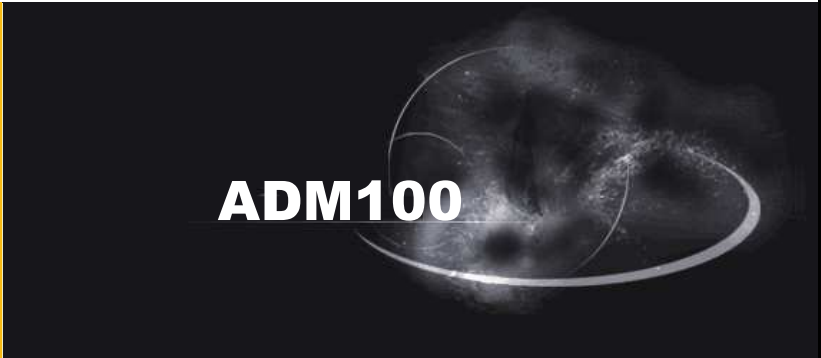


ADM100 mySAP Technology Administration



ADM100

- SAP Web Application Server
- 2002/Q2
- Material number: 50051802

Copyright 2002 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

© SAP AG 2002

Notes on Trademarks:

- Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.
- Microsoft®, WINDOWS®, NT®, EXCEL®, Word®, PowerPoint®, and SQL Server® are registered trademarks of Microsoft Corporation.
- IBM®, DB2®, OS/2®, DB2/6000®, Parallel Sysplex®, MVS/ESA®, RS/6000®, AIX®, S/390®, AS/400®, OS/390®, and OS/400® are registered trademarks of IBM Corporation.
- ORACLE® is a registered trademark of ORACLE Corporation.
- INFORMIX®-OnLine for SAP and INFORMIX® Dynamic Server™ are registered trademarks of Informix Software Incorporated.
- UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of the Open Group.
- HTML, DHTML, XML, and XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- JAVA® is a registered trademark of Sun Microsystems, Inc.
- JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology developed and implemented by Netscape.
- SAP, SAP Logo, R/2, RIVA, R/3, ABAP, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo, and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.



1. Overview

2. Basics

3. Starting and stopping the system

4. Introduction to system configuration

5. Technology components for Internet connection

6. Access to help

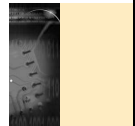
7. Basics of database administration

8. System landscapes and transport requests

© SAP AG 2002

- 9. Support Packages and Industry Solutions**
- 10. Background processing**
- 11. Printing from SAP systems**
- 12. Basics of user administration**
- 13. Remote connections**
- 14. SAP Solution Manager**
- 15. System monitoring**
- 16. Introduction to system security**

© SAP AG 2002



1. Overview

2. Basics

3. Starting and stopping the system

4. Introduction to system configuration

5. Technology components for Internet connection

6. Access to Help

7. Basics of database administration

8. System landscapes and transport requests

© SAP AG 2002

Contents:



- Definitions of the terms system and instance
- Process of a system logon
- Setting up SAP Logon
- Using logon groups
- Common administration transactions

Objectives:

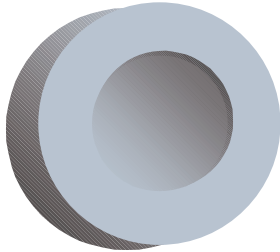
At the conclusion of this unit, you will be able to:

- Use the terms instance and system correctly
- Describe the process of a system logon
- Set up the SAP Logon program
- Explain the use of logon groups
- List central administration functions in the system

© SAP AG 2002

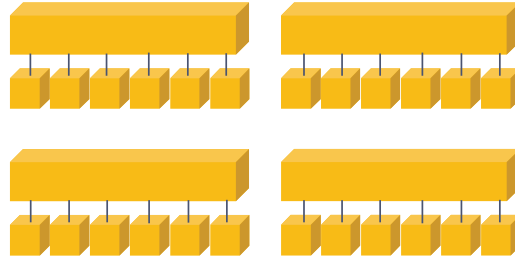
An SAP system always consists of:

exactly one database



with a three-character ID.

one or more instances



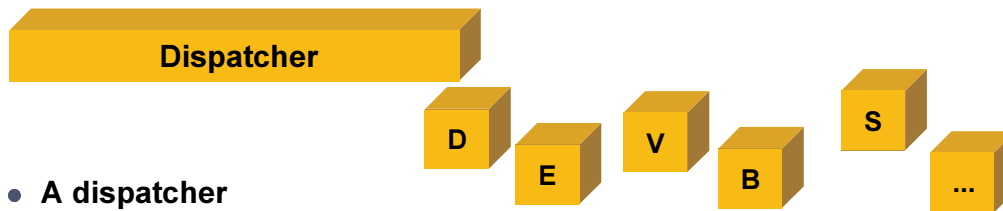
that could each be on its own server, or could share hardware.

© SAP AG 2002

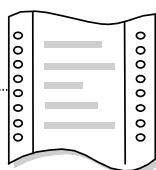
- An SAP system consists of a database and one or more instances. An instance that combines with a database to create a runnable SAP system is called the **central instance**. A central instance should be configured in every SAP system. A **central system** exists if the system contains only a single instance, and this is running together with its database on one server.
- You can install two instances of a system or even of different systems on one server. When configuring two systems (or their databases) on one server, you should ensure that the two systems are clearly differentiated. If not, a conflict situation will eventually occur during the use of common resources (such as program libraries), especially if one system is to be upgraded while the other is used with the same release.
- No SAP System IDs (SIDs) should be assigned more than once within a company. Performing a meaningful data exchange with two systems with the same SID is very difficult (renaming the systems). The three-character SID of the SAP system is always identical to the ID of the database used.

An instance of an SAP system is

made up of:



- A dispatcher
- The work processes that belong to this dispatcher
- Possibly an Internet Communication Manager
- The assigned main memory areas (buffer, and so on)



configured using an
instance profile

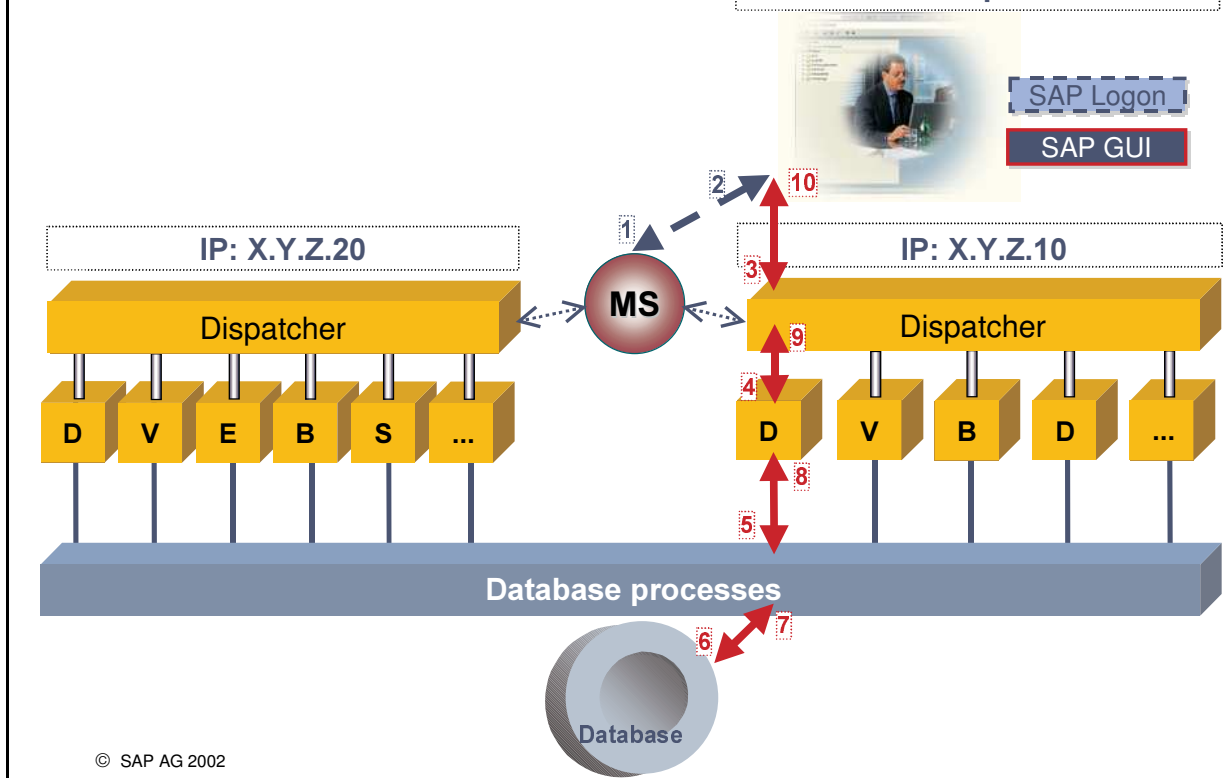


not the same as the
instance of a database system

© SAP AG 2002

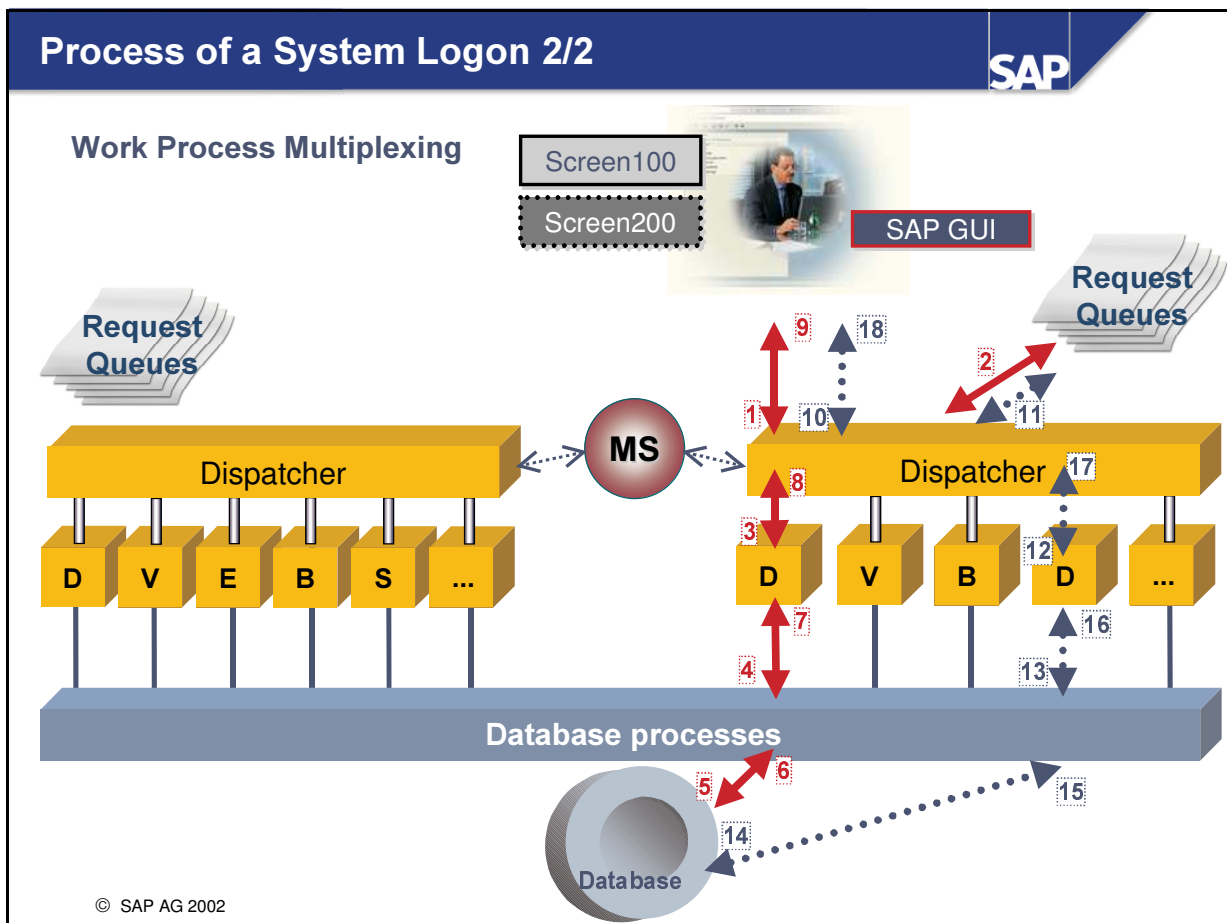
- An SAP system instance is an administrative unit in which the components of an SAP system that provide one or more services are combined. The services provided are started and stopped together. All components of an instance are provided with parameters using a common instance profile.
- These components are currently the dispatcher, the work processes, the assigned main memory areas, and an Internet Communication Manager (ICM), if one is configured.
- An instance always has exactly one dispatcher. The start of an instance always begins with the start of the associated dispatcher. A dispatcher requires a minimum of two dialog work processes. Otherwise it is not possible to start it. Multiple dispatchers can be configured on one host, but must have different system (instance) numbers. The default for the instance number of a dispatcher is 00. Port 3200 receives communication for this dispatcher. If there are two instances on one machine, the port numbers 3200 and 3201 are usually assigned, and so on.
- An instance is also called the **application server** in the software-oriented view of the client-server model. The application server provides the runtime environment for the business applications of the mySAP.com e-business platform. The mySAP Technology application server is called the **SAP Web Application Server** and is the Internet-oriented development of the previously used application server.

