

Network Integration Guide (BC-NET)



HELP.BCNET

Release 4.6C



Copyright

© Copyright 2001 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.

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, 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 invented 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 in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax
	Tip

Contents

Network Integration Guide (BC-NET)	6
Basics of SAP Communication	8
R/3 Architecture	9
Communication Connections of the R/3 System	13
Planning and Conception	20
Planning a Network	21
Important Aspects when Planning Your Network	22
Network Topology	26
Network Integration of R/3 Servers	29
Basic Principles of TCP/IP Configuration for R/3 Servers	30
Host Names of R/3 Servers	35
Name and Address Resolution for R/3 Servers	36
Configuring R/3 Servers with Multiple Network Cards	42
Network Integration of SAP Frontends	51
SAP Frontend Communication	52
Printing	57
SAP Online Documentation	61
Installing/Upgrading the SAP Frontend Software	65
SAP Internet Integration	73
Introduction	74
Internet Technology Overview	75
ITS Technology	80
ITS Network Connections	90
ITS Scaling and Availability	94
ITS Security	100
Web Client Connections	109
ITS System Landscape	114
SAP Communication in a Wide Area Network	117
Network Security	124
Controlling Access	126
Encrypting SAP Network Connections	133
Network Load	139
Frontend Network Load	140
Network Load with Windows Terminal Server	143
Network Load in the SAP Server Network	144
Network Load from Printing	146
ITS Network Load	150
Network Load from CPI-C Data Transmission	152
Network Load from RFC Data Transmission	153
Appendix: Examples of R/3 Network Configurations	157

Network Integration Guide (BC-NET)

Introduction

The SAP R/3 System is a client/server system. The client/server applications communicate with each other mainly over network connections, so that the quality of the network connection has a very strong influence on the stability and performance of the entire SAP R/3 System.

For this reason, integrating an SAP System into an existing corporate IT infrastructure is a demanding task.

It is particularly important when introducing or expanding an existing SAP R/3 System to begin the necessary planning early.

You need R/3 Basis specialists, and operating system and network specialists for this task. These three areas are often independent of each other.

This document tries to connect these areas and focuses on the following target groups:

- R/3 Basis specialists
- Network specialists with SAP contact
- IT project managers and consultants in the SAP environment

Structure of this Document

The Network Integration Guide is divided into the following main subsections:

I. Introductory information

The section [Basics of SAP Communication \[Page 8\]](#) describes the R/3 architecture and the network requirements that result from it, as well as the different communication connections for an SAP System.

II. Starting with network planning

The section [Planning and Conception \[Page 20\]](#) describes a general procedure for SAP network planning, and an extensive list of all the important points during the planning. There is also an explanation of [network topology \[Page 26\]](#) in this section.

III. Comprehensive documentation

This section describes in detail the following topics:

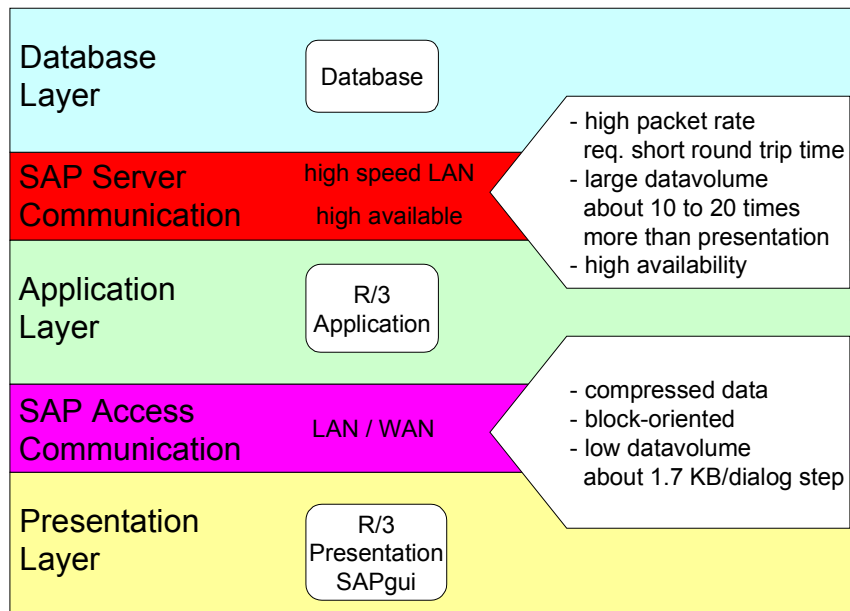
- [Network integration of R/3 servers \[Page 29\]](#)
- [Network integration of R/3 frontends \[Ext.\]](#)
- [SAP Internet integration \[Page 73\]](#)
- [SAP communication in a wide area network \[Page 117\]](#)
- [Network security \[Page 124\]](#)
- [Network load \[Page 139\]](#)

IV. In the [appendix \[Page 157\]](#), you can find examples of network topologies for small, mid-sized and large SAP Systems.

Basics of SAP Communication

R/3 Architecture

The SAP R/3 System has a three-tier client/server architecture. All data is stored in a database, and the data is processed in the application layer on the application servers. The SAPgui frontend (presentation layer) is the interface to the user. All three layers are connected to each other with networks. The following graphic depicts the client/server architecture of the R/3 System, and the communication requirements between the presentation and application layers and between the application and database layers:

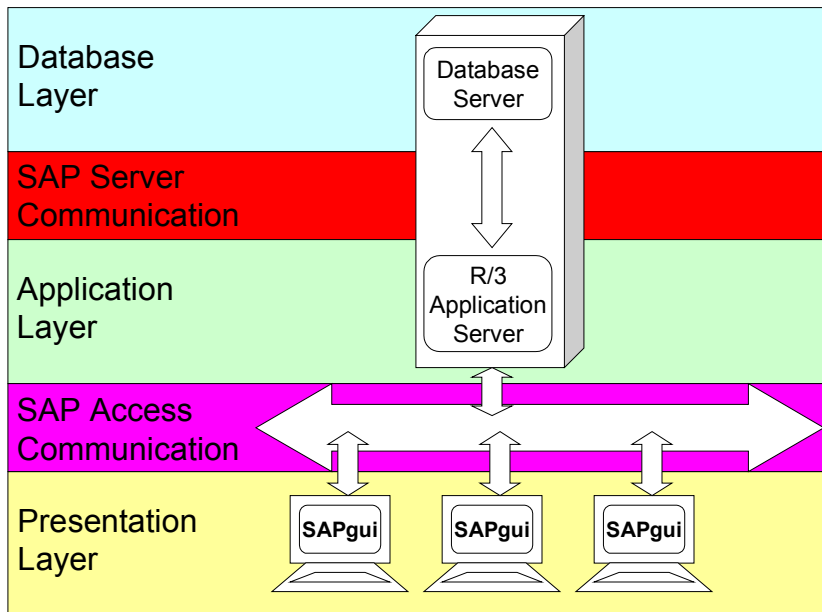


Depending on your requirements, you can distribute the services to different hosts.

Smaller applications keep the database and the R/3 application on the same host. The large volume of data that passes between the R/3 application and the database (SAP server communication) is processed locally and not through a network.

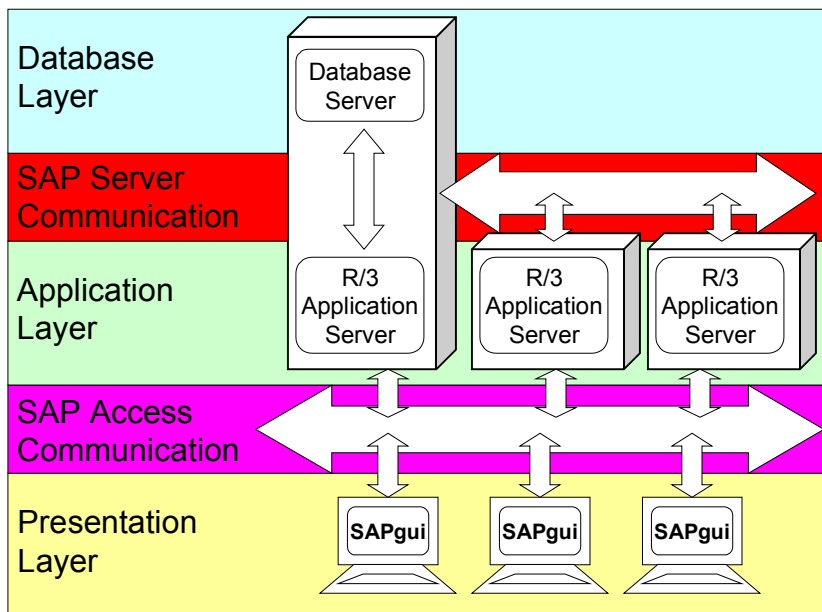
The presentation layer is usually made up of PCs on which the SAPgui frontend is installed. The SAPgui is not a terminal emulation but an application program that displays R/3 application data graphically. This means that there are no great demands placed on the connection between the SAPgui frontend PCs and the R/3 application (access communication).

R/3 Architecture



Higher processing demands on the R/3 application can be realized by additional application servers (application servers are hosts on which the R/3 application runs).

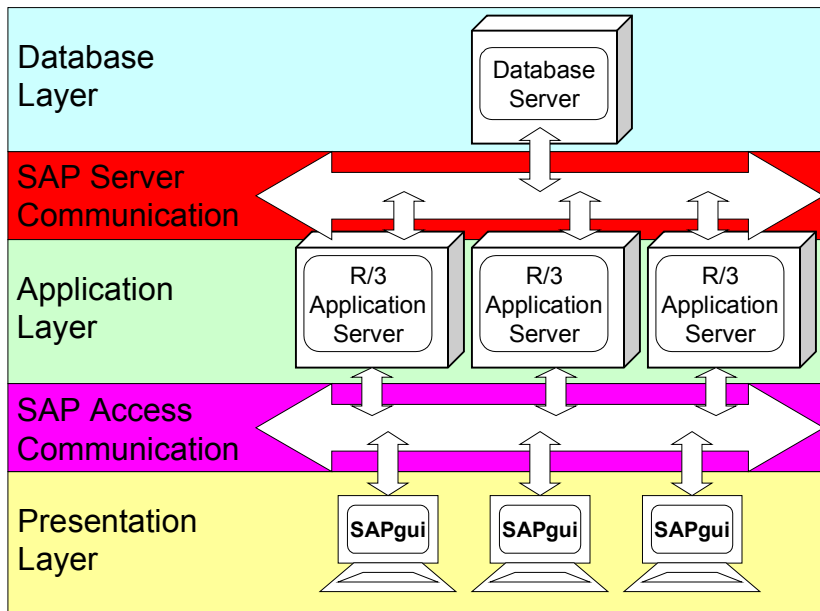
Very high demands are made on bandwidth and delay time between the application servers and the database server. You need to set up a suitable network connection to meet these demands.



R/3 Architecture

You can speed up and secure data throughput to the database by placing the database on a separate host. The database server host then communicates only with the R/3 application servers. By isolating the database completely from the rest of the corporate network, you prevent unauthorized access to sensitive data and ensure high performance.

For data backup purposes you may need to connect the database server to a dedicated network (SAN = Storage Area Network).



Access Communication

Access communication covers all access to the R/3 System. This includes user access through the SAPgui, and also links to other R/3 Systems and external applications.

An access network is not a dedicated network segment, but includes all network segments through which the R/3 System is accessed.

Important Design Criteria for the Access Network:

- Block-oriented data traffic → delay times in the network are relatively non-critical for the SAPgui
- Bandwidth must be determined separately for each location, depending on the number of users and their activities
- The availability of the network must also be specified according to the user group or location

Server Communication

Server communication covers all communication between the application servers and the database server, and is of great importance for the R/3 System. In each individual case you must decide whether you want to process the server communication through its own physical network (server network), or whether you want access and server communication to share a physical

R/3 Architecture

network. You must remember that any worsening in server communication also has a negative effect on the performance of the R/3 System.

A server network is the network connection between the servers (application servers and database servers) of an R/3 System.

Important Design Criteria for the Server Network:

- High throughput of data or high bandwidth
- Minimum delay time (round trip time)
- Includes the servers of the R/3 System only
- High availability
- No non-SAP data traffic (for example, data backup over the network)
- Direct server connection → no expensive cabling
- Greatest possible security against unauthorized access to the database server if the server network is set up as a separate segment from the rest of the corporate network

Network Topology for the R/3 System

See [Network Topology \[Page 26\]](#)

Communication Connections of the R/3 System

The R/3 System supports the following communication connections:

- Presentation frontend (SAPgui) to the R/3 System
- Connections from the R/3 System to printers
- Connections to other R/3 Systems
- Connections to external applications

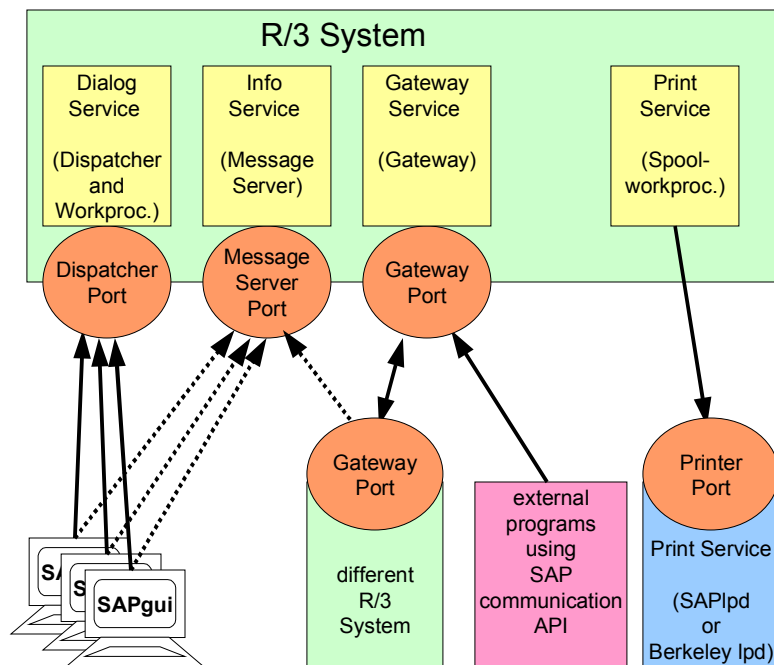
You can also use these connections with SAProuter.

The following communication connections exist within an R/3 System:

- From the application server to the database
- From the application server to the message server
- Communication between application servers

Representation of External Communication Connections

The following graphic shows you all the different ways an R/3 System can communicate. The graphic includes TCP connections only. The R/3 System can consist of one or more different hosts. The communication connections within the R/3 System are described later.



The arrows show in which direction the TCP connections are set up.

The following describes these connections in detail.

Communication Connections of the R/3 System

Connection to Frontends

A dispatcher runs on each application server that can connect the SAPgui clients with dialog work processes, if needed. You can access this dispatcher under the TCP port `sapdp<nr>`, where `<nr>` is the instance number (00 to 99) of the application instance. The default value is `sapdp00` and the corresponding TCP port number is 3200.

In each R/3 System there is one info service that can be used for the variable assignment of SAPgui frontends to an application server (dispatcher). This service is provided by the message server. You can use this info service to organize the frontends into groups (logon groups) and distribute them according to the load on the various application servers. The message server normally runs as part of the central instance of the R/3 System, and you can access it with the TCP port `sapms<SID>`. `<SID>` is the system ID of the R/3 System (for example, C11). You can choose different TCP port numbers for different R/3 Systems. They usually start at 3600 and increase by one each time.

Default values for instance number 00:

Dialog Service (dispatcher port):	<code>sapdp00</code>	3200/TCP
Info Service (message server port)	<code>sapmsC11</code>	3600/TCP

Connecting Printers

The R/3 System uses spool work processes to process print requests. There can be one or more spool work processes, which themselves can run on one or more application servers. The way in which the spool work processes print depends on your configuration. The system can print in one of the following ways:

- Output to local operating system spooler, on the application server host (no network connection)
- Output to the standard "line printer service" (lp) of the target host (TCP service printer with the TCP port number 515)
- Output to the SAP printer daemon (SAPlpd) that is running on the target host (TCP port number 515)
- Print output using the dialog connection of the SAPgui frontend. The existing SAPgui connection is used, which means that no specific port is needed.

Connections to External Partners or Other R/3 Systems

Each R/3 application server has an SAP gateway that it uses to communicate with other SAP Systems, or with other applications that use the SAP communication interfaces for CPI-C or RFC.

CPI-C (*Common Programming Interface for Communications*) is a standardized protocol for the simple transmission of data between two programs. CPI-C uses half duplex transmission. This means that only one party at a time can transmit data. The receiving party must keep receiving data until it is its turn to transmit.

RFC (*Remote Function Call*) is SAP's own communications interface. RFC communication always involves a caller (RFC client) and a receiver (RFC server). The RFC server provides one or more function modules that can be called. An RFC client can call one of these function modules, transmit data, and then read the results of the function module. Both the server and the client can be either external programs or the SAP System.

Communication Connections of the R/3 System

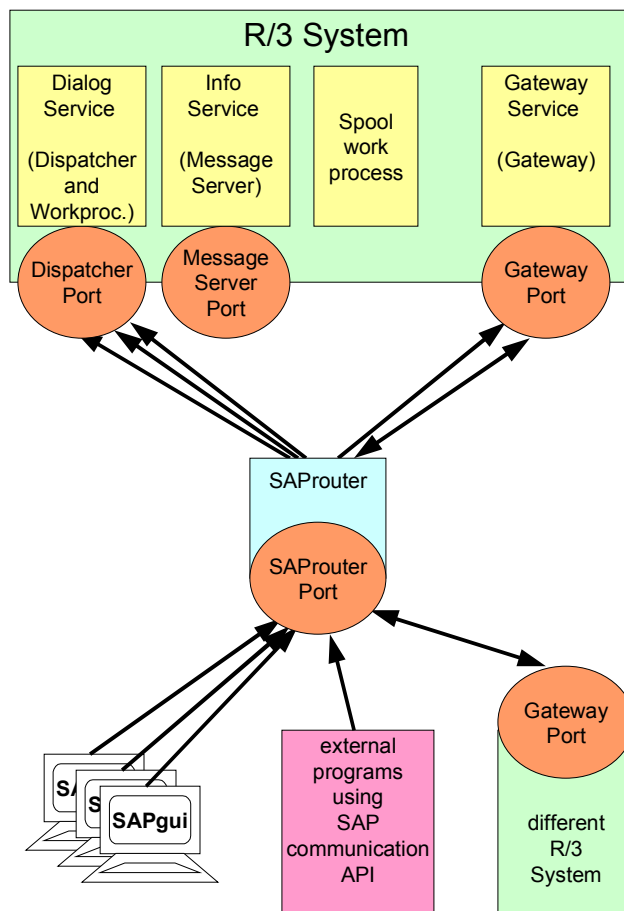
You can also use the gateway for communication between two applications within an SAP System. You can access the SAP Gateway on every application server under the TCP port `sapgw<nr>`, where `<nr>` is the instance number of the application instance.

