SIEMENS

SIMATIC IT Unilab 6.7

Installing Unilab

Installation Manual

Edition 10/2014

A5E00428222-09

Guidelines

This manual contains notes of varying importance that should be read with care; i.e.:

Important

It contains essential information regarding handling the product, the product itself or a specific part of the documentation.

Note Provides supplementary information regarding handling the product, the product itself or a specific part of the documentation.

Trademarks

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E00428222-09
10/2014

Preface

Where is this manual valid?

This manual is valid for release 6.7 of SIMATIC IT Unilab.

Basic knowledge required

This guide is intended for SIMATIC IT Unilab users who are responsible for system configuration, such as application managers and system integrators (consultants). To be able to understand the concepts and examples discussed in this guide, the reader should at least have taken the SIMATIC IT Unilab Basic Training.

Purpose

This Installation Manual explains how to install Unilab.

Important

This guide does not intend to describe the guidelines to install Terminal Server. For more information, see **Installation guidelines for Terminal Server**.

Related documentation

The following documents contain information related to the content of this Installation Manual.

- Unilab Concepts Guides (parts 1, 2 and 3)
- Prerequisites installation manual

These documents are all available online from the SIMATIC IT Unilab Documentation Library**.**

Conventions

The table below describes the specific typographic conventions that are used throughout this manual:

Symbol/Convention	Indicates...
E.g.	Where examples are given.
Text in bold	The names of menus, commands, dialog boxes and toolbar buttons and, in general, all strings (e.g. **File** menu; **Save** command).
KEY1+KEY2	Shortcut keys, which permit rapid access to commands (e.g. CTRL+C).
UPPERCASE	The names of keyboard keys (e.g. RETURN key).

Symbol/Convention	Indicates...
Italics	Noun with special importance or significance for which emphasis is needed. The names of parameters that must be replaced with a specific name or value.
>	A succession of commands in which the command preceding the symbol must be selected before the command following it.
 `Code example` 	Code example.

SIMATIC IT Documentation Library

The SIMATIC IT Unilab Documentation Library provides you with a comprehensive and user-friendly interface to access the overall product documentation where manuals and helps online can be browsed by functionality or by component.

Readme

The installation includes a readme file, which contains information on upgrade procedures and compatibility with previous releases. This file is supplied both in standard text (**Readme.wri**) and in Acrobat PDF (**Readme.pdf**) format.

This file is available in folder \ReleaseNotes of the setup DVD and is available from the SIMATIC IT Unilab Documentation Library.

SIMATIC IT Training Center

Siemens IA AS MES offers a number of training courses to familiarize you with the SIMATIC IT product suite. To successfully achieve this goal, training consists of lessons in both theory and practice.

Courses are held year-round, according to a program that is published well in advance of the first scheduled session.

The material on the basis of which our courses are conducted reflects the result of years of experience in process, LIMS, quality control and production management.

All courses are held by expert personnel that are aware of the developments and innovations in the Siemens IA AS MES product suite.

Courses are held in English at the Siemens IA AS MES Training Centers.

Upon request, training courses can also be organized on the customer's premises.

For more information on the training course calendar, please visit our technical web site (http://www.siemens.com/simatic-it/training).

SIMATIC IT Service & Support

A comprehensive Software Maintenance program is available with SIMATIC IT products. Software Maintenance includes the following services:

- **Software Update Service** (SUS): automatic distribution of upgrades and service packs
- **Technical Support Service** (TSS): support on technical problems with SIMATIC IT software (standard support and other optional services)
- **Online Support**: a technical web site, providing information such as Frequently Asked Questions and technical documentation on SIMATIC IT products

Software Update Service (SUS)

This service provides automatic shipment of new versions and service packs when released. When a new version / service pack is available for shipping, it is typically shipped within one month.

One copy of the installation DVD is shipped for each Server covered by Software Maintenance.

Hot fixes (officially tested and released) are not shipped and must be downloaded from the Technical Support Service web site.

Technical Support Service (TSS)

Siemens provides a dedicated technical support team for SIMATIC IT products.

The following options are available:

Bronze support: 9 hours/day, 5 days/week

Silver support: 24 hours/day, 5 days/week

Gold support: 24 hours/day, 7 days/week

The principal language of the SIMATIC IT hotline is English.

SIMATIC IT partners and customers covered by the Software Maintenance program are entitled to direct access to the TSS..

Access to TSS

To be able to access TSS, the customer needs to register as a user on the Technical Support web site. Connect to http://www.siemens.com/mes-simaticit/ and follow the **Technical Support Service** link.

The registration form must be completed with:

- Personal data
- The required company and plant information
- The Contract Number provided by Siemens Back Office when the contract is agreed.

Online Support

A customer who is a registered TSS user, can access the Technical Support web site (http://www.siemens.com/mes-simaticit/tss), which contains technical information such as:

- Service conditions (Phone numbers, Working hours, Reaction times,…)
- SIMATIC IT knowledge base: a technical support database that includes practical service solutions from Technical Support or the SIMATIC IT community
- SIMATIC IT software (e.g. hot fixes, software examples) and release notes that can be downloaded
- SIMATIC IT cross-industry libraries that can be downloaded (limited access to SIMATIC IT certified partners)
- SIMATIC IT product documentation that can be downloaded
- Frequently Asked Questions and useful tips.

Table of Contents

1 Introduction

Scope

This manual describes the procedure for the installation of the 2-tier (Client/Server) version of SIMATIC IT Unilab. This includes:

- The installation of a **SIMATIC IT Unilab Server**. The overview of this process is given in chapter 2.

  The server installation process is grouped in three main parts:

  - Installing the demo server
  - Installing the production database
  - Installing Unilab fundamentals for Electronic Notebooks and Formula Work Bench on an existing Interspec Oracle Server

- A production server installation is inherently a complex procedure that can best be performed by a person knowledgeable about Oracle and the OS on which the installation is to be performed. We assume that the installer is familiar with database terminology and has administration knowledge of the OS on which he/she installs the software (or a system administrator is available during the initial phase of setup). For the installation of a demo server, only a basic Oracle administration and OS knowledge is required. These two will include the creation of a new Oracle instance. The third part of the Server installation, named “Install Unilab fundamentals for ELN/FWB on an existing Interspec Oracle Server” is an extension of the production server which installs the Unilab production server using the Oracle Instance already created by the Interspec server installation.

- The installation of a **SIMATIC IT Unilab Client**. The procedure for installing a client is straightforward. This procedure is described in chapter 7.

This manual does not handle:

- The installation of **SIMATIC IT Unilab Web**.
  Some of the SIMATIC IT Unilab functionalities are also accessible from within a web browser. You will need to install the Unilab web site if you plan to install and use your own Unilab web site on your own web server. For details, please consult the SIMATIC IT Unilab – Web-based application installation manual in the Unilab Product Library.

- The configuration of optional modules, like:

  - **SIMATIC IT Unilab Archiving**:
    The SIMATIC IT Unilab server installer installs a pre-installed Archiving module. This module can be used to archive data to file only. The 'Archiving Module' application that is used for archiving uses the same java stored procedure mechanism as explained for Unilink, in the Unilink on-line help.

    For archiving data to another database, some additional steps are required. They are described in the SIMATIC IT Unilab – Archiving manual in the Unilab Product Library.

- **SIMATIC IT XML Interface**
  Please refer to the SIMATIC IT Installing Unilab Concepts and User Manual in the Unilab Product Library.
- **SIMATIC IT Uniconnect**
  Please refer to the Uniconnect online help.

- The **Upgrade** of an existing SIMATIC IT Unilab server. The upgrade procedure is given in the SIMATIC IT Unilab – Upgrading Unilab upgrade manual in the Unilab Product Library.

Installation prerequisites

- **Hardware and software prerequisites:** All prerequisites as described in the SIMATIC IT Unilab **Prerequisites** installation manual must be fulfilled before the installation of Unilab can be started.
- **License:** In order to install the SIMATIC IT Unilab 6.7 server software, you need to obtain a valid license from Siemens IA AS MES. Attempting to install with an invalid license will fail.

Installation media

In the server installation procedure, the string "**<<INSTALL BASE>>**" should be replaced with the appropriate directory containing the image of the distribution media (typically, your CD-ROM device), e.g.:

- e:\ (or other CD-ROM drive-letter) for a Windows installation of Unilab
- /cdrom/ (or other cdrom mount point) for a UNIX server installation of Unilab

For other parameters of the installation procedure, the same notation '**<<parameter name>>**' will be used. They should all be replaced with their appropriate value.

Please also read the file "**<<INSTALL BASE>>**\README.TXT" for troubleshooting and last-minute information on this installation procedure.

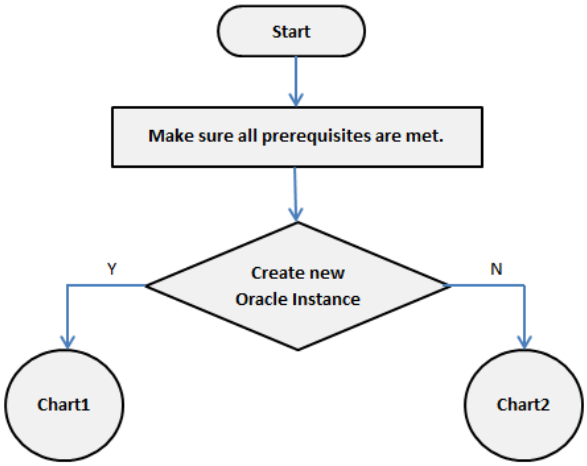
Important

This version of the Unilab server installer is only supported on Oracle versions as mentioned in the Unilab 6.7 prerequisites.

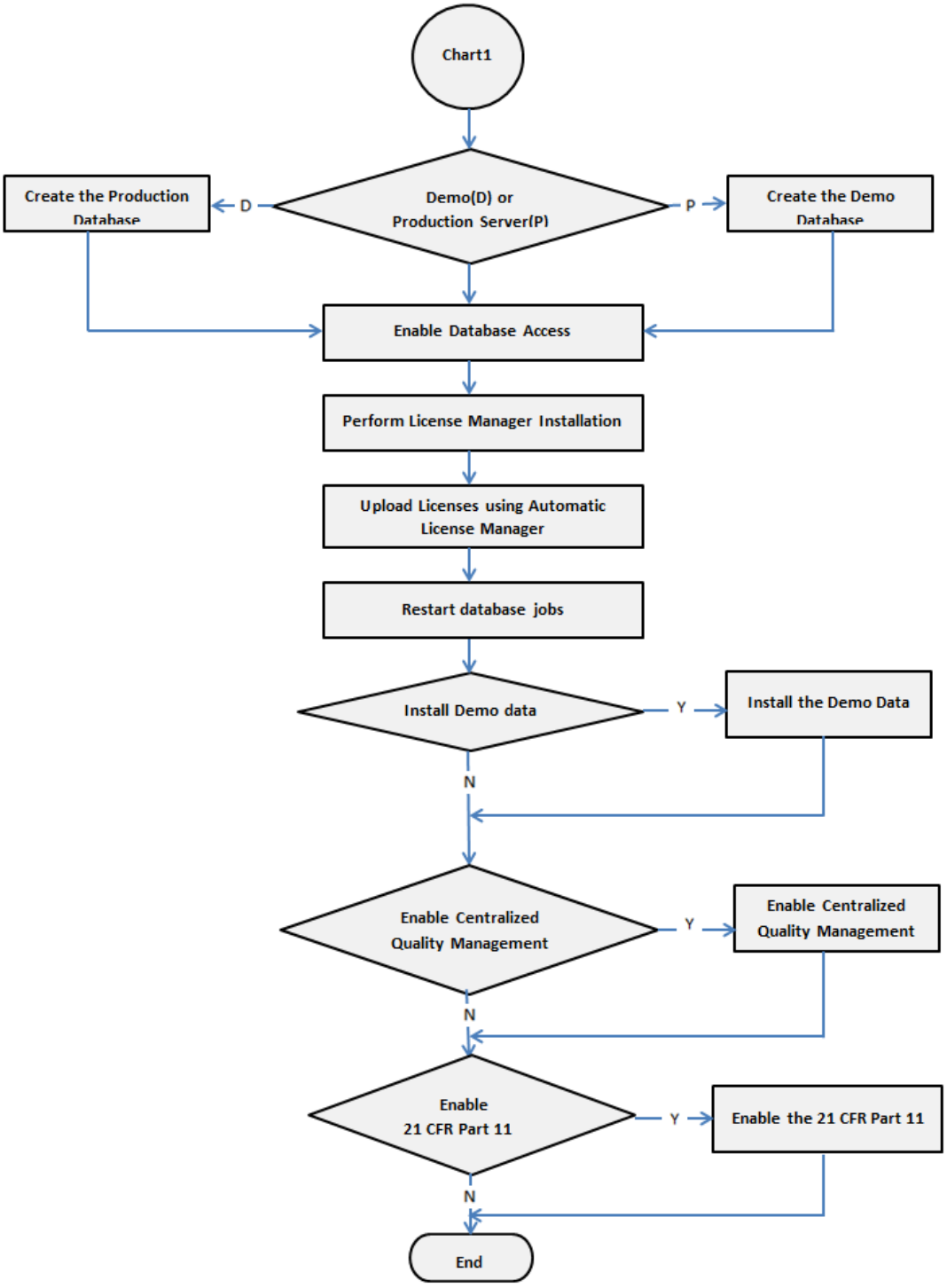
2 Overview Server Installation

2.1 Server Installation: flowchart

Below is a flowchart of the general installation procedure:



Server Installation: flowchart



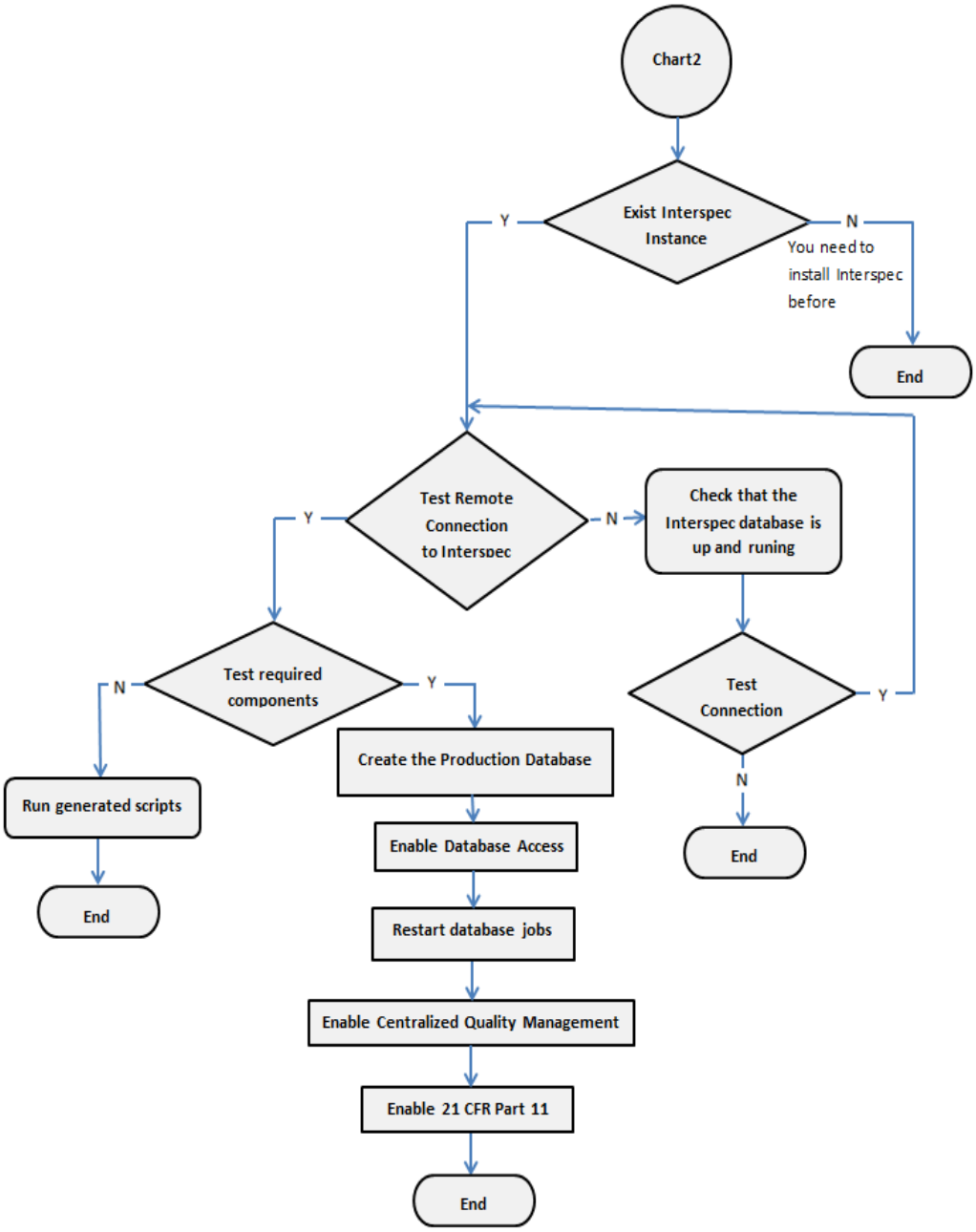


Chart2
Exist Interspec Instance
Y
N
You need to install Interspec before
End
Test Remote Connection to Interspec
Y
N
Check that the Interspec database is up and runing
Test required components
N
Y
Run generated scripts
End
Create the Production Database
Enable Database Access
Restart database jobs
Enable Centralized Quality Management
Enable 21 CFR Part 11
End
Test Connection
Y
N
End

2.2 Server installation

Below is an overview of the general installation procedure:

Step	Action	Refer to
1	Make sure all prerequisites are met. Refer to *SIMATIC IT Unilab - Prerequisites* in the Product Library.	---
2	If you want to install : • Creating a new Oracle instance – Chart 1 – Production Server, create the production database – Demo Server, create the demo database • Using an existing Oracle instance created by Interspec Oracle srever – Chart 2 – Install Unilab fundamentals for ELN/FWB	Chart 1 Chart 2

2.3 Chart 1 – Production or Demo Database Server Installation

Below is an overview of the general installation procedure:

Step	Action	Refer to
1	If you want to install a: • Production Server, create the production database. • Demo Server, create the demo database.	Chapter 3 Chapter 4
2	Enable the database access.	Chapter 6
3	Install the client application.	Chapter 7
4	Upload licenses.	Chapter 8
5	Restart database jobs.	Chapter 15.3
6	*Optional:* Install demo data.	Chapter 9
7	*Optional:* Enable Centralized Quality Management.	Chapter 10
8	*Optional:* Enable 21 CFR Part 11	Chapter 11

2.4 Chart 2 – Install Unilab Fundamentals for ELN/FWB on an Existing Interspec Oracle Server

Below is an overview of the general installation procedure:

Step	Action	Refer to
1	If you want to install Unilab fundamentals for ELN/FWB on an existing Interspec Oracle Server:	Chapter

	• Install Interspec database. • Test remote connection to Interspec database • Test required components	5.1 - 5.2
2	Create Unilab production database – only fundamentals for ELN/FWB	Chapter 5.4 – 5.10
3	Enable the database access.	Chapter 6
4	Restart database jobs.	Chapter 15.3
5	Enable Centralized Quality Management.	Chapter 10
6	Enable 21 CFR Part 11	Chapter 11

2.5 The SIMATIC IT Installer

2.5.1 SIMATIC IT Installer Application

The installer is a Windows application that will help you to setup the correct database structures for your database server. It can be used to create a demo database on the same computer but also to install a remote Windows or Unix computer.

It is possible to install the database server by means of this installer application as well as manually. To be able to do it manually, the installer application is adapted so that it will generate the required installation scripts. Those generated scripts have to be transferred to the database server and locally (or remotely) executed. In this manner, only a short connection time with the database server is required to transfer the scripts; their execution will take place on the server itself.

Starting up multiple instances of the installer application is not supported.

2.5.2 Running the Installer

You can find the installer on the product CD, in the folder **<<CD>>\Server Setup**.

You can run the installer (**Installer.exe**) from the CD or you can copy the complete setup directory to another location. Make sure all the files in the folder are readable. The installer does not write any data to the folder where the installer is located (so it can be run from the CD).

2.5.3 Help on Installer

See next chapters, and chapter Server Installation: Troubleshooting in this manual.

2.5.4 Troubleshooting and support

By default, the installer shows only important messages to the user. A lot of extra information is available and is logged in a log file. Sometimes the problem is so serious that even the log file is not created. In this case, the user will get a number of message boxes about the problem. These will typically give information about the creation of the log file.

Once the system starts installing, it keeps track of its progress in a log file per step. The log file name and location can be found in the **install.cfg,** so refer to this file for more info.

All commands that are executed have their own log during the execution. All this logging is also captured in a global log file. When a command terminates successfully, its individual log file is deleted, but the global log file retains its info. The **log file** and the **global log file** contain all you need to understand what happened during the execution of a script.

Finally, when something goes wrong, there will also be some files ending in *.dmp (**coredump.dmp** and **corestat.dmp**) that contain the current run-state of the installer. These files can give more information about what was going on inside the installer during a problem.

In case of problems with the installer, it will be helpful for support to include the following files with your problem report:

- Log file
- Global log file
- The complete **Installer** directory in the temporary directory; including the dump files as mentioned above.

3 Creation of a Production Database

3.1 Introduction

Setting up a production server requires careful planning with regard to hardware and software installation. For maximum performance and data safety, you should adhere to a number of rules, such as mirrored log files and control files, backup… for each production database installation.

Unlike the installation of a demo system (see further), for the installation of a real production server, a number of Oracle parameters must be specified. Most parameters can be changed after installation, and will therefore be set to a suitable default value. The **Prepare production database** is a quick and easy way to lead you through this process, but does not allow you to specify all possible Oracle parameters.

The **Prepare** is a two-stage set-up: first, this installer will generate a number of files in a specified directory (most notably, the **unisetup** script). Depending on the target OS, the procedure will be a little bit different. The generated scripts must then be used to create the actual database.

This two-stage process is also valid if you want to install the database on this PC.

Important

The standard installation procedure cannot be used as such for atypical setups (e.g. very large database, or a database where tablespaces must be split across multiple files). For more information on such cases, see Special Installation Issues.

3.2 How to Create a Production Database

When you use the installer to create the database, the installer does not perform actual creation. Instead, it generates a number of scripts that you need to run separately on the server machine. This procedure is more or less independent from the operating system of your server.

It is best to copy and keep these generated scripts on your hard disk, even after the installation, so you have an easy reference on how your database was created. The entire set of Unilab installation files generated by the installer requires approximately 140 MB diskspace.

Note

If something goes wrong during the installation of the production database, please delete the Oracle data files and the instance from the server.

3.3 Procedure

3.3.1 General Procedure

Proceed as follows:

Step	Action
0	Launch the **installer.exe** application in the **Server setup** folder of the installation CD.
1	On the window entitled **Step 1: Main Installation Options**, select **Prepare production database** and click **Next**.
2	On the window entitled **Step 2: Define Server Database Options**, complete all necessary options and click **Next**. Refer to subsection Define Server Database Options for more information.
3	On the window entitled **Step 3: Define dump directories and Oracle tablespaces**, define directories, file names and sizes, and click **Next**. Refer to subsection Dump Directories and Oracle Tablespaces for more information.
4	On the window entitled **Step 4: Define log and control files**, define directories and sizes, and click **Next**. Refer to subsection Log and Control Files for more information.
5	On the window entitled **Step 5: Define extra directories**, define directories and click **Next**. Refer to subsection Extra Directories for more information.
6	On the window entitled **Step 6: Define operational and configuration sizes**, complete all necessary parameters and click **Next**. Refer to subsection Define Sizes for more information.
7	On the window entitled **Step 7: Define product tablespaces**, define the directories, file names and sizes, and click **Next**. Refer to subsection Define Product Tablespaces for more information.
8	On the window entitled **Step 8: Generate scripts to create server database**, select the target directory where the scripts are to be stored and click **Generate Files**. Click **Finish** to end the wizard. Refer to subsection Generate Scripts for more information.
9	Prepare the environment for installation. Refer to subsection Prepare the Environment for a Server Installation for more information
10	Copy the generated scripts to the production server. Refer to subsection Copy Scripts for more information.
11	Run the scripts.

	Refer to subsection Run Script for more information.

Remarks on defining directories

- The directories will be created on the server. It is good practice to create directories that do not already exist. Make sure to verify all paths and drive letters.
- Make sure that the server has sufficient disk space for the given sizes.
- Do not put files on remote file systems (NFS, SMB) or network drives, as a network failure will have the same effect as a disk crash. Besides, an enormous amount of network traffic will result from such a setup.

What parameters affect performance most?

Where the redo-log files of your database are located on disk has a great impact on updates and inserts in the database, as changes can only be committed after being written in the redo-log files. Given that redo-log files are only used for database changes, they do not affect query performance.

The major performance parameter for queries is the size of your memory-caches. An in-memory operation is always much faster than any disk-operation, even if performed on the fastest disks.

Never put database files on a disk that is also heavily used by the operating system (for swapping or paging) or other applications.

It is best to spread your tablespaces over as many physical disks as possible. Put indices and tables on different disks, try to put the SYSTEM tablespace on a less frequently-used disk (almost any database operation requires operations on the SYSTEM tablespace).

RAID-5 systems can have a negative impact on performance since all disks are used for all database operations, unless database files are put in separate Raids (very expensive solution).

If disk-mirroring is used, the performance of your disks is that of the slowest disk in the mirror. It is best to mirror between equal disks, e.g. it is bad for performance to mirror two 2GB disks on one 4GB disk since performance will degrade to the level of the 4GB disk.

3.3.2 Define Server Database Options

In this window, you must enter a number of database options. Fill in the settings of the (server) PC where you intend to create the database. It is possible to create scripts on one PC to be executed on another server.

Do not confuse these settings with the settings of the PC you are working on! Even if you run the installer on the PC where you want to create the database, it is possible to have different settings for client and server.

If you generated the installation script before, a setting file called **<SID>**.set is generated in the target directory of the previous installation. In this case, you can use the button **Load Settings from File** to load the settings. This button is typically used when you have to fix a problem in an ongoing installation.

