsoft Navision Application Server, whether it is running or not. The **Service Status** field can have one of the following values: Stopped, Starting, or Started. You have the option to start or stop the Microsoft Navision Application Server as a service.

When you have made your changes to the configuration (by editing the values) and confirmed the changes by clicking **Apply These Settings**, the Microsoft Navision Application Server Manager does the following:

- It changes the properties for Microsoft Navision Application Server and places them in the Registry.
- If Microsoft Navision Application Server is running, the Microsoft Navision Application Server Manager notifies Microsoft Navision Application Server that changes have been added to the Registry. The changes take effect immediately. If Microsoft Navision Application Server is not running, the changes will only take effect when you restart Microsoft Navision Application Server.

If the configuration change fails, Microsoft Navision Application Server shuts down, and the program reacts in the following way:

- The value in the **Service Status** field changes to Stopped. An error message is then logged in the Event Viewer.

Message Logging

When you run Microsoft Navision Application Server as a service, it runs without displaying anything onscreen and requires no user interaction. In short, this means that there is no user interface. Consequently, the user is not able to see events or error messages onscreen. Instead, these messages are logged in the Event Viewer. When you run Microsoft Navision Application Server from the command line, you can see messages onscreen. Read more about this in the following section.

Error and Event Logging

There are slight differences with the error and event logging when you run Microsoft Navision Application Server as a service and when you run it from the Microsoft DOS command line. The reason for the differences is that in the former there is no user interface, while in the latter the console user interface is used.

Microsoft Navision Application Server Running as a Service

When you run Microsoft Navision Application Server as a service, events and error messages are written to the Event Viewer. The Microsoft Navision Application Server is registered as an event source, so you can filter the messages in the log.

The Event Viewer is accessible by clicking on START→PROGRAMS→OFFICE AUTOMATION→ADMINISTRATIVE TOOLS→EVENT VIEWER. From the Event Viewer Window, select Application to view any errors related to the Microsoft Navision Application Server.

Microsoft Navision Application Server Running from the Microsoft DOS Command Line

When Microsoft Navision Application Server is started from a command line, all events and error messages are, by default, displayed on the command line. Alternatively, they can be written to a file in the following way:

```
nas {property settings} > file name
```

Microsoft Navision Automated Data Capture System Toolkit

Overview

Using accurate data in warehouse documents is essential to keeping inventory accuracy in regards to item number and quantity. In supply chain operations, companies are experiencing increased pressure for faster operations. Working with batches of data capture represents a problem because it means that the warehouse workers must make a number of return trips to their desks to enter the collected information. Online operation with the use of radio frequency technology provides the user with continuous validation of every single item movement in the warehouse.

Along with the pressure for faster operations, the high staff turnover that many companies experience makes it harder and more time-consuming to make items known to and recognized by a number of inexperienced staff. This means that it is essential for a supply chain company to have a fast and accurate method of recording item data. Currently, the fastest and most reliable method of doing this is by using bar codes in connection with a number of different possible capture systems.

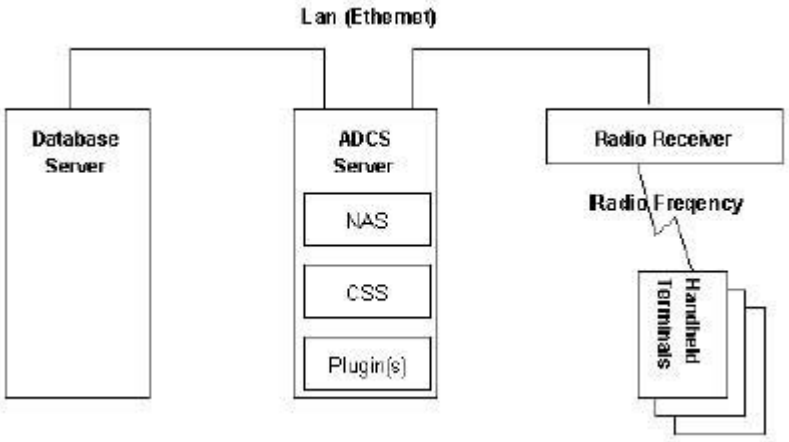
The Automated Data Capture Systems (ADCS) granule provides companies with the necessary functionality to capture accurate data for inbound, outbound, and internal documents, primarily in connection with warehouse activities.

This document first describes how to set up ADCS for Microsoft Navision. It then explains how the operating system is managed in connection with the use of ADCS.

NOTE: *Item tracking information such as lot number and serial number tracking is not supported in this version of ADCS.*

Introduction to ADCS

The ADCS solution is designed around the Microsoft Navision Application Server. The Microsoft Navision Application Server (NAS) can be compared to a normal Microsoft Navision Client without the user interface. However, it is possible to communicate with the NAS by the use of a Communication System Service (CSS) and a plug-in that supports the protocol used by the handheld terminals.

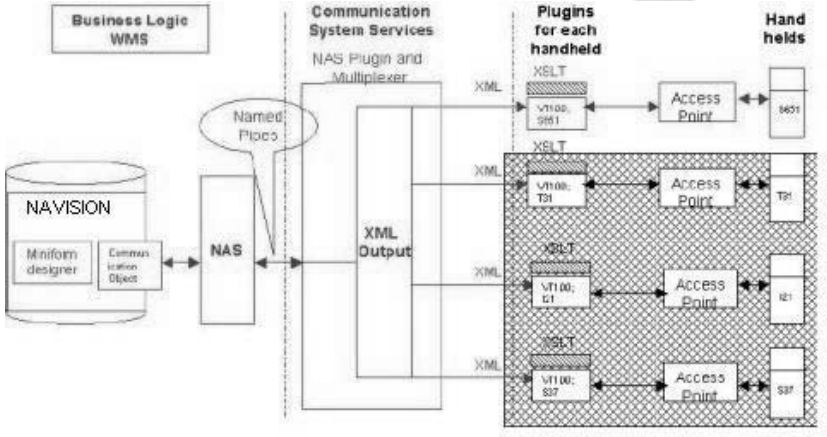


The NAS is responsible for processing data coming to it from the handhelds and sending the appropriate response back to them. A standard XML format is used to send and receive the data.

The Communication System Service is responsible for handling requests directly from the plug-in. The CSS is also responsible for ensuring that the response from the NAS is directed to the correct handheld, thus reducing the load on the NAS.

The plug-in provides the link between the CSS and the Access Points for the handhelds and acts as a virtual Telnet server. All connection requests and data transmissions are handled by this service, and the incoming native protocol is transformed into standard XML that the NAS can interpret. Once a request has been transformed, it is forwarded to the CSS for further transmission to the NAS. When a reply is received, it is converted from XML into the native protocol of the handheld. Function keys are mapped using an XSLT file that may be modified by the end user. Once the data has been transformed and formatted, it is forwarded to the handheld.

The diagram below illustrates the technical design of the ADCS solution. The functionality covered by the checkered area is not part of the standard solution but could be developed by a third party.



For a quick overview of typical procedures associated with specific ADCS activities, read the respective topic in online Help.

Types of Handhelds and Performance

All character-based handhelds that have a VT100 interface will work with ADCS, although you may have to implement minor adjustments with some models. With any handheld, you must be able to connect either the Access Point or the RF terminal to the IP or port address of the ADCS server.

If you want to use a graphics-based handheld, you will have to develop your own plug-in.

A setup with one NAS server and ten remote terminals has been tested with good performance results. In theory, you should be able to connect any number of remote terminals to a NAS server. However, if performance becomes an issue, you can always add an extra NAS server and direct some of the terminals to the new NAS server to achieve improved performance.

About the Installation Guide for ADCS

The purpose of this installation guide is to give the resellers and implementers of ADCS a full understanding of the installation procedures for the application.

Installation Requirements

In order to use the ADCS functionality, additional Microsoft Navision products besides the standard client must be installed. The additional programs are ADCS, Microsoft Navision Database Server and Microsoft Navision Application Server (NAS). In addition, the .NET Framework 1.1 and MSXML version 3 must be installed.

Operating System

This installation guide is based on the Microsoft Windows 2000 Professional operating system.

Microsoft Navision Database Server and Microsoft Navision Application Server require either Windows NT4, Windows 2000 Professional or Windows XP Professional. Windows95, Windows98 or Windows Millenium Edition cannot be used.

We do NOT recommend using a PC where Windows NT4, Windows 2000 Professional or Windows XP Professional has been installed on top of Windows95, Windows98 or Windows Millenium Edition.

Installation Procedure

ADCS has been designed to be used with access to a network. Without network access, it will not work.

However, if ADCS is installed on a PC without access to a network and is used for demonstration purposes, install a Microsoft Loopback network adapter.

Using ADCS without a Network Connection

When not connected to the network, disable the normal network adapter, and enable the Loopback adapter. Do the opposite when there is a connection to a network.

To install the Loopback network adapter, use the Add/Remove Hardware wizard in the Control Panel. During the installation, make the following selections:

- Add/troubleshoot a device
- Add a new device
- No, I want to select the hardware from a list
- Network Adapters
- Microsoft
- Microsoft Loopback Adapter

This installation manual is based on a situation where you have access to a network. The installation of ADCS consists of multiple application installations from the Microsoft Navision 4.00 Product CD.

***IMPORTANT**: In order for the ADCS to function, it is recommend installing the Microsoft Navision Database Server, Microsoft Navision Application Server, the Communication System Service server, and the VT100 plug-in all on the same computer.*

Microsoft Navision Client, Database Server and NAS Installations

The programs that you need are the Microsoft Navision Client, Microsoft Navision Database Server, Microsoft Navision Application Server (NAS), and the ADCS application. We strongly recommend that you start by installing the Client. If you install the Client first, you will not have to specify the path of the database to the database server later. See "Microsoft Navision Client Installation," "Microsoft Navision Database Server," and the prior sections of this chapter for instructions on installing the Microsoft Navision Client, Database Server and NAS. Remember that if there is already an instance of NAS running on the system, the new instance must be created manually. For more information, see the section titled "Running More Than One Microsoft Navision Application Server."

NAS Configuration for ADCS

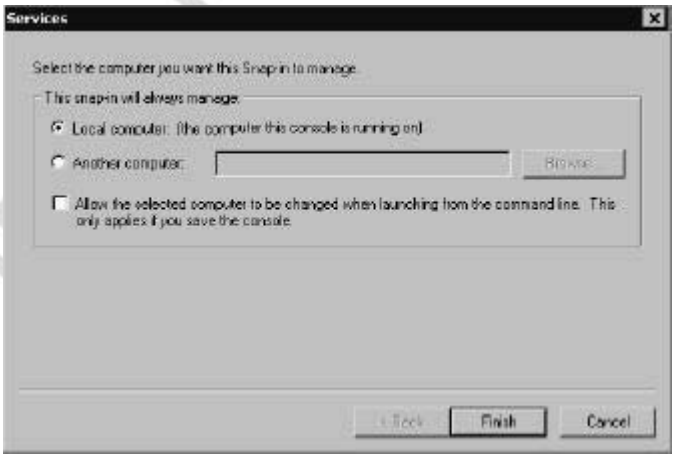
The Microsoft Navision Application Server configuration is done from the Application Server Manager. If a new installation of Application Server has completed, the Microsoft Management Console window should appear if the **Start Microsoft Navision Application Server 4.0 Manager** box was checked at the end of installation. If previously installed, go to START→PROGRAMS→MICROSOFT BUSINESS SOLUTIONS-MICROSOFT NAVISION→MICROSOFT BUSINESS SOLUTIONS-MICROSOFT NAVISION APPLICATION SERVER MANAGER.

To add a Microsoft Navision Application Server, follow this procedure:

1. Right-click the Microsoft Navision Application Server node.
2. Click NEW→APPLICATION SERVER, and the New Microsoft Navision Application Server window appears.
3. Enter the name of the Microsoft Navision Application Server that you want to add and the name of its host machine. The default value that is entered in the **Host Machine** field is localhost.
4. Click on **OK**.
5. Expand the Microsoft Navision Application Server node then select the Application Server created in step 3.

6. The **Company Name** and the **Net Type** fields may be already filled in; if not, fill these fields according to the company information of the application.
7. In the **Database Server Name** field, enter localhost.
8. Change the Company Name to the name of the company that you want the NAS server to access in the Microsoft Navision database.
9. In the **Start-Up Parameter Value** field, enter ADCS ADCSID=NAS1. The first ADCS specifies the NAS Type. ADCS then requires a unique ID for each Application Server. This is specified by the required text ADCSID= followed by any unique character string. In this example, the unique character string is NAS1.
10. In the **Object Cache Size** field, enter 8000.

    In order to make the application server available for the user of the computer, there are some additional preparations. The Windows Services and the Windows Event Viewer must be visible and applied to the Console Management window.
11. From the Console Management window, click FILE→ADD/REMOVE SNAP-IN. In the Add/Remove Snap-in window, click the **Add** button.
12. Select Services and click **Add**.



13. Click **Local Computer**, and click **Finish** to add Services to your Console Management window.
14. In the Add Standalone Snap-in window, select Event Viewer and click **Add** to add the Event Viewer.
15. Click **Services** and select Microsoft Business Solutions-Navision Application Server.
16. Right-click on Microsoft Business Solutions-Navision Application Server and click **Properties**. Then click the **Log On** tab.

17. Click **This Account**, and enter your user name and Windows login password. Confirm the password and click **OK**. If you are connected to a Windows Domain, enter the Windows login as Domain name\login.

NOTE: *Remember to copy your license file to the database and application server folders. Remember to rename the file to fin.flf after copying.*

ADCS Installation

An object file containing ADCS objects as well as a .txt file containing code changes will be provided.

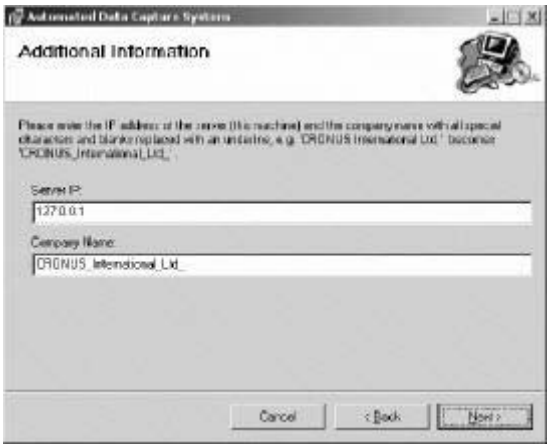
1. Open the MBS Microsoft Navision Database to update, and import the FOB-File provided.
2. Open Codeunit 1 in design mode and implement the changes specified in the file:

   Codeunit1_Procedure99.txt

NOTE: *Make sure the license file you are using is updated to match the new objects included in the FOB-file.*

3. Start the setup located in the directory, and follow the Installation Wizard instructions.
4. During the installation routine, enter the IP address of the server, which is the machine on which you are installing ADCS.

   The IP address of the machine can be found by typing IPconfig into the DOS prompt.
5. Enter the company name of the MBS Microsoft Navision Database.



6. When prompted, select the folder on the hard drive that contains the previously installed Microsoft Navision applications.

ADCS Configuration

Now that the ADCS has been installed, the next step is to configure the ADSC application.

1. On the task bar, click START→PROGRAMS→MICROSOFT BUSINESS SOLUTIONS-MICROSOFT NAVISION→MICROSOFT BUSINESS SOLUTIONS - MICROSOFT NAVISION. The Microsoft Navision application appears.
2. From the menu bar, click TOOLS→SECURITY→WINDOWS LOGINS.
3. Add the same user as on the Microsoft Navision Application Server.
4. Click **Roles**, and add the Role ID "Super."
5. From the menu bar, click TOOLS→SECURITY→DATABASE LOGINS.
6. Add one or more users and assign roles. In this example, select ADMIN as user and SUPER for the role.
7. Go to ADMINISTRATION→APPLICATION SETUP→WAREHOUSE→SETUP - WAREHOUSE→EMPLOYEES.
8. Enter the users you previously inserted, and fill in the Location. In this example, the ADMIN and USERONE users are set up to work in the WHITE location. They can be set up to work with other locations, but the WHITE location is their default location.

Test Your Knowledge – Microsoft Navision Application Server and Automated Data Capture System

1. Microsoft Navision Application Server is a _____ _____, which executes business logic without user intervention.
2. The user cannot use _____ _____ when logging on to Microsoft Navision Application Server.
3. The Microsoft Navision Automated Data Capture System is aimed at companies that need to use _____ _____ in their warehouse processes.
4. If you installed Microsoft Navision Application Server as a service and you set the Startup Type option to _____, Microsoft Navision Application Server will automatically start every time you start the computer.
5. When you set up Microsoft Navision Application Server to run as a service, you can use the _____ _____ _____ to change its properties.
6. In the right-hand pane of the Microsoft Navision Application Server Manager window, you can _____ _____ the properties of Microsoft Navision Application Server.
7. The ADCS granule provides companies with the necessary functionality to capture accurate data for _____ _____ _____ documents, primarily in connection with warehouse activities.
8. When you run Microsoft Navision Application Server as a service, it runs without displaying anything onscreen and requires no _____ _____.
9. When you run Microsoft Navision Application Server as a service, events and error messages are written to the _____ _____.
10. The _____ _____ _____ is responsible for handling requests directly from a plug-in that supports the protocol used by the handheld terminals.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 7: MICROSOFT NAVISION ODBC DRIVER

Overview

Open Database Connectivity (ODBC) is an interface defined by Microsoft Corporation as a standard interface to database management systems in the Microsoft® Windows® 98, Microsoft® Windows® NT, and Microsoft® Windows® 2000 environments. Applications using the ODBC interface can work with many different database systems.

The Microsoft Navision ODBC Driver is the implementation of Open Database Connectivity (ODBC) for Microsoft® Business Solutions–Navision®. The Microsoft Navision ODBC Driver lets you transfer data between a Microsoft Navision database and any program that supports ODBC.

Microsoft Navision ODBC functions largely in the same way as ordinary clients in Microsoft Navision. The main differences are:

- A Microsoft Navision ODBC client is a program other than Microsoft Navision, such as a spreadsheet or word processing program.
- Triggers are not run when a Microsoft Navision ODBC client writes data in a Microsoft Navision database.

The Microsoft Navision ODBC Driver makes data in a database stored locally or in a Microsoft Navision Database Server accessible to ODBC-enabled applications. You can use the Microsoft Navision ODBC Driver to retrieve Microsoft Navision data into an application such as a word processor or spreadsheet.

This chapter assumes that you are familiar with Microsoft Navision and with ODBC functionality. For additional information, refer to the Microsoft Navision documentation, as well as to documentation for the applications into which Microsoft Navision data will be retrieved.

Special Terminology

There are a number of terms used in this chapter that are not used in Microsoft Navision. Below is a short explanation of these terms. If you need a more detailed explanation, see the documentation for the product in which the term is used:

Term	Program	Definition
SQL	Microsoft Excel and Microsoft Query	Structured Query Language: a programming language that is specially designed for queries in databases.
Add-in	Microsoft Excel	A command or function that gives a program additional capabilities.
Visual Basic	Microsoft Excel	A programming language used for programming macros in Microsoft Excel among other things.
Criteria	Microsoft Query	The same as a filter in Microsoft Navision.

Installation

To install the Microsoft Navision ODBC Driver under Windows 2000, Windows XP or Windows 2003, follow this procedure:

1. Start the Microsoft Navision ODBC setup program. You find this in the NODBC subfolder on the Microsoft Navision product CD.

   The Welcome window appears. This is the first of three forms in a standard Windows Installer wizard.

2. Click **Next** to continue, and follow the instructions in the wizard.

   The installer copies the necessary files to the (Program Files\Common Files\Microsoft Navision\NODBC) folder. The installer then registers the Microsoft Navision ODBC Driver and creates a sample Microsoft Navision ODBC data source. The installation is finished when the Installation Complete windows appears.

Canceling the Installation

You can cancel the installation at any time. If you choose to cancel the installation, a dialog box appears asking you to confirm your decision. If you click **No**, the installation process will continue. If you click **Yes**, Windows Installer will perform a full rollback and restore the computer to the state it was in before the installation process began.

Uninstalling and Repairing the Microsoft Navision ODBC Driver

You can also use Windows Installer to repair or remove the Microsoft Navision ODBC Driver.

1. In the Control Panel, select **Add/Remove Programs**.
2. Select Microsoft® Business Solutions–Navision® ODBC Driver 4.0.
3. Click the **Change** button to change your installation of the Microsoft Navision ODBC Driver, or click the **Remove** button to remove the driver from your system.

Windows Installer will repair or remove the Microsoft Navision ODBC Driver automatically, depending on your choice.

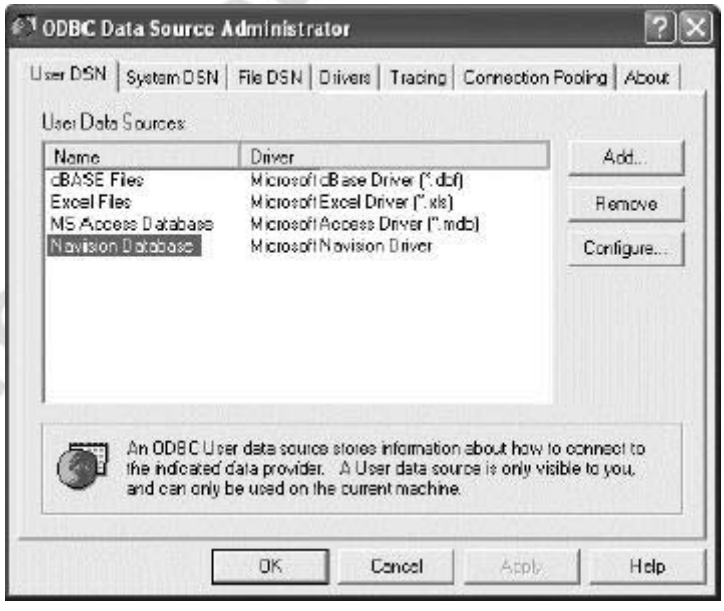
Configuration

After you have installed the necessary files, you can set up the sample data source that has been created by the installation program, and you can add new data sources. This is done from the control panel.

Setting Up a Data Source

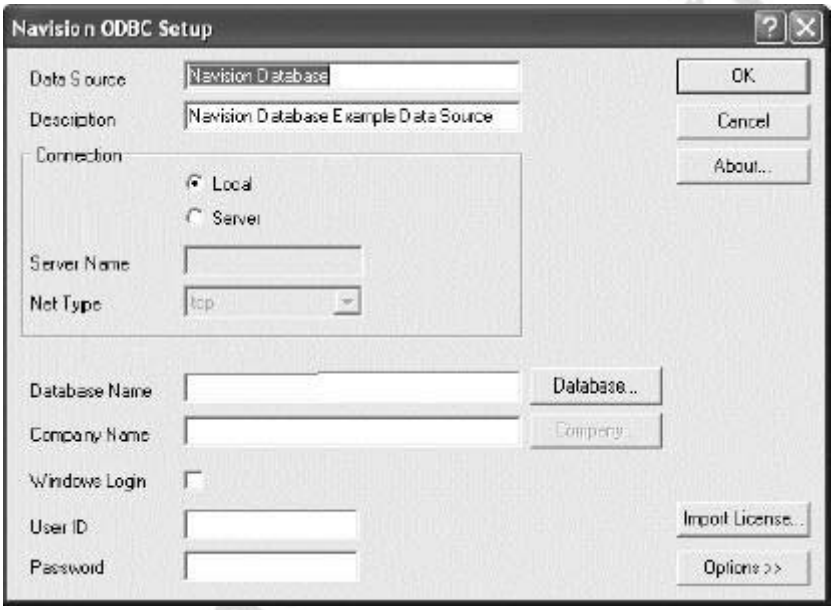
To set up a user data source, follow this procedure:

1. In the Control Panel, click ADMINISTRATIVE TOOLS→DATA SOURCES (ODBC). The following window appears, displaying a list of the data sources that are available on your system:



A data source contains information about where to find the data and how the driver formats the data when it is returned to an application. Each data source is identified by a unique name followed by the name of the driver. You can read about adding, changing, and deleting data sources in the section titled "Adding a Data Source" and the online Help for the ODBC Data Source Administrator explains about User, System, and File Data Sources.

2. Click **Microsoft Navision Database**, and click **Configure**. The Microsoft Navision ODBC Setup window appears:



Fill in the fields according to these guidelines:

Field	Comments	
Data Source	The field is already filled in with a default data source name. This name can be changed.	
Description	Enter a description of the data source.	
Connection	You must specify here whether Microsoft Navision is installed as a single-user or multiuser system (client/server):	
	Local (Default value)	Click here if the driver will function in a single-user installation.
	Server	Click here if the driver will function in a multiuser installation.
Server Name	If you have selected a server connection in the **Connection** field, enter the name of the server, that is, the server where the Microsoft Navision database is located. (If you have selected a local connection, leave this field empty.)	

Field	Comments
Net Type	In a multiuser installation, enter the name of the network protocol program, that is, *tcp* (for TCP/IP) or *netb* for NetBios.
Database Name	Enter the name of the database you want to connect to. You can see a list of available databases by clicking **Database**. Browse to the relevant folder, click the database file name, and then click **Open** to copy the name to the field.
Company Name	Enter the company name from which you want to retrieve data. You can see a list of available company names by clicking **Company**. Click the company name, and then click **OK** to copy it to the field.
Windows Login	Place a check mark in this field if you want to use your Windows login to access the database. Using a Windows login means that you do not have a separate user ID and password.
User ID	Enter the user ID that you use when logging in.
Password	Enter the password for the user ID. There will usually be a special user ID and password set up for the Microsoft Navision ODBC Driver – for example, a user ID with read permissions that has been created for a specific reason. This is what you should use. (You should not enter your personal user ID or password here because others will be able to see it.) If you do not enter anything in the **User ID** and **Password** fields, you will have to enter an ID and password every time you want to open the database from another program. You can read more about security in Chapter 6, "Microsoft Navision SQL Server Option" and Chapter 9, "Microsoft Navision Security."
About…	Opens the About dialog for the driver and displays your license information (if a database and a company are available.)
Import License	Click this button to open a standard Windows browse dialog. Locate and select your license file (.flf). This file is renamed *fin.flf* and copied to the directory where the driver is installed.

Specifying Options

In the Microsoft Navision ODBC Setup window, click the **Options** button to specify the data source options. The Options box appears:

Fill in the fields according to these guidelines:

Field	Comments	
Identifiers	In this field, select one of four options. The options in this field and their implications are described in detail in the section titled "Converting Identifiers."	
Language	All possible languages are shown in the Language list box. Besides the specific language options, the Microsoft Navision ODBC Driver offers the following general options:	
	Neutral	This option disables multilanguage functionality. The Microsoft Navision ODBC Driver will only show Name properties and not captions.
	Auto (Windows Language)	This option uses multilanguage functionality to show captions in the language of the operating system's regional settings.
Option Field Type	Click the **AssistButton** to select one of two ways that option field values can be transferred:	
	Text:	Values are transferred as text strings, that is, the texts that appear in the dropdown list.
	Integer:	Values are transferred as integers. The options in a drop-down list are numbered 0, 1, 2, 3 . . .
Commit Cache	Specifies whether commit cache should be used:	
	Yes (checked)	Use commit cache.
	No (unchecked)	Do not use commit cache.

Field	Comments	
DBMS Cache (KB)	Enter the size of the cache (0–30,000 KB)	
Enable BLOB Fields	Specifies whether BLOB fields should be visible from ODBC:	
	Yes (checked)	BLOB fields can be seen from ODBC.
	No (unchecked)	BLOB fields are hidden.
Read Only	Specifies whether access to the database should be read-only:	
	Yes (checked)	Access is read-only.
	No (unchecked)	Access is read/write.
Closing Date	Specifies whether the connection supports closing dates:	
	Yes (checked)	Closing dates are supported.
	No (unchecked)	Closing dates are not supported.

Converting Identifiers

The option you select in the **Identifiers** field controls the way identifiers such as table names and field names are transferred from Microsoft Navision to an external program. The choice you make affects the way you use identifiers in external programs and the way you must write SQL statements. The options are:

Option	Comments
All Except Dot	Letters, numbers, symbols, and spaces are transferred unchanged. Dots and question marks are converted to underscores (_).
All Characters	Letters, numbers, symbols, dots and spaces are transferred unchanged.
All Except Space	Letters, numbers and symbols are transferred unchanged. Spaces, dots, and question marks are converted to underscores (_).
a-z,A-Z,0-9	Only letters and numbers are transferred unchanged. Symbols (except %), spaces, dots, parentheses, and question marks are converted to underscores (_). The % sign is converted to PCT. The $ sign is converted to USD.

Example

This table shows how the **No.**, **Sales (LCY)**, **Profit %** and **Shelf/Bin No.** fields from the Item table are converted with the four different options:

Field Name	All Except Dot	All Characters	All Except Space	a-z,A_Z,0-9
No.	No_	No.	No_	No_
Sales (LCY)	Sales (LCY)	Sales (LCY)	Sales_(LCY)	Sales__LCY_
Profit %	Profit %	Profit %	Profit_%	Profit_PCT
Shelf/Bin No.	Shelf/Bin No_	Shelf/Bin No.	Shelf/Bin_No_	Shelf_Bin_No_

Using Identifiers in External Programs

In some cases, field names and table names with spaces and/or symbols must be converted by the Microsoft Navision ODBC Driver when they are returned as identifiers to an external program. This is necessary if the external program does not support spaces and/or symbols in identifiers (this may differ from program to program). You specify the kind of conversion that is necessary by choosing one of the options described in the preceding table.

As an example, Microsoft Query does not support identifiers with dots (for example, the **No.** field in many tables). To have Microsoft Query handle these names correctly, use a data source with the All Except Dot option in the **Identifiers** field.

Writing SQL Statements

When writing SQL statements, you must write field names according to the identifier option that has been chosen. In the section titled “Converting Identifiers,” you can read about how the various options work and see some examples of how field names are converted.

If you have chosen the “All Characters,” “All Except Space,” or “All Except Dot” option in the **Identifiers** field, you must use quoted identifiers, meaning include field names in quotation marks. For example, if you have chosen the “All Except Space” option, the **Sales_(LCY)** field name must be written as “Sales_(LCY)”.

If you have chosen the a-z,A-Z,0-9_ option, you do not have to use quoted identifiers.

Adding a Data Source

You can set up multiple data sources with the same driver. For example, you can have different data sources with different databases, or you can have data sources with different options. See the online Help in the ODBC Data Source Administrator for more information about the various types of data sources (User, System and File).

To set up a new user data source:

1. In the Control Panel, click ADMINISTRATIVE TOOLS→DATA SOURCES (ODBC).
2. In the ODBC Data Source Administrator window, click the **User DSN** tab, and then click **Add**.
3. Select the Microsoft Navision ODBC Driver, and click **Finish**.
4. Enter information in the Microsoft Navision ODBC Setup window as described in the section titled Setting Up a Data Source.
5. Click **OK** to close the window.

Changing a Data Source

You can change the information in the data source at ay time with the following steps:

1. In the Control Panel, click ADMINISTRATIVE TOOLS→DATA SOURCES (ODBC).
2. Select the data source you want to change, and click **Configure**.
3. Change the necessary fields in the Microsoft Navision ODBC Setup window by typing or selecting the new names or values.
4. Click **OK** to close the window.

Deleting a Data Source

If you no longer need a data source, you can delete it:

1. In the Control Panel, click ADMINISTRATIVE TOOLS→DATA SOURCES (ODBC).
2. Select the data source you want to delete, and click **Remove**.
3. Confirm the message that appears by clicking **OK**.

Technical Documentation

The Microsoft Navision ODBC Driver opens a Microsoft Navision Database Server or local database to ODBC-enabled applications, so that they can retrieve data from and write data to the database.

The Microsoft Navision ODBC Driver operates in the Windows 2000, Windows XP, and Windows 2003 environments. In these environments, it can function either as a stand-alone or as a client in a client/server configuration. You do not need to have a Microsoft Navision client installed to use the Microsoft Navision ODBC Driver.

***NOTE**: You cannot use Microsoft Navision ODBC Driver with the SQL Server Option for Microsoft Navision.*

Establishing a Connection

The following are examples of how to connect a Microsoft Navision ODBC data source, depending on the programming language you use.

Using C#

When using C# to connect, you could use the following example:

```
string myConnection = "DSN=Microsoft Navision Database";
OdbcConnection myConn = new OdbcConnection(myConnection);
myConn.Open();
...
myConn.Close();
```

Using Visual Basic .NET

The following is an example of how you could connect to a Microsoft Navision ODBC data source when using Visual Basic .NET:

```
oODBConnection As New Odbc.OdbcConnection("DSN=Microsoft
Navision Database")
oODBConnection.Open()
...
oODBConnection.Close()
```

Using C++

The following gives an example of how you could connect to a Microsoft Navision ODBC data source when using C++:

```
SQLHENV henv;
SQLHDBC hdbc;
SQLHSTMT hstmt;
SQLRETURN retcode;

// Allocate environment handle
retcode = SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE,
&henv);

if (retcode == SQL_SUCCESS || retcode ==
SQL_SUCCESS_WITH_INFO) {
// Set the ODBC version environment attribute
retcode = SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
(void*)SQL_OV_ODBC3, 0);
if (retcode == SQL_SUCCESS || retcode ==
SQL_SUCCESS_WITH_INFO) {
// Allocate connection handle
retcode = SQLAllocHandle(SQL_HANDLE_DBC, henv, &hdbc);

if (retcode == SQL_SUCCESS || retcode ==
SQL_SUCCESS_WITH_INFO) {
// Set login timeout to 5 seconds.
SQLSetConnectAttr(hdbc, (void*)SQL_LOGIN_TIMEOUT, 5, 0);
// Connect to data source
retcode = SQLConnect(hdbc, (SQLCHAR*)"Microsoft Navision
Database", SQL_NTS,
(SQLCHAR*) "Admin", SQL_NTS,
(SQLCHAR*) "Pass", SQL_NTS);
...
```

Microsoft Navision ODBC Driver Functionality

The Microsoft Navision ODBC Driver has the functionality required for it to be used as a read/write data source.

SQL Conformance

The Microsoft Navision ODBC Driver is developed using ODBC version 2.5. The specification for this version provides three levels of SQL grammar conformance: Minimum, Core, and Extended. Each higher level provides more fully-implemented data definition and data manipulation language support. ODBC version 2.5 fully supports the Minimum SQL grammar, as well as many Core and Extended grammar statements. The Microsoft Navision ODBC Driver's support for SQL grammar is summarized in the following table.

SQL Grammar Statement	Minimum	Core	Extended
Create Table	o		
Create View		o	
Delete (searched)	o		
Drop Index		o	
Drop Table	o		
Drop View		o	
Insert	o		
Left Outer Join			o
Select	o		
- approximate numeric literal		o	
- between predicate		o	
- correlation name		o	
- date arithmetic			o
- date literal			o
- exact numeric literal		o	
- extended predicates			o
- in predicate		o	
- set function		o	
- time literal			o
- timestamp literal			o
Subqueries		o	
Unions			o
Update (searched)	o		
Use (this is an extension to ODBC SQL grammar)			o

ODBC API Conformance

The Microsoft Navision ODBC Driver fully conforms to the ODBC version 2.5 specification for Core API and Level 1 API and supports most of the Level 2 function calls. The following table lists the ODBC API functions supported by the Microsoft Navision.

ODBC Driver. ODBC Function	Conformance Level
SQLAllocConnect	Core
SQLAllocEnv	Core
SQLAllocStmt	Core
SQLBindCol	Core
SQLBindParameter	Level 1
SQLBrowseConnect	Level 2
SQLCancel	Core
SQLColAttributes	Core
SQLColumns	Level 1
SQLColumnPrivileges	Level 2
SQLConnect	Core
SQLDataSources	Level 2
SQLDescribeCol	Core
SQLDescribeParam	Level 2
SQLDisconnect	Core
SQLDriverConnect	Level 1
SQLDrivers	Level 2
SQLError	Core
SQLExecDirect	Core
SQLExecute	Core
SQLFetch	Core
SQLForeignKeys	Level 2
SQLFreeConnect	Core
SQLFreeEnv	Core
SQLFreeStmt	Core
SQLGetConnectOption	Level 1
SQLGetCursorName	Core
SQLGetData	Level 1
SQLGetFunctions	Level 1
SQLGetInfo	Level 1
SQLGetStmtOption	Level 1
SQLGetTypeInfo	Level 1
SQLMoreResults	Level 2
SQLNativeSql	Level 2

ODBC Driver. ODBC Function	Conformance Level
SQLNumResultCols	Core
SQLNumParams	Level 2
SQLParamData	Level 1
SQLPrepare	Core

Data Types

The Microsoft Navision ODBC Driver supports the following SQL data types:

Microsoft Navision Data Types	SQL Data Types
BIGINTEGER	SQL_VARCHAR
BINARY	SQL_VARBINARY
BLOB	SQL_LONGVARBINARY
BOOLEAN	SQL_BIT
CODE	SQL_VARCHAR
DATE	SQL_DATE / SQL_TIMESTAMP
DATEFORMULA	SQL_VARCHAR
DATETIME	SQL_TIMESTAMP
DECIMAL	SQL_DECIMAL
DURATION	SQL_VARCHAR
GUID	SQL_VARCHAR
INTEGER	SQL_INTEGER
OPTION	SQL_INTEGER / SQL_VARCHAR
RECORDID	SQL_VARCHAR
TABLEFILTER	SQL_VARCHAR
TEXT	SQL_VARCHAR
TIME	SQL_TIME

When using the SQL statement CREATE TABLE, the Microsoft Navision ODBC Driver supports the following data types:

- Binary
- Boolean
- BLOB
- String
- Code

- BCD
- S32
- Date
- Time
- Timestamp

The following rules apply to entering search conditions in the <whereclause>.

Microsoft Navision Data Types	Rules
BOOLEAN	Enter the value 0 or 1 (No is 0, Yes is 1).
INTEGER	Enter a signed integer in the range - 2,147,483,648 to 2,147,483,647 (do not enter the commas).
BIGINT	Use this data type to store very large whole numbers. This data type is a 64 bit integer. **Note**: the SQL data type is SQL_VARCHAR, so you must enter data as a string using single quotes, for example *123456789*.
OPTION	If you have selected Text in the **Option Field Type** field in the Options box in the Microsoft Navision ODBC Setup window, then enter option fields as the option string (in single quotes). If you have selected Integer as the Option Field Type, then you should enter option fields as the numerical option value. Thus, if the first option string is Open, this value will be entered as *Open* if the Option Field Type is Text or as *0* (zero) if the Option Field Type is Integer. Option strings in a set are numbered from 0 (zero) upwards.
DATE	Enter a date in this format: {*d "yyyy-mm-dd"*} where y=year (1752-9999), m=month (01-12) and d=day (01-31). **Note**: if you have closing date support, you must use the timestamp format.
TIME	Enter a time in this format: {*t "hh:mm:ss"*} where h=hour, m=minute and s=second.
DATETIME	Use this data type to store timestamps in this format: {*ts "yyyy-mm-dd hh:mm:ss"*} where y=year, m=month, d=day, h=hour, m=minute and s=second. The timestamp is always shown in local time. DATETIME is always stored in the same format regardless of closing date support.

Microsoft Navision Data Types	Rules
DURATION	Use this data type to represent the difference between two datetimes in milliseconds. This value can be negative. It is a 64 bit integer, and you must enter data as a string using single quotes. You can enter the data either as a number, such as *122* or as text, such as *2 min 2 sec.* **Note:** the data type has a property "Standard date time unit" where you can set the standard unit of measure. Microsoft Navision ODBC users must know this unit of measure if they enter data as a number, since Microsoft Navision interprets input such as 60 as 60 milliseconds, seconds, minutes, hours or days depending on the standard date time unit. The duration will always be displayed in a readable format such as 2 min, 2 sec, rather than the number 122.
TEXT / CODE	Enter a string in single quotes.
DECIMAL	Enter a number (without quotes). The decimal precision of Microsoft Navision ODBC is 16, giving you a range from -999 999 999 999.99 to +999 999 999 999.99.
DATEFORMULA	Enter a date formula as 1Q or 1W+1D.
GUID	Use this data type to give a unique identification number to any database object in this format: {*12345678-1234-1234-12341234567890AB*}. Each character denotes a hexadecimal character.

If an invalid value is used, an error message will be displayed.

Comparison Operators

The Microsoft Navision ODBC Driver uses the following comparison operators:

Operator	Function
=	Equals
<=	Less than or equal to
>=	Greater than to equal to
<	Less than
>	Greater than
<>	Not equal to

Operator Precedence

An important property of an operator is its precedence. Precedence determines the order in which the Microsoft Navision ODBC Driver evaluates different operators in the same expression. When evaluating an expression containing multiple operators, the driver evaluates operators with higher precedence before those with lower precedence. The driver evaluates operators with equal precedence from left to right within an expression.

The following table lists the levels of precedence among SQL operators from high to low. Operators listed on the same line have the same precedence.

Precedence	Operator	Associate
1. (Highest)	( )	Left to right
2.	MUL DIV	Left to right
3.	ADD SUB	Left to right
4.	EQ GE GT LE LT NE	Right to left
5.	NOT	Left to right
6.	AND	Left to right
7.	ALL ANY BETWEEN IN LIKE OR	Left to right

Parentheses within an expression override operator precedence. The driver evaluates expressions inside parentheses before those outside.

Data Type Comparison Rules

This section describes how the driver compares values within each data type.

Numerical Values

A larger value is considered greater than a smaller one. All negative numbers are less than all positive numbers. Thus, -1 is less than 100; -100 is less than -1.

Date Values

A later date is considered greater than an earlier one.

A date entered in the SQL statement `{d "1995-12-31"}` is considered an ordinary date – not a closing date.

However, the Microsoft Navision ODBC Driver supports closing dates. To support closing dates, in the Microsoft Navision ODBC Setup window click **Options**, and place a check mark in the **Closing Date** field. When you have enabled closing date support, you must enter data in a field with the DATE data type in the following format: `{`*`ts "2001-01-01 59:59:59"`*`}`. The time part can hold one of two values: `23:59:59`, which means a closing date and `00:00:00`, which means an ordinary date. The default setting of the Closing Date option is disabled.

Note that even with the use of the SQL statement `{ts "2001-01-01 59:59:59"}`, fields with the DATE data type do not store timestamps. To store the actual timestamp, use the DATETIME data type.

Character String Values

Character values are compared using non-padded comparison semantics. This means that the driver compares two values character-by-character until it finds a character that varies. The value with the greater character in that position is considered the greater value. If two values of different lengths are identical up to the end of the shorter one, the longer value is considered greater. If two values of equal length have no differing characters, then the values are considered equal.

Example: "`Str 2`" is greater than "`Str 10`."

Comparing Option Fields

If, when setting up the options in the Options box in the Microsoft Navision ODBC Setup window, you selected Text as the Option Field Type, the comparison operators use the option strings as a basis for comparison for option fields. If you selected Integer as the Option Field Type, then the comparison operators use the numerical values of the options to compare the values in option fields.

Multilanguage Functionality

The Microsoft Navision ODBC Driver handles the multilanguage functionality in Microsoft Navision. The Microsoft Navision ODBC Driver can retrieve the application data from Microsoft Navision in different languages independent of the current Microsoft Navision application language.

When you are running Microsoft Navision ODBC and you open the ODBC Data Source Administrator window from the operating system's Control Panel, you can set up the Microsoft Navision ODBC Driver. Use the **Language** field properties to set up the connection appropriate to the user.

C/SIDE uses the following hierarchy when showing the application data:

1. Global language
2. Primary language of global language
3. Application language
4. Primary language of application language.

For more information about multilanguage functionality, see the manual Application Designer's Guide.

Specifications

The Microsoft Navision ODBC Driver covers the following multilanguage features:

- Table name
- Field name
- OptionString value

When you link a table by setting another application language other than the default language and this language has a corresponding output from Microsoft Navision, you will notice that the value of the table name, all the field names, and the option fields within that table will be shown in the chosen language.

Limitations

You cannot use the Microsoft Navision ODBC to create a table with multilanguage Caption properties. This means that no matter what language has been chosen in the Options box, in the Microsoft Navision ODBC Setup window the table that is created for any Microsoft Navision ODBC application will use the Name property. No Caption property can be written to the application database.

You will not be able to change the language choice in real-time mode. Microsoft Navision ODBC can only accept one setting at the time of loading. To switch language when using the Microsoft Navision ODBC Driver, you must close the Microsoft Navision ODBC connection and thereby release it from the memory, change to the preferred language in the Options box in the Microsoft Navision ODBC Setup window, and start the Microsoft Navision ODBC connection again.

Scenarios

There are three different scenarios that are possible when running multilanguage for the Microsoft Navision ODBC Driver. In the following scenarios, the user has the following settings:

Property	**Setting**
Operation system regional setting	German (Austrian)
Code base language	English (United States)
Available license file granules and language folders	English (United Kingdom) German (Standard) German (Austrian)

Neutral

In the first scenario, the global language of the Microsoft Navision application is English (United Kingdom), and the **Language** field in the choice in the Microsoft Navision ODBC Options box is set to Neutral.

Microsoft Navision ODBC Result

The application data retrieved and shown using the field and table names as they are in their respective Name properties in Microsoft Navision, and the Multilanguage captions are not used.

Auto (Windows Language)

In this scenario, the global language of the Microsoft Navision application is English (United Kingdom) but the **Language** field in the choice in the Microsoft Navision ODBC Options box is set to Auto (Windows Language).

Microsoft Navision ODBC Result

The application data retrieved is shown in German (Austrian) if the selected objects have multilanguage captions for the language code German (Austrian).

If the selected objects do not have multilanguage captions with the language code for German (Austrian), the application data is shown in the code base language, English (United States). If there is no caption in Microsoft Navision for English (United States), the Microsoft Navision ODBC Driver uses the field and table names as they are defined in their respective Name properties in Microsoft Navision.

Specific Language

In this scenario, the global language of the Microsoft Navision application is still English (United Kingdom) but the **Language** field in the Microsoft Navision ODBC Options box is set to German (Austrian).

Microsoft Navision ODBC Result

The application data retrieved is shown in German (Austrian) if the selected objects have multilanguage captions for the language code for German (Austrian).

If the selected, objects do not have multilanguage captions with the language code for German (Austrian), the application data is shown using the field, and table names as they are defined in Microsoft Navision.

SQL Statement Reference Guide

This section describes all supported SQL statements and serves as a reference guide.

The SQL statement reference guide is organized top-down, starting with the statements and proceeding to a description of the possible elements in the statements (clauses and predicates).

Conventions Used in the Reference Guide

The following graphical conventions are used:

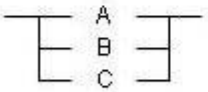
Indicates the beginning of a statement.

Indicates that the statement syntax is continued on the next line.

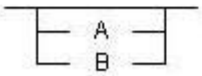
Indicates that the statement syntax is continued from the previous line.

Indicates the end of a statement.

Denotes the repeat symbol. Terms enclosed within the repeat symbol may be repeated any number of times with varying values.



Multiple choices for parameters are enclosed in boxes with horizontal lines. There will be as many lines as there are choices.



Optional parameters are enclosed in lines descending from the main diagram line, as shown in the diagram above. The statement is correct without the optional parameters. If the parameter is not specified, an underscore indicates the default value.

Some complex diagrams have been broken up by grouping several parameters and clauses by a specified name in the main diagram. This specified name is enclosed in angle brackets (<>). Such complex statements are later represented using sub-unit diagrams. A sub-unit diagram starts with

and ends with

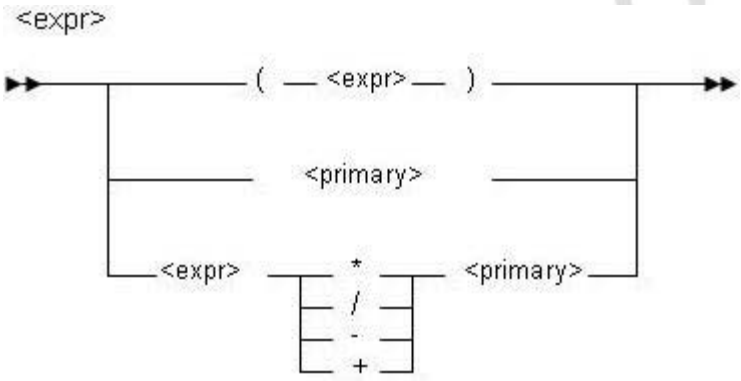
Uppercase letters denote keywords.