For example: IP: X.Y.Z.10



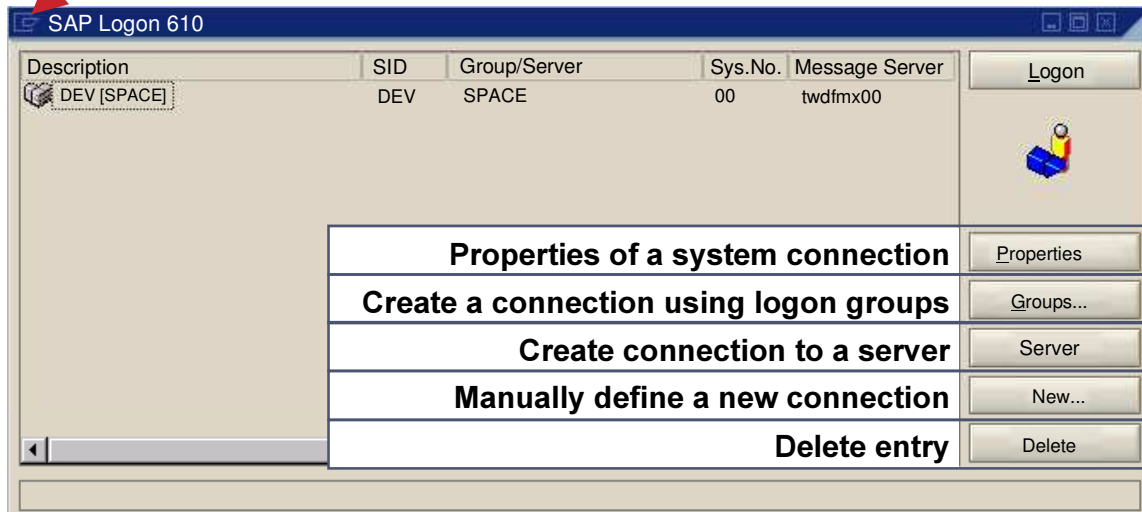
© SAP AG 2002

- To be able to create a connection between the frontend and an instance of an SAP system, the *sapgui.exe* program requires various information in the form of start parameters. This parameter string is normally created by the *saplogon.exe* program using information about the system selected for logon. This information comes partly from the configuration files of SAP Logon, and partly from a direct request to the message server of the selected system (1 and 2). SAP Logon then starts the SAP GUI with these specifications. In this way, an instance that will provide an appropriate level of performance is selected for the logon.
- After the transfer of the logon screen from the dispatcher to the frontend (not shown in the graphic), SAP GUI sends the user's logon data to the instance (3). After the dispatcher has determined a free work process to process the logon, it transfers the logon data to this work process (4). The work process checks whether the received combination of user ID and password is known to the system using a request to the database (5-8). A positive response from the database prompts the work process to return the initial screen of the system to the frontend.
- The assignment of the user to the instance is unique during a logon session. The message server can assign a user to a different instance only during a new logon.



- The processing of a transaction that consists of multiple screens is usually distributed over multiple, different dialog work processes. This distribution is called work process multiplexing. Work process multiplexing means that various dialog work processes can process a system function whose content is logically connected but which consists of multiple substeps. These steps whose content is connected are described as transactions. Multiple dialog work processes can also process a transaction that consists of multiple screens, such as screens 100 and 200.
- The graphic shows two screens of a transaction (100 and 200), for which the input is handled by two different dialog work processes. The multiplexing procedure is used exclusively for dialog work processes. All other work process types process entire functions (complete business processes).
- As dialog work processes may therefore process only parts of transactions that are connected from a business point of view; the asynchronous update procedure is widely used in SAP systems (see also the SAPTEC course).

Additional functions through Options



© SAP AG 2002

- The SAP Logon program provides users with a convenient logon to an SAP system with the SAP GUI for Windows (from Windows-based frontends). SAP Logon evaluates various configuration files that are stored on the frontend. These files can also be edited using SAP Logon.
- You can make various general settings through the SAP Logon *Options*. For example, you can set the trace level for SAP GUI connections. Passwords can also be written to the trace file generated in this way, so you should use this option only in exceptional situations. The trace files should be deleted afterwards.
- The settings for a connection can be changed using *Properties*. For example, using its name in SAP Logon, and through *Advanced* on the properties dialog box, you can restrict the amount of data transferred (system to frontend), to take account of slow network connections (see also **SAP Note 161053**).
- Using *Groups...* (for group selection), you can create a new connection to a system. This connection uses the logon groups concept, meaning that the message server always assigns a new logon to the instance with the best performance in the selected logon group. The configuration file *sapmsg.ini* is evaluated to display the systems available for selection. The message server of the selected system is queried to discover the available logon groups. So that the connection to the message server of the system specified in *sapmsg.ini* works, it is necessary for the Windows file *services* to specify the correct message server communication port for the selected system ID. A connection is then created to the relevant host and the message server running on it using the information from *sapmsg.ini* and *services*.



← Start the system logon

Using SAP Logon and associated files:

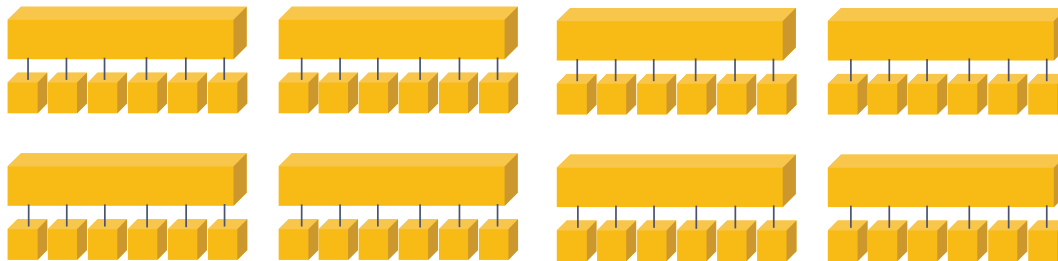
- | | |
|---------------------------------|---|
| • Start of SAP Logon | Read <i>saplogon.ini</i> |
| • Properties Pushbutton: | Edit <i>saplogon.ini</i> |
| • Groups Pushbutton: | Evaluation of <i>sapmsg.ini</i> and <i>services</i> |
| • Server Pushbutton: | Edit <i>saplogon.ini</i> |
| • New Pushbutton: | Edit <i>saplogon.ini</i> |
| • Delete Pushbutton: | Edit <i>saplogon.ini</i> |

All actions (except *Logon*) can change *saplogon.ini*

© SAP AG 2002

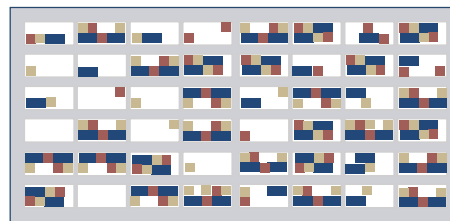
- *Server* allows you to create a connection to a specific instance of an SAP system. As with *Groups*, information about the instances that are available for selection is collected from the message server.
- With *New*, you can create a connection to an SAP system that need not be listed in the *sapmsg.ini* and *services* files. Therefore, you enter all of the system information that is relevant for a logon. The name of the host on which you want to contact an instance, and its system or instance number is essential. The instance number specifies the last two digits of the four digit port address through which the dispatcher of the specified instance can be contacted. The first two digits are fixed as 32. This means that port numbers between 3200 and 3299 are possible. The ports 3298 and 3299 are, however, assigned to the niping and saprouter programs by default (although this can be changed). For more information about this, see the SAP Library for the SAP Web AS with the search term 3298.
- You can also specify a SAProuter string for new connections to be created. Through this connection, a SAProuter is then assigned the task of transferring data. SAProuter strings can also be specified using aliases. These aliases are decoded using the *saprout.ini* file.
- The following list shows the files with their possible storage locations. In the case of multiple possible locations, the search sequence is shown:
 - *saplogon.ini*, *sapmsg.ini*, *saprout.ini*: SAP GUI directory, Windows directory
 - *services*: WINNT system32 → drivers → etc → Services

A system may consist of many instances:



Each of these instances has its own buffer areas which, for example, must hold all of the programs in the instance.

Without the use of logon groups, a typical program buffer for each of the eight instances shown here looks like this:



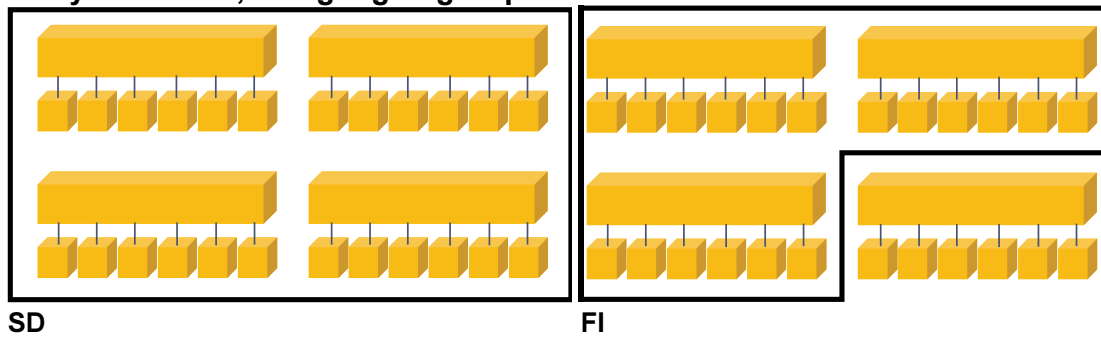
For example: Programs from the following areas:

- SD
- FI
- Programs used in common

© SAP AG 2002

- SAP systems sometimes have significantly more than just one or two assigned instances. Each of these instances provides buffer areas for various objects, such as for programs, dictionary objects, screen structures, and table contents.
- These buffers are created and continuously optimized during the usage time. The system attempts, using various algorithms, to organize the contents of the buffer so that the frequently queried data is in the buffer.
- A cross-section of all programs called on this instance is collected, especially in the program buffer. If a large number of different applications are run, old buffer contents may be replaced with newly requested contents. The next time the older contents are called, they must be requested from the database again. This reduces the response time behavior of the system, in some circumstances to an appreciable extent. A solution would be to create a program buffer that was sufficiently large to contain all requested programs. An alternative solution is the logon group concept explained on the next page.
- Using logon groups, you can group instances with transaction *SMLG*. In the example above, four instances are assigned to the logon group SD, and three instances to the logon group FI. No more than five minutes after the new group definitions are saved, the message server of the system also has these details. You can now access these groups using *Group selection* in SAP Logon.

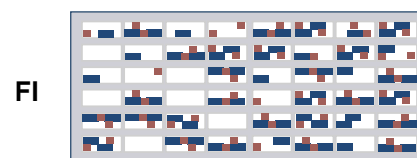
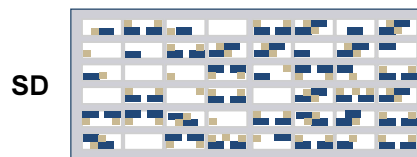
Many instances, using logon groups defined in SMLG:



Using logon groups, the program buffers of two instances from different logon groups could look like this:

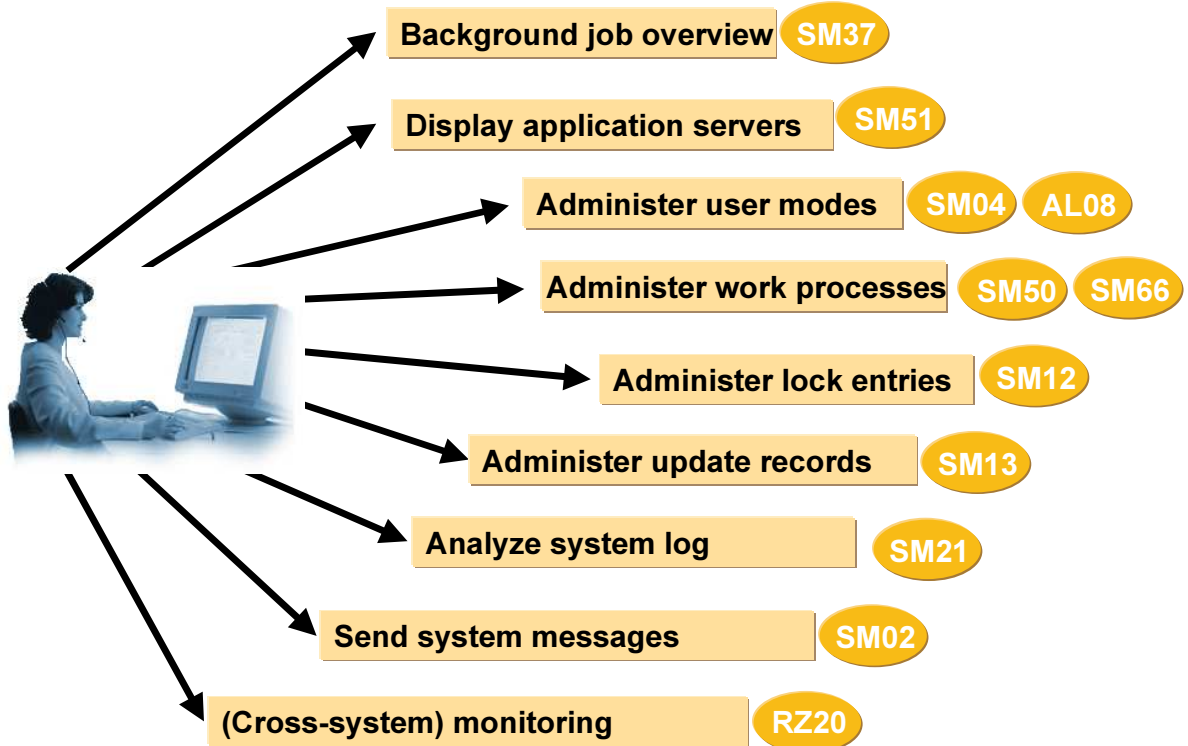
Programs from the following areas:

- SD
- FI
- Programs used in common



© SAP AG 2002

- In the example shown, the definition of the two logon groups has the following effects:
 - With a logon using a logon group, the SAP Logon program connects to the message server of the system, and receives from it the name of the currently best performing instance of the selected logon group. SAP Logon then starts the SAP GUI program with the connection parameters for the selected dispatcher.
 - Within a logon group, end users are, for example, only using a particular application. In the example, these are the FI and SD applications. Therefore, the buffers on the instances of each of the logon groups are filled with contents from these applications. This means that the program buffer of each of the instances shown contains **fewer different** programs than they would without the use of logon groups. This has two possible advantages:
 1. The program buffers can be set up more economically (smaller).
 2. Less swapping is performed for the program buffers, and they therefore perform better.
- If it is possible to set up very large program buffers (>1GB), setting up logon groups for the purpose described above would no longer provide any relevant advantages. This means that a single logon group (with the possible exception of the central instance) is the most useful. For example, this has the name *Public*. By default, every instance is assigned to the logon group SPACE. For more information, see the SAP Notes database, with the search term *SMLG*.