Setting	Description
Operating system	Select the target operating system. This selection will fill in a number of defaults in the following steps that are most suitable for your operating system.
Oracle version	Select the version of Oracle as installed on the server. Only versions that are supported will be listed. If your server version is not listed, please contact your support representative. Installing on a platform that is not supported can result in data loss or problems that are difficult to pinpoint.
Oracle home	Enter the Oracle home path of your server. The Oracle home is typically the location where most Oracle folders are found. (Typical examples are: C:\Oracle, C:\Oracle\Ora11).
Database name (SID)	This is the name of your database, also known as SID (System Identifier). It cannot be changed after database creation. It is recommended that SID length not exceed 8 characters. Use only alphanumeric characters and the first character must be a letter. Do not use any special characters in the SID (no characters other than the US7ASCII characters and only alphanumeric characters).
Oracle user name	This is the name of the Oracle DBA user.
Database character set	This is the native character set of your database, a very important parameter. It is recommended to select the character set you will use on your clients. In Western Europe and America, the default WE8ISO8859P1 is the recommended setting. If you want a character set other than those displayed, another character set can be entered. This character set must be known to Oracle. Specifying an invalid character set will cause problems during the installation. Verify the Oracle installation manuals to obtain a list of all supported character sets. Make sure to select a character set that supports ALL characters used by your various clients connecting on your database. Unilab supports single-byte character sets and the Unicode character set AL32UTF8. As of Unilab 6.1, Unicode support has been introduced: the only supported and tested character set for Unicode support is AL32UTF8. Your database will be able to correctly store the characters from clients that work with different character sets (as well as from our Unicode Unilab client). Oracle provides other Unicode-compliant character sets AL24UTFFSS, UTFE and UTF8), but please do not use these character sets to create a Unicode-compatible Unilab database. These character sets have been provided only for backward compatibility and will become obsolete in the future. Oracle also currently

	supports multi-byte character sets (e.g: JA16EUC): we do not recommend these character sets, since they have not been tested explicitly (we expect that they will be replaced by Unicode in the long term).
Time Zone	We strongly advise selecting a time zone offset that is the same as your os server time zone offset during the winter time period. E.g: On a server running in Rome, the offset in the winter time period is +01:00, the offset in summer time period is +02:00, you will specify +01:00 as your database time zone.
Concurrent sessions	This is the number of sessions that will connect to the database simultaneously. Note that a user can connect with multiple applications on one client-PC, so this is generally higher than the number of users. It also includes users running reports or other applications versus this database. This parameter is used to calculate the **max_processes** parameter in the initialization file. It is safe to set it higher than necessary (this is not directly related to your license).
Memory to allocate	The memory that can be allocated to the Oracle instance on that server. When possible, set it to 90%-95% of available physical memory on the server before installing the Unilab database instance. The init<SID>.ora settings will be set to values derived from that value. The number of concurrent connections (**Files** tab) is also important for a correct estimation. When possible, also take into account that other applications can be installed on that server and will also require a part of the memory. It is a good practice to tune memory parameters once the system is in production. Please note that, starting from Oracle 11, we strongly advise using the “Automatic Memory Management” feature introduced by Oracle (only specify MEMORY_TARGET as memory setting in the database parameters (init<SID>.ora or spfile)).
Version control and 21CFR11	Check this flag if the license contains the 21CFR11 module. **Note** If this installation option is NOT set and you decide to install an Advanced Test Server license (or any other license that supports 21CFR11), remember that the application will NOT behave as expected: therefore, remember to always select this installation option if you foresee using the 21CFR11 module. If you install a license that supports 21CFR11 without this installation option selected, you must necessarily modify the configuration elements (see 10.6 Modify Configuration Elements in this manual).
Lock DBA	Check this flag if the DBA must be locked after installation.
DBA name	This name indicates the owner of the Unilab database. A typical name is 'UNILAB'. Special characters other than alphanumeric characters in US7ASCII are supported, but

	not recommended. **Important remark:** For installation of **EasyQC** or **EasySL**, the default DBA name MUST be **UNILAB** (password may be anything). Choosing another DBA name will result in problems with the standard EASY installation.
DBA password	This is the password of the DBA user. Oracle supports single-byte characters for its passwords. [1]
Administrator name	The DBA user defined above is reserved for maintenance operations on the database and is not necessary for normal Unilab operations. It can be locked after database creation (checkbox). The administrator user is created for daily maintenance operations (starting jobs, creating users, etc.). Special characters other than alphanumeric characters in US7ASCII are supported, but not recommended.
Administrator password	This is the password of the administrator user. Oracle supports single-byte characters for its passwords. [1]

[1] As of Oracle11, Oracle has introduced case sensitivity for passwords: we suspect that Unicode passwords will also be supported. However, as we did not have sufficient time to perform exhaustive testing for this feature, we still recommend using single-byte characters for passwords. If you plan to not follow this rule, be sure to test it yourself.

3.3.3 Define Dump Directories and Oracle Tablespaces

Enter the directories for the dump files.

For these files, a RAID-1 or 5 system is recommended.

Dump file	Description
BDUMP	The background dump location. This directory is used by Oracle to write the database **alert** file.
UDUMP	The user dump location. This directory is used by Oracle to write database trace files.
CDUMP	The core dump location. This directory is used by Oracle to write database core dump file. Required for Unix only.

Enter the directories, file names and sizes for the main Oracle tablespace files.

Ideally, these files are put on a RAID-5 system for best performance and data safety.

Tablespace	Description
SYSTEM	Holds the Oracle System catalog.
TEMP	Can be used only for objects created by Oracle.
UNDO	Used by Oracle to keep track of transactions.
SYSAUX	Oracle internal tablespace.

3.3.4 Define Log and Control Files

Enter the directories for the Oracle redo-log and control files. For the log files, also the file sizes can be entered.

For the log files, a RAID-0 system is recommended. For the control files, a RAID-1 or 5 system.

This window allows you to define the 2 (optionally, 3) mirrors of log files and control files. It is recommended that these files be set on different physical disks, as losing your control or log files can result in losing your database! Note that, even when your disks are already mirrored, it is still good practice to keep a software-mirror of these files. This is because the hardware mirror will not protect you against accidental deletion of these files.

Control files contain structural information about the Oracle database including static data, such as file names, and more dynamic information, such as the current redo log sequence number. Each Oracle database should have at least two control files, each stored on a different disk. If a control file is damaged due to a disk failure, the damaged control file can be restored using an intact copy of the control file.

Log files contain a record of all committed changes. They are used for recovery after an instance failure. The log files are the speed-determining factor for write-operations to the database: they are written in sequential order and the redo-log-files are used in a cyclic order. Use more log groups if you have more simultaneous users. For greater protection against data loss, we recommend archiving the log files. This is done by using archive log mode for the database. A bad location for redo-log files can adversely affect system performance. Log files must always be mirrored. If all log files are lost, you might also lose data.

3.3.5 Define Extra Directories

This window allows you to define some extra directories.

For the Unilab archive directory, a RAID-1 system is recommended, while for the Unilink directories, a RAID-1 or 5 system.

Directory	Description
Archive log	The Oracle archived redo-log directory.
Unilab archive	The Unilab archive directory. Used by the Archiving Module.

Directory	Description
	This should not be confused with the Oracle entry above.
Unilink input	The input directory for Unilink.
Unilink output	The output directory for Unilink.
Unilink error	The error directory for Unilink.
Unilink log	The logging directory for Unilink.

Note

Archive redo-log should not be activated during initial database creation. It is far more efficient to take an offline backup immediately after installing the server and activating archive-log at that stage. The installation requires more redo-log information than do normal database activities.

A standard Unilab database creation does not activate archive redo-log mode.

Implications in using archive redo-log

If a database crashes, the redo-log files are used to make your database consistent. If a data file must be replaced from a backup, you will need another file containing the changes made since the last backup. Oracle has an option, archive log mode, which enables keeping redo-log files for an extended time. Without archive log mode, you lose all changes since your last backup in case of a disk-crash. If such a loss is not tolerable, you need to activate archive log mode. Consult your Oracle manuals for information on how to enable log-file archiving. You will need some extra disk space, depending on the number of changes performed on your database. You must have sufficient space for a worst-case scenario: typically, this will be in the range 500MB - 1GB. You can use tapes to backup your archived redo-log files. Never put your archived redo-log files on the same disk as other database files, since a disk crash could destroy both and you might still lose data.

3.3.6 Define Operational and Configuration Sizes

This window allows you to define the size of an operational sample, and the size of the configuration.

Not all Unilab databases are used in the same way: therefore, tables and data files will have different sizes in the different implementations of Unilab. In this sheet, you should fill in the number of each object type that should be kept on line (i.e. not archived): for example, 4 years on line, with 10000 samples per year. These settings should be estimated carefully with your customer in order to avoid maintenance problems.

Then click **Estimate Size** to automatically calculate the appropriate sizes for Unilab tables and data files (storage clauses).

3.3.7 Define Product Tablespaces

Enter the directories, file names and sizes for the Unilab specific tablespace files.

Ideally, these files are put on a RAID-5 system for best performance and data safety.

3.3.8 Generate Scripts to Create Server Database

This window allows you to select the directory where the files for server installation will be created.

The Generator creates:

- All necessary scripts and will also create
- A settings file called <*SID*>*.set*, where <*SID*> is the name of the database. You can use this file to rerun the installer later.

3.3.9 Prepare the Environment for a Server Installation

How to prepare the environment depends on the Operating System of your server. There are two possibilities:

- Windows server
- UNIX Server

Windows server

Proceed as follows:

Step	Action
1	Check if a user **<<Administrator>>** with administrator privileges exists (usually 'Administrator', but this depends on the Windows language).
2	Log on as **<<Administrator>>**.
3	Check the Oracle home directory in the registry (or use Oracle Home Selector to set ORACLE_HOME to the correct Oracle version). **HK_LOCAL_MACHINE > SOFTWARE > ORACLE > <<HOMEX>> > ORACLE_HOME** E.g. C:\ORANT
4	Create a directory to which the required installation files will be transferred. If you ran the installer on the server, you can skip this step, since all required files can be found in the given destination directory. This directory is **<<Unisetup>>**. Using a DOS-prompt:

Step	Action
	`$ mkdir` **<<Unisetup>>** or create the directory **<<Unisetup>>**, using Explorer. E.g. C:\Unisetup

Settings used during installation
<<Administrator>> := password :=
<<ORACLE_HOME>> :=
<<Unisetup>> :=

Unix server

Proceed as follows:

Step	Action
1	Check if an Oracle DBA user **<<oracle>>** exists (usually 'oracle' – this user must be a member of the 'dba' group).
2	Check if a root user **<<root>>** with password exists
3	Create a user **<<Unilab>>** with the home directory r/w for this user.

4	In the home directory of Unilab, create a subdirectory to which the required installation files will be transferred. The user 'oracle' must have r/w access to this directory.
	`$ mkdir Unisetup`
	`$ chmod a+rwx Unisetup`

Settings used during installation
<<oracle>> := password :=
<<root>> := password :=
<<unilab>> := password :=

3.3.10 Copy the Generated Scripts to the Production Server

Where to copy the files depends on the Operating System of your server. There are two possibilities:

- Windows server
- UNIX Server

Windows server

Note

If you ran the installer on the server, you can skip this step, since all required files can be found in the given destination directory. This directory is then the same as **<<Unisetup>>**.

Proceed as follows:

Step	Action
1	Log on as **<<Administrator>>**.
2	Copy all generated files from **<<Unisetup>>**.
3	Verify that all generated files are present on the production server, and that they are writable.

UNIX server

You can transfer files from a PC to a Unix server with any 'ftp' client.

Example: In the following example, we use the standard Windows ftp-client (this is a very user-unfriendly tool, so if you have a graphical-oriented ftp-client, things will be easier than in this example). In this example, we assume that the name of the Unix system is 'unixhost'. The files you want to transfer are located at D:\Unisetup and must be copied to /unilab/Unisetup. We assume that the destination directory /unilab/Unisetup exists and is writeable by all (see previous paragraph).

Proceed as follows:

Step	Action
1	Perform the commands as illustrated in the example below to transfer the files. The commands that must be entered are shown in **bold**. The rest is command feedback:

```
D:\Unisetup> ftp unixhost
Connected to unixhost.mydomain.be.
220 unixhost FTP server (Version 1.7.212.1 Sat Feb
1 01:30:15 GMT 1997) ready.
331 Password required for Unilab.
Password:
230 User Unilab logged in.
ftp> pwd
257 "/home/myuser" is current directory.
ftp> cd Unisetup
250 CWD command successful.
ftp> prompt
Interactive mode Off.
ftp> ascii
200 Type set to A.
ftp> mput *.cmd *.txt *.h *.sql *.run *.plb *.par
*.pack *.bat *.ksh *. *.java
200 PORT command successful.
150 Opening ASCII mode data connection for …
…
… (all specified files are being copied)
…
226 Transfer complete.
ftp> quit
```

2	Verify that all generated files are present on the production server.

FTP Problems

The above example worked when using the ftp software provided on Windows 2000. Some of our customers experienced problems while transferring the **.plb** files with commercial or shareware ftp software. Be careful with the configuration of such tools: it could be that no error is issued during the transfer of a plb file but that the package doesn't get compiled. For the details concerning this problem, see the problem description provided below:

Description:
UNAPIRA5.PLB package doesn't get compiled when file transferred through ftp on Unix server. Error message in sys.dba_errors: PLS-00801: internal error [56104]
Bug rejected:
This is not a bug in fact. This is an error in the configuration of ftp software used to transfer the file from the CD to the Unix server. I experienced the same problem when working with Reflection FTP.PLB files are transferred in ASCII mode. In ASCII mode, Reflection FTP has some default translating options (convert tabs into spaces - a checkbox in the translation tab of the site properties). That translation is corrupting the encrypted file (.plb file)
UNAPIRA5 was the only package that is containing some tabs. This explains why only that package is concerned by the problem.

3.3.11 Run the Creation Script

How you run your scripts depends on the Operating System of your server. There are two possibilities:

- Windows server
- UNIX Server

Important

In case of problems, see Server Installation: Troubleshooting . Each step and possible errors are explained therein.

Windows server

Proceed as follows:

Step	Action
1	Log on as **<<Administrator>>**.
2	Make sure all files are in the same directory as the Unisetup.cmd script: then switch to that directory and start Unisetup.cmd. It is best to start the Unisetup.cmd script from within a Windows command prompt (cmd.exe). You can also start the script by clicking its shortcut, but this is **not** recommended. `C:\<<Unisetup>>> Unisetup.cmd`

Overview of the steps: the times specified are merely estimates (PIII-533MHz):

Step	Action	Duration
1	Check prerequisites	10 sec
2	Windows registry changes	2 sec
3	Directory creation	1 sec
4	Database parameter file	1 sec
5	Create Oracle instance and start Windows services	1 min
6	Start instance and create SYSTEM tablespace	2 min
7	Create Oracle catalogs	13 min
8	Create other tablespaces	8 min
8a	Create JVM server	6-15 min
9	Create DBA user	1 min
10	Create database objects	4 min – 32min
11	COMPILE ALL PACKAGES	5 min
12	INSTALL UNILINK PACKAGES	1 min
13	Create initial data	1 min
14	Start event manager and insert data	2 min
15	Initialize custom functions	1 min
16	OPTIONAL: lock the Unilab dba user	<1 min
17	OPTIONAL: start the jobs as the LimsAdministrator user	<1 min

UNIX server

Proceed as follows:

Step	Action

1	Log on as user **<<root>>**
	`# cd ~<<Unilab>>/Unisetup`
	`# chmod 777 cr_area`
	`# chmod 777 *`
	`# chmod 777 .`
	`# . /cr_area`
	`# exit`
	Make sure the file ‘cr_area’ is in lowercase. Rename the file if it is created in uppercase (e.g. mv CR_AREA cr_area). All other files will be changed to lowercase by the installation program.