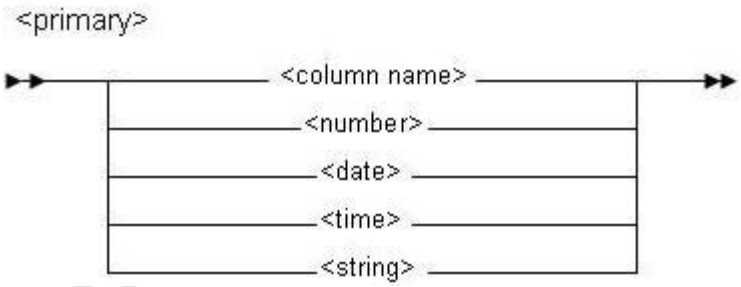
***NOTE**: This convention is used only to make the syntax diagrams easier to read: the SQL reserved words (keywords) are not case-sensitive.*

Notice that all the examples of SQL statements in this section are written assuming that the driver has been set up with the a-z,A-Z,0-9,_ option in the **Identifiers** field.

Expressions

The following diagram illustrates how expressions are constructed in the Microsoft Navision ODBC Driver:

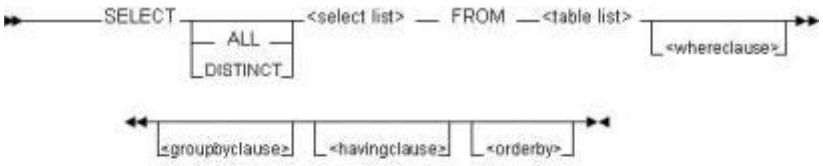




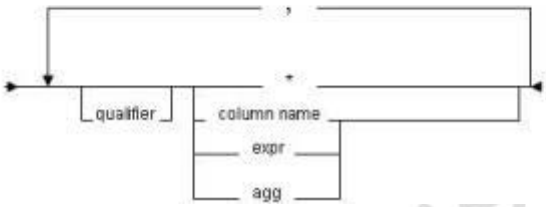
Select Statement

Selects rows from one or more tables.

Syntax



<select list>



General Rules

The SELECT statement retrieves data from one or more tables. It takes the tables listed in the <tablelist> as input and produces an output table that includes only those rows that satisfy the search condition specified in the <whereclause>.

By default, all rows that satisfy the search condition are included in the output table. You can, however, prevent duplicate rows from being included by using the DISTINCT keyword.

Syntax Rules

Syntax	Definition
qualifier	The name of a table or its alias, if one has been specified, in the FROM clause. If only one table is specified, the qualifier is not needed.
asterisk (*)	This symbol includes all columns of the table.
column name	The specific column.
expr.	A field that contains an expression, with or without a column name.
agg.	An aggregate function. There are these aggregate functions: COUNT(* | expr), AVG(expr), MAX(expr), MIN(expr), SUM(expr).
<tablelist>	The <tablelist> lists the tables (and aliases) used in the SELECT statement. See Diagram #1 below

Syntax	Definition
<whereclause>	The <whereclause> specifies the search condition against which the rows in the <tablelist> are evaluated. See Diagram #2 below
<search condition>	See Diagram #3 below
<predicate>	See Diagram #4 below

Diagram #1

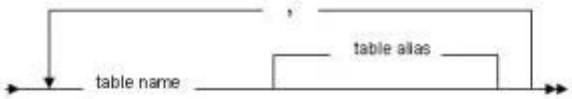


Diagram #2

Diagram #3

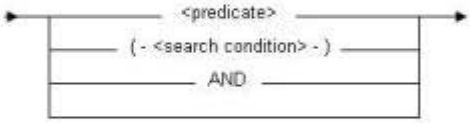
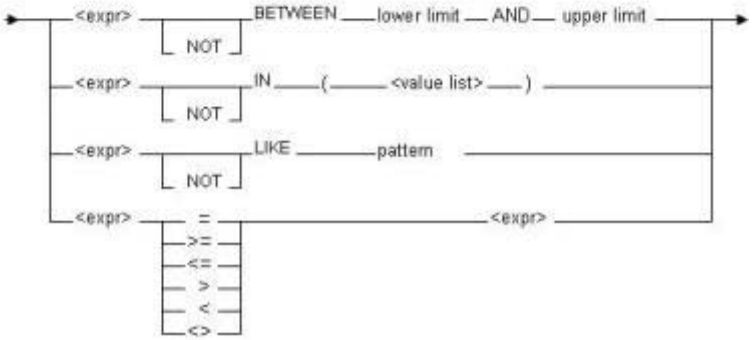


Diagram #4



Example

This example produces a list of customers whose balance is greater than or equal to 20000:

```
SELECT * FROM Customer WHERE Balance >= 20000
```

Parameter Markers

A parameter is a variable in an SQL statement. For example, suppose an Item table has columns named No., Description, and Price. To add a part without parameters would require constructing an SQL statement such as:

```
INSERT INTO Item (No_, Description, Unit_Price) VALUES
("70012," "Glass Door," 75)
```

Although this statement inserts a new order, it is not a good solution for an order entry application because the values to insert cannot be hard-coded in the application.

An alternative is to construct the SQL statement at run time using the values to be inserted. This also is not a good solution because of the complexity of constructing statements at run time.

The best solution, if the client application supports it, is to replace the elements of the VALUES clause with question marks (?) or parameter markers:

```
INSERT INTO Item (No_, Description, Unit_Price) VALUES (?,
?, ?)
```

The parameter markers are then bound to application variables. To add a new row, the application has only to set the values of the variables and execute the statement. The driver then retrieves the current values of the variables and sends them to the data source.

An application cannot place parameters in the following locations:

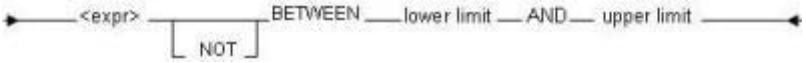
- In a SELECT list
- As both expressions in a comparison-predicate
- As both operands of a binary operator
- As both the first and second operands of a BETWEEN operation
- As both the first and third operands of a BETWEEN operation
- As both the expression and the first value of an IN operation
- As the operand of a unary + or – operation

Predicates in WHERE Clauses

WHERE Clause BETWEEN Predicate

Compares a value to a range of values.

Syntax



General Rules

The BETWEEN predicate checks a value against a range bounded by the lower and upper limits. The condition is true if the value being checked is greater than or equal to the lower limit and less than or equal to the upper limit. Each row for which this condition is true is included in the result set. The value being compared should be comparable with the lower and upper limits.

By using the logical operator NOT, you can test a value outside the specified range.

One important thing to remember about the BETWEEN predicate is the order of the lower and upper limits. The lower limit must be less than or equal to the upper limit.

Syntax Rules

Lower limit: The lower limit of the range that is being checked.

Upper limit: The upper limit of the range that is being checked.

Example

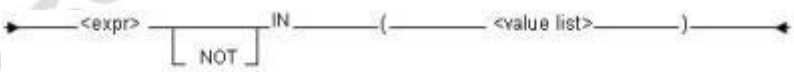
This example retrieves all customers with a post code in the range of 1000 to 1234:

```
SELECT * FROM Customer WHERE Post_Code BETWEEN "1000" AND
"1234"
```

WHERE Clause IN Predicate

Compares a value against a list of values for equality.

Syntax



General Rules

The IN predicate checks a value against a set of values for equality. The condition is true if the value being compared matches any of the values in the value list. If you use the logical operator NOT, the checking principle is reversed.

Syntax Rules

Value list: a list of values against which a value is checked for equality.

Example

This example retrieves all customers whose post code is 1000, 2000, or 3000:

```
SELECT * FROM Customer WHERE Post_Code IN ("1000," "2000,"
"3000")
```

WHERE Clause LIKE Predicate

Compares a string value against a pattern for equality.

Syntax

General Rules

The LIKE predicate compares a string type value with a pattern. The condition is true if a match is found, false if it is not. Every row for which the condition is true is included in the result set.

The pattern is any character pattern against which the value is compared, and it may include some special characters. A percent sign (%) matches any number of characters including zero characters in the same position.

If you want to include a percent sign in the search pattern, then enter a backslash (\) before it to remove its special meaning. For example, \\ represents the backslash itself.

The logical operator NOT negates the LIKE predicate.

Syntax Rules

Pattern: The string against which a value is compared.

Example

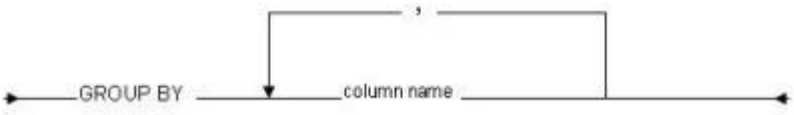
This example retrieves all customers whose name contains Hansen:

```
SELECT * FROM Customer WHERE Name LIKE "%Hansen%"
```

GROUP BY Clause

This groups rows of data based on the value in one or more columns.

Syntax



General Rules

The GROUP BY clause groups the selected rows of data according to values in the group column referred to by the column name. This produces an output that contains one row for each distinct value in the group column.

When you use the GROUP BY clause, the value expression list in the SELECT statement can only contain group columns, expressions containing group columns and aggregate functions. Otherwise, an error will occur.

If you specify multiple columns, the selected rows will be grouped first according to the first group column and within that grouping according to the second group column.

See the section titled "Select Statement" for a list of the aggregate functions that are available. In all of the aggregate functions, the value expression can be quantified by a quantifier that can be either DISTINCT or ALL. The default is ALL, which means that all values of the value expression for all rows in the group should be considered. If the quantifier is DISTINCT, only distinct values of the value expression in the rows of the group are considered for computing the value of the function.

Syntax Rules

Column name: The name of the column on which rows will be grouped.

Example

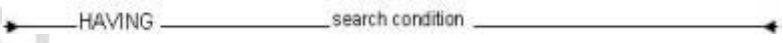
This example retrieves the number of customers per country. We join the Customer and the Country tables in order to get the names of the countries instead of the country codes:

```
SELECT a.Name, Count(*) FROM Country a, Customer b WHERE
a.Code = b.Country_Code GROUP BY a.Name
```

HAVING CLAUSE

This specifies conditions for including groups in the output.

Syntax



General Rules

The HAVING clause makes it possible to specify conditions on grouped data so as to eliminate some of them and include the rest in the output.

There is an important difference between the HAVING and the WHERE clauses. The WHERE clause filters rows before they are passed on to the GROUP BY clause. The HAVING clause filters the output of the GROUP BY clause, and you can use aggregate functions in the HAVING clause.

Syntax Rules

Search condition: Refers to the search condition that you may specify on grouped rows so as to include them selectively in the output. See the section titled "Select Statement" for a diagram of search conditions.

Example

This example retrieves the name and balance of all customers with a balance that is greater than 5000 and sorts the list by balance. This statement could be rewritten using WHERE.

```
SELECT Name, Balance FROM Customer
GROUP BY Name, Balance HAVING Balance > 5000
ORDER BY Balance DESC
```

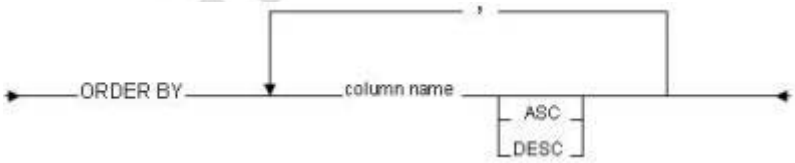
This example retrieves information from the Sales Invoice Header table. It produces a list of customers and the average, maximum, and minimum amounts of their sales invoices, but only if there is more than one invoice for the customer. This statement could not be rewritten using WHERE.

```
SELECT Sell_to_Customer_Name, Avg(Amount), Max(Amount),
Min(Amount) FROM Sales_Invoice_Header
GROUP BY Sell_to_Customer_Name HAVING Count(*)>1
```

ORDER BY CLAUSE

This sorts the output of a query in ascending or descending order on the basis of values in one or more columns.

Syntax



General Rules

The ORDER BY clause sorts the output of a query in the desired order. By default, rows are sorted in ascending order of values. To reverse the order of the sort, use the keyword DESC.

You can sort the output of a query by sorting multiple columns. In this case, the output is first sorted by the values in the first column. Within each distinct value in the first column, the rows are sorted by the values in the second column, and so on. When sorting rows based on multiple columns, each column can be assigned its own sorting order with ASC or DESC.

Syntax Rules

Column name: The column in which the data will be sorted.

Example

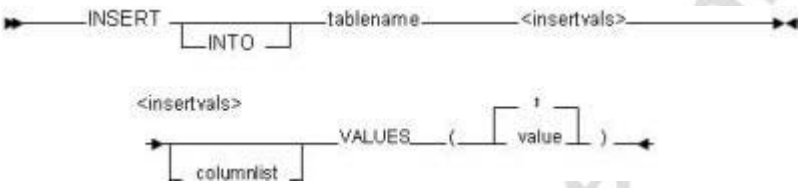
This example retrieves all customers sorted by post code and name:

```
SELECT * FROM Customer ORDER BY Post_Code, Name
```

INSERT Statement

This inserts rows into a table.

Syntax



General Rules

The INSERT statement inserts data into a table. If you are inserting data in only some of the columns in the table, you must explicitly mention these column names in the INSERT statement. If you do not provide a column list explicitly, the INSERT statement will try to insert values in all columns of the table.

There should be as many data values as there are columns in the table or the column list, and the corresponding data types should match. If they do not, an error will be reported and no values will be inserted.

You can insert explicit data values one row at a time using the VALUES clause. Each data value should be separated from the next by a comma.

You can only insert records in tables where your license gives you permission to insert. You cannot insert into a virtual table.

You should be aware that triggers are not run when you insert records through the Microsoft Navision ODBC Driver.

Syntax Rules

Tablename: Name of the table into which you are inserting rows.

Columnlist: A list of columns in the table into which you are inserting data values. The column names must be separated by commas.

Value: Refers to a data value that is being inserted into a column. When specifying a data value, follow these conventions:

- Character-type values must be enclosed in single quotes.
- Date-type and time-type values must be enclosed in the {d"<date value>"} and {t"<time value>"} formats respectively, unless you have closing date support in which case you must use the {ts"<date and time value>} format instead.

  You can also use the DATETIME data type to store the actual timestamp in the following format: {ts"<date and time value>}. For more information about date formats, see the section titled "Date Values."
- All numeric values can be entered literally as values.
- A data value can be expressed as an expression provided the expression evaluates to a type that is compatible with the base data type of the column.
- To identify data as undefined data, use the date 1753-01-01.
- In the case of a partial column list, the values for columns that are not in the column will be set to the null value implicitly.

Example

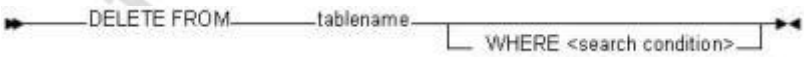
This example inserts a record in the Country table. The column list contains only two of the columns of the table. The rest of the columns will be inserted as null values.

```
INSERT INTO Country (Code, Name)
VALUES ("NZ," "New Zealand")
```

DELETE Statement

Deletes rows in the specified table.

Syntax



General Rules

The DELETE statement deletes rows from a table. If no conditions are specified, all rows in the table will be deleted.

You can optionally specify a WHERE clause to select the rows from the table for deletion. The WHERE clause can specify any valid search condition that selects the rows. You can select the WHERE clause in the same way as you do in a SELECT statement.

You can only delete records in tables where your license gives you permission to delete. You cannot delete from a virtual table.

You should be aware that triggers are not run when you delete records through the Microsoft Navision ODBC Driver.

Syntax Rules

Tablename: Name of the table from which you are deleting rows.

Search condition: This refers to a condition for choosing rows for deletion from the named table. You may specify any valid condition that you can use in the WHERE clause of a select statement.

Example

This example deletes all rows from the Customer table:

```
DELETE FROM Customer
```

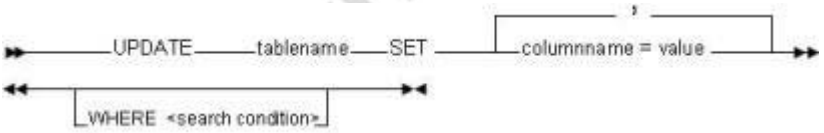
This example deletes a single customer from the Customer table:

```
DELETE FROM Customer WHERE No_ = "12345"
```

UPDATE Statement

This updates rows in a table.

Syntax



General Rules

The UPDATE statement updates data in a table. If no conditions are specified, all rows in the table are updated.

Set values into the columns to be updated by using the SET clause. The left-hand side of the SET clause must be a column in the table being updated. The right-hand side must contain a data value that can be assigned to the column.

You can optionally specify a WHERE clause to select the rows from the table for update. The WHERE clause can specify any valid search condition that selects the rows.

You can only update record in tables where your license gives you permission to update. You cannot update in a virtual table.

You should be aware that triggers are not run when you update records through the Microsoft Navision ODBC Driver.

Syntax Rules

tablename: name of the table in which you are updating rows.

columnname: this refers to the name of the column in the table whose data is being updated.

search condition: this refers to a condition for choosing rows for updating from the named table. You may specify any valid condition that you can use in the WHERE clause of a select statement.

Example

This example updates the **Unit Price** field in all records of the Item table:

```
UPDATE Item SET Unit_Price = Unit_Price * 1.25
```

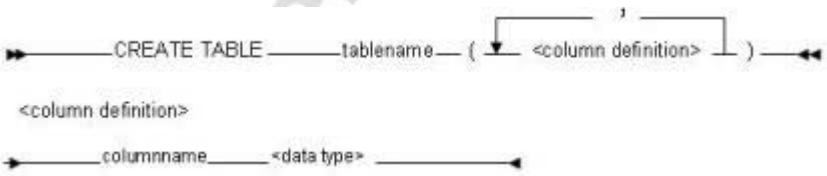
This example updates the **Country Code** field in selected records of the Customer table:

```
UPDATE Customer SET Country_Code = "CN"
WHERE Country_Code = "HK"
```

CREATE TABLE Statement

This creates a table.

Syntax



General Rules

The CREATE TABLE statement creates a table in the database.

You can define the columns of the table by specifying a column name and its data type. The data types that are supported are shown in the syntax rules below. Some of the data types optionally take one or more numeric arguments. In the case of character data types, the length of the column can be specified. In its absence, a default value of one (1) is assumed. The length specifies the maximum length of the string data that can be stored in the column (the declared length in C/SIDE).

When a table is created, the Microsoft Navision ODBC Driver generates a table number automatically. You must have the necessary permissions to insert tables, and there must be free table numbers in the database. The range 49,999 – 99,999 is used, with allocation starting from the top.

Syntax Rules

tablename: name of the table you wish to create. A table name can be up to 30 characters long, and it must be unique within a database. A table number will be generated automatically.

columnname: name of the column (field) being defined. A column name can be 30 characters long, and it must be unique within the table. A field number will be generated automatically.

data type: the data type of the column (field) being defined. The following table shows the relationship between the data types you can use in the Microsoft Navision ODBC Driver and the C/SIDE data types.

Microsoft Navision ODBC Type	C/SIDE Type	Comments
BCD	Decimal	
BLOB	BLOB	
BINARY	BINARY	Takes one argument: length.
BOOL	Boolean	
CODE	Code	Takes one argument: length.
DATE	Date	
TIMESTAMP	DateTime	
S32	Integer	
STRING	Text	Takes one argument: length.
TIME	Time	

Example

This example creates a table with three fields: an **Integer**, a **Decimal** and a **Text** field:

```
CREATE TABLE Sample (
    Code S32,
    Value BCD
    Name STRING(50)
)
```

DROP TABLE Statement

Drops a table from the database.

Syntax

General Rules

The DROP TABLE statement drops the named table from the database. When a table is dropped (deleted), all data in the table is lost.

You must have the necessary permissions in order to drop (delete) a table.

Syntax Rules

tablename: name of the table to drop.

Example

The following example drops (deletes) the table named "Sample" from the database:

```
DROP TABLE Sample
```

Microsoft Navision Flowfields

Microsoft Navision has a special field type called a FlowField, which contains values from other tables. As the values in the original tables change, the values in the FlowField change accordingly. FlowField values are retrieved by applying a Microsoft Navision field class − called a FlowFilter − to the FlowField.

The data type of a FlowFilter is always SQL_VARCHAR (string). The syntax of the FlowFilter is specific to Microsoft Navision.

Setting a FlowFilter on a FlowField is done as a work-around in the WHERE clause. An example for the syntax is:

```
{pred SetFlowFilter, "<TableName>", "<FieldName>", "A
searchString"}
```

The section in brackets is called the extended predicate. The following example shows how to set a FlowFilter on the **Customer Filter** field in the Currency table.

```
SELECT * FROM Currency
WHERE {pred SetFlowFilter, "Currency," "Customer_Filter,"
"10000..40000"}
```

SetFlowFilter is the name of the extended predicate, Currency is the table containing the filter, and Customer_Filter is the field being filtered. The expression "10000..40000" is the filter that will be set on the **Customer Filter** field in the Currency table.

The extended predicate always returns a value of TRUE, so if you use an OR expression with an extended predicate, the value of the entire expression will always be TRUE. For example, consider the following statement:

```
SELECT * FROM Currency
WHERE {pred SetFlowFilter, "Currency," "Customer_Filter,"
"10000..40000"} OR
Last_Date_Modified > 01.01.04
```

This SELECT statement returns all the records in the table because the first condition in the WHERE clause, `{pred SetFlowFilter,"Currency," "Customer_Filter," "10000..40000"},` is always TRUE. The OR operator has no effect.

Consider the following statement:

```
SELECT * FROM Currency
WHERE {pred SetFlowFilter, "Currency," "Customer_Filter"
"10000..40000" } AND
Last_Date_Modified > 01.01.04
```

This will return all records where the customer number falls within the filter and where Last Date Modified is greater than 01.01.04.

Test Your Knowledge – Microsoft Navision ODBC Driver

True	False		
❑	❑	1.	You can use the Microsoft Navision ODBC Driver to retrieve Microsoft Navision data into an application such as a word processor or spreadsheet.
❑	❑	2.	Criteria is a command or function that gives a program additional capabilities.
❑	❑	3.	A database contains information about where to find the data and how the driver formats the data when it is returned to an application.
❑	❑	4.	The Microsoft Navision ODBC Driver can retrieve the application data from Microsoft Navision in different languages independent of the current Microsoft Navision application language.
❑	❑	5.	You can use the Microsoft Navision ODBC to create a table with Multilanguage Caption properties.
❑	❑	6.	The acronym ODBC stands for Open Database Correlation.
❑	❑	7.	Visual Basic is a programming language that is specially designed for queries in databases.
❑	❑	8.	The option selected in the Identifiers field controls how table names and field names are transferred from Microsoft Navision to an external program.
❑	❑	9.	In the All Except Space Identifier option, the Profit % field would be transferred over as Profit_PCT.
❑	❑	10.	In the a-z,A-Z,0-9 Identifier option, the Sales (LCY) field would be transferred over as Sales_LCY_.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 8: MICROSOFT NAVISION SECURITY

Overview

An enterprise business solution must have a built-in security system that protects your database and the information that it contains from being accessed by unauthorized people. It must also allow you to specify what the authorized users are allowed to do in the database – whether they can read, enter, or modify data.

The minimum acceptable level of security requires that each user is assigned an ID and a password. This ensures that only authorized personnel can gain access to your database. This is database level security.

A medium level of security requires that you can limit the users' access so that they can only access certain types of information stored in the database. In other words, they can only gain access to particular tables in the database. This is table level security.

A high level of security requires that you can limit the access that users have to the information that is stored in the tables – that they can only gain access to specific records in the tables. This is record level security and is available using the Microsoft® SQL Server® Option for Microsoft® Business Solutions–Navision®.

This chapter explains how the various parts of the security system work together. It also discusses how to set up an effective security system that controls the access that each user has to the program as well as the rights that he or she has within the program.

Authentication in Microsoft Navision

With several security systems interacting, the terminology can be confusing, so before explaining how the Microsoft Navision security system works, it is necessary to clarify two key concepts:

- Authentication: the process by which the system validates the users identity. This can be done by having the user enter an ID and password when she logs on.
  - Microsoft Navision supports two kind of authentication: Windows authentication and database server authentication.
- Login: when a user has identified herself and been recognized by the system, she is granted access to the parts of the system for which she has permission.

  If the user has used Microsoft® Windows® authentication to log on to the system, then she has been assigned a Windows login.
  If the user has used database server authentication to log on to the system, then she has been assigned a database login.

Windows Authentication

One of the main features of Windows security is the single sign-on system. Microsoft Navision supports this feature and can also use more of the features contained in the Active Directory security system.

The Windows single sign-on and the unified login supported by Windows are the same. In this manual, we will refer to both of these systems as Windows authentication.

With Windows authentication, when a user tries to connect with the server to open a database, he will not have to supply a user ID or password. Microsoft Navision will automatically ask Windows to confirm whether or not this user, who has already logged on to the network, has a valid Windows account and whether this account gives him the right to access this particular server.

If the user is allowed to access the server, then Microsoft Navision will check to see if the user has been assigned a Windows login within Microsoft Navision. If the user has a Windows login, he will be granted access to the database.

The user will be granted access to Microsoft Navision and be given the permissions specified for that Windows user and those specified for any Windows groups of which he is a member.

If the user does not have a valid Windows account or if her account does not include permission to log on to the Microsoft Navision database, authentication fails and the user receives an error.

Advantages of Windows Security

The Windows authentication system includes the following security features:

- Secure validation and encryption of passwords
- A time limit on passwords
- Minimum password length
- Account lockout after an invalid password is entered

Database Server Authentication

If the server does not support Windows authentication, then database server authentication must be used. It is also used when the network administrator has chosen not to use Windows authentication. The Microsoft Navision administrator decides which kind of authentication each individual user should use by assigning each user or group a Windows login or a database login.

If the Microsoft Navision administrator has decided to use database server authentication, he must assign each user a database login. This entails creating a user ID and password for the user within Microsoft Navision. The user will have to enter this user ID and password when she tries to access a database.

The user ID and password must be created by a system administrator, superuser, or somebody else with permission to create Microsoft Navision users. If the user has a valid user ID and supplies the correct password, access will be granted to the relevant areas of the system. If the user ID does not exist or if the password supplied by the user is not valid, then authentication fails and the user receives an error message.

In the SQL Server Option for Microsoft Navision, the database server authentication is based on Microsoft's SQL Server authentication. For more information, see "Microsoft Navision SQL Server Option."

Active Directory Service Security

The Active Directory Service gives Microsoft Navision several new security features. These include allowing administrators to:

- Grant or deny users access to Microsoft Navision by simply adding them to or deleting them from a Windows security group.
- Grant other people in the organization the power to create and administer users and groups (for example heads of departments).

Active Directory Security also supports Windows authentication and provides you with:

- Secure validation and encryption of passwords.
- A time limit on passwords.
- Minimum password length.
- Account lockout after an invalid password is entered.

Active Directory and Microsoft Navision

To take full advantage of the features provided by the Active Directory Security system, the Microsoft Navision client computers, and the domain controller must all either be running on Microsoft® Windows® 2000, Microsoft® Windows® XP, or otherwise have access to Active Directory.

If your Microsoft Navision client computers do not have access to Active Directory, they will not be able to see the Windows Users & Groups window.

Active Directory allows the administrator to give administrative permissions to other users, thereby delegating large areas of responsibility to other members of the organization. This feature makes administering Microsoft Navision more flexible. Others users, for example, department managers, can administer all the groups that they need within their department from the Microsoft Management Console.

With this tool, you can make Windows users members of specific security groups that have already been given roles within Microsoft Navision. You can control access to and permissions within Microsoft Navision, without having to open the program, provided that the Windows security groups have been given the appropriate roles within Microsoft Navision.

In an Active Directory environment, Microsoft Navision allows you to create Microsoft Navision users and roles from Windows accounts and modify the rights of these users and roles. All Active Directory security groups will be visible within Microsoft Navision and can be given roles within Microsoft Navision. For more information, see the section called "Additional Security Features Provided by Active Directory."

Logins, Roles and Permissions

The Microsoft Navision security system is company-specific and contains information about the permissions that have been granted to each individual user with access to each particular company. This includes information about what roles the users have been given as well as any particular rights that they have been granted as individual users.

To create a Microsoft Navision user, you must give him an identity within the database that allows him to log on to the system. When the user has logged on to the database, he is able to perform tasks in accordance with the permissions that he has been allocated.

Initiating the Security System

The Microsoft Navision security system is initiated when you create a database login for a superuser. The superuser then owns and administers all access to this database from within Microsoft Navision. Until you create a superuser, any user with access to the system can do anything she likes in a Microsoft Navision database.

One of the first things that the superuser should do is create user IDs for the other people who will have access to the database and assign roles to these users. Permissions are allocated at company level in Microsoft Navision.

Who Can Grant Permissions in Microsoft Navision?
Microsoft Navision allows only specific users to administer security and grant permissions. Only users who have been given the Super or Security roles are allowed to administer security. These Microsoft Navision users can only grant to other users permissions that they themselves possess.

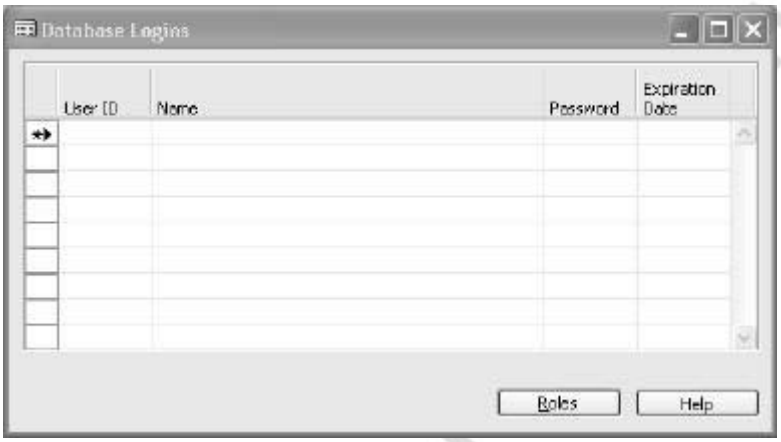
Creating Logins

Before you start to create user IDs for the people who will use your Microsoft Navision installation, you must decide whether each user must have a Windows login or a database login. The procedures for setting up the different logins are very similar.

Creating a Database Login

Creating a database login involves giving the new user a user ID and one or more roles that contain the permissions that are appropriate to their position within your organization.

To create a database login, click TOOLS→SECURITY→DATABASE LOGINS on the menu bar. The Database Logins window appears:



To initiate the security system, you must start by creating a database login for a superuser. To do this, you must set up the user ID of a superuser who has permission to perform everything that is possible in the entire program.

The following example illustrates how to set up a superuser. You can use the same procedure to create the database logins for the other users.

To create a database login for a superuser:

1. Open the Database Logins window.

2. In the **User ID** field, enter the user ID of one of the people who will administer this database, in this case SUPERUSER. You can type uppercase and lowercase letters as you like – the program will convert all letters to uppercase. You can change the user ID in this field at a later time.

***NOTE**: If using SQL Server Option, remember that this ID must be identical to the ID of the SQL Server login or Windows user or group.*

3. In the **Name** field, enter the name of the user to whom this ID belongs.

4. If you want to be able to use database server authentication, you must enter a password. It is encrypted as you type, so neither you nor anyone else can see it.

A Microsoft Navision password can contain a maximum of ten characters. It is important to remember where you have used uppercase and lowercase letters because passwords are case-sensitive. The password can be used for as long as you want. In a multiuser installation, however, where a system manager has set up all the user IDs, each user should change her own password immediately after logging on for the first time so that no one else knows her password. It cannot be seen anywhere in the system.

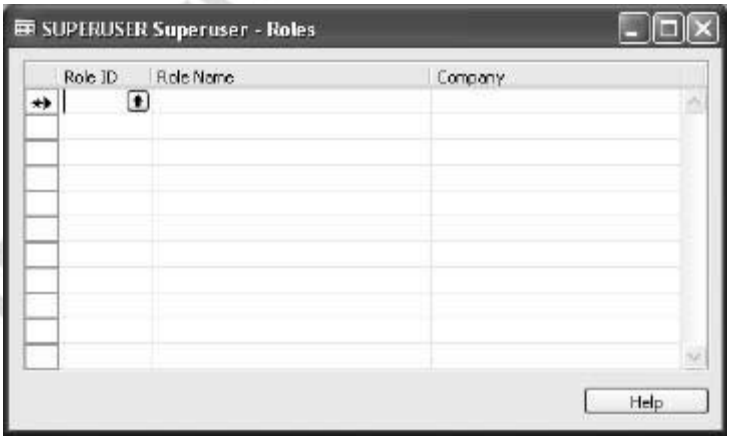
If you set up passwords for other users, remember to tell them that passwords are case-sensitive. Users do not need passwords if they are going to use the unified login supported by Windows servers.

5. In the **Expiration Date** field, you can enter a final date on which a user will be able to log on to the program. As a security precaution, superusers should not be given a date limitation.

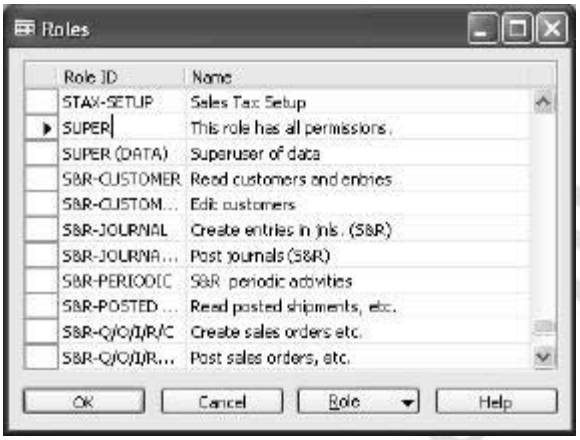
Giving the Superuser a Role

You must assign the SUPER role to your superuser before you assign roles to any other users.

1. In the Database Logins window, make sure that the user SUPERUSER is selected, and click **Roles** at the bottom of the window. The SUPERUSER – Roles window appears, listing the security roles (groups of permissions) that this user has been assigned.

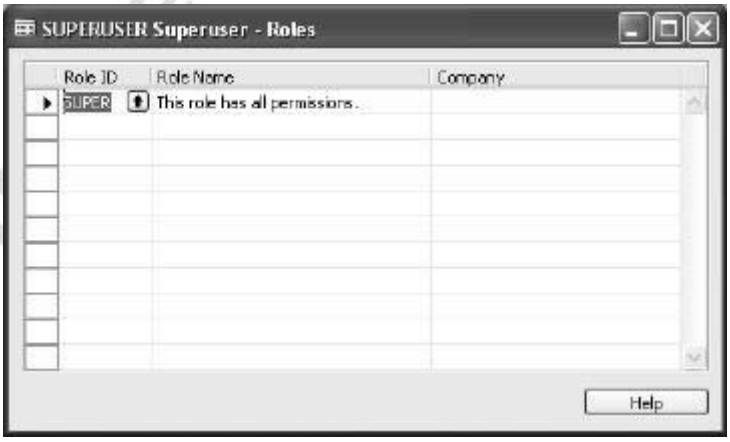


2. In the **Role ID** field, click the **AssistButton**. The Roles window appears:



3. If you have not changed them, the window will list the standard roles that come with the program. You must assign the role named SUPER to the superuser by double-clicking the line containing the SUPER role or by selecting the line and clicking **OK**.

4. This will return you to the Superuser – Roles window, and the SUPER role will have been added to the list of roles that have been assigned to this user.



NOTE: *The standard roles come with the standard database. When you create a new database, these roles will be copied to it when you restore a Microsoft Navision backup of a database containing the Data Common to All Companies and Application Objects into the new database.*

As a default, the roles apply to all the companies in the database, but they can be restricted to apply to only a particular company. To do this, enter the name of the company (or use the **AssistButton**) in the **Company** field. To have permissions apply to several (but not all) companies, you must set up one line per company (each line starting with the same role ID). If you specify that the permissions a user has only apply to a particular company in the database, the user in question will only be able to see that company.

NOTE: *At least one user must have the SUPER role for all companies.*

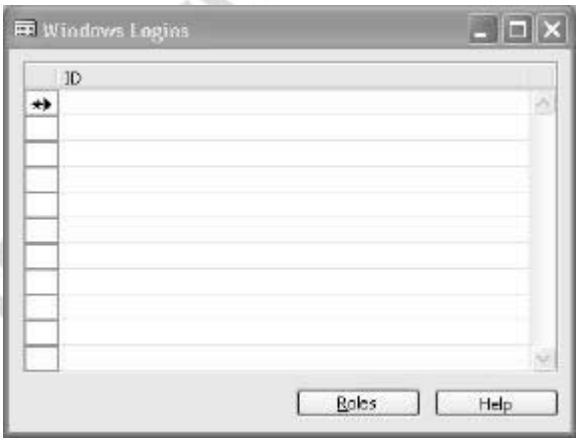
In the example above, you created a superuser. The superuser only needs to be assigned a single role because the SUPERUSER role has permissions to everything in the system. All other users, however, will need to be assigned at least two or more roles. For more information, see the section titled "The All Role."

Creating a Windows Login

It is also possible to create a Windows logins for a Windows user or group that allows them to access the system.

To create a Windows login:

1. Click TOOLS→SECURITY→WINDOWS LOGINS, and the Windows Logins window appears.



2. In the **ID** field, click the **AssistButton**, and the Windows Users & Groups window appears.

This window lists all of the Windows users and groups that are available in the current forest of domains and domain trees. This window is only available if both the domain controller is running on Windows 2000 Server or Windows Server 2003 and the clients are running on Windows 2000/XP or have been Active Directory enabled. However, if you are running a Windows NT network, you can type in the names of the Windows users and groups in the Windows Logins window. Remember to use the Domainname\Username format.

3. Select the Windows user or group for which you want to create a windows login.

4. Click **OK** and the Windows user or group will be added to the Windows Logins window.

You have created a Windows login and can now give it a role within Microsoft Navision.

You can link each user to one or more roles as needed by following the procedure for giving superusers a role described in the section titled "Giving the Superusers a Role."

There is no point in granting permissions to areas that your license file does not permit you to use. However, granting such permissions does not cause any problems. If you have the program customized or purchase additional application areas, remember to change or add to the permissions.

Giving a Database Login a Role

Sometimes logins may be set up first and then assigned roles at a later time. You can assign roles to existing users directly from the Roles window.

To give a database login a role, follow this procedure:

1. To see the users or logins have been given a particular role, click TOOLS→SECURITY→ROLES. The Roles window appears.

2. Select the role you are interested in and click ROLES→DATABASE LOGINS. The Database Logins window appears listing the database logins that have been given this role in the database.

If you have already set up the database logins and the user IDs, you can easily give a number of new users this role. To do this, you must be a superuser or have the permissions you want to give to others, as well as access to security.

3. Click the **AssistButton** in the **User ID** field. The Database Logins window appears.

4. Select the user and then click **OK**. You must add each user individually.

Giving a Windows Login a Role

To give a windows login a role, follow this procedure:

1. To see the Windows logins that have been given a particular role in Microsoft Navision, open the Roles window and select the role you are interested in.

2. Click ROLES→WINDOWS LOGINS. The Windows Logins window appears, listing the Windows logins that have been given this role in the database.

3. To give a new Windows login this role, click the **AssistButton** in the **Object ID** field. The Windows Login window appears.

4. Select the Windows login that will be assigned this role and click **OK**.

Permissions for all Other Users

Before assigning roles to all the other users, you should look at the standard roles that come with the program. You can use these roles as they are, modify them, or set up completely different ones.

IMPORTANT: *While users can modify standard roles, it is recommended to create new roles based on the standard Microsoft Navision roles since upgrades and updates could overwrite user changes. This can be done by copying permissions from an existing Microsoft Navision Role and pasting those permissions into a new Role, and then making modifications.*

Each role describes a set of access permissions to the following objects:

Object	Description
Table Data	The actual data that is stored in the tables.
Table	The tables themselves.
Form	The forms that are used to view and enter data.
Report	The reports that are used to present the data.
Dataport	The dataports that are used to import and export data.
Codeunit	The codeunits that are used in the database.
XMLport	The XMLports that are used to import and export data in XML format.
MenuSuite	The object that contains the menus that are displayed in the Navigation Pane.
System	The system tables in the database that allow the user to make backups, change license file, and so on.

The various permission types that a role can have on an object are:

Permission	Description
Read	You can read this object.
Insert	You can insert data into this object.
Modify	You can modify data in this object.
Delete	You can delete data from this object.
Execute	You can run this object.

The following options appear in the permissions fields:

Option	Comments
<blank>	Not selected (the field is empty), and you do not have this permission.
Yes	This permission is granted, and you have full access to this object. You can always, for example, read this object.
Indirect	This permission is granted indirectly. An indirect permission allows you to, for example, read the object via another object that you have permission to use, such as a codeunit or a form. Example: You have permission to run Codeunit 80, Sales-Post. The Sales-Post codeunit performs many tasks. One of these is to modify Table 39, Purchase Line. When you run the Sales-Post codeunit, Microsoft Navision checks whether or not you have permission to modify the Purchase Line table. If you do not have permission to modify the Purchase Line table, the codeunit will not be able to complete its tasks and you will receive an error message. If you have permission to modify the Purchase Line table, the codeunit will run successfully. However, you do not need to have full access to the Purchase Line table in order to run the codeunit. If you have indirect permission to modify the entries in the Purchase Line table, the Sales-Post codeunit will run successfully. When you have indirect permission, you can only modify the Purchase Line table when you run the Sales-Post codeunit or another object that has permission to modify the Purchase Line table.

The All Role

Any user who is not a superuser must be assigned the ALL role as well as roles/permissions that give access to the specific areas of the program that they will need to use. The ALL role gives you fundamental permissions to sign in, access the main menu, and move around in the program. You cannot access any tables, table data, forms and so on without further permissions, so there is no logic in assigning only the ALL role to a user.

A user who will administer security in only certain areas (such as a department manager who will administer security for their own department) must be assigned at least three roles: ALL, SECURITY, and then the role or roles that provide access to the relevant areas of the database (such as Sales & Receivables).

Creating a Microsoft Navision Role

If the existing roles do not meet the needs of your organization, you can always create new ones.

To create a new role:

1. Click TOOLS→SECURITY→ROLES. The Roles window appears.

2. Click EDIT→INSERT NEW (or use F3) to get an empty line on which to create the new role.

3. Enter an ID for the role in the **Role ID** field and a description of it in the **Name** field.

4. Press Enter or click the next line to accept the new role. Then click the new role again.

5. Click ROLE→PERMISSIONS. The Permissions window for this new role appears. Because this is a new role, the window will be empty.

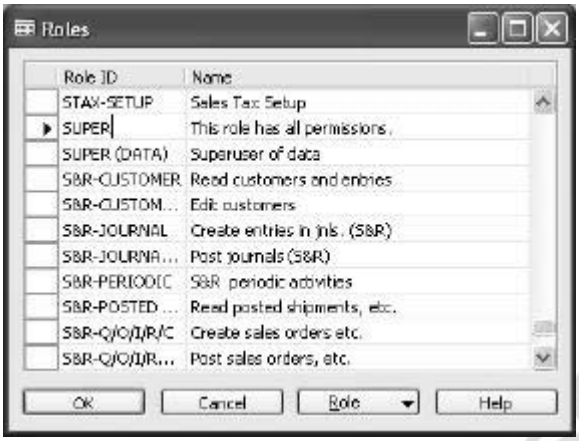
6. In the Permissions window, you can use the AssistButtons to select the type of object that you want to grant permission for and then the specific instance of that object. Enter Yes under the permissions you want to assign to the role, or use the **AssistButton** to select Yes. These permissions will then be granted to the new role.

7. Alternatively, if you want to grant the role extensive permissions, you can click **All Objects** at the bottom of the window. A list of all the objects in the application appears.

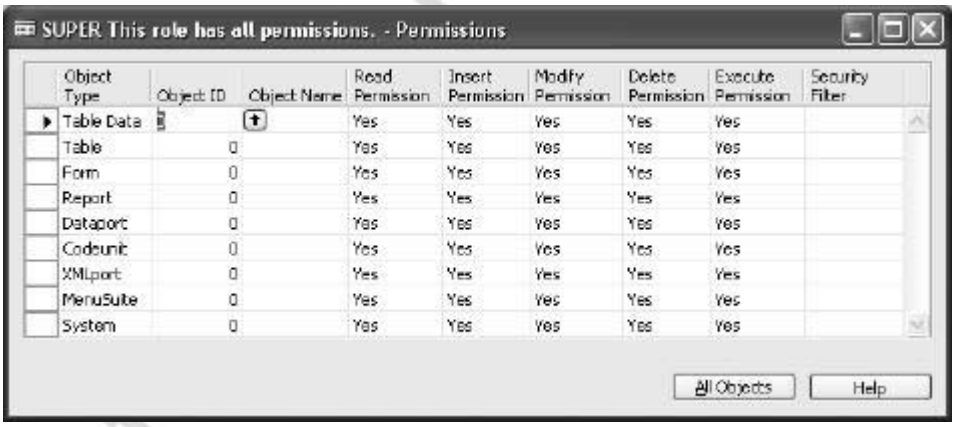
   Enter Yes under the permissions you want to give the role, or use the **AssistButton** to select Yes. When you click **OK** in the Permissions (All Objects) window, the lines marked Yes will be copied to the Permissions window for this role, where you will subsequently be able to see them.

Modifying the Permissions Granted to Roles

To view, modify, or create permissions, click TOOLS→SECURITY→ROLES. The Roles window appears. It lists the IDs and names of all the roles in the database:



For detailed information about the permissions that are included in a role, select the role (SUPER is selected in the picture), and click ROLE→PERMISSIONS.



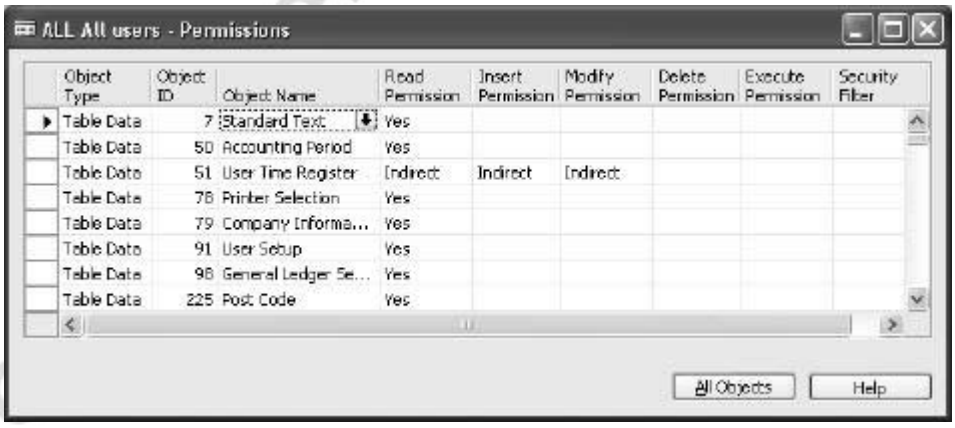
For more information, see the section called “Modifying Existing Roles” on the next page. Some special roles are described in the following table:

Role	Permissions
SUPER	This role can read, use, change, and delete all data and all application objects (that your license permits). Microsoft Navision requires that at least one user is assigned this role in each database. You cannot alter the permissions that have been granted to this role.
SUPER (DATA)	This role can read, use, change, and delete all data. This is a role that you usually assign to an accounting manager or another person who can access all the data but does not need to make changes in the program.

Role	Permissions
SECURITY	This role has access to the tables and functions related to security information (users, permissions). Users within this role can grant permissions to others, but only those permissions they themselves have. Therefore, if you want to create an "area superuser," you should give the person the SECURITY role plus permissions for the areas (such as Purchases & Payables) in which they can grant and revoke permissions for other users.
ALL	This role can use fundamental (but not "high-security") tables and functions. The permissions the user gains with this role can only be used in the tables that users must normally have access to, such as the Main Menu. Assign this role to all users (except SUPERUSER) because this is a prerequisite for all other roles you will assign to them.

Modifying Existing Roles

To modify a role, open the Roles window, and select the role. Click ROLE→PERMISSIONS. The Permissions window for this role appears. You can see which permissions are included in this role. The following window is for the role ALL (All users).



In this window, you can modify, delete or insert lines. Each line represents an object (table, report, form, dataport, codeunit, system, and so on) in Microsoft Navision. In the **Object Type** field, you can click the **AssistButton** to see (or select) the types of objects for which you can grant permissions. In the **Object ID** field, use the **AssistButton** to select the object that you want to grant permission to. The **Object Name** field is filled in automatically when you select an object ID.