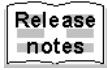


© SAP AG 2002

- The transactions listed in the graphic help you with the daily system administration work. They have already been listed in the SAPTEC course, from which this graphic is taken. You should already be familiar with the usage and interpretation of most of these transactions. The functions *RZ20* and *SM37* are dealt with in more detail in this course. This course also introduces other administration functions.



- **SAPTEC - mySAP Technology Fundamentals**

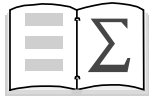


- **SAP Library**

- **SAP Notes**

21960: Several instances/systems on one UNIX computer
(and SAP Notes linked from this note)

39412: How many work processes to configure



You are now able to:

- **Use the terms instance and system correctly**
- **Describe the process of a system logon**
- **Set up the SAP Logon program**
- **Explain the use of logon groups**
- **List a few central administration functions**

© SAP AG 2002

Exercises

In these exercises, you set up your local SAP Logon appropriately for your training system, learn about setting up logon groups, and optionally, repeat some administrative transactions. It may be that you are not able to perform part 1 on your training frontend as described below.

No.	Exercises
	Part 1: Setting Up the Local SAP Logon Optional (Depends on the setup of your training frontend)
1	Clear existing SAP Logon entries
1.1	As there could already be entries in your local SAP Logon from other training courses, if it exists, use the call: <i>Start → Training → SAPLOGONreset.</i> This call edits your local <i>SAPLogon.ini</i> file so that only a dummy entry still remains.
2	Set up SAP Logon using New
2.1	First copy the SAP Router string from the dummy system to the buffer using copy and paste.
2.2	Choose <i>New</i> and create a new entry using the copied SAP Router string. The description of the new connection should include the system ID and the host name of your system. For example: QAS_twdf0500 . Specify your host, such as twdf0500 , as the application server. The system number could be 00 or 01 for a DEV system and 10 or 11 for a QAS system.
	Part 2: Setting Up and Using Logon Groups
3	Setting up logon groups
3.1	Log on to your system using your new entry in SAP Logon (with the parameters that your instructor gives you). Call transaction SMLG .
3.2	Create a new logon group with a name of your choice by choosing the “white page” (such as QAS_## , where ## is your group number). Assign at least one of the two instances of your system to your new logon group. Copy and save your entries.
4	Using logon groups through <i>Group Selection</i> in SAP Logon
4.1	Choose <i>Groups...</i> in your SAP Logon. On the following window, specify the ID of your system, such as QAS , under <i>System ID</i> . Note that an entry can already exist in the <i>sapmsg.ini</i> file for a system (here, for example, the system <i>DEV</i>). However, this entry does not point to the training system that you are using. You should therefore change the entry under <i>Message Server</i> to correspond to the host that you are to use. For example: twdf0500.wdf.sap-ag.de . Choose the correct entry for the SAProuter to be used either using input help, or as specified by your instructor. Then

	choose <i>Generate List</i> . If the logon group that you created in task 3 does not yet appear in the display, the message server has not yet registered the newly created logon group. This is done automatically every five minutes. You can therefore either generate the list again a few minutes later, or use the <i>Space</i> entry that always exists. The logon group <i>Space</i> always contains all currently active instances of a system.
4.2	Log on to your system using the new entry in your SAP Logon.
	Part 3: Repetition: Basic Administrative Transactions
5	Optional: Repetition of some basic transactions
5.1	Determine the number of instances available to you in your system and their names.
5.2	How many work processes does your entire system have? Of which types?
5.3	How many users are currently logged on to the same instance on which you are working?
5.3	Call your <i>Own Data</i> (user defaults). In a new session, check the currently active locks in your system. Delete the locks by closing the session that holds the locks. There are two options for doing this.

Solutions

In these exercises, you set up your local SAP Logon appropriately for your training system, learn about setting up logon groups, and optionally, repeat some administrative transactions. It may be that you are not able to perform part 1 on your training frontend as described below.

No.	Solutions
	Part 1: Setting Up the Local SAP Logon Optional (Depends on the setup of your training frontend)
1	Clear existing SAP Logon entries
1.1	<p>As there could already be entries in your local SAP Logon from other training courses, if it exists, use the call:</p> <p><i>Start → Training → SAPLOGONreset.</i></p> <p>This call edits your local <i>SAPLogon.ini</i> file so that only a dummy entry still remains.</p> <p>Follow the exercise instructions.</p>
2	Set up SAP Logon using the New
2.1	<p>First copy the SAP Router string from the dummy system to your keyboard buffer using copy and paste.</p> <p>Display the <i>Properties</i> of the dummy connection and copy the string into the buffer using <i>Ctrl + c</i>.</p>
2.2	<p>Choose <i>New</i> and create a new entry using the copied SAP Router string. The description of the new connection should include the system ID and the host name of your system. For example: QAS_twdf0500. Specify your host, such as twdf0500, as the application server. The system number could be 00 or 01 for a DEV system and 10 or 11 for a QAS system.</p> <p>Follow the exercise instructions.</p>
	Part 2: Setting Up and Using Logon Groups
3	Setting up logon groups
3.1	<p>Log on to your system using your new entry in SAP Logon (with the parameters that your instructor gives you). Call transaction <i>SMLG</i>.</p> <p>Follow the exercise instructions.</p>
3.2	<p>Create a new logon group with a name of your choice by choosing the “white page” (such as DEV_##, where ## is your group number). Assign at least one of the two instances of your system to your new logon group. Copy and save your entries.</p> <p>Follow the exercise instructions.</p>

4	Using logon groups through <i>Group Selection</i> in SAP Logon
4.1	<p>Choose <i>Groups...</i> in your SAP Logon. On the following window, specify the ID of your system, such as QAS, under <i>System ID</i>. Note that an entry can already exist in the <i>sapmsg.ini</i> file for a system (here, for example, the system DEV). However, this entry does not point to the training system that you are using. You should therefore change the entry under <i>Message Server</i> to correspond to the host that you are to use. For example: twdf0500.wdf.sap-ag.de. Choose the correct entry for the SAProuter to be used either using input help, or as specified by your instructor. Then choose <i>Generate List</i>. If the logon group that you created in task 3 does not yet appear in the display, the message server has not yet registered the newly created logon group. This is done automatically every five minutes. You can therefore either generate the list again a few minutes later, or use the <i>Space</i> entry that always exists. The logon group <i>Space</i> always contains all currently active instances of a system.</p> <p>Follow the exercise instructions.</p>
4.2	Log on to your system using the new entry in your SAP Logon.
	Part 3: Repetition: Basic Administrative Transactions
5	Optional: Repetition of some basic transactions
5.1	<p>Determine the number of instances available to you in your system and their names.</p> <p>To do this, call transaction SM51 (alternatively, choose: <i>Tools</i> → <i>Administration</i> → <i>Monitor</i> → <i>System Monitoring</i> → <i>Servers</i>. Your system should have two instances, the names of which could be, for example, twdf0500_DEV_00 and twdf0500_DEV_01.</p>
5.2	<p>How many work processes does your entire system have? Of which types?</p> <p>Transaction SM66 (Menu path, for example: <i>Tools</i> → <i>Administration</i> → <i>Monitor</i> → <i>Performance</i> → <i>Exceptions/Users</i> → <i>Active Users</i> → <i>All processes</i>) displays that your system has, for example eight dialog work processes, four spool work processes, and so on, if the <i>process selection</i> is set up correctly (<i>waiting</i> processes also displayed).</p> <p>Transaction SM66 shows all work processes of a system, unlike transaction SM50.</p>
5.3	<p>How many users are currently logged on to the same instance on which you are working?</p> <p>Transaction SM04 (or <i>Tools</i> → <i>Administration</i> → <i>Monitor</i> → <i>System Monitoring</i> → <i>User Overview</i>) displays the number of users that are logged on to the same instance as you.</p>
5.3	<p>Call your Own Data (user defaults). In a new session, check the currently active locks in your system. Delete the locks by closing the session that holds the locks. There are two options for doing this.</p> <p>Choose <i>System</i> → <i>User Profile</i> → <i>Own Data</i>. You can set user defaults in a new session that opens automatically. Use the call /osm12 to display an overview of currently active locks in your system. Do not restrict the selection</p>

	of locks. There should be two active locks that are both connected to changing your user defaults. You can either end the session that holds the locks using transaction <i>SM04</i> (Select the user and choose <i>Sessions</i>) or obtain an overview of your sessions by entering <i>/o</i> , from which you can identify and end the session that holds the locks.
--	---

1. Overview

2. Basics



3. Starting and stopping the system

4. Introduction to system configuration

5. Technology components for Internet connection

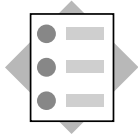
6. Access to Help

7. Basics of database administration

8. System landscapes and transport requests

© SAP AG 2002

Contents:



- Starting and stopping the SAP system
- Appendix: Starting and stopping on supported operating systems

Objectives:

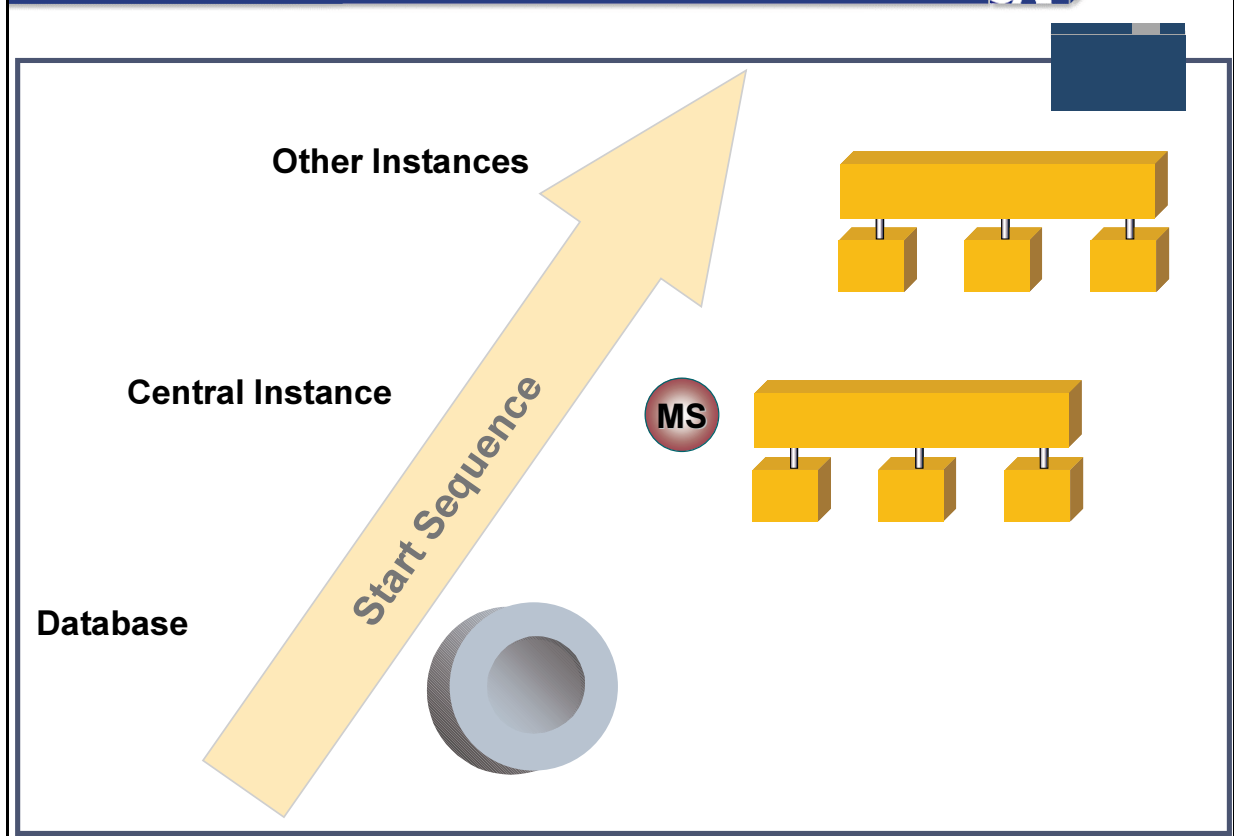
At the conclusion of this unit, you will be able to:

- Start and stop the entire SAP system or individual instances
- Evaluate start logs to analyze problems

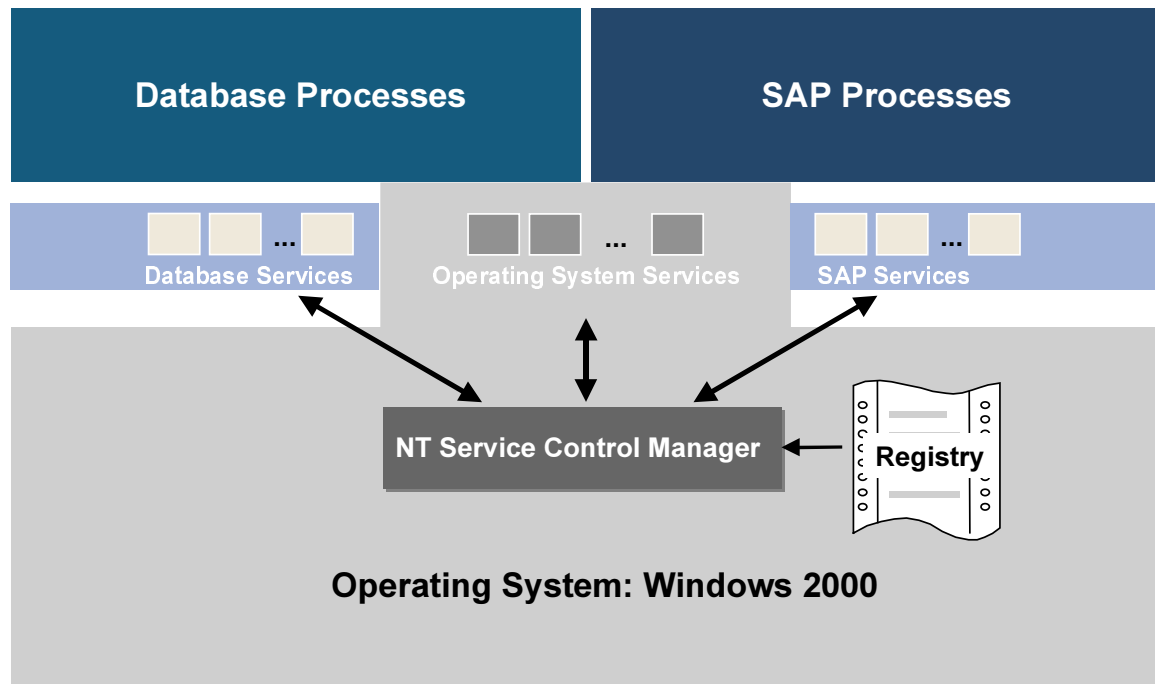
© SAP AG 2002

Start Process of the SAP System

SAP



- Starting an SAP System is performed in a number of steps and is the task of the operating system user <sid>adm.
- Start the database:
 - The underlying element of the entire SAP system is the database. Before the SAP instances are started, this must have operational status. The database is therefore always started as the first step.
- Start the central instance:
 - Next, the operating system collector **SAPOSCOL** is started, if it is not already active. This is a stand-alone program that runs in the operating system background, independently of SAP instances. It collects data about operating system resources and makes this data available through the shared memory of all SAP instances.
 - The central instance with the message server and the dispatcher and its work processes is then started. Only once the message and enqueue servers are active can other instances be started, if desired.
- Start other instances:
 - If the dialog instance is not running on the same host as the central instance, the **SAPOSCOL** operating system collector is first started on this host.
 - The dispatcher is then started with its work processes.