2	Log on as **<<oracle>>**
	`$ cd ~<<Unilab>>/Unisetup`
	`$ ./unisetup`
	`$ exit`

The table below gives an overview of the steps (the times specified are merely estimates - (PIII-533MHz).

Script	Step	Action	Duration
cr_area	1	Change /etc/oratab	10 sec
	2	Create directories	2 sec
	3	Change scripts to lowercase and alter permissions	45 sec

Script	Step	Action	Duration
Unisetup	4	Database parameter file	1 sec
	5	Create Oracle password file	5 sec
	6	Start instance and create SYSTEM tablespace	2 min(*)
	7	Create Oracle catalogs	15 min
	8	Create other tablespaces	8 min
	8a	Create JVM on server	6–30min
	9	Create DBA user	2 min
	10	Create database objects	15 min
	11	COMPILE ALL PACKAGES	15 min
	12	INSTALL UNILINK PACKAGES	2 min
	13	Create initial data	2 min
	14	Start event manager and insert data	5 min
	15	Initialize custom functions	2 min
	16	OPTIONAL: lock the Unilab dba user	< 1 min
	17	OPTIONAL: start the jobs as the LimsAdministrator user	< 1 min

3.4 Post-Installation

3.4.1 Enable e-mail Account for UNILAB DBA User

It is strongly advised to fill out the e-mail address for the UNILAB DBA user with a valid e-mail address in your company. This address must be filled out in:

- The SYSTEM SETTINGS table
- The properties of the UNILAB application user

Also verify all SMTP settings in the system settings: SMTP_SERVER must be set to a valid SMTP server on your network. These settings will be used i.e. to send an e-mail containing the new created user. This e-mail contains the temporary password (account is automatically expired) for the new users, and is send to both the newly-created user and UNILAB DBA user. Note that this can be customized in the UNUSER package.

Important As of Oracle 11, system setting SMTP_SERVER is no longer sufficient to set up the SMTP_SERVER to be used. Additional security configurations (ACL: Access control Lists) are necessary to enable network communication between the database server and the SMTP server.

Script **sysgrant4sendmail.sql** must be adapted and executed by the SYS user to enable the communication (the script is located under folder **Server Setup\db** on the Simatic IT Unilab installation CD.

3.4.2 Locked DBA

The UNILAB DBA user may be locked for normal operations (default installation). If you do not want developers to work as the UNILAB DBA user (keep in mind that only the UNILAB DBA user may perform operations such as compiling packages and creating tables), 2 scripts have been provided to create special users on your database: **crdevuser.txt** and **crsupuser.txt**, which permit the creation of development users and support users, respectively. These scripts are delivered as templates and may be adapted to meet the specific requirements of any project. Read the comments in the 2 scripts for more details.

Please refer to the SIMATIC IT Unilab – Security – Database and Logon document for more details regarding these roles.

How to unlock the DBA user is described in section 15.2 How to Unlock the DBA User.

4 Creation of a Demo Database

4.1 Introduction

This is a simplified installation, only suitable for test and demo purposes. The database is a complete SIMATIC IT Unilab database, but resides on a single hard disk: therefore, it is not a good setup for a real production server.

The Installer will create a database with minimal options: thus only minimal Oracle knowledge is required.

After installing the Demo database, the user still needs to set up an Oracle Sql*Net configuration.

A demo database is always created with as character set **AL32UTF8** (Unicode supported) and the name of the DBA user is always UNILAB and cannot be changed.

Note

Do not use a demo database installation for a production server.

4.2 Prerequisites

To install a demo server, you need the following:

- A Windows PC with a supported Windows operating system
- Sufficient disk space on one logical drive to create the database files (+/- 2 GB)

  Note

  The demo installer will create all the database files on the same logical drive. The location can be selected only for the installation directory, not for individual files.

- A supported version of Oracle server software.
- The installer program and its subfolders

The supported versions are described in the SIMATIC IT Unilab Prerequisites installation manual.

4.3 Procedure

4.3.1 General Procedure

The demo installation is performed using the SIMATIC IT Unilab Installer application. Proceed as follows to create a demo database:

Step	Action
1	On the window entitled **Step 1: Main Installation Options** select **Install demo server** and click **Next**.
2	The installer now checks if the local PC is suitable for a demo installation. It checks if you have the Oracle server software installed and if you have the appropriate operating system and user rights.
3	On the window entitled **Demo database options**, select the options you require and click **Next**. Refer to Subsection Demo Database Options Window for more information.
4	On the window entitled **Select Installation Steps**, click **Start Installation**. The installer now creates the demo database and indicates the progress of this operation.
5	If you want to use your database with the client software, or install additional software or data, you must configure your database for network access. For more information, refer to Oracle Sql*Net configuration. This is also required if you intend to use only your PC only as stand-alone, without a physical network connection.

Note

After having created the demo database, the demo data must be installed. This can be done using the Unilab Client Application (specifically, the Analyzer application). Therefore, the Client Application must be installed first.

4.3.2 Demo Database Options Window

This is the only window where the user can adapt settings for the database that needs to be created.

By default, the installer automatically selects the drive with the most disk space available. You can adapt the drive, but it will only let you create a database if there is sufficient disk space available. It is not possible to create part of the database on one drive and another part on another drive. If this is required, you must follow the procedure for installing a production server, which offers more installation options.

5 Install Unilab Fundamentals for ELN/FWB on an Existing Interspec Oracle Server

5.1 Introduction

Installing the SIMATIC IT Unilab 6.7 objects in an existing SIMATIC IT Interspec 6.5.2 database server requires careful planning with regard to hardware and software installation. To ensure maximum performance and data safety, you must follow a number of rules for every production database installation (for example, mirrored log files and control files, backups, and so on).

The database will have three schemas: Interspc, Unilab and Rndlicense.

The **Prepare** is a three-stage set-up: first, the installer tests the remote connection to the Interspec Oracle Server and some Oracle components required by Unilab. When the Interspec database server is up and running and also all required Oracle components are installed, the installer generates a number of files in a specified directory (most notably, the **unisetup** script). The generated scripts must then be used to create the actual Unilab database.

The installation of a production server is a multi-step process:

Step	Action
1	Creation of the database using the Interspec 6.5.2 installer Follow the Interspec Server installation documentation, available in the **Product Library** folder of the Interspec DVD.
2	Configure **INSTALL.CFG** – set default parameters, password, folders
3	**Prerequisite check** for the PC used to generate the scripts which will be executed on the server
4	**Start Unilab Installer - Test (**Remote Server**) Connection** • Database connection: the Sql*Net connection string (use the database name created with the Unilab installer) • SYS Password: the Oracle password for user SYS (use the same password that you already define during the Interspec installation) • Interspec DBA: the username for the Interspec DBA • Interspec DBA Password: the password for the above user The application will verify all the necessary components: • Oracle version • if Oracle JServer/XSK is installed • if XDB / XDK is installed If some of these components are missing then click **Next,** generate and save the necessary files in a specified folder. Copy the folder to your server and execute them.

5	On the window entitled **Step 2: Define Server Database Options**, complete all necessary options and click **Next**. Refer to subsection Define Server Database Options for more information.
6	On the window entitled **Step 6: Define operational and configuration sizes**, complete all necessary parameters and click **Next**. Refer to subsection Define Sizes for more information.
7	On the window entitled **Step 7: Define product tablespaces**, define the directories, file names and sizes, and click **Next**. Refer to subsection Define Product Tablespaces for more information.
8	On the window entitled **Step 8: Generate scripts to create server database**, select the target directory where the scripts are to be stored and click **Generate Files**. Click **Finish** to end the wizard. Refer to subsection Generate Scripts for more information.
9	Prepare the environment for installation. Refer to subsection Prepare for more information
10	Copy the generated scripts to the production server. Refer to subsection Copy Scripts for more information.
11	Run the scripts. Refer to subsection Run Script for more information.

Important

The standard installation procedure cannot be used as such for atypical setups (e.g. very large database, or a database where tablespaces must be split across multiple files). For more information on such cases, see Special Installation Issues.

Note

Do not use Unilab fundamentals for ELN/FWB installation until the Interspec server is not up and running.

5.2 Prerequisite Check

The installer is a Windows application that helps you to set up the correct database structures for your Unilab fundamentals for ELN/FWB.

In this step, the installer checks the minimal prerequisites of the Installer client PC. Note that this is not a check of the server on which you are installing.

Depending on the installation type, your computer needs more specific software installed and specific user rights:

- You only need client Oracle software on your PC
- You need a correctly set-up Oracle client
- Make sure you can connect to the remote database (preferable using Oracle Sql*Plus).

Note

Even when installing the product on a database on the same PC on which you run the installer, it is required to have properly set up Oracle Networking.

If the prerequisites are met, it is then possible to click the **Next** button to continue the installation process.

5.3 Remote Server Connection Properties

In this step you need to define the connection to the remote server. You need the passwords for the **SYS** (as SYSDBA) user and the password for the **UNILAB** user (it is possible that this user does not exists yet).

By default, the screen has the default passwords as set in the install.cfg file. If you changed the passwords after installing the database, you have to make sure to change the passwords in this dialog box. Note also that it is possible to deny SYSDBA access to a remote connection. However, for the installer to work, SYSDBA access must be granted. By default, our installer created scripts with a database that allows access to SYSDBA over the network. Refer to the Oracle documentation for an explanation of the SYSDBA role and the use of the **orapwd** password tool to create this access point for the user SYS. Click **Test Connection** to check that the entered data is correct. You will not be able to proceed if this connection is invalid.

The installer will verify that you can connect to the target database. The application will verify all the necessary components, like Oracle JServer/XSK, XDK, XDB.

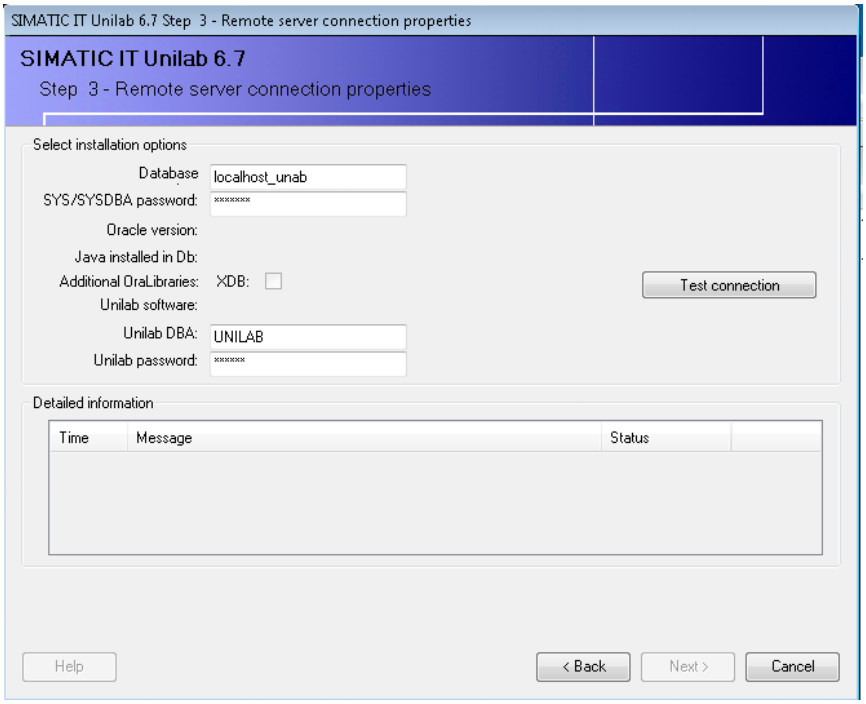
Change the **localhost_unab** with your database (SID) name, chosen during the installation of unilab.

Here you can change the SYS/SYSDBA password (this is set in the **install.cfg**).

Note

This must be the same that you chose when you installed the Unilab database.

Here you can also change/set the Unilab DBA name and password.



After these steps, click **Test connection.**

The **Test Connection** process can take a while, depending on the speed of the network connection and the database server under test.

The connection test will report its progress through messages in the details window (marked in red if something goes wrong) and additional information is displayed behind the information entered by the user.

If some of the required components are missing, the last message will be: “Click next to configure/install these elements and repeat Test Connection”.

The installer will generate the scripts for the necessary additional components.

First select or create a directory that will contain the generated scripts.

Then press **Generate Files**; a number of files will be created underneath the folder you selected.

You can copy these files to the server where you want to create your database (this can be another server, or this computer; it is even possible to generate scripts for a Unix server).

Note

The scripts must be executed on the machine where you want to create a database, and it is advisable to have Administrator (Windows) or root (Unix) access to the server.

If you already have licenses installed, please move the license outside the database to a local pc drive with the ALM console. This is highly recommended.

Running the script on the database server:

- Copy the scripts to the server and open a 'cmd' window (use the same folder where you copied the scripts)
- Connect to sqlplus, set the instance and run the script extend_db.sql
  - **cmd>sqlplus /nolog**
  - **sqlplus>spool extend_db.log;**

- **sqlplus>set serverout on size 999999;**
- **sqlplus>set define off;**
- **sqlplus>set timing on;**
- **sqlplus>set echo on;**
- **sqlplus>set instance <<Your_Instance_Name>>;**
- **sqlplus>@extend_db.sql;**
- **sqlplus>spool off;**
- **sqlplus>exit;**

After you finish the install click "Next" or start the installer again and repeat "TestConnection".

Note

If the Interspec installer ran without error, then the necessary components like Oracle JServer/XSK, XDK, XDB are already installed. If not, we recommend checking the log files created during the Interspec installer. For more information, please refer to the Insterpec Server Installation Manual.

5.4 Define Server Database Options

In this window, you must enter a number of database options. Fill in the settings of the (server) PC where you intend to create the database. It is possible to create scripts on one PC to be executed on another server.