Default values for instance number 00:

SAP Gateway port: `sapgw00` 3300/TCP

Connections with SAProuter

The SAProuter program helps you to set up and control remote connections easily. SAProuter functions as an intermediary between the source system and the target system. First, the source system sets up a TCP connection to SAProuter and tells it to which target system it wants to connect. SAProuter then sets up a connection to the target system. A SAProuter connection from a source system to a target system consists of multiple TCP connections.



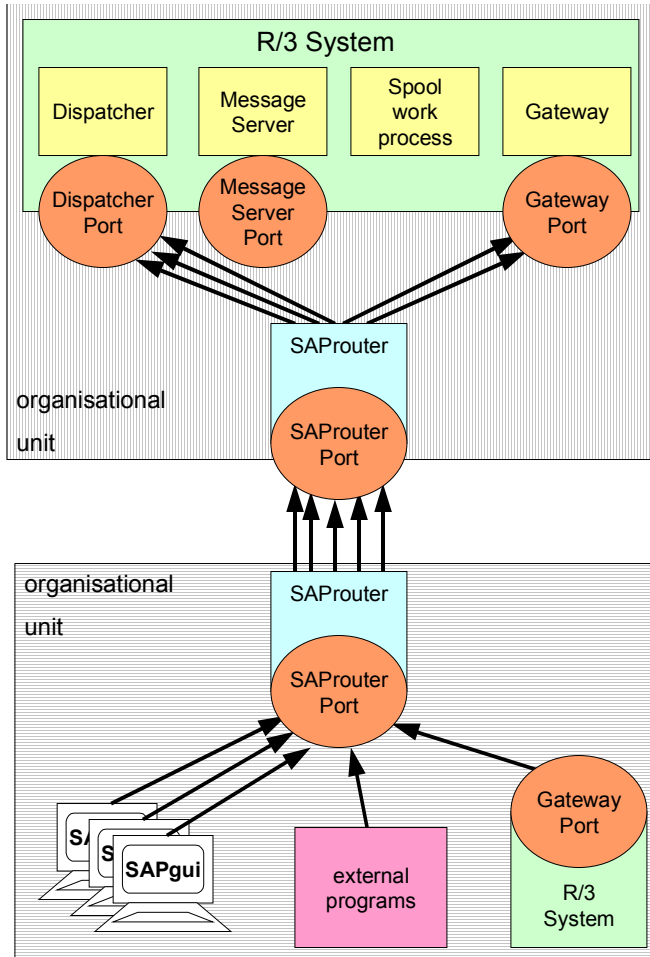
If necessary, a connection can be made using multiple SAProuters in sequence. If this is the case, all SAP connections between the SAProuters are relayed through defined TCP ports between defined partners (SAProuters). This makes it much easier to implement mechanisms that protect access to the connections.

Separate organizational units that administrate their networks and security separately can also take advantage of double SAProuters. Each unit defines its own SAProuter as a communication

Communication Connections of the R/3 System

port for all SAP communication that goes outside its limits. SAProuter then becomes an **application level gateway** that can run, for example, on a firewall host. You can use authorization tables to control who may set up a connection with whom.

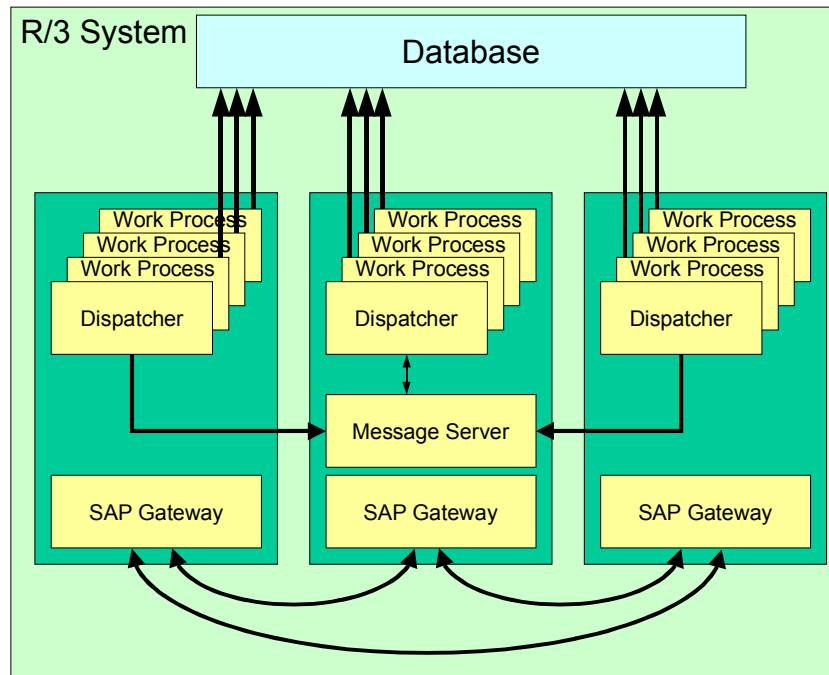
Each organizational unit can allow or disallow connections as it wants.



When you start the program, you can determine under which port you want to access the SAProuter.

Default values: SAProuter port: **sapdp99** 3299/TCP

Internal Communications



The following section describes these connections in detail:

Connecting to the Database

The connection between the R/3 application servers and the database server is based on a Remote SQL database interface.

The TCP port you use depends on the individual database system:

Oracle	tlisrv	1527/TCP
Informix	sapinf<sid>	3800/TCP
MS SQL Server	ms-sql-s	1433/TCP
ADABAS	sql30	7200/TCP
	sql6	7210/TCP
DB2	sapdb2<SID>	
	sapdb2i<SID>	

Connecting to the Message Server

The various instances (application servers) use the message server to access central services, such as the enqueue or update services. The message server also receives regular information from all instances about current system load and the services that are on offer. This means that the message server can provide information on load balancing for different applications, such as the SAPgui or background applications.

Communication Connections of the R/3 System

When an R/3 application server is started it sets up a connection to the message server of the central system (more precisely, each dispatcher sets up connection to the message server port `sapms<SID>`).

Connecting Application Servers

If application servers need to communicate with each other, a connection is set up between the SAP gateways of the application servers (port `sapgw<nr>`). Once it is set up, this connection remains until the application server stops running.

Table of Communications Connections

R/3 Service	Default TCP service name	Default TCP port number	Possible range for TCP service name	Possible range for TCP port number
Dispatcher	sapdp00	3200	sapdp00 – sapdp99	3200 – 3299
Message server	sapms<SID>	3600	sapms<SID>	Free choice
SAP Gateway	sapgw00	3300	sapgw00 – sapgw99	3300 – 3399
SAPlpd	-	515	-	Free choice
SAProuter	-	3299	Free choice	Free choice
Test program niping	-	3298	Free choice	Free choice
Oracle database	tlisrv	1527		
Informix database	sapinf<SID>	3800		
MS SQL Server database	ms-sql-s	1433		
DB2 UDB for AIX and NT database	sapdb2<SID> sapdb2i<SID>			Free choice
ADABAS database	sql30 sql6	7200 7210		

Planning and Conception

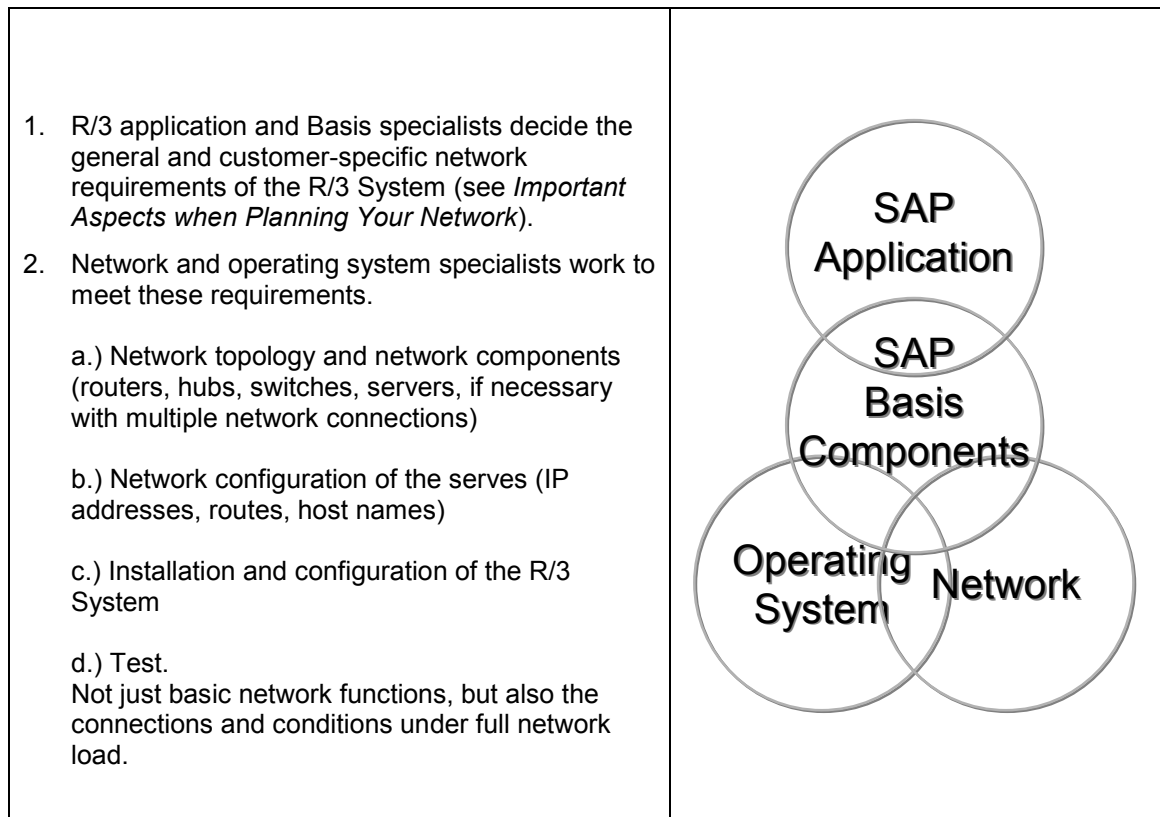
Planning a Network

[Wichtige Aspekte bei der Netzwerkplanung \[Page 22\]](#)

When you plan a network you need to consider all possible demands that might be placed on the network infrastructure by the R/3 System. Experience has shown that a completely new network is not usually set up for the R/3 System, instead it is integrated into an existing network infrastructure. Here as well, you need to compare the demands of the R/3 System with the characteristics of the existing infrastructure, and make any necessary adjustments or extensions.

The following graphic explains the areas that are affected, and where they overlap. Network planning needs to be a joint effort between several different specialist departments.

You should plan your network in two stages:



Important Aspects when Planning Your Network

Server Integration

Questions to be Answered:

- Does your SAP System consist of a single host or are there additional application servers?
- Does the database run on a separate host or together with an SAP instance?
- Where are the R/3 servers located?

Recommendations and Tips:

- Ensure that all the servers in an SAP System are located centrally, and that there is no WAN between the servers.
- For SAP Systems with several application servers, you must determine if the servers for this SAP System (database server and application server) are to be connected to each other by an additional, unique and separate network segment (server network).
See: [Network Topology \[Page 26\]](#)
- When choosing the host names for the servers, IP addresses, static routes, see [Host Names of R/3 Servers \[Page 35\]](#).

Frontend Integration

Questions to be Answered:

- Where are the branch office SAP users located and how many are there?
- What is the network bandwidth required for the branch offices connected over a WAN?
- What is the procedure for installing and upgrading the frontend software (SAPgui)?
- Which additional standard applications (e-mail, Internet browser, emulation, Office) will you use on the frontends?

Recommendations and Tips:

- A stable TCP connection to the R/3 Server is a prerequisite for frontends.
- If the frontends are connected to the R/3 Server using a LAN, the existing bandwidth is usually sufficient.
- For frontends connected by a WAN, you need to determine the required bandwidth for each location as accurately as possible.
See [SAP Communication in a Wide Area Network \[Page 117\]](#).
- Also consider additional frontend integration (printing, online documentation, installation and upgrade, mobile frontends...)
See [Network Integration of R/3 Frontends \[Ext.\]](#).

Printer Integration

→ Questions to Be Answered:

- At which branch offices are the printers located and how many are there?
- Which method is used to access the printers from the SAP System?
- What is the bandwidth required for the expected printing volume at the branch offices?

Recommendations and Tips:

- You can connect printers either directly to the R/3 Server or position them as network or frontend printers at any location on the network (depending on the access method). When you position printers, you must consider criteria such as availability, and printing speed and volume.
See [Printing \[Page 57\]](#) and the *BC R/3 Printing Guide - Planning and Installing the R/3 Print Architecture*.
- If printers are connected over a network, the same criteria applies as for frontends. This means that for all branch offices connected by a WAN, you need to estimate the bandwidth requirements for the expected printing volume. See [Network Load from Printing \[Page 146\]](#).

Links Between Applications

→ Questions to Be Answered:

- Are there transports between different SAP Systems, such as a development system and production system?
- Is there a temporary data transfer or connection to legacy systems?
- Are there long-term connections to other systems?

Recommendations and Tips:

- You can connect the R/3 System with other SAP Systems or external systems in very different ways. The basis for the connection between the two systems is usually a TCP connection. The direction in which the connection is set up depends on the application and its configuration. The connection can generally be made in both directions and include any application server of the R/3 System.
- For connections with mainframe-based legacy systems, the SAP gateway also supports SNA LU6.2 connections.
- To estimate the required bandwidth you need to know the application sufficiently well. See: [Network Load from CPI-C Data Transmission \[Page 152\]](#) and [Network Load from RFC Data Transmission \[Page 153\]](#).

Important Aspects when Planning Your Network

High Availability

Questions to Be Answered:

- How high must the availability be for the R/3 System (server)?
- Will you be using special high availability products such as *ServiceGuard*, *HACMP OBserve*, *MSCS* or *Solstice*?
- How high must the availability be for the different locations or user groups?
- Which dependencies arise when linking to external applications and what availability has to be ensured for these links?



Recommendations and Tips:

- If you use high availability products for the R/3 Server, you must follow certain rules for the network configuration. See *SAP R/3 in Switchover Environments*.

Security

Questions to Be Answered:

- What security requirements are made on the database server?
- What security requirements are made on the communication between the R/3 servers?
- What security requirements are made on the different connections between the clients and the R/3 servers?
- Clients in the LAN
- Clients in the WAN
- Will you be using the *Internet Transaction Server* (ITS)? If yes, which security measures are set up for accessing clients in the Intranet, Extranet or Internet?



Recommendations and Tips:

- There are different ways of protecting access and encryption when clients access the R/3 Server. See [Network Security \[Page 124\]](#).
- The connection between the R/3 servers (database server and application server) is normally unprotected. Therefore, you must protect these servers using network tools (such as a separate network segment for the server communication). See [Network Security \[Page 124\]](#).

Data Backup

Questions to Be Answered:

- How is the data backup of the database server organized?

Important Aspects when Planning Your Network

- How is the data backup of the application server organized?



Recommendations and Tips:

- When operating an SAP System, you need to continually back up the R/3 database. If the data is backed up at a central point over the network for administrative reasons (for example, by a tape library), there is a heavy network load during the backup. If the backup occurs during productive operation of the SAP System (which includes background processing), you need to consider that in the server network small increases in the delay time and limitations in the bandwidth can significantly decrease the performance of the SAP System.
- If you back up the data over the network, use a separate network segment (*Storage Area Network*, SAN) set up only for the data backup. To do this, you must install an additional network adapter on the database server.

Network Topology

Network Topology

As already described in the section [Basics of SAP Communication \[Page 8\]](#), an SAP System communicates using server communication and access communication.

There is no standard solution for access communication (client access or access methods using external systems). You must find an appropriate network topology to access the R/3 Server depending on the type and number of clients, and particularly depending on the location of the system.

For server communication topology, you must first determine the location of the SAP server installation (database server and application server). The SAP System consists of one or several servers that are usually responsible for the entire enterprise and are located at a central point, the computer center. This is a centralized concept (as with mainframes) and is an advantage for maintenance and operation. Numerous clients that are widely spread out access these servers.

The concept of a "server farm" is well suited to SAP Systems.

Note the following basic principles:

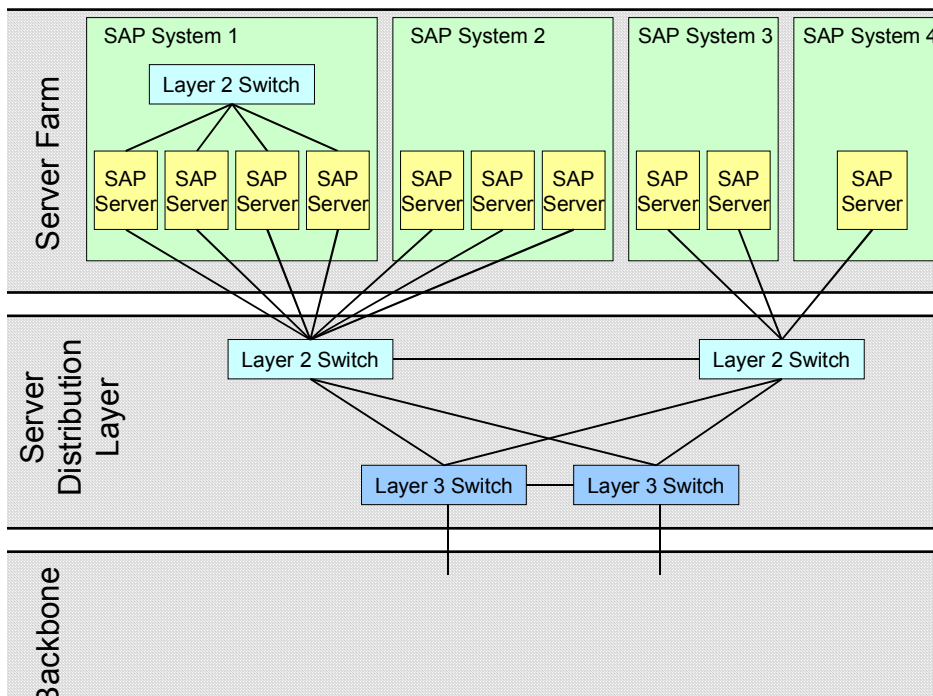
- All servers are connected to each other directly with a high bandwidth and minimal delay times. Switches are well suited for this.
- If the servers are connected using shared media (for example, the Ethernet), there may easily be temporary network overloads, called collisions. Since this lowers performance, we recommend that you use an additional, separate network segment to connect the servers of an SAP System.
- If you connect the R/3 Server with the campus or backbone network, you need routing functions. You can use routers or modern layer 3 switches (OSI Layer 3 = network layer).
- By creating redundant network paths, you can protect your system against failure.