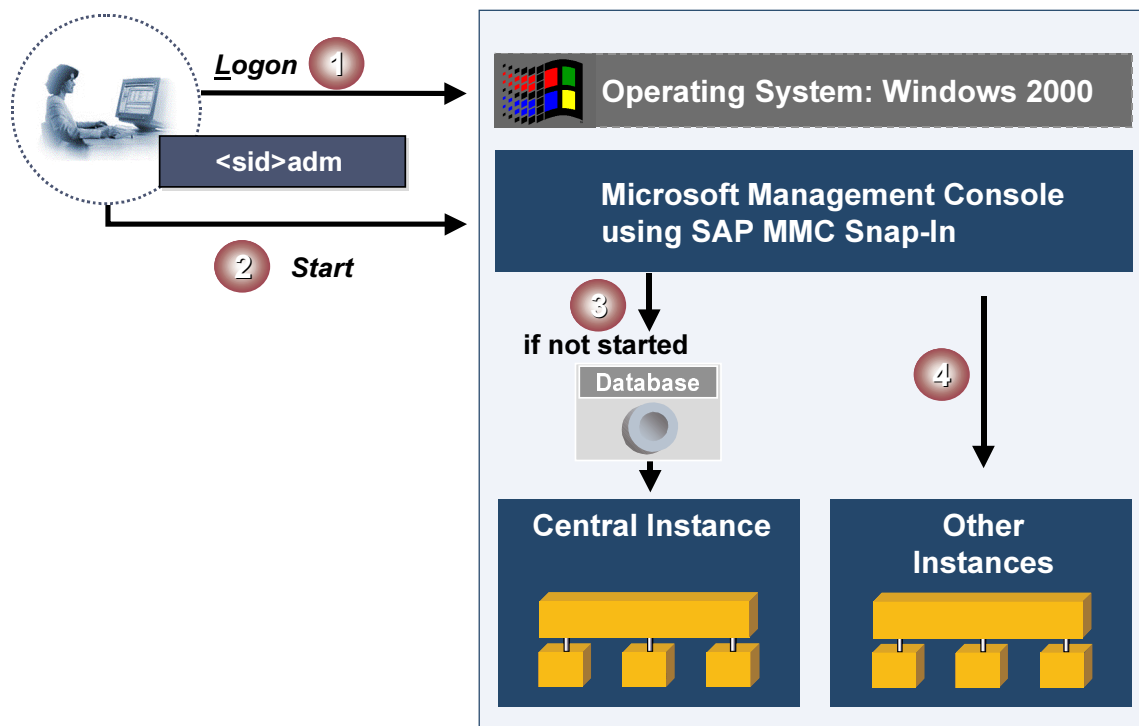


© SAP AG 2002

- When starting programs in the Microsoft Windows environment, you should note that these programs are only active as long as the user is logged on to the system. When a user logs off, all of his or her programs are ended. The SAP system therefore uses the concept of services. These are programs that are automatically started and administered by the operating system. Services provide support to other programs and run even if there are no users logged on to the host.
- The Service Control Manager starts the services installed in the registry during the startup of Microsoft Windows 2000. All services can be configured for automatic startup.
- During the installation of the SAP system, SAP and database services are installed in addition to the operating system services.
- SAP Services:
 - SAPOSCOL: Collects performance data for one or more SAP instances and runs once for each host.
 - SAP<SID>_<instance no.>: Controls the SAP instances and runs once for each instance.
- Database Services:
 - Create the connection to the database.
 - Control DB actions

Starting the SAP System

SAP

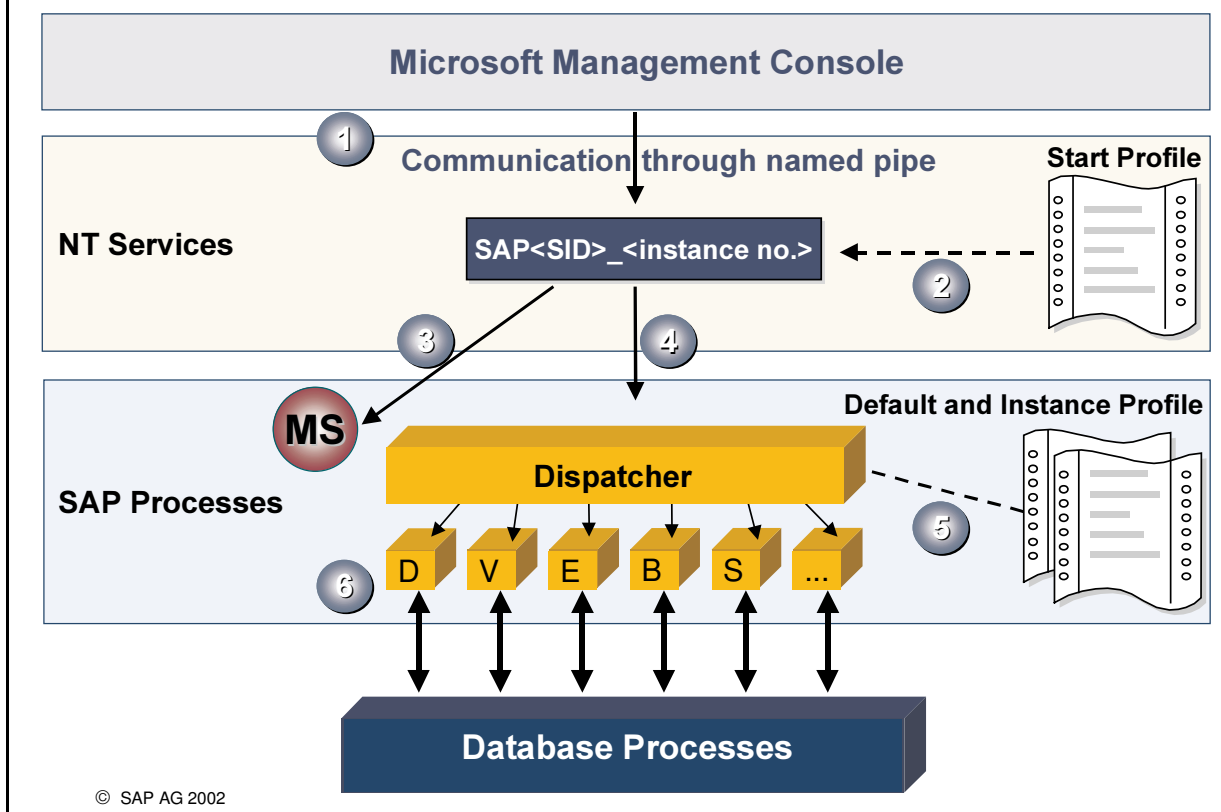


© SAP AG 2002

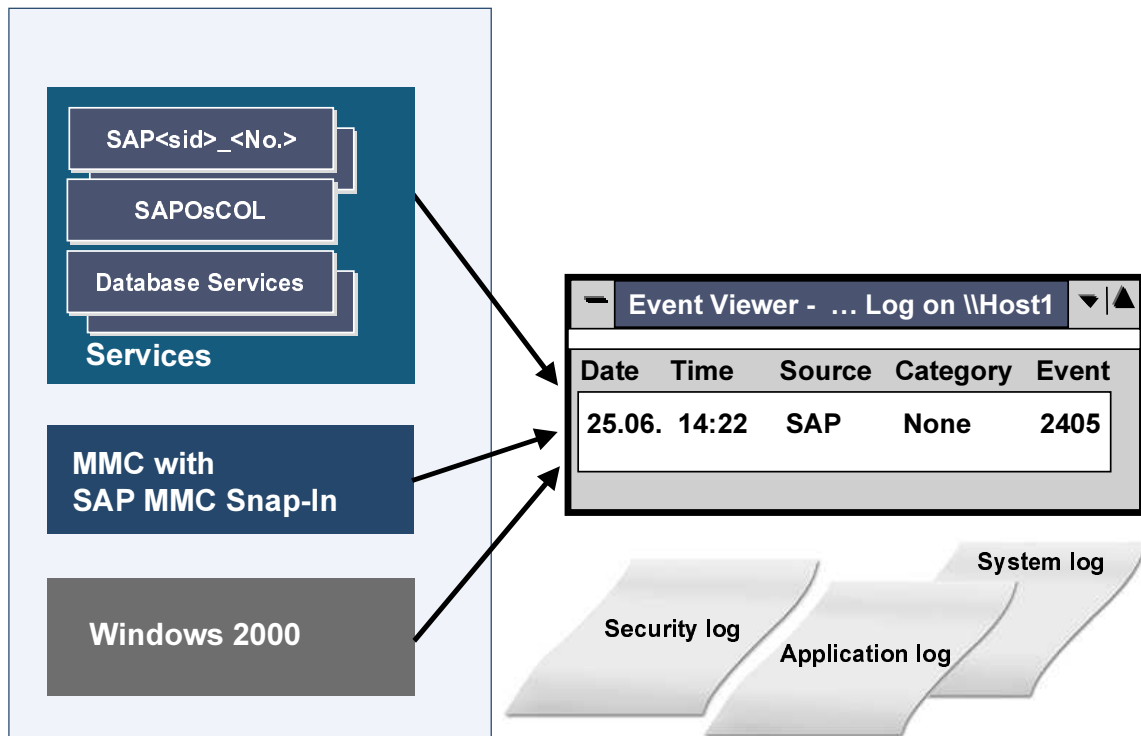
- Under Windows 2000, you can start and stop the SAP system with the Microsoft Management Console (MMC).
- To do this, the administrator logs on to the operating system as user **<sid>adm**, and opens the Microsoft Management Console.
- Choose the node for the central instance in the tree. Call up the context menu with the right mouse button and choose the *Start* function. The system first checks whether the database is active. If not, it is automatically started. If the database is active, the central instance (message server and dispatcher) is started by SAP Service SAP<SID>_<Instance no.>. The communication between the Microsoft Management Console and the SAP Service takes place through a named pipe.
- Other instances can then be started
- The status of SAP systems, individual instances, and the message server and dispatcher are displayed in the Microsoft Management Console in accordance with the following color legend:
 - gray: is not running
 - yellow: is starting
 - green: active
 - red: terminated after errors

Start Sequence of Processes of the SAP System

SAP



- The specifications of the instances: the type and number of processes, main memory sizes, and other options are controlled at the start of the instances using profiles. These are files at operating system level that are stored in the directory \\<SAPGLOBALHOST>\sapmnt\<SID>\profile.
- When the instances are started, the SAP service reads which processes (message server, dispatcher) are to be started from the instance-specific start profile. The start profile can be displayed in the Microsoft Management Console by choosing the right mouse button on the entry for the instance and selecting the function *All tasks* → *View Start Profile*.
- The specification of the configuration of the instances is stored in the default profile and in the instance profile. These profiles are read by the dispatcher, which starts the work processes and creates the instance-specific configuration.
- After the instance has been successfully started, all work processes connect to the database.