Do not confuse these settings with the settings of the PC you are working on. Even if you run the installer on the PC where you want to create the database, it is possible to have different settings for client and server.

If you generated the installation script before, a setting file called **<SID>.set** is generated in the target directory of the previous installation. In this case, you can use the button **Load Settings from File** to load the settings. This button is typically used when you have to fix a problem in an ongoing installation.

Setting	Description
Operating system	Select the target operating system. This selection will fill in a number of defaults in the following steps that are most suitable for your operating system.
Oracle version	Select the version of Oracle as installed on the server. Only versions that are supported will be listed. If your server version is not listed, please contact your support representative. Installing on a platform that is not supported can result in data loss or problems that are difficult to pinpoint.
Oracle home	Enter the Oracle home path of your server. The Oracle home is typically the location where most Oracle folders are found.
Database name (SID)	This is the name of your database, also known as SID (System Identifier). This must be the name of your Interspec database.
Oracle user name	This is the name of the Oracle DBA user.
Database character set	This is the native character set of your database, a very important parameter. It is recommended to select the character set you will use on your clients. In Western Europe and America, the default WE8ISO8859P1 is the recommended setting. If you want a character set other than those displayed, another character set can be entered. This character set must be known to Oracle. Specifying an invalid character set will cause problems during the installation. Verify the Oracle installation manuals to obtain a list of all supported character sets. Make sure to select a character set that supports ALL characters used by your various clients connecting on your database. Unilab supports single-byte character sets and the Unicode character set AL32UTF8. As of Unilab 6.1, Unicode support has been introduced: the only supported and tested character set for Unicode support is AL32UTF8. Your database will be able to correctly store the characters from clients that work with different character sets (as well as from our Unicode Unilab client). Oracle provides other Unicode-compliant character sets AL24UTFFSS, UTFE and UTF8), but please do not use these character sets to create a Unicode-compatible Unilab database. These character sets have been provided only for backward compatibility and will become obsolete in the future. Oracle also currently supports multi-byte character sets (e.g.: JA16EUC): we do not recommend these character sets, since they have not been tested explicitly (we expect that they will be replaced by Unicode in the long term).

Time Zone	We strongly advise selecting a time zone offset that is the same as your OS server time zone offset during the winter time period. E.g.: On a server running in Rome, the offset in the winter time period is +01:00; the offset in summer time period is +02:00, you will specify +01:00 as your database time zone.
Concurrent sessions	This is the number of sessions that will connect to the database simultaneously. Note that a user can connect with multiple applications on one client-PC, so this is generally higher than the number of users. It also includes users running reports or other applications versus this database. This parameter is used to calculate the **max_processes** parameter in the initialization file. It is safe to set it higher than necessary (this is not directly related to your license).
Memory to allocate	This is the memory that can be allocated to the Oracle instance on that server. When possible, set it to 90%-95% of available physical memory on the server before installing the Unilab database instance. The init<SID>.ora settings will be set to values derived from that value. The number of concurrent connections (**Files** tab) is also important for a correct estimation. When possible, also take into account that other applications can be installed on that server and will also require a part of the memory. It is a good practice to tune memory parameters once the system is in production. Please note that, starting from Oracle 11, we strongly advise using the “Automatic Memory Management” feature introduced by Oracle (only specify MEMORY_TARGET as memory setting in the database parameters (init<SID>.ora or spfile)).
Version control and 21CFR11	When you install ELN/FWB on top of an existing Interspec database, version control and 21 CFR 11 is required.
Lock DBA	Check this flag if the DBA must be locked after installation.
DBA name	This name indicates the owner of the Unilab database. A typical name is 'UNILAB'. Special characters other than alphanumeric characters in US7ASCII are supported, but not recommended.
DBA password	This is the password of the DBA user. Oracle supports single-byte characters for its passwords. [1]
Administrator name	The DBA user defined above is reserved for maintenance operations on the database. It can be locked after database creation (checkbox). The administrator user is created for daily maintenance operations (starting jobs, creating users, etc). Special characters other than alphanumeric characters in US7ASCII are supported, but not recommended.

Administrator password	This is the password of the administrator user. Oracle supports single-byte characters for its passwords. [1]

5.5 Define Operational and Configuration Sizes

This window allows you to define the size of an operational sample, and the size of the configuration.

Not all Unilab databases are used in the same way: therefore, tables and data files will have different sizes in the different implementations of Unilab. In this sheet, you should fill in the number of each object type that should be kept on line (i.e. not archived): for example, 4 years on line, with 10000 samples per year. These settings should be estimated carefully with your customer in order to avoid maintenance problems.

Then click **Estimate Size** to automatically calculate the appropriate sizes for Unilab tables and data files (storage clauses).

5.6 Define Product Tablespaces

Enter the directories, file names and sizes for the Unilab specific tablespace files.

Ideally, these files are put on a RAID-5 system for best performance and data safety.

5.7 Generate Scripts to Create Server Database

This window allows you to select the directory where the files for server installation will be created.

The Generator creates:

- All necessary scripts, and will also create
- A settings file called <*SID*>*.set*, where <*SID*> is the name of the database. You can use this file to rerun the installer later.

5.8 Prepare the Environment for a Server Installation

How to prepare the environment depends on the Operating System of your server. There are two possibilities:

- Windows server
- UNIX Server

Windows server

Proceed as follows:

Step	Action
1	Check if a user **<<Administrator>>** with administrator privileges exists (usually 'Administrator', but this depends on the Windows language).
2	Log on as **<<Administrator>>**.
3	Check the Oracle home directory in the registry (or use Oracle Home Selector to set ORACLE_HOME to the correct Oracle version). **HK_LOCAL_MACHINE > SOFTWARE > ORACLE > <<HOMEX>> > ORACLE_HOME** E.g. C:\ORANT
4	Create a directory to which the required installation files will be transferred. If you ran the installer on the server, you can skip this step, since all required files can be found in the given destination directory. This directory is **<<Unisetup>>**. Using a DOS-prompt: `$ mkdir <<Unisetup>>` or create the directory **<<Unisetup>>**, using Explorer. E.g. C:\Unisetup

Settings used during installation
<<Administrator>> := password :=
<<ORACLE_HOME>> :=
<<Unisetup>> :=

Unix server

Proceed as follows:

Step	Action
1	Check if an Oracle DBA user **<<oracle>>** exists (usually 'oracle' – this user must be a member of the 'dba' group).
2	Check if a root user **<<root>>** with password exists
3	Create a user **<<Unilab>>** with the home directory r/w for this user.

4	In the home directory of Unilab, create a subdirectory to which the required installation files will be transferred. The user 'oracle' must have r/w access to this directory.
	`$ mkdir Unisetup`
	`$ chmod a+rwx Unisetup`

Settings used during installation			
<<oracle>>	:=	password	:=
<<root>>	:=	password	:=
<<unilab>>	:=	password	:=

5.9 Copy the Generated Scripts to the Production Server

Where to copy the files depends on the Operating System of your server. There are two possibilities:

- Windows server
- UNIX Server

Windows server

Note

If you ran the installer on the server, you can skip this step, since all required files can be found in the given destination directory. This directory is then the same as **<<Unisetup>>**.

Proceed as follows:

Step	Action
1	Log on as **<<Administrator>>**.
2	Copy all generated files from **<<Unisetup>>**.
3	Verify that all generated files are present on the production server, and that they are writable.

UNIX server

You can transfer files from a PC to a Unix server with any 'ftp' client.

Example:

In the following example, we use the standard Windows ftp-client (this is a very user-unfriendly tool, so if you have a graphical-oriented ftp-client, things will be easier than in this example). In this example, we assume that the name of the Unix system is ‘unixhost’. The files you want to transfer are located at D:\Unisetup and must be copied to /unilab/Unisetup. We assume that the destination directory /unilab/Unisetup exists and is writeable by all (see previous paragraph).

Proceed as follows:

Step	Action
1	Perform the commands as illustrated in the example below to transfer the files. The commands that must be entered are shown in **bold**. The rest is command feedback:

```
D:\Unisetup> ftp unixhost
Connected to unixhost.mydomain.be.
220 unixhost FTP server (Version 1.7.212.1 Sat Feb
1 01:30:15 GMT 1997) ready.
331 Password required for Unilab.
Password:
230 User Unilab logged in.
ftp> pwd
257 "/home/myuser" is current directory.
ftp> cd Unisetup
250 CWD command successful.
ftp> prompt
Interactive mode Off.
ftp> ascii
200 Type set to A.
ftp> mput *.cmd *.txt *.h *.sql *.run *.plb *.par
*.pack *.bat *.ksh *. *.java
200 PORT command successful.
150 Opening ASCII mode data connection for …
…
… (all specified files are being copied)
…
226 Transfer complete.
ftp> quit
```

Step	Action
2	Verify that all generated files are present on the production server.

FTP Problems

The above example worked when using the ftp software provided on Windows 2000. Some of our customers experienced problems while transferring the **.plb** files with commercial or shareware ftp software. Be careful with the configuration of such tools: it could be that no error is issued during the transfer of a plb file, but still the package doesn't get compiled. For the details concerning this problem, see the problem description provided below:

Description:
UNAPIRA5.PLB package doesn't get compiled when file transferred through ftp on Unix server. Error message in sys.dba_errors: PLS-00801: internal error [56104]
Bug rejected:

This is not a bug in fact. This is an error in the configuration of ftp software used to transfer the file from the CD to the Unix server. I experienced the same problem when working with Reflection FTP.PLB files are transferred in ASCII mode. In ASCII mode, Reflection FTP has some default translating options (convert tabs into spaces - a checkbox in the translation tab of the site properties). That translation is corrupting the encrypted file (.plb file)
UNAPIRA5 was the only package that is containing some tabs. This explains why only that package is concerned by the problem.

5.10 Run the Creation Script

How you run your scripts depends on the Operating System of your server. There are two possibilities:

- Windows server
- UNIX Server

Important

In case of problems, see Server Installation: Troubleshooting . Each step and possible errors are explained therein.

Windows server

Proceed as follows:

Step	Action
1	Log on as **<<Administrator>>**.
2	Make sure all files are in the same directory as the Unisetup.cmd script: then switch to that directory and start Unisetup.cmd. It is best to start the Unisetup.cmd script from within a Windows command prompt (cmd.exe). You can also start the script by clicking its shortcut, but this is **not** recommended. `C:\<<Unisetup>>> Unisetup.cmd`

UNIX server

Proceed as follows:

Step	Action

1	Log on as user **<<root>>**
	`# cd ~<<Unilab>>/Unisetup`
	`# chmod 777 cr_area`
	`# chmod 777 *`
	`# chmod 777 .`
	`# . /cr_area`
	`# exit`
	Make sure the file 'cr_area' is in lowercase. Rename the file if it is created in uppercase (e.g. mv CR_AREA cr_area). All other files will be changed to lowercase by the installation program.

2	Log on as **<<oracle>>**
	`$ cd ~<<Unilab>>/Unisetup`
	`$ ./unisetup`
	`$ exit`

6 Enabling Server Access (Oracle SQL*net)

This chapter describes the configuration steps required for enabling server access. The client/server communication is handled with Oracle SQL*net. Enabling server access involves both server (Windows or Unix) and client configuration steps.

For more details, please refer to Oracle documentation.

Note

This configuration description only describes Sql*Net issues specific to Unilab. The Official Oracle Sql*Net documentation remains the only authoritative reference for Sql*Net configuration. If you install at a site where Sql*Net or other Oracle networking products are already installed, check with the customer before configuring Sql*Net.

6.1 Server Side

Note

Refer to the SIMATIC IT Unilab V6.7 prerequisites for the supported SQL.NET versions. We advise installing only one Oracle networking version on one PC to avoid possible confusion and configuration problems.

Add an entry to listener.ora

Note

On Windows, the listener is already configured in part during Oracle installation. Unless you have specific requirements, you should use the Oracle tools (Network Configuration Assistant) for the configuration of the listener.

You can add an entry to your listener.ora with the Oracle networking tools (recommended), or with an ASCII text editor (e.g. Notepad).

You can then restart the listener. The (optional) password (here below, '**listener**') can be different on your site:

```
lsnrctl  (version 11 or 12)
set password listener
stop
start
exit
```

Note

Oracle recommends using port 1521 as default port for a listener.

If all goes well, you have a new service handler for your database running. It is possible to have more than one listener running on the server. In this case, the name of the listener must be added to each command line.

6.2 Client Side

Add an entry to tnsnames.ora

Starting with this release, the client applications are using Oracle Instant Client in order to connect to the database. In order to set up the connection to the database, the proper tnsnames entry has to be inserted into tnsnames.ora file situated at the following path:

<InstallDir>\SIT\Unilab 6.7\OracleInstantClient\tnsnames.ora

If you have other oracle clients installed on the machine the application will ignore those.

Notes

Oracle provides the possibility to use a name server instead of configuring a **tnsnames.ora** file for every database to which to be connected. The sql*net configuration just points to a name server where the possible database connections are managed for all clients on a specific network. Please refer to the Oracle documentation for more details if you plan to use such a name server. This document and other Unilab documents mention the **tnsnames.ora** file, but the same result can be also achieved by using sql*net name server(s), instead of **tnsnames.ora** files. Please keep this in mind when reading our documentation.

Testing the connection

Test the connection with 'tnsping'.

```
C:\>tnsping myhost_mydb

TNS Ping Utility for 32-bit Windows: Version 2.3.4.0.0 -
Production on 31-JUL-00 15:46:57

Copyright (c) Oracle Corporation 1995.  All rights reserved.
Attempting to contact
(ADDRESS=(PROTOCOL=TCP)(HOST=myhost)(PORT=1521))
OK (120 msec)
```

The error message received in some applications (Configuration, Analyzer, etc.) is:

ORA-06413: Connection not open

In other applications (group-key definitions, life-cycle definitions, etc.), the error message will be:

ORA-12154: TNS could not resolve service name

Just eliminate the parentheses from your path to solve this problem.

7 Client Installation

7.1 Introduction

Make sure the database access (SQL.NET configuration) is enabled on the PC where the client installation is performed. How to enable this access is described in the previous chapter.