If the network topology covers several SAP Systems, the server connections should only include those servers of one system only so that optimal performance is ensured for each SAP System.

Network Topology for Switched Networks

If the R/3 Servers are connected using switches, access to the individual servers has a high and constant bandwidth. The connection between two servers has a minimal delay time.

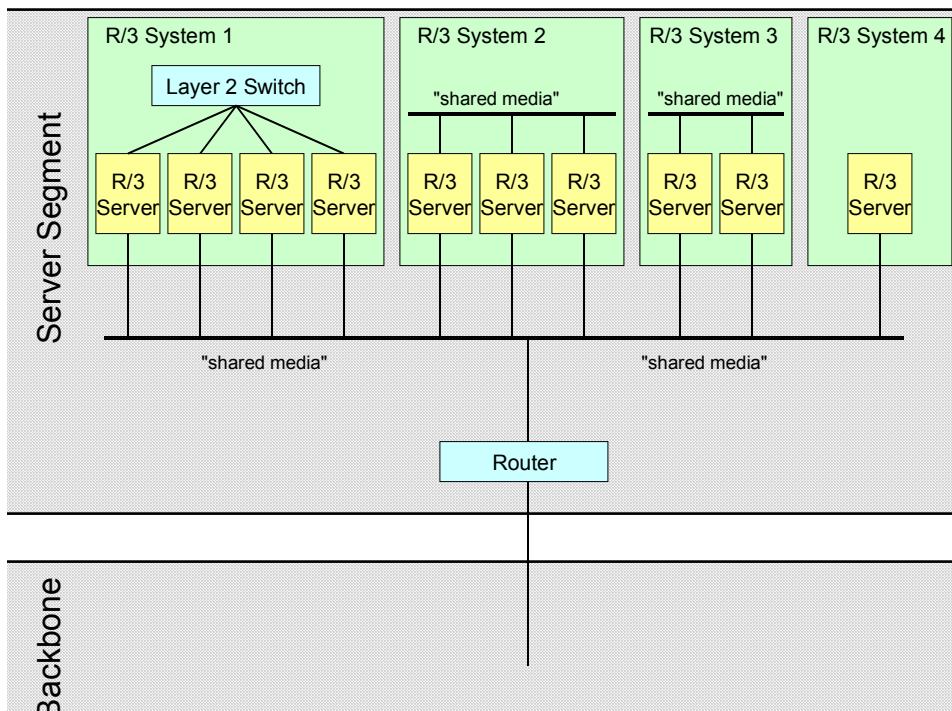
Therefore, you only need to connect the servers of an SAP System to each other with an additional network adapter and switch if there is extremely high throughput and bandwidth requirements (for example, if the throughput of the network adapter is not sufficient). (See SAP System 1).



Network Topology for "Old Style" Shared Media Networks

For the SAP servers, you need a connection with a high bandwidth with minimum delay times. Shared media networks can contribute to this only partially. Communication between servers may lead to conflicts and network overloads. In this case, we recommend an additional network segment that connects the servers of an SAP System. We also recommend that you use a switch to connect the servers (see SAP System 1) rather than a shared media connection (see SAP Systems 2 and 3).

Network Topology



Network Integration of R/3 Servers

Basic Principles of TCP/IP Configuration for R/3 Servers

Basic Principles

To transmit data between two hosts in one network, you need two **network adapters** (network interface card, NIC). One adapter sends a data packet and the other one receives it.

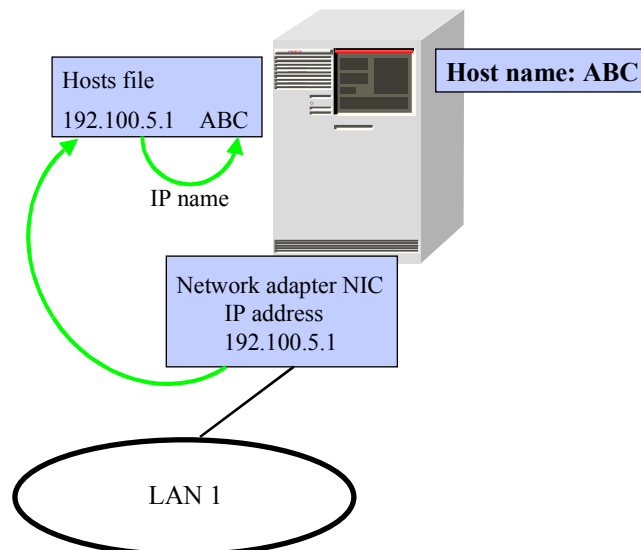
An **IP address** (IP = Internet protocol) is assigned to each network adapter. These numerical IP addresses are represented in the form *nnn.nnn.nnn.nnn*, where *nnn* is a number from 0 to 255 (for example, 192.168.1.10). An IP address functions as a unique ID for identifying the sender and the receiver in a network with multiple hosts.

The **IP name** is the name under which the network adapter is known in the network. An **IP name** is assigned to each IP address. This simplifies the use of IP addresses. This IP name is often mistakenly described as the host name.

The **host name** or computer name is a logical name for the host itself and is set in the operating system. The operating system knows its own host under its host name. You can query this name under UNIX or Windows NT by using the command `hostname`.

The IP name is mapped to the IP addresses using a name database. Here, it can be a local host file, or a name resolution service, such as DNS (Domain Name System/Service) or WINS (Windows Internet Naming Service).

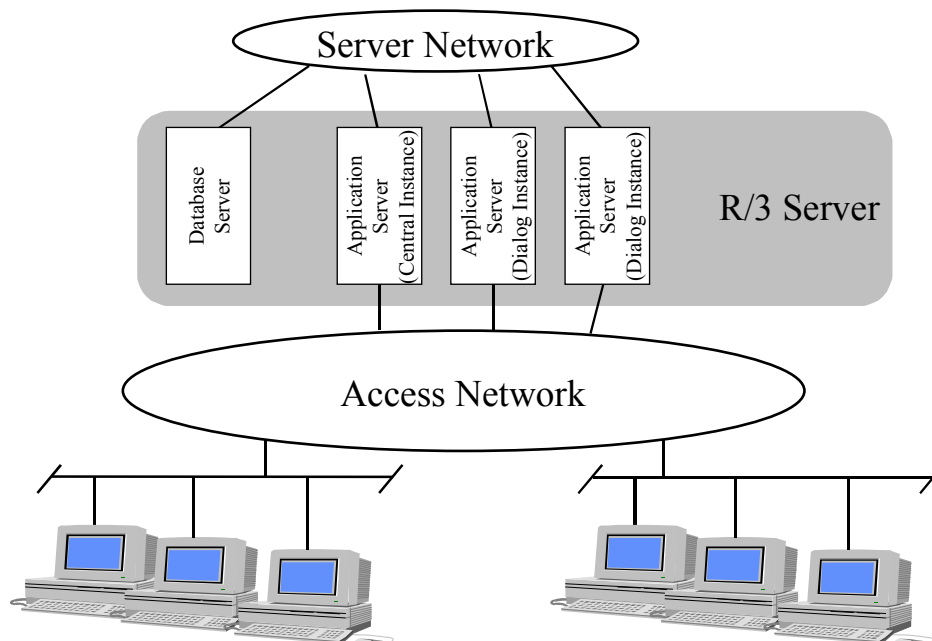
If the relevant host only has one network adapter, the host name and the IP name are normally identical. In this case, there is a unique relationship between the logical host name and the IP name (or IP address). This means that this host can be addressed by other computers using its host name.



Basic Principles of TCP/IP Configuration for R/3 Servers

Terminology

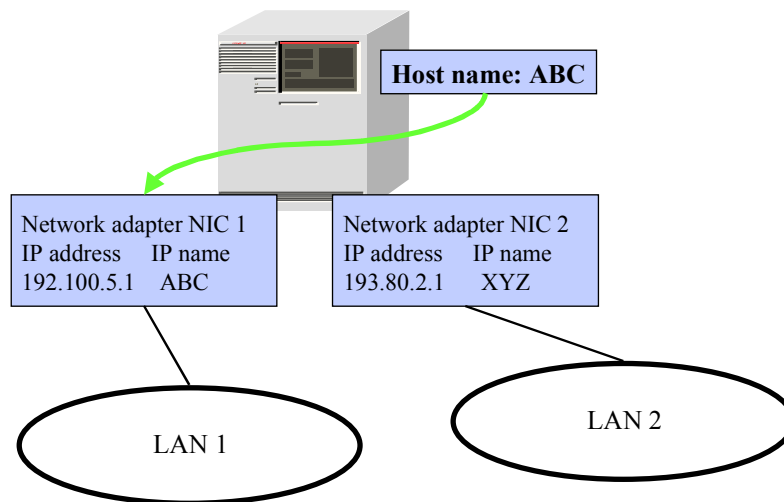
SAP server	All hosts (computers) that belong to an SAP System (database server and application server)
Database server	Host on which the database service runs. If an R/3 instance (central instance or dialog instance) also runs on the database server, the server must be handled like an application server when you configure it.
Application server	Host on which an R/3 application service runs (central instance or dialog instance)
Server network	High speed network segment through which the SAP servers communicate with each other
Access network (frontend network)	All network segments through which the frontend is connected with the application servers
IP address	Network address in the Internet protocol (for example, 192.5.2.1), assigned to a network adapter
IP name	Logical name for an IP address. Is often equated with the host name, although the relationship of the host name and the IP name is not always unique.
Host name	Logical name of the host. It is normally assigned like the IP name of an IP address for the host. A host can have several IP addresses and IP names, but it can only have one host name.



Basic Principles of TCP/IP Configuration for R/3 Servers

Multihomed Hosts

If a host has multiple network adapters (multiple NICs) or multiple IP addresses for each network adapter, it is called a **multihomed host**. In this case, the relationship between the IP addresses (with the assigned IP names) and the host name is no longer unique. The host name can be identical to the IP name for an IP address, or it can take on its own unique name. If a multihomed host is addressed by another host, one of the IP names (or the corresponding IP address) must be used. In the following graphic, LAN1 must use the IP name ABC to address the host ABC, whereas LAN2 must use the IP name XYZ to address the same host.



The name you choose by which a host is addressed, influences the route of the data traffic in the network. Application programs do not usually know the various IP names, or their accompanying LAN segments, but they do know the logical host name. Therefore, choosing the right host name is very important. If static IP routes are used, you can also access IP names and/or IP addresses that are not in your own subnet.

Using IP Routing to Choose the Route

If you want to use TCP/IP to send a data packet, the network layer (layer 3 in the OSI layer model or the **IP layer**) must determine if and how the target IP address can be accessed. The routing information necessary for this is stored in the routing table. You can set the entries in the routing table as being static (manually) or dynamic (using the routing daemon process). The entire route to the target address is not described in the routing table, instead, only the route up to the next node (next hop) to which the data packet is forwarded. There, a check is made in one of its own routing tables as to how the target address should be accessed. This is called an *indirect route*, since the route goes through at least one **gateway**. If the address can be directly accessed, this is called a *direct route*.

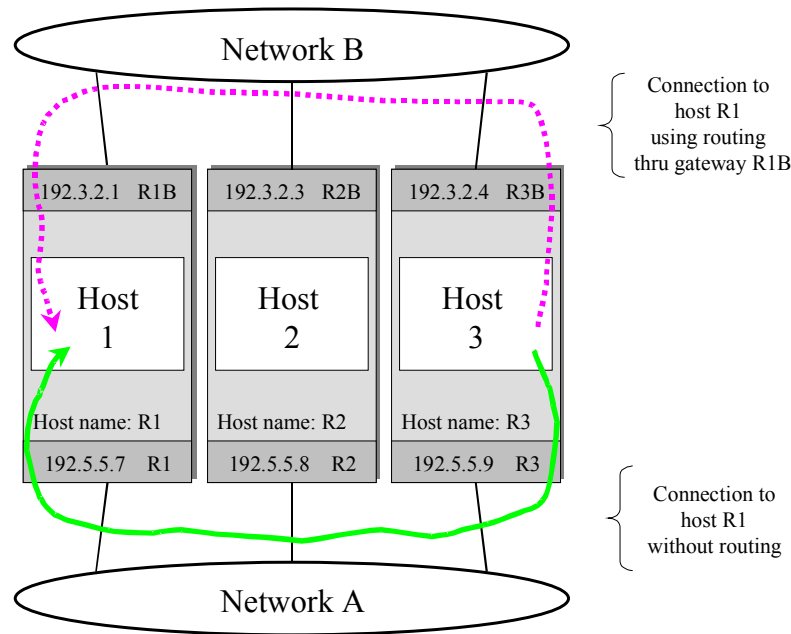
There are three ways of defining the target address as the routing entry:

Basic Principles of TCP/IP Configuration for R/3 Servers

- Complete IP address of the target host (**host route**)
- Only the network portion of the IP address (**network route**)
- Default route (**only one entry**)

Example of a Simple Route

In this example, the three hosts are connected together by network A and network B:



A host is usually addressed by its host name. If Host 3 makes a connection to Host 1, the target address 192.5.5.7 (R1) is addressed. This connection normally goes over network A. If you want the connection to the same target address to run over network B instead, you must have a corresponding routing entry in Host 3. You may want to route the connection over network B, for example, to increase performance (network B is faster than network A). For communication to run in both directions over network B, there must also be a corresponding routing entry for Host 3 on Host 1.

Example of Routing under UNIX

The above example described an indirect host route. The target address is a complete IP address (a host route), and it is also accessed through a gateway using the IP name R1B (an indirect route). On UNIX hosts, you can display the routing table by using the command `netstat -r`. For indirect routes, the flag **G** is used, and for host routes, the flag **H** is used. To create an indirect route, you must specify in the command `route add` a *Metric* (hop count) greater than 0.

Action	Command
Create route on host 1:	<code>route add R3 R3B 1</code>

Basic Principles of TCP/IP Configuration for R/3 Servers

Create route on host 3:	<code>route add R1 R1B 1</code>
Display all static routes:	<code>netstat -r</code>
Test routes:	<code>tracert</code>

Host 3:

Destination	Gateway	Flag
R1	R1B	UGH

Host 1:

Destination	Gateway	Flag
R3	R3B	UGH

Example of Routing Under Windows NT

On Windows NT hosts, you can display the routing table by using the command `route print`. Indirect routes are indicated by a *Metric* = 1 and are made defaults with the command `route add`.

Action	Command
Create route on host 1:	<code>route add R3 R3B 1</code>
Create route on host 3:	<code>route add R1 R1B 1</code>
Display all static routes:	<code>route print</code>
Test routes:	<code>tracert</code>

Host 3:

Network address	Gateway address	Metric
R1	R1B	1

Host 1:

Network address	Gateway address	Metric
R3	R3B	1

Host Names of R/3 Servers

The host name of R/3 servers is used in many places, for example, when installing R/3, in the start scripts, and for communication between servers. You can choose this name if you observe certain rules, which are explained in the following section. You must follow these rules for your SAP System to operate smoothly and efficiently.



Read and take these rules into consideration **before** you install an SAP System.

As a prerequisite, you must have basic knowledge of the TCP/IP protocol.

See also: [Basic Principles of TCP/IP Configuration for R/3 Servers \[Page 30\]](#)

Requirements for Host Names and IP Names

The host name must be assigned **before** you install the SAP System. SAP does not support a name change after installation.

If you use only one network card, the host name must match the IP name that is assigned to this network card.

See also: [Configuring R/3 Servers with Multiple Network Cards \[Page 42\]](#).

Rules for Host Names for R/3 Servers

The host name of an R/3 server is a text string that can consist of letters (a-z), numbers (0-9), and the minus symbol (-). The last character **cannot** be a minus symbol. The host name is not case-sensitive. You cannot use a period (.) as part of the host name; you can only use it in the full qualified domain name.

Up to and including Release 4.5, the host name can have a **maximum length of 8 characters**.

Except for the maximum character length, this definition corresponds to the Internet standards (standards RFC 952 and RFC 1123).

The full qualified domain name (for example, `r3app01.fiboflex.com`) can have a maximum of 60 characters.

Name and Address Resolution for R/3 Servers

Overview

The R/3 System uses IP names and IP addresses to identify R/3 servers and external communication partners. Therefore, the correct configuration for resolving IP names and IP addresses is extremely important for the R/3 System to function properly.