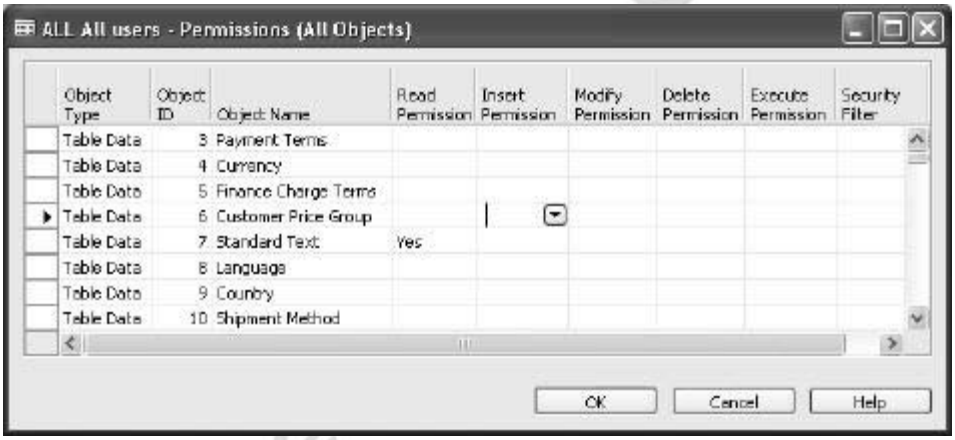
IMPORTANT: *When you modify a role for one user, it is also altered for all the other users who have been assigned this role in the current database.*

You can also modify these permissions by applying security filters. For more information about security filters, see the section "Applying Security Filters."

Selecting Additional Objects

You can also grant permissions for multiple objects. This is a useful time saving feature if you want to radically modify the permissions that apply to a role.

To see all the objects for which you can grant permissions, click **All Objects**. The Permissions (All Objects) window appears:



You can use the scroll bar on the right to scroll up and down in the window. Enter Yes in the fields for those permissions you want to add. (When you return to the Permissions window, it will include the ones for which you entered Yes.) Click **OK** to save your selection.

You can also apply security filters in this window. For more information about security filters, see the next section.

Applying Security Filters – SQL Server Option Only

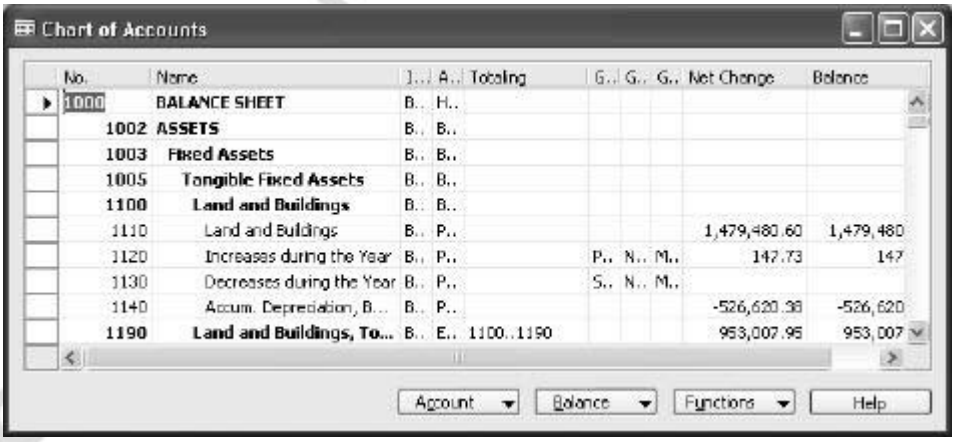
The SQL Server Option for Microsoft Navision supports record level security and allows you to tailor the security system to meet the needs of your organization. You may, for example, want some of your employees to be only able to read, edit, and enter information in the accounts of a particular customer or of a particular department. This is done by applying security filters that limit the access that your users have to the records stored in specific tables in the database.

Security filters can only be applied to tables and the records that they contain. In the following example, security filters are applied that limit the access that a user has to the entries in the database. The filters that are applied are based on Department and Project (Global Dimensions 1 and 2). After the filters have been applied, the user will only be able to see the accounts and entries that have to do with the Sales department and the Toyota project.

The user has been assigned the G/L-ACCOUNT and the ALL roles in the current database. These are two of the standard roles that are provided with the application. The G/L-ACCOUNT role allows the user to read all the accounts and entries in the General Ledger Chart of Accounts by granting them read access to the G/L Account table (15) and the G/L Entry table (17). The user's permissions have not been limited to any particular company.

However, you can specify that this user can only read specific accounts and entries by applying one or more security filters to these tables. The user will only be able to see the accounts and entries that comply with these filters. If you set a security filter that specifies that the user who has been given this role is only able to see the records in the G/L Account table that relate to a particular department (Global Dimension 1), they will not be able to access any of the other records that are stored in this table.

The Chart of Accounts window displays the information that is stored in the G/L Account table. Before applying the filter the Chart of Accounts window looks like this:



To apply the security filter:

1. Click TOOLS→SECURITY→ROLES, and the Roles window appears.

2. Select the G/L-ACCOUNT role and click ROLE→PERMISSIONS. The Permissions window for that role appears.

3. Select the G/L Account table.

4. In the **Security Filter** field, click the **AssistButton** to open the Table Filter window.

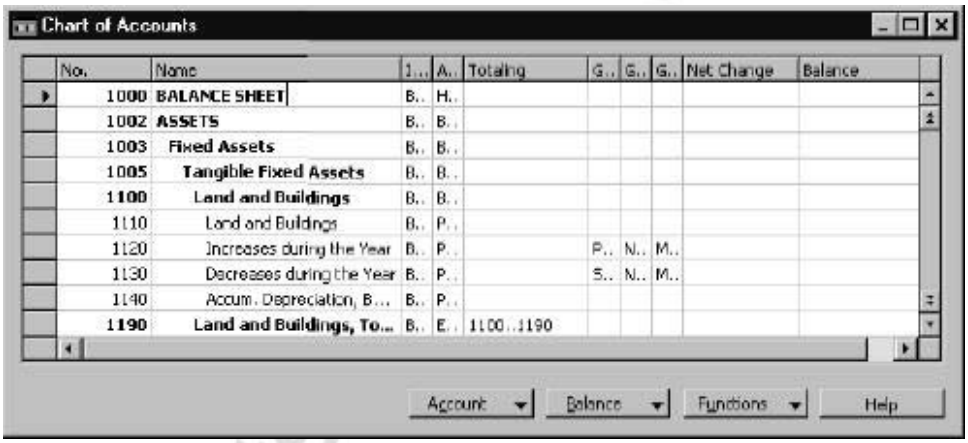
5. In the **Field** field, click the **AssistButton**, and select Global Dimension 1 Filter in the list that appears.

6. In the **Filter** field, enter Sales.

7. Click **OK** to apply the filter.

You have now applied the security filter to the G/L-ACCOUNT role.

When a user who has the G/L-ACCOUNT role opens the Chart of Accounts window, there is less information displayed in the **Net Change** and **Balance** fields:



The G/L ledger entries are stored in a separate table. Therefore, you must remember to apply the same filter to the G/L Entry table. You must apply the filter to both tables in order to ensure that the user does not gain access to any entries that are not specified in the filter that has been applied to the G/L Account table. This will ensure that the user can only see the accounts and the entries that comply with the security filter. Security permissions and filters are table specific.

When you are applying the security filter to the G/L Entry table, you must select Department Code in the **Field** field.

To apply the same filter to the G/L Entry table:

1. Select the G/L-ACCOUNT role and click ROLE→PERMISSIONS. The Permissions window for that role appears.

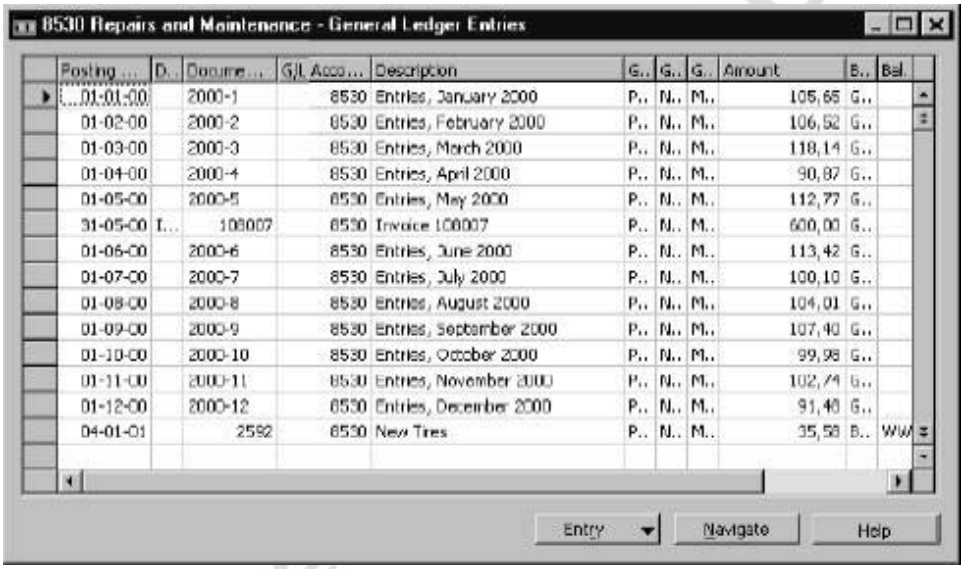
2. Select the G/L Entry table.

3. In the **Security Filter** field, click the **AssistButton** to open the Table Filter window.

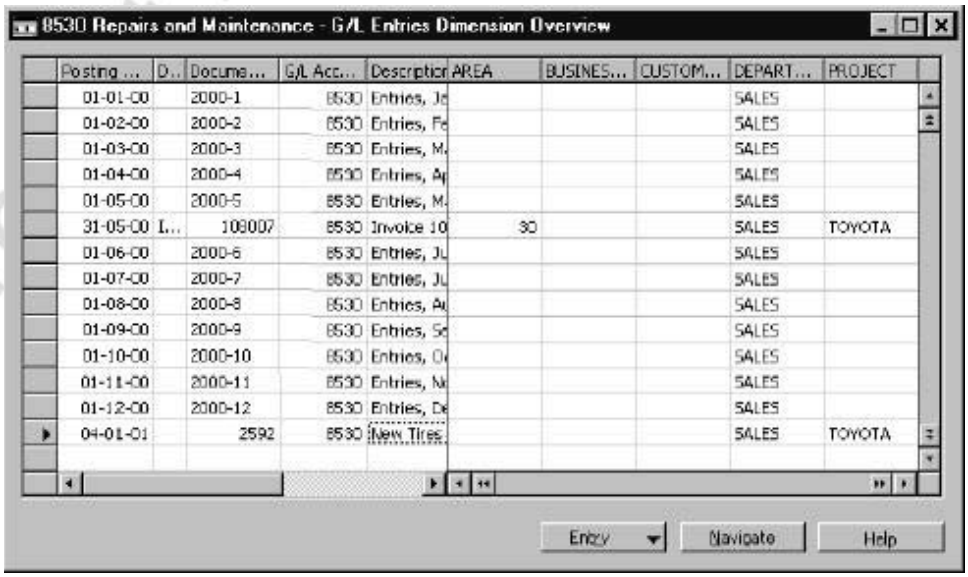
4. Click the **AssistButton** in the **Field** field and select Global Dimension 1 Code.

5. In the **Filter** field, enter Sales.
6. Click **OK** to apply the filter.

You have now applied a security filter to the G/L-ACCOUNT role. After you have applied this security filter, the users who have this role can only see the G/L accounts and ledger entries that relate to the Sales dept. (Global Dimension 1). G/L Account 8530 contains 40 ledger entries that relate to all the departments. After you have applied the security filter, the users who have the G/L-ACCOUNT role can only see 14 of these entries:



To see the dimensions that these entries relate to, click ENTRY→G/L DIMENSION OVERVIEW, and the G/L Entries Dimension Overview window for account 8530 appears:



These changes will only take affect the next time those users who have been assigned this role in the database log on. If any are currently logged on, their permissions are not affected by this security filter.

You can refine this role even further by applying another security filter that allows the users who have been given the G/L-ACCOUNT role to see only the G/L accounts and entries that relate to a specific project (Global Dimension 2).

To apply this security filter:

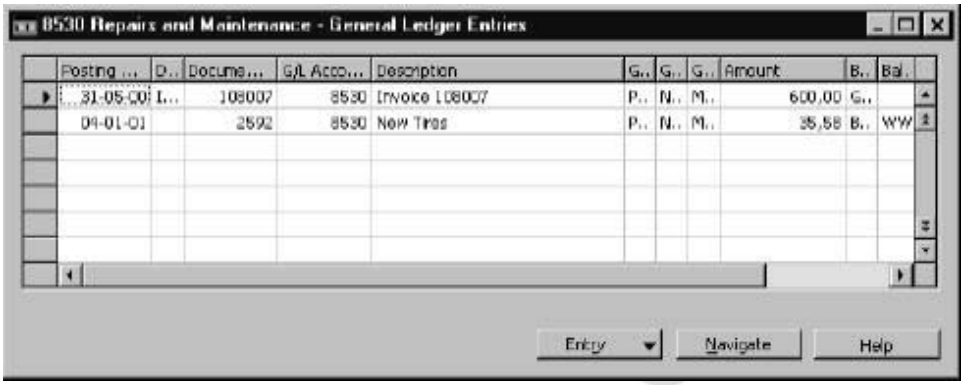
1. Click TOOLS→SECURITY→ROLES and the Roles window appears.
2. Select the G/L-ACCOUNT role and click ROLE→PERMISSIONS. The Permissions window for that role appears.
3. Select the G/L Account table.
4. In the **Security Filter** field, click the **AssistButton** to open the Table Filter window.
5. In the **Field** field, click the **AssistButton**, and select Global Dimension 2 Filter in the list that appears.
6. In the **Filter** field, enter Toyota.
7. Click **OK** to apply the filter.

To apply the same filter to the G/L Entry table:

1. Select the G/L-ACCOUNT role, and click ROLE→PERMISSIONS. The Permissions window for that role appears.
2. Select the G/L Entry table.
3. In the **Security Filter** field, click the **AssistButton** to open the Table Filter window.
4. Click the **AssistButton** in the **Field** field and select Global Dimension 2 Code.
5. In the **Filter** field, enter Toyota.
6. Click **OK** to apply the filter.

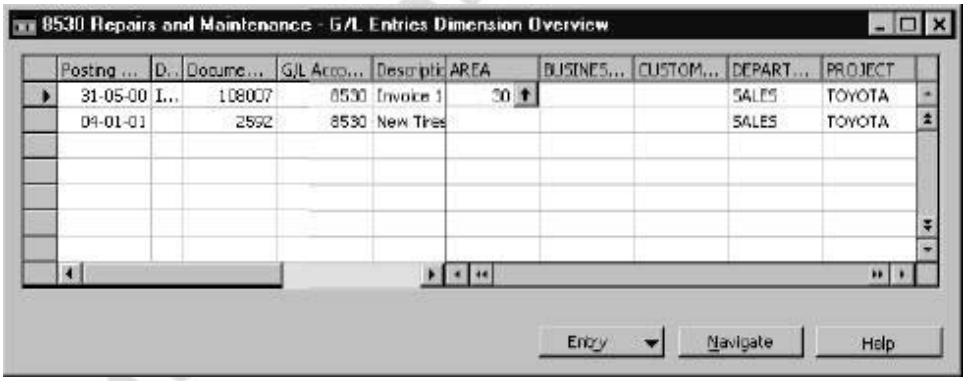
You have now applied another security filter to the G/L-ACCOUNT role.

After you have applied this security filter, the ledger entries for G/L account 8530 look like this:



Now the users who have been given the G/L-ACCOUNT role can only see 2 entries.

To see the dimensions that these entries relate to, click ENTRY→G/L DIMENSION OVERVIEW and the G/L Entries Dimension Overview window for account 8530 appears:



You have now applied a more refined security filter to the G/L-ACCOUNT role. Any user who has been given this role will only be able to see information in the G/L accounts that conforms to the security filters. Furthermore, they will only be able to read the G/L entries of the Sales department that relate to the project called Toyota.

IMPORTANT: *When you apply a security filter to a role, you modify that role. This means that all the other users who have been assigned that role will also have their permissions changed*

If you do not want these modified permissions to apply to all the other users who have been assigned this role, you should consider creating new security roles before changing any of the standard security roles. You might also want to create new roles that contain security filters, which reflect the security needs of your company. Each department may need its own set of security roles each with their own individual security filters.

Joining Security Filters

Setting up one security filter will not in itself ensure that the user can only see those records that are specified in the filter. Each user will generally have more than one role in the current database and will receive permissions from each of these roles. The permissions that the user possesses are the sum of all the permissions specified for all the roles that the user has been assigned.

If more than one role gives the user permissions to access data from the same table, the security filter specified for this table in one role will have no effect if another role gives the user permissions to perform the same operations on the same table but without any security filter. Not applying a security filter means that the user can, for example, read all the entries in that table.

In the same way, if the user has two roles that give permission to the same table and both roles have security filters applied to them, it is the sum of these filters that is applied. This means that if one filter specifies that the user should only be able to read entries 1 – 10 and the other filter specifies entries 5 – 20, the user will be able to read entries 1 – 20.

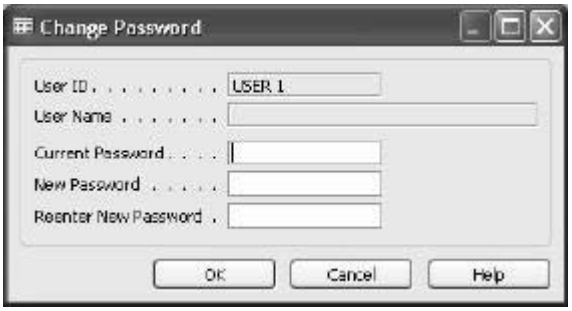
For more detailed information about record level security as well as an example of how to implement security filters in a posting scenario, see the manual Application Designer's Guide.

Changing Passwords

If you have been given your user ID and password by the system manager, it is a good idea to change your password the first time you use the program. That will ensure that you are the only one who knows your password.

NOTE: *Passwords in Microsoft Navision do not have time limits. If you want to set time limits for access, you can place an expiration date on the user ID or you can specify an allowed posting period for each user in User Setup.*

To change your program password, click TOOLS→SECURITY→PASSWORD and the Change Password window appears:



The **User ID** and **User Name**, which appear in the two uppermost fields, are not editable fields.

Enter your current password in the **Current Password** field (remember to distinguish between uppercase and lowercase letters). Enter your new password in the **New Password** field. You must then confirm the new password by entering it again in the **Reenter New Password** field. This verifies that you entered it correctly the first time and that you can remember it. The password will not appear when you enter it, and you cannot see it anywhere else in the program.

If a message appears informing that the password is incorrect, there are two possible causes:

- You typed it wrong the second time – try again.
- The password that you created is not what you think it is. Perhaps you made a typing error the first time, or maybe you used uppercase and lowercase letters differently in the two fields. Enter the password in the **New Password** field again, and then reenter it again.

Click **OK** if you want to change the password; click **Cancel** to stop the password from being changed.

If You Forget Your Password or User ID

This information is primarily of interest to end users. However, you may get a request for new passwords, in which case, you will need to know this. If your customer has forgotten her password or user ID, she is allowed an unlimited number of attempts to enter it. She can also request a new password from a user who has permission to change other people's passwords.

If all users have forgotten their user IDs or passwords (passwords are encrypted, so they cannot be seen anywhere in the program) so no one can set up new ones, they can get a special password from the Microsoft Navision Solution Center (NSC).

The customer should follow this procedure:

1. Open Microsoft Navision. The Login window appears.

2. In the **User ID** field, enter 20 question marks (?????????…). The program now displays a window containing a code.

3. Give the code to the NSC, and they will supply you with a new password.

***WARNING**: Do not use the keyboard or mouse before you enter the new password.*

4. Enter the password from the NSC. It can be used only once.

5. Click **OK** to close the window.

6. Click TOOLS→SECURITY→DATABASE LOGINS, and set up your own password.

User Specific Setups

User Time Limits and Time Registers

Microsoft Navision allows you to put time limits on user IDs. If you are using Windows authentication, you can also put a time limit on passwords within the Windows domain. If you have the required permissions, you can always delete a user's login from the system or cancel all their permissions. Alternatively, you can specify a limited time period during which a particular user ID is only allowed to post in the program.

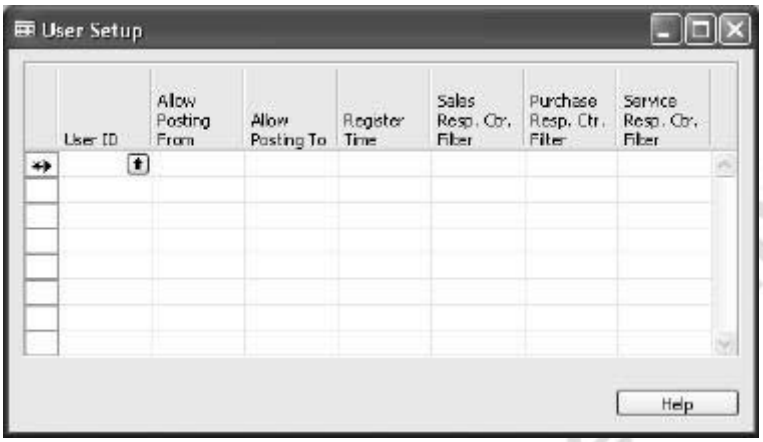
After you have created a user ID, you can specify that the user with that ID can post only during certain time periods (for example, June 1 – June 15) and that the program must keep track of the amount of time the user has been working in each company.

This can be used, for example, by accountants who post for others, to document the amount of time spent working on the accounts of the various companies. The User Setup and User Time Register windows are used for this.

User Setup

In the User Setup window, you define when each user will be allowed to post and whether the program will record the amount of time that each were logged on. You can also assign responsibility centers to the user.

To open the User Setup window, click ADMINISTRATION→APPLICATION SETUP→USERS→USER SETUP.



Fill in the fields in the User Setup window according to these guidelines:

Field	Comments
User ID	Enter the user ID for which you want to set up conditions. The user must have been set up already. If you cannot remember the user ID, click the **AssistButton** to see a list of the user IDs that have been set up in the current database.
Allow Posting From	Enter the date on which the user will be allowed to start posting.
Allow Posting To	Enter the last date on which the user will be allowed to post.
Register Time	If you want to register the amount of time a user works on the company, enter a check mark by clicking the field or pressing the spacebar.
Sales Responsibility Center Filter	Enter the code for the responsibility center to which you want to assign the user. Click the **AssistButton** to see the responsibility centers that have been created. This responsibility center will be the default responsibility center when the user creates new sales documents. The user will only see sales orders that are created from his responsibility center. If you leave this field blank, the default responsibility center in Customer or Company Information (in order of priority) will be used.

Field	Comments
Purchase Responsibility Center Filter	Enter the code for the responsibility center to which you want to assign the user. Click the **AssistButton** to see the responsibility centers that have been created. This responsibility center will be the default responsibility center when the user creates new purchase documents. The user will only see purchase orders that are created from his responsibility center. If you leave this field blank, the default responsibility center in Customer or Company Information (in order of priority) will be used.
Service Responsibility Center Filter	Enter the code for the responsibility center to which you want to assign the user. Click the **AssistButton** to see the responsibility centers that have been created. This responsibility center will be the default responsibility center when the user creates new service documents. The user will only see service orders that are created from his responsibility center. If you leave this field blank, the default responsibility center in Customer or Company Information (in order of priority) will be used.

For more information about responsibility centers, see the online Help.

NOTE: *Other fields for posting periods are found by clicking* FINANCE→SETUP→GENERAL LEDGER SETUP***,*** *but those periods refer to the entire company and thus apply to all users. Anything that you enter for a particular user under User Setup will take precedence over the general choices you made under* FINANCE→SETUP→GENERAL LEDGER SETUP *for that user.*

User Time Register

If the **Register Time** field in the User Setup window contains a check mark, the User Time Registers window will contain information about when and how long individual users have been logged on to the company. Click ADMINISTRATION→IT ADMINISTRATION→USERS→TIME REGISTERS.

The User Time Registers window appears:

This window displays the time use registered for a number of users. The lines are generated automatically, but you can also enter information in them.

Time use is registered in whole minutes, rounded to the nearest minute. The program creates one line per user, per day. If the same user uses the company more than once on a day, the line displays the total time used on that day.

If a user finishes using the company after midnight, the time use will be registered to the date when work began – not the date it was completed.

Selecting a Printer

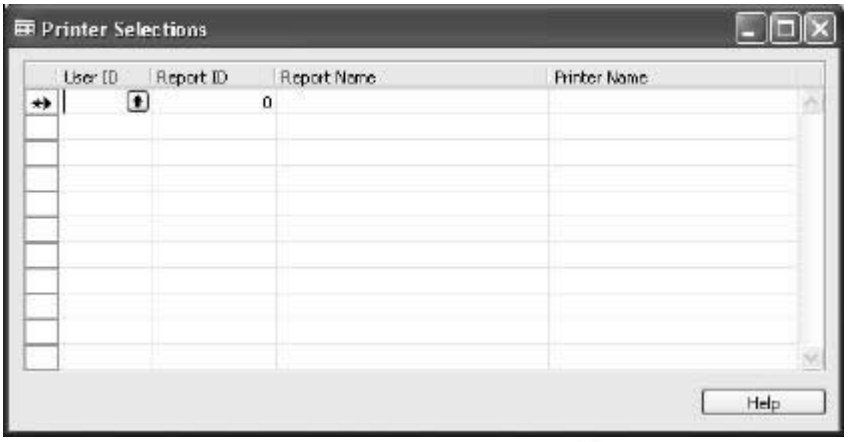
When you want to print from Microsoft Navision, you can use the printers you have installed in Windows. The printer that has been designated as the default printer in Windows will be used as a default in Microsoft Navision.

Individual Printer Selection

If you want a particular user to always use a specific printer, or if you want a particular report always to be printed on the same printer, you can set these options as fixed printer selections. A fixed printer selection will apply no matter what printer selections or other changes have been made in the program. The fixed printer selection does not determine options such as paper format.

To choose fixed printers, click ADMINISTRATION→IT ADMINISTRATION→GENERAL SETUP→PRINTER SELECTIONS.

The Printer Selections window appears:



Fill in the fields in the Printer Selections window according to these guidelines:

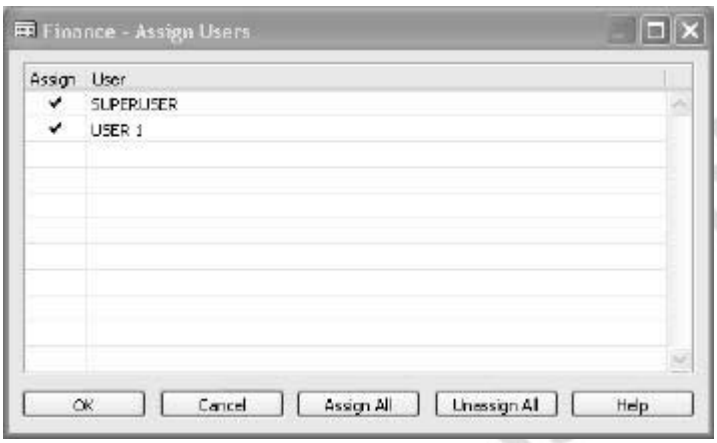
Field	Comments
User ID	Enter the ID of the user for whom the printer selection applies. The user ID must have been created in the User table already. To see a list of all the users, click the **AssistButton**. If you would like the selection to apply to all users, leave the field blank.
Report ID	Enter the ID of the report to which the printer selection applies. To see a list of all the reports, click the **AssistButton**. If you would like the selection to apply to all users, leave the field blank.
Report Name	This field is filled in automatically when you enter the report ID.
Printer Name	Enter the name of the printer that will be used for both the user ID and the report specified on this line. To see a list of all the available printers, click the **AssistButton**. The Printers window appears, and you can select the printer you want from there.

Assigning Menus to Users

The menu suite administrator can use the Navigation Pane Designer to assign menus to users. A whole menu can be assigned to a user – not a menu group or a menu item.

To Assign Menus to Users:

1. Right-click the menu button of the menu that you want to assign to one or more users, and then click **Assign Users....**

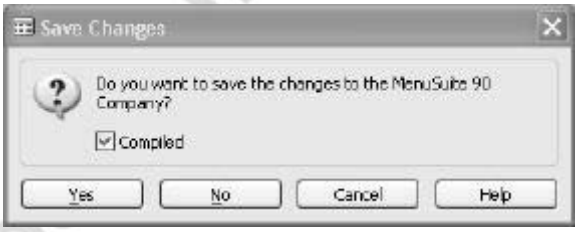


2. In the Assign Users window in the **Assign** field, place a check mark next to the name of each user that you want to assign the menu to. To assign a menu to all users, click **Assign All**.

3. Click **OK**.

4. Repeat the procedure for each of the menus that you want to apply to one or more users.

5. When finished, press **Escape** and a message will appear asking to save changes. Click **Yes** to save the changes:



If the menu is disabled, you must enable it to make it available to the users. The users will not be able to see it until they refresh the Navigation Pane. To refresh the Navigation Pane, right-click below the menu buttons in the space next to the scroll arrows, and select **Refresh**.

Additional Security Features Provided by Active Directory

If your network and clients are using Active Directory or are Active Directory enabled, you have access to extra security features. You are able to give Windows users and groups roles within Microsoft Navision. You can also make Microsoft Navision roles members of Windows security groups. However, the individual permissions that are granted to the roles can only be administered from within Microsoft Navision.

Giving Windows Users or Groups a Microsoft Navision Role

Active Directory allows you to give Windows users and groups a Microsoft Navision role.

1. On the menu bar click TOOLS→SECURITY→ROLES to open the Roles window.

2. Select the Microsoft Navision role you want to assign, and click ROLE→WINDOWS LOGINS, and the Windows Logins window for this role appears. This window lists all the Windows users and groups that have already been given this role in Microsoft Navision.

3. To add a Windows user or group to the list, select an empty row or create a row by clicking EDIT→INSERT NEW (or use F3).

4. Click the **AssistButton** in the **Login ID** field. The Windows Logins window appears. This window contains a list of all the Windows users and groups that can log on to Microsoft Navision.

5. Select the user or group to whom you want to give this Microsoft Navision role and click **OK**. This user or group will now be added to the list shown in the Windows Logins window for this role.

Giving a Windows User or Group a Microsoft Navision Login

It is also possible to add a Windows user or group to the list of Windows logins that can access the system.

1. On the menu bar, click TOOLS→SECURITY→WINDOWS LOGINS. This opens the Windows Logins window.

2. Click the **AssistButton** and the Windows Users & Groups window appears listing all the Windows users and groups that are available in the current forrest of domains and domain trees.

3. In the Windows Users & Groups window, select the Windows user or group for which you want to create a Windows login. Double click it or click **OK**, and it will be added to the Windows Login table.

The Windows Users & Groups window is only available if Active Directory is enabled in the domain and on the clients. However, if you are running a network without Active Directory, you can still type in the names of the Windows users and groups in the Windows Logins window. In that case, remember to use the Domainname\Username format, for example myserver\myID.

Adding Microsoft Navision Roles to a Windows Security Group

Active Directory also allows you to make Microsoft Navision logins and roles members of Windows security groups.

1. On the menu bar, click TOOLS→SECURITY→WINDOWS LOGINS to open the Windows Logins window. This window lists all of the Windows users and groups that can currently access the system.

2. Select the Windows login to which you want to add a Microsoft Navision role.

3. Click **Roles**, and select the relevant role from the Roles window that appears. This role and all the individual users that have been given this role will now be added to the Windows login that you selected earlier.

Test Your Knowledge – Microsoft Navision Security

True	False		
❑	❑	1.	A minimum level of security requires that you can limit the user's access so that they can only access certain types of information stored in the database.
❑	❑	2.	Record level security is available using the SQL Server Option for Microsoft Navision.
❑	❑	3.	In an Active Directory environment, Microsoft Navision allows you to create users and roles from Windows accounts and modify the rights of these users and roles.
❑	❑	4.	The Microsoft Navision Security system is initiated when you create a database login for a user.
❑	❑	5.	At least one user must have the SUPER role for all companies.
❑	❑	6.	The ALL role must be assigned to all users, even the Superuser.
❑	❑	7.	Security Filters are available for both Microsoft Navision database and SQL Servers.
❑	❑	8.	Passwords in Microsoft Navision have time limits.
❑	❑	9.	Microsoft Navision allows you to put time limits on user ID's.
❑	❑	10.	In Microsoft Navision, Printers and Menus can be user defined.

Test Your Skills – Create Database Logins and Assign Roles

Scenario: As the IT administrator, you need to create database logins and passwords for three new employees and assign the appropriate roles.

All Levels

Your tasks are as follows:

- Create a database login of JR for a superuser called John Roberts. Use the password of Cronus and assign him the role SUPER.
- Create a database login of AH for Alice Hart. Use the password of Alice and grant security permissions for the Purchases & Payables area.
- Create a database login of RQ for Richard Quist and use the password of Richard. Richard will be a superuser of data.

Need a Little Help?

- Go to TOOLS→SECURITY→DATABASE LOGINS
- Create the User ID's and Passwords
- Apply the appropriate Roles

Test Your Skills – Create a new Role

Scenario: Your company hires a consultant who will be working in-house for several months doing telemarketing for your Contacts. The consultant will need access to read the Contact tables and enter (insert, modify and delete) comments on the Cards. Set up a new role that can be used to allow the consultant the appropriate access to information in your system.

All Levels

Your tasks are as follows:

- Create a new Role called Madeleine
- Add read permissions for Contacts as well as insert, modify, and delete permissions to Contact Comments.

Need a Little Help?

- Go to TOOLS→SECURITY→ROLES
- Create a new Role
- Select the following role: RM-CONT
- Click on ROLES→PERMISSIONS
- Copy all Permissions
- Select the new Role created above
- Click on ROLES→PERMISSIONS
- Paste in the permissions
- On the Rlshp. Mgt. Comment Line table, give the appropriate access.

Test Your Skills – Change a Password

Scenario: You are Alice, and you want to change your password.

All Levels

Your tasks are as follows:

- As Alice, change the password from Alice to AliceNew
- Practice logging in with the new password.

Need a Little Help?

- Log in as Alice.
- Go to TOOLS→SECURITY→PASSWORD.
- Change the password.
- Log out and back in with the new password.

Test Your Skills – Set up Time Limits for a User

Scenario: Richard Quist is a seasonal sub-contractor. As the IT Administrator, John Roberts, you must set up a time limit for him so that he can access Microsoft Navision from November 1, 2000 to January 2, 2001. The finance department would also like his time registered and has asked that you set that up as well.

All Levels

Your tasks are as follows:

- As John Roberts, set the Allow Posting from and Allow Posting to for Richard Quist from November 1, 2000 to January 2, 2001.
- Set up the time register.

Need a Little Help

- Log in as John Roberts.
- Go to ADMINISTRATION→APPLICATION SETUP→USERS→USER SETUP.
- Set the Time Limits.
- Set the Time Register.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 9: BASIC CUSTOMIZATIONS

Overview

Microsoft® Business Solutions–Navision® has aligned its main menu with Microsoft® Office 2003 provided through the Office 2003 Navigation Pane Style experience. This new style allows users to have a main menu that has their main activities as a basis. And with the new structure, users who maximize forms will now maintain their large workspace and still have direct access to the main menu.

This new approach to the main menu, considered to be a User Layer, allows database administrators to create a main menu that they can configure and personalize themselves, without needing their Microsoft Solution Provider to design new menus. There is also a personalization to the structure of the main menu now available to end users that allows for hiding/showing, moving, and creating shortcuts to menus.

This chapter introduces the new main menu structure, cycle/toggle and the tree level structure. The steps that a database administrator will take to create new menus as well as modifying existing menus will be described and the available end-user personalization will be covered. Finally, this chapter describes the MenuSuite object, which contains the menu suite that is displayed in the Navigation Pane and in the Navigation Pane Designer.

Menu Suite Fundamentals

The MenuSuite object contains the menu content that is displayed in the Navigation Pane and in the Navigation Pane Designer. A menu suite is a set of menus. Each menu contains content for a specific departmental area, for example, Finance or Manufacturing.

The following screenshot shows an example of menu suite content displayed in the Navigation Pane at runtime and in the Navigation Pane Designer at design time:

Navigation Pane

Navigation Pane Designer

In the Navigation Pane at runtime, there is a Shortcuts menu. Here end-users can add shortcuts to forms, reports and batch jobs in Microsoft Navision. To visually indicate the difference between the Navigation Pane and the Navigation Pane Designer, a different color scheme is used for each. Also, in the header section of the Navigation Pane Designer, the MenuSuite object level that you are working on is shown before the menu name.

The Navigation Pane

Navigation Pane

The Navigation Pane gives you access to the business application areas in Microsoft Navision. You use the Navigation Pane to open windows in the program.

If you click a menu item when the related window is already open, the program will either open a new copy of the window or bring the already open window into focus, depending on the options that you have chosen.

This menu item is accessible by clicking VIEW→NAVIGATION PANE.

Navigation Pane Options

You use the Navigation Pane Options window to control how the program behaves when you click a group or item in the Navigation Pane. You can place a check mark next to the options that you want to activate.

To Set the Navigation Pane Options:

1. Right-click below the menu buttons, next to the scroll arrows, and then click **Navigation Pane Options**. The Navigation Pane Options window appears:

2. Place a check mark next to the options that you want to activate.

Option	Description
Single-click groups	A check mark next to this option means that menu groups will open and close on one click. Otherwise, you must double-click to open and close groups.
Show group lines	A check mark next to this option means that there will be a line from each item to the group to which it belongs.
Track active windows	A check mark next to this option means when you move between windows in the program, the corresponding menu item will be selected. However, this only applies to windows that were opened from the menu.
Swap open and new actions	A check mark next to this option means when you left-click an item, you will open a new instance of the window.

Opening a Menu Item from the Navigation Pane

When you left-click a menu item for a report or batch job, the program will always open a new window.

When you left-click a menu item for a window that represents a form (for example, a card), if there is already an open window, the program will bring the open window into focus, rather than opening a new window. However, if the **Swap open and new actions** field is checked in the Navigation Pane Options, a new form will open each time.

You can right-click the item and then select:

Option	Descriptions
Open in the New Window	Each time you left-click the menu item, a new window will open.
Cycle Window	Each time you left-click the menu item, the program will switch focus to one of the open windows for the menu item. If you click repeatedly, you will cycle through all of the open windows for that menu item.
Toggle Window	Each time you left-click the menu item, the program will switch focus to one of the last two windows that you opened for the menu item. If you click repeatedly, you will switch between the last two windows that you opened for the menu item.

Keyboard Shortcuts for the Navigation Pane

Keystroke	Mouse Usage	Action
F12	Click menu area or **Main Menu** button in the Toolbar	Give focus to the Navigation Pane
CTRL + F12	Click the active window	Give focus to the active window
ALT + F12	Menu: Tools / Navigation Pane Designer	Switch to the Navigation Pane Designer
ALT + F1	Menu: View / Navigation Pane	Show / Hide the Navigation Pane
SHIFT + ALT + UP	Navigation Pane Scrollbar	Scroll Menu Buttons down
SHIFT + ALT + DOWN	Navigation Pane Scrollbar	Scroll Menu Buttons up
CTRL + ALT + UP		Activate previous menu
CTRL + ALT + DOWN		Activate next menu
CTRL + 0		Activate Shortcuts menu

Keystroke	Mouse Usage	Action
CTRL + 0 … CTRL + 9		Activate menu at position 1 … 9
	Left-double-click Navigation Pane resize border	Toggle current and minimum Navigation Pane sizes

Keyboard Shortcuts for the Navigation Pane Designer

Keystroke	Mouse Usage	Action
HOME		Select first item
END		Select last item
UP		Select previous item
DOWN		Select next item
PGUP	Menu scrollbar	Move selection up a page
PGDN	Menu scrollbar	Move selection down a page
LEFT	Click +/- button	Collapse group
RIGHT	Click +/- button	Expand group
CTRL + HOME		Scroll to first item
CTRL + UP	Menu scrollbar	Scroll menu down
CTRL + DOWN	Menu scrollbar	Scroll menu up
CTRL + PGUP	Menu scrollbar	Scroll selection up a page
CTRL + PGDN	Menu scrollbar	Scroll selection down a page
CTRL + SHIFT + UP		Move item up
CTRL + SHIFT + DOWN		Move item down
ENTER / SPACE	Left-click	Cycle open window*
CTRL + ENTER / SPACE	CTRL +Left-click	Open item in new window*
SHIFT + ENTER / SPACE	SHIFT +Left-click	Toggle open window
ALT + ENTER		Properties
CONTEXT MENU	Right-click	Show context menu
CTRL + X		Cut item
CTRL + C		Copy item

Keystroke	Mouse Usage	Action
CTRL + V		Paste item
F2	Left-click selected item label	Rename item
DELETE		Delete item
INSERT		Create item
SHIFT + INSERT		Insert Items
CTRL + INSERT		Create group
CTRL + S	File / Save	Save the current menu suite and remain in the Navigation Pane Designer
F11	Tools / Compile	Compile the current menu suite
ESC	File / Close	Close the Navigation Pane Designer, prompting to save changes

***NOTE**: Not all actions for the keystrokes are always available since their use depends on whether you are working with the Navigation Pane or the Navigation Pane Designer and if the Shortcuts menu is open.*

You can swap the default and CTRL actions by changing the Navigation Pane Options.

The Navigation Pane Designer

The Navigation Pane Designer allows editing access to the company's whole menu suite. Existing menus can be edited, new menus created, the order of the menu buttons can be changed, or menu buttons can be deleted in the Navigation Pane. One or more menus can then be assigned to each user.

In order to work with the Navigation Pane Designer, a user must be set up as an administrator for the company's MenuSuite objects. He must have basic design permissions for the MenuSuite object and all permissions for their company's MenuSuite objects.

To access the Navigation Pane Designer, click TOOLS→NAVIGATION PANE DESIGNER.

Creating Menus with the Navigation Pane Designer

There are two ways to create a new menu:

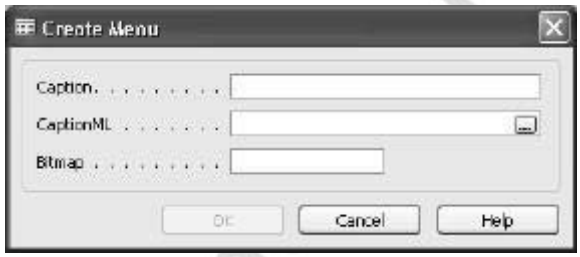
- Create an empty menu that you fill in by cutting and pasting parts of another menu and by creating menu groups and inserting items, or
- Copy a menu and then modify it.

In addition, you can edit the menus that you receive from your solution provider.

***NOTE**: When upgraded menu suites are received, menus that have been created (including a menu based on a copy of an original menu) will not be upgraded automatically at all, while the original menus will be upgraded to some extent, even if they have been modified .See the section titled, "How Upgrades Affect New Menus."*

To Create an Empty Menu:

1. Right-click any menu button, and then click **Create Menu**. The Create Menu window appears:



2. The cursor appears in an edit box. Complete the fields based on the descriptions in the table below, and then click **OK**. The new menu button is inserted below all of the other buttons, and the blank new menu is displayed.

Field	Descriptions
Caption	The name of the new menu
CaptionML	If you are designing a menu that will be used in different languages by different users, you must give the menu a name in each language. Once you enter a name in the **Caption** field, the **Caption ML** field contains the code for the language you use on your computer and the name of the menu. When you click the **AssistButton** in the field, a window opens where you can enter a menu name for each language code.
Bitmap	The **Bitmap** field contains the number of the bitmap that appears before the name on the menu button. The bitmap number is an integer between 0 and 14.

To Copy an Existing Menu:

1. Right-click the menu button of the menu that you want to copy, and then click **Copy**.
2. Right-click again, and then click **Paste**. A new menu button will appear, with the same name as the original, and the new menu is displayed.

Deleting Menus from the Navigation Pane Designer

Menus can be deleted from the menu suite, whether it is a menu that was created by a Menu Suite Administrator or a solution provider.

***ATTENTION**: There is no undo function once you have completed the deletion.*

Since there is no undo function, menus can be hidden rather than deleted. Hiding menus also gives extra space in the menu pane area of the Navigation Pane. The menu is only hidden in design view. Any users that have been assigned to the menu will still be able to see it in their own menu suite.

Even though a menu is deleted, all of the items in the menu will still be available in the list of menu items when **Insert Items** is selected in the Navigation Pane Designer.

To Delete Menus from the Navigation Pane Designer:

1. Click TOOLS→NAVIGATION PANE DESIGNER.
2. Right-click the menu button for the menu to delete, and then click **Delete**.
3. Click **Yes** to confirm that you want to delete the menu.

Enabling and Disabling Menus

When you are making changes to a menu, you can disable it if you do not want users to see your changes immediately when you save them. When it is ready to be distributed to the users, you enable it again. The first time the user refreshes the Navigation Pane after a menu in enabled, she will see the new version.

To Enable or Disable a Menu:

Right-click the menu button, and select **Enable** or **Disable**.

To Refresh the Navigation Pane:

Right-click below the menu buttons and next to the scroll arrows, and then click **Refresh**.

Editing Menus with the Navigation Pane Designer

The following procedure describes how to edit an existing menu. It can either be one of the menus created by the solution provider, or it can be a menu created by a Menu Suite Administrator.

To Edit Menus with the Navigation Pane Designer:

1. Click TOOLS→NAVIGATION PANE DESIGNER.
2. Click on the menu button for the menu that you want to edit.
3. Use right-click menus to access the editing options.

Changing the Name of a Menu Button, Menu Group, or Menu Item

1. Right-click the menu button, group, or item, and then click **Rename**. The cursor appears in an edit box.
2. Type the name as you want it to appear. Press **Enter** to apply the change.

Creating a New Menu Group or Inserting a Menu Item

To Create a New Menu Group:

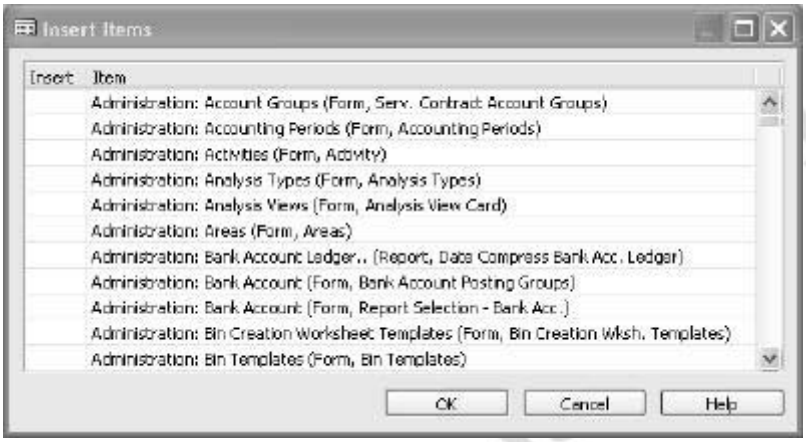
1. Right-click an item in the menu group to which you want to add a new group, and then click **Create Group**. The new group appears as the last item within the group.
2. Type the name of the new menu group. Press **Enter** to apply the change.

Inserting a New Menu Item

The Insert Items window lists all of the menu items in all of the menus in your menu suite. Each line lists one menu item. In the **Item** field, the name of the menu that the item comes from appears, followed by the name of the menu item. The items are grouped by the menu from which they come and then alphabetically by menu item name.

To insert a new menu item:

1. Right-click an item in the menu group to which you want to add a new item, and then click **Insert Items...** to display the Insert Items window:



2. In the **Insert** field, place a check mark next to the menu item(s) that you want to add and then click **OK**. The menu item(s) are added to the menu group.

Deleting a Menu Group or Menu Item

To Delete a Menu Item:

Right-click the item, and then click **Delete**. The item is removed from the menu, but you can add it again later.

To Delete a Menu Group:

Right-click the group, and then click **Delete**. The menu group and any menu items it contains will be deleted from the menu.

Moving a Menu Button, Group, or Item Up or Down

To move a menu button, group or item, right-click the button, group, or item, and then click **Move Up** or **Move Down**.