SIMATIC IT Unilab 6.7 is delivered with a setup offering 3 options.

- **Unicode:** this setup is to be used whenever only input from languages with single-byte character sets is required. Almost all Western languages (including English, German, Greek, Russian, etc.) are single-byte languages. This installation is described more in detail further on in this chapter.
- **Automation License Manager (ALM):** Installation of the Automation License Manager is only required on those PC(s) that will be used to upload licenses in the database (see the next chapter for a description of how to upload licenses). This installation is not described in further detail.
- **SIMATIC Logon (SLS):** Installation of SIMATIC Logon is required only if SIMATIC Logon will be used for authentication when logging on to SIMATIC IT Unilab, or when the Automation License Manager (ALM) will be installed. This installation is not described in further detail.

7.2 Overview

This Chapter covers the following topics:

7.3 Unicode Client Installation on PC

7.3.1 Overview

This subsection covers the following topics:

	See
Overview	7.3.1
Prerequisite	7.3.2
Standard Unilab Installation on the PC	7.3.3
Oracle Client Installation	7.3.4
Starting Unilab Applications from the Command Prompt	7.3.5
Automatically Starting UNICONNECT after Rebooting the System	7.3.6
Silent (Unattended) Installation	7.3.7

7.3.2 Prerequisite

For proper Client installation, you must have Administrative rights and .Net framework 4.0 installed.

7.3.3 Standard Unilab Installation on the PC

Proceed as follows:

Step	Action
1	Insert the installation DVD into the DVD drive.
2	Use Windows Explorer to browse the **Client Setup** folder and run **Setup.exe**.
3	Follow the on-screen instructions. Click **Next** to go to the following screen. **Note:** Reboot when prompted by the Install Wizard.
4	If the PC on which you installed the Client does not have an English version of Windows, or if you performed the installation in a language other than English, perform the additional steps below to install ADO 2.5 or higher.
5	In order to be able to connect to an Oracle database, the Oracle client files must be installed; please refer to the Oracle Client Installation section.
6	In order to connect to the database a proper tnsname entry have to be added into the tnsnames.ora file which can be found at the following location: <InstallDir>\SIT\Unilab 6.7\OracleInstantClient\tnsnames.ora Or use easy connect syntax: <ip>:<port>/<dbname> In case the DB uses the default, you do not have to provide the port. Example: 172.20.0.8/unab67 or 172.20.0.8:1523/unab67
7	The registry contains a number of Unilab initialization parameters. Detailed directions on the use of each of these parameters are provided in the Unilab Registry Settings installation manual, which is available in the Product Library. All parameters default to working values. Modifying the following registry setting will help the user to connect immediately to the right database. If the setting is not filled out, the user will have to enter this information in the Logon dialog. • **DbName:** Enter the Database Name (Oracle name).

Note
This client installation requires Microsoft ADO 2.5 or higher. On all supported platforms, this should be already present.

7.3.4 Oracle Client Installation

Important

In order to execute the PowerShell script, the execution of scripts has to be enabled on the system.

To enable script execution, execute the following command in PowerShell:

Set-ExecutionPolicy RemoteSigned

Answer with Y(es) when prompted by the system.

Make sure the PowerShell script is executed with **admin** rights!

The installation package does not include Oracle client. The Oracle client files must be included after installation.

Unilab 6.7 is using ODP.NET 4.11.2.0.2 as part of ODAC 11.2.0.2 or Oracle full client 11.2.0.2.

If there is no Oracle 11.2.0.2 installation present on your system, download ODAC from http://www.oracle.com/technetwork/database/windows/downloads/utilsoft-087491.html.

Note

In order to download the ODAC package, a free Oracle Web Account is needed.

To facilitate installation on multiple systems, you can store the downloaded and extracted package on a network share. Once downloaded and extracted, do the following:

- Start PowerShell console as an administrator;
- Execute the Run-CopyOracleFiles.ps1 found in the <InstallDir>SIT\Siemens\OracleInstantClient folder (e.g.: *c:\Siemens\SIT\Unilab 6.7\OracleInstantClient*);
- When prompted, specify the following:
    - The source location of the Oracle files.

      Note: This can be the path to the downloaded and extracted package (local or network share). You can omit this if you want to use the Oracle files from an existing installation.
    - The destination path for the Oracle files.

      Note: By default, this is the same location as the script, therefore specifying it is optional.

7.3.5 Starting Unilab Applications from the Command Prompt

How to Start the Applications

You can start all Unilab applications from the command prompt using the following syntax:

```
<application name> [Us=<user>/<password>][Db=<database>]
```

For the sample management application (**scmgt.exe**), you can also define the language at the command prompt using the following syntax:

```
<application name>
[Us=<user>/<password>][Db=<database>][Lang=<language>]
```

For some applications, you can also define the object and object type using the following syntax:

```
<application name> [Us=<user>/<password>][Db=<database>]
[Ob=<object>[/<object type>] [/NoStartup]
[process_mode=<process mode>] [run_mode=<run mode>]
```

Example

Below is an example:

```
lydef us=LimsAdministrator/ULABSU Ob="Basic Layout/sclist"
```

This command starts the **Define Layout** application with:

- **Username**: LimsAdministrator
- **Password**: ULABSU
- **Object**: basic layout
- **Object type**: sclist.

Arguments

An overview of all available arguments is provided below:

Argument	Meaning/Possible Values
Ob/obtype	Indicates which object will be loaded first. The object type is only required when relevant. Syntax: • **addef us=<*user*>/<*password*> Ob=<ad>** • **eqdef us=<*user*>/<*password*> Ob=<eq>/<lab>** • **lcdef us=<*user*>/<*password*> Ob=<lc>** • **lydef us=<*user*>/<*password*> Ob=<ly>/<ly_tp>** • **tkdef us=<*user*>/<*password*> Ob=<tk>/<tk_tp>** • **updef us=<*user*>/<*password*>**
Lang	Indicates the language of the application. You can also use **'LanguageId='** in accordance with the registry key. Examples: • **scmgt.exe Lang=Dutch** • **scmgt.exe LanguageId="English_(UK)"**
NoStartup	Suppresses the splash window
process_mode	• **== NOWAIT**: The caller starts the application in a parallel path and can continue to work • **<> NOWAIT**: The caller waits until the application is ready
run_mode	• **SW_HIDE**: Hides the window and activates another window. • **SW_MINIMIZE**: Minimizes the specified window and activates the next top-level window in the Z order. • **SW_RESTORE**: Activates and displays the window. If the window is minimized or maximized, the system restores it to its original size and position. An application should specify this flag when restoring a minimized window. • **SW_SHOW**: Activates the window and displays it in its current size and position. • **SW_SHOWMAXIMIZED**: Activates the window and displays it as a maximized window. • **SW_SHOWMINIMIZED**: Activates the window and displays it as a minimized window. • **SW_SHOWMINNOACTIVE**: Displays the window as a minimized window. This value is similar to **SW_SHOWMINIMIZED**, except the window is not activated. • **SW_SHOWNA**: Displays the window in its current size and position. This value is similar to **SW_SHOW**, except the window is not activated. • **SW_SHOWNOACTIVATE**: Displays a window in its most recent size and position. This value is similar to **SW_SHOWNORMAL**, except the window is not activated. • **SW_SHOWNORMAL**: Activates and displays a window. If the window is minimized or maximized, the system restores it to its original size and position. An application should specify this flag when displaying the window for the first

Argument	Meaning/Possible Values
	time.

Some Notes:

There is a difference between starting a Unilab application from Windows and 'internally' from within another Unilab application, because the options **process_mode** and **run_mode** function only from within Unilab.

Arguments that contain one or more spaces must be written between double quotes.

Examples:

- Us="Kirk"
- Ob="Basic Layout/stlist"-

7.3.6 Automatically Starting UNICONNECT after Rebooting the System

Introduction

After rebooting the system, UNICONNECT can be started up automatically on the Unilab server without a Windows logon window.

Procedure

Proceed as follows to automatically start UNICONNECT after rebooting the system:

Step	Action
1	Start UNICONNECT as a scheduled task for Windows: • Indicate that the task needs to be performed **When my computer starts**. • Enter the user name and password that will be used to connect to the network.
2	Change some of the strings of the registry settings for \HKEY_LOCAL_MACHINE\SOFTWARE\SIMATIC IT Unilab\Version x.x.x\Uniconnect\General: • Enter the Unilab user name for the string *UserID*. • Create a new string *PassWord* and enter the password as value. • Enter the database name of your Unilab server in the string *DbName*.

7.3.7 Silent (Unattended) Installation

Creating the Setup File

For an unattended installation, you need file **setup.iss**. This file can be created as follows:

- Set the parameter **EnableLangDlg** in the setup.ini to N ('EnableLangDlg=N').
- Run **setup.exe** with command-line options –NoOraTest–r

After the installation, there will be a file **setup.iss** in the Windows system directory.

Copy this file to the installation directory.

Silent Installation

Run **setup.exe** with command-line options *–NoOraTest –s* to perform a silent installation.

The unattended installation will create file **setup.log** with the result code in the installation directory. ResultCode=0 means that the installation was successful. A negative resultcode means that an error occurred.

7.4 Upgrading the Unilab Client

For more details on upgrading the Unilab Client, refer to the related document on the Unilab product CD. This document can be found in the Product Library.

7.5 Supported Oracle Client Versions

Unilab 6.7 is using ODP.NET 4.11.2.0.2 as part of ODAC 11.2.0.2 or Oracle full client 11.2.0.2. Oracle client installation is not mandatory in order to use the application. Please refer to the Oracle Client Installation section**.**

8 Uploading the Licenses

8.1 Prerequisites

Before uploading the licenses, make sure the following prerequisites are met:

- Make sure the Automatic License Manager (ALM) is properly installed.
- Make sure you have the license key available (on diskette, on the hard disk, etc.).

About the installation of ALM

- ALM can be installed using the dedicated installer which is included on the CD and can be found in the ALM_and_Simatic_Logon folder.
- The SIMATIC IT Unilab database installation must already have been completed.

8.2 How to Activate a License Key

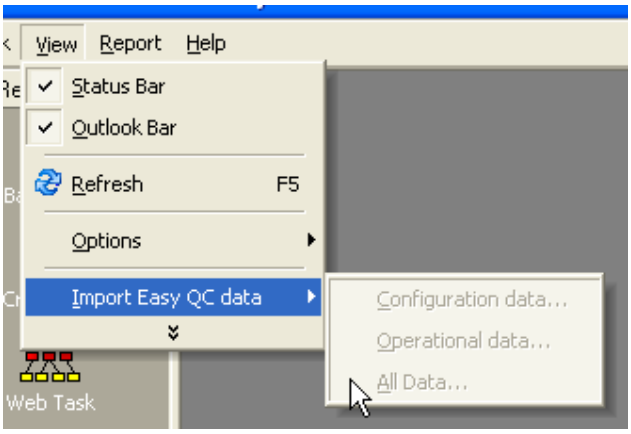
Proceed as follows to activate a License Key:

Step	Action
1	Select the following menu-bar entry: **Edit > Connect Target System > Connect to Oracle**. Make sure the selection in the Navigation pane is on 'My computer'.[1] *If this entry is not present, please refer to section Troubleshooting & known issues in the SIMATIC IT Unilab – Use of ALM (Licenses) End-User Document in the Product Library.*
2	Fill out the requested connection info. The User/Password must be the "LimsAdministrator" user (default pwd "ULABSU"), or have the UNILAB_DBA role.
3	Click the 'ACTIVATE database connection' button, and make sure the connection is OK.
4	Click **OK**.
5	If the database connection was activated correctly, the database is added to the navigation pane of the ALM GUI. *It might be necessary to select 'My Computer' in the navigation pane before the database connection is added.*
6	Select the licenses you want to transfer (on the hard disk, floppy, etc.).
7	Select the following command: **License Key > Transfer**.
8	Click **OK**.
9	Select the database to which you want to transfer the licenses.
10	Click **OK**. **Result**: The license files are transferred

[1] The Oracle connection will be assigned to the selected PC. During the actual transfer, the connection will be created by the 'Automation License Manager Service' that is running on this selected PC. The transfer will fail, if the SIMATIC IT add-on is not correctly installed on that specific PC, or if the Oracle Connection cannot be established on it.

9 Installing the Demo Data

Proceed as follows:

Step	Action
1	Start Unilab Analyzer with the following command path: **Start -> Programs -> Unilab -> Analyzer** Log on with the LimsAdministrator account (default: LimsAdministrator/ULABSU)
2	Import the data with the following command path: **View -> Import Easy QC Data -> All Data** 

Important notes on importing demo data

A dedicated access right is required to import the demo data into the database. This access right is granted to the LimsAdministrator user after installation. Newly-created users do not have this access right.

The import is based on the SIMATIC IT Unilab XML Interface. This offers a great deal of flexibility (data must not be empty, versioning is applied if applicable, independency of the Oracle version). Since this implies that all data is not just imported, but really processed, the import can take a while (1 hour).

The application will look in the folder specified by the registry setting "EasyQcDir" in the registry (it is the user's responsibility to check that this setting satisfy his/her requirements). Keep in mind that only those files that match the expression "Config*.xml" will be considered.

The difference between the 3 available import menus (i.e. **Configuration data**, **Operational data** and **All data**) is just in the file masks.

The file names are processed in alphabetical order. To force a specific order in which the files must be processed, a numbering sequence must be added to the file names' prefix (i.e. **Config***). For example, in order to import file Config*_PR before Config*_PP, you can assign a specific numbering sequence to the file names' prefix: Config 07_PR.xml and Config08_PP.xml. Considering that the number 7 precedes the number 8, in this case, the prs will be processed before the PPs. Otherwise, if such numbering sequences are not used, the pps will be processed before the prs (e.g. in the given example, Config08_PP.xml will be processed before Config 07_PR.xml).

10 Enabling Centralized Quality Management Support on the Database

10.1 Before proceeding

Carefully read the Unilab **Centralized Quality Management** Concepts and User Manual.