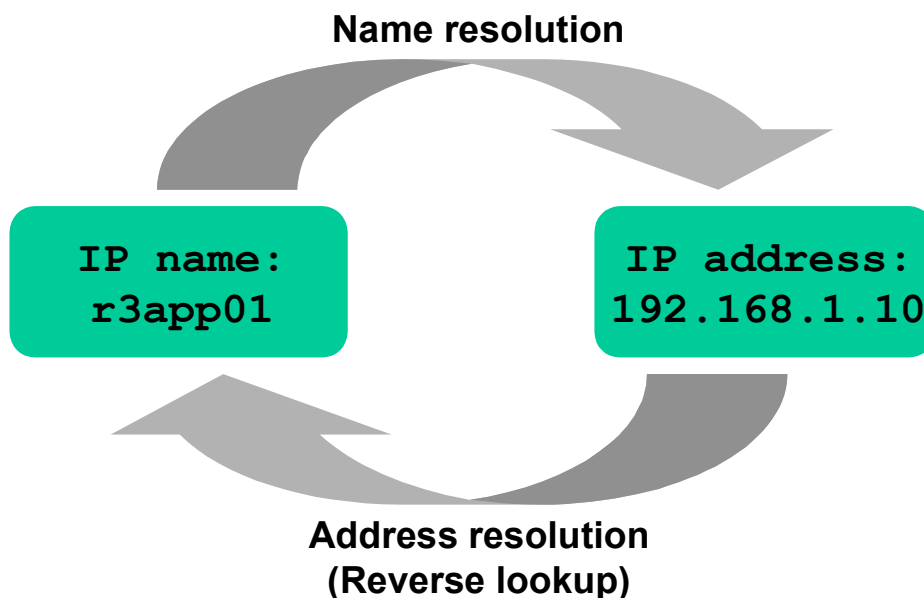
Read and take these rules into consideration **before** installing an R/3 System.

All the operating system platforms supported by SAP offer various mechanisms for name and address resolution (for example, the Hosts file and DNS). These are explained briefly in the second part with respect to the R/3 System.

As a prerequisite, you must have basic knowledge of the TCP/IP protocol. (See also: [Basic Principles of TCP/IP Configuration for R/3 Servers \[Page 30\]](#).)

Terminology

Translating an IP name into an IP address or the reverse, is called **resolution**. This can occur in two directions (see graphic). The most common case involves resolving an IP name into an IP address. For the R/3 System, the opposite resolution is also very important (this is called **reverse lookup**).



Requirements for the R/3 System

Due to the client/server architecture of the R/3 System, there are specific requirements for R/3 servers and R/3 clients when resolving IP names and IP addresses.

R/3 Servers

The sections [Host Names for R/3 Servers \[Page 35\]](#) and [Configuring R/3 Servers with Multiple Network Cards \[Page 42\]](#) list the rules for assigning host names, IP names, and IP addresses for R/3 servers. Follow these rules to ensure smooth operation of the R/3 System.

You must meet the following requirements for your R/3 System to function properly.



- The IP names used to configure the application servers must be able to be resolved into an IP address by all the other application servers in the same R/3 System. The assignment must be identical on all servers.
- Reverse lookup for these IP addresses must be possible and they must return the same IP name. Resolution in the sequence *IP name* → *IP address* → *IP name* must yield the identical name.
- The IP name may not be an alias. On Windows NT, the sequence of the network adapters must be configured correctly in the system.
- The name *local/host* must be able to be resolved (normally, the accompanying IP address is 127.0.0.1).

The following requirements also apply for communication functions:

- RFC or CPI-C client programs connect externally to the R/3 System. In the R/3 System, the IP address of the client host does not have to be resolved. This also applies for RFC server programs that register on the gateway or that start from a SAPgui. One exception is the gateway security functions, which can be activated in the file `secinfo` (for more information, see the R/3 Library under [BC - SAP Communication \[Ext.\]](#)).
- The SAP gateway may start RFC or CPI-C server programs on another host. If this is the case, reverse lookup from the IP address of the host where the server program is started to the corresponding IP name must be possible. If you want to use host names to configure the RFC destinations, you must ensure that the IP name can be resolved.
- You do not need additional name resolution to print on the frontend, since the existing connection between the application server and SAPgui is used. If you want to address a printer or print server using names, these names also have to be resolved on the R/3 server. If you use one or more print servers, the names of the print servers do not have to be resolved by themselves.

In addition to the name resolutions necessary for the system to function, other reverse lookups occur that are not functionally necessary. This type of name resolution occurs, for example, so that the actual names are displayed in monitors and in the system log, instead of the numerical IP addresses.

- When the R/3 System is started, a reverse lookup is performed for the IP addresses of all the network adapters that are built into the application server. The IP addresses of the network adapters from other application servers are also resolved under certain circumstances.

Name and Address Resolution for R/3 Servers

- The R/3 System tries to resolve the IP addresses into IP names for all frontend computers that use SAPgui to log on to the system directly, and for all RFC and CPI-C client computers.

In these cases, the resolution does not have to return a valid result for the R/3 System to function properly. However, if the configuration for resolving names is incorrect (in the operating system or in the name server), there may be long wait times due to timeouts, which can have a highly negative affect on R/3 System performance. Therefore, ensure that the configuration is correct when using name resolution with DNS (Domain Name System).

Buffering Names and Addresses in the R/3 Server

The SAP Network Interface (NI) is a part of the R/3 System, and stores the result of the name resolution and reverse lookups in a special buffer. This means that the operating system only has to perform the name resolution once. Successful resolutions remain in the buffer until the R/3 System is shut down, or until an overflow, or until the buffer is manually deleted (Transaction SM51). Manually deleting the buffer does not include the message server. For this reason, restart the R/3 System to make the changes of IP names or IP addresses effective in the system.

If a resolution fails, this is noted for a duration of 10 minutes (as of R/3 Release 4.0). This is because the name server may not be available. If another attempt is made to resolve the name after this period of time, the R/3 System repeats the operating system call.

The advantage of buffering in the R/3 System is that it increases the performance during name resolution and it only very rarely reaches a critical situation.

R/3 Frontends (SAPgui)

To be able to use all of the R/3 functions, the frontend computer must be able to resolve the names of the message and application servers into IP addresses.

During load balancing, the SAPgui receives a list of IP addresses from the message server. Resolution does not occur.

If the frontend computer cannot resolve the names of the R/3 application servers, you can also start the SAPgui by using numerical IP addresses. However, the functions may become limited, for example, when you use certain desktop components.

External RFC and CPI-C Communication Components

An RFC or CPI-C client establishes a connection to the R/3 System. To do this, it requires the IP address of an application server. You can enter this address directly to establish the connection. In this case, no resolution occurs. You can also use load balancing with logon groups for RFC. Here, the RFC client connects to the message server and receives a list of IP addresses from the server. No resolution occurs here.

Server programs are usually started by an SAP gateway, or by a SAPgui. Each R/3 instance has a gateway. You can also operate an SAP gateway without an R/3 instance (*standalone*). After the server program starts, it must establish a connection with the gateway within a specific period of time. You can start the program in different ways:

Start using a remote shell: On the command line, the server program receives the IP name of the gateway to which it should connect. This name must be able to be resolved.

Start using the SAPgui (only RFC): The command line communicates the information with which the SAPgui was started to the RFC program. It also communicates a SAProuter string is used. This means that the resolution is identical to the resolution with the SAPgui.

Naming Resolution Mechanisms

There are four common mechanisms for assigning names with addresses:

- *Hosts* file
- Domain Name System (DNS)
- Windows Internet Naming Service (WINS)
- Network Information Service (NIS, used to be called *Yellow Pages*)

The operating system determines which mechanisms are available, which are active by default, and how they are configured. The administrator can decide which of the available mechanisms are used. In most operating systems, you can configure several mechanisms with different priorities. These are used one after the other, if the first mechanism does not give you the result you want.

Choosing a Naming Resolution Mechanism

The configuration of the naming resolution mechanism on R/3 servers must meet the following requirements:

- Robustness
- Flexibility
- Maintenance
- Simplicity
- Performance
- Integration in an existing infrastructure

Hosts File

The *Hosts* file contains a list of IP addresses and the accompanying IP names. The resolution occurs by searching through the list. Under UNIX, the file is called `/etc/hosts`, under Windows NT it is called `\winnt\system32\drivers\etc\hosts`.

The *Hosts* file is a robust and simple solution, since all of the information for naming resolution is available on the computer. No network access to other computers is required to resolve names. The advantage is very high performance. However, maintaining the file requires some effort, especially if there are many hosts, or the hosts are divided among many separate networks.

When you maintain the *Hosts* file, you must enter the IP name that corresponds to the host name as the first name directly after the IP address, and not as an alias (second name).

NIS

The Network Information Service (NIS) is a distributed database system that replaces configuration files, which usually have multiple copies, with a central administration. Files such as `/etc/hosts`, `/etc/passwd`, and so on do not have to be administrated in each host. Instead, the information is retrieved as needed from the database.

If you use NIS to administrate the *Hosts* file, only the settings in NIS are valid. The local *Hosts* file is no longer used. This behavior depends, however, on the operating system.

Name and Address Resolution for R/3 Servers

NIS is a centrally administrated system with a master server. To improve availability and performance, you can set up additional slave servers that answer the client queries. Since access to an NIS server is required for each resolution, you must have high availability, short wait times, and you must carefully plan the NIS system for your system to work smoothly.

WINS

WINS (Windows Name Service) is suited for use in a pure Microsoft Windows environment and cannot be directly compared with the other mechanisms described here for naming resolution. One main difference is that no IP names, but NetBIOS names, are mapped to IP addresses. The advantages of WINS are its simplicity, robustness, and flexibility. The disadvantage is that WINS is only suited for smaller, flatly structured and homogeneous Microsoft networks.

DNS

The Domain Name System (DNS) can map complex, hierarchical structures with a large number of hosts. It lets you use efficient centralized and decentralized administration. For example, the entire Internet uses a single DNS infrastructure for naming resolution. DNS is the most robust of all methods of naming resolution, and is the most well suited for the future, for example, for potential integration with directory services. DNS is available on all server and frontend platforms supported by SAP.

With regards to availability and wait times, the same applies here as it does for NIS. With DNS, planning is particularly important, since configuring and maintaining DNS is complex and, therefore, prone to errors. The most dangerous errors are not those that lead to a total failure, rather those that lead to performance problems, because they remain undiscovered.

Tips for DNS Configuration

The effects of an incorrect DNS configuration are explained in the following example:

If a reverse lookup in the DNS is not maintained, and the name server in the corporate internal network is configured incorrectly so that it accesses the central name server in the Internet for unknown domains (this is the default setting for certain name servers), the name server waits for several minutes for a response from the inaccessible Internet name server. During this time, the R/3 work process from the operating system is stopped and the user cannot continue working.

Wait times can also occur due to connection problems between the application server and the name server, or between two name servers.

If you want to use DNS (you need some basic knowledge of DNS):

- You need at least one backup name server. DNS contains methods for replicating the configuration data. In all operating systems, backup name servers can be configured that are automatically used if the first server does not respond within a specific period of time.
- Ensure that there is high availability for the DNS server and for the network connection between the DNS servers and to the SAP servers.
- Ensure that all the domains in your company are entered in the DNS. To do this, you must create a zone file for all name domains (for example, `accounting.fiboflex.com`), and set up a name server for this zone as the primary name server. You also need a zone file for all network domains (for example, `192.168.1`) that resolves IP addresses into IP names (reverse lookup).
- You do not have to enter all your corporate computers in the zone files, if the information is not needed for operating your R/3 System. It is only important that the zones exist. The

Name and Address Resolution for R/3 Servers

above sections have already described which name resolution is required to operate your R/3 System. If, for example, you do not need naming resolution from client computers, the zone files for the network domain and naming domain of your frontend network can remain empty.

- Ensure that queries for internal names or addresses are never forwarded to external name servers (for example, in the Internet). This is guaranteed if root name servers only are contained in the DNS cache files that are located in your local network.

Conclusion

Deciding on which mechanisms you should use for naming resolution depends on your priorities. When making your decision, remember that the name resolution is a critical function in the R/3 System that influences the system operation.

WINS is only used in pure Windows environments. In the future, it will not be supported in certain circumstances, and we therefore do not recommend it.

DNS is the most flexible and powerful solution, almost a standard. However, its use requires a high level of knowledge about implementation and maintenance. Buffering in the R/3 System means that wait times are not critical, within certain limits. However, you must have high availability for the DNS server and the related network connections. If you already have a functioning and well-maintained DNS infrastructure, we recommend using DNS on the R/3 servers and frontends.

Simplicity and robustness are the advantages of the *Hosts* file. It is therefore highly recommended for the R/3 servers. Generally, the R/3 server does not have to use the same mechanism as in the rest of your corporate network. However, maintaining and distributing the *Hosts* file may require a great deal of effort to administrate.

On UNIX servers, you can replace the *Hosts* file by NIS. Here, like DNS, you must pay attention to proper implementation and high availability of the NIS server to ensure that the R/3 System operates properly.

Configuring R/3 Servers with Multiple Network Cards

Configuring R/3 Servers with Multiple Network Cards

Assigning Server Services to Network Addresses

In a client/server environment, the clients use the network addresses or IP names to access the server services. This means that a server service is assigned to a particular network address.

If a server service runs on a multihomed host, this service can be accessed through several network addresses. Since this scenario is no longer a unique assignment of a server service to one network address, this can lead to problems if the clients access it using different network addresses from different subnets.

Static Assignment

With static assignment, a client always uses the same network to address the server. This address is usually determined when installing the client.

Dynamic Assignment

If multiple identical server services exist, static assignment of the clients to one server service is not ideal. In this case, dynamic assignment of the clients to the server services is better. For this dynamic assignment, criteria such as performance, load balancing, availability, and organizational assignment all play a part. A higher-level instance takes over the dynamic assignment of clients for a specific server service, or gives information about which network addresses can be used to access a specific server service.

Switchover Environments

Switchover is a process where server applications are protected against failure by being forwarded to another standby host if one computer fails. For the switchover to remain transparent to the clients, the server must be able to be accessed by the clients in the same way after switching over to another host. To do this, a virtual IP address and a virtual IP name are assigned to the network adapter, in addition to the normal IP address and IP name for each server application. If one host fails, this virtual IP address switches with the server application to the standby host. Therefore, the switchover of a server application from one host to another remains transparent for all clients that use the virtual IP name to access the server application.

The SAP System

In the R/3 System, the message server knows the active R/3 server services or instances, and the network addresses through which these services can be accessed. The system knows only one network address for each instance, the address that corresponds to the host name of the host where the R/3 server service runs. This means that each client must access its server service through the host name. A client in this form is not only a SAPgui frontend, but also an R/3 application server that requests a service from another R/3 server. This accessibility over the host name is important if the R/3 servers work with multiple network adapters and a separate server network.

Network Setup for the R/3 System

Network Setup Guidelines

When setting up your network configuration, follow these guidelines to make sure that your system runs optimally in the network:

1. Each R/3 application server must have at least one network adapter that all the other R/3 application servers in the same R/3 System, and all frontend computers and other communication partners can access. Configure your network to meet this requirement.
2. The IP name of this network adapter (according to point 1) must match the host name (which you can find out by using the command `hostname`). For high availability configurations with switchover, special rules apply (see *Switchover Environments*).
3. To configure the R/3 System, the frontends and all communication interfaces only use the host name of the R/3 application server. Do not use a different IP name, such as another network adapter, or an alias in a table or in a configuration file of the R/3 System. This ensures that only that name is used whose accompanying IP address can be accessed by any communication partner.
4. If the R/3 servers are interconnected by a separate, fast server network, static IP routing entries must ensure that the data traffic between the servers is sent through this network.



For network configurations that do not follow these guidelines, R/3 Basis functions such as system monitoring, CPI-C, RFC communication and SAPlogon, will not function, even if the R/3 System functions partially. In this case, SAP cannot guarantee the functioning of the R/3 System. Further developments in future SAP Releases may lead to other problems if you did not follow the SAP guidelines

Procedure for Configuring Your Network

1. Connect the R/3 application servers through the access network.
2. Set the host names for all R/3 application servers to the IP name of the network adapter that is connected to the access network. To find the restrictions for choosing the host name and the mechanisms for assigning the host name (for example, Hosts file, DNS), see [Name and Address Resolution for R/3 Servers \[Page 36\]](#).
3. Configure your computers according to the relevant guideline **before** installing the R/3 System, since the host names from the installation procedure are used automatically for various parameters. SAP does not support a name change after installation.
4. Only this host name is used for the R/3 servers when you configure your R/3 System.
5. If the R/3 servers are interconnected through a server network to distribute the network load, the data traffic between R/3 servers must be routed over the server network using static IP routing entries.

Conclusions for Network Topology

To achieve a stable and, above all, easy-to-manage network, choose the most simple and clearly divided network topologies.

Examples of R/3 Configurations

The following two sections contain recommendations on simplifying the following configurations:

- One network adapter for each R/3 server
- Two network adapters for each R/3 server

One Network Adapter for each Server

Only one network adapter and only one IP address is used for each server for communication between the SAP components (communication between the R/3 servers, and with the frontends).

In this case, the host name must be identical to the IP name that belongs to the IP address of the network adapter.

The network must allow for direct communication between all the R/3 servers using these selected network adapters (database servers and application servers).

In this case, no special routing entries are required.

Two Network Adapters for each Server (Multihomed Host)

For larger R/3 Systems, the data traffic must be divided into multiple network segments. The data traffic between the R/3 database and application servers is a particularly heavy load. This server network must have as high a bandwidth possible, and must be separated from the remaining network traffic either physically or by using a router.

To do this, the R/3 servers need a second network adapter that is only used for communication in the server network. This network adapter handles the R/3 data traffic between the R/3 servers only; it does not handle communication to the SAPgui frontends, or data backups. The host name of the server must match the IP name of the network adapter used for communicating with the frontends.

The data traffic between the R/3 servers (database server, central instance and dialog instances) is routed through the relevant static IP routes over the server network.

Two network adapters for each server are sufficient for most of your network communication needs.



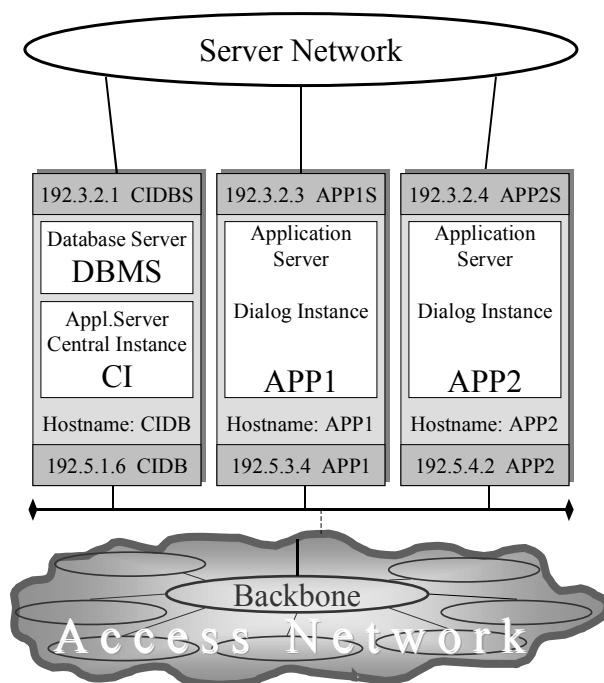
You can use more than two network adapters for each server (for example, for data backup). However, you must follow the guidelines above for setting up your network.