***NOTE**: You can only use this method to move an item within the group where it appears. To move it to another group, use cut and paste.*

Copying a Menu Group from one Menu to Another

To copy a menu group from one menu to another:

1. Open the menu with the menu group that you want to copy.
2. Right-click the menu group, and then click **Copy**.
3. Click the menu button for the menu where you want to add the menu group.

4. Right-click in the menu group to which you want to copy the menu group, and then click **Paste**.
5. The new group appears as the last item within the group.
6. To leave design mode, click FILE→CLOSE.

The changes are saved and will be shown next time the Navigation Pane Designer is opened. The changes will be visible to the users when they refresh the Navigation Pane.

Personalizing the Navigation Pane

End-users can change the number of menu buttons displayed in the Navigation Pane, as well as the order in which they are displayed. Users can also hide or show menu groups and menu items. By hiding parts of the menu that end-users normally do not use, it becomes easier to find the items that they do use. Hidden items are not deleted; they can always be shown again.

Hiding Menus

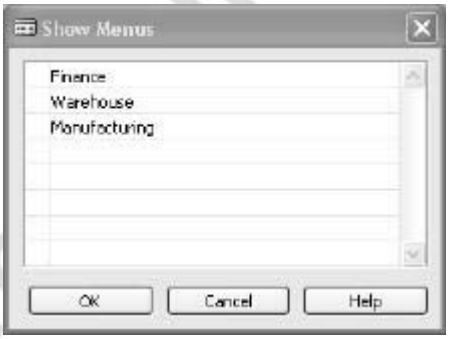
To hide a menu button right-click on the button, and then click **Hide**.

Showing Menus

The Show Menus window lists all of the menus that have been hidden. A check mark is placed next to the menus that the end-user wants to show.

To show a menu button:

1. Right-click any menu button, and then click **Show**. The Show Menu window will appear:



2. Enter a check mark next to the name of the menu button that you want to show, and then click **OK**.

Moving Menu Buttons Up or Down

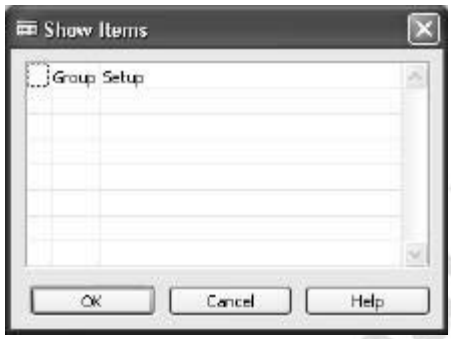
To move a menu button up or down, right-click the button that you want to move, and then click **Move Up** or **Move Down**. Repeat until the button is where you want it.

Hiding and Showing Menu Groups

To hide a menu group, right-click the group, and then click **Hide**.

To show a menu group:

1. Right-click any menu group, and then click **Show**. The Show Items window appears:



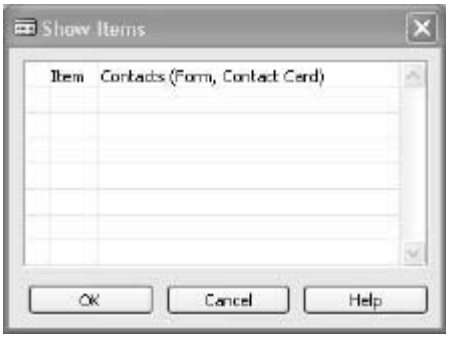
2. Notice that the second field indicates that it is a Group and the third field displays the group name. Enter a check mark next to the name of the menu group that you want to show, and then click **OK**. It will be inserted where it was before it was hidden.

Hiding and Showing Menu Items

To hide a menu item, right-click the item, and then click **Hide**.

To show a menu item:

1. Right-click any menu item, and then click **Show**. The Show Items window appears:



2. Notice that the second field indicates that it is an Item, and the third field displays the item group name. Enter a check mark next to the name of the menu item that you want to show, and then click **OK**. It will be inserted where it was before it was hidden.

The Shortcuts Menu

The Shortcuts menu is a user-defined menu located in the Navigation Pane. End-users can put together their own personal menu in the Shortcuts menu. Users can include shortcuts to:

- Forms and reports in the program
- Specific records in the database (for example, customer card 10000 or sales order 30XB2)
- Documents, folders or programs on their computer or a server
- Web sites

Creating Shortcuts in the Navigation Pane

If a user only uses a small fraction of the menu items in the menus that have been assigned to them, he can copy all of the menu items that they use to the Shortcuts menu so that everything is in one place, and then he can hide the other menus. The shortcuts can be organized to make their work more efficient.

There are two ways to create shortcuts to windows in the program.

To Copy a Menu Item or Group from an Existing Menu in the Navigation Pane

1. Click the menu button for the menu that contains the group or item you use frequently.
2. Locate the menu item or group, and right-click it.
3. Click **Add to Shortcuts**.

To Create a Shortcut to a Form or Report that You Have Open

1. Open the window or report that you want to create a shortcut to.
2. You can create a shortcut to a specific record in your database. For example, you can create a shortcut to a specific customer card.
3. If you want the shortcut to open a specific record, browse to the record (for example, a contact).
4. Click FILE→SEND→LINK TO SHORTCUTS.

   The Shortcuts menu includes shortcuts to menu items and reports:

In addition, you can include shortcuts to files outside of the program:

To Create a Shortcut to a Document, Folder, or Program

1. In the Shortcuts menu, right-click, and then click **Create Shortcut...**
2. For a shortcut to a folder, in the **Open** field, enter the name of the folder, for example C:\PROGRAM FILES OR \\PUBLICSHARE\MYFILES.
3. For a shortcut to a document or program, in the **Open** field, click the **AssistButton** and browse to the relevant file. Click **Open**. The file address and name, including the file extension, appears in the **Open** field.
4. If the file is a program file (an .exe file), enter any parameters the program expects after the file name.
5. In the **Caption** field, enter the name of the menu item as you want it to appear in the Shortcuts menu.

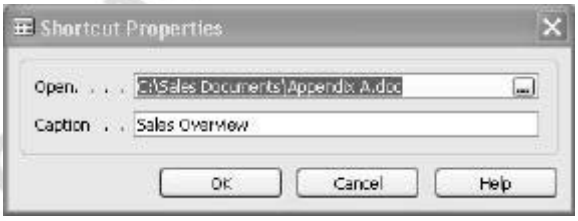
To Create a Shortcut to a Web Site

1. In the Shortcuts menu, right-click and then click **Create Shortcut...**
2. In the **Open** field, type the Web site address, for example www.microsoft.com.
3. In the **Caption** field, enter the name of the menu item as you want it to appear in the Shortcuts menu.

The menu item or group is added to the bottom of the Shortcuts menu. You can organize the shortcuts into groups and you can delete an item if you no longer need it.

Shortcut Properties

You use the Shortcut Properties window to view or modify the properties of a shortcut in your Shortcuts menu.



Field	Descriptions
Open	The **Open** field contains the name and address of the shortcut target. For example, the field can contain a Web site address or the path and file name of a program or document on your computer
Caption	The **Caption** field contains the name that you have assigned the menu item in the Shortcuts menu.

Organizing Shortcuts in the Navigation Pane

Once some shortcuts have been created in the Shortcuts menu, you can reorganize the menu. New groups can be created, and the order of the groups and items in the Shortcuts menu can be changed. Items can also be deleted if they are no longer needed.

Creating a New Menu Group

To create a new group:

1. Right-click in the group where you want to add the new group, and then click **Create Group**. The new group appears as the last item within the group.
2. Type the name of the new menu group. Press **Enter** to apply the change.

Moving Items into a Menu Group

To move an item into a group:

1. Right-click the item that you want to move, and then click **Cut**.
2. Right-click an item in the menu group to which you want to move the item, and then click **Paste**.

Moving a Menu Group or Item Up or Down

To move a menu group or item up or down, right-click the menu item or group, and then click **Move Up** or **Move Down**.

NOTE: *You can only use this method to move an item within the group where it appears. To move it to another group, use cut and paste.*

Changing the Name of a Menu Group or Item

To change the name of a menu group or item:

1. Right-click the menu group or item, and then click **Rename**. The cursor appears in an edit box.
2. Type the name as you want it to appear. Press **Enter** to apply the change.

Deleting a Menu Group or Item

To delete a menu item, right-click the item, and then click **Delete**. The item is removed from the menu, but you can add it again later.

To delete a menu group, right-click the group, and then click **Delete**. The menu group and any menu items it contains will be deleted from the menu.

Creating and Designing MenuSuite Objects

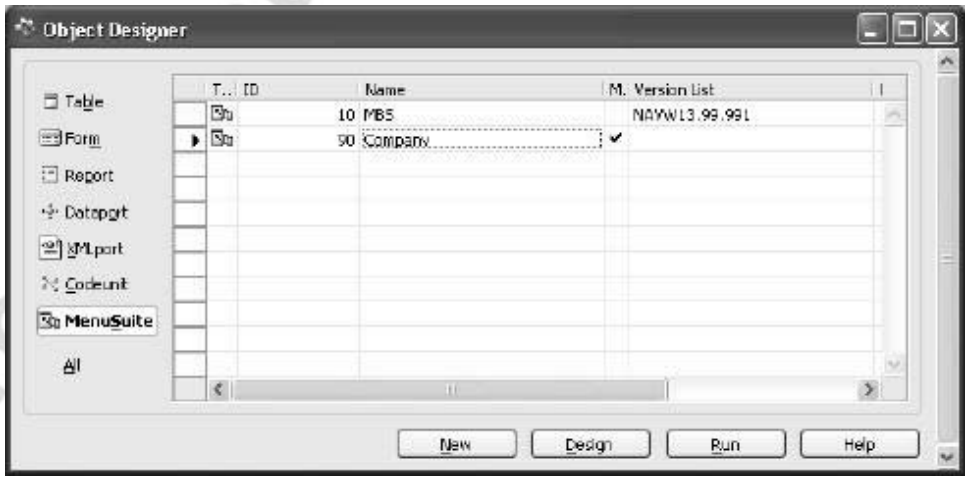
MenuSuite Object Levels

Microsoft Business Solutions provides a generic MenuSuite object, which we call the MBS level. This object is changed in various ways before end-users see its content. These changes are applied at different levels, which are categorized as Region, Country, Add-on, Partner, and Company. For example, the MenuSuite object is changed when the application undergoes localization changes, which takes place at the Country level. If you are a developer working at a Microsoft Business Solutions partner, you customize a MenuSuite object at the Partner level. You can also configure a MenuSuite object at the Company level, which is the level that administrators work on.

Changes that are made to a MenuSuite object are stored as the differences between the previous MenuSuite object level and the current one. For example, when a company administrator configures a MenuSuite object at the Company level, the modifications are stored as the differences between the Company level and the Partner level, which was the previous level. If you export a MenuSuite object in text format and then open the text file, you will see information about the changes that you have made seen in relation to the previous level.

With the exception of the Add-on MenuSuite object for which a maximum of 10 instances is allowed, there can only be one MenuSuite object for each level.

To create a new MenuSuite object or design an existing one, go to TOOLS→OBJECT DESIGNER→MENUSUITE. The MenuSuite Objects will appear:



Function Button	Description
New	This opens a dialog asking you to specify which design level you want to create an object for. If you have already created a MenuSuite object for all the levels you have permission to, a message will appear informing you of this. Once you have made a selection, the Navigation Pane Designer opens. The MenuSuite object level that you are working on is shown in the header section of the Navigation Pane Designer.
Design	This opens the Navigation Pane Designer with the chosen menu suite content displayed. The MenuSuite object level that you have selected is shown in the header section of the Navigation Pane Designer. You now have the following options:
	You can select the FILE→IMPORT/EXPORT and TOOLS→TRANSLATE→IMPORT/EXPORT options.
	You can compile MenuSuite objects (F11 or TOOLS→COMPILE). It is the object references that are compiled. If the MenuSuite object contains a reference to a nonexistent form, report, dataport or codeunit, a compilation error occurs.
Run	The **Run** button in the Object Designer is disabled when you click the **MenuSuite** button because a MenuSuite object cannot be run.

Customizing a Menu Suite

You customize a menu suite in the Navigation Pane Designer. In the following, you can read about the design options that are available for developers. You access design options by right-clicking on menu buttons, menu groups, menu items, or anywhere in the content pane area.

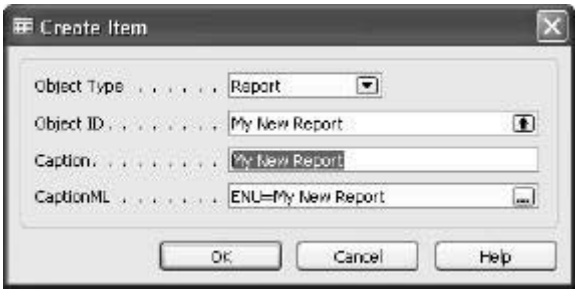
Creating Menu Items

To create a menu item:

1. Right-click in the content pane area or on a menu group or menu item, and select **Create Item**.

2. The Create Item window opens in which you specify the menu item's Object Type, Object ID, Caption, and CaptionML (multilanguage caption).

An example is shown in the following:



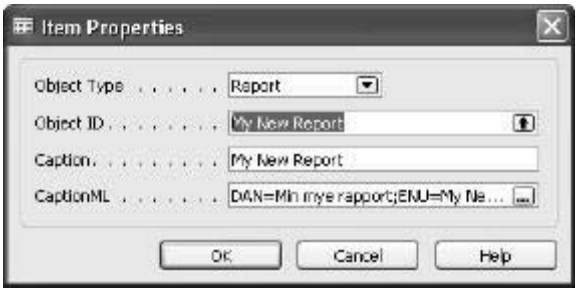
For example, if you have created a new report and want to add it as a menu item to a particular menu, then you select Report in the dropdown box in the **Object Type** field, and use the lookup in the **Object ID** field to select the report from the Report List. When you then click the **Caption** field, the name of the report is automatically filled in. You can rename it here if necessary.

The **CaptionML** field will by default show the code for the language that you use on your computer and the name of the menu item. If you have designed a menu item that will be used in different languages by different users, you must give the menu item a name in each language. When you click the **AssistButton** in the field, the Multilanguage Editor window opens where you can enter a menu item name for each language code:

Setting Properties

There are four properties that can be set for a menu item in the Item Properties window: Object Type, Object ID, Caption, and CaptionML.

The Item Properties window is accessible by right-clicking a menu item and selecting **Properties**.



Function Button	Description
Object Type	In the **Object Type** field, you can make a selection in the dropdown box. The following object types are available: Table, Form, Report, Batch Job, Codeunit, Dataport.
Object ID	You use the lookup in the **Object ID** field to select a specific object from a list. For example, if you had selected Report as the Object Type, clicking the lookup will display the Report List.
Caption	By default, the **Caption** field contains the name that has been assigned to the menu item. You can rename it here if you want.
CaptionML	The **CaptionML** field contains the code for the language that you use on your computer and the name of the menu item, for example, ENU=General Journals. If the menu item will be used in different languages by different users, you must give the menu item a name in each language. When you click the **AssistButton** in the **CaptionML** field, a window opens where you can enter a menu item name for each language code.

Exporting a MenuSuite Object

You can export a MenuSuite object in Microsoft Navision object format (*.fob) or in text format. The text file contains information about the changes that have been made to the object since the previous level. In the following, you can see an example of a text file that contains the changes that have been made to a MenuSuite object at the Company level.

Example

When an administrator makes configuration changes to a menu suite, these changes are saved at the Company level. When this Company-level MenuSuite object is exported in text format, the text file contains information about the changes that have been made to the object since the previous level, which in this case would be the Partner level. Let us suppose that an administrator has made the following changes to his company's menu suite:

- In the Finance menu, he has deleted a menu group called Setup, which contained one menu item called Accounting Periods.
- He has added a new menu item, G/L Account Card, to the General Ledger menu group in the Finance menu.
- He has moved the G/L Account Card menu item so that it is placed between the Chart of Accounts and Bank Accounts menu items in the General Ledger menu group.

The text file would contain the following information:

```
OBJECT MenuSuite 70 Company
{
OBJECT-PROPERTIES
{
Date=19-04-04;
Time=16:04:10;
Modified=Yes;
Version List=;
}
PROPERTIES
{}
MENUNODES
{
      { MenuItem ;[{FA8395A4-7A0B-4524-B0FD-20436D9711A4}]
      ;Name=G/L Account Card;
            CaptionML=ENU=G/L Account Card;
            MemberOfMenu=[{F8D2429D-034B-4C58-9B5E-
            81BE962DB1BC}];
            RunObjectType=Form;
            RunObjectID=17;
            ParentNodeID=[{B12180CF-0EFB-43AD-9118-
            7765E953AAFD}];
            Visible=Yes;
            NextNodeID=[{7A5B6DDE-B44B-41A0-95DA-
            69B7109F0E32}] }
{     ;[{1BB06483-AFFD-4750-BF62-3ABA035E11B7}];
            Deleted=Yes}
```

```
{     ;[{DF601C8A-07F7-4841-8929-9F2065BCB302}];
            Deleted=Yes}
{     ;[{8AC7917D-2C91-457D-80D6-A24B42F71AE7}];
            NextNodeID=[{FA8395A4-7A0B-4524-B0FD-
            20436D9711A4}]}
      }
}
```

The content of a MenuSuite object can be described as having the following characteristics:

- It consists of a set of menus.
- A menu contains a collection of menu nodes, which are displayed in the Navigation Pane/Navigation Pane Designer in a tree structure.
- A menu node can be either a menu group or a menu item.
- A menu node has a globally unique identifier (GUID) and various properties.
- A menu group contains a collection of menu nodes.
- A menu item is the lowest level in the tree. When you click a menu item, its associated form, report, data port, or codeunit is run.

How Upgrades Affect New Menus

Prior to modifying the menu suite, a strategy must be decided upon:

- To modify the original menus
- To create new menus (these can be based on copies of the original menus)

Menus in the menu suite that are original menus from a solution provider have the symbol >> on the menu button to the left of the menu name. When a customer has their menu suite upgraded, the original menus will be upgraded, and any configuration changes that have been made to them will be merged into the new menus. For example, a menu item that was added to a menu group – but where the menu group's contents now have changed after an upgrade – will be placed as the last menu item in the menu group.

However, there will be cases where a straightforward merge cannot be carried out. For example, if a menu item had been added to a menu group that is no longer a part of the menu after an upgrade, then no merge can be made. Instead, the menu item will be placed in a Lost Items group at the bottom of the menu tree. After an upgrade, action can be taken on the menu items in this group – either by inserting them somewhere in the current menu, in another menu or by deleting them. If the menu for the lost items cannot be found, a Lost Items menu will be created. The Lost Items group or menu is not visible to the users.

New menus that users have created are not affected by an upgrade, so if there are new menu items available to the customer from the upgrade that they would like to include, they will have to manually add them to their menus.

Test Your Knowledge – Basic Customization

1. What is the difference between the Navigation Pane and the Navigation Pane Designer?

2. Describe the ways an end-user can personalize the Navigation Pane.

3. What are the MenuSuite Object Levels?

4. Describe the Shortcuts Menu and how an end-user might use it.

5. Explain why a MenuSuite Object might be exported.

Test Your Skills – Copy and Modify a Menu

Scenario: As the Menu Suite Administrator, you have been asked to create a new Finance Menu without Fixed Assets.

All Levels

Your tasks are as follows:

1. Copy, paste, and edit an existing menu.
2. Remove the Fixed Assets menu item.

Need a Little Help?

- Go to TOOLS→NAVIGATION PANE DESIGNER.
- Copy the Finance Menu.
- Paste the Finance Menu.
- Rename the Finance Menu.
- Remove the Fixed Assets menu item.

Test Your Skills – Create a Shortcut Menu

Scenario: You are in the Marketing department and want to see menu items related to Marketing, as well as the Salesperson card and the Salesperson – Opportunity Report. You also access the Microsoft Web site quite often so you want a direct link there as well.

All Levels

Your tasks are as follows:

1. Create a Shortcut Menu directly related to the Marketing sub-folder.
2. Add the Salesperson Card and the Salesperson – Opportunity Report to the Shortcut Menu.
3. Create a link to the Microsoft Web site.

Need a Little Help?

- From the Sales & Marketing menu, add the Marketing folder to the Shortcut menu.
- Go to SALES & MARKETING→SALES and add the Salesperson Card.
- Go to SALES & MARKETING→SALES→REPORTS→SALESPEOPLE/TEAMS, and add the Salesperson – Opportunity Report to the Shortcut menu.
- From the Shortcuts menu, create a shortcut to the Microsoft Web site: http://www.microsoft.com/.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 10: MICROSOFT NAVISION DATABASE ADMINISTRATION TOOLS

Overview

The database is the heart of Microsoft® Business Solution–Navision®. All the information, companies, modifications, reports, and so on are stored here. Therefore, it is important that you know how to manage the database and that you are familiar with the tools that Microsoft Navision provides for doing this.

This chapter contains guidelines for maintaining Microsoft Navision databases on both the Database Server and the Microsoft® SQL Server®. This includes when and how to back up your Microsoft Navision data, some guidelines for maintaining SQL Server statistics that are used to optimize performance, and some information about more advanced features.

Managing the Databases

This section describes the basic operations involved in working with a database, as well as some more advanced features.

Testing Databases

To safeguard against errors, you should frequently check the integrity and consistency of your databases.

Without Opening the Program

You can test the database without opening Microsoft Navision. This is done by setting the program property DB Test In the **Target** field. Here is an example for SQL Server:

```
C:\PROGRAM FILES\MICROSOFT BUSINESS SOLUTIONS–MICROSOFT
NAVISION\CLIENT\FINSQL.EXE NTAUTHENTICATION=YES, SERVERNAME=MY SERVER,
DATABASE=MY DATABASE, DBTEST=NORMAL
```

The possible settings for DB Test in the Target field are Minimum, Normal, and Maximum. You can customize the database test if you start the test from within Microsoft Navision.

From Within the Program

To start the test from within the program, click FILE→DATABASE→TEST. The Test Database window appears:

You determine the extent of the test by selecting one of the options at the top (such as Minimum). When you select an option, the individual tests included in that option are run.

***NOTE**: In the SQL Server Option for Microsoft Navision, most of the tests also involve running the SQL Server database consistency checker (DBCC).*

The following table lists what the different tests involve. It also lists the SQL Server DBCC tests that are carried out:

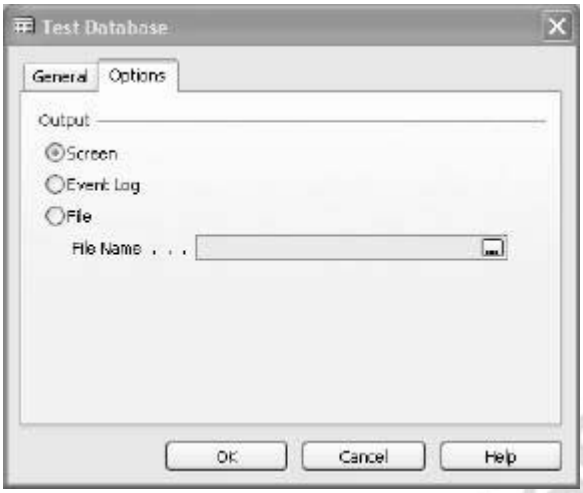
Level	Test Fields	Features Checked
Minimum	Test primary keys and data	All records in all tables can be read. Records are sorted in ascending order according to the primary key. All fields are correct in relation to the field type. DBCC CHECKTABLE (<Table>,NOINDEX)
	Test BLOBs	All BLOBs (fields for pictures – on the item card, for example) can be read. DBCC CHECKTABLE (<Table>,NOINDEX) You can read about BLOBs in the Introduction manual and the Application Designer's Guide.

Level	Test Fields	Features Checked
Normal	All the fields included in the Minimum test, plus:	
	Test secondary keys	All secondary keys in all tables can be read. Sorting is done correctly according to the secondary key. All fields in the sorting have the correct field type. DBCC CHECKTABLE (<Table>)
	Test space allocation	All space in the database is either used by a sorting key or is available. DBCC CHECKALLOC (<Current Database>)
Maximum	All the fields included in the Normal test, plus:	
	Test relationships between primary and secondary keys (not applicable for SQL Server)	The connection between the primary and secondary keys is correct.
	Test field relationships between tables	All fields that have relationships to other fields can be accessed from the field to which they are related.
Custom	The same fields as in the Maximum test, but you can deselect the ones you do not want to use.	Whatever you select. However, some of these tests correspond to SQL Server tests and must run concurrently.

Primary and secondary keys, which are mentioned in the table shown above, are sometimes described as indexes and are used, for example, when you sort information. The keys determine how information in a table is ordered. You switch keys to sort the information in a table in a different way. For example, you might want to sort your customers by name or by number. See the Application Designer's Guide manual for a description of how information is sorted and how to change the sorting.

How often you need to test the database depends on how secure the rest of the system (including the network) is and what level of security you need. It is a good idea, however, to test before you make a backup – especially if you do not use the Microsoft Navision backup function.

Click the **Options** tab to specify how the output from the database test is managed:



The output from the database test consist of the errors messages that are generated. These can be handled as follows:

Field	Comment
Screen	The error messages are displayed on the screen. Each time an error message is displayed, you must click OK before the test continues. This can be quite troublesome if the test generates a large number of messages.
Event Log	The error messages are written in the operating system's event log. For more information about the event log, see the operating system's documentation.
File	The error messages are written to a text file. Use the **AssistButton** to specify the name and location of the text file.

If you select Event Log or File, the database test is not interrupted and will not take so long. You can then review any error messages that were generated and repair the things that caused them.

***NOTE**: For the SQL Server Option, it is recommended that the Single user database option is selected when performing tests. To do this click FILE→DATABASE→ALTER, and in the Alter Database window, select the **Options** tab and select the **Single user** field.*

If an error occurs, the program stops and displays an error message indicating what the error is and where it occurs.

The following steps must be taken:

For Microsoft Navision Database Server:

1. Export a Microsoft Navision backup copy of the database.
2. Create a new database and restore the backup into it.
3. Run a Maximum test on the new database to check it.
4. If it is error-free, check to make sure that the company data is correct. Afterwards, you can continue to work in the new database.
5. If this does not work, contact your Microsoft Certified Business Solutions Partner for help.

WARNING: *Never delete a corrupted database (one that contains an error) before a new, tested database has functioned without errors for a period of time.*

For SQL Server Option:

- If the error refers to SIFT™ sum totals, the index containing the fields listed should be deleted and re-created. The test should then be run again. If there is still an error, follow the procedure explained in the section called "Finding Errors In the Tables."
- Any other type of error could mean that the database has become corrupt and inconsistent. It may be possible to correct these inconsistencies by running the SQL Server DBCC CHECKDB command, using a REPAIR option. We recommended that only the REPAIR_FAST or REPAIR_REBUILD options be used to prevent loss of data.

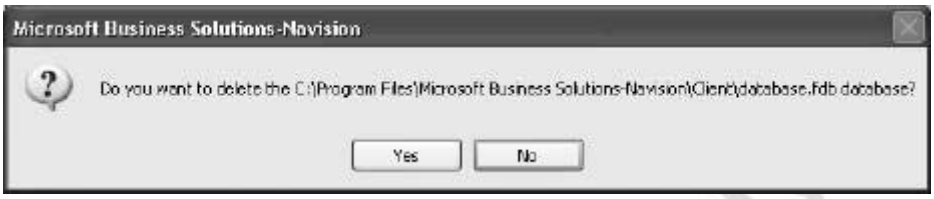
An alternative to repairing the database and the preferred method of correcting database errors is to restore the database from backups. For more information, see the section titled "Database Maintenance."

It is important to have a reliable backup procedure in place, and you must ensure that your backups can be restored successfully.

Deleting the Database

Never delete a database without making a backup. Save one or more copies of the backup in a secure place.

After you have made the backup, you can click FILE→DATABASE→DELETE to remove the copy of the database that will no longer be used. Before the database is deleted, you will have to answer two messages like this one:



When you delete a database, everything in the database is deleted, including any customizations you have made.

You are not able to delete a database that you cannot open from within Microsoft Navision. You cannot delete a database if other clients are connected to the database.

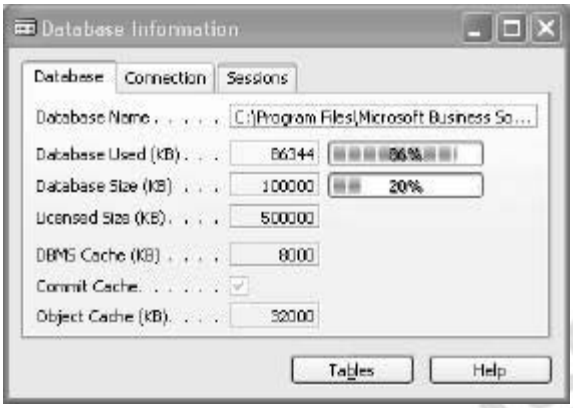
Deleting Part of a Database

If you do not want to do anything quite so drastic as deleting the entire database, there are various other ways to delete information:

- To remove *old information,* use the Date Compress batch jobs. On the main menu, click ADMINISTRATION→IT ADMINISTRATION→DATA DELETION→DATE COMPRESSION for the relevant application area.
- To remove *individual records,* click EDIT→DELETE. There must be no open entries or nonzero balances for the records you want to delete.
- To remove a *company,* click FILE→COMPANY→DELETE.
- If you have access to the development environment for Microsoft Navision, you can delete *individual objects.* You can read about the development environment in the *Application Designer's Guide.*
- If you need to delete *almost everything* except a couple of objects, such as some reports, you can save the objects by exporting them before you delete the database. You can then click FILE→DATABASE→DELETE to delete the database. Finally, you can restore the old objects to a new database.

Database Information

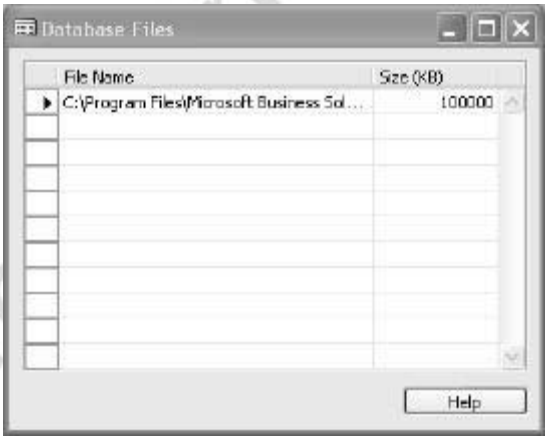
To see information about the current status of the database in Microsoft Navision, click FILE→DATABASE→INFORMATION. The Database Information window appears:



This is where you can check to see whether everything is set up correctly, whether there is enough space available in the database, how much cache has been allocated to the program, and so on. (You cannot change anything in this window – you can only view information.)

Database

The **Database** tab contains information about the database that is currently open. The name of the database appears in the **Database Name** field. A database can be set up to consist of several individual files. You can see the names and sizes of the files if you click the **AssistButton**. The Database Files window appears:



If you click TOOLS→ZOOM (CTRL + F8) from the menu bar while this window is displayed, you will get more detailed information about the database – information that can be used for statistical purposes and to check the performance level. You can read more about performance issues in "Troubleshooting Microsoft Navision."

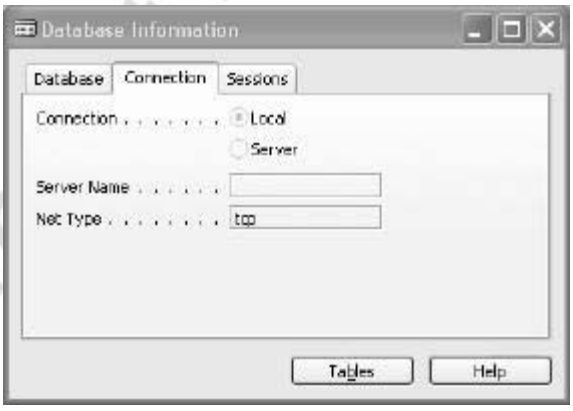
The Database tab also displays information about how much of the total database is currently in use and what percentage of the total space in the database is being used. This should not exceed 90%. If it does, you can expand the database by clicking FILE→DATABASE→EXPAND. The **Database** tab displays the following information about the database:

- How large the total database is and what percentage this is of the total database size you have permissions for.
- How much database space you have purchased permissions for.
- Whether DBMS cache has been allocated in the system – and how much. The value affects the program speed. To specify cache size, use the program property Cache or click TOOLS→OPTIONS and enter the size as the value for DBMS Cache (KB).
- Whether Commit Cache is turned on (a check mark means that it is). The commit cache affects the program speed. To specify the size of the commit cache, use the program property Commit Cache, or click TOOLS→OPTIONS and enter the size as the value for Commit Cache.
- Whether object cache has been allocated in the system. The Object Cache program property is used only on clients where it affects the speed of the program.

***NOTE**: Due to the size of the memory allocation blocks (8KB), the **DBMS Cache** field can display a slightly different figure than expected. The DBMS Cache rounds down to the nearest block, for example, if you have allocated 500, it will be rounded down to 496.*

Connection

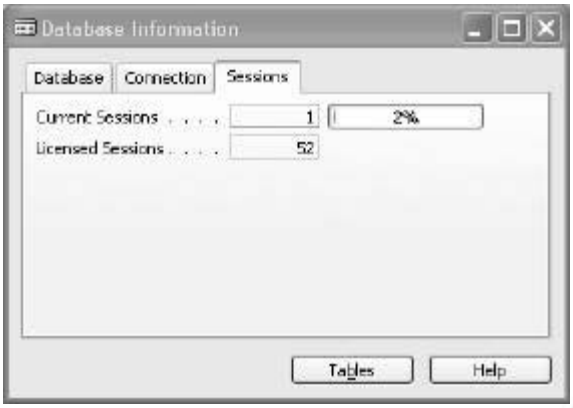
On the **Connection** tab, you can see the following information:



- Whether the database you are working in is located on your computer or on a server.
- If you are working in a database that is located on a server, the name of the server and the network protocol that is being used.

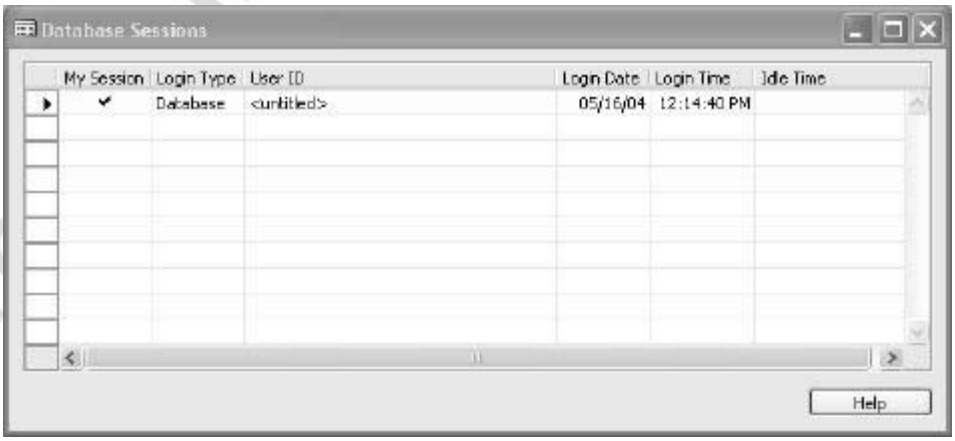
Sessions

The **Sessions** tab contains the following information:



- The **Current Sessions** field tells you how many sessions (active Microsoft Navision programs) are presently connected to the database. The number is shown both as the number of sessions and as a percentage of the maximum number of sessions for which you have permissions. Each computer can have several sessions running at once.
- The **Licensed Sessions** field tells you the maximum number of sessions (active Microsoft Navision programs) that can connect to the database. If you need more, you must obtain them from your Microsoft Certified Business Solutions Partner.

If you click the **AssistButton** in the **Current Sessions** field, the Database Sessions window appears, listing details of all the sessions connected to the server:

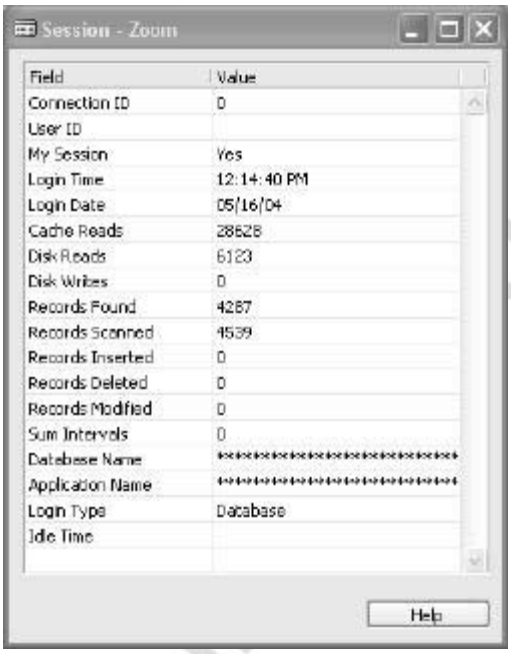


The window lists all the current connections to the database and the time when each user logged on. Each line represents one login. A user can appear more than once if they have started more than one session. The session that you are currently using has a check mark in the **My Session** field.

Terminating a Session

You can terminate one of the sessions by selecting the line in question and deleting it. The session will no longer take up any resources on the server. This user will then be disconnected from the server and will have to restart the program if they want to continue working. To terminate a session, you must have permission to delete data from the Session table.

If you put the cursor on a line in the Database Sessions window and click TOOLS→ZOOM on the menu bar, you will get more detailed information about a particular session. The following window appears:



The information here can be used for various statistical purposes.

The **Tables** button at the bottom of the Database Information window opens a window displaying information that is used for analyzing where and how data is distributed in the database. You can read about this in the following section.

Standby and Hibernation

Microsoft Navision supports the standby and hibernation facilities provided by Microsoft® Windows® 2000.

Putting your computer on standby means that the entire computer switches to a low power state. When on standby, all devices, such as the monitor and hard disks, turn off and your computer uses less power. When you want to use the computer again, it comes out of standby quickly, and your desktop is restored exactly as you left it. Standby is particularly useful for conserving battery power on portable computers. Because standby does not save your desktop state to disk, a power failure while on standby can cause unsaved information to be lost.

Putting your computer in hibernation means that before shutting down, your computer saves everything that is currently in memory to disk, turns off your monitor and hard disk, and then turns off your computer. When you reactivate your computer, your desktop is restored exactly as you left it. It takes longer to bring your computer out of hibernation than out of standby.

Shutting Down

Individual workstations can go to standby or hibernate after being idle for a certain length of time. It is also possible to force the computer to go to standby from the Windows' Shut Down dialog box.

However, Microsoft Navision will not allow Windows to go into hibernation or standby in the following situations:

- Carrying out a transaction, for example, posting an order
- Printing a report
- Making a backup

If you attempt to make the computer go to standby or hibernation from the windows Shut Down dialog box during any of these situations, a window will appear informing you that Microsoft Navision is busy and that shutting down is not yet possible.

When you click **Cancel** in this window, the hibernation or standby procedure will be postponed.

If you ignore this window, the computer will go into hibernation or standby when Microsoft Navision has completed its task.

Restarting

When you restart your computer after it has gone to standby or is in hibernation, it will restart with the desktop exactly as it was when you left it, and you will see the same Microsoft Navision windows that were open. However the information displayed in these windows will also be the same as before and is, therefore, not necessarily up to date.

The window will not be updated until you use the program and actively update the window in question.

Using a Laptop

If you are using a laptop computer and close the lid, Windows will shut down. However, an active Microsoft Navision client will force Windows to wait until the operation it is carrying out is finished before shutting down. When the user reactivates Windows, the Microsoft Navision client will be exactly as it was when you left it.

Database Maintenance

Making Backups

This chapter contains guidelines for how, when, and why to back up your Microsoft Navision data. It describes various methods of making backups and the advantages and disadvantages of the various storage media.

Why, When, and How to Make Backups

You make backups so that there is always an extra copy of your company data to restore into the application if a problem should arise with the working copy. You should make backups for your own sake, but in most countries it is also a legal requirement.

This means that if you upgrade to a new version of Microsoft Navision or change the installation in another way, you must still keep at least one copy of your company data in a readable format (and in a safe place).

If you upgrade to a new version of the accounting system, for example, from Microsoft Navision (the old text-based version) to the new Microsoft Navision, it may be necessary to save the old system in order to be able to access the old information. You do not need to have the old system installed – you can just save it and install it if you need to.

Determining When to Make Backups

Determining a procedure for creating backups is a vital part of maintaining your database. If you make frequent entries in your database, you will need a backup procedure that guarantees the reliability of your data and will allow you to fully recover your data after any failures that may occur.

We also recommend that you always make a backup before:

- Expanding the database.
- Deleting a database. (You cannot retrieve it after it has been deleted unless you have saved a backup.)
- Installing or removing equipment from the computer or computers on which the Microsoft Navision database is stored.
- Performing date compression (both server options) and optimizing tables (Microsoft Navision Database Server only). For more information about these topics, see “Optimizing Microsoft Navision.”
- Copying the database or parts of it with an operating system command.
- Using programs to optimize your hard disk.

It is always a good idea to have an up-to-date copy of your company data in a secure place (a bank box, for example) in case of fire, theft, computer viruses, and so on.

Procedures for Making Backups

There is no formula for how often you should make backups, but remember that if you restore a backup that is a week old, you will have to reenter all the information for the past week. A backup is no more secure than the original data, so you must also protect yourself against errors in the backups. One way to do this is to create a system of backups at several levels:

- Every day, back up the company. If you have made design changes, make backups of the application objects or the database.
- Every week, back up the database. Store the results so that the five most recent backups are in different locations. (About every month or so, put a backup copy in the company's safe or other secure place.) If a problem occurs in one place, the previous backup can probably be used.
- Every six months, restore the latest backup onto a different computer than the one you normally work on. Then test the contents of the hard disk. If there are no errors, put the copy in the company's safe or another secure location outside the company.

Using this backup procedure, you will at most lose only a limited amount of data, because you should be able to restore data from a previous backup without any problem.

Backups in a Network

If you work with Microsoft Navision in a network, a good procedure is to make a backup to the local hard disk, copy it to the network server, and compare the copy on the server to the backup on the local hard disk.

Making Backups on Microsoft Navision Database Server

You can make backups by copying the database with an operating system command, but there are several important advantages to using the Microsoft Navision backup function:

- The system tests the database for errors, so incorrect information is not copied to a backup.
- The data is compressed, so it takes up as little space as possible.
- The system calculates how much space the backup will use.
- You can keep working in Microsoft Navision while you are making a backup.

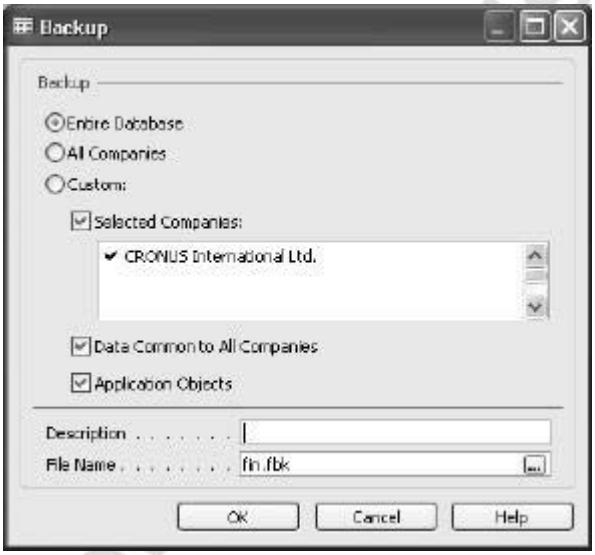
Whenever you create a new database, you must always use a Microsoft Navision backup to restore the "Data Common to All Companies" and the "Application Objects" into the new, empty database. Data common to all companies includes the report list, permissions groups, user Ids, and printer selections, but no real company data.

HINT: *You can make a backup while other users are using the database because when you do, the program backs up the latest version of the database. If a user enters something into the database, a new version of the database will be generated, but the backup program will continue to back up the version that existed just before this new version of the database was created.*

Making a Backup

Before you make a backup, you must open the database.

To make a backup, click TOOLS→BACKUP, and the Backup window appears:



You can now specify how extensive you want the backup to be. Click the **Option** button next to the type of backup that you want to make:

- Entire Database (including all companies in the database, data common to all companies, and application objects)
- All Companies (that is, only the companies)
- Custom (whatever you select)

If you select Custom, you must place a check mark next to the companies that you want copied. You do this in the list of companies under Selected Companies. You can make a backup of any item or combination of objects that you like – for example, the application objects and one company. You cannot make a backup of part of a company.

It is best to make an complete backup. If you need to restore a backup later on, you do not have to restore the entire backup – you can choose how much of the backup to restore.

Description, Name, and Location of the Backup

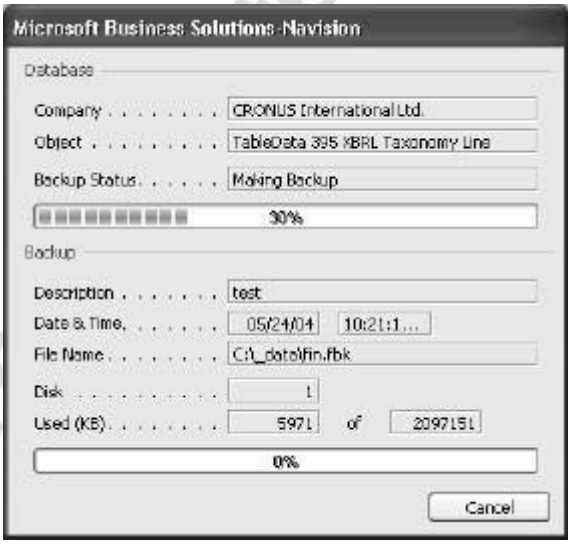
You must give the backup a unique description in the **Description** field at the bottom of the window. In the **File Name** field, enter a name for the backup. Microsoft Navision will suggest file names containing consecutive numbers and the file name extension .fbk. It is a good idea to use the default because Microsoft Navision will use the same default when you restore backups. It will also help you get a quick overview of the backups you have.

The file name of the backup includes the path (location on the disk or network). If you enter only a file name, the backup will be saved in the current folder on the current drive. This will normally be in the same place as Microsoft Navision. If you want to save the backup in a different location (because it takes up too much space, or because you want to save it on disks, for example), enter the path and the name in the **File Name** field.

To save the backup with the correct name and location, click the **AssistButton** in the **File Name** field. A standard Windows dialog box appears, and you can use this to select a target drive and directory where you want to store the backup. Type the file name of the backup. Click **Save** when you have finished.

The **File Name** field now contains the name of the backup.

Click **OK** to start making the backup, and the following window appears:



While the backup is being made, you can see how much of the database has been copied so far in the top status indicator. You can see the status of the disk or disk location it is being copied to on the bottom status indicator. To stop the backup, click **Cancel**. If you do not cancel it, the backup will proceed, and you will receive a message when it has finished.

Verifying a Microsoft Navision Backup

To verify that a backup is consistent, simply restore it to a new, empty database. If you are able to read the data in the database, the backup is consistent. If you have only backed up a company, restore the backup to a database that contains only the "Application Objects" and "Data Common to All Companies," and see if you can read any data in the company.

Automatic Backup on Windows Server

If you want to make an automatic backup of the database at a specific time on Windows, and you run a client/server installation with the server running as a Windows service, you could use the built-in AT and net start/stop command. The AT command runs a given batch file at a given time. Type "AT" after a command prompt to get further help for this command. The batch file could look like this:

```
net stop <myserver>
backup <path and databasename>
net start <myserver>
```

Note how myserver is the name of the Microsoft Navision Database Server, running as a service on Windows. Backup is the name of the backup program that backs up your database named "databasename." Store these three lines in a batch file, and supply this batch file as a parameter to the AT command. Remember to supply the full path to the batch file. The AT command requires that the Windows Schedule service is running.

Server Based Backup and Microsoft Navision Database Server

Microsoft Navision also has a server based backup program called HotCopy. This program is installed with Microsoft Navision Database Server and is stored in the same directory as Microsoft Navision Database Server.

HotCopy can only be run from the server location and can only create backups on hard disks. You cannot make incremental or differential backups. You can make a backup of a database while clients are using it. The backups are file copies of the database and are not compressed.