10.2 Procedure

Important: If you have EasyQC installed, Centralized Quality Management is already enabled: therefore, disregard the procedure described below.

To enable Centralized Quality Management support on the database:

Step	Action
1	Start the Event Manager
2	Customize the script enableppkey.sql in the <<unisetup>> directory. This script creates: • The entries in the table utkeypp. • The necessary sample-type and sample group keys. • The event rules for child and derived objects synchronizations. • The right number of entries in the hierarchic layout for parameter profiles (should be consistent with the number of entries in utkeypp.

3	• Set the current directory properly in order to have all necessary scripts available:
	cd <<Unisetup>>
	• For Oracle 10 or 11: **<<SQLPLUS>>**
	• SQLPLUS /nolog
	• Connect as the **unilab dba** user to the Unilab database in SQLPLUS
	• SET DEFINE OFF
	• SET SERVEROUT ON
	• SPOOL enableppkey.log
	• Verify that you are connected on the right database.
	`select * from global_name;`
	`@enableppkey`
	• Stop all jobs @stopevmr OR BEGIN cxapp.stopalldbjobs; END;
	• Start all jobs @strtevmr OR BEGIN cxapp.startalldbjobs; END;
	• SPOOL OFF

4	Extend UNACTION with the functions called in the event rule and customize these functions when necessary

11 Enabling 21 CFR Part 11 Support

Important

This chapter concerns only databases in which 21 CFR Part 11 has not been enabled during Unilab database installation. Therefore, it has been specifically written to provide information on how to upgrade the database manually.

11.1 Before proceeding

Carefully read the Unilab 21 CFR Part 11 User Guide.

11.2 Procedure

To enable 21 CFR Part 11:

1. Check the Unilab License.
2. Prepare the Execution of the Scripts in SQL*Plus.
3. Adapt the Layouts.
4. Modify Configuration Elements.
5. Enable and Tesunapiget E-mail Notification.
6. Verify the Life Cycles for Configuration Objects.
7. Disable Normal Operations for the DBA.
8. Turn On Session Auditing.

11.3 Check the Unilab License

To enable 21CFR Part 11 for Unilab, you must install the Unilab Advanced Server License.

To check all installed licenses on your db, open the **Automation License Manager** application and connect to your database.

A Unilab license for 21 CFR Part 11 is indicated as **Unilab Advanced Server**.

For more information about licenses and the use of Automation License Manager, please refer to the SIMATIC IT Unilab – Use of ALM (Licenses) End-User Document in the Product Library.

11.4 Prepare the Execution of the Scripts in SQL*Plus

Location of the scripts on the CD

You can find all scripts in the folder **\Server Setup\db**. Copy all files in this folder (except for the files in the subfolders) to a local directory.

In this context, we will refer to this local directory as **server_scripts**

You can fill in the setting you will use during the execution of the scripts:

```
server_scripts =
```

Prepare the UNIX environment

See the **Upgrading Unilab** upgrade manual to set up the environment and transfer the files from the folder **server_scripts** to your Unix server.

Prepare the Windows environment

See the **Upgrading Unilab** upgrade manual to set up the environment and transfer the files from the folder **server_scripts** to your Windows server.

Spooling the output

In the steps below, replace **<<SQLPLUS>>** with the command **SQLPLUS /nolog** to start the Oracle DB Administration tool on the corresponding server OS platform.

Proceed as follows to spool the output:

1. Connect in **<<SQLPLUS>>**as **unilab dba** user.
2. Turn on **DBMS_OUTPUT** and spool output to a file.

```
SET SERVEROUT ON

SPOOL enable21CFR11.log
```

Make sure the event manager is still running.

11.5 Adapt the Layouts

There are two options to adapt the layouts:

- If you want to replace all layouts that are installed during a new installation, execute the script **@dblydata**.

  Remark: This script will only change the basic layouts.

- If you want to add the version column in all layouts (standard and custom), execute the script **@updlydata421CFR11**. The script will add the version control columns where logically necessary and when these columns are not already available. You will need to make some adjustments (e.g. the display width).

The script can run several times on the same database without causing any damage.

Run both scripts when you want to enable 21 CFR Part 11 on **easyqc** and **easysl** databases.

11.6 Modify Configuration Elements

When necessary, the scripts **@easysetcon** and **@upddata421CFR11** will make the following modifications to the configuration elements:

- Set all columns to **protected** on the **Audit trail** tab.
- Disable the editing of operational specifications (functional access rights and layouts).
- Disable the deletion of functional access rights.
- Extend the Windows titles with version info.
- Set the preference **LockApplication** to 10 minutes on all levels (system, user profiles and users).
- Enable **Audit trail** and **Audit trail details** for all new operational objects. The audit trail is not selected for operational objects that were created before this script was run.
- Enable **Audit trail** for all configuration objects (existing and future objects). In this release, **Audit trail details** are not available for configuration objects.
- Check the object status. For the status **Approved**, the setting **allow_modify** must be set to **0** and **active** must be set to **1**. The status definition is corrected and, when necessary, a warning is triggered.
- Correct the status for version control. Change the status of all existing configuration objects to **Approved** when:
  - **version_is_current** is **1** and **allow_modify** is **1**
  - **version_is_current** is **1** and **active** is **0**

11.7 Enable and Tesunapiget E-mail Notification

Execute the script **@testemail421CFR11** to create a sample type **DUMMY_ST** and perform the necessary actions to trigger e-mail notification when the life-cycle rules for 21 CFR Part 11 are violated.

Use the command below to disable spooling:

```
SPOOL OFF
```

When the script does not send e-mails, check the system settings related to e-mail notification (all settings that start with **SMTP%** and **DBA_EMAILADDRESS**).

Please see para. 3.4.1 regarding email notification.

11.8 Verify the Life Cycles for Configuration Objects

Normally the system life cycle is upgraded as such during the upgrade to Unilab 5.0 (or higher), so it is impossible for a normal user to change the status of an object from current (**allow_modify** = **0** and **active** = **1**) to modifiable or inactive (**allow_modify** = **1** or **active** = **0**).

What to check?

You should check the following with care:

- The system life cycle
- The status definitions
- The different life cycles for configuration objects with version control
- All transitions to verify whether it is impossible to change the status of an object from current (**allow_modify** = **0** and **active** = **1**) to modifiable or inactive (**allow_modify** = **1** or **active** = **0**).

E-mail notification

When a configuration object in Unilab is modified and Unilab detects the violation of a rule, the application will send e-mail to the Unilab administrator (in the system setting **DBA_EMAILADDRESS)**.

Please see para. 3.4.1 regarding email notification.

Transitions in life cycles

In the system life cycle, the transition from **Approved** to **In Editing** has been modified:

- The user authorization has been set to **~Dynamic~**.
- The verification is coded in the package function **UNACCESS.TransitionAuthorised**.

It is best to use the same principle for similar transitions in life cycles for configuration objects.

11.9 Disable Normal Operations for the DBA

You should not use the Unilab DBA user account for normal operations, such as entering results or creating samples. Reserve this user account user for maintenance operations such as: creating and maintaining databases, creating users, archiving operations, load data operations, etc.

Disable the normal operational tasks for the DBA user account in the **User Management** application.

Remark: The DBA user is allowed to change configuration objects from **Approved** to **In Editing**. In this case, it will be impossible to set that object version as the current one. First, the status must be changed to **Approved** again (notification is done through e-mail).

11.10 Turn On Session Auditing

Turn on database connection auditing in the **init<sid>.ora** (or in the spfile, when applicable) with the following statement:

```
audit_trail = DB
```

You can check this setting on your database with the following query:

```
SELECT *
FROM v$parameter
WHERE name='audit_trail'
```

Connect as the Unilab DBA in **<<SQLPLUS>>** and submit the following statement:

```
AUDIT SESSION BY SESSION
```

Now all connections will be traced in the table **sys.aud$**. Depending on the project, the DBA must organize the regular maintenance of this table, because, once the table is full, normal users will not be able to connect on the database.

12 SIMATIC IT Unilab Archiving

Post-installation

The Unilab server installer installs a pre-installed Archiving module. This module can be used to archive data to file only. The 'radef' application that is used for archiving uses the same mechanism for accessing files on the database server as explained for Unilink in the Unilink on-line help.

For archiving data to another database, some additional steps are required. They are described in the SIMATIC IT Unilab – Archiving manual in the Product Library.

13 Special Installation Issues

Customizing init.ora

The database parameter file is created during the execution of 'Unisetup'. The parameters are taken from the sheet 'init.ora'. A number of parameters can be customized for your database.

Important

The most important parameters during database creation are the database name (the SID) and the database block size. These two parameters cannot be changed after the database has been created.

All other parameters can be changed after database creation, although it is often necessary to restart the database to change one of these parameters.

If you are not sure about the meaning of these parameters, start creating your database without changing any of these parameters. Adding an invalid entry or removing a required entry will prevent the database from being started.

Some parameters are only changed during a tuning session. If this is necessary, you should keep a log of changes that have been made, and the result you obtained.

Some parameters are used only in specific versions of Oracle or the operating system.

The Event Manager does not start after installation

It is possible that the **Event Manager** does not start after executing the installation procedure; the following error is displayed: **Requesting lock U4EVMGRSTART_OR_STOP returned 4**. In this case, there are two options to start the **Event Manager**:

- Stop and start the database again.
- Do the following:

```
delete from sys.dbms_lock_allocated
where name='U4EVMGRSTART_OR_STOP;
commit;
```

Stopping and starting the database

The Unilab installation CD contains a script to start (**StrDb.bat**) and stop (**Stopdb.bat**) the database.

- When you run the Stop script, all database jobs will be stopped and a complete shutdown of the database will be performed.
- When you run the Start script, the database and all database jobs will be restarted.

You can also stop and start the database service using **Services** (**Control Panel > Administrative Tools > Services**), but this will only stop the database jobs that are running. When you restart the service, only these particular jobs will be started.

UNDO Retention and size of UNDO tablespace in Oracle 9 (or higher)

As of Oracle 9, parameter UNDO_RETENTION may be set neither too high, nor too low. Setting this parameter too low causes the exception 'ORA-01555: snapshot too old' to be raised too frequently. Setting it too high may result in an UNDO tablespace that is too small. There must be a good balance between the size of tablespace and the UNDO_RETENTION parameter. The view v$UNDOSTAT will help the dba in tuning these 2 factors.

Extract from a technical note from Oracle:

> It should be noted that 9i introduced an optional init.ora parameter called UNDO_RETENTION which allows the DBA to specify how long the system will attempt to retain undo information for a committed transaction without being overwritten or recaptured. This parameter, based in units of wall-clock seconds, is defined universally for all undo segments.

Using UNDO_RETENTION can potentially increase the size of the undo segment for a given period of time, so the retention period should not be arbitrarily set too high. The UNDO tablespace still must be sized appropriately. The following calculation can be used to determine how much space a given undo segment will consume given a set value of UNDO_RETENTION.

```
Undo Segment Space Required = (undo_retention_time *
undo_blocks_per_seconds)
```

As an example, an UNDO_RETENTION of 5 minutes (default) with 50 undo blocks/second (8k blocksize) will generate:

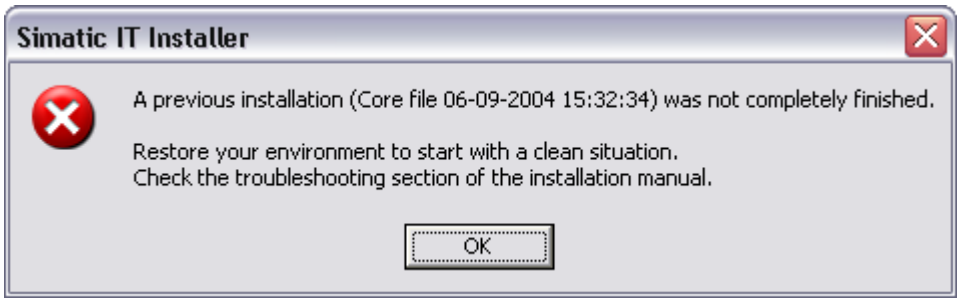
```
Undo Segment Space Required = (300 seconds * 50 blocks/
seconds * 8K/block) = 120 M
```

The retention information (transaction commit time) is stored in each transaction-table block and each extent-map block. When the retention period has expired, SMON will be signaled to perform undo reclaims, which is done by scanning each transaction table for undo timestamps and deleting the information from the undo-segment extent map. Only during extreme space constraint issues will the retention period not be observed.

14 Server Installation: Troubleshooting

14.1 Previous Installations

When the Installer application shows the following message, something went wrong with a previous installation.



Actions

- Restore your environment on the basis of the actions executed in the previous installation. When parts of the database (service, data files, etc.) have been created, use the standard Oracle tools to remove the database.
- Remove all files from the **%TEMP%\Installer** directory. Note that the **%TEMP%** points to your temporary directory as defined in the environment variable **TEMP**.
- Restart the Installer application.

14.2 Troubleshooting Script creation

- **Problem**: SYSTEM tablespace should be at least 600 MB with 16 datadomains; 800 MB for 128 datadomains.

  Solution: If you encounter this, you must increase the size.
- **Problem**: SYSTEM tablespace seems too large.

  Solution: The size of the SYSTEM tablespace is exceptionally high. Since a user should not store data in the SYSTEM tablespace, a size of approx. 600MB should be sufficient for this tablespace (depending on the number of datadomains). If you are sure about the size, you can leave this without performing any corrective action. The size of the SYSTEM tablespace must be increased only if you have an exceptionally high number of users in your system.
- **Problem**: Tablespace is larger than x GB. Most 32-bit file-systems have problems with files > x GB (x is function of the chosen OS and RDBMS versions).

Solution: The given size is higher than x GB. This may cause unexpected behavior on a number of operating systems, as well as possible installation failure. Some operating systems cannot handle files that are so large, whereas others will work less efficiently. If you are sure your OS is capable of handling files of this size, you can leave this without performing any corrective action. On some operating systems, you must increase OS-limits (e.g. IBM/AIX: **/etc/security/limits**).

Also check 'ulimit' settings on Unix. You might also want to optimize the db_block_size of your database and the block size of your file-system.