Examples of R/3 Configurations

Database (DB) with Central Instance (CI)

The following example shows the simplest variation of a distributed R/3 System. The central system, consisting of a central instance and a database on one host, is the core of the R/3 System. You can add as many additional application servers as you want. In these R/3 Systems, the communication of the servers among themselves (DB with APP and APP with APP) makes up the majority of the network load. Therefore, a separate high speed network (the server network) is used for server communication.

Examples of R/3 Configurations



As described in guideline 3 for setting up your network, the host name is always used for accessing the R/3 System. This applies to the connections of SAPgui to the central instance or to the application server, and also among the application servers, and for the application server to the database. To simplify the configuration, the host name must be identical to the name of the network adapter that the frontends can access over the access network. The servers also use this host name to address each other. To route network traffic between the servers over the server network, you have to use static IP routes on all the servers.

Example of routing entries (indirect host routes with UGH flags) on all servers:

<u>CIDB</u>		
Destination	Gateway	Flags
APP1	APP1S	UGH

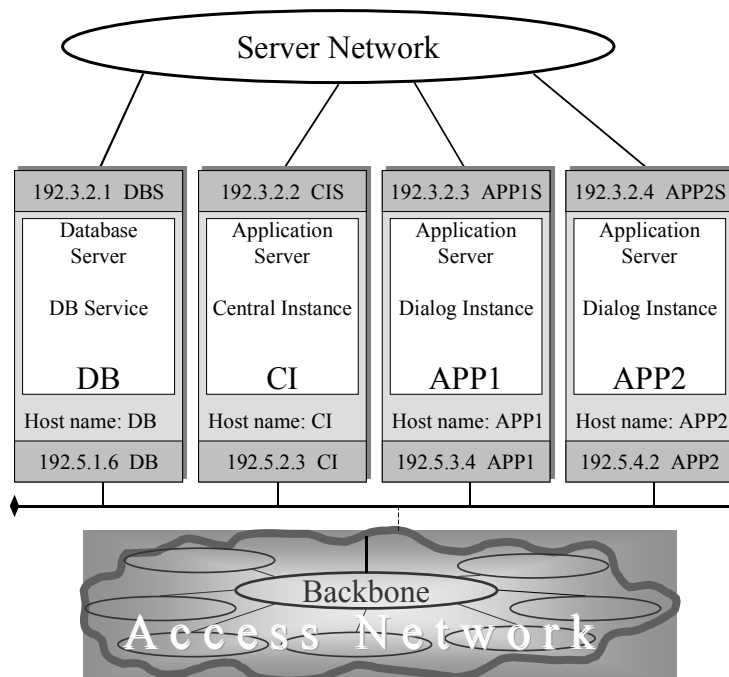
<u>APP1</u>		
Destination	Gateway	Flags
CIDB	CIDBS	UGH

<u>APP2</u>		
Destination	Gateway	Flags
CIDB	CIDBS	UGH

Database (DB) Separate from the Central Instance (CI)

For large R/3 Systems with high throughput requirements, you must distribute the central instance and the database over separate hosts. Here as well, you must ensure that the entire data traffic between the database and application servers passes over the server network. As you can see in the graphic, the server network is a separate subnet that only connects the R/3 System servers with each other, and ensures that no external data traffic, and particularly no broadcasts, interfere with the data traffic between the R/3 servers.

Examples of R/3 Configurations



This configuration lets you meet guideline 2 most easily, because the host name of all R/3 servers points to the network adapter that the frontends can access over the access network.

To route the data traffic among the servers over the server network when addressing using host names, the corresponding static IP routes must be set up on all servers.

Example of routing entries (indirect host routes with UGH flags) on all servers:

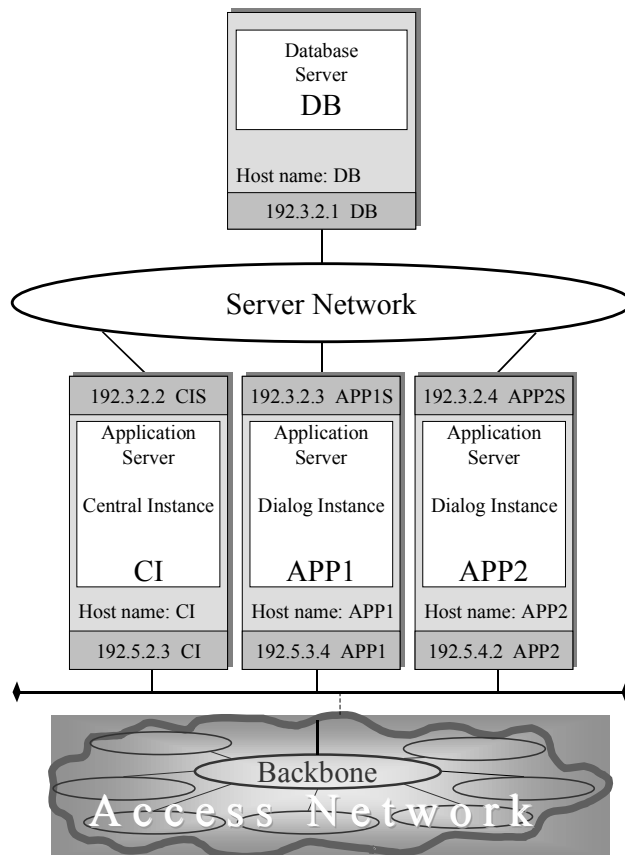
<u>DB</u>			<u>C1</u>			<u>APP1</u>		
Destination	Gateway	Flags	Destination	Gateway	Flags	Destination	Gateway	Flags
CI	CIS	UGH	DB	DBS	UGH	DB	DBS	UGH
APP1	APP1S	UGH	APP1	APP1S	UGH	CI	CIS	UGH
APP2	APP2S	UGH	APP2		UGH	APP2	APP2S	UGH

The same applies for additional application servers.

Isolated Database Server

The following example shows a database server that is only connected to the server network (isolated). No R/3 applications run on this host (frontends have no access). The advantage of this configuration is the physical isolation of the database server. This server, on which all the data is located and which is essential for the R/3 System to operate, cannot be accessed by each member in the network. This prevents unwanted access to the data, maintains performance and the ability of this important host to continue running in a simple and clear way. This increases the security and availability of the entire R/3 System.

Examples of R/3 Configurations



In this configuration, you can access the database from other R/3 servers directly (without routing) through its host name. Since the database server only has one network adapter, this ensures that the communication with the database server is handled by the server network. To ensure that the data traffic between the R/3 servers (central instance and application server) also occurs on the server network, you must set up the corresponding routes on these servers.

Example of routing entries (indirect host routes with flags UGH) on the database server:

DB		
Destination	Gateway	Flags
CI	CIS	UGH
APP1	APP1S	UGH
APP2	APP2S	UGH

Example of routing entries for a central instance and application server:

CI		
Destination	Gateway	Flags

APP1		
Destination	Gateway	Flags

APP2		
Destination	Gateway	Flags

Examples of R/3 Configurations

APP1	APP1S	UGH
APP2	APP2S	UGH

CI	CIS	UGH
APP2	APP2S	UGH

CI	CIS	UGH
APP1	APP1S	UGH

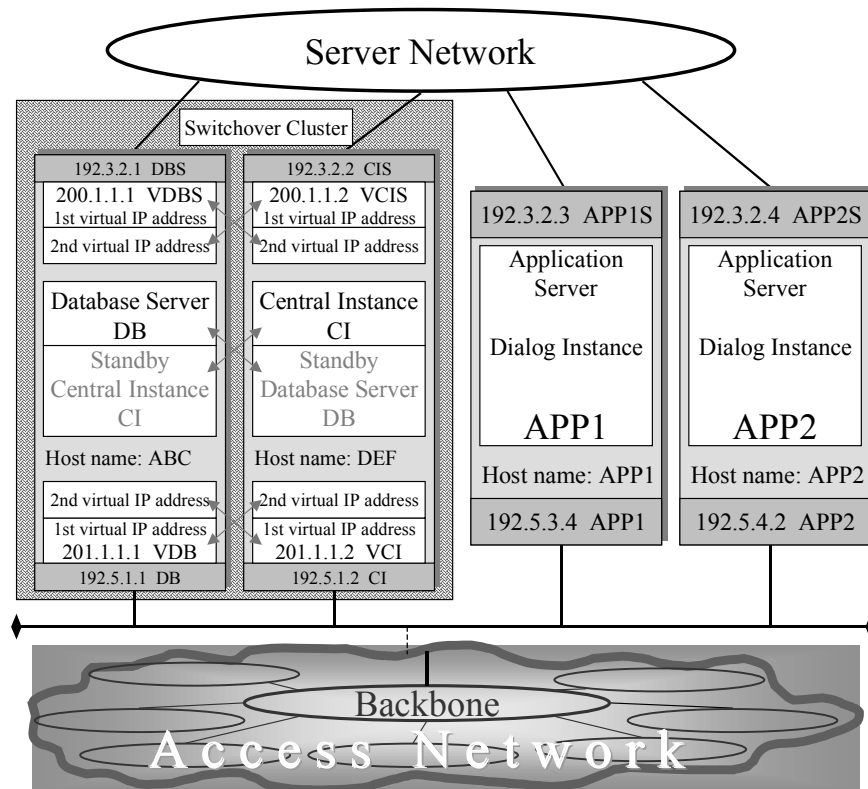
Switchover Environments

To ensure high availability, various switchover products are offered for UNIX and NT platforms. There are two procedures for this:

- **Identity takeover**
In the identity takeover procedure, a standby host completely takes over the identity and task of the failed host. In this case, no particular SAP configuration is necessary, since the standby host is identical to the failed host after the switchover.
- **Virtual IP address takeover**
In the virtual IP address takeover, each network adapter also has a virtual IP address and a virtual IP name, in addition to the normal IP address and the accompanying IP name. During a switchover, another host takes over the failed virtual IP address and virtual IP name. For this virtual IP address takeover to function with R/3 software, you may only use the virtual IP name in the configuration, which can be taken over by another host. The advantage of this procedure is that an empty standby host is not needed; instead the failed service (indicated by the virtual IP name) is taken over by another host in addition to its normal functions.

In the following example, the database server (DB) and the central instance (CI) are protected from each other. For example, if the central instance fails, the database server also takes over the functions of the central instance and also the virtual addresses VCI (access network) and VCIS (server network). For all the application servers and frontends involved, the central instance remains accessible through the names VCI or VCIS. However, the central instance does not run on the host with the host name DEF; instead it runs on the host with the host name ABC. Therefore, you must set the local host name that is valid for the R/3 System by using the SAP profile parameter `SAPLOCALHOST`. You determine the name by which the database server can be accessed by using the parameter `SAPDBHOST`.

Do not use the following example as a standard solution for high-availability R/3 Systems. It only serves as an example to explain the effects of an IP address takeover on the network configuration.



For more detailed information on high availability, see the documentation *SAP R/3 in Switchover Environments*.

Frontend Configuration

In order for the frontends to access their dialog instances in the same way after the switchover (or, if using SAPlogon, the message server), you can use the virtual IP name only. This name must exist in the file `/etc/hosts` or in the DNS.

Server Configuration

In all profile parameters containing the host names of servers, you may only use the virtual IP name. Also, the instance names contain host names that may only be virtual IP names.

Example of the Most Important Parameters:

DEFAULT.PFL	SAPSYSTEMNAME = C11 (SID for example, C11 and the instance number 00)
DEFAULT.PFL	SAPDBHOST = VDB
DEFAULT.PFL	rdisp/mshost = VCI
DEFAULT.PFL	rdisp/vbname = VCI_C11_00

Examples of R/3 Configurations

DEFAULT.PFL	rdisp/enqname = VCI_C11_00
DEFAULT.PFL	rdisp/btcname = VCI_C11_00
Instance CI	SAPLOCALHOST = VCI
Instance CI	SAPLOCALHOSTFULL = fully qualified domain name from VCI (only required for NT)
Instance APP1	No special entries

Example of Static IP Routes:

This configuration lets you meet guideline 2 most easily, because the host name of all R/3 application servers are identical to the network adapter that the frontends can access over the access network.

To route the data traffic in the server over server network when addressing using host names, you must set up the corresponding static IP routes on all servers. Note that with the hosts belonging to the switchover cluster, you must use the the virtual IP names, or use the profile parameter SAPLOCALHOST to set the virtual IP name.

Example of Routing Entries (Indirect Host Routes with UGH Flags) on All Servers:

DB			C1			APP1		
Destination	Gateway	Flags	Destination	Gateway	Flags	Destination	Gateway	Flags
VCI	VCIS	UGH	VDB	VDBS	UGH	VDB	VDBS	UGH
APP1	APP1S	UGH	APP1	APP1S	UGH	VCI	VCIS	UGH
APP2	APP2S	UGH	APP2	APP2S	UGH	APP2	APP2S	UGH

Same for APP2

Network Integration of SAP Frontends

SAP Frontend Communication

SAP Frontend Communication

The SAPgui is the user interface of the SAP System and is the presentation level in the 3-tiered client-server architecture of the SAP System. It displays the user interface and the interaction of the users with the system. The SAPgui is a graphical, window-based program that you control using a keyboard and mouse. It runs on a PC that is linked by a network to the SAP server computers.

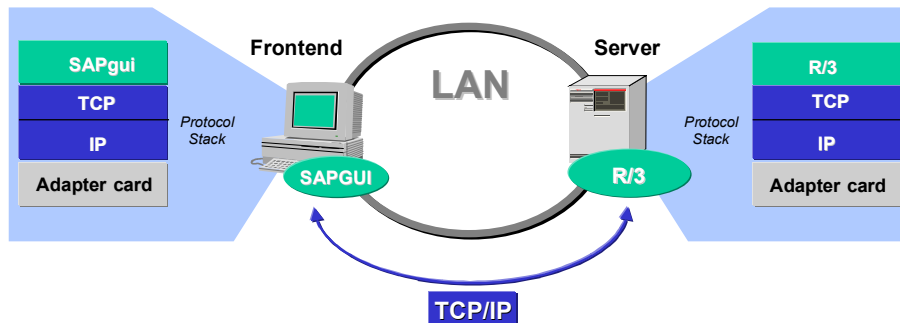
The following sections explain the attributes and requirements of network communication for the SAPgui. The network load arising from the SAPgui is described under [Frontend Network Load \[Page 140\]](#).

SAP offers the SAPgui on various platforms: Microsoft Windows, OS/2, Unix and Macintosh. All these variants have the same network requirements. In addition to the SAPgui, there is a SAP frontend written in Java and a version that lets you operate the SAP System using a Web browser. These frontend types have other network attributes and are not described here. For more information on Web applications, see [Internet Transaction Server Technology \[Page 80\]](#).

Accessing the online documentation is naturally a part of working with the SAP Frontend, but is separately implemented on a technical level. For more information, see [SAP Online Documentation \[Page 61\]](#).

Network Protocol

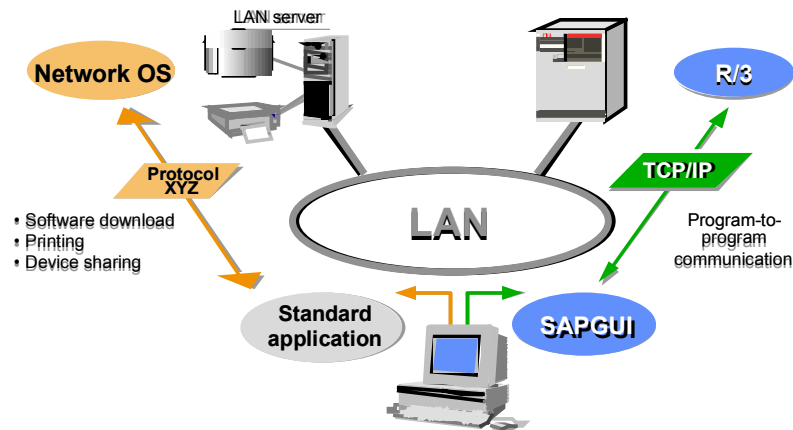
The TCP/IP protocol is used exclusively to communicate between the SAP System and the SAPgui. The following graphic displays the relation between the frontend as the client and the SAP System as the server:



Additional proprietary protocols (for example, IPX, NetBEUI) are often used in PC networks. The frontend computer is often equipped with two protocols, which use the same network adapter on

SAP Frontend Communication

the PC. Two protocols coexisting on one computer do not hinder the operation of the SAPgui. However, you must make sure that there are no conflicts between the different software components. The following graphic displays using 2 different protocols on one frontend computer.



The TCP/IP protocol has established itself as the standard network protocol. Therefore, it is already integrated in all modern operating systems. The operating systems supported by SAPgui with a specific R/3 Release and TCP/IP stacks, which are the concrete implementation of the protocol on a computer platform, are listed in the brochure *SAP System Requirements*. You can find this in SAPNet under <http://sapnet.sap-ag.de/SSR>. The hardware used (for example, token ring- or Ethernet-adapter) is determined not by SAP software, rather the operating system and the TCP/IP stack.

Attributes of SAPgui Communication

TCP Ports

For more information in the TCP ports used by SAPgui, see [Communication Connections of the R/3 System \[Page 13\]](#). The following table is a quick overview of the most important ports that are used when working with a SAPgui:

Connection	Symbolic port name	Example: <nn> = 01
SAPgui – application server (dispatcher)	sapdp<nn> ¹⁾	3201
SAPgui – message server (load distribution)	sapms<SID> ²⁾	3600 ³⁾
RFC program – application server (gateway)	sapgw<nn>	3301
As required – SAProuter	sapdp99 ⁴⁾	3299 ⁴⁾

SAP Frontend Communication

- 1) **nn** is the instance number of the SAP Application server.
- 2) **SID** is the 3-character system ID of the SAP System (for example, PRD).
- 3) Can be chosen at random, but must be defined in `/etc/services`. Another port is often chosen for each SAP System, but this is not necessary.
- 4) The default is 3299, but you have a free choice.