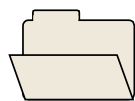


© SAP AG 2002

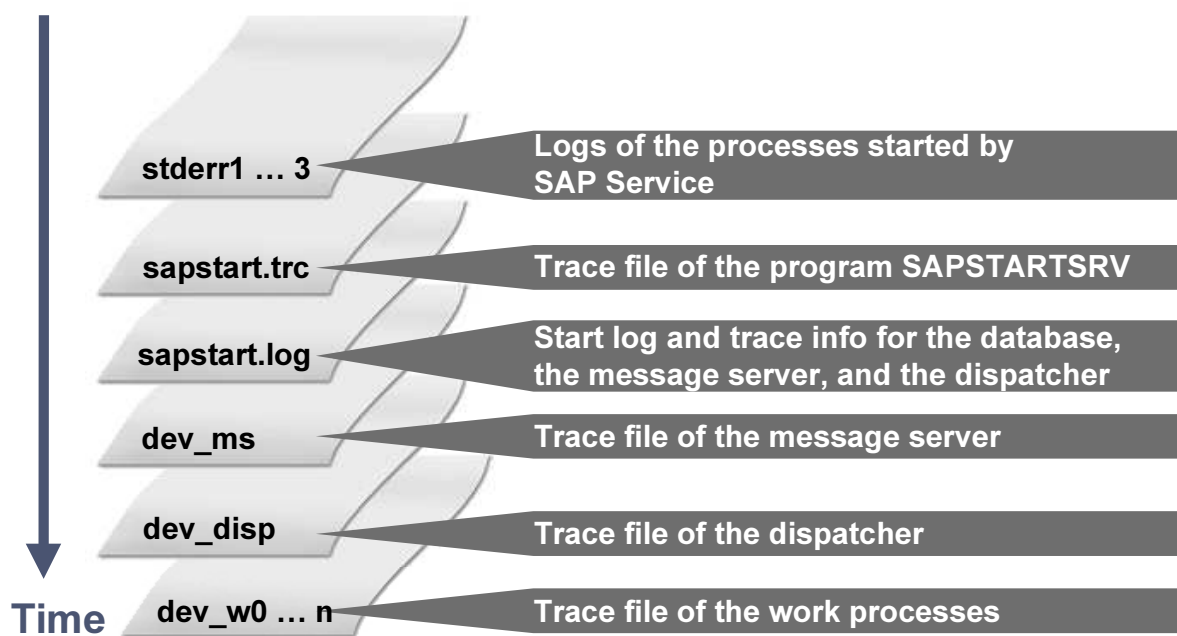
- All messages that are created by services or the SAP Service Manager are recorded at operating system level by an event logging service, the Event Manager. This Event Manager writes an event log which contains the following components:
 - System Log:
Operating system and application messages
 - Application Log:
List of errors, warnings, and information that is generated by application software.
 - Security Log:
Events such as log ons and log offs and user access to files.
- You can call the event log by choosing *Start → Programs → Administrative Tools → Event Viewer*. Choose the relevant component from the menu bar. The system displays a list of errors, warnings, and information that have been generated. For detailed information, double click a particular log.

Logging of the Start Process in the SAP System

SAP

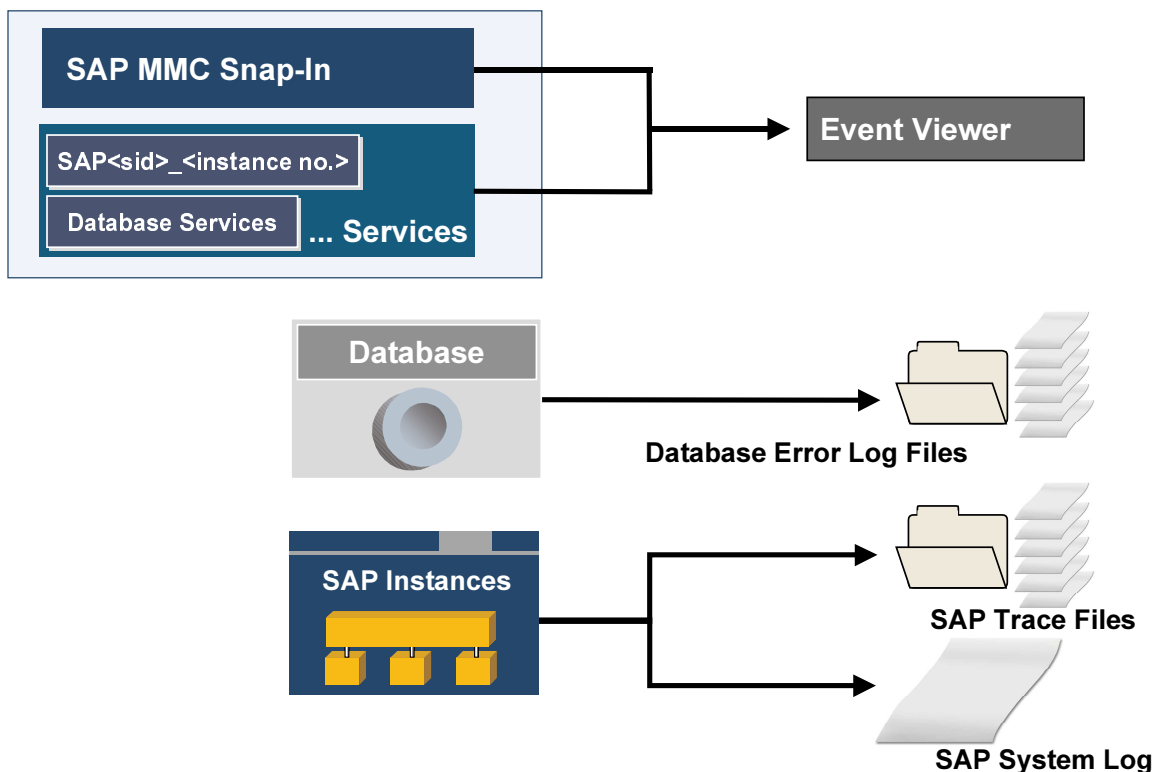


\\<SAPLOCALHOST>\saploc\<SID>\<Instance>\<No>\work\ ...



© SAP AG 2002

- Logs about the start process of the SAP system are stored in the file system. If there are problems during the start, these logs can provide useful information such as error messages or problem descriptions. These files are stored in the home directory of user <sid>adm.
- The following logs are created during the start process of the SAP service:
 - STDERR1: Information about the start process of the database system.
 - STDERR2: Information about the start process of the message server.
 - STDERR3: Information about the start process of the dispatcher.
- The granularity of the logged information can be set in 4 levels using the profile parameter *rdisp/TRACE*. The possible values for this parameter are:
 - 0: Errors only
 - 1: Error messages and warnings (Default)
 - 2: Error messages and a short trace
 - 3: Error messages and a complete trace
- The higher the trace level, the larger the amount of logged information, and therefore the larger the size of the files. You should therefore only increase the default value for short periods for problem analysis.
- The trace level can be set separately for individual work processes in the process overview (transaction *SM50*).



© SAP AG 2002

- If the SAP system does not start correctly, this can have a variety of causes. To analyze the problem, proceed as follows:
 - Check the error messages and warnings in the Event Viewer. Check that the service SAP<SID>_<Instance Number> (such as SAPDEV_00) was started. To do this, go to the *Services* window (*Start* → *Programs* → *Administrative Tools* → *Services*).
 - Check the status of the respective database system using the error log files. For more information, see the *Database Logs* appendix of this unit.
 - Check the start log in the MMC. To do this, select the affected instance and choose *All Tasks* → *View Developer Traces* in the context menu.
 - Check the error files stderr<n> that were created by SAP Service.
 - Check the trace files of the individual SAP work processes:
 - dev_ms : Developer trace for the message server
 - dev_disp : Developer trace for the dispatcher
 - dev_w<m> (m is the work process number): Developer trace for the work process
 - If you can still log on to the SAP system, check the system log of the SAP system using transaction *SM21*.

➔ Check the status of

- **Logged-on users *SM04***
- **Background processing *SM37* and Batch Input *SM35*:**
 - Are jobs active or planned?
 - Are jobs triggered by external systems?
- **Update: *SM13***
- **External connections**

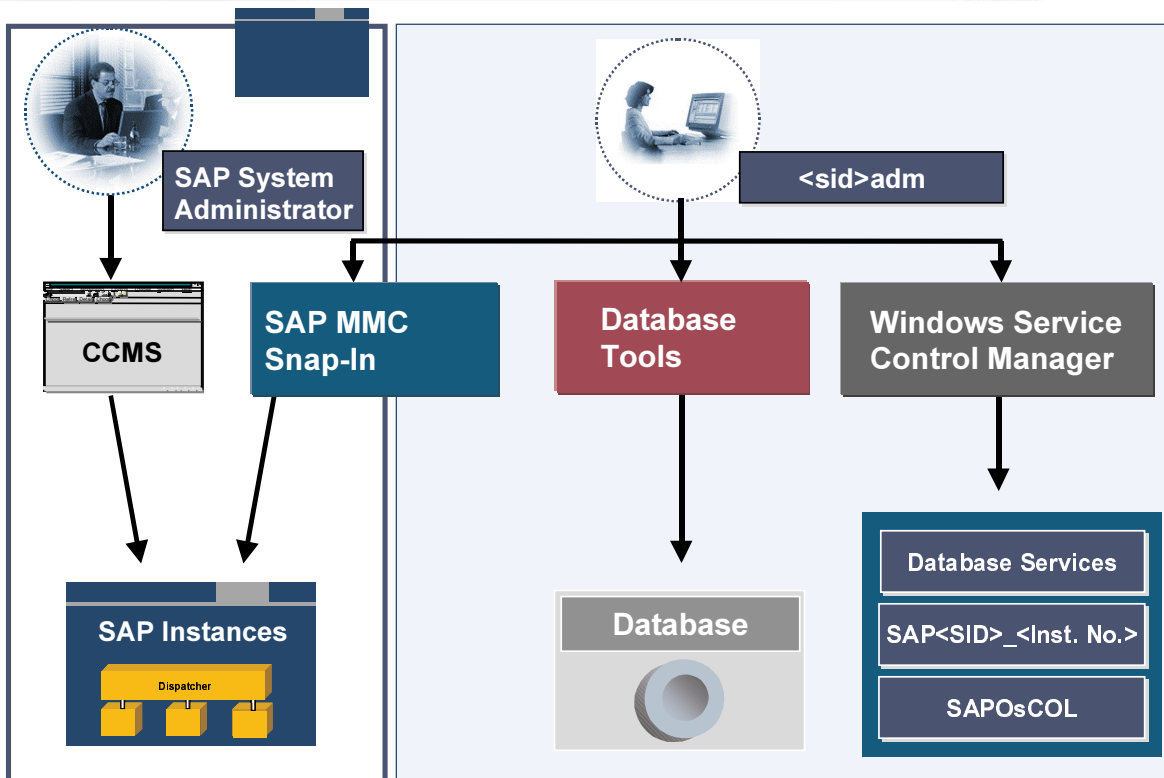
➔ Send a system message *SM02*

© SAP AG 2002

- Before you stop the system, you should check the status of the system. This involves, among other things:
 - **Active Users:**
Check which users are logged on using the User List (*SM04*).
 - **Background Processing:**
Check which jobs are active using the Job Overview (*SM37*). If jobs are terminated by the system stop, these must be rescheduled. Jobs that are scheduled for the time in which the system is stopped run automatically once the system is restarted.
 - **Batch Input:**
Batch Input: Session Overview (*SM35*) displays running batch input jobs.
 - **Update:**
Use the Update Overview (*SM13*) to check whether update processes are terminated by the system stop. These update records are rolled back during the stop, and these are set to the status *init*. These records are then processed again during the restart.
- Before you stop your system, you should inform users using a system message (*SM02*).

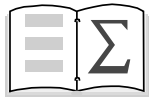
Stopping the SAP System

SAP



© SAP AG 2002

- The stopping of the SAP system happens in the opposite order of the start.
- There are two ways to stop instances:
 - In the SAP system itself in the CCMS (transaction *RZ03*) by choosing *Control* → *Stop SAP instance*.
 - In the Microsoft Management Console, click the right mouse button to show the context menu and choose the *Stop* function. Depending on whether you have selected an individual instance or the entire SAP system, the following are stopped:
 - A single instance
 - Central instance and all dialog instances
- The SAP service waits for a stop message from the MMC or from the CCMS and then stops the SAP system. The service itself is not stopped.
- The services themselves can be stopped and restarted with the Windows Service Control Manager.
- The database is stopped using the corresponding database system tool.



You are now able to:

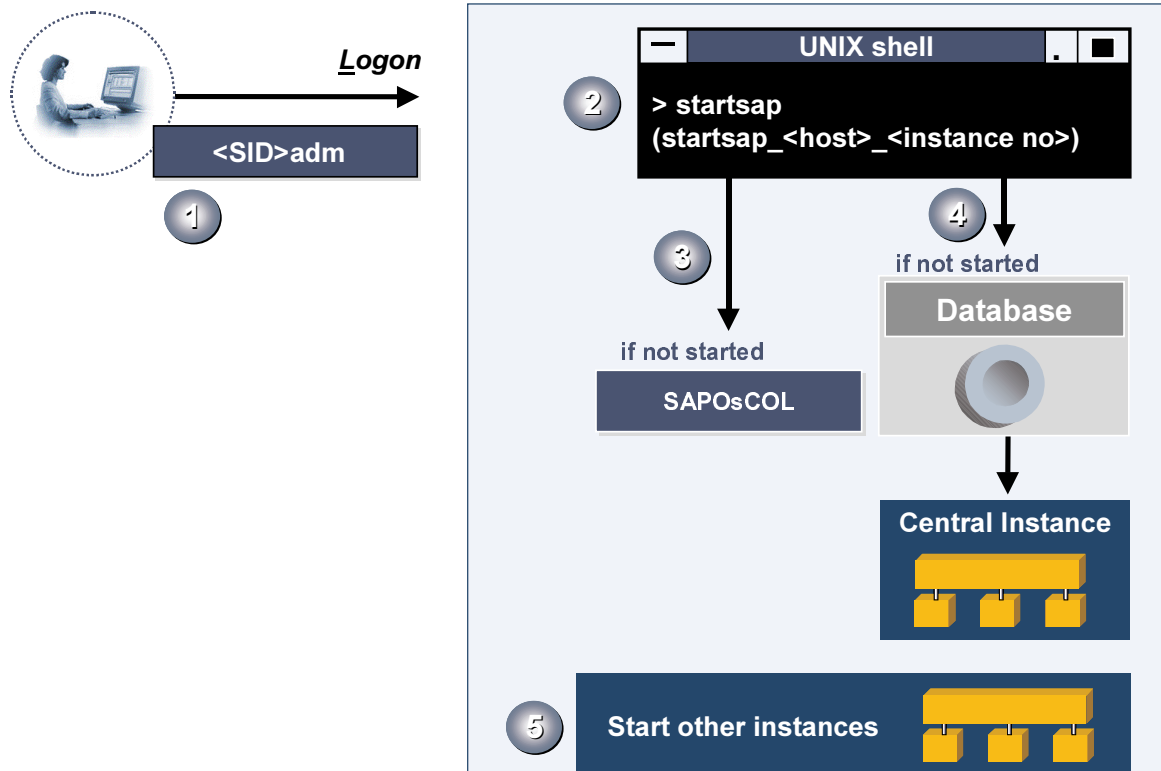
- **Describe the start sequence of the SAP System**
- **Start and stop the entire SAP System or individual instances**
- **Evaluate start logs to analyze problems**