Note

The maximum Oracle file-size limit is 64GB (max 4,000,000 blocks x max 16kb/block).

If you need a datafile of this size, you probably have to split up your tablespace into multiple files. When necessary, the installer sees to splitting the operational datafiles. On most 32-bit systems, 2GB is the maximum supported filesize: on Windows, the maximum supported filesize is 4GB

Large NTFS file-systems can be created, but Oracle bugs prevent the creation of files exceeding 4 GB. Also note that it is not only a matter of creating large file-systems, but also the manner in which device drivers treat the space exceeding 2Gb.

If you want to use files > 2GB, be sure to read the Oracle technical note (available from Oracle Support):

Article-ID:	<Note:62427.1> Alias: 2GB.GENERIC
Circulation:	PUBLISHED (EXTERNAL)
Folder:	server.Rdbms.DBA.Admin
Platform:	GENERIC Generic issue
Topic:	** Database, Tablespace and File Management
Subject:	2Gb or Not 2Gb - File limits in Oracle

- **Problem**: TEMPORARY tablespace is too small.

  Solution: The minimum size is 20MB. If you want a small temp tablespace, you can keep the 20MB: otherwise, reconsider this. The temporary tablespace is used by Oracle internally for intermediate results or for sorting and index-creation.

- **Problem**: Your database is very large; you will probably have to customize the installation manually! (Edit cr_db[.cmd] before running Unisetup[.cmd]).

  Solution: See above (i.e. Problem: Tablespace larger than 6 GB).

- **Problem**: UNDO tablespace is too small.

  Solution: The minimum size is 20MB. For optimal performance in Oracle 10 and higher, the UNDO tablespace is adapted to the number of users. If you want to keep this tablespace small, decrease the value for 'number of concurrent users'. This will not affect other settings and can be done safely. In the other case, just increase the tablespace size.

- **Problem**: The number of concurrent users you specified is very high. You probably have to customize your database manually! (Edit cr_db[.cmd])

  Solution: This is not really an error, but if you want to support a very high number of concurrent users, you will probably have to tune your database manually. Note that, in this case, you probably need a SYSTEM tablespace that is larger than normal.

14.3 Troubleshooting Database creation (Step 1 – Step 8)

This part creates the Oracle instance and is a pure OS/RDBMS installation. Problems in this field are not specific to Unilab and can be solved by anyone knowledgeable about Oracle and the target OS.

Note

You can restart from a specific installation step by specifying the step-number as an argument:

E.g. Unisetup 6 will start from step 6.

If you cannot get past a specific step, it is still possible that something went wrong in a previous step (e.g. if you create a small tablespace, that will succeed in its step, but creating tables will fail due to this first step). The next chapter explains how to completely restart a partial installation.

STEP 1:

on Windows: Installer prerequisites

- On Win; make sure 'cmd'-extensions are enabled (this is the default Windows behaviour, but can be overridden in the registry).

on Unix: create oratab entry

- You must be root-user to perform this step.
- Make sure the file is present in the current directory.
- If you get 'syntax errors' in the scripts, make sure that files **cr_area** and **Unisetup** have been transferred in ASCII mode
  (if you open file **cr_area** in vi and see **^M** at the end of each line, you transferred the files in binary mode!)
- Check the file /etc/oratab if you have problems.

STEP 2:

on Windows: update registry

- Permission problem (no Administrator rights)
- Oracle has not been installed properly (e.g. the ORACLE-tree does not exist in the registry)

on Unix: create Oracle directories

- Check paths and permissions.

STEP 3:

on Windows: create Oracle directories

- Check the paths and permissions.

on Unix: make scripts lower-case

- Make sure all scripts are copied and that you are root

STEP 4: create init<<SID>>.ora file

- Check permission and filename

STEP 5:

on Windows: create Oracle services

- Check file **ORADIM.LOG** in the RDBMS directory: there is a Oracle technical note illustrating all possible problems that may be encountered in this step.
- The installation might have been considered as failed due to an old entry in this file. Check if old entries are present or not. Clear old entries from **ORADIM.LOG** or delete this file when it contains old errors before reinstalling.
- Permission problem or database already exists
- init<<ORACLE_SID>>.ora files not found or invalid
- Error in provided path
- Some files not deleted after a failed installation
- ORACLE_HOME specified incorrectly
- Database or service with this SID already exists

on Unix: create Oracle password file

- Try the 'orapwd' command yourself. The file must be named orapw<sid> and put under $ORACLE_HOME/dbs.

STEP 6: create Oracle instance and SYSTEM tablespace

- Check the paths, ORACLE_SID
- Check whether you specified the correct Oracle version (compatible parameter) in **init<sid>.ora** file
- Check available diskspace / required diskspace
- Check OS resources (memory, swap space, kernel parameters, etc)
- Check password file cfr. step 5
- If you get an ORA-3113: check 'Oracle Names' configuration file for hidden file '.sdns.ora'. A number of Oracle versions core-dumps when this file exists.
- If this step takes considerably longer (e.g. more than 10 minutes) to finish than expected, it could be that the process is locked. Check what happens if you press ctrl-D (on UNIX) or ctrl-Z (on Windows)

STEP 7: create SYSTEM catalogs

- This step executes the standard Oracle scripts (catalog.sql, catproc.sql, etc.)
- Try to find out which script failed for which reason (resource problems, database not open, etc.)
- You can re-execute any of these scripts without clean-up

STEP 8: create other tablespaces

- This step creates a number of files in the file system
- Check the paths, ORACLE_SID
- Check available disk space / required disk space

STEP 8a: installs the Java virtual machine in the Oracle database

- This step adds a Java virtual machine in the Oracle database.
- It is a mandatory install step.
- Standard Unilab will use it to access directories and ASCII files on the server. It is also used to send e-mails.
- Check available disk space / required disk space
- Check the size of the system tablespace (a lot of Java classes are added).
- DROP JAVA SYSTEM before rerunning this script.
- Step 11 will fail when not installed correctly.

14.4 Troubleshooting Unilab Installation (Step 9 – Step 17)

If you have reached this point, you have successfully created an empty Oracle database. The next steps create the Unilab user, data structures and PL/SQL packages. Unless you run out of resources (disk-space or memory), it will be a Unilab-specific problem.

Make sure you have all the necessary files with the correct permissions located in a single installation directory. Make sure your DBUTCO.SQL/DBUTOP.SQL are up-to-date with relation to the other *.SQL and *.H files.

STEP 9: create database administrator

- Make sure you copied the correct files (*.sql, *.plb, .so, .dll and *.h)
- Performing this step can be reattempted without cleaning up, but this may result in the error `ORA-01920.` This is normal after a restarted installation. If that is the only error, you can restart with step 10.
- If you want to restart the procedure from this step, drop the unilab dba user and the lims administator user before relaunching from this step.

STEP 10: create database objects

- This step creates all tables, indices, sequences and so on.
- This fails if there is not enough space in the related tablespaces. Check the sizes of the tablespaces in table SYS.DBA_DATA_FILES. It might be necessary to recreate the tablespaces with larger sizes
- Performing this step can be reattempted without cleaning up, unless the tablespaces were created too small
- In extreme cases, it might occur that the first extent of a table does not fit in the datafile. If this happens, you will have to alter the storage clauses for these tables manually.

STEP 11: compile all packages

- This step requires a lot of memory and CPU resources. Make sure you have sufficient memory in your system or extend your swap files.
- Make sure SYSTEM tablespace is sufficiently large.
- If one package compilation fails, other packages might fail as well. First try to solve the first error and perform this step again. Make sure you have all the necessary files (.plb, .sql, .h)
- Check the table SYS.DBA_ERRORS for more info on compilation failures
- Performing this step can be reattempted without cleaning up (a second run can take a considerable amount of time, because existing packages have to be rebuilt).

STEP 12: install unilink

- This step is dependent on the Java virtual machine installed.
- This step attempts to create Unilab Java classes.

STEP 13: install initial data

- Check for missing files
- Performing this step can be reattempted without cleaning up

STEP 14: start event manager and install initial data

- Check for missing files
- Performing this step can be reattempted without cleaning up

STEP 15: initialize custom functions

- Check for missing files
- Performing this step can be reattempted without cleaning up

STEP 16: OPTIONAL: lock the Unilab dba user

- Check for missing files
- Performing this step can be reattempted without cleaning up

STEP 17: OPTIONAL: stop and start the jobs as the Lims Administrator user

- Check for missing files
- Performing this step can be reattempted without cleaning up

14.5 Cleaning up a Failed Installation or Uninstalling the Server

It might happen that there were so many problems that it would be best to restart the installation from the beginning. Depending on the point you reached in the installation process, you might end up with a partially-installed database.

Notational convention: substitute <<*NAME*>> for its actual value

To clean up, you need to know:

- The name <<*ORACLE_SID*>> of your database
- The Oracle home directory <<*ORACLE_HOME*>>
- The exact location of all data-files, log-files, and control-files.

Specific to Windows

1. Go to the Control Panel; select Administrative Tools > Services and search for
   - OracleService<<ORACLE_SID>>

   Stop this service.
2. It might be necessary to 'shutdown abort' the database.
3. If you cannot find any lines that refer to this service, then the service has not been installed, and, most probably, something went wrong very early in the installation.

Important

Be extremely careful when deleting files, especially if there are other databases installed on your system.

4. If you are unsure, rename the files before deleting them, and check if other systems still work properly. By default, all filenames related to the database contain the <<*ORACLE_SID*>> in their names.
5. You can delete the service with the following command:

```
Oracle 11 or 12: ORADIM   -DELETE -SID <<ORACLE_SID>
```

6. Search your local hard drives (you should never install the database on a network drive) for the <<*ORACLE_SID*>> (with 'Find Files'). You will find one or more directories with this name. Make sure these directories are really the database directories and delete them. If you have the creation-scripts at hand, you can look in cr_area.cmd for paths and directories that were created by the installer.
7. Find and delete the following files in <<*ORACLE_HOME*>>\DATABASE

```
init<<ORACLE_SID>>.ora
install_init<<ORACLE_SID>>.ora
hc_<<ORACLE_SID>>.dat
```

8. The following is a hidden file. It might be necessary to change its properties before you can delete it.

```
pwd<<ORACLE_SID>>.ora
```

9. Find and delete the following files in <<*ORACLE_HOME*>>
   - Oradim.log
   - Oradim*.log

Specific to UNIX

1. Perform:

```
$ ps -ef |grep <<ORACLE_SID>>        (on UX)
$ ps auxw |grep <<ORACLE_SID>>       (on Digital UNIX or
AIX)
```

2. Check if you find lines such as:

```
oracle 25895     1  0  Jan 11  ?         0:04
ora_smon_un42
```

3. If you do not find these lines, your database is not running and you can proceed with step B.

Step A: stop the running instance

If there are lines like those above, your database (in this case, 'un42') is running. To stop it, do the following:

1. Log on as an Oracle DBA user (e.g. 'oracle')

```
$ export ORACLE_SID=un42
$ . /usr/local/bin/oraenv (on UX and Digital UNIX)
OR
$ . /usr/lbin/oraenv      (on IBM AIX)
```

2. If this does not work, either Oracle has not been installed properly, or the database has no entry in /etc/oratab. If this is the case, you have to specify ORACLE_HOME manually and adapt your PATH appropriately.

```
$ sqlplus /nolog
> connect sys/<<sys_password>> AS SYSDBA
> select * from global_name ;
```

Important

Make sure this returns the correct database-name (in this example, un42).

```
> shutdown abort
> exit
```

Step B: delete the database files

Find and delete all files from the database. You can check in the 'cr_area' which directories have been created.

Important

Be extremely careful when deleting files, especially if there are other databases installed on your system.

If you are unsure, rename the files before deleting them, and check if other systems still work properly. You can use the 'fuser' command to check if a file is in use by another process. By default, all filenames related to the database contain the <<*ORACLE_SID*>> in their names.

14.6 General Errors

Invalid packages

While the jobs are running, it may happen that, for some reason, packages become invalid. Attempting to stop/start jobs may result in a major error: "*Major Error: Requesting lock U4EVMGRSTART_OR_STOP returned 4*", and a new entry will be created in the **uterror** table.

Action

To solve the problem:

- Delete the record in *sys.dbms_lock_allocated*

How

- Use the following SQL query:

```
DELETE FROM sys.dbms_lock_allocated WHERE name LIKE
'U4EVMGRSTART_OR_STOP'
COMMIT
```

15 Server Installation: Tips & Tricks

15.1 Cannot Connect with SYS AS SYSDBA to Remote DB

On Windows, this is generally not a problem, but, on UNIX, it is important to create a password file using the orapwd tool. If you created the database using the installer scripts, then this is done for you; if you used another tool to create the database, this might be a manual process. Make sure you have a password file and the correct settings in your init.ora (REMOTE_LOGIN_PASSWORDFILE= EXCLUSIVE).

Try to connect to the database using **Sql*Plus**. The Installer uses **Sql*Plus** behind the scenes, so manually performing what the installer attempts to do might reveal more details about the source of the problem.

15.2 How to Unlock the DBA User

After installation, the DBA user is locked (by default). To unlock this user, execute the following steps.

```
<os> sqlplus /nolog
<sqlplus> connect SYS/<<sys password>> AS SYSDBA
<sqlplus> ALTER USER <<unilab dba>> ACCOUNT UNLOCK
<sqlplus> EXIT
```

15.3 How to Stop and Restart the Unilab DB Jobs

Stopping/restarting the Unilab DB jobs requires the appropriate privileges. After a default installation, this can be done by the following users:

- SYS (default password: Manager)
- The Unilab DB user (default user/password: UNILAB/ulabsu)
- The LimsAdministrator user:
  (default user/password: LimsAdministrator/ulabsu)

```
<os> sqlplus <user/password>
<sqlplus> BEGIN
<sqlplus> UNILAB.CXAPP.STOPALLDBJOBS();
<sqlplus> UNILAB.CXAPP.STARTALLDBJOBS();
<sqlplus> END;
<sqlplus> /
<sqlplus> EXIT
```