You can use the SAProuter to route all communication between the frontend and the SAP System through a port to a computer. For more information on the SAProuter, see the SAP Library under *BC - SAProuter*.

Establishing a Connection

All connections are established from the frontend, and never from the SAP Server.

If you use the message server for load-balancing, the SAPgui (or the SAPLogon for R/3 Releases before 4.0) first makes a TCP connection to the Message server at logon to determine the most suitable application server. The connection can occur with one or several SAProuters; in this case, the TCP connection is only made to the SAProuter, which in turn connects to the next communication partner. The connection to the message server is subject to a timeout that by default takes 10 seconds. You can configure the timeout in the SAPLogon options directly and in the SAPgui using the environment variable `TDW_TIMEOUT`.

At logon, the SAPgui establishes a TCP connection to the dispatcher on the application server, that can also connect over the SAProuter, just like the connection to the message server. There is also a timeout with this connection with a default of 10 seconds. If this time is exceeded, the SAPgui reports an error and terminates the connection. You can configure the timeout using the environment variable `TDW_TIMEOUT`.

All modes that are opened during a session use the same TCP connection. To do this, the data flow of the various modes is sent through a multiplexer to the SAP System.

Desktop components such as F1 help or Screen Painter communicate with the SAP System using RFC. To do this, they connect to the SAP gateway on the same application server with which the SAPgui is also connected.

Duration of the Connection

The SAPgui uses a TCP connection for the entire time that the user is logged on to the SAP System. Therefore, it is not necessary for each dialog step to establish a new connection. The SAP System is also informed if the user ends the session with SAPgui without logging off from the system. In this case, the user is automatically logged off from the system.

If the connection between the SAPgui and the application server is interrupted for a long period of time, the operating system (more specifically, the TCP/IP stack) terminates the connection and signals this to both programs that are using the connection. In this case, the SAPgui reports an error and closes itself. The dispatcher makes an entry in the system log and ends the user session, which results in all the resources that were used by the session, are released after a short waiting period. During this wait time the user can log on again use the previous context of the session.

The SAPgui therefore requires a stable network connection during the entire duration of the user session.

Keep-Alive

If the user's SAPgui session ends, the SAP System must be informed as quickly as possible to release the resources that were used by this session. It is important that data records are released that are being processed by the session and are therefore locked.

When the TCP connection is ended, the system is automatically informed if the SAPgui process is terminated on the frontend. However, the operating system continues to run. The situation is different if the frontend computer is turned off, or fails, or the network connection is broken. Since the SAP System normally waits for queries from the SAPgui and does not send data by itself to the frontend, it is not informed about this situation, because no error has occurred. The user session would remain in the system until the auto-logout is triggered (profile parameter `rdisp/gui_auto_logout`) or it is manually deleted.

To avoid this situation, the application server sends a network packet to the SAPgui, if it has not sent any data for a specific period of time. The application server expects an answer from the SAPgui within 40 seconds, otherwise the user session ends. This procedure is repeated periodically.

The wait time is 1200 seconds by default and you can configure this time using the profile parameter `rdisp/keepalive`. A value of 0 deactivates the mechanism, which we only recommend in conjunction with a relatively short auto-logout time.

Network Address Translation

Network Address Translation (NAT) modifies the IP addresses and if necessary the TCP port numbers in the network packets. This occurs transparently for the entire network protocol stack. NAT reduces official IP addresses when many computers communicate over the Internet, and it connects communication partners that cannot otherwise be reached directly due to address conflicts.

When using Network Address Translation with the SAP System, there are several limitations which arise from network addresses being transferred within the transaction data. There are two possibilities:

- The addresses from SAP server computers to the SAPgui may **not** be modified.
- The addresses from SAP frontend computers to the SAP server may be modified.

If you use the SAP message server for load-balancing, it sends the IP addresses and TCP services of the available SAP server back to the GUI. These IP addresses are then used to connect to the most suitable server. This procedure makes it necessary for the frontend computers to be able to reach the application server under the IP address that they have in their local network. (For the rules on assigning these IP addresses, see [Configuring SAP Servers with Multiple Network Cards \[Page 42\]](#).) This is why the addresses of the SAP server may not be translated on the transmission path.

In certain cases, due to the network topology, the frontend computer may not be able to establish a direct TCP/IP connection to the SAP server without Network Address Translation. In this case, you can use a SAProuter to establish the connection. For more information see [Communication Using SAProuter \[Ext.\]](#) and in the SAP Library under *BC - SAProuter*.

Network Address Translation is mostly used, however, in the other communication direction, which means for the network address of the frontend computers. This does not cause any problems for SAP communication, if there are no server services running on the frontend computer that require a connection from the server to the frontend. For example, you can start RFC server programs from SAPgui, as it is used for many desktop components. Frontend- printing

SAP Frontend Communication

is also possible, but you cannot use the computer as the print server in the SAP printer administration.

Dynamic Host Configuration Protocol

The *Dynamic Host Configuration Protocol* (DHCP) automatically configures the network configuration of computers that are linked to a network. This is used primarily for administrating workstation computers in local networks and for dial-in connections. Using DHCP can lead to the same computer is assigned a different IP address after each restart or after each logon to the network.

You can generally operate SAP frontends with DHCP and changing IP addresses if there are no server services running (in the same way as mentioned in the previous section). Buffering name and address resolution in the SAP server (see [Name and Address Resolution for R/3 Servers \[Page 36\]](#)) has consequences when operating frontend computers with DHCP:

- The SAP System resolves IP addresses into names when you log on. This name resolution is buffered in the SAP Server. If another frontend logs on with the same IP address, the old name (now incorrect) is still stored in the system. Therefore, the incorrect name is entered in some log files. However, in the User Overview (Transaction *SM04*) the correct name is displayed, which is sent to the application server when logging on from the SAPgui.
- For the SAP printing administration, it is not useful to specify a computer with a changing IP address as the print server. This also applies if your name resolution is able to map the host name correctly to the changing IP addresses (for example, with dynamic DNS). The reason is that the name resolution by the SAP server is performed only once and then buffered. New accesses to the same host names do not result in new name resolution, so that the print request is not sent to the correct computer. Buffering in the SAP System cannot be deactivated.

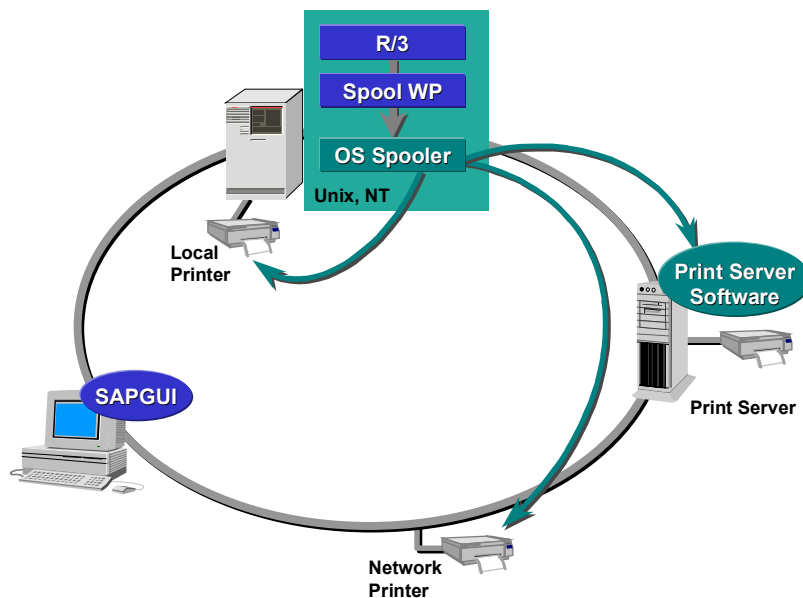
Printing

This section explains the different ways you can print with the R/3 System. In the R/3 System, a document is readied for printing using a special spool work process, which then passes the data to the spooler of the local computer. You can also access a print server using the network or you can print directly from the SAPgui.

Local Printing

When you print locally, the R/3 spool process sends the data to a printer that is set up on the operating system of the application server. The print request is then forwarded from the spool system of the operating system to the relevant printer. You can connect the printer locally to the computer or you can access it over the network.

Local printing is easy to implement and is also the fastest method to transmit data from the R/3 spool system to the printer. This is because there is no network connection between the R/3 spool process and the printer spooler, and the printer spooler takes over all communication tasks.



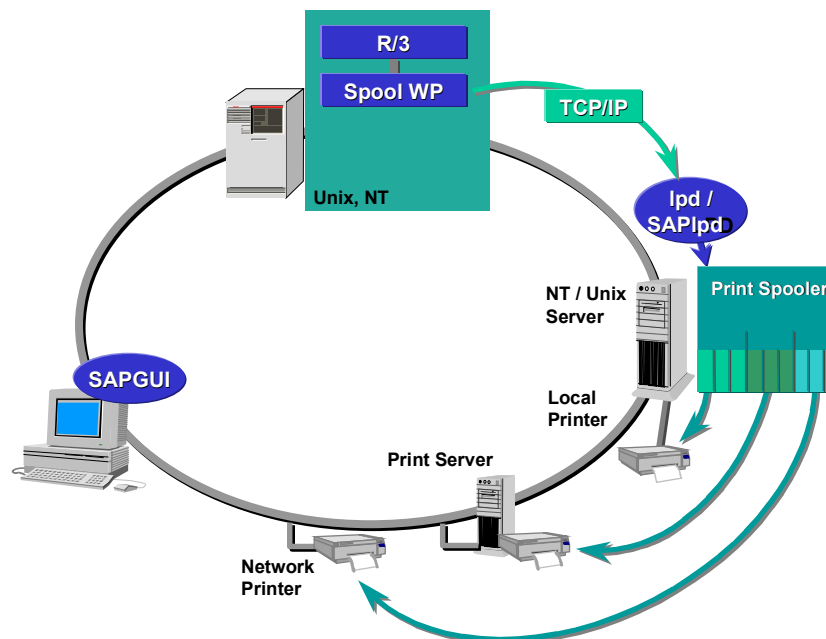
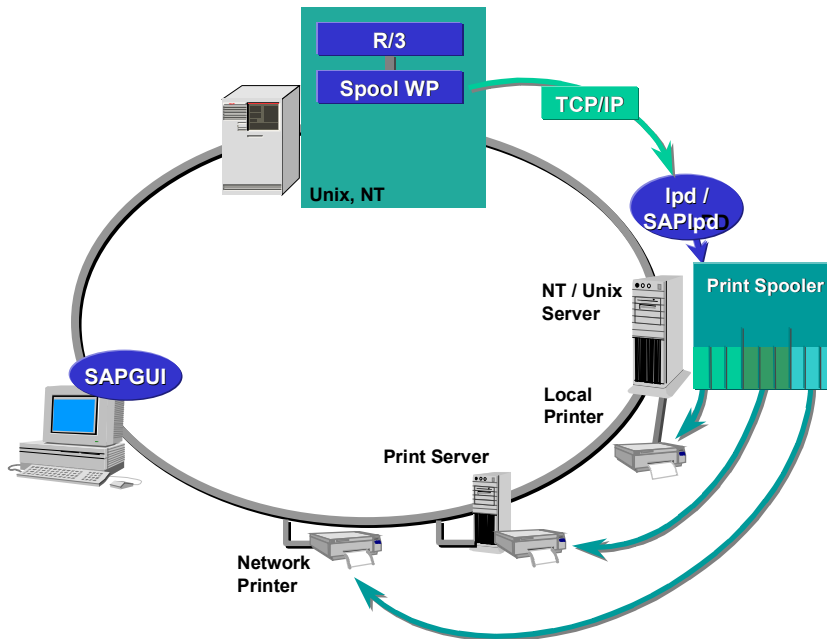
Remote Printing

The R/3 spool process can access any printing system in the network that can handle the *lpr/lpd* protocol (*line printer requester/ line printer daemon*). This protocol originates from BSD UNIX and is an industry standard for controlling printing, and is supported by many modern operating systems and by network-enabled printers.

The line printer requester (*lpr*) is a client and sends the print data to the server (*lpd* side). Various data formats, for example, PostScript or PCL can be transmitted using this protocol.

Printing

Windows PCs do not usually have a line printer daemon (*lpd*). To be able to print using *lpr/lpd*, SAP developed its own *lpd* program for PCs called *SAPlpd*, which is installed with the SAP frontend. *SAPlpd* contains some extensions of the *lpd* protocol, for example, data compression, encryption using the SAP standard SNC (*Secure Network Communication*), and data transmission using the *SAProuter*.

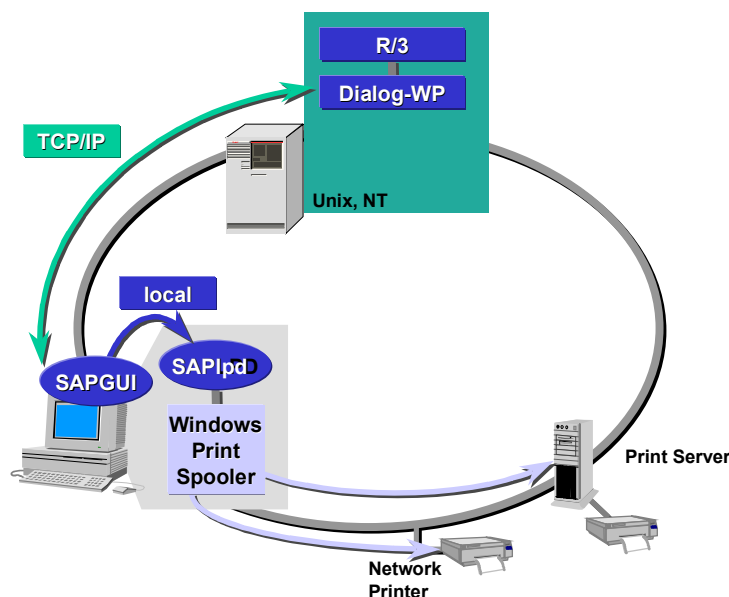


Frontend Printing

Another option when printing from the R/3 System is frontend printing, which lets you access printers that are not defined in the R/3 spool system.

The technical aspects of frontend printing have changed often. The following describes the most up-to-date variant. We recommend that you use this variant as of Release 4.5. In this variant, the output data is transmitted directly over the SAPgui to the frontend PC. To do this, the existing connection between the SAPgui and the application server is used, so that an additional network connection is not required. The SAPgui starts SAPIpd on the frontend PC and transmits the print data to it. SAPIpd then sends the data to the Windows standard printer or to another printer installed on the operating system.

Unlike normal printing, the resolution of the document with frontend printing occurs in the dialog work process of the user. For this reason, frontend printing is only suitable for occasional interactive printing of small documents.



Integrating the Existing Printing Infrastructure

In a network with frontends, a printing infrastructure already exists in which you control the printers by using the network server or specially installed print servers. The existing network printers can also be used for printing from the SAP System.

The integration options depend on the system that the print server makes available. You can easily integrate the printing infrastructure into existing UNIX or Windows NT environments by using remote or frontend printing.

You can also integrate Novell environments by using an *IP print gateway*.

Printing

SAP Online Documentation

Overview

As of R/3 Release 4.0, SAP delivers online documentation in HTML Help format, which is displayed on the frontends using a web browser or the Microsoft HTML Help Viewer.

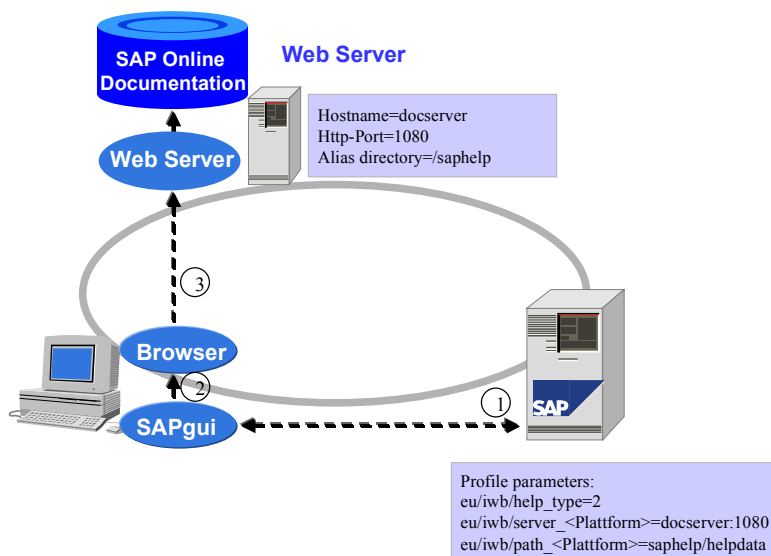
The online documentation is offered in three different help types in order to meet the different requirements of frontend platforms. They differ in file format (HTML or compressed HTML) and in how you access the documents (Web server or file server).

You must specify the help type in the R/3 System by using the profile parameter `eu/iwb/help_type`. You can do this in the global profile, or for each application server in the instance profile. You require additional parameters to configure the access to the documentation for each frontend platform you use: `eu/iwb/*`. You can also make these settings locally on the frontend computer. For more information, see the installation guide *Installing Online Documentation*.

The browsers and HTML Help Viewers supported by SAP are listed in the brochure *SAP System Requirements*. This brochure is in SAPNet under <http://sapnet.sap-ag.de/SSR>.

PlainHtmlHttp

In the PlainHtmlHttp help type, the documents are stored as normal HTML files and are displayed using a web server. You can use PlainHtmlHttp on all supported frontend platforms. As a prerequisite, you need to install a web browser on the frontend.