© SAP AG 2002



Starting and Stopping on Supported Operating Systems

© SAP AG 2002

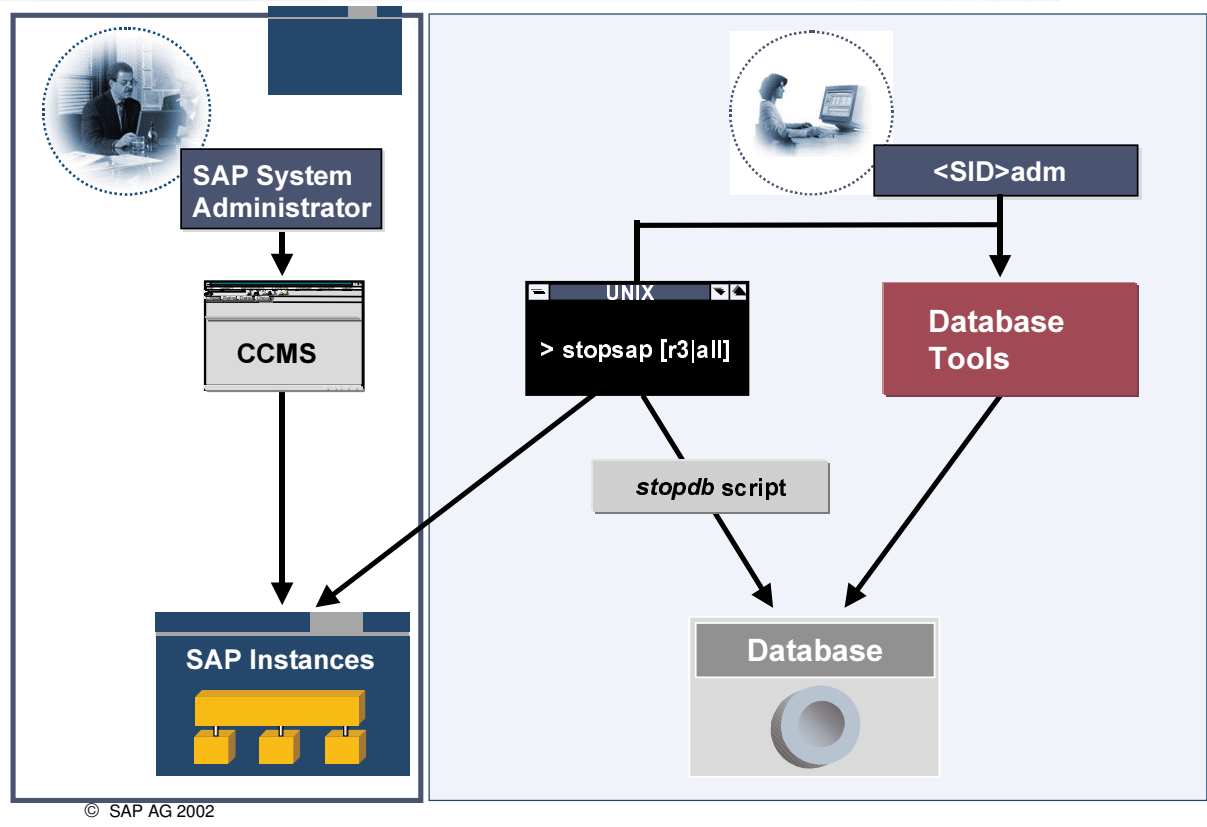


© SAP AG 2002

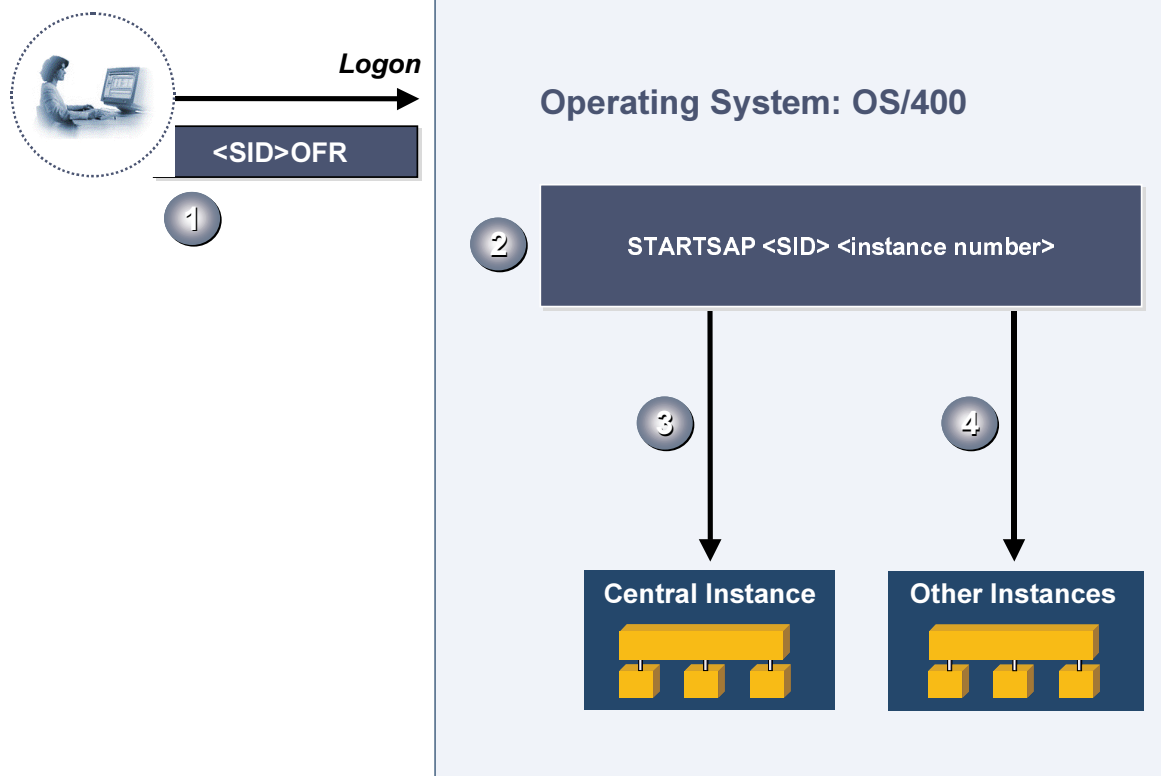
- Log on under UNIX as a user with SAP administrator authorization (`<SAP System Name>adm`).
- To start the central instance call the shell script `startsap_<host>_<instance no>` in the home directory of the `<sid>adm` user. If there is only one instance for each application server, the alias `startup` points to the script `startsap_<host>_<instance no>`.
- The script `startsap` first starts the `saposc` process that collects statistical data for the operating system that it provides to the SAP system.
- The next step is a check to see if the database is running. If not, the script `startdb` is called to start the database.
- The final step starts all of the processes for the instance (message server, dispatcher, work processes, and so on).
- After the central instance is started, you can optionally start other instances.
- The `startsap` script can be called with the following options:
 - DB: Starts the database system.
 - R3: Starts the instances and associated processes for the instance.
 - ALL: Starts the database system and the instance (default setting, can be omitted).
- To start the SAP system, the `startsap` script calls the `sapstart` process with the start profile specified in the script in the variable `START_FILES`.

UNIX: Stopping

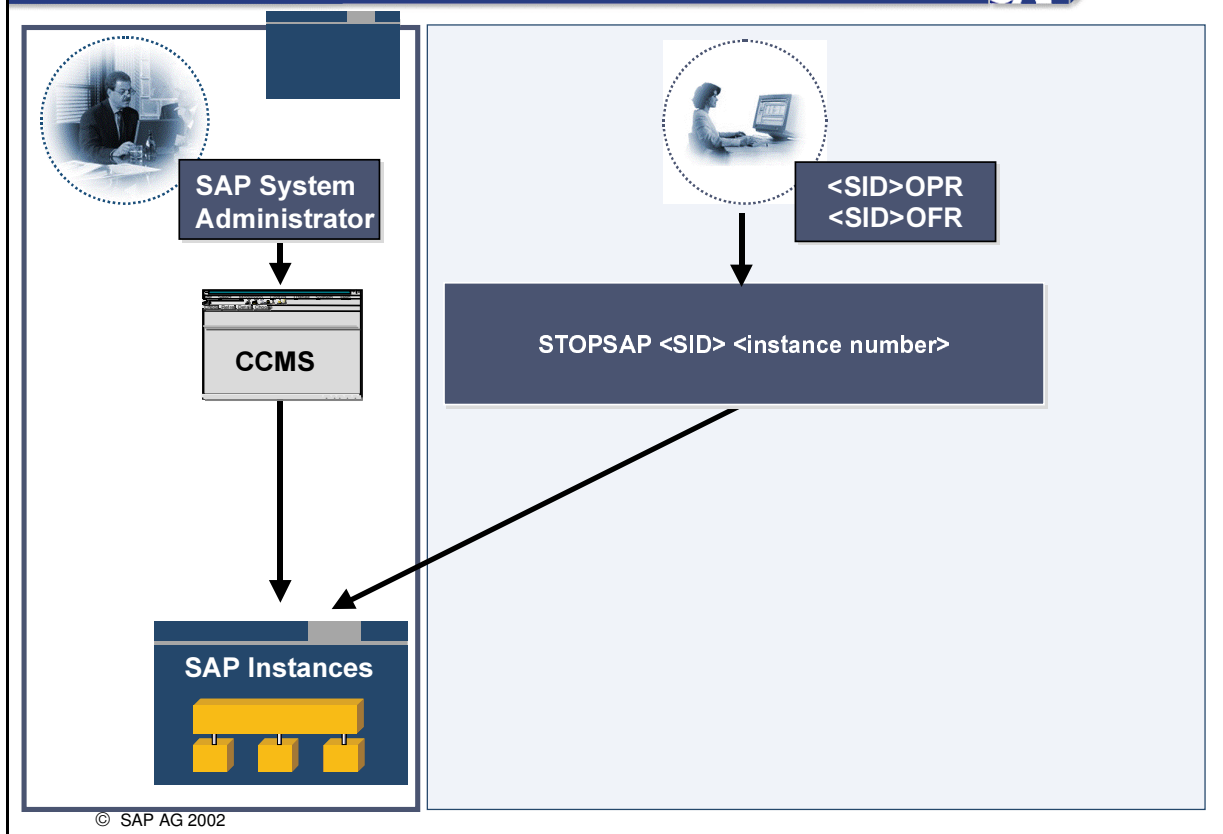
SAP



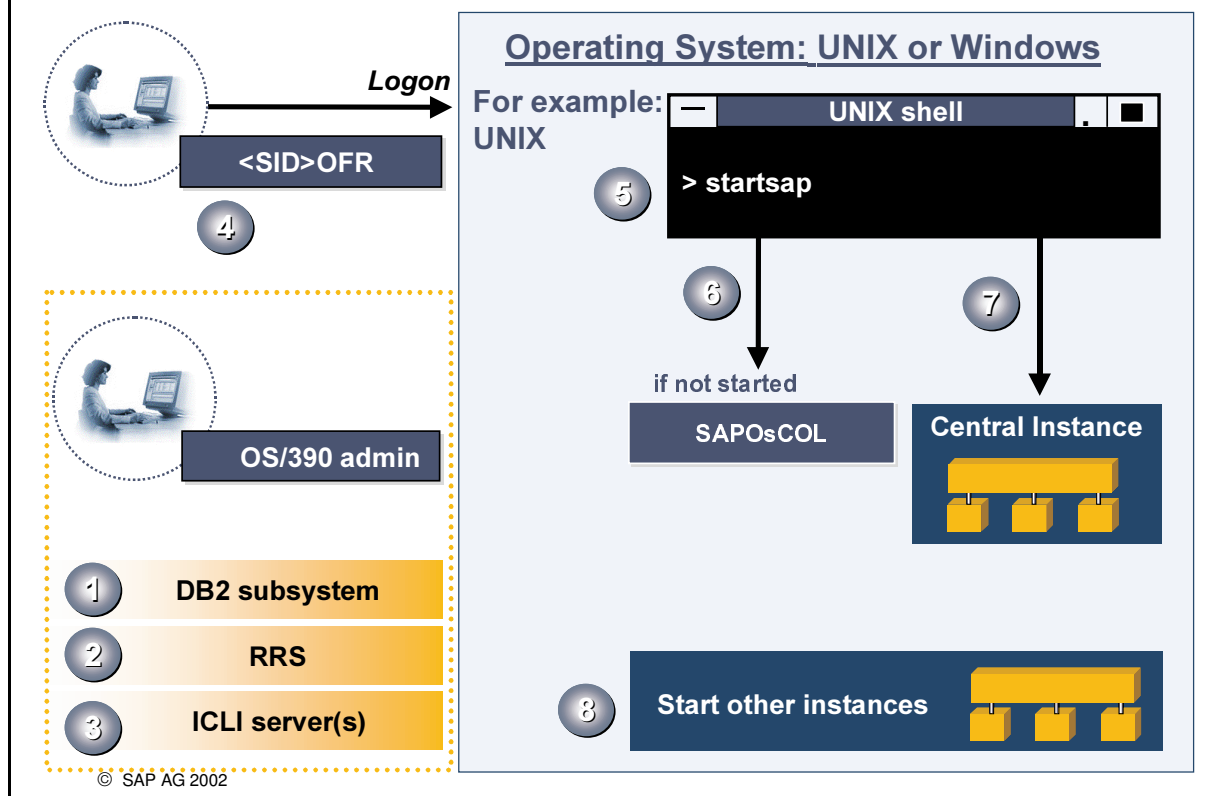
- When you stop the SAP system, you should first stop all dialog instances and then stop the central instance. There are two ways to do this:
 - From the SAP system using the CCMS Control Panel.
 - Log on as the SAP administrator (<sid>adm) at operating system level and enter the command *stopsap*.
- The *stopsap* script can be called with the following options:
 - DB: Stops the database system with the help of the *stopdb* script.
 - R3: stops the instances of the SAP system.
 - ALL: Stops the database system and the instance (default setting, can be omitted).
- The database can be stopped separately with database tools.