You can make a backup of a database while clients are using it. However, if these clients are entering large amounts of data, an error can occur in the database backup. Therefore, we recommend that you make your database backups when no clients are using the system.

Using HotCopy

You can only create a backup from the command prompt. You must specify a number of parameters:

```
hotcopy

source=databasefile1 (databasefile2…databasefile16)

destination="backup directory1" ("backup
directory2"…"backup directory16")

[description="This is the backup for Monday"]
[email=username@Microsoft Navision.com]

[servername=myserver]

[user=myname password=mypassword | osauthentication=yes]
[dbtest=normal]
[cc=yes]

[nettype=tcp|netb]
```

The parameters are case sensitive, and those that are enclosed in [ ] are optional.

Every entry that contains a space must be placed inside quotation marks.

The following example only uses the mandatory parameters. The database is stored in the same directory as the server based backup program, and the backup is stored on another computer:

```
hotcopy source=database.fdb
destination=\\backupcomputer\backup\Monday
```

The following example uses all the parameters, and the database is divided into two files:

```
hotcopy

source=C:\database\database1.fdb C:\database\database2.fdb
destination=\\backupcomputer\backup\Monday

description="This is the backup for Monday"

email=username@Microsoft Navision.com servername=myserver

user=myname
```

```
password=mypassword

dbtest=normal cc=yes

nettype=tcp
```

Note that you must enter the path to each database file.

Backup Description File

Alternatively, you can refer to a backup description file that contains all the relevant parameters. We recommend that you use the backup description file.

```
hotcopy source=backupdescription.txt
```

You can give the backup description file any name that you like. The parameters are:

Parameter	Explanation
source	The database file to be used for the backup. You must use this parameter. This can be either the name of the database or the name of a database file. If you want to make a backup of several database files, you must specify all of the file names. If you are using a backup description file, you must enter the name of the backup description file. You must enter either the absolute path or the relative path to the database file(s). When no path is specified the database file(s) must be stored in the same directory as HotCopy.exe. You can also enter the full path of the backup description file. If a backup description file is specified as the source file, all the other command line arguments are ignored. The format of the backup description file is described below.

Parameter	Explanation
destination	The path of the directory to which the database file is to be copied. You must use this parameter if you are creating the backup with the command line arguments. If only one path is specified for the destination but several files are specified as the source, all the files will be backed up in the same destination. If all the database files have the same name, the backups must be created in different directories. The database file(s) that you backup will be saved in the destination directory with the same name as the original(s). If you have previously made a backup in this directory, HotCopy will overwrite it. HotCopy will not inform you that it is overwriting the earlier backup. If the destination path is the same as the source path, the database file will be saved with a new suffix ".bak". For example, database.fdb will be saved in the source directory as *database.fdb.bak*. If you have previously made a backup in this directory, HotCopy will overwrite it. HotCopy will not inform you that it is overwriting the earlier backup.
dbtest	A database test is performed before the backup is made. You can choose between three different tests that correspond to the standard Microsoft Navision database tests. The possible values are: minimum, normal, and maximum. For more information about database tests, see the section titled "Testing the Database."
cc	A consistency check should be performed after the backup files are saved. The consistency check is a bit for bit comparison of the original database with the backed up one. The possible values are: yes and no.
description	This is a free text that describes the backup. This text is displayed as the subject in the E-mail that the administrator can receive and as the "source" entry in the Event Viewer.

Parameter	Explanation
email	The E-mail address of the person who is to be informed about the final status of the backup procedure. The E-mail message contains the following information: Subject: This is the backup for Monday (The description of the backup). Backup started at 2001/03/15 10:02:50. Backup ended successfully at 2001/03/15 10:06:06. If the backup failed the E-mail will only contain the description of the backup and a message informing you of the error that caused the backup to fail.
servername	The name of the Microsoft Navision Database Server that HotCopy should connect to when you initiate the backup procedure. If you omit this argument, HotCopy will try to create a connection using the "local host" name. If this also fails because Microsoft Navision Database Server is not running, HotCopy will create a "local" connection. In other words you can create a backup when the database server is not running. We recommend that you only use HotCopy when Microsoft Navision Database Server is running.
osauthentication	Whether or not the person creating the backup can use single sign-on (NT authentication). The only value you can enter is yes. You can either enter this parameter or the "user" and "password" parameters.
user	If Windows authentication is not used, this is the user ID of the person performing the backup.
password	If Windows authentication is not used, this is the password of the person performing the backup.
nettype	The network protocol that is used for communication between the server and the clients. The possible values are: netb (NetBIOS) and tcp (TCP/IP).

The Backup Description File

The backup description file contains information that HotCopy will use to perform the backup. The file has two section containing one or more entries. The backup description file can include comments – any text string that begins with a semicolon and does not take up more that one line.

The backup description file must have the following format: [Backup Files]

```
\\backupcomputer\backup\Monday\database.fdb=\\computer\shar
eName\Bac kupDirectory

\\backupcomputer\backup\Monday\database1.fdb=\\computer\sha
reName\Ba ckupDirectory

\\backupcomputer\backup\Monday\database2.fdb=\\computer\sha
reName\Ba ckupDirectory

[Options]
;This is the description of the Monday backup
description=This is the backup for Monday dbtest=normal

cc=yes

email=operator@Microsoft Navision.com

servername=myserver

user=myname

password=mypassword

osauthentication=no nettype=netb
```

You must fill in the section called [Backup Files]. In this section, you specify the database name on the left hand side and the backup directory on the right hand side. You can only specify a directory as the destination for the backup.

Output

The outcome of the backup procedure is always specified in the Event Log of the computer that was used to start the backup procedure. The backup administrator can also be notified of the result of the backup procedure by E-mail.

***NOTE**: If you have set up Microsoft Navision Database Server as a service and want Hotcopy to store the database backup on a remote computer, you must change the Log On options of the Microsoft Navision Database Server service. You must give the Microsoft Navision Database Server service the credentials of a domain user that has the appropriate rights on the remote computer and not use the default value (local system account).*

Testing Before Using other Backup Methods

The Microsoft Navision backup function makes a backup by creating a file containing the most basic information from the database. When the backup file is restored back into the system, Microsoft Navision recreates the rest of the information from this file. The larger the database, the longer it takes to back it up and to restore the backup. Because of this, users of installations with large databases often choose to back up onto tape, using the tape station's backup program.

You can also use the Microsoft Navision backup function to make frequent backups to the hard disk and then back up the entire hard disk onto tape. Doing this backs up both the working database and the Microsoft Navision backup of the database.

Testing the Database before Copying It to a Tape Station

If you choose to make backups without using the Microsoft Navision backup function, you must first test the database for errors. To do this, use the DB Test program property or click FILE→DATABASE→TEST on the menu bar. If you choose to use the built-in backup function, you do not have to run the test. While the backup is in progress, the backup program checks the database for errors. (It tests primary keys and data.)

Restoring Microsoft Navision Backups

Before you can restore a Microsoft Navision backup, you must create an empty database into which you can restore the backup. This is because you cannot restore data over existing data.

Creating a new database is described in the chapters for the server option that you are using. Remember that all the database files should be the same size.

Although you do not want the new database to be too large, it must be large enough to contain the backup. If the backup was made using the Microsoft Navision backup facility, it will be packed, and you may not know its size when it is unpacked. In this case, start with the size that you think you need. If that is not large enough, you will get an error message when you try to restore, and you can expand the database a little at a time. (Expanding the database is described in the "Microsoft Navision Database Server.") Then you can try to restore the backup again.

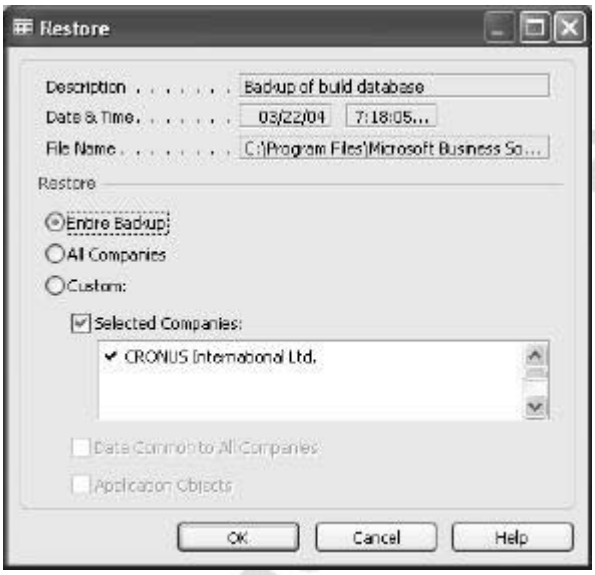
When you have created the database and restored "Data Common to All Companies and Application Objects,' proceed in one of the following ways:

- If the backup was made with the Microsoft Navision backup function, click TOOLS→RESTORE to restore the backup.

- If the backup was exported to a tape station using the tape station's backup program, use the tape station's corresponding restore program. Remember that using the tape station's own restore program will probably overwrite the entire existing database.

When you click TOOLS→RESTORE, a standard Windows dialog box appears and you can use this to locate the backup on the disk, hard disk or network. Find the folder containing the backup, and select the backup file.

Click **Open** and the Restore window appears:

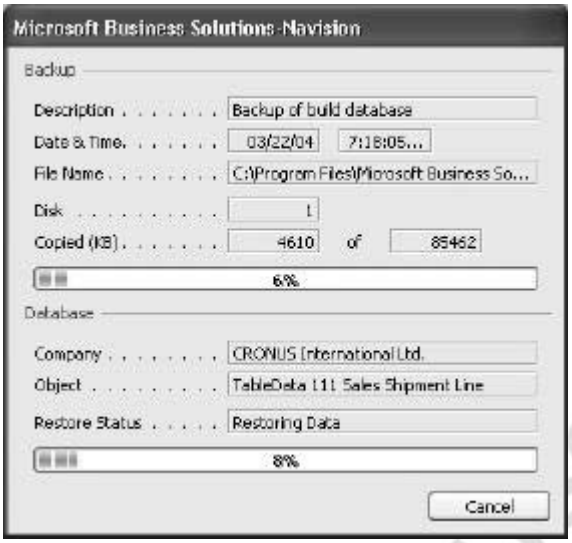


The name of the backup appears in the **File Name** field with its description and the time and date of its creation in the two fields above it. In the lower part of the window, you can select how much of the backup will be restored:

- Entire Backup (including all the companies in the database, Data Common to All Companies, and Application Objects)
- All Companies (that is, only the companies)
- Custom (the Selected Companies and can also include Data Common to All Companies and Application Objects). Remember that whenever you have a completely empty database, you must start by restoring a backup that contains at least these two options. You cannot restore the data common to all companies into a database that already has other tables in it.

You can limit the amount of the backup that you want to restore. To do this, remove some of the check marks in the window.

After you have made your selection, click **OK** to start the restore. A status window appears allowing you to monitor the progress of the restore procedure:



The amount of time it takes to restore a database depends on the size of the database.

The Minimum Size Needed to Restore

To calculate the amount of free space that is required in the database to restore a backup, use the following formula:

Free space needed in database = twice the size of the selected items to be restored + the size of the generated secondary keys + twice the amount of the data in the largest table.

The last term represents a worst-case scenario, and you can usually manage with less space than this formula calculates. The generated secondary keys are those keys that were active when the backup was made.

Making Backups in the SQL Server Option

Determining a procedure for creating backups is a vital part of maintaining your database. If you make frequent entries in your database, you will need a backup procedure that guarantees the reliability of your data and will allow you to fully recover your data after any failures that may occur.

We also recommend that you always make a backup before:

- Altering a database.
- Changing the collation used by the database.
- Deleting a database. (You cannot retrieve it after it has been deleted unless you have saved a backup.)

- Changing any server-wide or database configuration options.
- Adding SQL Server logins or carrying out any other security-related operation.
- Installing or removing equipment from the computer or computers on which the Microsoft Navision database is stored.
- Performing date compression and optimizing tables.
- Using programs to optimize your hard disk.

It is always a good idea to have an up-to-date copy of your company data in a secure place in case of fire, theft, computer viruses, and so on.

We also recommend that you make a backup of the master database in SQL Server after performing any operation that changes the information in the master database.

The operations that update the master database and require a backup include:

- Creating or deleting a database. However, if a database grows automatically as a result of the autogrow feature, this does not affect the master database. Adding and deleting files and filegroups does not affect the master database.
- Adding logins or carrying out other login security-related operations.
- Altering server-wide or database configuration options.
- Creating or removing backup devices.

For more information about operations that update the master database and when to make backups, see Microsoft's SQL Server documentation.

Using the Microsoft SQL Server Backup Facility

An enterprise business solution must be able to manage a substantial amount of input and output activity every day. This increases the need to guard against information loss in case of database or hardware failure. Therefore, it is important that you implement a suitable backup procedure and that the system is set up so that the possibilities for data loss are minimized. If the system fails, you must be able to recover all of your data, including the data that has been modified since you made your last backup.

Applying Transaction Log Backups

The SQL Server Option allows you to use two different types of backup: Microsoft SQL Server backup and Microsoft Navision backup. We recommend that you use the backup facilities provided by SQL Server for your daily needs.

SQL Server uses a roll forward capability to recover all the committed transactions that were carried out up to the point of failure. Roll forward is achieved by restoring your last database backup and applying all subsequent transaction log backups to recreate these transactions.

In such cases, only uncommitted work (incomplete transactions) will be lost, provided the active transaction log is also backed up and applied. The active transaction log also contains details of all uncommitted transactions. When you apply the active transaction log backup, SQL Server will roll back the uncommitted transactions.

Losing the active transaction log will prevent the system from successfully applying all the transaction log backups. One way of protecting both the transaction log files and the data files against hardware failure is to place them on mirrored disks.

When you place the primary data file and the transaction log files on different physical disks than the data files containing the user objects, you ensure that any media failure on the disks containing the user database files affects only those files. You can further protect the files from isolated media failure by placing the primary data file and the transaction log files on mirrored disks.

For more information, consult your Microsoft Certified Business Solutions Partner or Microsoft's SQL Server documentation.

In order to apply transaction log backups, you must choose the correct options when you create your databases and implement suitable backup procedures. The database options that affect your ability to apply transaction log backups are the following:

- Select into/bulk copy
- Truncate log on checkpoint

Both of these should remain disabled to prevent SQL Server from truncating your log file and to ensure that the log file contains detailed information. These options are located on the **Options** tab of the Alter Database and New Database windows. For more information, see "Microsoft Navision SQL Server Option."

SQL Server Backups

Microsoft SQL Server supports four different types of backup. You should choose the type of backup you will be using carefully in order to ensure that you get the level of security you require.

The four types of backup are the following:

- Database backup – This makes a backup of the entire database.
- Transaction log backup – This makes a backup of the entire transaction log.
- Differential backup – This makes a backup of all committed entries since the last database backup.
- File and filegroup backup – This makes a backup of individual files or filegroups within a database.

These can be combined to form many different types of backup and restore procedures, thereby allowing you to make your backup and restore strategy fit your database needs.

For more information about SQL Server backup and restore strategies, consult Microsoft's SQL Server documentation.

The SQL Server backup/restore system is server-based and is, therefore, considerably faster than the Microsoft Navision backup/restore system, which is client-based.

It is possible to restore a SQL Server backup of a Microsoft Navision database directly into SQL Server without using Microsoft Navision. You can also create a database directly in SQL Server without first having to create it in Microsoft Navision and then restore a SQL Server backup of a Microsoft Navision database directly into the database on SQL Server.

SQL Server allows you to make backups when the system is in use. With SQL Server, you can also automate many of your administrative tasks, including making backups. SQL Server also allows you to establish a database maintenance plan (with the help of a wizard) that includes database optimization, integrity tests, and a backup plan.

SQL Server Tests

You should run SQL Server database consistency tests (using the SQL Server `dbcc` options) before making backups. SQL Server also allows you to include integrity tests in its backup procedure.

For more information about the backup facilities contained in Microsoft SQL Server and the different strategies that can be implemented, consult your Microsoft Certified Business Solutions Partner or Microsoft's SQL Server documentation.

Using the Microsoft Navision Backup Function

Whenever you create a new database, you must always restore a Microsoft Navision backup to retrieve the "Data Common to All Companies" and "Application Objects," and place them in the new, empty database. Data common to all companies includes the report list, permissions groups, user IDs and printer selections, but no real company data. For more information, see the section called "Constraints on Restoring a Backup."

When carrying out a Microsoft Navision backup in the SQL Server Option for Microsoft Navision, every object that is backed up gets locked and other users are given read-only access. This means that depending on what is being backed up, other users will not be able to work in the database.

***NOTE**: Selecting the Single user database option before you make or restore a Microsoft Navision backup will improve performance. Ensuring that you are the only user using the database means that SQL Server does not have to lock resources thereby improving performance. To do this click FILE→DATABASE→ALTER, and in the Alter Database window, select the **Options** tab, and then select the **Single user** field.*

Migrating to the SQL Server Option for Microsoft Navision

You must use the Microsoft Navision backup function when you want to migrate your data from Microsoft Navision to the SQL Server Option for Microsoft Navision.

Migrating involves making a Microsoft Navision backup of the database and then restoring it into an empty database in the new version of Microsoft Navision.

Before converting your earlier versions of Microsoft Navision databases to the SQL Server Option for Microsoft Navision, you should be aware of the following issues.

Security

If Windows authentication is being used, the SQL Server Option for Microsoft Navision requires that each user has the same ID in the Microsoft Navision Windows Login table and in her Windows account. If database server authentication is being used, the user's SQL Server login must be the same as their ID in the Microsoft Navision User table.

The synchronization of the security system is carried out at the end of the restore procedure. If the user IDs are not identical, then restoring the database will result in a message informing you that the security system is inconsistent and that you should run the synchronization process after these user ID problems have been corrected. The database will still be restored successfully. For more information about the synchronization process, see the section called Security and User Setup in "Microsoft Navision SQL Server Option."

To avoid this problem, make sure the user IDs are the same as the IDs used in the Windows or SQL Server accounts before you make a backup of the database in the previous version of the program. It is probably easier to change the IDs in the Microsoft Navision Windows Login table than to change the IDs used in the Windows accounts.

Transaction Log Size

Restoring a database increases the size of the existing SQL Server transaction log. The transaction log contains information about every change that has been made to the database, including information about newly created and modified records and logs of, for example, index creation operations. This means that the size of the transaction log can be increased by at least the amount of data restored into the database. Make sure that you have enough space for the enlarged transaction log.

Linked Objects

When you make a Microsoft Navision backup, the table descriptions of all the linked objects are also backed up but the table data is not backed up. When you restore a Microsoft Navision backup that contains linked objects, the SQL Server objects that the linked objects refer to must already exist in the database into which you are restoring the backup.

If the SQL Server database contains a view that refers to a Microsoft Navision table that is contained in the backup but does not yet exist in the database, you must create a dummy view that selects literal values for its columns before restoring the backup. These literal values must be defined in terms of the appropriate SQL data types. Creating the dummy view allows the linked objects to be successfully restored. After the database backup has been successfully restored, you can redefine this view so that it refers to the Microsoft Navision table that has now been restored into the database.

After successfully using Microsoft Navision to restore your database, you should make a SQL Server backup of the entire database. There are two reasons for this:

- When you make a SQL Server backup of the entire database, the transaction log will be truncated, thereby reducing the size of the transaction log that was created during the migration procedure.
- You will now have a SQL Server backup of your database that you can restore if it becomes necessary. This backup also serves as the starting point for subsequent transaction log backups. You will no longer have to use the Microsoft Navision restore facility, which is time-consuming and can create a large amount of data in the transaction log.

Table Relationships

If the synchronization of the table relationships fails when you are restoring a database backup, the backup will still be restored successfully. The only task that will not be completed is the synchronization of the table relationships.

Advanced Backup and Restore Information

The backup file is a compressed copy of the database. The backup program copies small blocks from the database, compresses them and gives them a header. Among other things, the header has a header checksum, which ensures data integrity. When the backup program has written all the small blocks, it calculates and writes a master checksum. These two checksums have two purposes:

- To verify that data in the backup is consistent and not corrupted
- To protect the backup from modifications

The Header in the Backup File

At the beginning of the backup file, you will find a header in ASCII format containing a list of all the objects in the file. To see a list of the objects included in the backup (for example, named backup.fbk and located in the folder `d:\backup\`), go to a command prompt and type the following:

```
type backup.fbk | more
```

The backup file contains an end of file marker after the list, so you do not see the binary content of the file. For each object, the list will show you its type, number, name, date and time of creation, size in bytes, and a version number. If the object is table data, the version number is replaced with the name of the company to which the data belongs. This list is useful for seeing which objects are included in the backup, without restoring the entire backup.

Constraints on Restoring a Backup

There are certain constraints when you restore a backup into a non-empty database. When you create an empty database, it is actually not completely empty. An empty database is defined as a database that contains only an empty security system and the company table definition. You can define and limit user access permissions to the database in the security system. An empty security system is identical to the security system that is automatically generated when you generate an empty database in Microsoft Navision. An empty security system consists of the tables listed below:

Table	Defines…
User Group	the user groups that you want in your system
Permissions	which rights the different user groups have
User	the individual users and their passwords
Member Of	which groups the individual users belong to

Of the tables here, only the User Group and Permission tables, where a superuser has been set up, hold data. If you change the contents of the four security tables, the security system is no longer empty.

The contents of the backup can be divided into the following data types:

- Application Objects, which form the application, such as the Customer table, the Item table, and the G/L Account table
- Data Common to All Companies, which includes the report list, permissions groups, user IDs, printer selections, and the security system tables
- Company Data, which is all the data in the tables

These three data types can be restored individually from different backups. To restore a backup from three different backups into an empty database:

1. Restore the objects of the type Application Objects from one backup.
2. Restore the objects of the type Data Common to All Companies from another backup.
3. Restore Selected Companies from yet a third backup.

The following table contains answers to questions or problems that may arise during the restore process:

Question or Problem	Answer
Can I restore everything into an empty database?	All data types can be restored into an empty database.
Can I restore objects of the type Application Objects into a nonempty database?	No. Objects of the type Application Objects can only be restored into an empty database.
I cannot restore my company into the database.	You cannot restore the data of a company if there is already another company with the same name in the database. If, however, you want to restore the company that is in the backup, rename the existing company in the database, and then restore the company in the backup.
I cannot restore objects of the type Data Common to All Companies.	If the database contains objects of the type Data Common to All Companies other than the security system tables, it is not possible to restore objects of type Data Common to All Companies.
I am prompted to confirm overwriting of the existing security system while restoring. What does that mean?	If all objects of the type Data Common to All Companies in the database are security system tables, then all objects of the type Data Common to All Companies can be restored. If neither the security system in the database nor the security system in the backup is empty, you will be prompted to confirm whether the existing security system in the database should be overwritten by the security system from the backup.

Question or Problem	Answer
When a database has been completely restored, I receive an error message telling me that the Microsoft Navision and SQL Server security systems have not been synchronized successfully.	Ensure that the user IDs in the Microsoft Navision User table match either the Windows accounts or the SQL Server logins. When you have made the necessary changes, initiate the synchronization process by clicking TOOLS→SECURITY→SYNCHRONIZE. For more information on this topic, see the "Microsoft Navision SQL Server Option."
While I am restoring a database, I receive a message asking whether or not I would like to link a Microsoft Navision table to a SQL Server object. What does this mean?	This means that the database you are restoring contains a table that has the same name as an existing SQL Server object. Linking them means that Microsoft Navision will not create a table in SQL Server but will use the existing object instead and will start entering data from the backup into this SQL Server object. To avoid this, cancel the restore procedure and rename or delete the SQL Server object before beginning the restore process again.

Changed Table Definition

If a field in a table definition has changed in type or number in relation to the backup, it is not possible to restore data into the corresponding table. If you have changed some of the properties of a field in a table definition, the restore program will attempt to fit the data into the table anyway. If, for example, you have shortened the length of a text field from 80 to 40 characters, and there is no data in that specific table in the backup that is longer than 40 characters, the backup will be successfully restored. Otherwise the restore procedure will stop, and you will have to increase the length of that particular field. After that, you can continue the restore process from where it stopped.

Maintaining and Updating Statistics

When you create an index (known as a key in Microsoft Navision), SQL Server automatically generates and stores statistical information about the distribution of values in the indexed column(s). The query optimizer in SQL Server uses these statistics to estimate the cost of using the index for a query.

This means that every time Microsoft Navision applies a filter to a table, SQL Server uses this statistical information to generate the most effective query that can be used to retrieve the information from the corresponding SQL Server tables. These statistics are therefore vitally important to the performance of the SQL Server Option for Microsoft Navision.

However, as the data in the columns changes, the index and column statistics can become out-of-date and this results in the query optimizer making less-than-optimal decisions on how to process a query. For example, if you create a table with an indexed column and 1,000 rows of data, all with unique values in the indexed column, the query optimizer considers the indexed column the best way to collect the data for a query. If you update the data in the column so that it contains many duplicate values, the column is no longer an ideal candidate for use in a query. However, the query optimizer still considers it to be a good candidate based on the index's outdated distribution statistics, which are based on the data before the update.

SQL Server automatically updates this statistical information periodically as the data in the tables changes. The frequency with which the statistical information is updated is determined by the volume of data in the column or index and the amount of changing data. For example, the statistics for a table containing 10,000 rows may need updating when 1,000 index values have changed because 1,000 values may represent a significant percentage of the table. However, for a table containing 10 million index entries, 1,000 changing index values is less significant, and so the statistics may not be automatically updated.

SQL Server minimizes the cost of this automatic statistical update by analyzing samples of the data, rather than analyzing all of it. SQL Server, however, always ensures that a minimum number of rows are sampled; tables that are smaller than eight MB are always fully scanned to gather statistics.

Under some circumstances, statistical sampling will not be able to accurately characterize the data in a table. You can control the amount of data that is sampled during manual statistical updates on a table-by-table basis by using the SAMPLE and FULLSCAN clauses of the UPDATE STATISTICS statement. The FULLSCAN clause specifies that all of the data in the table is scanned to gather statistics, whereas the SAMPLE clause can be used to specify either the percentage of rows to sample or the number of rows to sample.

For more information about generating and updating statistics, see Microsoft's SQL Server documentation.

Maintaining Statistics for Microsoft Navision Databases

When you create a Microsoft Navision database on SQL Server, this statistical information is created automatically. In order for the SQL Server Option for Microsoft Navision to function optimally, you must update these statistics regularly.

To update the statistics:

1. Open the SQL Query Analyzer.
2. Select the database that contains the statistics you want to update.

3. Run one of the following queries:

   `update statistics [table name]` – to update the statistics for a single table.
   `sp_updatestats` – to update the statistics for all the tables.

Updating the statistics for Microsoft Navision tables can be a time consuming task depending on the number of records that the tables contain. The tables that are updated most often are the tables whose statistics must be updated most regularly. Therefore, we recommend that you create a job on SQL Server that performs regular updates of all the tables when the system is not in use. If performing the update is still too time consuming, you can divide it into smaller jobs that update the statistics for some of the tables. Creating a SQL job allows you to automate the task and generate reports containing details about the success of the job.

For more information, see Microsoft's documentation.

Test Your Knowledge – Microsoft Navision Database Administration Tools

1. Microsoft SQL Server supports four different types of backups: _____, _____, _____ and _____.
2. If the _____ of the table relationships fails when you are restoring a database backup, the backup will be _____.
3. You must use the _____ function when you want to migrate your data from Microsoft Navision to the SQL Server Option for Microsoft Navision.
4. Server Based Backups are called _____ and are used with the _____ _____ _____.
5. The three types of backup options available are _____ _____ _____.
6. To safeguard against errors, you should frequently check the _____ and _____ of your databases.
7. When an index (key) is created, SQL Server automatically generates and stores _____ _____ about the distribution of values in the indexed column(s).
8. A recommended backup procedure includes backing up the database _____ _____.
9. _____ _____ is achieved by restoring your last database backup and applying all subsequent _____ _____ _____ to recreate these transactions.
10. _____ and _____ _____ are sometimes described as indexes and can be used for sorting information.

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 11: MICROSOFT NAVISION OPTIMIZATION

Overview

In a multiuser installation, Microsoft® Business Solutions–Navision® handles the problem of several users accessing the database concurrently (at the same time) with a technique called optimistic concurrency, which uses the database version principle. This chapter contains information on the Microsoft Navision database structure, which is key to understanding how tables are optimized in Microsoft Navision.

This chapter then covers the process of optimizing tables, using keys and key groups to improve performance as well as creating space in the database using the Date Compression functionality.

And finally, this chapter contains guidelines on Administering the SQL Server that are used to optimize performance, such as SQL Profiler, Trace Logs, and System Monitor.

Advanced Database Information

In order to understand how a table is optimized in Microsoft Navision, you need to understand how the database is constructed.

Optimizing a Table

A database table contains a primary key and several secondary keys. There must be a primary key, and there can be a maximum of 40 keys (both primary and secondary keys) in all. The keys determine how data is sorted.

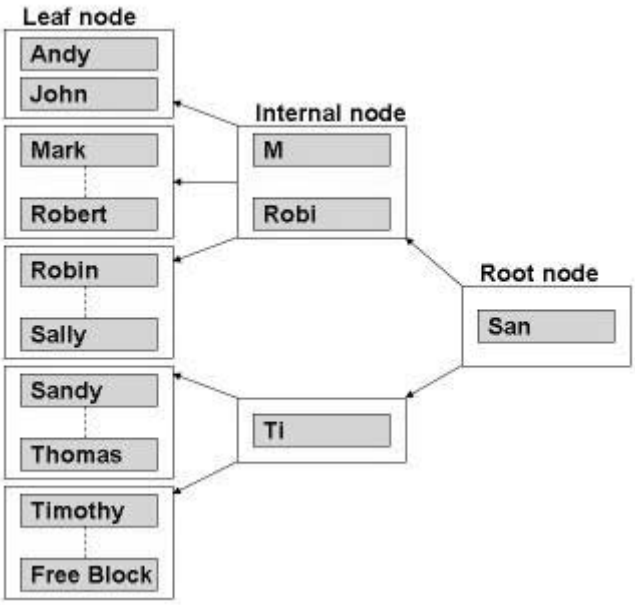
Keys

The keys are arranged in so-called B+ trees (balanced trees). A balanced tree is one in which the program always passes through the same number of levels to access data at the bottom of the tree, regardless of which branch of the tree the data is located on.

A tree data structure is made up of nodes (see the figure on the next page) where each node in the tree, except for a special node called the root, has one parent node and may have one or more child nodes. The root node has no parent. A node that has no child node is called a leaf; a nonleaf node is called an internal node.

The level of a node is defined as the level of its parent plus one. The level of the root node is defined as zero. Searching for a record in a B+ tree is very fast because it is not necessary to search the entire tree to find the record.

B+ Tree



Data is only found in the leaf nodes, not in the internal nodes. The internal nodes and the root node only contain dividers (San, M, Robi and Ti in the diagram) and pointers (arrows in the diagram) to the next nodes in the tree.

A divider contains information that separates the data in the levels below it. Depending on whether the value of the data you are searching for in the level below is smaller than or greater than the value in the divider, the search path goes either to the left or to the right (up or down in the figure).

For example, consider the search for the name Sandy. Sandy comes after San in alphabetical order, so the search path would go from the root node to the internal node containing the divider labeled *Ti.* Sandy comes before Ti, so the search would go to the left of the divider to the leaf node containing Sandy *(*up in the diagram).

One internal node can contain several dividers. This minimizes the number of levels in the tree and ensures that the path to the record is as short as possible, and the tree can be said to be balanced.

As you insert more data in the tree, the leaf nodes may be only partly filled. When you optimize a table in Microsoft Navision, the leaf nodes are packed together in order to save space in the tree. When more information is inserted, data will be moved to new nodes to make room for the insertion.

How the Database Works

What if two or more users are reading the same record at the same time? And what if two or more users working on the same record try to save the record at different points in time? Microsoft Navision uses several strategies to deal with these problems. They are explained in the following sections.

Optimistic Concurrency

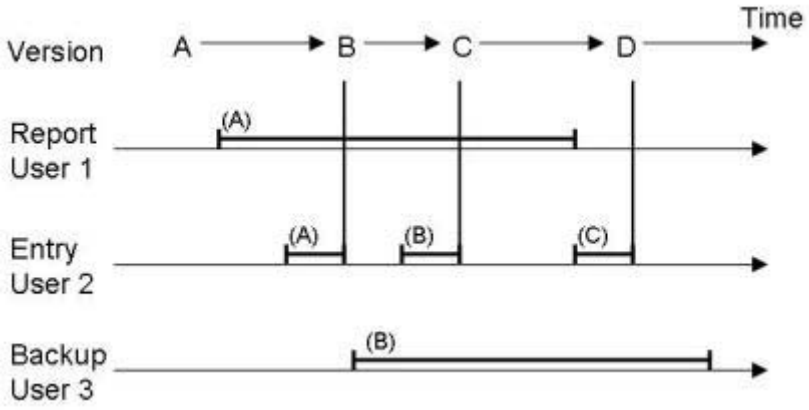
Microsoft Navision uses a technique called optimistic concurrency. With optimistic concurrency, you always have access to the record you want to work with. Thus, if two or more users try to access the database at the same time, they will all be allowed to do so. This is possible because the Microsoft Navision database is based on the database-version principle. Every time a user commits something to the database, a new version of the database is created. (A transaction is said to be committed when it is physically written to the database or to commit cache – if it is turned on. See the section called "Commit Cache – Writing the Cache".)

When you enter new data, your changes are private. It is not until you commit the changes that the new data becomes public, and the newest version of the database is established. The DBMS (database management system) enables several Microsoft Navision clients to access and modify the database concurrently by letting them work on individual versions, which are merely snapshots of the database at the time when they first accessed it.

Read Consistency and Concurrency

The following figure shows three Microsoft Navision clients accessing the same database. Imagine that the first access is made by a report. The second access is made by a user who inserts new entries in the database, and the third access is made by a backup procedure.

Optimistic Concurrency



The generation of the report is based on version A of the database. It is a time consuming process, and while the report is being generated, User 2 enters or modifies records in the database. After each entry is committed, a new version of the database is created. These are versions B, C, and D. However, a snapshot of the database (version A) was made when the report started, and the report continues to work on version A of the database. User 3 starts a backup procedure. When this process starts, the most recent version of the database is B, created by the completion of the first entry by User 2. A snapshot of this version is made, and this version (B) is backed up. The backup process is unaffected by User 2, who continues to enter new data during the backup process. This example shows that working with data versions makes it possible for many users to access the database without interfering with each other.

However, if two or more users access the same record at the same time the user who first tries to write the changes to the record will be allowed to do so. The other user(s) will receive the following message from the system when they try to commit their changes to the database (if they are trying to modify customer number 10000):

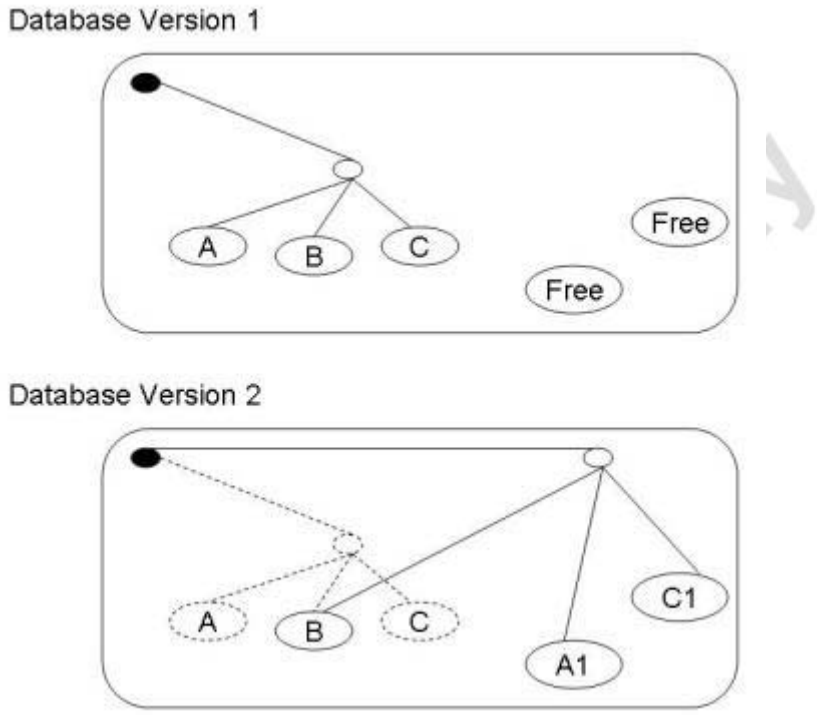
In this case, the user has to repeat his or her work using a new snapshot of the latest (modified) version of the database.

The implications of the database-version principle approach are many. Most important is that different Microsoft Navision clients may read different versions of the same database. These versions are snapshots of the database at the point in time when the clients start to access the database. In this way, the DBMS allows for concurrency (that is, for more than one user to have access to the database at the same time) while still maintaining read consistency. If the accesses to the database involve only data retrieval and no changes, then the newest version will persist – for all clients – until a write transaction is performed.

Database Versions

The following figure illustrates the concept of database versions. The data in the Microsoft Navision database is stored in the data structure mentioned previously, the B+ tree. Imagine that the tree structure in our database contains a branch with customers A, B and C. Furthermore, there are two free database blocks available. Assume that you need to modify customers A and C. When you modify the records, the DBMS makes a copy of the original customers A and C from database version 1. These copies use two free database blocks. You then perform your modification on the copies (they are now named A1 and C1), and the system creates a new internal node.

Data Versions



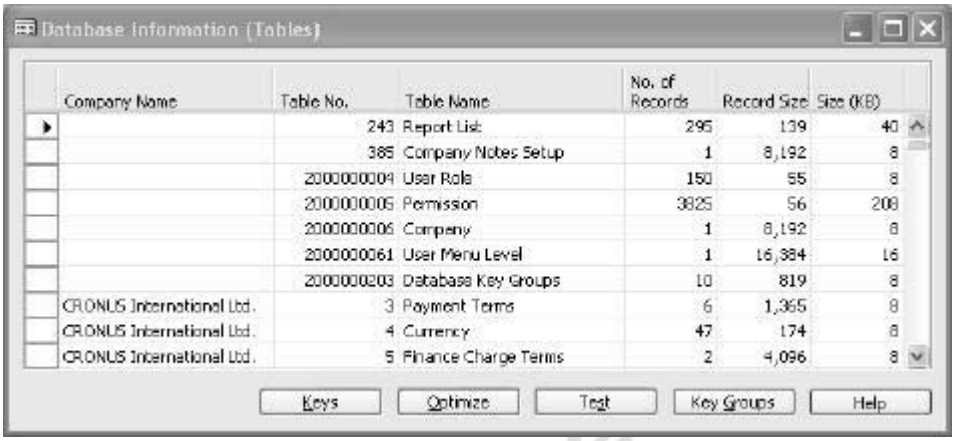
If an error occurs during the transaction, or the user decides to abort the changes, the database blocks occupied by the copied branch will be released and be available for new database updates.

If the transaction is committed, this new internal node will replace the old node, and the database blocks used by the old versions of customer A and C will now be available as free database blocks which can be used by database updates. Now, the newest database version is version 2 in the figure.

Database Efficiency

After you have been using Microsoft Navision for a while, it is a good idea to check how effectively the database is being utilized. There are a number of special tools for doing this.

Click FILE→DATABASE→INFORMATION, and then click **Tables** at the bottom of the Database Information window. A list of all the tables in the database appears:



All the information in Microsoft Navision is organized in tables. Each line in the window represents one table. Each line displays the following information:

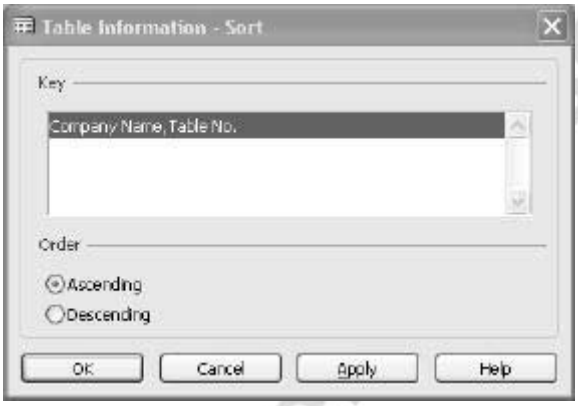
Field	Comment
Company Name	The name of the company to which the table on the line belongs.
Table No.	The number of the table. (Each table in Microsoft Navision has a unique number.)
Table Name	The name of the table. This is the name that is used in C/SIDE rather than the name of the window that displays the table on the screen (they are often the same).
No. of Records	The number of records (entries) in the table. By keeping track of how many new records have been added in a certain period, you can estimate the number of records you can expect in the next period.
Record Size	The average number of bytes per record in a table. Combining this with the number of records in the **No. of Records** field enables you to estimate whether you will have enough space in your database.
Size (KB)	The total size of the table, in kilobytes.
Optimization (Database Server)	This is a percentage that tells how effectively Microsoft Navision uses this table in the database. The higher the percentage, the more effectively the table is utilized. To optimize the location of data in the table, click **Optimize** at the bottom of the window. (This column is not shown in the picture.)

The values displayed in the Database Information (Tables) window cover all the information related to the individual tables in Microsoft Navision, both financial information and keys. The next section explains how to use this information to improve the utilization of the database.

***NOTE**: For the SQL Server Option, Linked objects are not included in the table list.*

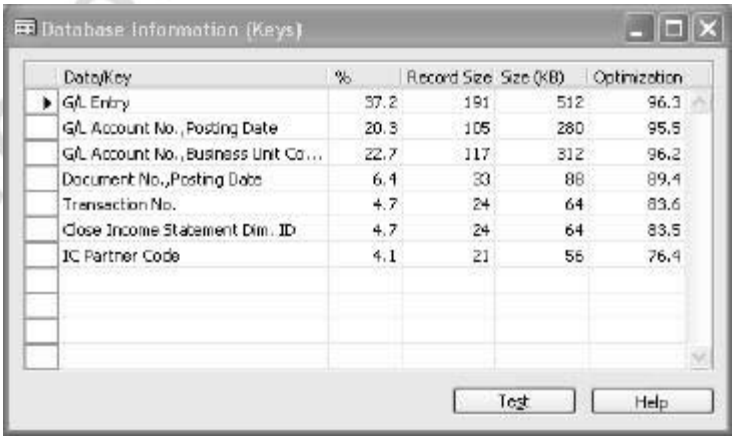
Using Keys and Key Groups to Improve Performance

To make the information in the tables as useful as possible, many of the tables have several predefined sorting keys. To view these predefined sorting keys, click FILE→DATABASE→INFORMATION→TABLES. Select a table, and click VIEW→SORT on the menu bar. The Table Information - Sort window appears:



You can change the sorting order for the table.

For information about the keys, click FILE→DATABASE→INFORMATION→TABLES in the Database Information window. In the Database Information (Tables) window, place the cursor on the table whose keys you want to see and click **Keys**. The Database Information (Keys) window appears:



Keys have several uses, but maintaining them (if, for example, one is renamed) places a large demand on the system's resources. Therefore, you can improve database utilization and performance by deactivating or completely removing keys that are not used in your installation. However, you should not remove or deactivate keys without giving it some thought because there are certain functions the program cannot run when certain keys are missing.

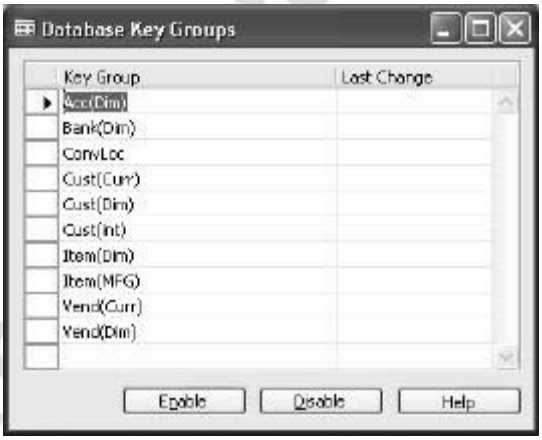
Because of this, Microsoft Navision does not allow you to change the status of keys unless you have access to the C/SIDE development environment.

If you do not have access to C/SIDE, you can have your Microsoft Navision Certified Business Solutions Partner create a set of keys that makes a particular type of task easier, for example, running a lengthy report.

These keys can be set up as a key group, which you can enable and disable without risk. The program will perform better when you have disabled the key group because the system does not have to maintain the keys included in the key group.

Adding a large number of keys to database tables decreases performance. However, by making the keys members of predefined key groups, you can have the necessary keys defined and only activate them when they are going to be used.

To enable or disable the key groups, click FILE→DATABASE→INFORMATION→TABLES→KEY GROUPS. The Database Key Groups window appears:



Select the key group names, and click **Enable** or **Disable**. You must remember the names of the key groups.

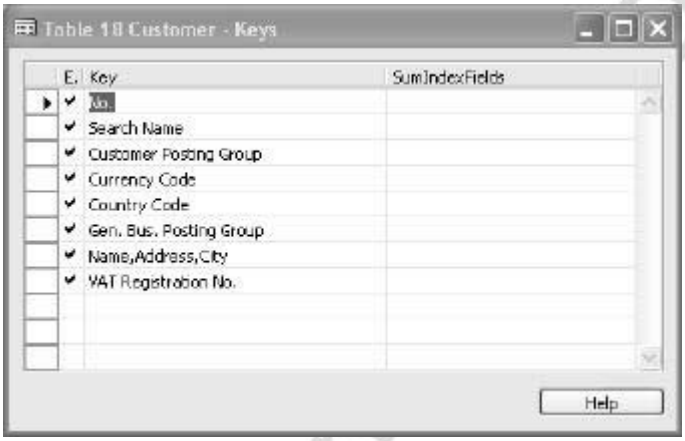
Setting Up a Key Group

To set up key groups (only possible if you have access to C/SIDE), you must specify which key group each key belongs to. The keys are found in the definitions of the tables.

To set up a key group:

1. Click TOOLS→OBJECT DESIGNER→TABLE.
2. Select the table that contains the key you want to include in the key group.
3. Click **Design**.
4. On the menu bar, click VIEW→KEYS. You can now see the keys for the selected table.

This window shows the keys for the Customer table:



Disabling a Key Group

If you do not need to sort the Customer table by the Currency Code key very often, you could put the Currency Code key in a key group called, for example, RarelyUsed and disable it. The name of the key group must not be longer than ten characters.

To disable the key:

1. In the Keys window (as shown above), select the line containing Currency Code.
2. On the menu bar, click VIEW→PROPERTIES.
3. In the Properties window, enter RarelyUsed for the Key Groups property.
4. Save the table definitions by closing the Table Designer.
5. Click FILE→DATABASE→TABLES.
6. Select the Customer table, and click **Key Groups**.

7. To disable the key group RarelyUsed, type RarelyUsed in the **Key Group** field.
8. Click **Disable**.

Now, if you select SALES & MARKETING→SALES→CUSTOMERS, and then on the menu bar, click the VIEW→SORT, you are no longer able to sort the Customer table by currency code.

***NOTE**: When you re-enable keys that have been disabled, the program performs a sorting procedure that requires a certain amount of free space in the database.*

Decreasing the Table Seek Time (Database Server)

As stated earlier, the Microsoft Navision database is organized in a tree structure. As this tree expands with new records, these records are distributed at the bottom level of the tree. The tree is designed so that data is placed only at the bottom level.

It is possible to decrease the seek time in the tree by optimizing the table in question. Optimizing the table will make finding the specific record quicker by removing any empty space in the table and possibly minimizing the number of levels in the tree.

To minimize the number of levels in the tree for a particular table, select the table in the Database Information (Tables) window and click **Optimize**.

On the other hand, if you often insert new records in the table, the program inserts the record in its proper place, between all the other records, by moving all the other records to make room for the new record. So inserting records in a table that is often used can take longer if its tree structure has been optimized and the empty space removed. You can read more about the database structure in the section titled “Advanced Database Information.”

Key Properties (SQL Server Option)

The SQL Server Option for Microsoft Navision contains some key properties that set relationships between Microsoft Navision keys and SQL Server indexes.

MaintainSQLIndex

This key property determines whether a SQL Server index is created (when the property is set to Yes) or dropped (when the property is set to No) for the corresponding Microsoft Navision key.

A Microsoft Navision key allows data in a table to be sorted by the key fields. However, SQL Server does not need an index in order to sort by particular fields. If an index exists, sorting by the fields matching the index will be faster, but modifications to the table will be slower. The more indexes there are on a table, the slower the modifications will be.