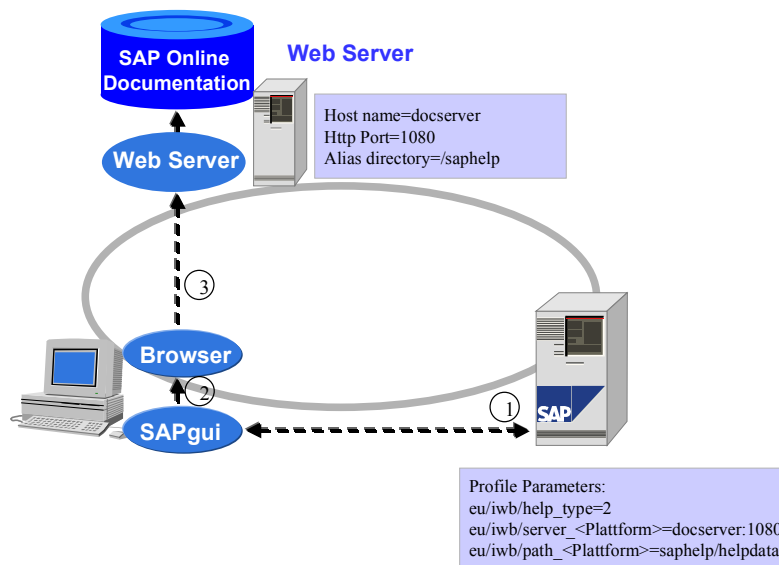
- (1) Application Help is called from the SAPgui
- (2) Browser is started

SAP Online Documentation

- (3) Online documentation is accessed under the URL `http://docserver:1080/saphelp/helpdata/...`. The network protocol used to access the web server is HTTP through TCP/IP.

PlainHtmlFile

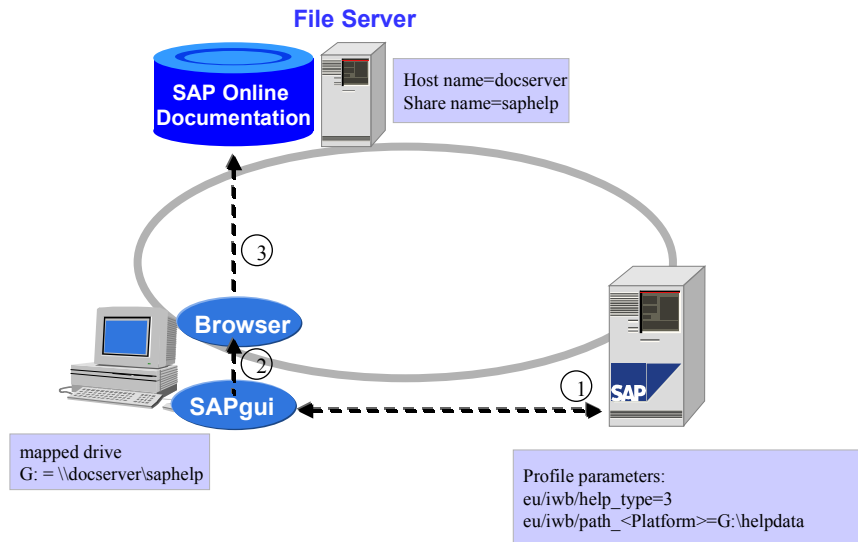
The help documents are also stored as normal HTML files here, however they are accessed from a file server. The file server must allow access to the files from the corresponding frontend platform. Your frontend must also support a web browser.



- (1) Application Help is called from the SAPgui
- (2) Browser is started
- (3) Online documentation is accessed from the path `G:\helpdata\...`. The network protocol that is used to access the files depends on the type of file server. It could be TCP/IP, Novell IPX/SPX or NetBEUI.

HtmlHelpFile

In this help type, the documents are stored in a "Compressed HTML" (CHM) format. CHM is a format developed by Microsoft for storing HTML files in a compressed format. The document is displayed in the HTML Help runtime environment that replaces Microsoft WinHelp. HTML Help is currently available only for Windows NT 4.0, Windows 95 and Windows 98.



- (1) Application Help is called from the SAPgui
- (2) The HTML Help Viewer is started
- (3) The online documentation is accessed under the path
`\\docserver\docu\helpdata\...`. The network protocol that is used to access the files depends on the type of file server. It could be TCP/IP, Novell IPX/SPX or NetBEUI.

Local Help Settings on Windows 32-Bit Frontends

You can override the settings locally for the online help on frontends with Microsoft Windows 95/98 and Windows NT 4.0. These settings are determined in the profile parameters of the R/3 System. The information required for this is entered in the file `SAPDOCCD.INI`. The file is searched in sequence in the Windows directory of the frontend PC (`C:\WINDOWS` or `C:\WINNT`), in the SAPgui directory or in the directory directly above it.

When you first call the Application Help, an attempt is made to read the file. If the file `SAPDOCCD.INI` was not found or the file does not contain any information for overriding settings, the settings from the profile parameters of the R/3 System are used.

For more information on the parameters you can set in the file `SAPDOCCD.INI` see the guide *Installing the Online Documentation*.

Online Documentation in a WAN

If you use SAP frontends in different locations you need to guarantee access to the online documentation. You have two options:

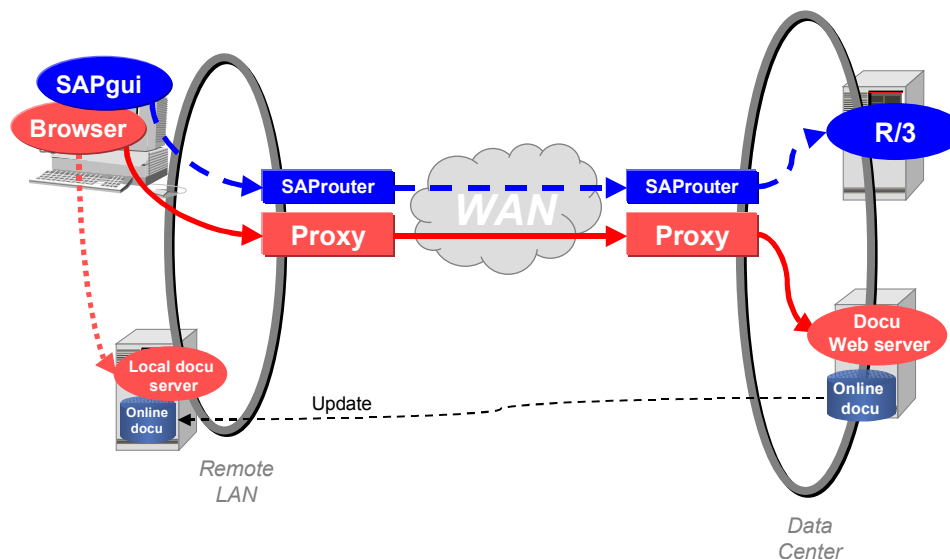
- Documentation server in the computing center that houses the SAP System
- Decentralized documentation server

SAP Online Documentation

If you want the remote frontend host to access the documentation in the computing center, you may experience problems if the SAPgui connection is made through SAProuter. This is because the SAProuter cannot relay access to the documentation. You can use HTTP proxy servers that allow indirect access, however only for the help type PlainHtmlHttp. You can install the proxies on the hosts where the SAProuters are running. You can also make this connection through multiple intermediaries. The proxy must be accessed through `SAPDOCCD.INI` on each frontend host. The WAN must bear the network load caused by users accessing the documentation.

Another option is to set up a separate frontend server for the online documentation (and other services) at each remote location. The server must be accessed through `SAPDOCCD.INI` on each frontend host. You can use all help types. The local network bears the entire network load caused by users accessing the online documentation. You must keep the documentation up-to-date, either by copying from the central documentation server, or by importing the documentation from the SAP Online Documentation CD.

The following graphic shows both options:



Installing/Upgrading the SAP Frontend Software

You must install the SAP frontend software (SAPgui) on all hosts whose users log on directly to the SAP System. Installing and upgrading this software can require a lot of time and effort to administrate. The SAP frontend installation programs let you install and upgrade the SAP frontend software with as little effort as possible.

Users who access Web-based SAP applications exclusively (for example, *Employee Self Service* or *Electronic Commerce*) only require a Web browser. You also need a Web browser for the SAP Online Help.

The installation of the SAP frontend software was fundamentally changed with R/3 Release 4.5A. This new installation is explained here. For more information, see the installation guide *Installing SAP Frontend Software for PCs*.

Installation Options and Attributes

You can install the SAP frontend software in two ways:

- **Local installation:** All the required programs are installed locally on the hard disk of the PC. The software is installed using an SAP frontend installation program delivered on the SAP presentation CD.
- **Server-based installation:** Most programs are located centrally on a frontend server and are loaded from this server each time the program starts. Links to the frontend server and certain programs that must always be available locally are located on the PC. This saves hard disk space for the executable programs on the frontend hosts. Since R/3 Release 4.0 you have to execute the SAP frontend installation program on the workstation for a server-based installation as well. This removes most of the advantages of this type of installation.

There are two types of installation procedures:

- **Installing from the CD or network drive:** You can install the SAPgui on individual PCs directly from the CD or from a network drive that holds the contents of the installation CD. Only a local installation is possible with this installation procedure.
- **Installing SAPgui packages from a frontend installation server:** The contents of the installation CD are stored on the server. Installation packages are also created for the SAP installation program. These packages contain a selection of SAP frontend components and determine certain installation parameters that would normally be queried when installing SAPgui without packages. The characteristics of installation packages are:
 - Greatly simplified installation for the end user
 - You can install the software without dialogs on the workstation
 - You can only perform the server-based installation with an installation package (mass installation)
 - The installation server can provide a service that enables users without the required user rights to install SAPgui on a Windows NT machines.

The tasks of the SAP installation programs are:

- Setting up frontend servers from the SAP presentation CD
- Administrating frontend servers and setting up installation packages

Installing/Upgrading the SAP Frontend Software

- Setting up the Windows NT installation service
- Installing the software on the frontend PCs (locally or server-based)

Planning the SAP Frontend Installation

If several PCs are connected over a local network, use fixed frontend servers to distribute and administrate the software instead of installing the software on all the workstations individually from the CD. This also applies to the SAP frontend software. The SAP frontend installation programs simplify the setup of frontend servers.

Before the installation, the administrator must consider the following:

- *On which host should the frontend server be installed?*
The host must be available to all workstations at any time. The network connection must allow a high throughput. There must be enough free disk space available.
The advantages of a frontend server for the SAP frontend software can be implemented without a central server when using a dedicated server in peer-to-peer networks (see *SAPgui in a Local Network*). In peer-to-peer networks, one of the workstation PCs takes over the function of the frontend server.
- *Which components are used by which user groups?*
For various user groups, you can create special packages with which only the necessary components are installed. The administrator must determine ahead of time which components are necessary and which packages should be created.
- *Should the installation be server-based or local?*
If you are planning server-based installations, the frontend server and the connection to it must have enough availability and capacity in case of a disruption where you cannot work with the SAP Frontend. This factor should be weighed against the advantage of saving on hard disk space on the frontend computers.
You must run the installation program in both cases on each workstation, since it must make entries in the Windows Registry. (Up to R/3 Release 3.1, this was not necessary for server-based installations; it was sufficient to install a link to the frontend server on the workstation, for example, by using an icon.)
- *Which parameters should the user enter during the installation and which values should be determined by the administrator? What are the fixed values?*
You can ensure a uniform installation by entering fixed values in the parameters. Parameters that do not have fixed values entered when the packages are created, must be entered by the users at installation. It is best to enter all the parameters, since then user entries are not required during the installation.
- *How should the frontend software be distributed on the workstations?*
If you are installing the software on many computers, use automatic software distribution. The installation program for the frontend software supports *Microsoft Systems Management Server (SMS)* and other systems for software distribution.
You can also start the installation program in a logon script. There is a command line option for this with which the installation program first checks if an installation is necessary. If all the components are already up to date, the program ends without any further actions.

The Installation in Detail

You can install the frontend software from a CD or by using a frontend server. The frontend server can be a pure file server that makes the installation CD data available to the PCs. You can also install a SAPgui installation server that offers the end user predefined software packages.

Installing the Installation Server

To use a host as a frontend installation server for the SAP software, the relevant contents of the SAP presentation CD must be sent to the hard disk. You can do this by using the setup program on the installation CD. You then have to release the directory to which the software is copied as a shared directory in the network.

After you have installed the installation server, you choose the installation package by calling the administration program. This program also installs the Windows NT installation service, if this is needed.

The Windows NT installation service makes sure that the installation program is always executed with the user ID of the person who has logged on. This security mechanism prevents specific installation steps from being executed by non-authorized users, depending on the configuration of the workstation. In this case, the installation program searches for an installation service within the domain. The service takes over the relevant installation step and then gives back control to the installation program that runs under the normal user ID.

An installation package contains a selection of SAPgui and Desktop components. Certain installation parameters, such as the installation directory and work directory, can be predefined on the workstation. You can also use computer- and user-specific placeholders, for example, for the system directory. Furthermore, you can release packages for specific user groups, and choose different packages for various operating systems.

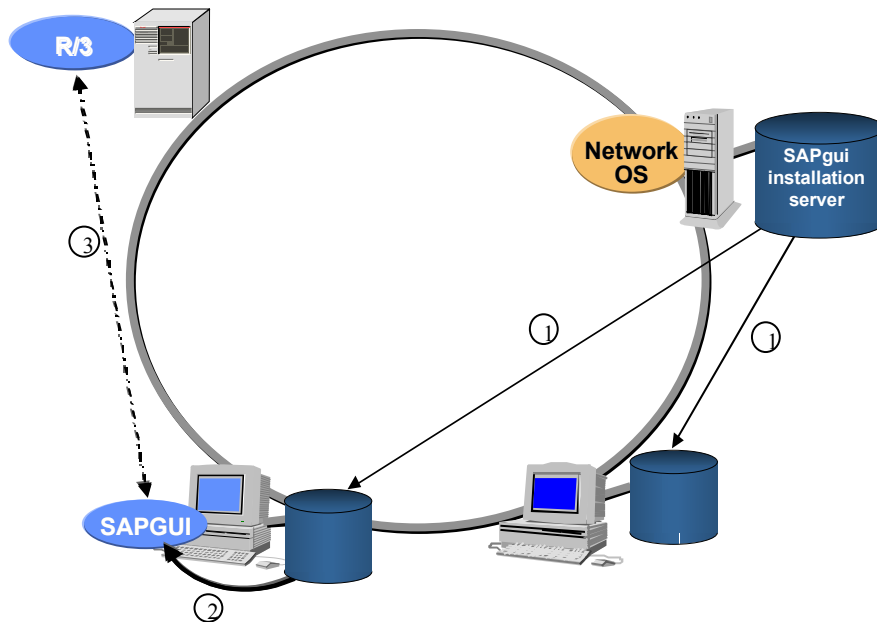
Local Installation from the CD or Network Drive

The local installation is always interactive, where you run the SAP installation program on the client PC. The user can choose the required components that are available. You must also specify the target directory and work directory on the client PC.

To perform the local installation, the presentation CD must be inserted in the workstation CD drive or the workstation must have access to the installation software on the frontend server using a drive connection.

After the installation, the SAP frontend software is located on the hard disk of the client PC. You do not need the connection to the frontend server anymore when using the SAPgui.

Installing/Upgrading the SAP Frontend Software



- (1) Frontend is installed
- (2) SAPgui is started
- (3) Communication with the R/3 System

Advantages:

- You can work with the SAP System independently of the frontend server
- No network load by loading the software (software located locally on each PC)

Disadvantages:

- Disk space is filled on each PC with the SAP frontend software
- Interactive installation is necessary on each PC
- Users must choose the installation parameters and components themselves

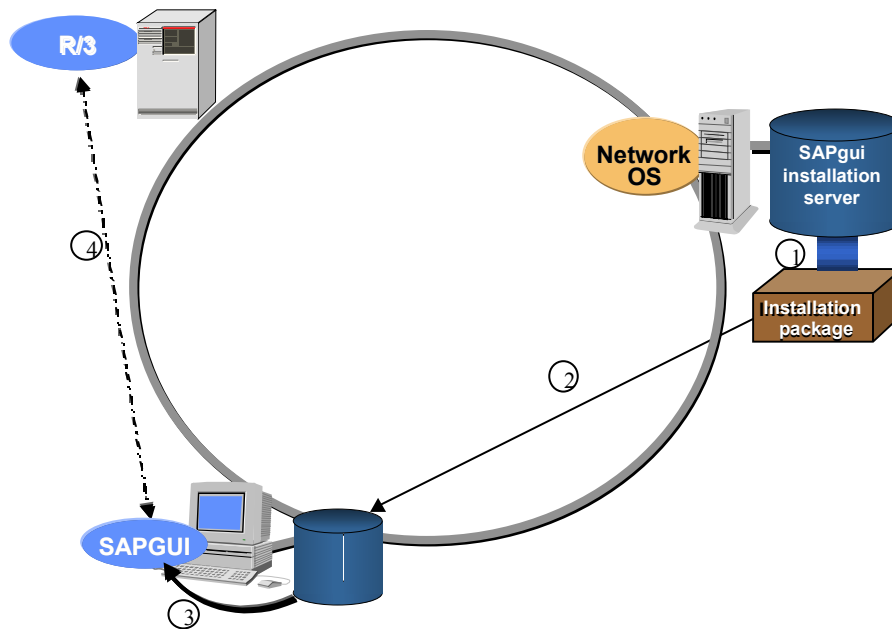
Local Installation from a Frontend Server with Installation Package

You install installation packages on the workstation by using the program *NetSetup*. In interactive mode, the user can choose the package needed from the ones available. If you enter the relevant command line parameters, which specify the installation package, the installation can run without any dialogs.

Another program option is a check to see if an installation is required. If all the components of the package are up-to-date and available on the workstation, the program ends immediately; otherwise the installation begins without any dialogs. This is particularly useful for automatically distributing the SAP frontend software in logon scripts.

Installing/Upgrading the SAP Frontend Software

After the installation, the SAP frontend software is located on the hard disk of the client PC. You do not need the connection to the frontend server anymore when using the SAPgui.