- Logon on to the AS/400 system with the SAP user profile for administrators. The authorizations of the group profile <SID>OPRGRP are required for this user (such as user profile <SID>OFR or <SID>OPR).
- Enter the AS/400 command STARTSAP and request parameters with F4.
- Under *SAP System ID*, enter the name of the SAP system that you want to start (such as DEV).
- Under *R/3 Instance*, enter the number of the instance that you want to start (such as 00). To start all instances on one or more hosts, choose *ALL.
- Under *R/3 Instance Host Name*, enter the name of the host on which the instance is to be started. To start all instances on all hosts, choose *ALL. (You must also have selected *ALL under *R/3 Instance*).
- Confirm your entries with ENTER. The subsystem R3_nn is then started for every started instance (*nn* is the instance number). All associated SAP services are started together with the subsystem (such as dispatcher, work processes, and spool processes).
- Enter the AS/400 command WRKACTJOB SBS (R3_nn) to check whether the SAP system(s) was started correctly.
- The system displays a screen with the column *Status* on the right side. If an SAP system was started correctly, the first line of this column should contain the value RUN (running), SELW (waiting), or SEQW (waiting for completion of unlocking). It can take a few minutes for the file system to reach this status.



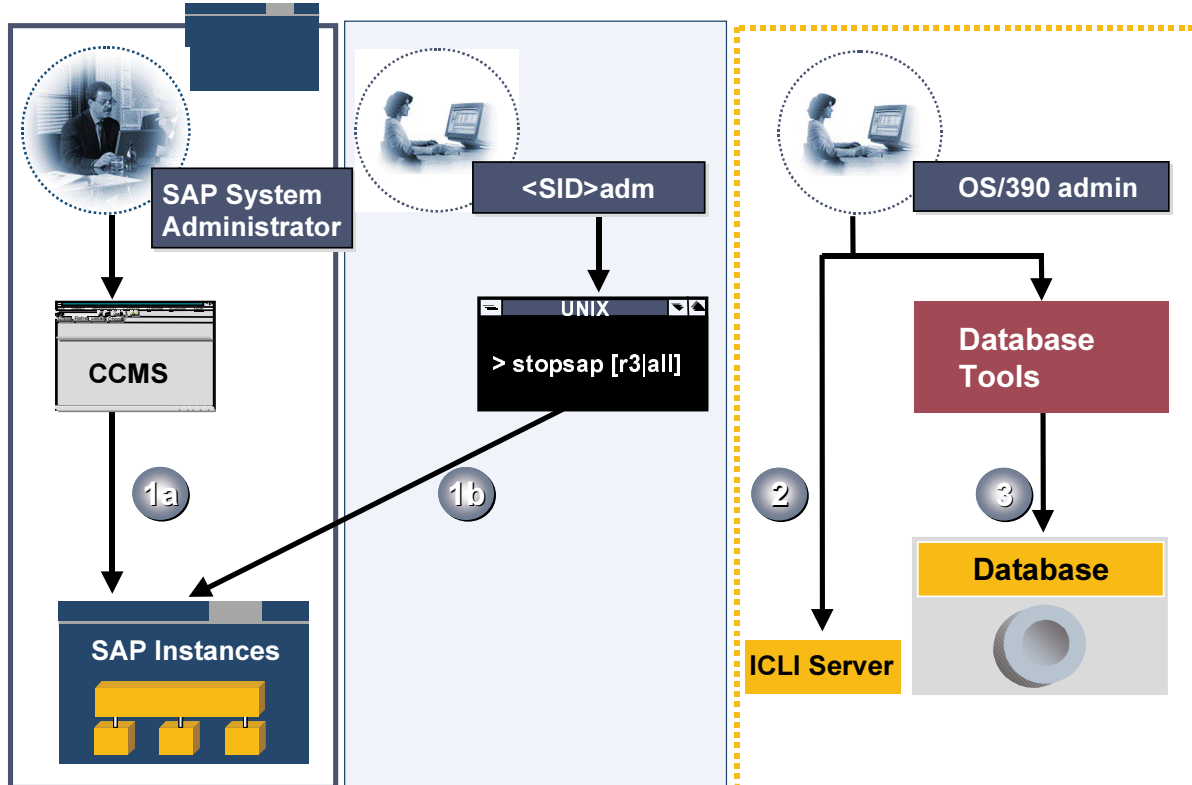
- Logon on to the AS/400 system with the SAP user profile for administrators (<SID>OFR or <SID>OPR).
- Enter the AS/400 command STOPSAP and request parameters with F4.
- Under *SAP System ID*, enter the name of the SAP system that you want to stop.
- Under *R/3 Instance*, enter the number of the instance that you want to stop, such as 00. To stop all instances on one or more hosts, choose *ALL.
- Under *R/3 Instance Host Name* enter *LOCAL to stop one or more instances on the local host. To stop all instances on all hosts, choose *ALL. (You must also have selected *ALL under *R/3 Instance*.)
- If you enter *YES under *Wait for instance to end*, the command STOPSAP waits until the SAP instance is shut down before stopping the SAP system. (The instance is regarded as shut down when the number of active instance user jobs in the instance subsystem, excluding the job SAPOSCOL, is zero.)
- Under *Maximum wait time (seconds)*, you can enter the maximum time that the command should wait for the instance to be shut down. The default value is 120 (two minutes). If it takes longer than two minutes for the instance to be shut down, an exception message is sent.
- Confirm your entries with *Enter*.



- In an OS/390 environment, an S/390 system serves as a database server. You can use Microsoft Windows or UNIX systems as application servers.
- First start the processes that allow the SAP work processes to connect to the database.
 - Start the DB2 database on the OS/390 system with the appropriate operating system commands.
 - Start the Recoverable Resource Manager (RRS) to synchronize all resources supported by OS/390.
 - Instances on non-OS/390 systems require the ICLI servers to allow communication with the database. To allow this, start the ICLI server(s) on the OS/390 system. Use either the OS/390 JCL operating language or a script that uses the OS/390 UNIX system services.
- To start the SAP system, log on as operating system user <sid>adm on the UNIX or Microsoft Windows application server.
 - First, start the central instance. Call the *startsap* script under UNIX or use the Microsoft Management Console under Microsoft Windows. If the process SAPOsCOL has not yet been started, this is started first. The processes of the central instance are then started.
 - The work processes then connect to the database.
- Other instances can then optionally be started

OS/390: Stopping

SAP

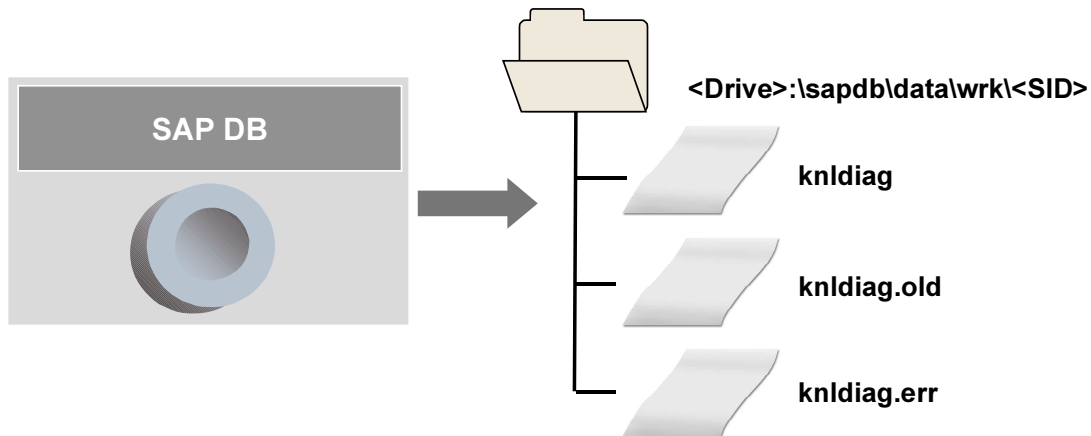


- You can stop the SAP system using either the CCMS Control Panel or operating system resources. At operating system level, under UNIX, run the command *stopsap r3*, or use the Microsoft Management Console under Microsoft Windows.
- The ICLI servers and the DB2 database are stopped on the S/390 system.
 - Stop the ICLI servers either on the OS/390 system console with the command *MODIFY BPXINIT, TERM=<PID>* or by using the OS/390 UNIX system services with the command *kill <PID>*.
 - Check that the ICLI servers have been stopped.
 - Stop the database.



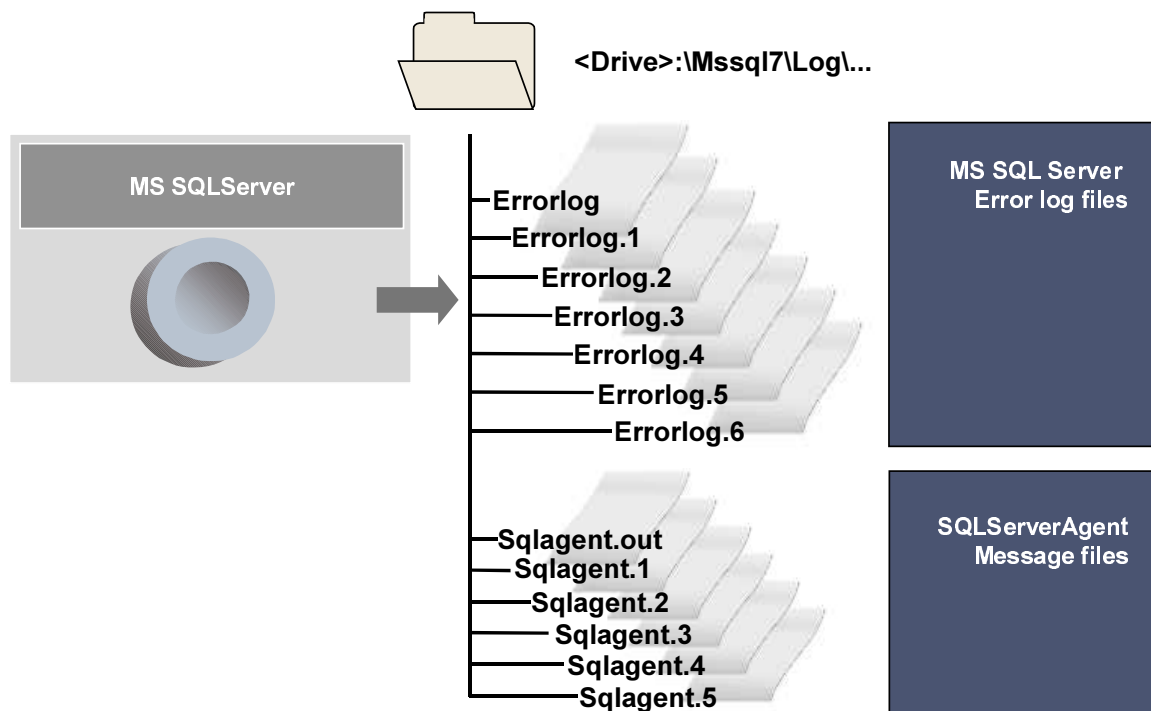
Log Files of the Database Systems

© SAP AG 2002



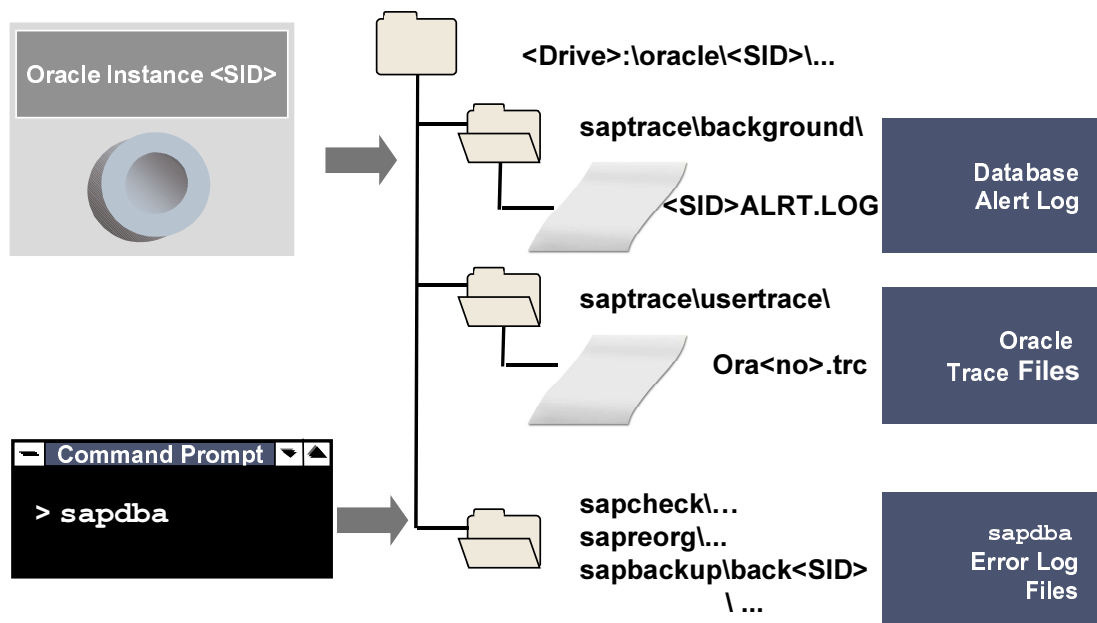
© SAP AG 2002

- SAP DB logs system messages and error messages in the directory: `<Drive>:\sapdb\data\wrk\<SID>`
- The system messages are recorded in the kernel log *knldiag*. This contains the following messages in chronological order:
 - Database start and stop
 - Information about the physical storage areas
 - User processes
 - System error messages
- The log is run as a circular memory file that is overwritten as soon as it reaches a certain size. A new log file is created after every start of the database system. A backup copy of the old log (*knldiag.old*) is created before a restart of the database system.
- All error and warning messages concerning the database system are recorded in the error log (*knldiag.err*), including the messages for the system start and stop.