You can disable the MaintainSQLIndex key property for Microsoft Navision keys that are only used occasionally (for example, when running infrequent reports). This will prevent modifications to the table from being too slow.

NOTE: *You cannot disable this property for the primary key of a table. The index corresponding to the primary key is always created in SQL Server.*

MaintainSIFTIndex

This key property determines whether SIFT structures are created (when the property is set to Yes) or dropped (when the property is set to No) in SQL Server to support the corresponding SumIndexField® for the Microsoft Navision key.

SumIndexFields are created in Microsoft Navision to support, for example, FlowField® calculations and other fast summing operations. SQL Server can sum numeric data by scanning the table. If the SIFT structures exist for the SumIndexFields, summing the fields will be faster, especially for large sets of records. On the other hand, modifications to the table will be slower because the SIFT structures must also be maintained.

There are situations where SumIndexFields must be created on a key to allow FlowField calculations, but the calculations are performed infrequently or on small sets of data. In these situations, you can disable this property to prevent modifications to the table from being too slow.

Optimization

To initiate the optimization process, click FILE→DATABASE→INFORMATION→TABLES→OPTIMIZE.

The optimization process performs two functions:

- For each Microsoft Navision table, the SQL Server indexes, other than the primary key, are rebuilt to optimize their layout and usage.
- For each SIFT structure, any entries that contain zero values in all numeric fields are removed. The removal of these redundant entries will free space and provide more efficient updating and summing up of SIFT information.

Finding Errors in the Tables

If you suspect that there are errors in some tables, you can perform error tests.

To perform a test on selected tables, click FILE→DATABASE→INFORMATION→TABLES. Then highlight the tables to be tested and click **Test**. The Test Tables window appears:

If you do not select any tables, the test will be carried out on all the tables.

From this window, you can activate the same tests (except the space allocation test) that you can activate by using the DB Test program property or by clicking FILE→DATABASE→TEST. If you choose to test only one table and select, for example, "Test field relationships between tables," the relationships between the selected table and the related tables will be tested.

NOTE: *The Test relationships between primary and secondary keys option is not available in the SQL Server Option.*

Creating Space in the Database

The number of entries in the system will grow as you continue to use Microsoft Navision. Therefore, at some point you must choose between expanding the database or combining some of the old entries, so they take up less space.

Date Compression

Combining entries is called compression. Although you can only compress entries from closed fiscal years, you can compress them more than once.

NOTE: *Date compression removes information from the entries, so you must always make a backup of the database before you run the function.*

What Is Combined?

During the compression process, several old entries are combined into one new entry. For example, G/L entries from previous fiscal years can be compressed so that there is only one positive and one negative entry per account per month (one each for debit and credit).

After the compression process has been carried out, certain fields will always remain in the combined new entry. The contents of each of these fields will be set to a common or total value. In G/L accounts, for example, these are Posting Date, G/L Account No., Gen. Bus. Posting Group, Gen. Prod. Posting Group, Gen. Posting Type, Amount, and VAT Amount. The contents of the other fields will be cleared by the date compression.

You can also select other fields to be retained. If you select more fields to be retained, this will result in more new, compressed entries per time period. The fields that you can select are listed in the window that appears when you choose the Date Compression batch job.

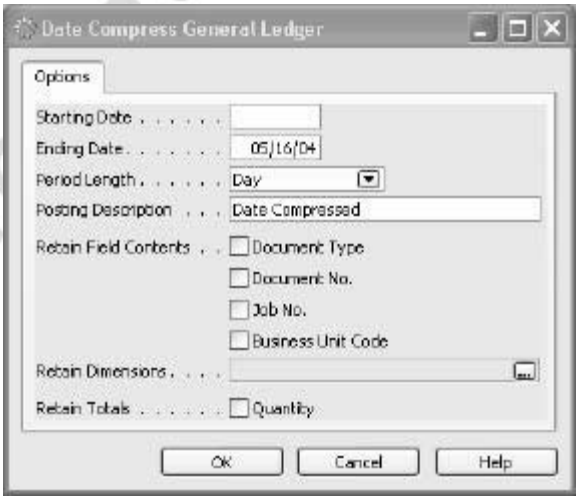
What Is the Result?

The number of entries that remain after the date compression process has been completed depends on how many filters you set before you start the batch job, which fields you want to have combined, and the compression period you select. There will always be at least one combined entry. When the batch job is finished, the result is displayed in the date compression register in the area of the program where you ran the batch job.

The amount of a date-compressed entry is the sum of all the entries that have been compressed into it. The date is set at the starting date of the compression period, for example, the first day of the month if you are compressing by month. After compression has taken place, you can still see the net change in each account for each compression period.

Starting Date Compression

There are date compression batch jobs for each type of entry that can be created in Microsoft Navision. The batch jobs are found in the appropriate application areas on the main menu. For example, to find the batch job for date compression of G/L entries, click FINANCE→GENERAL LEDGER→PERIODIC ACTIVITIES→DATE COMPRESSION→GENERAL LEDGER. If you click the batch job, the following window appears:



NOTE: *When you choose other date compression batch jobs (other than G/L Entries), a tab appears containing filters that you can set on the entries before they are removed for compression. You can use this facility if you only want entries with particular values in a field to be included in the compression.*

The **Options** tab contains the following fields in which you can set up the conditions for the compression:

Field	**Comments**	
Starting Date	Enter the first date you want to include in the compression. The compression will include all (for example, G/L) entries from that date until the date in the **Ending Date** field. If you do not enter a date, the compression will begin with the first posting date in the program.	
Ending Date	Enter the last date you want to include in the compression. All (for example, G/L) entries from the starting date through this date will be compressed.	
Period Length	Enter the time period over which you want to combine entries. To see the options, click the **AssistButton**:	
	Options	**Entries Combined**
	Day	From the same posting date
	Week	From the same week (only entries with a common month and accounting period)
	Month	From the same month (only entries with a common accounting period)
	Year	From the same year (only entries with a common accounting period)
	Quarter	From the same quarter (only entries with a common accounting period)
	Year	From the same fiscal year and calendar year
	Period	From the same accounting period (grouped within the same calendar year)
Posting Description	Enter the description that will accompany the entry or entries created by the compression. Date Compressed is suggested as a default.	
Retain Field Contents	Enter a check mark by the fields whose contents you want to save. Selecting a field here means that when a group of entries for a period is compressed, a common or total value for the field will be retained in the combined entry. Thus, you will still have all the information about this field for each compression period. The more fields you select here, the more detailed the information is in each compressed entry.	

Field	Comments
Retain Dimensions	Click the **AssistButton** and the Dimension Selection window appears. In this window select the dimensions that you want to retain in the compressed entries.
Retain Totals	Enter a check mark if you want each combined entry to include the total contents of the **Quantity** field. This option is available for G/L entries to allow you to retain the totals if you want to. In many cases, this total will be meaningless because the **Quantity** field is used in various contexts. However, you may want to retain the totals if you have used the field when registering purchases, for example, so you can choose to have it included in the combined entry.

NOTE: *If you select Day, Week, Month, Quarter, or Period as the time period, you will later be able to compile various statistics about the compressed entries by period.*

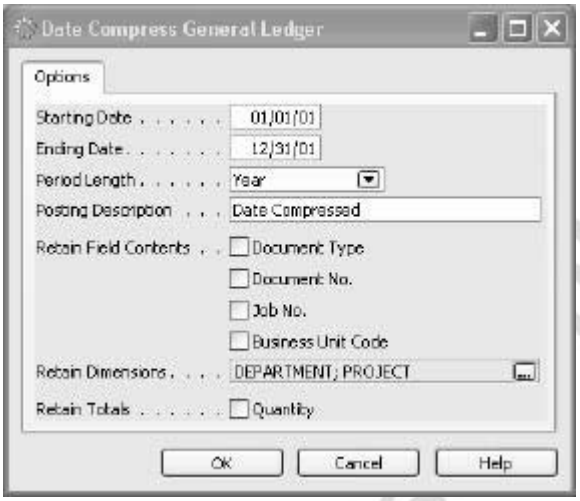
Example of Date Compression

In the following scenario, a company has used Microsoft Navision since January 1, 1995 for five complete fiscal years that follow the calendar year. The company is now in the middle of the sixth fiscal year (2000). Until now, no entries have been deleted or compressed, but the company believes it no longer needs to have a complete historical record of everything that has happened.

Here is a suggested compression strategy for G/L entries:

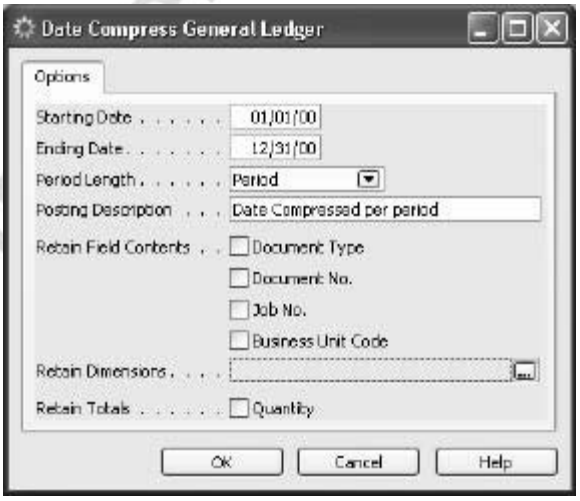
1. In the present fiscal year (Fiscal Year 6), save all entries (no date compression at all).
2. In the preceding fiscal year (Fiscal Year 5), create one combined transaction per account per day per department per project. It will still be possible to create statistics for each day based on department and project.

3. Fill in the fields in the window as shown here:

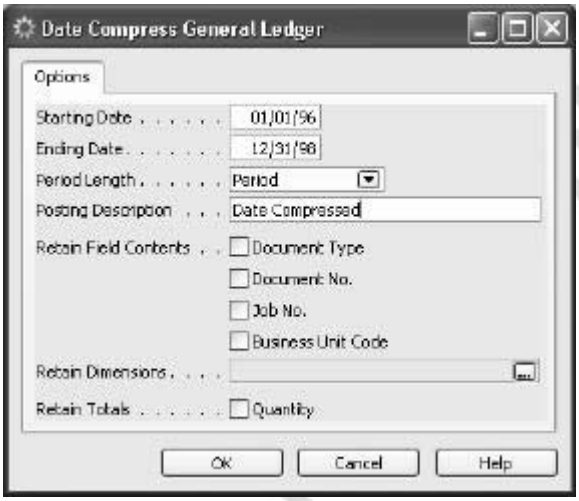


4. For the fiscal year before that one (Fiscal Year 4), create one combined transaction per account per accounting period. It is not necessary to preserve information about departments and projects. From now on, the statistics for this fiscal year can be printed only on the basis of period, without reference to departments and projects.

   Fill in the window as shown here:



5. For fiscal years that are more than three years old, only one combined entry per account per fiscal year needs to be saved. For the three oldest fiscal years, you will be able to generate statistical information based only on totals for entire years. Fill in the window as shown here:



Administering SQL Server

Monitoring with SQL Profiler

SQL Profiler is a graphical tool that allows system administrators to monitor events in an instance of Microsoft® SQL Server®. You can capture and save data about each event to a file or SQL Server table to analyze later. For example, you can monitor a production environment to see which stored procedures are hampering performance by executing too slowly.

Use SQL Profiler to monitor only the events in which you are interested. If traces are becoming too large, you can filter them based on the information you want, so that only a subset of the event data is collected. Monitoring too many events adds overhead to the server and the monitoring process and can cause the trace file or trace table to grow very large, especially when the monitoring process takes place over a long period of time.

After you have traced events, SQL Profiler allows captured event data to be replayed against an instance of SQL Server, thereby effectively re-executing the saved events as they occurred originally.

Use SQL Profiler to:

- Monitor the performance of an instance of SQL Server.
- Debug Transact-SQL statements and stored procedures.
- Identify slow-executing queries.
- Test SQL statements and stored procedures in the development phase of a project by single-stepping through statements to confirm that the code works as expected.
- Troubleshoot problems in SQL Server by capturing events on a production system and replaying them on a test system. This is useful for testing or debugging purposes and allows users to continue using the production system without interference.
- Audit and review activity that occurred on an instance of SQL Server. This allows a security administrator to review any of the auditing events, including the success and failure of a login attempt and the success and failure of permissions in accessing statements and objects.

SQL Profiler provides a graphical user interface to a set of stored procedures that can be used to monitor an instance of SQL Server. For example, it is possible to create your own application that uses SQL Profiler stored procedures to monitor SQL Server.

You must have at least 10 megabytes (MB) of free space to run SQL Profiler. If free space drops below 10 MB while you are using SQL Profiler, all SQL Profiler functions will stop.

Starting SQL Profiler

SQL Profiler is started from the Microsoft® Windows NT® 4.0, Microsoft® Windows® 2000 or Microsoft® Windows® 98 Start menu, or from SQL Server Enterprise Manager.

With Windows Authentication mode, the user account that runs SQL Profiler must be granted permission to connect to an instance of SQL Server.

To start SQL Profiler from Enterprise Manager:

1. From the Start menu, select PROGRAM FILES→MICROSOFT NAVISION SQL SERVER→ENTERPRISE MANAGER.
2. On the Tools menu, click **SQL Profiler**.

Reporting Services Trace Logs

Trace logs contain highly detailed information about report server operations. Trace logs include redundant information that is recorded in other log files, plus additional information that is not otherwise available. Trace logs are useful if you are debugging an application that includes a report server or investigating a specific problem that was written to the event log or execution log.

Finding Trace Logs

Reporting Services provides three trace log files, which are located at `\Microsoft Navision SQL Server\<SQL Server Instance>\Reporting Services\LogFiles`. They are described in the following table.

Log file name	Description
ReportServerService_<timestamp>.log	Trace log for the Report Server Windows service and Web service.
ReportServerWebApp_<timestamp>.log	Trace log for Report Manager.
ReportServer_<timestamp>.log	Trace log for the report server engine.

Viewing Log Information

Trace logs are text files. You can use any text editor to view a log. The following information can be found in a trace log:

- System information, including operating system, version, number of processors, and memory.
- Reporting Services component and version information.
- Events logged the Application log.
- Exceptions generated by the report server.
- Low resource warnings logged by a report server.
- Inbound SOAP envelopes and summarized outbound SOAP envelopes.
- HTTP header, stack trace, and debug trace information.

You can review trace logs to determine whether a report delivery occurred, who received the report, and how many delivery attempts were made. Trace logs also record report execution activity and the environment variables that are in effect during report processing. Errors and exceptions are also entered into trace logs. For example, you may find report timeout errors (indicated as a ThreadAbortExceptions entry).

Managing Log File Content

Trace logs are created daily, starting with the first entry that occurs after midnight (local time). The local time of the computer is used for naming the trace log. By default, trace logs are deleted after fourteen days. You can determine how long the files are kept by setting the KeepFilesForDays configuration setting.

Profiling ODBC Driver Performance

The SQL Server ODBC driver can profile two types of performance data:

- Long-running queries.
- The driver can write to a log file any query that does not get a response from the server within a specified amount of time. Application programmers or database administrators can then research each logged SQL statement to determine how they can improve its performance.
- Driver-performance data.
- The driver can record performance statistics and either write them to a file or make them available to an application through a driver-specific data structure named SQLPERF. The file containing the performance statistics is a tab-delimited file that can be easily analyzed with any spreadsheet that supports tab-delimited files, such as Microsoft® Excel.

Either type of profiling can be turned on by:

- Connecting to a data source that specifies logging.
- Calling SQLSetConnectAttr to set driver-specific attributes that control profiling.

Each application process gets its own copy of the SQL Server ODBC driver, and profiling is global to the combination of a driver copy and an application process. When anything in the application turns on profiling, profiling records information for all connections active in the driver from that application. Even connections that did not specifically call for profiling are included.

After the driver has opened a profiling log (either the performance data or long-running query log), it does not close the log until the driver is unloaded by the ODBC Driver Manager, when an application frees all the environment handles it opened in the driver. If the application opens a new environment handle, a new copy of the driver is loaded. If the application then either connects to a data source that specifies the same log file or sets the driver-specific attributes to log to the same file, the driver overwrites the old log.

If an application starts profiling to a log file and a second application attempts to start profiling to the same log file, the second application is not able to log any profiling data. If the second application starts profiling after the first application has unloaded its driver, the second application overwrites the log file from the first application.

If an application connects to a data source that has profiling enabled, the driver returns SQL_ERROR if the application calls SQLSetConnectOption to start logging. A call to SQLGetDiagRec then returns:

```
SQLState: 01000, pfNative = 0

ErrorMsg: [Microsoft Navision][ODBC SQL Server Driver]

      An error has occurred during the attempt to

      access the log file, logging disabled.
```

The driver stops gathering performance data when an environment handle is closed. If an ODBC 3.x application has multiple connections, each with its own environment handle, then the driver will stop gathering performance data when any of the associated environment handles are closed.

The driver's performance data can either be stored in the SQLPERF data structure or logged in a tab-delimited file. The data includes the following categories of statistics:

- Application profile
- Connection
- Network
- Time

In the following table, the descriptions of the fields in the SQLPERF data structure also apply to the statistics recorded in the performance log file.

SQLPERF Field	Description
Application Profile Statistics	
TimerResolution	Minimum resolution of the server's clock time in milliseconds. This is usually reported as 0 (zero) and should only be considered if the number reported is large. If the minimum resolution of the server clock is larger than the likely interval for some of the timer-based statistics, those statistics could be inflated.
SQLidu	Number of INSERT, DELETE, or UPDATE statements after SQL_PERF_START.
SQLiduRows	Number of INSERT, DELETE, or UPDATE statements after SQL_PERF_START.
SQLSelects	Number of SELECT statements processed after SQL_PERF_START.
SQLSelectRows	Number of rows selected after SQL_PERF_START.

SQLPERF Field	Description
Transactions	Number of user transactions after SQL_PERF_START, including rollbacks. When an ODBC application is running with SQL_AUTOCOMMIT_ON, each command is considered a transaction.
SQLPrepares	Number of SQLPrepare calls after SQL_PERF_START.
ExecDirects	Number of SQLExecDirect calls after SQL_PERF_START.
SQLExecutes	Number of SQLExecute calls after SQL_PERF_START.
CursorOpens	Number of times the driver has opened a server cursor after SQL_PERF_START.
CursorSize	Number of rows in the result sets opened by cursors after SQL_PERF_START.
CursorUsed	Number of rows actually retrieved through the driver from cursors after SQL_PERF_START.
PercentCursorUsed	Equals CursorUsed/CursorSize. For example, if an application causes the driver to open a server cursor to do "SELECT COUNT(*) FROM authors," 23 rows will be in the result set for the SELECT statement. If the application then fetches only three of these rows, CursorUsed/CursorSize is 3/23, so PercentCursorUsed is 13.043478.
AvgFetchTime	Equals SQLFetchTime/SQLFetchCount.
AvgCursorSize	Equals CursorSize/CursorOpens.
AvgCursorUsed	Equals CursorUsed/CursorOpens.
SQLFetchTime	Cumulative amount of time it took fetches against server cursors to complete.
SQLFetchCount	Number of fetches done against server cursors after SQL_PERF_START.
CurrentStmtCount	Number of statement handles currently open on all connections open in the driver.
MaxOpenStmt	Maximum number of concurrently opened statement handles after SQL_PERF_START.
SumOpenStmt	Number of statement handles that have been opened after SQL_PERF_START.

SQLPERF Field	Description
Connection Statistics:	
CurrentConnectionCount	Current number of active connection handles the application has open to the server.
MaxConnectionsOpened	Maximum number of concurrent connection handles opened after SQL_PERF_START.
SumConnectionsOpened	Sum of the number of connection handles that have been opened after SQL_PERF_START.
SumConnectionTime	Sum of the amount of time that all of the connections have been opened after SQL_PERF_START. For example, if an application opened ten connections and maintained each connection for five seconds, then SumConnectionTime would be 50 seconds.
AvgTimeOpened	Equals SumConnectionsOpened/ SumConnectionTime.
Network Statistics:	
ServerRndTrips	The number of times the driver sent commands to the server and got a reply back.
BuffersSent	Number of Tabular Data Stream (TDS) packets sent to SQL Server by the driver after SQL_PERF_START. Large commands can take multiple buffers, so if a large command is sent to the server and it fills six packets, ServerRndTrips is incremented by one and BuffersSent is incremented by six.
BuffersRec	Number of TDS packets received by the driver from SQL Server after the application started using the driver.
BytesSent	Number of bytes of data sent to SQL Server in TDS packets after the application started using the driver.
BytesRec	Number of bytes of data in TDS packets received by the driver from SQL Server after the application started using the driver.

SQLPERF Field	Description
Time Statistics:	
msExecutionTime	Cumulative amount of time the driver spent processing after SQL_PERF_START, including the time spent waiting for replies from the server.
msNetworkServerTime	Cumulative amount of time the driver spent waiting for replies from the server.

How to Profile Driver Performance Data (ODBC)

To log driver performance data using ODBC Administrator:

1. In Control Panel, double-click **32-bit ODBC**.
2. Click the **User DSN**, **System DSN**, or **File DSN** tab.
3. Click the data source for which to log performance.
4. Click **Configure**.
5. Navigate the Microsoft Navision SQL Server Configure DSN Wizard to the page with **Log ODBC driver statistics to the log file**.
6. Select **Log ODBC driver statistics to the log file**. In the box, place the name of the file where the statistics should be logged. Optionally, click **Browse** to browse the file system for the statistics log.

To log driver performance data programmatically:

1. Call **SQLSetConnectAttr** with fOption set to SQL_COPT_SS_PERF_DATA_LOG and vParam set to the full path and file name of the performance data log file. For example:
2. `"C:\\Odbcperf.log"`
3. Call **SQLSetConnectAttr** with fOption set to SQL_COPT_SS_PERF_DATA and vParam set to SQL_PERF_START to start logging performance data.
4. Optionally, call **SQLSetConnectAttr** with fOption set to SQL_COPT_SS_LOG_NOW and vParam set to NULL to write a tab-delimited record of performance data to the performance data log file. This can be done multiple times as the application runs.
5. Call **SQLSetConnectAttr** with fOption set to SQL_COPT_SS_PERF_DATA and vParam set to SQL_PERF_STOP to stop logging performance data.

To pull driver performance data into an application:

1. Call **SQLSetConnectAttr** with fOption set to SQL_COPT_SS_PERF_DATA and vParam set to SQL_PERF_START to start profiling performance data.
2. Call **SQLGetConnectAttr** with fOption set to SQL_COPT_SS_PERF_DATA and pvParam set to the address of a pointer to a SQLPERF structure. The first such call sets the pointer to the address of a valid SQLPERF structure that contains current performance data. The driver does not continually refresh the data in the performance structure. The application must repeat the call to **SQLGetConnectAttr** anytime it needs to refresh the structure with more current performance data.
3. Call **SQLSetConnectAttr** with fOption set to SQL_COPT_SS_PERF_DATA and vParam set to SQL_PERF_STOP to stop logging performance data.

The SQLPERF structure is defined in Odbcss.h as follows:

```
typedef struct sqlperf
{
   // Application profile statistics

   DWORD TimerResolution;
   DWORD SQLidu;
   DWORD SQLiduRows;
   DWORD SQLSelects;
   DWORD SQLSelectRows;
   DWORD Transactions;
   DWORD SQLPrepares;
   DWORD ExecDirects;
   DWORD SQLExecutes;
   DWORD CursorOpens;
   DWORD CursorSize;
   DWORD CursorUsed;
   LDOUBLE PercentCursorUsed;
   LDOUBLE AvgFetchTime;
   LDOUBLE AvgCursorSize;
   LDOUBLE AvgCursorUsed;
   DWORD SQLFetchTime;
   DWORD SQLFetchCount;
   DWORD CurrentStmtCount;
   DWORD MaxOpenStmt;
   DWORD SumOpenStmt;
```

```
    // Connection statistics
    DWORD CurrentConnectionCount;
    DWORD MaxConnectionsOpened;
    DWORD SumConnectionsOpened;
    DWORD SumConnectionTime;

    LDOUBLE AvgTimeOpened;

    // Network statistics
    DWORD ServerRndTrips;
    DWORD BuffersSent;
    DWORD BuffersRec;
    DWORD BytesSent;
    DWORD BytesRec;

    // Time statistics
    DWORD msExecutionTime;
    DWORD msNetworkServerTime;

}    SQLPERF;
```

Examples:
The example shows both the creation of a performance data log file and displaying performance data directly from the SQLPERF data structure. Error-checking code is removed to simplify this example.

The complete sample code is in this file: CreateAndDisplayPerformanceLog.cpp. You can download an archive containing the sample from the SQL Server Downloads page.

This sample was developed for ODBC version 3.0 or later. It was developed with Microsoft® Visual C++® version 6.0 and may expose properties of the Microsoft Foundation Classes.

NOTE: *When possible, use Windows Authentication. If Windows Authentication is not available, prompt users to enter their credentials at run time. Avoid storing credentials in a file. If you must store credentials, you should encrypt them with the Win32 cryptoAPI.*

Running System Monitor

System Monitor (Performance Monitor in Microsoft® Windows NT® 4.0) collects information from SQL Server using remote procedure calls (RPC). Any user who has Microsoft Windows 2000 permissions to run System Monitor can use it to monitor SQL Server.

NOTE: *When using either System Monitor or Performance Monitor, you cannot connect to an instance of SQL Server running on Microsoft Windows 98.*

As with all performance monitoring tools, expect some performance overhead when monitoring SQL Server with System Monitor. The actual overhead in any specific instance will depend on the hardware platform, the number of counters, and the selected update interval. However, the integration of System Monitor with SQL Server is designed to minimize the impact.

To start the System Monitor in Windows 2000:

- On the Start menu, select PROGRAM FILES→ADMINISTRATIVE TOOLS→PERFORMANCE.

ODBC Tracing

The ODBC Driver Manager has a trace facility that allows the sequence of function calls made by an ODBC application to be recorded and transcribed into a log file. Tracing is performed by a trace DLL that captures calls between the application and the Driver Manager and between the Driver Manager and the driver. This method of tracing replaces the tracing performed by the ODBC 2.*x* Driver Manager and the tracing performed in ODBC 2.*x* by ODBC Spy.

Setting Tracing Options

The **Tracing** tab of the ODBC Data Source Administrator dialog box enables you to configure the way ODBC function calls are traced.

How Tracing Works

When you activate tracing from the **Tracing** tab, the Driver Manager will log all ODBC function calls for all subsequently run applications. ODBC function calls from applications running before tracing is activated are not logged. ODBC function calls are recorded in a log file you specify.

Tracing ceases only after you click **Stop Tracing Now**. Remember that while tracing is on, the log file continues to increase in size and that this affects the performance of all your ODBC applications.

Changes in ODBC Tracing

Prior to MDAC 2.7 SP2, ODBC tracing was only allowed to occur on a machine-wide basis, in which trace captures exposed details about all ODBC applications running under any identities. This included tracing for ODBC-related activity that might occur for processes created or run on behalf of other local user accounts and built-in security principals such as the Local Service and Network Service.

ODBC tracing now defaults to per-user mode. If you are a local administrator, however, you can still enable machine-wide tracing using the ODBC Data Source Administrator.

To configure the ODBC tracing mode:

1. If necessary, log on using an account with membership in the Local Administrators' group.
2. On the Start menu, select PROGRAM FILES→ADMINISTRATIVE TOOLS→ODBC (DATA SOURCES).
3. Click the **Tracing** tab.
4. Configure the tracing mode using the Machine-Wide tracing for all user identities check box:
   a. To enable machine-wide tracing, select the check box.
   b. To return to per-user tracing, clear the check box.
5. Click **Apply**.

NOTE: *If you have already started tracing in one mode, you need to stop tracing and switch to the other mode in order for the mode to be changed successfully.*

IMPORTANT*: Machine-wide tracing should only be enabled when it is needed; otherwise, it should be left turned off.*

Visual Studio Analyzer Tracing

Microsoft® Visual Studio® Analyzer Tracing provides performance and debugging information about the ODBC layer. All outbound events will be fired at the top-level interface to present as accurate a picture as possible regarding time spent in ODBC components. Visual Studio Analyzer Tracing requires any event source to register when the source is set up. For more information about this type of tracing, see the Visual Studio documentation.

Test Your Knowledge – Optimizing Microsoft Navision

True	False	
❑	❑	1. A database table contains a primary key and a secondary key.
❑	❑	2. A B+ tree, or balanced tree, is one in which the program always passes through the same number of levels to access data at the bottom of the tree, regardless of which branch of the tree the data is located on.
❑	❑	3. Keys can be setup as key groups, which can be enabled and disabled without risk.
❑	❑	4. SQL needs an index in order to sort by particular fields.
❑	❑	5. In date compression, you can compress entries from closed fiscal years only once.

Test your Knowledge – Optimizing Microsoft Navision

1. The SQL Profiler is a graphical tool that allows system administrators to monitor events in an instance of Microsoft SQL Server. List the various reasons the SQL Profiler would be used.

2. Long-running queries and driver-performance data are the two different ODBC drivers used to profile performance data. What is the difference between the two?

3. What are the implications of the database-version principle approach in Microsoft Navision?

4. What are some of the limitations or restrictions with using SQL Profiler?

5. What is some of the information found in a trace log?

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

CHAPTER 12: TROUBLESHOOTING MICROSOFT NAVISION (INCLUDING PERFORMANCE TOOLKIT)

Overview

This chapter is designed to help you identify performance problems in a Microsoft® Business Sokutions–Navision® application. It describes how to troubleshoot on both server options and describes and explains how to use the debugging tools that exist in Microsoft Navision to identify performance problems. It also describes how to use the troubleshooting tools that come with this guide. Furthermore, it contains a brief description of some of the Microsoft® SQL Server® tools that you can use.

This document describes some of the most common performance problems and the reasons that can cause them. The topics covered include:

- The Client Monitor and the Code Coverage tool
- The new performance trouble shooting tools
- Hardware setup and performance
- Setting up the test environment
- Identifying the clients that cause performance problems
- Profiling a task with the Client Monitor
- Identifying the worst server calls and the keys and filters that cause them
- Identifying the tasks that cause deadlocks on Microsoft Navision Database Server
- Using the SQL Error Log to identify the clients involved in deadlocks on Microsoft® SQL Server®.
- How to identify locking problems
- How to set up locking order rules and check whether or not your application follows these rules
- Identifying index problems on SQL Server
- How to identify bad C/AL NEXT statements on SQL Server
- How to use Microsoft® Excel pivot tables to get an overview of the data in the Client Monitor

Performance problems can be caused by bottlenecks in the hardware setup or by problems within Microsoft Navision. Performance problems in Microsoft Navision can be caused by, for example, the way that keys are designed, by the way that keys are used together with filters, or by the way that tables or records are locked.

This guide is mainly concerned with identifying performance problems that exist within Microsoft Navision, even though poor performance can be caused by inadequate or badly configured hardware. For more information about hardware considerations, see the section titled “Hardware Setup.”

Database Servers

The two database server options for Microsoft Navision, Microsoft Database Server, and SQL Server, behave differently both with regard to performance and locking. However, the methodology and the tools that you can use to identify these bottlenecks are almost the same.

The Test Environment

You will generally need to set up a separate test environment before you can start troubleshooting and solving the performance problems that exist in a working installation.

Setting up a test environment means:

1. Setting up a separate database server.

   The test server should be set up on a separate computer and not on the computer that is used by the production system. Using a separate test environment gives you complete control over the system and over who has access to it. It also means that the customer can continue to use the production system.

2. Copying the production system database to the test server.

   If you are running on SQL Server, use the backup/restore functions in Enterprise Manager to make a fast copy of the database to the test server. If you are running on Microsoft Navision Database Server, you can use the server-based backup program HotCopy.

3. Warming up the server to ensure that you get realistic measurements.

   You must warm up the test server regardless of which server option you are using.

Warming Up SQL Server

If you have just turned SQL Server on or if you have just created the database or company, you must warm up SQL Server by using the database and the company. This ensures that the system resembles the actual customer installation and means that you can generate realistic performance measurements.

You only need to run an initial test to warm up SQL Server. When SQL Server is warmed up, the execution plans for most queries have already been generated and are ready for use. Furthermore, the most frequently used data is now available in memory.

When SQL Server is not warmed up, you will, for example, see that inserting, modifying or deleting the first record in a table that contain keys that have SumIndexFields associated with them can take up to several seconds to finish. This would normally be done much faster in a working installation.

Warming Up Microsoft Navision Database Server

If you have just started Microsoft Navision or if you have just opened the database and company in Microsoft Navision, you must warm up Microsoft Navision Database Server before you can generate realistic performance measurements. Running an initial test will warm up Microsoft Navision by ensuring that all the objects you need are available on the client.

Debugging Tools

To ensure that you have the most recent version of the debugging tools, install the newest version of the Microsoft Navision client on the computers that you are going to use to test the installation.

NOTE: *The earlier version of Microsoft Navision, Microsoft Navision Financials does not have a Client Monitor for the SQL Server Option. The Client Monitor is an essential element in the troubleshooting procedures described in this training material.*

If you develop corrections in Microsoft Navision that you would like to implement in Microsoft Navision Financials, you can use the REMID feature from the Microsoft Navision Upgrade Toolkit to change the Microsoft Navision objects before importing them into Microsoft Navision Financials.

Client Performance Indicators

When you connect a client to Microsoft Navision Database Server, you can see status information about server calls that take two or more seconds in the status bar.

Typical server calls that generate status information are:

- Server calls that modify or delete sets of records.
- Server calls that scan an index or an entire table to find some data.
- Server calls that need to lock a record or a table can be forced to wait until other transactions are committed and release the locks that they placed.

Therefore, you should keep an eye on the indicator in the status bar when you are trying to identify problematic tasks because this information might be all you need to break down a performance problem.

However, if you are using the Microsoft SQL Server Option for Microsoft Navision, the client's user interface has no indicators to tell you how much time is spent on long running tasks. Therefore, you need some other procedures to break down a performance problem on SQL Server. However, when clients are waiting for locks to be released by other clients, you can see this information by using the session monitor as described in the section titled "Using the Session Monitor to Locate the Clients that Cause Performance Problems on SQL Server."

Common Performance Problems

Many things can cause poor performance but some of the most common causes are:

- The way that keys are defined combined with the way that they are used in filters or queries when you want to read data.
- The way keys are defined with SumIndexFields combined with the way that summing FlowFields are defined.
- The number of keys that are defined with SumIndexFields when running on SQL Server.

The procedures described in this material will help you identify the places where these problems occur.

If you are running on SQL Server, you must also be aware of a very specific performance problem that applies to forms. Setting the SourceTablePlacement property to the default value (Saved) will often make opening forms that display data from tables that contain many records (1,000,000 or more), for example G/L entries, very slow. To fix this problem, set the SourceTablePlacement property to First in these forms.

Performance and Locking

Performance problems that are related to specific tasks should always be tested in the test environment, when no other users are logged on to the database server. This will help you determine whether the performance problem is related to the task itself or if the problem only occurs when the task is executed in combination with other tasks on the same server.

Performance problems that are caused by a specific task are described in the section titled The Client Monitor. Performance problems that are caused by clients spending time waiting for other clients to release locks on resources that the client in question wants to place exclusive locks on are described in the section titled Locking.

Deadlocks occur when concurrent transactions try to lock the same resources but do not lock them in the same order. This can either be solved by always using the same locking order or by using a "locking semaphore" that will prevent these transactions from running concurrently. You can find out which resources are causing deadlocks by following the procedure described in the section titled Locking.

Hardware

When you are trying to identify any bottlenecks that exist in an installation, you must also check the hardware that the installation is running on. You should check the hardware that is being used by both the server and the client computers. See the section titled "Hardware Setup."

Getting Some Assistance

Depending on your expertise in the technical areas that are essential for troubleshooting performance problems, you can choose to follow all the steps described in this troubleshooting guide, or you can stop at some point and give your results to other experts.

For example, the data that you get by using the Client Monitor (as described in the next section) can easily be transferred to other experts. They will then be able to identify performance problems solely on the basis of this data, before looking at the objects from the database and before having access to the data in the database.

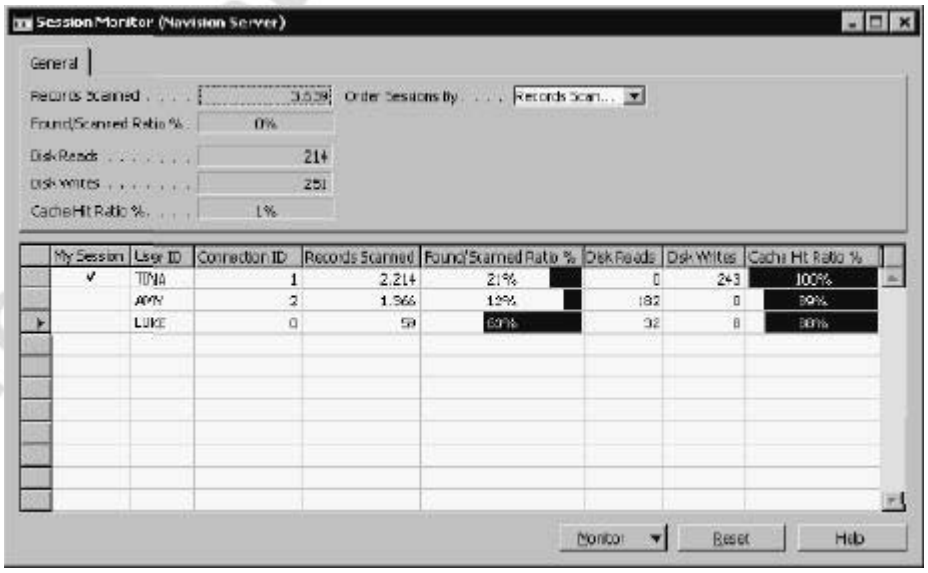
Identifying Performance Problems

This section explains how to identify performance problems in Microsoft Navision.

Using the Session Monitor to locate the clients that cause performance problems on Microsoft Navision Database Server

You use the Session Monitor (Microsoft Navision Database Server) to identify the clients that cause performance problems on Microsoft Navision Database Server.

You must import some helper objects before you can start to identify the clients that are causing performance problems:



Import the Session Monitor (Microsoft Navision Server).fob file. The objects imported include form 150010, Session Monitor (Microsoft Navision Srv).

The Session Monitor (Microsoft Navision Server) window displays updated information from the Session table. The information is refreshed every second by default, but you can change this setting by clicking MONITOR→SETUP.

By default, the most active sessions in terms of the amount of records scanned are shown at the top of the list. The **Records Scanned** field tells you how many records the database server has scanned in order to find the records that this session wanted.

The sessions with the largest number of scanned records are the ones that should be investigated first. Follow the guidelines in the other sections to investigate these sessions.

***NOTE**: In the Session Monitor (Microsoft Navision Server) window, if the value in the **Found/Scanned Ratio** field is high, this indicates that the indexes and queries match. A value of 30-50% is normal, while 3% is low.*

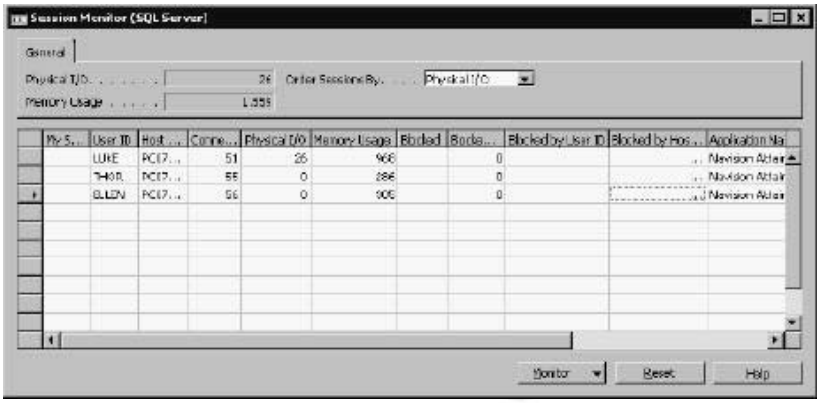
Using the Session Monitor to locate the clients that cause performance problems on SQL Server

You use the Session Monitor (SQL Server) to locate the clients that cause performance problems when you are using the SQL Server Option for Microsoft Navision.

You must import some helper objects before you can start to identify the clients that are causing performance problems:

1. Install the client components for SQL Server from the Microsoft SQL Server CD.
2. Open the Query Analyzer tool, and click FILE→OPEN to open the Session Monitor (SQL Server) .sql file.
3. Click QUERY→CHANGE DATABASE, and select the Microsoft Navision database that you want to monitor.
4. Click QUERY→EXECUTE.
5. Import the Session Monitor (SQL Server).fob file into Microsoft Navision.

6. Run form 150014, Session Monitor (SQL Server).



The Session Monitor (SQL Server) window displays updated information from a view. This view is similar to the one that lies behind the Session table. The Session Monitor (SQL Server) window tells you which clients are currently connected to the server as well as the current load on the server. The information is refreshed every second by default, but you can change this setting by clicking MONITOR→SETUP.

By default, the most active sessions in terms of physical I/O are listed at the top of the Session Monitor (SQL Server) window. These are the sessions that should be investigated first. You can also list the sessions according to their memory usage because this is also is a good indicator of activity. SQL Server can also give you information about the CPU usage, but unfortunately this information is not very reliable on SQL Server 2000.

To investigate these sessions, follow the guidelines described in the following sections.

The Session Monitor (SQL Server) window also lists information about the clients that are waiting for locks held by other clients to be released, as well as the identity of the clients that placed the locks. If you want to concentrate on this area only, look at and/or filter the fields starting with Blocked (Blocked, Blocked by Connection ID, Blocked by User ID, and Blocked by Host Name).

Time Measurements

When you are working with performance problems, it is recommended that you make accurate measurements of the amount of time it takes to perform the tasks in question.

You can monitor and measure the time the task takes by using a stopwatch or preferably by using some helper objects.

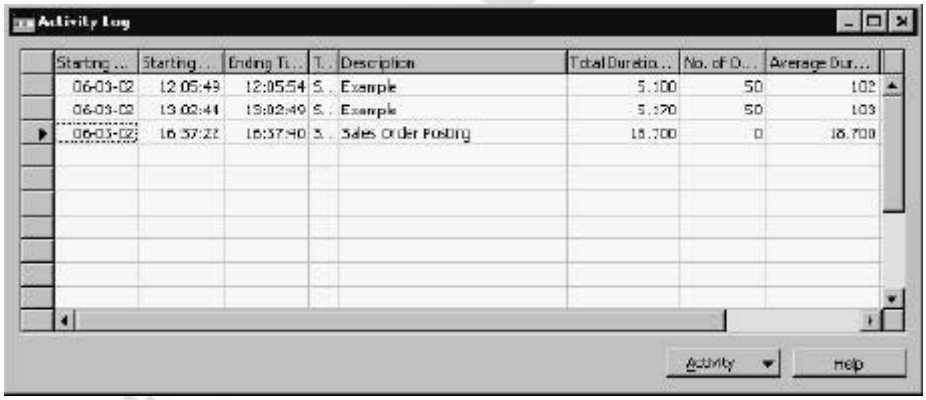
The ActivityLog.fob file contains some objects that enable you to measure precisely the time spent on a particular task. These measurements are stored in a table enabling you to review the amount of time spent on the task at a later date.

To measure the time spent on a task:

1. Import the ActivityLog.fob file, and compile the objects that are imported.
2. Use the objects in the same way as they are used in the Codeunit contained in Sample use of Activity Log.fob.

To see how this codeunit works:

1. Import the Sample use of Activity Log.fob file, and open codeunit 150000, Sample use of Activity Log in the editor to see how the Activity Log is used to monitor the task in the codeunit.
2. Run codeunit 150000, Sample use of Activity Log. This creates a new line in the Activity Log.
3. Run form 150000, Activity Log to open the Activity Log.



The Activity Log window contains the following information:

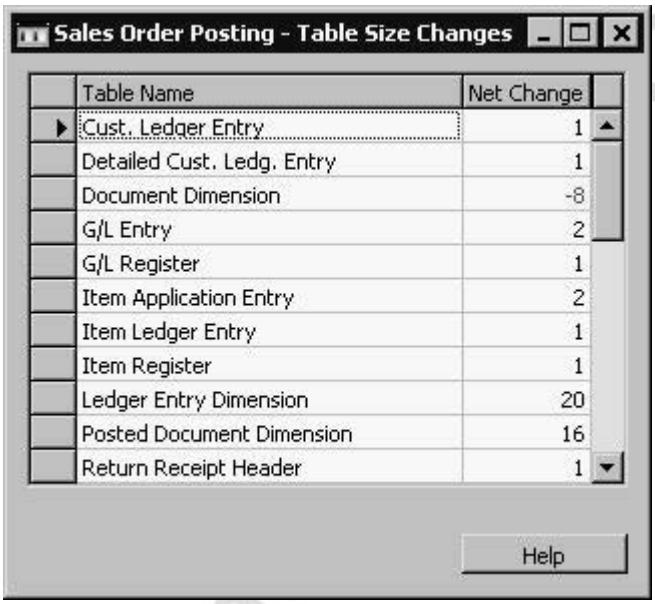
- The date and time at which the activity was started.
- The date and time at which the activity finished.
- The status of the activity.
- The total time in milliseconds that the activity took.
- The number of operations that the activity involved.
- The average time that each operation took.
- A check mark indicating which session was yours.
- The connection ID of the session that carried out the activity.

Details of the Activity

To see the changes, if any, that an activity made to the database, you must enable this feature. Codeunit 150000, Sample use of Activity Log shows you how to enable this feature.

After you have enabled the feature and carried out the activity that you are interested in monitoring, select the activity and click ACTIVITY→TABLE SIZE CHANGES to see the changes that were made to the database.

The Table Size Changes window appears:



This window contains information about the net amount of records that were entered into or deleted from the tables in the database during the activity. Knowing the amount of changes that an activity involves helps you understand the amount of time used by a particular task.

The Client Monitor

The Client Monitor is an important tool for troubleshooting performance and locking problems. You can also use it to identify the worst server calls and to identify index and filter problems in the SQL Server Option. The Client Monitor and the Code Coverage tool now work closely together allowing you to easily identify, for example, the code that generated a particular server call.

Using the Client Monitor to Profile a Task

The Client Monitor displays all the details of the server calls made by the current client, including the time spent on each server call. This makes it an invaluable tool when you want to analyze a particular task and study the server calls that the task makes as well as the code that initiates the server calls.

To profile and analyze a given task in Microsoft Navision using the Client Monitor, you must have some Client Monitor helper objects:

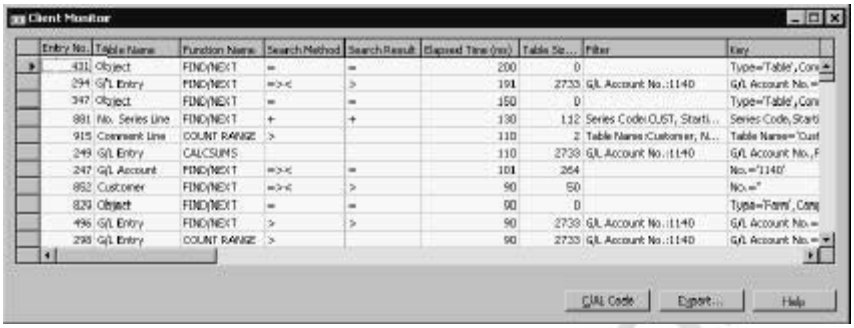
1. Import the Client Monitor.fob file, including Form 150020, Client Monitor.
2. Compile all of the objects that are imported. This must be done because some of the field definitions are different on the two database server options.
3. Click TOOLS→DEBUGGER→CODE COVERAGE to open the Code Coverage window. Start the Code Coverage tool, and then start the Client Monitor just before you are ready to perform the task that you want to investigate.
4. Perform the task that you want to test.
5. When you have finished the task, stop the Client Monitor and then stop the Code Coverage tool. The Client Monitor uses many lines to describe a single server call, and this makes it difficult to use for data analysis.
6. Run form 150020 Client Monitor. This processes the data from the Client Monitor and displays it in a new window.

The Client Monitor window displays and formats the data that has been gathered by the Client Monitor so that it can be more easily analyzed. It carries out a kind of cross tabulation of the operations and parameters and uses one line per server call.

IMPORTANT: *When you use these tools, make sure that your test tasks are focused on the area that you are interested in testing. If you spend time doing other tasks, both the Client Monitor and the Code Coverage tool will fill up with irrelevant information.*

You can analyze the Client Monitor data within Microsoft Navision, or you can perform a more detailed analysis by importing the data into pivot tables in Excel. If you are analyzing a lengthy task that takes an hour or more to run, you should consider restricting the scope of the task. You can limit the task by applying filters that will make the task handle less data or by stopping the task after several minutes. You can then use the Client Monitor data from the part of the task that was performed as the basis for your analysis.

Here is an example of the kind of data that you can see in the Client Monitor (taken from SQL Server):



The Client Monitor displays the database function calls that are made by the C/AL code as follows:

Function call in C/AL:	**Function Name (+Search Method) shown in the Client Monitor:**
GET	FIND/NEXT(‘=’)
FIND(‘-‘)	FIND/NEXT(‘-’)
NEXT	FIND/NEXT(‘>’)
ISEMPTY	ISEMPTY (as long as no MARKEDONLY filter is used)
CALCSUMS	CALCSUMS
CALCFIELDS	If the FlowField is of type sum: CALCSUMS If the FlowField is of type lookup: FIND/NEXT(‘-‘)
LOCKTABLE	LOCKTABLE
INSERT	If the table is not locked already: LOCKTABLE INSERT
MODIFY	If the table is not locked already: LOCKTABLE Often on SQL Server: FIND/NEXT(“=”) MODIFY
DELETE	If the table is not locked already: LOCKTABLE DELETE
MODIFYALL	If the table is not locked already: LOCKTABLE MODIFYALL (as long as validation code is not executed)
DELETEALL	If the table is not locked already: LOCKTABLE DELETEALL (as long as validation code is not executed)

Generally, the FIND/NEXT function in the Client Monitor means:

Function Name:	Search Method:	Means:
FIND/NEXT	–	FIND ("-") or find the first record (within the current filter using the current key).
FIND/NEXT	+	FIND ("+") or find the last record (within the current filter using the current key).
FIND/NEXT	>	NEXT or find the record with key values greater than the current key values (within the current filter using the current key).
FIND/NEXT	<	NEXT (-1) or find the record with key values less than the current key values (within the current filter using the current key).
FIND/NEXT	=	GET or find one record with key values equal to the current key values (within the current filter).

In the Client Monitor:

- For each FIND/NEXT, the **Search Method** field tells you what kind of FIND/NEXT server call it is (see the previous table).
- For each FIND/NEXT, the **Search Result** field contains a value (the same as the **Search Method** field) if data was found. Otherwise, it is blank.
- For each COMMIT, the **Commit** field normally contains the value 1, indicating that the transaction was committed. If the transaction was rolled back and not committed, the value is 0.
- The **Elapsed Time (ms)** field shows the amount of time that elapsed between the start and the end of a server call. This means that it gives you the total time spent on the client, the network, and the server.

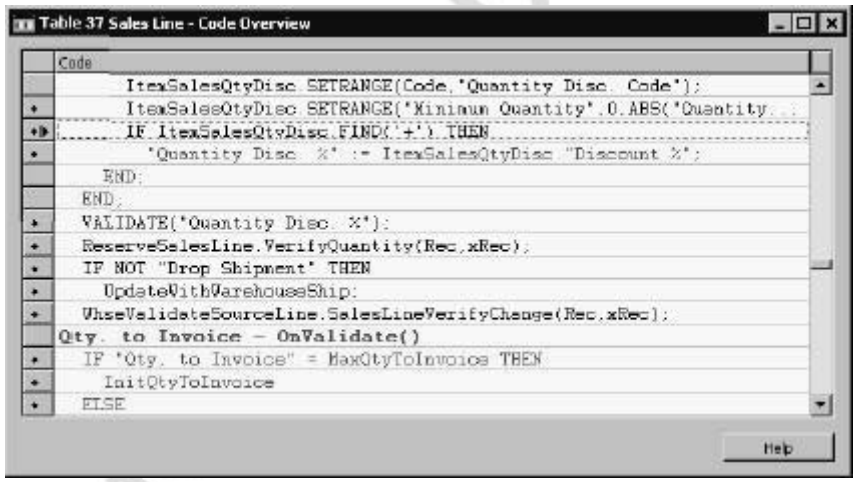
  Note that the information in the **Elapsed Time (ms)** field tends to be slightly inaccurate for fast server calls. If a server call takes 1 ms, then the average Elapsed Time (ms) is 1 ms. However, 90% of the server calls will have 0 in the **Elapsed Time (ms)** field, while 10% of the server calls will have 10 in the **Elapsed Time (ms)** field.
- The **Table Size (Current)** field tells you the amount of records that were in the table when you started the Client Monitor. This information can help explain why server calls to a particular table can take such a long time (that is, when the Elapsed Time (ms) values are high).
- The **SQL Status** field tells you whether the data was read from the server or from the client cache.

- A check mark in the **Locking** field indicates that the server call has locked data. To focus on locking issues, you can place a filter on this field.

The Client Monitor cannot be used to analyze temporary tables because they are stored on the client and inserts into temporary tables do not involve server calls.

The Client Monitor gathers and displays all the database function calls that are made by the C/AL code, as well as the server calls that are made indirectly by, for example, opening a form. The C/AL code that initiates a database function call can be seen in the **Source Object** and **Source Text** fields.

However, this is not the most informative way of viewing the code. To get a more detailed overview of the code that made a particular database call, select the line in question in the Client Monitor and click **C/AL Code**. The Code Overview window opens displaying the code that made the database function call. This gives you a quick way of locating the relevant piece of code.



The focus of the Code Overview window is the line in the code that made the database call. If C/SIDE made the database call, the Code Overview window will point to the last C/AL code that was executed before C/SIDE made the database call.

***NOTE**: When the code contains an IF..THEN ..ELSE statement, the break point that is displayed in the Code Overview window will often be slightly inaccurate.*

You can also specify that the information collected by the Code Coverage tool is stored permanently in a table. This means that the data is always available and you will not have to profile the task every time you need to see this data. However, permanently storing this information makes the Client Monitor and the Code Coverage tool work more slowly.

To store the code coverage information in a table:

1. Run form 150021, Client Monitor Setup.
2. Enter a check mark in the **Save Code Coverage Info.** field.

The Most Problematic Server Calls

To identify the most problematic server calls:

1. Profile your task as described in the section titled Using the Client Monitor to Profile a Task.
2. Run form 150020, Client Monitor.
3. Click VIEW→SORT to sort the data in the Client Monitor window. Sorting by Elapsed Time (ms) in descending order is one of the more useful ways of viewing the data. The server calls that took the longest time will then be listed at the top. This will help you identify the most problematic server calls.

After you have identified the problematic server calls, you can optimize the slow queries that are caused by filters and keys that do not match (especially on SQL Server) by using the appropriate keys in the queries or possibly by changing the existing keys.

NOTE: *Rearranging the fields in a key, for example, by moving the first field in a key to the end and by changing the references to the key (both in the code and in the properties), can solve a performance problem. Furthermore, any FlowFields in the key that are calculating sums are guaranteed to work as long as all the original fields are left in the key. If you remove some of the fields from a key, you can cause some FlowFields that are calculating sums to produce run time errors.*

When you are developing an application, you will not encounter problems like the one described above, unless you enter some pseudo-realistic amounts of data into the database.

Carefully read the information about performance in the section titled “Keys, Queries and Performance.”

You can use the Client Monitor together with Excel to analyze the time spent by tasks that make many server calls (that is, 100+). You must begin by profiling the task as described in the section titled Using the Client Monitor to Profile a Task. The data must then be transferred into Excel.

To transfer the data into Excel:.

1. Run form 150020, Client Monitor.
2. Click **Export** and save as a .txt file.
3. In Excel, import the .txt file that you have just saved.

You now have a spreadsheet containing the basic Client Monitor data.

You use the pivot tables in Excel to generate a sorted list of the tables that take the most time. The pivot table must also list the functions that are used as well as the search method and the search result for each table. You must also check the server calls that generated the sums to see the average amount of elapsed time for each server call.

You can also create new spreadsheets that summarize different operations on various tables by using the Pivot Table feature.

To create a pivot table:

1. Click DATA→PIVOTTABLE→PIVOTCHART REPORT, and click **Finish** in the wizard that appears.

   You can now choose which breakdown of the Client Monitor elements you would like to analyze. This example uses the typical elements.

2. In the PivotTable window, select the **Table Name** button and drag it over to the range that says "Drop Row Fields Here."

3. Repeat this procedure for **Function Name**, **Search Method** and **Search Result** placing each field to the right of the previous field.

4. Drag Elapsed Time (ms) over to the range that says "Drop Data Items Here."

You can now see the breakdown of timings per table/function etc, summed up.

To list the most important tables first:

1. Double-click the **Table Name** field heading.

2. Select **Advanced**.

3. In the AutoSort options, select **Descending**. In the Using drop-down list select Sum of Elapsed Time (ms).

To list the most important functions per table first, repeat this procedure for the **Function Name** field.

If there are any totals that you do not want to see, right-click the field that contains the word Total and click **Hide**.

For more information about pivot tables, see the online help in Excel. Here is a snapshot of an analysis of elapsed time:

Analysis at Table and Function Level – Sum of Elapsed Time (ms)

Table Name	Function Name	Search Method	Search Results	Total
Reservation Entry	CALCSUMS	(blank)	(blank)	380
	DELETE	(blank)	(blank)	1382
	FIND/NEXT	-	- (blank)	630 361
		+	+ (blank)	0 30
		=	= (blank)	140 0
		>	> (blank)	40 1851
	INSERT	(blank)	(blank)	20
	LOCKTABLE	(blank)	(blank)	0
	MODIFY	(blank)	(blank)	1673
Reservation Entry Total				6507
Calendar Entry	FIND/NEXT	+	+	490
		<	(blank)	1792
Calendar Entry Total				2282

These pivot tables contain information about the amount of time that was spent performing certain operations. The table contains a table-by-table breakdown of the time spent on each of the tables that were involved in the operations as well as the total time used. The tables are listed in ascending order of time spent. The table also lists the functions that were called on each table and the time it took to perform each operation.

The first table listed is the Reservation Entry table. You can see that a total of 380 milliseconds were spent calculating sums and 1382 milliseconds were spent performing deletions. A list of the FIND operations follows, which details the operations that were performed and the amount of time that each operation took.

A total of 20 milliseconds were spent performing inserts into the Reservation Entry table. A total of 1673 milliseconds were spent modifying data in the Reservation Entry table.

If time is spent on modifications (INSERT, MODIFY, DELETE, MODIFYALL, DELETEALL) and the average time spent on modification server calls is high, you should check the keys in the table. The number and length of the keys greatly influence the time it takes to make modifications on both database servers. On SQL Server, if the average time spent on modification server calls is extremely long, check whether or not there are SumIndexFields in the keys and whether or not the MaintainSIFTIndex property is set to Yes for these keys. Setting the MaintainSIFTIndex property to No for these keys can greatly improve the speed of modification server calls, but there will be some loss of performance for those tasks that generate sums using these keys.

Locking

You can also use the Client Monitor to see whether or not locking prevents concurrent tasks from being executed at the same time and to identify where deadlocks occur in a specific multi-user scenario. Before using the Client Monitor to identify potential locking problems, you must import some Client Monitor helper objects:

1. Import the Client Monitor.fob file, including the following forms:

   Form 150020, Client Monitor

   Form 150024, Client Monitor (Multi-User)

2. Compile all the objects that are imported. This must be done because some of the field definitions are different on the two database server options.

To identify locking problems:

1. Before trying to identify any locking problems, you must make sure the clocks are synchronized on all the client machines. You can set up computers running most Windows operating systems so that their clocks are automatically synchronized with the time on a server when they log on by using the following command:

   "`net time \\computername /SET`".

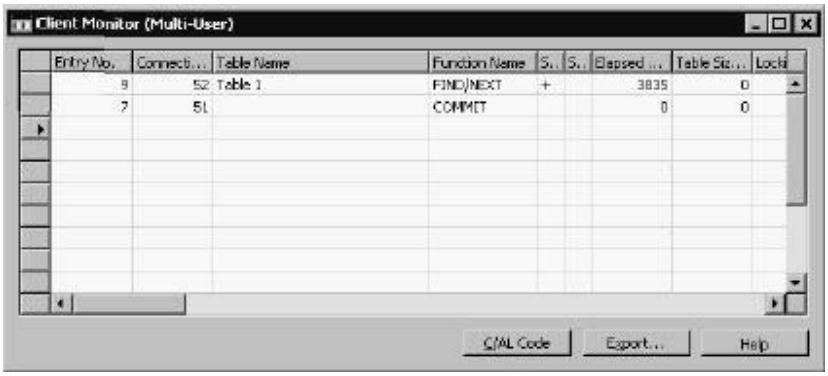
2. Start the Client Monitor on all the computers that are involved in the multi-user test.

3. Perform the tasks that you want to test.

4. Stop the Client Monitor on all the computers.

5. On each client computer, process the common client monitor data by running form 150020, Client Monitor.

6. Run form 150024, Client Monitor (Multi-User) on one of the client computers.



The Client Monitor (Multi-User) window contains information about the transactions that might have blocked other clients. The COMMIT in the transactions that might have blocked other clients is shown together with the server calls made by the clients that are potentially waiting for the COMMIT to be completed. The other server calls are listed before the COMMIT.

For example, in the window shown here, you can see that the client with Connection ID 52 has waited almost four seconds. This indicates that there might have been a block.

7. Check the values in the **Elapsed Time (ms)** field to see if there are any server calls that are taking longer than normal.

   A high value in the **Elapsed Time (ms)** field indicates that a server call is waiting for locks to be released. Use the filtering features in Microsoft Navision to see all the details of the locking scenarios. The value in the **Locking** field is Yes when a server call locks data. You should put a filter on this field to limit the data.

8. If a deadlock has occurred on SQL Server, the **SQL Error** field in the Client Monitor (Multi-User) window will show the error message generated by SQL Server. To see all these lines, set the filter "SQL Error<>". These lines are marked red and bold.

For more information about locking, see the section titled "Locking in Microsoft Navision–A Comparison of Microsoft Navision Database Server and SQL Server."

- You can "optimize" locking by checking whether a set is empty or not. If it is not empty, you can continue locking and reading through the set. If a table is empty, it should not be used at all, and this will remove all the locking problems that are caused by that table. Such a solution can, for example, be acceptable for the comment lines on a sales order during posting.

- You can limit the locking on SQL Server by setting the MaintainSIFTIndex property on a key with SumIndexFields to No.
- In Microsoft Navision, you can use the SIFTLevelsToMaintain property to more precisely control the level of performance and locking on SQL Server.

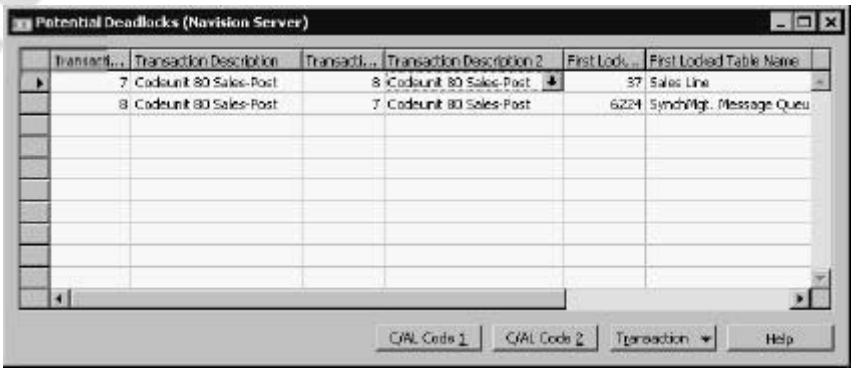
Locating the Tasks That Cause Deadlock Problems on Microsoft Navision Database Server

Two transactions can only cause a deadlock if they both lock some of the same tables. However, if both of the transactions are defined so that the first lock they place is on the same table, no deadlock will occur. In other words, a deadlock occurs when two or more transactions have a conflicting locking order.

When you want to identify potential locking problems, you only need to use one client. You run the tasks on the client that you think might cause locking problems and gather all of the relevant data in the Client Monitor and then open a special form to see if there are any potential deadlocks.

To find potential deadlocks on Microsoft Navision Database Server:

1. Import the Client Monitor.fob file, if you have not already imported it.
2. Compile all the objects that are imported. This must be done because some of the field definitions are different on the two database server options.
3. Prepare the tasks that you want to run concurrently without any deadlocks occurring.
4. Open and start the Code Coverage tool, and then open and start the Client Monitor.
5. Perform the tasks.
6. Stop the Client Monitor, and then stop the Code Coverage tool.
7. Run form 150030, Potential Deadlocks (Microsoft Navision).



This form lists all the potential deadlocks or locking order conflicts that occurred during the tasks that you performed and is based on an analysis of the locking order that is used in each write transaction that was carried out. Each line in the window contains information about two transactions that represent a potential deadlock. These transactions represent a potential deadlock because they both lock some of the same tables but lock them in a different order. Sets of transactions that do not contain a potential deadlock are not displayed.

Each line in the window contains the following information:

Field Name	Contains
Transaction No.	The number of the first of the two transactions that represent a potential deadlock.
Transaction Description	By default, this field contains the name of the object that initiated the first transaction.
Transaction No. 2	The number of the second transaction.
Transaction Description 2	By default, this field contains the name of the object that initiated the second transaction.
First Locked Table ID	The ID and the name of the first table that is common to both transactions and is locked by the first transaction.
First Locked Table Name	
First Locked Table ID 2	The ID and the name of the first table that is common to both transactions and is locked by the second transaction.
First Locked Table Name 2	

From this form, you can access more detailed information about the locks that were placed by each transaction, as well as the code that was used.

- To see the details about one of the transactions, select a line in the Potential Deadlocks (Microsoft Navision Server) window, and click TRANSACTION→CLIENT MONITOR→TRANSACTION 1 OR 2. The details are displayed in the Client Monitor.
- To see the locking operations that were performed by one of the transactions, select a line in the Potential Deadlocks (Microsoft Navision Server) window, and click TRANSACTION→CLIENT MONITOR (LOCKING OPERATIONS ONLY)→TRANSACTION 1 OR 2. The locking operations are displayed in the Client Monitor.
- To see the locking order used by one of the transactions, select a line in the Potential Deadlocks (Microsoft Navision Server) window, and click TRANSACTION→LOCKING ORDER→TRANSACTION 1 OR 2. The locking order used by the transaction you select is displayed in the Transaction Locking Order window.
- To see the code that made the first server call in the first transaction, select the appropriate line in the Potential Deadlocks (Microsoft Navision Server) window, and click **C/AL Code 1**. The code is displayed in the Code Overview window.

Locking Order Rules on Microsoft Navision Database Server

As stated in the previous section, a deadlock occurs when two or more transactions have a conflicting locking order, and no deadlock can occur if the first lock the transactions place is on the same table.

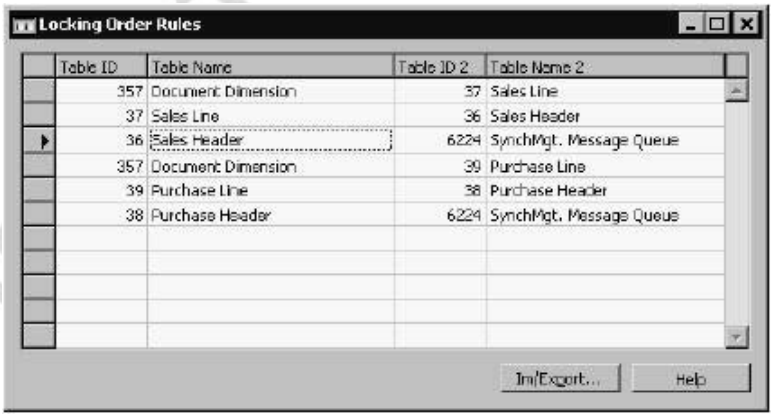
From this, we can deduce that if you have an agreed set of rules that determine the locking order that must be used in your application, then no deadlocks will occur. The problem is that agreeing to a set of rules is one thing, adhering to the rules is another thing entirely. Remembering the rules is not as easy as it sounds – there could be a large number of them.

There is now a tool that can help you see whether or not your application follows the locking rules that you have specified.

This involves determining which rules must apply in your application, entering them into the system, and then checking your application to see if it violates the rules or not.

After you have determined the rules that must govern locking order in your application, you can enter them into the system:

1. Run form 150029, Locking Order Rules, and the Locking Order Rules window appears:

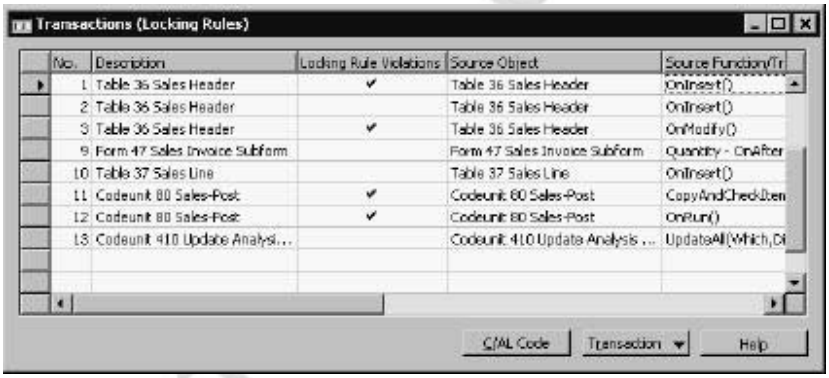


2. Enter the rules that you want your application to follow. Each entry represents a rule, and you can enter as many rules as you need.

Each rule specifies that one table must be locked before another table. Needless to say, your rules must not contain any conflicts. The consistency of the rules is checked when you test your application to see if it follows the rules. If the rules contain a conflict, you will receive an error message.

After you have entered the rules, you can test whether or not your application follows the rules.

1. Open and start the Code Coverage tool, and then open and start the Client Monitor.
2. Perform the tasks that you want to test.
3. Stop the Client Monitor, and then stop the Code Coverage tool.
4. Run form 150027, Transactions (Locking Rules).
5. The Transactions (Locking Rules) window appears listing the transactions that you performed:



   - If any of the transactions violated the rules that you specified earlier, a check mark is displayed in the **Locking Rule Violations** field.
   - To see the C/AL code that broke the locking rule, select the transaction in question and click **C/AL Code**. The C/AL Code – Code Coverage window appears displaying the relevant code.
   - To see all the operations and tables that were involved in a particular transaction, select the transaction and click TRANSACTION→CLIENT MONITOR.
   - To see only the locking operations and the tables that were locked in a particular transaction, select the transaction and click TRANSACTION→CLIENT MONITOR (LOCKING OPERATIONS ONLY).
   - To see the order in which tables were locked by a particular transaction, select the transaction and click TRANSACTION→LOCKING ORDER.
   - To see the locking rules that were violated by a particular transaction, select the transaction and click TRANSACTION→LOCKING RULES VIOLATED.

Locating the Clients That Cause Deadlock Problems on SQL Server

When a deadlock occurs, one transaction continues, while the others are aborted. To find out which clients are involved in a deadlock on SQL Server and which clients are not stopped, see the section titled "SQL Server Error Log."

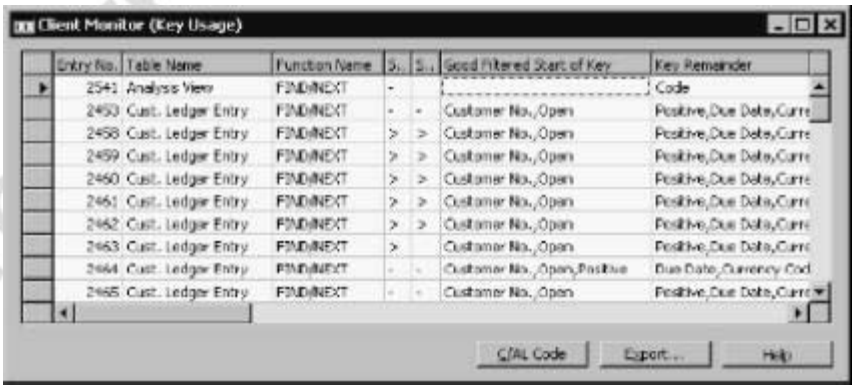
Identifying the Keys That Cause Problems on SQL Server

When you are using the SQL Server Option, it is important that any customizations that you develop contain keys and filters that are designed to run optimally on SQL Server. Therefore, we have developed a tool that helps you test your keys and filters in a development environment and ensure that they conform to the demands made by SQL Server.

When you want to see if an application contains any keys that might cause problems, you only need a demonstration database and not a copy of the customer's database. Inserting, modifying, and deleting records will take a similar amount of time in both large and small databases. However, the amount of time that it takes to read will be very different, especially for tables that become very large in the customer's database. This means that an analysis based on the **Elapsed Time (ms)** field in the Client Monitor is not enough when you are troubleshooting performance problems in a small database.

To check whether the keys and filters are designed correctly:

1. Open and start the Code Coverage tool, and then open and start the Client Monitor.
2. Perform the task that you want to test.
3. Stop the Client Monitor, and then stop the Code Coverage tool.
4. Open form 150022, Client Monitor (Key Usage).



Queries with filters that do not use the keys appropriately are shown in this window. The key that is being used is split into two fields: **Good Filtered Start of Key** and **Key Remainder**. Those fields that are filtered to a single value, but are not used efficiently on SQL Server because of the selection and ordering of fields in the key that is used, are shown in the **Key Candidate Fields** field.

Remember, SQL Server always wants a single-value field as the filter or as the first field in the filter. For more information about keys and filters, see the section titled "Keys, Queries and Performance."

The information in the Client Monitor (Key Usage) window is sorted by table name and only displays the queries with filters that can potentially cause problems. Therefore, you will have to use your knowledge of the application that you are developing and the theory behind the design of keys for SQL Server to decide which of the queries you can ignore and which you will have to modify.

In general, you should:

- Ignore those queries that use small tables that will not grow very large in the customer's database. An example of a small table that you can readily ignore is table 95, G/L Budget Name.
- Focus on the large tables and the tables that will grow rapidly in the customer's database.
- Focus on the Key Candidate Fields and the Good Filtered Start of Key fields.
- As mentioned earlier you should look at the **Good Filtered Start of Key** field. If this field is empty, check the **Key Candidate Fields** field and decide whether the fields shown here would have made a difference if they had been used efficiently. This is where your understanding of the application will help you.

You need to decide whether the suggested key will make the query run more efficiently or not. If the suggested filter is a field that contains many different values, it will probably help.

If you really understand the theory behind the design of efficient queries, you will know whether or not it makes sense to change the application.

However, if you are uncertain about the theory you will have to test the suggested query. This means that you must use a database that contains a realistic amount of data and then test the existing filter and the suggested new filter to see which one works more efficiently.

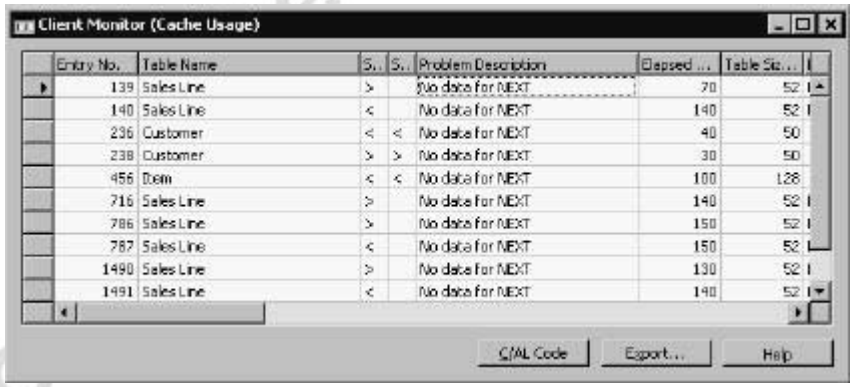
NOTE: *The Client Monitor (Key Usage) window also gives you information about how a key works on Microsoft Navision Database Server. Some keys will give problems on both servers, and others will only be a problem on SQL Server.*

Reading Problems on SQL Server

When you are running on SQL Server and are reading data, some NEXT function calls can generate separate SQL statements instead of using the data that is stored in cache, and this can cause performance problems. The Client Monitor also contains a tool that can help you locate the C/AL code that generated these problematic SQL NEXT statements.

To locate this C/AL code, you must perform the tasks in question, identify the problematic NEXT statements, and then locate the C/AL code that generated these statements.

1. Import the Client Monitor.fob file, if you have not already imported it.
2. Compile all the objects that are imported. This must be done because some of the field definitions are different on the two database server options.
3. Prepare the tasks that you want to perform.
4. Open and start the Code Coverage tool, and then open and start the Client Monitor.
5. Perform the tasks that you want to test.
6. Stop the Client Monitor, and then stop the Code Coverage tool.
7. Run form 150023, Client Monitor (Cache Usage).



The Client Monitor (Cache Usage) window lists the problematic NEXT statements that were generated by the tasks that you performed. All of the normal NEXT statements are ignored.

These NEXT statements are problematic because they generate their own SQL statement and database call to retrieve data from the server and do not use the data that is already cached on the client. It is difficult to know with certainty why these NEXT statements behave like this. It might be:

- Because C/SIDE is unable to optimize this function.
- A result of the way that the code is written.

However these NEXT statements only cause problems if they are run repeatedly as part of a long-running batch job and generate a large number of extra server calls.

To see the C/AL code that generated the SQL NEXT statements, select the line you are interested in and click **C/AL Code**. The code that generated the statement is displayed in the Code Overview window.

To export the data in this window as a text file, click **Export**.

Other Issues

This section discusses some of the other issues that you must take into consideration when you are trying to identify and solve performance problems.

Detecting Bottlenecks

A bottleneck causes the system to run slowly and perform poorly. Bottlenecks can occur in the software or the hardware.

Software Bottlenecks

A bottleneck in the software may result from the system being poorly adjusted. There are three relevant properties to adjust in Microsoft Navision:

- DBMS (Database Management System) Cache
- Commit Cache
- Object Cache

If the DBMS Cache is set too high and if there is not enough of the faster physical memory for the system's requirements, the system will begin to swap (using the slower hard disk as memory). If the DBMS cache is set too low, there may not be enough DBMS cache for Microsoft Navision to operate. Both situations cause the system to run slowly. Turning off the Commit Cache can also slow down the system.

DBMS Cache and Commit Cache

The DBMS cache and commit cache are database related. Therefore, in a client/server environment, these two caches are on the server along with the database, and the object cache is on the client machine. (In a stand alone installation, all three caches are together.)

The system may perform badly if these caches are not optimally sized for your system. The following formula is a way of calculating the amount of DBMS cache you need:

Size of DBMS Cache =

+ Amount of physical memory

– Memory requirements of the operating system –

50 kilobytes x number of workstations – 1 MB for the Microsoft Navision Database Server

– memory requirements of other programs running

The maximum size of the DBMS cache is 1000 MB. The commit cache takes up space in the DBMS cache (but it does not take up additional space in memory). It can use up to 2/3 of the allocated size of the DBMS cache.

Object Cache

Object cache is used on the client computer to store the objects (code, descriptions and windows) retrieved from the server. The client needs to retrieve these objects only once from the server and can then store them in the object cache. The total size of all the objects used in the standard application is around 20 MB. If you have enough memory, set the object cache to 20 MB. The size of the most important objects (the table descriptions) is 1 MB. You should set the object cache to 1 MB as a minimum. This is the default value.

When you install the server program in a network, you can choose to install it on the file server or on a dedicated server. Installing the program on a dedicated server gives better performance because a dedicated server does not have to do other things. Remember to place the database on the same server as the server program so the server program does not have to use the network to get information from the database.

Hardware Setup

It is also possible that poor performance is caused by the hardware that you are using. There are three aspects of the hardware that you should consider: the server, the clients, and the network.

Hardware Setup for the Database Server

To identify any hardware bottlenecks that may exist on a server, use the Windows Performance Monitor. You should check the time usage for the both the disks and the CPU.

If the disks cause the bottleneck, you can lessen the problem and improve performance by:

- Adding more RAM to decrease disk activity, such as swapping.
- Spreading the SQL Server database and log files across more disks.
- Adding one disk controller per disk.
- Installing faster disks.

- If the bottleneck is caused by the CPUs, you can improve performance by installing faster CPUs.

On SQL Server, if the performance problems are only significant when multiple users are working simultaneously, adding more CPUs will lessen the problem. If the bottleneck is not caused by the hard disks or the CPUs, it is outside the server and lies either in the network or on the client computers.

It is always a good idea to expand the search for bottlenecks beyond the server hardware and try to identify the reasons behind the unacceptable usage of resources.

Hardware Setup for Clients

To identify any hardware bottlenecks that may exist on a client computer, use the Windows Performance Monitor. You should check the time usage for both the disks and the CPU.

If the disks on the client computer cause the bottleneck, you can lessen the problem and improve performance by:

- Adding more RAM to decrease disk activity such as swapping.
- Installing faster disks.

If the bottleneck is the CPU, you can improve performance by changing to a faster CPU.

If it is not the hard disks or the CPU that are causing the bottleneck, the problem (if it is a hardware problem) is located outside the client and lies either in the network or on the server.

The Performance Monitor

Another way to get similar information about CPU, disks, net, and memory is to use the Performance Monitor.

HINT: *To see the disk performance with the Performance Monitor, make sure that the device named diskperf has been started.*

The following table illustrates the ways in which the Performance Monitor can help you deal with bottlenecks that can occur when running Microsoft Navision:

Problem	Solution
Pages/sec shows that the memory has been swapping to the disk when the Microsoft Navision Database Server is running. (You add Pages/sec to the Performance Monitor by clicking **Edit**, Add to Chart and in the Add to Chart window, select the Object Memory and the Counter Pages/sec.)	Buy some more RAM for the computer, or try to decrease the Microsoft Navision cache size according to the formula given earlier in the Software Bottlenecks section.

Problem	Solution
% Disk Time is more than 80% over a long period (not peak value), and you are sure that the system is not swapping. Remember to select the correct instance (that is, the correct disk), and make sure that the diskperf device is started. (You add % Disk Time to the Performance Monitor by clicking **Edit**, Add To Chart and in the Add to Chart window, selecting the Object PhysicalDisk and the Counter % Disk Time.)	By default, this field contains the name of the object that initiated the first transaction.
% Processor Time for the server process is 80% or more over a long period (not peak value), indicating that the CPU is used all the time for the Server process. (You add % Processor Time to the Performance Monitor by clicking **File**, Add To Chart, and in the Add to Chart window, selecting the Object Process – not Processor and the Counter % Processor Time and then selecting the Server process in the **Instance** field.)	Buy a faster CPU for the machine.

Network

If neither the database server nor the clients seem to be the bottleneck in a system, look at the network. To find out if a critical task runs slowly because the network is slow, run the same task directly on the database server. The difference in speed tells you the maximum improvement you can hope to achieve by having an optimal network. You can also create a form based on the Performance virtual table to see some measurements for the network you are using.

The good thing about hardware problems is that the solution is always hardware. It will always be possible to improve performance by adding new hardware. The improvement may only be slight, but it is still an improvement. Unfortunately, this can give you a false sense of security. Your installation may have some serious problems, the symptoms of which you are only alleviating by adding new hardware, while the real source of the problem remains unaltered.

SQL Server Error Log

To monitor deadlocks on SQL Server, enable trace flags 1204 and 3605 (if you are running SQL Server 2000 and have not upgraded to Service Pack 3) in the Startup Parameters of SQL Server, and then restart the server. This will generate diagnostic messages in the SQL Server Error Log.

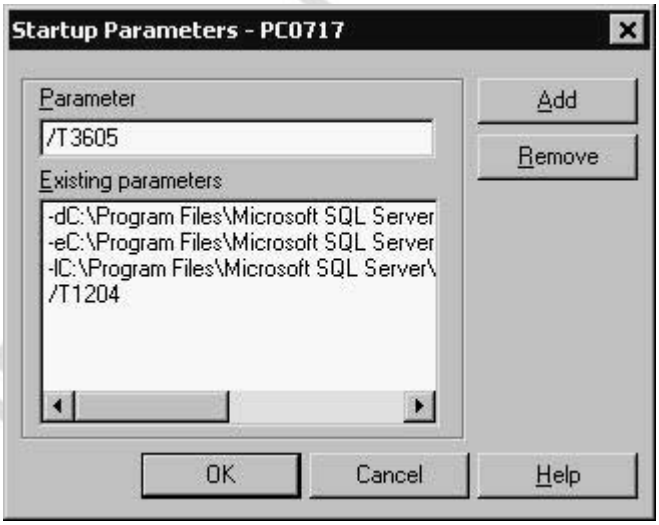
Trace Flag 1204 − returns the type of locks that are participating in the deadlock and the current command affected.

Trace Flag 3605 − sends the trace output to the error log. This is always done when you are running SQL Server 2000 Service Pack 3 or later. (If you start SQL Server from the command prompt, the output also appears on the screen.)

Second, you should implement a SQL Server alert to notify an operator when a deadlock occurs so that an investigation of the deadlock scenario can start immediately.

To enable a trace flag:

1. Open Enterprise Manager.
2. Expand Microsoft SQL Servers and expand SQL Server Group.
3. Right-click the server for which you want to enable the trace flags, and select Properties.
4. In the SQL Server Properties (Configure) window, click **Startup** P**arameters**, and the Startup Parameters window appears:

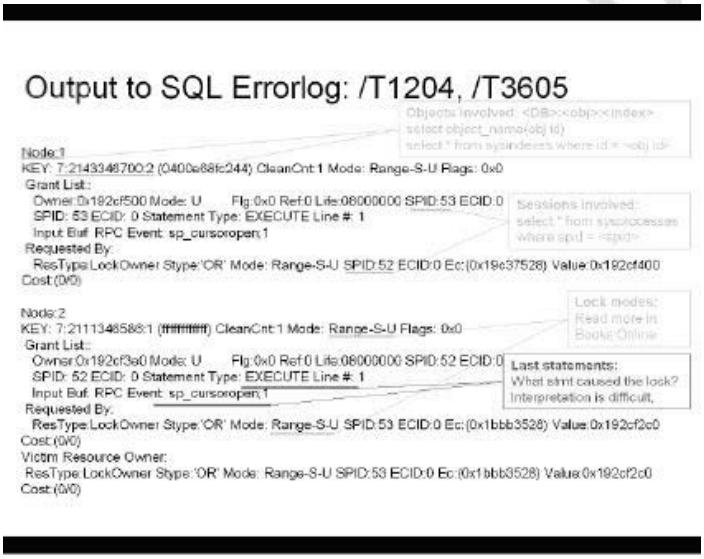


5. Enter the name of the trace flag in the **Parameter** field, and click **Add** and then click **OK**.

If your application experiences some locking problems, you can open the SQL error log and see the types of locks that were placed and the clients that were involved.

To open the SQL error log:

1. Open the Startup Parameters window as described earlier.
2. Use the scroll bar in the **Existing Parameters** field to see which of the parameters is the error log. This also tells you the path to where the error log is stored. This is normally `C:\Program Files\Microsoft SQL Server\MSSQL\log\ERRORLOG`.
3. Locate the error log and open it in Notepad.



Keys, Queries and Performance

When you write a query that searches through a subset of the records in a table, you should always carefully define the keys both in the table and in the query so that Microsoft Navision can quickly identify this subset. For example, the entries for a specific customer will normally be a small subset of a table containing entries for all the customers.

If Microsoft Navision can locate and read the subset efficiently, the time it will take to complete the query will only depend on the size of the subset. If Microsoft Navision cannot locate and read the subset efficiently, performance will deteriorate. In the worst-case scenario, Microsoft Navision will read through the entire table and not just the relevant subset. In a table containing 100,000 records, this could mean taking either a few milliseconds or several seconds to answer the query.

To maximize performance, you must define the keys in the table so that they facilitate the queries that you will have to run. These keys must then be specified correctly in the queries.

For example, you would like to retrieve the entries for a specific customer. To do this, you apply a filter to the **Customer No.** field in the Cust. Ledger Entry table. In order to run the query efficiently on SQL Server, you must have defined a key in the table that has Customer No. as the first field. You must also specify this key in the query.

The table could have these keys:

```
Entry No.
Customer No.,Posting Date
```

The query could look like this:

```
SETCURRENTKEY("Customer No."); SETRANGE("Customer
No.",“1000”); IF FIND('-') THEN
REPEAT
UNTIL NEXT = 0;
```

You should define keys and queries in the same way when you are using Microsoft Navision Database Server. However, Microsoft Navision Server can run the same query almost as efficiently if Customer No. is not the first field in the key. For example, if you have defined a key that contains **Country Code** as the first field and **Customer No**. as the second field, and if there are only a few different country codes used in the entries, it will only take a little longer to run the query.

The table could have these keys:

```
Entry No.
Country Code, Customer No.,Posting Date
```

The query could look like this:

```
SETCURRENTKEY("Country Code","Customer No.");
SETRANGE("Customer No.",'1000'); IF FIND('-') THEN
REPEAT
UNTIL NEXT = 0;
```

But SQL Server will not be able to answer this query efficiently and will read through the entire table.

In conclusion, SQL Server makes stricter demands than Microsoft Navision Database Server on the way that keys are defined in tables and on the way they are used in queries.

This section has been copied from the Application Designer’s Guide.

Locking in Microsoft Navision – A Comparison of Microsoft Navision Database Server and SQL Server

The following information explains the differences and similarities in the way that locking is carried out in the two database options for Microsoft Navision, Microsoft Navision Database Server, and SQL Server.

IMPORTANT: *The following information only covers the default transaction type UpdateNoLocks for the SQL Server Option for Microsoft Navision. For information about the other transaction types, see the online C/SIDE Reference Guide.*

Both Server Options Locking

In the beginning of a transaction, the data that you read will not be locked. This means that reading data will not conflict with transactions that modify the same data. If you want to ensure that you are reading the latest data from a table, you must lock the table before you read it.

Locking Single Records

Normally, you must not lock a record before you read it even though you may want to modify or delete it afterwards. When you try to modify or delete the record, you will get an error message if another transaction has modified or deleted the record in the meantime. You receive this error message because C/SIDE checks the timestamp that it keeps on all of the records in a database and detects that the timestamp on the copy you have read is different from the timestamp on the modified record in the database.

Locking Record Sets

Normally, you lock a table before reading a set of records in that table if you want to read these records again later to modify or delete them. You must lock the table to ensure that another transaction does not modify these records in the meantime.

You will not receive an error message if you do not lock the table even though the records have been modified as a result of other transactions being carried out while you were reading them.

Minimizing Deadlocks

To minimize the amount of deadlocks that occur, you must lock records and tables in the same order for all transactions. You can also try to locate areas where you lock more records and tables than you actually need, and then diminish the extent of these locks or remove them completely. This can prevent conflicts on these records and tables.

If this does not prevent deadlocks, you can, as a last resort, lock more records and tables to prevent transactions from running concurrently.

If you cannot prevent the occurrence of deadlocks by programming, you must run the deadlocking transactions separately.

Locking in Microsoft Navision Database Server

Data that is not locked will be read from the same snapshot (version) of the database.

If you call LOCKTABLE or a modifying function (for example, INSERT, MODIFY or DELETE) on a table, the whole table will be locked.

Locking Record Sets

As mentioned previously, you will normally lock a table before reading a set of records in that table if you want to read these records again later to modify or delete them. With Microsoft Navision Database Server, you can choose to lock the table with LOCKTABLE(TRUE,TRUE) after reading the records for the first time instead of locking with LOCKTABLE before reading the records for the first time.

When you try to modify or delete the records, you will receive an error message if another transaction has modified the records in the meantime.

You will also receive an error message if another transaction has inserted a record into the record set in the meantime. However, if another transaction has deleted a record from the record set in the meantime, you will not be able to notice this change. The purpose of locking with LOCKTABLE(TRUE,TRUE) after reading the records for the first time is to postpone the table lock that Microsoft Navision Database Server puts on the table. This improves concurrency.

Locking in SQL Server

When data is read without locking, you will get the latest (possibly uncommitted) data from the database.

If you call Rec.LOCKTABLE, nothing will happen right away. However, when data is read from the table after LOCKTABLE has been called, the data will be locked.

If you call INSERT, MODIFY, or DELETE, the specified record will be locked immediately. This means that two transactions, which insert, modify or delete separate records in the same table will not conflict. Furthermore, locks will also be placed whenever data is read from the table after the modifying function has been called.

It is also important to note that even though SQL Server initially puts locks on single records, it can also choose to escalate a single record lock to a table lock. It will do so if it determines that the overall performance will be improved by not having to set locks on individual records. The improvement in performance must outweigh the loss in concurrency that this excessive locking causes.

If you specify what record to read, for example, by calling Rec.GET, that record will be locked. This means that two transactions, which read specific, but separate records in a table will not cause conflicts.

If you browse a record set (that is, read sequentially through a set of records), for example, by calling Rec.FIND("-") or Rec.NEXT, the record set (including the empty space) will be locked as you browse through it. However, the locking implementation used in SQL Server will also cause the record before and the record after this record set to be locked. This means that two transactions, which just read separate sets of records in a table, will cause a conflict if there are no records between these two record sets. When locks are placed on a record set, other transactions cannot put locks on any record within the set.

Note that C/SIDE decides how many records to retrieve from the server when you ask for the first or the next record within a set. C/SIDE then handles subsequent reads with no additional effort, and fewer calls to the server will give better performance. In addition to improving performance, this means that you cannot precisely predict when locks are set when you browse.

NOTE: *Microsoft Navision tables that have keys defined for SumIndexFields cause additional tables to be created in SQL Server to support SIFT functionality. One table is created for each key that contains SumIndexFields. When you modify a Microsoft Navision table that has keys defined for SumIndexFields, modifications can be made to these SQL Server tables. When this happens, there is no guarantee that two transactions can modify different records in the Microsoft Navision table without causing conflicts.*

Locking Differences in the Code

A typical use of LOCKTABLE(TRUE,TRUE) in Microsoft Navision Database Server is shown in the first column of the table below. The equivalent code for the SQL Server Option is shown in the second column. The code that works on both servers is shown in the third column.

The RECORDLEVELLOCKING property is used to detect whether record level locking is being used. If this is the case, then you are using the SQL Server Option for Microsoft Navision. This is currently the only server that supports record level locking.

Microsoft Navision Server	**SQL Server**

IF Rec.FIND("-") THEN
REPEAT
UNTIL Rec.NEXT = 0;
Rec.LOCKTABLE(TRUE,TRUE);
IF Rec.FIND("-") THEN
REPEAT
Rec.MODIFY;
UNTIL Rec.NEXT = 0;
	Rec.LOCKTABLE;
	IF Rec.FIND("-") THEN
	REPEAT
	UNTIL Rec.NEXT = 0;
	IF Rec.FIND("-") THEN
	REPEAT
	Rec.MODIFY;
	UNTIL Rec.NEXT = 0;

Both Servers

IF Rec.RECORDLEVELLOCKING THEN
Rec.LOCKTABLE;
 	
IF Rec.FIND("-") THEN
REPEAT
UNTIL Rec.NEXT = 0;
 	
IF NOT Rec.RECORDLEVELLOCKING THEN
Rec.LOCKTABLE(TRUE,TRUE); IF Rec.FIND("-") THEN
REPEAT
Rec.MODIFY;
UNTIL Rec.NEXT = 0;

This section has been copied from the Application Designer's Guide.

Configuration Parameters

You can configure a Microsoft Navision database by creating a SQL Server table configuration parameter table and entering parameters into the table that will determine the behavior of Microsoft Navision when it is using this database.

In the database, create a table, owned by dbo:

CREATE TABLE [ndodbconfig] (config VARCHAR(512) N0T NULL)

(You can add additional columns to this table, if necessary. The length of the config column should be large enough to contain the necessary configuration values, as explained below, but need not be 512.)

There is one record in this table for each parameter that is required.

The following sections describe the parameters that you can enter into this table. It is possible to force SQL Server to use a particular index when executing queries for FIND("-"), FIND("+"), FIND("=") and GET statements. This can be used as a workaround when SQL Server's Query Optimizer picks the wrong index for a query.

Index Hinting

Index hinting can help avoid situations where SQL Server's Query Optimizer chooses an index access method that requires many page reads and generates long-running queries with response times that vary from seconds to several minutes. Selecting an alternative index can give instant "correct" query executions with response times of milliseconds. This problem usually occurs only for particular tables and indexes that contain certain data spreads and index statistics.