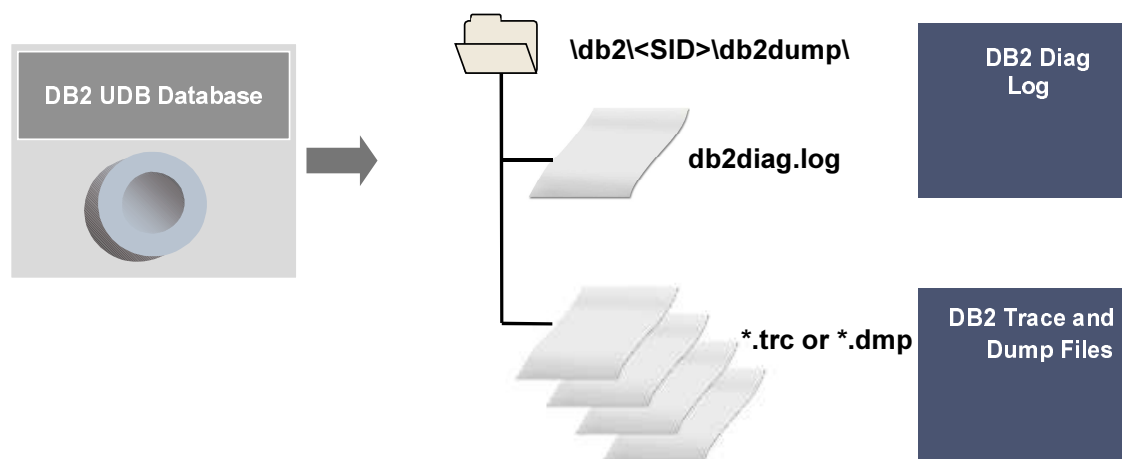
© SAP AG 2002

- The MS SQL Server logs all significant events such as starting and stopping the database and error messages in the file <Drive>:\MSSql7\Log\Errlog. A new error log file is written at every start of the MS SQL server. Multiple versions of these error log files are stored as Errlog.1, Errlog.2, and so on. The oldest available version is stored as Errlog.6. At every restart of the SQL server, the oldest file is overwritten. If a serious database problem occurs, you should backup the oldest file before the restart, so that this information is not lost.
- These error log files can be displayed in the Enterprise Manager.
- Messages of the SQLServerAgent service, which is required for backups, are also logged in the above directory in the file *Sqlagent.out*. The last six versions of this log are also retained.



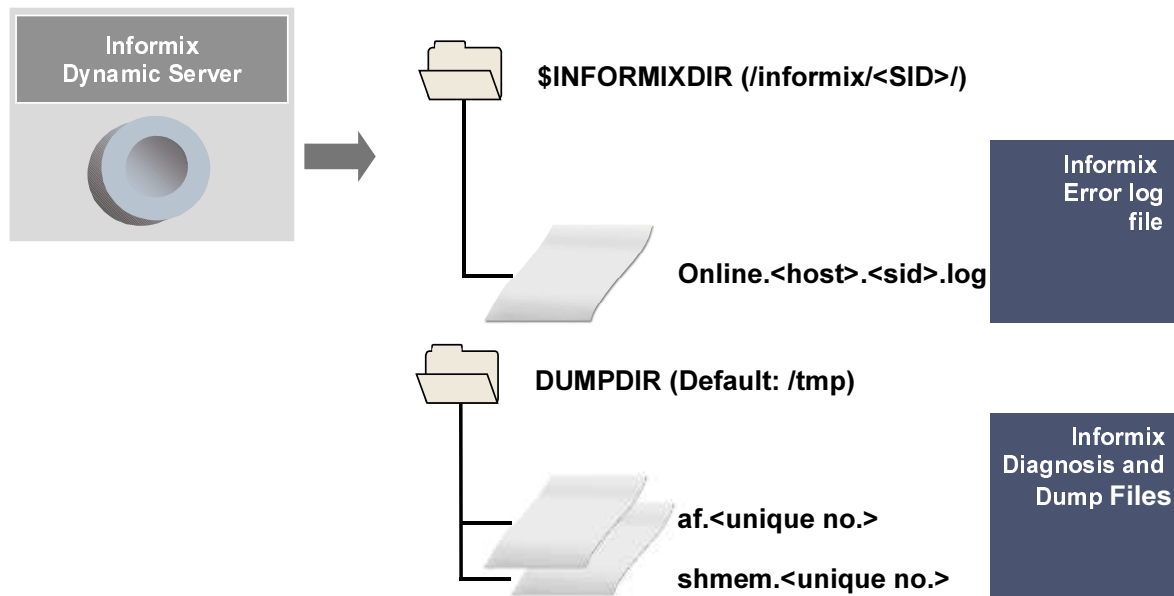
© SAP AG 2002

- The Oracle database logs all significant events such as starting and stopping the database and error messages in the file `<Drive>:\oracle\<SID>\saptrace\background\<SID>ALRT.LOG`.
- Detailed information about errors is logged in the Oracle Trace File: `<Drive>:\oracle\<SID>\saptrace\usertrace\Ora<no>.trc`.
- If the system administrator administers the database with `sapdba`, this writes its own log files in the following directories:
 - `<Drive>:\oracle\<SID>\sapreorg`
 - `<Drive>:\oracle\<SID>\sapcheck`
 - `<Drive>:\oracle\<SID>\sapbackup`



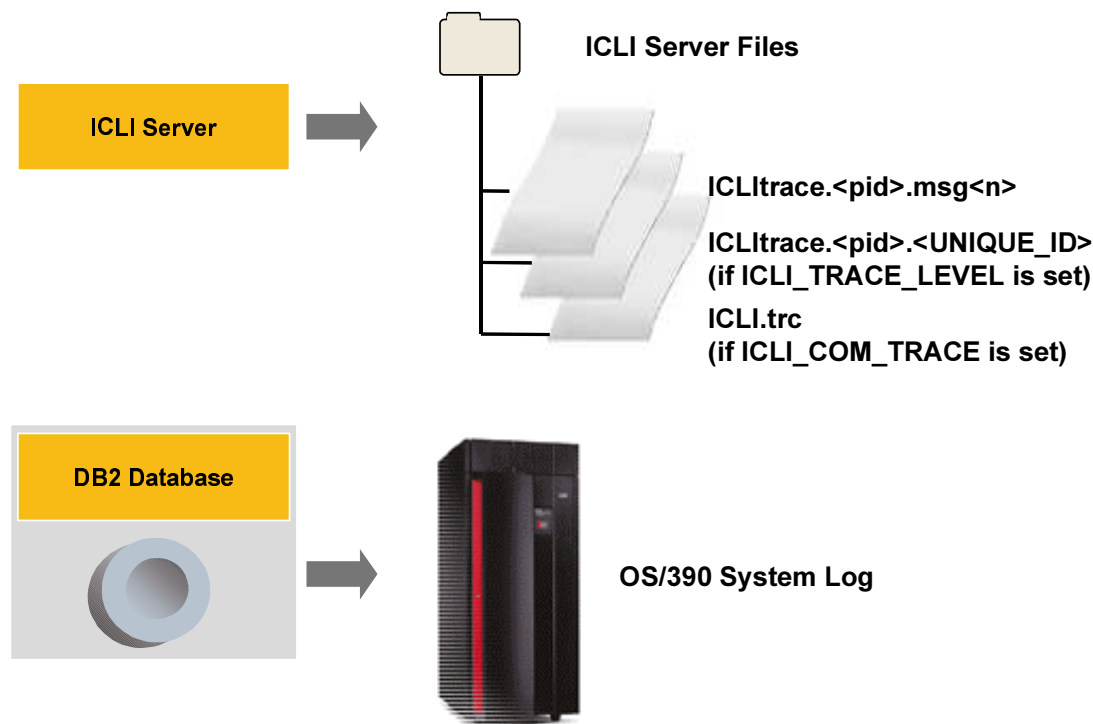
© SAP AG 2002

- The DB2 database logs all significant events in the file `db2diag.log`. The path under which this is stored is defined using the Diagnostic Directory Data Path parameter (DIAGPATH). This path is configured in the database manager configuration. The default value is `$DB2INSPROF/DB2INSTANCE`.
- The `db2diag.log` file contains the following information:
 - The place at which the reported error occurred. Application IDs allow the comparison of entries that belong to one application in the file `db2diag.log`.
 - A diagnostic message with the reason for the error. The message usually begins with DIA.
 - All other available data such as SQLCA data structures and pointers to other dump or trace files.
- Detailed information about errors is logged in the DB2 trace or dump files, which are also stored in the path defined using parameter DIAGPATH. These files are only created if a serious internal DB2 error occurs.
- You can access the dump directory by calling transaction DB6COCKPIT and choosing *Diagnosis* → *Dump Directory* in the navigation box.
- If you want to display the contents of an error log or a trace file, select the file.



© SAP AG 2002

- All significant events, such as starting and stopping the database and error messages, are logged by the INFORMIX database in the file `$INFORMIXDIR/online.<hostname>.<sid>.log`.
- Detailed information for individual errors is logged in the trace file `af.<unique no.>`. In certain cases, the contents of the shared memory are copied to the files `shmem.<unique no.>`. The directory in which these files are stored is defined using the parameter `DUMPDIR`. The default value of this parameter is `/tmp`.



© SAP AG 2002

■ ICLI Server Message Reporting:

- ICLI Server Messages for one or more ICLI servers are written to one or more ICLI error log files. The file names follow the naming convention `ICLI.<pid>.msg<n>`, where `<pid>` is the process ID of the ICLI server and `<n>` identifies multiple error log files of an ICLI server.
- The error log files of the ICLI are either stored in a directory that is determined during the start of the ICLI servers using the option `-LOGDIR`, or in the home directory of the user under which they were started.

■ ICLI Client Message Reporting:

- The ICLI servers also send messages to the communication partners of the client. These are logged in the developer traces and in the system log of the SAP system.

- The DB2 database logs all significant events and errors in the OS/390 system log.

Exercises

The exercises for Starting and Stopping the SAP System are divided into two parts. The second part is optional.

As the training systems are already started, the SAP systems should first be stopped and then started.

No.	Exercises
	Part 1: Starting and Stopping with the Microsoft Management Console
1	Stopping the SAP System
1.1	Before you stop the SAP system, check which users are logged on to your system. Send a system message, if necessary.
1.2	Stop the SAP system using <i>Microsoft Management Console</i> . To do this, log on to the server for your training system using the <i>Terminal Server Client</i> .
2	Starting the SAP System
2.1	Start the SAP system using <i>Microsoft Management Console</i> . Monitor the start of the processes of your system.
2.2	Which process types are started at operating system level after your system is started up?
2.3	Check whether your system started correctly. To do this, log on to your SAP system and call the process overview. Compare the list of processes at operating system level with the process overview in the SAP system.
	Part 2 (Optional): Start and Stop with a Command Call
3	Stopping the SAP System
3.1	Log on to the server with the user <sid>adm through a <i>Telnet</i> connection. Check whether your SAP system is running at operating system level. To do this, run the command <code>tlst /tl more</code> , to display an overview of operating system processes.
3.2	Enter the command <code>stopsap</code> to stop an instance. To stop the entire SAP system, first stop the dialog instance and then the central instance. The command <code>stopsap</code> only stops the selected instance. If the database is to be stopped, this must be done using database tools.
4	Starting the SAP System
4.1	Start the SAP instances by entering the command <code>startsap</code> . Start the central instance first and then the dialog instances.

No.	Solutions
	Part 1: Starting and Stopping with the <i>Microsoft Management Console</i>
1	Stopping the SAP System
1.1	<p>Before you stop the SAP system, check which users are logged on to your system. Send a system message.</p> <p>You can display the users logged on to your SAP system using the global user overview (<i>Tools → Administration → Monitor → Performance → Exceptions/Users → Active Users → Users global</i>, Transaction AL08). In this transaction, you will find your own user more than once. Transaction AL08 collects the user information for all instances using RFC connections. This means that your own user that initiates the RFC connections is displayed additionally once per instance.</p> <p>To send a system message to the users, call <i>Tools → Administration → Administration → System Messages</i> (Transaction SM02). Choose <i>Create</i>, enter the system message text, and choose <i>Save</i>.</p>
1.2	<p>Stop the SAP system using <i>Microsoft Management Console</i>.</p> <p>To do this, log on to the server for your training system using the <i>Terminal Server Client</i>.</p> <p>Start the <i>Terminal Server Client</i> and enter the physical host name under <i>Server</i>. Choose the display resolution for the remote connection on your frontend, and choose <i>Connect</i>. Log on to the operating system as user <sid>adm.</p> <p>Start the <i>Microsoft Management Console</i> by double clicking the appropriate icon. Select the desired instance(s) and choose the <i>Stop</i> function from the context menu (right mouse button). Depending on whether you have chosen an individual instance or the SAP system, either an individual instance or the entire SAP system is stopped.</p>
2	Starting the SAP System
2.1	<p>Start the SAP system using <i>Microsoft Management Console</i>. Monitor the start of the processes of your system.</p> <p>Choose the node for the central instance in the <i>Microsoft Management Console</i> tree. Select the desired instance(s) and choose the <i>Start</i> function from the context menu (right mouse button). Start the dialog instance in the same way.</p>
2.2	<p>Which process types are started at operating system level after your system is started up?</p> <p>You can monitor the processes at operating system level with <i>Task Manager</i> or the <i>Quick Slice</i> tool (<i>Start → Run: qslice</i>).</p> <p>The following process types are started at operating system level after your system is started up: <i>saposcol.exe</i>, <i>msg_server.exe</i>, <i>gwrld.exe</i>, <i>icman.exe</i> and multiple <i>disp+work.exe</i>.</p>