In the rare situations where it is necessary, you can direct Microsoft Navision to use index hinting for such problematic queries. When you use index hinting, Microsoft Navision adds commands to the SQL queries that are sent to the server. These commands bypass the normal decision making of SQL Server's Query Optimizer and force the server to choose a particular index access method.

NOTE: *This feature should only be used after all the other possibilities have been exhausted, for example, updating statistics, optimizing indexes, or reorganizing column order in indexes.*

The index hint syntax is:

```
IndexHint=<Yes,No>;Company=<company name>;Table=<table
name>;Key=<keyfield1,keyfield2,...>; Search Method=<search
method list>;Index=<index id>
```

Each parameter keyword can be localized in the "Driver configuration parameters" section of the .stx file.

The guidelines for interpreting the index hint are:

- If IndexHint=No, the entry is ignored.
- All the keywords must be present or the entry is ignored.
- If a given keyword value cannot be matched, the entry is ignored.
- The values for the company, table, key fields, and search method must be surrounded by double-quotes to delimit names that contain spaces, commas, etc.
- The table name corresponds to the name supplied in the Object Designer (not the Caption name).
- The key must contain all the key fields that match the required key in the Keys window in the Table Designer.
- The search method contains a list of search methods used in FIND statements, that must be one of "-", "+", "=", or "!" (for the C/AL GET function).

The index ID corresponds to a SQL Server index for the table: 0 represents the primary key; all other IDs follow the number included in the index name for all the secondary keys. Use the SQL Server command sp_helpindex to get information about the index ID associated with indexes on a given table. In this example we are looking for index information about the Item Ledger Entry table:

sp_helpindex "CRONUS International Ltd_$Item Ledger Entry"

When Microsoft Navision executes a query, it checks whether or not the query is for the company, table, current key, and search method listed in one of the IndexHint entries. If it is, it will hint the index for the supplied index ID in that entry.

Note that:

- If the company is not supplied, the entry will match all the companies.
- If the search method is not supplied, the entry will match all the search methods.
- If the index ID is not supplied, the index hinted is the one that corresponds to the supplied key. This is probably the desired behavior in most cases.
- If the company/table/fields are renamed or the table's keys redesigned, the IndexHint entries must be modified manually.

Here are a few examples that illustrate how to add an index hint to the table by executing a statement in Query Analyzer:

Example 1

```
INSERT INTO [$ndo$dbconfig] VALUES
('IndexHint=Yes;Company="CRONUS International
Ltd.";Table="Item Ledger Entry";Key="Item No.","Variant
Code";Search Method="-+";Index=3')
```

This will hint the use of the $3 index of the CRONUS International Ltd_$Item Ledger Entry table for FIND("-") and FIND("+") statements when the Item No.,Variant Code key is set as the current key for the Item Ledger Entry table in the CRONUS International Ltd. company.

Example 2

```
INSERT INTO [$ndo$dbconfig] VALUES
('IndexHint=No;Company="CRONUS International
Ltd.";Table="Item Ledger Entry";Key="Item No.","Variant
Code";Search Method="-+";Index=3')
```

The index hint entry is disabled.

Example 3

```
INSERT INTO [$ndo$dbconfig] VALUES
('IndexHint=Yes;Company="CRONUS International
Ltd.";Table="Item Ledger Entry";Key="Item No.","Variant
Code";Search Method="-+";Index=')
```

This will hint the use of the Item No.,Variant Code index of the CRONUS International Ltd_$Item Ledger Entry table for FIND("-") and FIND("+") statements when the Item No.,Variant Code key is set as the current key for the Item Ledger Entry table in the CRONUS International Ltd. company. This is probably the way that the index-hinting feature is most commonly used.

Example 4

```
INSERT INTO [$ndo$dbconfig] VALUES
('IndexHint=Yes;Company=;Table="Item Ledger
Entry";Key="Item No.","Variant Code";Search Method="-
+";Index=3')
```

This will hint the use of the $3 index of the CRONUS International Ltd_$Item Ledger Entry table for FIND("-") and FIND("+") statements when the Item No.,Variant Code key is set as the current key for the Item Ledger Entry table for all the companies (including a non-company table with this name) in the database.

Example 5

```
INSERT INTO [$ndo$dbconfig] VALUES
('IndexHint=Yes;Company="CRONUS International
Ltd.";Table="Item Ledger Entry";Key="Item No.","Variant
Code";Search Method=;Index=3')
```

This will hint the use of the $3 index of the CRONUS International Ltd_$Item Ledger Entry table for every search method when the Item No.,Variant Code key is set as the current key for the Item Ledger Entry table in the CRONUS International Ltd. company.

When Microsoft Navision is reading data from tables, it places forced ROWLOCK hints by default. These rowlock hints prevent SQL Server from automatically determining the granularity (row, page or table) of the locks that it places. This can lead to a high locking overhead on the server, even though concurrency is optimum.

Lock Granularity

To allow SQL Server to determine the granularity of the locks that it places, the DefaultLockGranularity parameter can be used in the database configuration table.

The syntax of the DefaultLockGranularity parameter is:

```
DefaultLockGranularity=<Yes,No>
```

When the parameter is Yes, SQL Server will choose the granularity of the locks that it places. When the parameter is No, Microsoft Navision will override SQL Server and place ROWLOCKs.

This section has been copied from the Application Designer's Guide.

Object List

This section contains a list of all the objects in the fob files described in the Performance Troubleshooting Guide.

Performance Troubleshooting Object List

The following tables list all the objects contained in the .fob files described in this document:

Session Monitor (Microsoft Navision Server) .fob		
Table	150010	Session Monitor Setup
Table	150011	Session Information
Form	150010	Session Monitor (Microsoft Navision Server)
Form	150011	Session Monitor Lines
Form	150012	Session Monitor Setup (SQL)
Form	150013	Sessions
Codeunit	150010	Session Monitor Mgt.
Table	150012	Session Monitor Setup (SQL)
Table	150013	Session Information (SQL)
Table	150014	Session (SQL)
Form	150014	Session Monitor (SQL Server)
Form	150015	Session Monitor Lines (SQL)
Form	150016	Session Monitor Setup (SQL)
Form	150017	Sessions (SQL Server)
Codeunit	150011	Session Monitor Mgt. (SQL Server)

Activity Log .fob		
Table	150000	Activity Log Entry
Table	150001	Table Size Change
Form	150000	Activity Log
Form	150001	Table Size Changes
Sample Use of Activity Log .fob		
Codeunit	150000	Sample Use of Activity Log

Client Monitor .fob		
Table	150020	Client Monitor
Table	150021	Client Monitor Setup
Table	150022	Code Coverage Line
Table	150023	Transaction
Table	150024	Transaction Lock

Client Monitor .fob		
Table	150025	Locking Rule
Table	150026	Potential Deadlock
Table	150027	Locking Rule Violation
Form	150020	Client Monitor
Form	150021	Client Monitor Setup
Form	150022	Client Monitor (Key Usage)
Form	150023	Client Monitor (Cache Usage)
Form	150024	Client Monitor (Multi-User)
Form	150025	Transactions
Form	150026	Transaction Locking Order
Form	150027	Transactions (Locking Rules)
Form	150028	Locking Rule Violations
Form	150029	Locking Order Rules
Form	150030	Potential Deadlocks (Microsoft Navision)
Form	150031	C/AL Code – Code Coverage
Form	150032	C/AL Code – Adv. Code Coverage
Report	150020	Code Coverage
Dataport	150020	Export Client Monitor
Dataport	150021	Export/Import Code Coverage
Dataport	150022	Import/Export Locking Rules
Codeunit	150020	Client Monitor Mgt.
Codeunit	150021	Client Monitor to C/AL Code
Codeunit	150022	Transaction Mgt.
Codeunit	150023	Transaction to C/AL Code
Codeunit	150024	Transaction Lock to C/AL Code
Codeunit	150025	Locking Order Rules Mgt.
Codeunit	150027	Deadlock to C/AL Code
Codeunit	150028	Deadlock to C/AL Code 2

Test your Knowledge – Troubleshooting Microsoft Navision

1. Name the three typical server calls that generate status information.

2. What is the purpose of using the Client Monitor? What information is contained in the Client Monitor window?

3. What are the most common causes of performance problems in Microsoft Navision?

4. Describe deadlocks, how they occur and how they can be resolved.

5. Bottlenecks can occur in the software or the hardware. Describe how bottlenecks can be handled in both instances to resolve performance problems.

6. Discuss Locking in Microsoft Navision. What is the main difference between Locking in the Microsoft Navision Database Server and the SQL Server?

7. What is Index Hinting and how is it used in Microsoft Navision?

How do you maximize the performance of a query?

9. What is the purpose of trace flags 1204 and 3605 in the SQL Server Error Log?

Quick Interaction: Lessons Learned

Take a moment to write down three Key Points you have learned from this chapter:

1.

2.

3.

APPENDIX A: TERMINOLOGY LIST

The section contains an alphabetical list of most of the technical computer terms, which are used in this manual.

Active Directory: The directory service used in Microsoft® Windows Server™ 2000 and in Microsoft® Windows Server™ 2003. It allows any object on a network to be tracked and located. Active Directory provides the ability to build applications that give a single point of access to multiple directories in a network environment.

Authentication: The process by which a user's ID and password is verified when he logs on to a computer or network. This can be done by having the user enter an ID and password when he logs on or uses a smart card to identify himself to the system. This manual contains references to two kinds of authentication: windows authentication and database server authentication. See login.

ADCS Device: Automated Data Capture System: Any device such as a bar code reader or optical character reader that mechanizes the entry of information into an information system.

Backup: To keep your data safe, you should frequently make a copy of it. This copy is called a backup.

Batch job: A program or set of commands that can run without any interaction from the user.

BLOB (Binary Large Object): An object that can store anything but is normally used to store graphics, memos, and other Microsoft® Business Solutions–Navision® objects.

Cache: Cache means a store or hiding place. As a computer term, it refers to part of the computer's memory. The cache serves as a way station for data that is on its way to or from the hard disk and needs to be processed by the CPU. Cache can be accessed quickly and using it makes less work for the hard disk. The bigger the cache, the better the performance. In Microsoft Navision, Cache is a program property representing the program's database cache. See also object cache.

Client: A client is one of the computers in a network from which an individual user works with the common data that is found on the server. See single-user.

Client/Server: This is the term for a particular way in which several computers (a server and some clients) work together in a network. Each computer does some of the work itself (in contrast to old-fashioned systems, in which the server did all the work and was therefore heavily loaded and slow). The client is generally optimized for user interaction and the server provides the centralized multiuser functionality. A client/server system is faster than one in which all the work is done by the server.

Command line: You can type command lines (such as copy, del, and rename) in response to a prompt from an operating system (MS-DOS, OS/2 or UNIX) and specify parameters for running programs. A database server can be started from a command line.

Commit Cache: When the Microsoft Navision database is updated by write transactions or deletions (for example, when you are posting), the screen is locked for a certain period. When you use commit cache, Microsoft Navision can register write transactions a soon as they are logically present in the database, before they are physically written to the disk. In this way, the screen is not locked as long as it would otherwise be, and your work is speeded up. The function is especially useful in larger networks.

Compression: A method for reducing the size of files so that they can be stored in less space. See Date Compression.

Configuration: See setup.

CPU: The CPU (Central Processing Unit) is the "brain" of a computer. It is the element that contains all the computing power and makes it possible to run programs.

C/SIDE® (Client/Server Integrated Development Environment): C/SIDE is the name of the development system in which Microsoft Navision is programmed. It is partly a programming language and partly a collection of tools that standardize routines, functions and codes, and in other ways take care of trivial tasks.

Data file: A file that contains data such as tables, rows, and stored procedures.

Database (relational): This is a collection of tables containing rows and columns used for storing data. A database is organized so that operations can be carried out on the information it contains, such as sorting, searching, and recombining.

Database login: An entry in the User table that Microsoft Navision uses to verify the user ID and password that the user enters in order to gain access to a Microsoft Navision database. Database logins are given users who access the system with Database Server authentication. See login.

Database option: This allows the user to determine the characteristics of a database such as the handling of certain non-logged operations.

Database server: The computer in a network that contains and manages shared databases is called a database server.

Database server authentication: This type of authentication is used when the network does not support Windows authentication or the network administrator has chosen not to use Windows authentication. In the Microsoft® SQL Server® Option for Microsoft Navision, it is based on Microsoft's SQL Server authentication. See authentication.

Database user account: The account on SQL Server that contains information about the permissions that individual users have been granted in a particular database. The SQL Server Option for Microsoft Navision uses it to store information about whether or not you can access this particular database. All the information about Microsoft Navision object permissions is stored in Microsoft Navision.

Date compression: A method used by Microsoft Navision for reducing the amount of space taken up by data. Data can be combined according to various time periods, for example, day, month or year.

DBMS (Database Management System): A layer of software between the physical database and the user. The DBMS manages all requests for database action (for example queries or updates) from the user. In addition, a DBMS permits centralized control of security and data integrity requirements.

Dedicated server: A dedicated database server performs only tasks involving the database and no other network tasks.

Disk cache: This is a portion of the computer's memory that is set aside to temporarily hold information read from or written to the disk.

Domain: A domain is a single security boundary of a Windows computer network. Active Directory is made up of one or more domains. On a stand-alone workstation, the domain is the computer itself. A domain can span more than one physical location. Every domain has its own security policies and security relationships with other domains. When multiple domains are connected by trust relationships and share a common schema, configuration and global catalog, you have a domain tree. Multiple domain trees can be connected together into a forest.

Domain controller: With Windows 2000 or later, a server in a domain can be a domain controller. This is one of two roles that a server in a domain can have. The other server role is as a member server.

A domain controller contains matching copies of the user accounts and other Active Directory data in a given domain. It is possible to change the role of a server back and forth from domain

Domain tree: When multiple domains are connected by trust relationships and share a common schema, configuration and global catalog, you have a domain tree. Multiple domain trees can be connected together in a domain forest.

ECC RAM: Error Correction Code RAM. RAM that implements a method of detecting and correcting errors that may occur.

Enterprise Manager: A program that is part of the SQL Server Client tools and is used for organizing and maintaining SQL Server databases.

External cache: See level 2 cache.

File group: A database specific collection consisting of one or more files that have been grouped together to form a single unit which is used for allocation and administration purposes.

Forest: A forest is a collection of Active Directory domain trees. Forests serve two main purposes: to simplify user interaction with the directory and to simplify the management of multiple domains. See domain tree.

Handheld: A device similar to that as described in the definition of ADCS, but using a display as the user interface to display data and as a means to guide them.

Hard disk controller: Controls access to one or more disk drives, CD-ROMs and diskette drives.

Hosts file: This file contains the mappings of IP addresses to host names.

Host name: The name of the computer in the network running TCP/IP. Connected to this name is an address that is the IP address of the computer.

Index: In a database an index is a list or table that contains reference information that points to stored data. Indexes are often used to implement keys.

IP address: Internet Protocol address. Every computer running in a TCP/IP network has a host name and a corresponding IP address. Other computers in the network use this IP address to communicate with your computer. There is a host name for each IP address; the host name and the IP address are listed in the hosts file.

Key: Fast and efficient searches in a database require a sorting system. The information you are looking for is located in fields. By assigning an order to some of these fields, you can determine the order in which the fields will be searched. This field combination is called a key (or sorting key). A key can consist of a single field. Microsoft Navision comes with a number of keys, but you can set up more. Updating the keys consumes a lot of the computing power, so you can improve the performance by disabling some keys.

Lazy-write cache: This is a type of cache that holds information in the cache between the disk and the disk controller. When the computer writes something to the hard disk, the disk controller actually writes the information to the lazy-write cache and not to the disk. The data is then written to the disk when demand on the system is low.

Level 2 cache: Also called external cache. Level 2 cache is a cache that exists between the CPU and the memory. This is a very fast type of cache that improves the system's performance.

Login: In Microsoft Navision a login is a user account. It contains the information about the user's ID and password that the system uses to verify the user when they log on to the database. Microsoft Navision has two types of logins: Windows logins and database logins. They are created either in the Windows Login table (for Windows logins) or the User table (for database logins) and must possess a valid Windows account or SQL Server login, respectively.

Log on/off: To log on or off means to open or close a Microsoft Navision database.

Master database: A database used by SQL Server to store information about the server and all the other databases that exist on the server. This information includes the location of the primary files that contain the startup information for the databases.

Memory: Computer memory refers to Random Access Memory (RAM). When a new program is started, the operating system places the program in the memory so that the central processing unit (CPU) can work with it.

Menu: a menu is a collection of menu items, which can be organized in menu groups and displayed as a tree.

Menu button: button at the bottom of the Navigation Pane that selects which menu is displayed.

MenuSuite: a collection of menus, for example a user's menu suite is all of the menus assigned to the user.

Migration: This is the process by which you move your data (database) from one version of a program to a newer version of the same program.

Model database: The database that SQL Server uses as the template for all new databases.

Multiprotocol: This is one of the network types that are necessary for different programs, including Microsoft Navision, to be able to communicate with SQL Server. The network type uses a particular network protocol for communication.

Multiuser installation: A multiuser installation is one in which Microsoft Navision is used on several computers connected in a network and using common data on a server, also known as a client/server installation.

Named pipes: This is a network type that uses a particular network protocol for communication.

NetBIOS: NetBIOS is an abbreviation for Network Basic Input Output System. This is one of the protocols (sets of rules) that are necessary for different programs, including Microsoft Navision, to be able to communicate in a network. Another network protocol is TCP/IP.

Net type: This is a type of network library that uses a network protocol for communication between a client and a server.

Network server: The computer in a network that provides shared services to the other computers is called a network server. If it contains the Microsoft Navision database, it is also called a database server. We do not recommend that you use one computer as the server for both the network and Microsoft Navision because each role requires a large amount of computing power.

Nondedicated database server: A non-dedicated database server is a computer that not only manages the database program but also runs other programs.

Object: The Microsoft Navision application consists of a number of objects, including: tables, forms, reports, codeunits, and data ports.

Object cache: The object cache, like cache, allows the program to work faster. The object cache is used only on clients. Its task is to store the code, descriptions and windows that will be used on the client, so they only need to be retrieved once. Using object cache requires the client computer to have enough memory to store the objects while they are used.

Parameter: See program property.

Plug-in: An accessory software program that extends the capabilities of an existing application.

Primary data file: This data file contains the startup information for a database and can also be used for storing data. Every database has one primary data file.

Primary key: The column or combination of columns that uniquely identify a row in a table.

Program property: A program property is something you can use to set various system values that affect how the program works. For example, in Microsoft Navision you can use the Database program property to specify the database that will be used. Many program properties can be set by clicking Tools and then Options on the menu bar. In addition, most of them can be entered in the **Target** field or the command line used to start the program.

Protocol: A protocol is a set of rules or standards that determines how computers communicate with each other and how data can be exchanged. NetBEUI and TCP/IP are examples of such a protocol.

RAID (Redundant Array of Inexpensive (or Independent) Disks): A storage system based on a disk array that holds a certain amount of redundant information. The redundant information can be used to detect and (in some configurations) correct errors that may occur.

RAM: See memory.

Record level security: Record level security is a system that allows you to limit the access that users have to the information that is stored in the tables so that they can only gain access to specific records in the tables.

Role: This is an administrative unit in Microsoft Navision that is used for determining the permissions that are given to individual users. Each role has specific rights and all the users that are mapped to a role inherit the rights that belong to that role. Roles can be given to both types of Microsoft Navision login and to Active Directory security groups. There is also an item called Roles on the Security submenu of the Tools menu. Note that the old concept of groups has been replaced by roles.

Rollback: A feature of the Microsoft Installer service. The system restores itself to its initial state if the installation should fail. It is only available during installation. After an application has been installed, an uninstallation is required to remove the application.

Schema: The Active Directory schema is the set of definitions that defines the kinds of objects and the types of information about those objects, which can be stored in Active Directory. The definitions are themselves stored as objects, so that Active Directory can manage the schema objects with the same object management operations that are used for managing the rest of the objects in the directory.

Security accounts: SQL Server stores information in these accounts about who has access to the various databases and which fixed server and database roles they have been assigned.

Security identifier (SID): A unique number that identifies user, group, and computer accounts in Windows. Internal processes in Windows refer to an account's SID rather than the account's user or group name.

Server: A server is the computer that serves the other computers in a network. A server carries out several functions. It can be a database server, network server, and/or printer server. Thus, it enables the computers in the network to use the shared equipment or data. See also these terms: dedicated, non-dedicated, database server, network server.

Service: A process under Windows that performs a specific system function or executes a specific program. This execution is started as soon as Windows is loaded, before any user is logged onto the system.

Services file: This file contains port numbers for services.

Service connection point (SCP): The service connection point is an object that contains information about the server, such as binding information, the Domain Name System (DNS) name and type, product, version, globally unique identifier of the object (GUID), and so on. The Microsoft Navision Database Server must have an SCP object that contains the necessary server information.

Session: A session is a connection between a client and a server. A running instance of Microsoft Navision that is connected to a server is an example of a session.

Setup: The setup of a computer is the way in which the system is adjusted to work with other hardware or software elements, such as a printer or a particular program. The setup is also called a configuration. The setup of a program is the way in which its functions are modified to fit the needs of a particular user.

SIFT: This is an acronym for SumIndexField Technology and is used to generate sums for fields that contain decimals. This is useful for getting such information as the balance of an account, the number of items in stock, the invoiced total outstanding.

Single sign-on: A key feature of Windows authentication is its support of single sign-on. This is a process that allows a user with a domain account to log on to a network once with his or her password and to gain access to any computer in the domain.

Single-user installation: A single-user Microsoft Navision installation is one in which Microsoft Navision is used on an independent computer. The difference between a client and a singleuser installation is that a single-user uses its own database, whereas a client is connected to a server computer and uses a database that is stored on this server.

SMS (System Management Server): If you have installed SMS software on every computer in the network, you can manage the whole network from the computer at the top of the network hierarchy.

Sorting key: See key.

Swapping: This is a process used by a computer to enable it to use more memory than is physically present. If the active programs together use more memory than the computer has, the least-used data is "swapped" (moved) to the hard disk in a "swap file." When this data is needed again, it is retrieved back into the memory, and other data is placed in the swap file on the hard disk.

Table: This is an object in a database that is used for storing data. A table consists of rows and columns.

Target field: The field in which you specify the location of the program that you want to run, along with any parameters that you want to set.

TCP/IP: Like NetBIOS, this is a network protocol.

TCP/IP Sockets: This is a network type that uses a particular network protocol for communication.

Transaction log: The set of transaction log files. These are the database specific files that SQL Server uses to store detailed information about every change that is made to the database. In SQL Server, you can apply a backup of the transaction log to recover data after hardware failure.

Upload: The process by which a copy of a file is transferred from a local computer to a remote computer.

User: See login.

View: This is a type of virtual table that can be used in SQL Server and is an alternative way of looking at data from one or more tables in a database. A view generally consists of a subset of columns or a filtered set of rows from one or more tables.

Windows authentication: The process by which the system validates the users identity. In this manual, we refer to both the single sign-on and the unified login as Windows authentication. See authentication.

Windows login: An entry in the Windows Login table that Microsoft Navision uses to verify the user ID and password that the user. Windows logins are given users who access the system with Windows authentication. See login.

APPENDIX B: TEST YOUR KNOWLEDGE AND TEST YOUR SKILLS SOLUTIONS

Test Your Knowledge – Hardware and Software Requirements (Chapter 1)

1. Microsoft® Business Solutions–Navision® can be used in single-user or multiuser installations.

2. Besides having a powerful computer for the server, the other server aspects that require further consideration are the hard disk and controller, the RAID system, the memory, the network adapter and the CPU.

3. The acronym RAID stands for Redundant Array of Independent Disks.

4. Microsoft Navision is a two-tier OR client-server solution. (Either answer is acceptable)

5. The Microsoft Navision Application Server is a middle-tier server that supports an n-tier architecture.

Test Your Knowledge – Microsoft Navision Architecture (Chapter 2)

1. Microsoft Navision has two database options. What are they?

– The Standard Microsoft Navision database
– The SQL Server database

2. Besides being responsible for the user interface, what else does the Microsoft Navision client do?

The client can connect directly to a standard database file without going through the server. This is the stand-alone setup that was mentioned earlier. The client is also responsible for executing all the business logic. The client reads objects from the database and is also responsible for running the objects and controlling their behavior. Most of the Microsoft Navision application runs on the individual clients.

3. Even though the Microsoft Navision client does most of the work, what are some of the responsibilities of the server?

– controls the number of users that can connect to the database at one time.
– controls access to the data through locking.

- keeps track of all the read and write transactions performed by each user.
- sends data to each client, as requests are made.
- performs all the key-based filtering and calculates the SumIndexFields.
- caches data that can be requested again.

4. What are the different server options, and what are some of their differences?

Microsoft Navision Database Server
Microsoft® SQL Server®

- the way you create a database.
- the backup facilities that are available.
- the ability to access the data in the database with third party tools.
- the way that SIFT™ works.
- performance monitoring
- scalability
- multi-processor support

5. The Microsoft Navision security system, even though it is a homogenous integrated system, can be said to consist of four different levels of security. What are they?

- Database Level Security
- Company Level Security
- Object Level Security
- Record Level Security

6. There are three important things to remember about the Microsoft Navision Security System. Can you remember them?

- The Microsoft Navision security system is initiated when you create the first database login. Therefore, it is imperative that the first login you create is a database login. Furthermore, the first login you create must be that of a superuser. The superuser then owns and administers all access to this database from within Microsoft Navision.
- You can only grant permissions to other users that you already possess yourself. Therefore, we recommend that the user who administers security in Microsoft Navision should be a superuser. Until you create the first database login, anyone can carry out all the transactions they wish in a Microsoft Navision database.
- One of the first things that the superuser should do is create logins for the other people who will have access to the database and grant the appropriate permissions to these users.

7. What is the difference among Business Areas, Functional Areas, and Granules?

Each business area covers a large amount of functionality. Each granule represents a smaller area of functionality.

For example: Business Area = Finance
Functional Area = General Ledger
Granule = Account Schedules

Test Your Knowledge - Microsoft Navision Client Installation (Chapter 3)

True	False		
☒	❑	1.	Microsoft Navision comes with a standard setup that enables it to be used immediately.
❑	☒	2.	The three types of installations are: Minimum, COMPLETE, and Custom.
❑	☒	3.	If you have Microsoft Navision 2.5 or earlier installed on the computer, the Installation Wizard will ask you to uninstall the old version before you can install Microsoft Navision.
☒	❑	4.	The System Setup file, or ZUP file, remembers the various settings that are used by different clients and single-users.
☒	❑	5.	You can customize the system setup by changing the settings of the various program properties.
❑	☒	6.	If ntauthentication=YES on the Target line, Microsoft Navision will start and automatically connect to the server and open the database specified.
❑	☒	7.	Program properties CAN be entered in ANY order.
☒	❑	8.	Microsoft Navision will automatically open the database and company that you were last working on when you reopen the program.

❑ ☒ 9. The Change License functionality is a TEMPORARY change to which license is used.

☒ ❑ 10. If the license on the server expires, you will not be able to access the server.

Test Your Knowledge – Microsoft Navision Database Server (Chapter 4)

1. What are the two different types of multi-user installations, and how are they setup?

– A classic client/server installation, where the server is installed on one computer and the clients are installed on other computers in the network.
– A multiuser installation, where both the server and the client programs are installed on a file server and the clients are run remotely from other computers in the network.

2. There are four ways to run Microsoft Navision database servers with the same or different databases. What are they?

Two or more servers, same database

Two or more servers, different databases

Two servers on the same computer, both using TCP/IP

Two servers on the same computer, both using NetBIOS

3. There are setup properties that apply to both the client and server. What are they five setup properties that apply only to the server?

DBMS Cache

Stoptime

Sessions

Install as Service

Uninstall as Service

4. What is the Microsoft Navision Database Server Manager?

The Microsoft Navision Database Server Manager is a Microsoft Management Console snap-in that makes it possible for you to manage Microsoft Navision Database Servers across a domain. With the Microsoft Navision Database Server Manager, you can see and reconfigure a number of Microsoft Navision Database Server properties.

5. What is the maximum size of a database? Can a database be broken into several small files? Why would this be done?

The maximum size of a database is 128 GB. Yes, it can be broken into several databases and placed on different hard drives. Dividing your database into several files on several (physical) disks lets you utilize disk space optimally. It also reduces the access time to the database, which improves the programs performance.

Test Your Knowledge – Microsoft Navision SQL Server Option (Chapter 5)

1. You must add an extended stored procedure to each instance of SQL Server that you want to access using Windows authentication. However, if you select the Complete Installation option of Microsoft Navision and are going to access the default SQL Server instance on your local computer with Windows authentication, the installation program will add the above for you.

2. In order to work with the SQL Server Option for Microsoft Navision, you must upload the license file you want to use to all the SQL Servers you want to access – instead of copying it to the individual clients.

3. The six tabs on the New Database window are: General, Database Files, Transaction Log Files, Collation, Options, and Integration.

4. The placement of data and transaction log files is important in determining the efficiency and integrity of a database.

5. The server collation specifies the rules by which character data is stored and compared.

6. Microsoft SQL Server has two levels of security: server security and database security.

7. The Alter Database function allows you to change the database properties whenever you think it's necessary.

8. SQL Server employs two types of authentication. These correspond to the two types of logins that can be created in SQL Server: Window logins and SQL Server logins.

9. The synchronization process compares the Microsoft Navision login tables with the security system in SQL Server.

10. To create or alter a database, the administrator must be a member of either the sysadmin or dbcreator SQL server roles.

Test Your Knowledge – Microsoft Navision Application Server and ADCS (Chapter 6)

1. Microsoft Navision Application Server is a middle-tier server, which executes business logic without user intervention.

2. The user cannot use database authentication when logging on to Microsoft Navision Application Server.

3. The Microsoft Navision Automated Data Capture System is aimed at companies that need to use handheld devices in their warehouse processes.

4. If you installed Microsoft Navision Application Server as a service and you set the Startup Type option to automatic, Microsoft Navision Application Server will automatically start every time you start the computer.

5. When you set up Microsoft Navision Application Server to run as a service, you can use the Microsoft Management Console to change its properties.

6. In the right-hand pane of the Microsoft Navision Application Server Manager window, you can define or change the properties of Microsoft Navision Application Server.

7. The ADCS granule provides companies with the necessary functionality to capture accurate data for inbound, outbound and internal documents, primarily in connection with warehouse activities.

8. When you run Microsoft Navision Application Server as a service, it runs without displaying anything on-screen and requires no user interaction

9. When you run Microsoft Navision Application Server as a service, events and error messages are written to the Event Viewer.

10. The Communication System Service is responsible for handling requests directly from a plug-in that supports the protocol used by the handheld terminals.

Test Your Knowledge – Microsoft Navision ODBC Driver (Chapter 7)

True	False		
☒	❑	1.	You can use the Microsoft Navision ODBC Driver to retrieve Microsoft Navision data into an application, such as a Word processor or spreadsheet.
❑	☒	2.	Criteria is the same as a filter in Microsoft Navision while an ADD-IN is a command or function that gives a program additional capabilities.
❑	☒	3.	A DATA SOURCE contains information about where to find the data and how the driver formats the data when it is returned to an application.
☒	❑	4.	The Microsoft Navision ODBC Driver can retrieve the application data from Microsoft Navision in different languages independent of the current Microsoft Navision application language.
❑	☒	5.	You CANNOT use the Microsoft Navision ODBC to create a table with Multilanguage Caption properties.
❑	☒	6.	The acronym ODBC stands for Open Database CONNECTIVITY.
❑	☒	7.	SQL is a programming language that is specially designed for queries in databases.
☒	❑	8.	The option selected in the Identifiers field controls how table names and field names are transferred from Microsoft Navision to an external program.
❑	☒	9.	In the All Except Space Identifier option, the Profit % field would be transferred over as Profit_%.
☒	❑	10.	In the a-z,A-Z,0-9 Identifier option, the Sales (LCY) field would be transferred over as Sales_LCY_.

Test Your Knowledge – Microsoft Navision Security (Chapter 8)

True	False		
❑	☒	1.	A MEDIUM level of security requires that you can limit the user's access so that they can only access certain types of information stored in the database.
☒	❑	2.	Record level security is available using the SQL Server Option for Microsoft Navision.
☒	❑	3.	In a Active Directory environment, Microsoft Navision allows you to create users and roles from Windows accounts and modify the rights of these users and roles.
❑	☒	4.	The Microsoft Navision Security system is initiated when you create a database login for a SUPERUSER.
☒	❑	5.	At least one user must have the SUPER role for all companies.
❑	☒	6.	The ALL role must be assigned to all users, EXCEPT the Superuser.
❑	☒	7.	Security Filters are available for SQL SERVERS ONLY.
❑	☒	8.	Passwords in Microsoft Navision DO NOT have time limits.
☒	❑	9.	Microsoft Navision allows you to put time limits on user ID's.
☒	❑	10.	In Microsoft Navision, Printers and Menus can be user defined.

Test Your Skills – Create Database Logins and Assign Roles (Chapter 8)

Scenario: As the IT administrator, you need to create database logins and passwords for three new employees and assign the appropriate roles.

Step-by-Step

1. Go to TOOLS→SECURITY→DATABASE LOGINS
2. In the **User ID** field, type JR.
3. In the **Name** field, type John Roberts.
4. In the **Password** field, type Cronus.
5. Click on the line below, and then select the JR User ID again.
6. Click on the **Roles** button.
7. In the **Role ID** field, type Super.
8. Click on the line below to commit the Role to the User.
9. Escape back to the Database Logins window.
10. In an empty line, in the **User ID** field, type AH.
11. In the **Name** field, type Alice Hart.
12. In the **Password** field, type Alice.
13. Click on the line below, and then select the AH User ID again.
14. Click on the **Roles** button.
15. In the **Role ID** field, type ALL.
16. Click on the next line and in the **Roles ID** field, click on the **AssistButton** (or press F6).
17. Locate the Roles staring with P&P (for instance, P&P-JOURNAL).
18. Assign all Roles that begin with P&P to Alice by selecting the Role ID and clicking on OK.
19. You will repeat steps 16 & 18 until all Roles beginning with P&P are assigned to the AH Database Logins. (Note, there are 11)
20. On the last role entered, be sure to press Enter or click on the line below to commit the Role to the User.
21. Escape back to the Database Logins window.
22. In an empty line, in the **User ID** field, type RQ.
23. In the **Name** field, type Richard Quist.

24. In the **password** field, type Richard.

25. Click on the line below, and then select the RQ User ID again.

26. Click on the **Roles** button.

27. In the **Role ID** field, type ALL.

28. Click on the next line, and click on the **AssistButton** (or press F6).

29. Select "SUPER (DATA)"

30. Click on the **OK** button, and press Enter or click on the line below to commit the Role to the User.

31. Escape back to the Navigation Pane.

Test Your Skills – Create a new Role (Chapter 8)

Scenario: Your company hires a consultant who will be working in-house for several months doing telemarketing for your Contacts. The consultant will need access to read the Contact tables and enter (insert, modify and delete) comments on the Cards. Set up a new role that can be used to allow the consultant the appropriate access to information in your system.

Step-by-Step

1. Go to TOOLS→SECURITY→ROLES.
2. Click EDIT→INSERT NEW (or press F3).
3. In the empty **Role ID** field, type Madeleine.
4. In the **Name** field, type Madeleine.
5. Press **Enter** or click on the next line.
6. Locate the RM-CONT Role ID.
7. Click on ROLES→PERMISSIONS.
8. Select all (CTRL + A) the permissions, and then copy (CTRL + C) the permissions.
9. Escape back to the Roles window.
10. Locate the MADELEINE Role ID.
11. Click on ROLES→PERMISSIONS.
12. Paste (CTRL + V) in the permissions.
13. Locate the Object Type = Table Data, Object ID 5061, Object Name: Rlshp. Mgt. Comment Line.

14. In the **Insert Permission**, **Modify Permission**, and **Delete Permission** fields, type a Y (this will translate to a Yes once you leave the field).

15. Escape back to the Navigation Pane.

Test Your Skills – Change a Password (Chapter 8)

Scenario: You are Alice, and you want to change your password.

Step-by-Step

1. Log in to Microsoft Navision as Alice:

– User ID = AH
– Password = Cronus

2. Go to TOOLS→SECURITY→PASSWORD.

3. In the **Current Password** field, type Cronus.

4. In the **New Password** field, type AliceNew.

5. In the **Reenter New Password** field, type AliceNew.

6. Click on the **OK** button.

7. Log out of Microsoft Navision.

8. Log in to Microsoft Navision using the new password.

9. Log out of Microsoft Navision.

Test Your Skills – Set up Time Limits for a User (CH 8)

Scenario: Richard Quist is a seasonal sub-contractor. As the IT Administrator, John Roberts, you must set up a time limit for him so that he can access Microsoft Navision from November 1, 2000 to January 2, 2001. The finance department would also like his time registered and has asked that you set that up as well.

Step-by-Step

1. Log in to Microsoft Navision as John Roberts:

– User ID = JR
– Password = Super

2. Go to ADMINISTRATION→APPLICATION SETUP→USERS→USER SETUP.

3. In the **User ID** field, type RQ.

4. In the **Allow Posting From** field, type 110100.

5. In the **Allow Posting To** field, type 010201.

6. Click in the **Register Time** field.

7. Escape back to the Navigation Pane.

Test Your Knowledge – Basic Customization (Chapter 9)

1. What is the difference between the Navigation Pane and the Navigation Pane Designer?

The Navigation Pane gives you access to the business application areas in Microsoft Navision. The Navigation Pane allows for some minor end user modifications as well. The Navigation Pane Designer gives editing access to the MenuSuite. In the Navigation Pane Designer, a MenuSuite Administrator can create new menus, modify existing menus and assign users to menus.

2. Describe the ways an end user can personalize the Navigation Pane.

End Users can create Shortcut Menus, Hide/Show menus, Hide/Show Menu Groups, Hide/Show Menu Items and Move Menu buttons up and down.

3. What are the MenuSuite Object Levels?

Changes that are made to a MenuSuite object are stored as the differences between the previous MenuSuite object level and the current one. Those levels are MBS, Region, Country, Add-on, Partner, and Company. The MBS level is a generic MenuSuite object. When an object is localized, it is changed at the Country level. A Microsoft Business Solutions partner developer would make changes at the Partner level. Users normally make changes at the Company level.

4. Describe the Shortcuts Menu and how an end user might use it.

The Shortcuts Menu allows an end user to easily customize a menu that contains only the forms, reports, activities that they need to perform in Microsoft Navision. It also provides a shortcut to Web sites, files, and documents outside of Microsoft Navision. An end user could set up her Shortcut menu with only the tasks they do on a daily basis, so instead of having to click on four levels to reach the Salesperson Opportunity Report, she could click it once from the Shortcut Menu.

5. Explain why a MenuSuite Object might be exported.

When an administrator makes configuration changes to a menu suite, these changes are saved at the Company level. When this Company-level MenuSuite object is exported in text format, the text file contains information about the changes that have been made to the object since the previous level, which in this case would be the Partner level.

Test Your Skills – Copy and Modify a Menu (Chapter 9)

Scenario: As the Menu Suite Administrator, you have been asked to create a new Finance Menu without Fixed Assets.

Step-by-Step

1. Click on TOOLS→NAVIGATION PANE DESIGNER.
2. Right-click on the Financial Management menu.
3. Right-click on another menu and select Paste.
4. Scroll to the bottom of the Menu list to the new Financial Management menu. Note that there are no markings (>>) next to the name as the other menus show.
5. Right-click on the menu and select Rename.
6. Make the menu name: "Financial Management, No FA."
7. Press **Enter**.
8. Go up to the Menu Groups displayed above, and right-click on Fixed Assets.
9. Select **Delete**.
10. Press **Escape**, and click **Yes** to the message asking if you want to save the changes.
11. From the Navigation Pane window, scroll down the list and verify that the Financial Management, No FA menu does not contain Fixed Assets.

Test Your Skills – Create a Shortcut Menu (Chapter 9)

Scenario: You are in the Marketing department and want to see menu items related to Marketing, as well as the Salesperson card and the Salesperson Opportunity Report. You also access the Microsoft Web site quite often so you want a direct link there as well.

Step-by-Step

1. Click on the Sales & Marketing menu.
2. Right-click on the Marketing subfolder, and select Send to Shortcuts.
3. Click on the Sales subfolder, right-click on the Salesperson Card, and select Send to Shortcuts.
4. Click on the Reports subfolder, and select Salespeople/Teams.
5. Right-click on the Salesperson Opportunity report, and select Send to Shortcuts.

6. Click on the Shortcuts menu.

7. Right-click in the Navigation Pane area, and select Create Shortcut.

8. In the **Open** field, type "www.microsoft.com," and in the **Caption** field type "Microsoft Web Site."

9. Click on the **OK** button.

10. Click through the Shortcuts Menu to see if everything you added is there.

Test Your Knowledge – Microsoft Navision Database Administration Tools (Chapter 10)

1. Microsoft SQL Server supports four different types of backups: Database backup, Transaction log backup, Differential backup, and File and filegroup backup.

2. If the synchronization of the table relationships fails when you are restoring a database backup, the backup will be restored.

3. You must use the Microsoft Navision backup function when you want to migrate your data from Microsoft Navision to the SQL Server Option for Microsoft Navision.

4. Server Based Backups are called HotCopy and are used with the Microsoft Navision Database Server.

5. The three types of backup options available are Entire, All Companies, and Custom.

6. To safeguard against errors, you should frequently check the integrity and consistency of your databases.

7. When an index (key) is created, SQL Server automatically generates and stores statistical information about the distribution of values in the indexed column(s).

8. A recommended backup procedure includes backing up the database every week.

9. Roll forward is achieved by restoring your last database backup and applying all subsequent transaction log backups to recreate these transactions.

10. Primary and Secondary Keys are sometimes described as indexes and can be used for sorting information.

Test Your Knowledge – Optimizing Microsoft Navision (Chapter 11)

True	False		
❑	☒	1.	A database table contains a primary key and SEVERAL secondary key.
☒	❑	2.	A B+ tree, or balanced tree, is one in which the program always passes through the same number of levels to access data at the bottom of the tree, regardless of which branch of the tree the data is located on.
☒	❑	3.	Keys can be setup as key groups, which can be enabled and disabled without risk.
❑	☒	4.	SQL Server DOES NOT needs an index in order to sort by particular fields.
❑	☒	5.	In date compression, you can compress entries from closed fiscal years MORE THAN once.

Test Your Knowledge – Optimizing Microsoft Navision (Chapter 11)

1. The SQL Profiler is a graphical tool that allows system administrators to monitor events in an instance of Microsoft SQL Server. List the various reasons the SQL Profiler would be used.

- Monitor the performance of an instance of SQL Server.
- Debug Transact-SQL statements and stored procedures.
- Identify slow-executing queries.
- Test SQL statements and stored procedures in the development phase of a project by single-stepping through statements to confirm that the code works as expected.
- Troubleshoot problems in SQL Server by capturing events on a production system and replaying them on a test system. This is useful for testing or debugging purposes and allows users to continue using the production system without interference.
- Audit and review activity that occurred on an instance of SQL Server. This allows a security administrator to review any of the auditing events, including the success and failure of a login attempt and the success and failure of permissions in accessing statements and objects.

2. Long-running queries and driver-performance data are the two different ODBC drivers used to profile performance data. What is the difference between the two?

– In Long-running queries, the driver can write to a log file any query that does not get a response from the server within a specified amount of time.

In Driver-performance data, the driver can record performance statistics and either write them to a file or make them available to an application through a driver-specific data structure named SQLPERF.

3. What are the implications of the database-version principle approach in Microsoft Navision?

The most important implication is that different Microsoft Navision clients may read different versions of the same database. These versions are snapshots of the database at the point in time when the clients start to access the database. In this way, the DBMS allows for concurrency (that is, for more than one user to have access to the database at the same time) while still maintaining read consistency. If the accesses to the database involve only data retrieval and no changes, then the newest version will persist – for all clients – until a write transaction is performed.

4. What are some of the limitations or restrictions with using SQL Profiler?

Monitoring too many events adds overhead to the server and the monitoring process and can cause the trace file or trace table to grow very large, especially when the monitoring process takes place over a long period of time.

5. What is some of the information found in a trace log?

– System information, including operating system, version, number of processors, and memory.
– Reporting Services component and version information.
– Events logged the Application log.
– Exceptions generated by the report server.
– Low resource warnings logged by a report server.
– Inbound SOAP envelopes and summarized outbound SOAP envelopes.
– HTTP header, stack trace, and debug trace information.

Test Your Knowledge – Troubleshooting Microsoft Navision (Chapter 12)

1. Name the three typical server calls that generate status information.

– Server calls that modify or delete sets of records.
– Server calls that scan an index or an entire table to find some data.
– Server calls that need to lock a record or a table can be forced to wait until other transactions are committed and release the locks that they placed.

2. What is the purpose of using the Client Monitor? What information is contained in the Client Monitor window?

The Client Monitor is an important tool for troubleshooting performance and locking problems. You can also use it to identify the worst server calls and to identify index and filter problems in the SQL Server Option.

The Client Monitor window displays and formats the data that has been gathered by the Client Monitor so that it can be more easily analyzed. It carries out a kind of cross tabulation of the operations and parameters and uses one line per server call.

3. What are the most common causes of performance problems in Microsoft Navision?

– The way that keys are defined combined with the way that they are used in filters or queries when you want to read data.
– The way keys are defined with SumIndexFields combined with the way that summing FlowFields are defined.
– The number of keys that are defined with SumIndexFields when running on SQL Server.
– In SQL Server, setting the SourceTablePlacement property to the default value (Saved) will often make opening forms that display data from tables that contain many records (1,000,000 or more), for example G/L entries, very slow. To fix this problem, set the SourceTablePlacement property to First in these forms.

4. Describe deadlocks, how they occur and how they can be resolved.

Deadlocks occur when concurrent transactions try to lock the same resources but do not lock them in the same order. In other words, a deadlock occurs when two or more transactions have a conflicting locking order. This can either be solved by always using the same locking order or by using a "locking semaphore" that will prevent these transactions from running concurrently. When you want to identify potential locking problems, you run the tasks on the client that you think might cause locking problems and gather all of the relevant data in the Client Monitor and then open a special form to see if there are any potential deadlocks.

5. Bottlenecks can occur in the software or the hardware. Describe how bottlenecks can be handled in both instances to resolve performance problems.

In the Software, there are three relevant properties to adjust in Microsoft Navision:

- DBMS (Database Management System) Cache
- Commit Cache
- Object Cache

In the Hardware, to identify any hardware bottlenecks that may exist on a server, use the Windows Performance Monitor. You should check the time usage for the both the disks and the CPU.

If the disks cause the bottleneck, you can lessen the problem and improve performance by:

- Adding more RAM to decrease disk activity, such as swapping.
- Spreading the SQL Server database and log files across more disks.
- Adding one disk controller per disk.
- Installing faster disks.
- If the bottleneck is caused by the CPUs, you can improve performance by installing faster CPUs.

6. Discuss Locking in Microsoft Navision. What is the main difference between Locking in the Microsoft Navision Database Server and the SQL Server?

Performance problems that are caused by clients spending time waiting for other clients to release locks on resources that the client in question wants to place exclusive locks on is called Locking. The main difference between locking in the Microsoft Navision Database Server and the SQL Server is the way the locking Code is written for each server option.

7. What is Index Hinting and how is it used in Microsoft Navision?

Index hinting can help avoid situations where SQL Server's Query Optimizer chooses an index access method that requires many page reads and generates long-running queries with response times that vary from seconds to several minutes. Selecting an alternative index can give instant "correct" query executions with response times of milliseconds. This problem usually occurs only for particular tables and indexes that contain certain data spreads and index statistics.

In the rare situations where it is necessary, you can direct Microsoft Navision to use index hinting for such problematic queries. When you use index hinting, Microsoft Navision adds commands to the SQL queries that are sent to the server. These commands bypass the normal decision making of SQL Server's Query Optimizer and force the server to choose a particular index access method.

8. How do you maximize the performance of a query?

Define the keys in the table so that they facilitate the queries that you will have to run. These keys must then be specified correctly in the queries.

9. What is the purpose of trace flags 1204 and 3605 in the SQL Server Error Log?

The trace flags, when enabled, monitor deadlocks on SQL Server. They will generate diagnostic messages in the SQL Server Error Log.

Trace Flag 1204 returns the type of locks that are participating in the deadlock and the current command affected.

Trace Flag 3605 sends the trace output to the error log.