- (1) Installation package is created
- (2) SAP frontend software is installed on the workstation
- (3) SAPgui is started
- (4) Communication with the SAP System

Advantages:

- You can work with the SAP System independently of the frontend server
- No network load by loading the software (software located locally on each PC)
- You can perform the installation for users without the required user rights (Windows NT)
- You can choose installation packages for user groups
- You can install the software without any dialogs

Disadvantages:

- Disk space is filled on each PC with the SAP frontend software

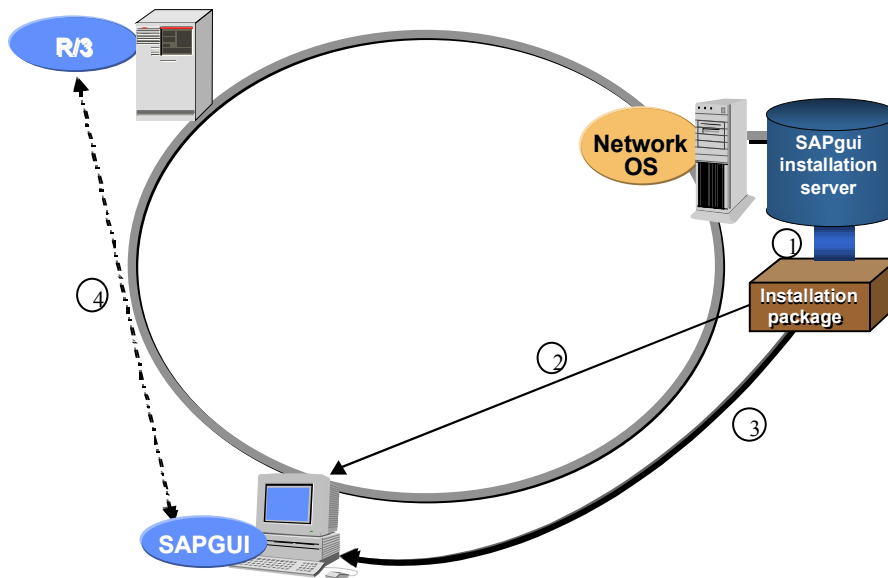
Server-Based Installation from Frontend Server with Installation Package

You can only perform the server-based installation with installation packages from a frontend server. When choosing the installation package by using the administration program, you have to specify that the installation is to be server-based. The software has to be installed on each workstation and corresponds directly to a local installation with installation packages. This installation is much faster since fewer files have to be copied.

Installing/Upgrading the SAP Frontend Software

The SAP frontend software is loaded each time the frontend server is started. Only a few system files and ActiveX controls, which have to be registered locally on the frontend computer, are stored on the PC. This saves a large amount of hard disk space on each workstation.

Each time the SAPgui is started, approximately 3 MB of data is transmitted. Additional data traffic is created when desktop components are used, such as SAP Business Graphics, the graphical Screenpainter, and so on.



- (1) Installation package is created
- (2) SAP frontend software is installed on the workstation
- (3) SAPgui is started over the network from the frontend server
- (4) Communication with the SAP System

Advantages:

- Less hard disk spaces is used on the workstation

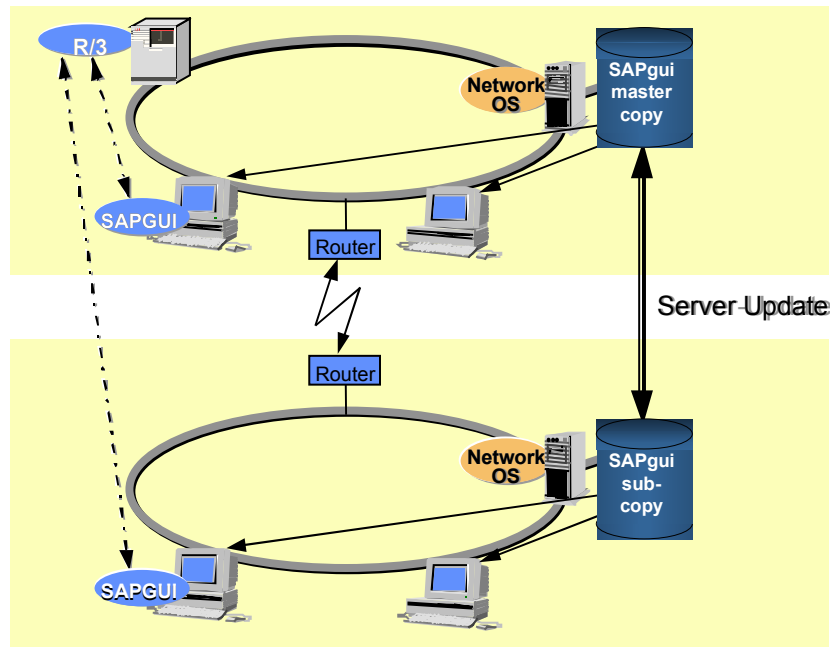
Disadvantages:

- The ability of the client to work depends on the availability of the server (if the frontend server fails, the clients cannot work with the SAP System)
- Increased network load when loading the software over the network

Remote Networks

For physically remote locations, you need to consider the same factors that were covered in the previous sections about a purely local network solution. When distributing the software to the client PCs, consider each location as an individual unit. When providing new versions for the decentralized locations, you should do this centrally (see graphic).

Installing/Upgrading the SAP Frontend Software



Install one or several frontend servers at each location - depending on the number of clients connected there. For locations with less than 5 PCs, a peer-to-peer solution can also be used. You can implement all of the described approaches at the locations.

We recommend controlling the software distribution using a central server, since this ensures consistency for all locations.

The current frontend software is installed from the SAP installation CD on the central server, and then it can be distributed to the servers at the decentralized locations. The new version of the software is distributed only once for each location.

Laptops

The SAPgui communication protocol has been optimized to keep network loads to a minimum. This means that it is suited to connections with a narrow bandwidth (for example, telephone lines with a modem. See *SAPgui in Wide Area Networks*).

When you install frontend components, you must install the SAP frontend software locally (depending on the chosen components, approximately 25 to 150 MB) on the hard disk of your laptop. A server-based installation is usually not an option since the programs used must be transmitted each time before you use them. There are several methods for installing the SAPgui on laptops, three of which are described in the following section.

Installing Software over Wide Area Network (WAN)

Field sales employees may need to get the newest version of the SAP frontend software from a remote connection. If your laptop is connected to a WAN corporate network, you can install the software like a local installation. When choosing installation packages, consider the available telephone line transmission speed and the amount of data to be transmitted.

Installing/Upgrading the SAP Frontend Software

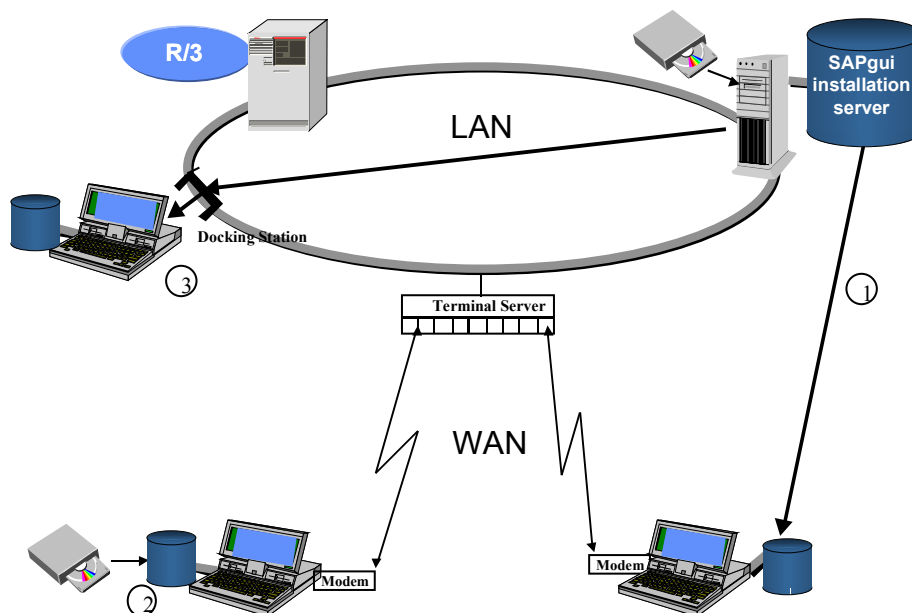
If you cannot establish a direct connection to a corporate network, you can also use the file transfer protocol (*FTP*) to transmit the installation directories and files to the laptop. Then the software is installed locally. If you do not need all of the files from the installation CD, ensure that you can actually install the files that you want to transmit. We recommend the installation of software using WAN connections for minimal installations only (SAPgui without additional components), since the amount of data can grow significantly with additional components.

Installing Software from a CD

If your laptop has a CD drive, you may want to install the software locally directly from the CD.

Installing Software over a Temporary LAN Connection

You can install the software if your laptop is connected temporarily to a LAN (for example, using a docking station) while you are at a corporate branch. The SAP frontend software is installed or updated in a few minutes just like the installation on a normal workstation.



- (1) Installation using a WAN connection
- (2) Installation from a CD
- (3) Installation using a temporary connection to the LAN

SAP Internet Integration

Introduction

Introduction

This documentation contains technical information for planning and setting up a Web-enabled SAP system with the SAP Internet Transaction Server (ITS). Special emphasis is put on the integration of the SAP software with the other network components needed for an Internet connection, such as Web servers, firewall products and TCP/IP networks.

The development and administration of ITS applications is covered only briefly. For more information, see the SAP Library under *BC – Basis Components → Frontend Services → Internet Transaction Server (BC-FES-ITS)*.

This document is intended for SAP Basis specialists interested in the network integration aspects of an ITS scenario and for network specialists interested in the special requirements of SAP Internet technology.

Internet Technology Overview

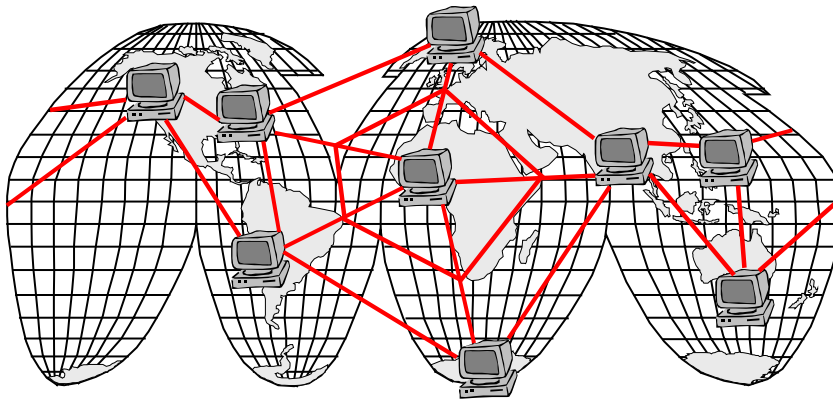
The term "Internet" is used in a variety of meanings. Strictly speaking, the Internet results from the connection of many smaller networks, which are operated by a variety of organizations, to form one large network. This means that there is only one Internet. It was founded as a research project by the US Department of Defense, and subsequently expanded to include international research institutions and universities. In recent years it has become a public, worldwide, and increasingly economically oriented network with unlimited access for both business and private users.

Internet Protocol (IP) Network Technologies

All participants in the Internet communicate using standard networking technologies based on the Internet Protocol (IP). These standards are publicly available and are continuously developed by specialists in the field. They are thus called "open standards".

The World Wide Web

One of these standards is the World Wide Web technology, abbreviated as "WWW" or "Web" technology. It lets the user retrieve different materials over the network, including textual information and multimedia content. These materials are retrieved from different sources in a uniform way by using a Web browser. Using additional software, such as the SAP Internet Transaction Server, a Web browser can also enable you to perform interactive transactions on the Internet.



The Web is the part of the Internet that is accessed through Web browser technology. E-mail technology is an example of a different Internet technology. Often the terms "Internet" and "Web technology" are used synonymously. The phrase "connecting SAP to the Internet" in this sense means "building a Web interface for SAP".

Some other terms that are used frequently in this context should also be mentioned. An "intranet" is a corporate network, which is not necessarily connected to the Internet. By setting up an

Internet Technology Overview

intranet, a company uses the open networking technology of the Internet internally, to connect its different branches or subsidiaries. The intranet thus offers a World Wide Web service to internal users. An extranet is usually a network that connects different companies for business purposes, but that is restricted to a defined set of partners and not publicly available. Many companies use the Internet, intranets and extranets: the *Internet* for public information about the corporate and shopping applications, an *extranet* to enable communication with external customers or partners, and the *intranet* to enable communication within the company.

HyperText Markup Language (HTML)

Web pages are retrieved by a Web browser and displayed in a window. The format and contents of a page are determined by the page-description language, HyperText Markup Language (HTML). This language not only forms the content of the text, but also contains formatting instructions, called "tags", to format:

- Text characters (such as bold formatting), lists and tables
- Links to other Web resources
- Images embedded in Web pages
- Forms that can be filled out by the user and returned to the Web source (containing text fields, buttons and similar components generally found in graphical user interfaces).

The HyperText Transfer Protocol

The Web server (also called "HTTP server"), is a program that handles Web browser requests (or refers to the machine running such a program). The protocol used for communication between the Web browser and Web server is called HyperText Transfer Protocol (HTTP).

Communication between the Web browser and the Web server is based on a client/server model: The Web browser opens a connection to the Web server and sends a request for a certain Web page. The Web server answers by sending the contents of the page (or an error message). Then the connection is closed. Status information about the connection is not stored. Status information used in complex Web applications (for example, when the user collects items in a virtual shopping basket) requires an additional program for the Web server.

Encryption

Data is normally transmitted in HTTP in non-encrypted text. If the data is sensitive, containing, for example, credit card numbers, medical records, user IDs or passwords, it should be encrypted. Secure Socket Layer (SSL) is the encryption protocol supported by most modern Web browsers and professional Web servers. The SSL-encrypted HTTP transmission protocol is called HTTPS.

Web Resources

A Uniform Resource Locator (URL) is used to specify resources on the Web. The URL contains the address of the Web server, the name of the resource, and the transmission protocol. The structure of a URL is:

```
protocol://address/name
```

For example: `http://www.sap.com/internet/index.htm`. In this example, the protocol is HTTP. The address – `www.sap.com` – is the IP address of the central SAP Web server. The name of the resource is `internet/index.htm`, which is the name of a specific file on that server.

Web resources can include:

- Static Web pages
- Images, sounds and other multimedia data, in data formats known as Multipurpose Internet Mail Extensions (MIME).
- Dynamic Web pages that are generated online, for example by a database or an SAP System.

Static Web Pages

In the early days of the Web, the user could only retrieve static HTML pages that had already been compiled in a file before retrieval. Examples of static pages include product information, catalogues, and telephone lists. Static information can be updated frequently, but this requires considerable administrative effort. Instead of static Web pages, you can use a special program to generate Web pages as required. Interactive applications are not possible using static pages. Today, however, you can create (dynamic) Web pages to meet your needs. To do this, you usually need additional software on the server.

The Common Gateway Interface (CGI)

One method that enables a Web user to access dynamically generated pages uses the Common Gateway Interface (CGI) to connect the Web server to an additional program, which generates Web pages dynamically in response to user input. CGI programs or scripts can manipulate database data and other external information. They are programmed either in a compiled language such as C or C++ (CGI programs) or in a scripting language such as Perl (CGI scripts).

The CGI determines how:

- Parameters of user requests are encoded in a URL
- User requests are transmitted from the Web server to the gateway program
- Data is returned to the Web server

How does the request get to the Web server? HTML forms refer to a certain URL. When the form is submitted by the user, a Web request is sent to the Web server, consisting of the URL and the data contained in the form.

The Web Server is configured to automatically start an external CGI program when certain URLs are called. The CGI program then receives the URL and the form data, performs the required processing and generates the answer page which is returned to the Web server.

CGI Scripts have some disadvantages:

Internet Technology Overview

- Every new call creates a new process, which has to initialize itself and log on to the database, thereby increasing system load. If there are many calls, performance may be reduced.
- HTML pages are generated entirely by the CGI program. That means that every time a change has to be made to page functions or design, a programmer has to alter the program code. This makes complex CGI applications hard to manage.

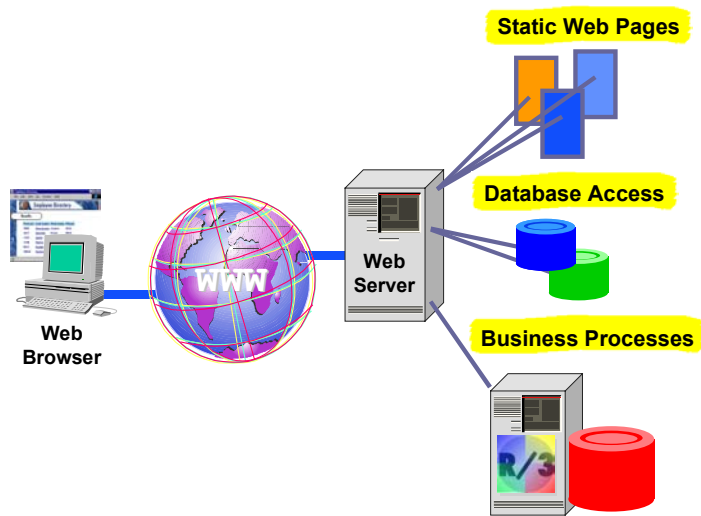
Modern Web Technology

Many concepts have evolved to overcome the disadvantages of CGI and build powerful Web applications:

- **Closer Integration of Web Server and Extension Program**
All modern Web Servers support advanced extension interfaces that overcome most of the problems of CGI. Unfortunately there is no single standard that all Web Servers adhere to. Two widely used standards are Microsoft's Internet Server API (ISAPI) and Netscape's Netscape Server API (NSAPI). They work as follows: The extension consists of a dynamic library (DLL) or shared object library that implements the API of the server. When the appropriate URL is called for the first time the Web server loads the object into its own process space and passes the request to a function in the object. This function processes the request and passes the response back to the Web server. Subsequent requests are handled much more efficiently because the object is kept in memory.
- **Web Servers with Integrated Programming Environment**
Web server capabilities can be extended by integrating a scripting language into the Web server. In this way, pieces of script code can be embedded into HTML pages. The code is usually marked with a special tag and executed by the Web server before the page is transmitted back to the user. An example is Microsoft's Active Server Pages (ASP). Web applications can be implemented with server scripts that may also access additional data sources, such as a database or an SAP System.
- **Multi-Tier Web Application Servers**
Web application servers are an additional software layer for connecting the Web server with a back end system such as an SAP System or a database. They handle Web requests, user sessions and communication with the back end. The SAP Internet Transaction Server falls into this category.

The Evolution of Web Applications

Since the Web was originally designed for information exchange among scientists, everything started with simple access to static Web pages. The next steps to increase interactivity were search engines and simple database access. The latest step is the complete integration of business processes with the Web technology. The following diagram shows the evolution of Web applications:



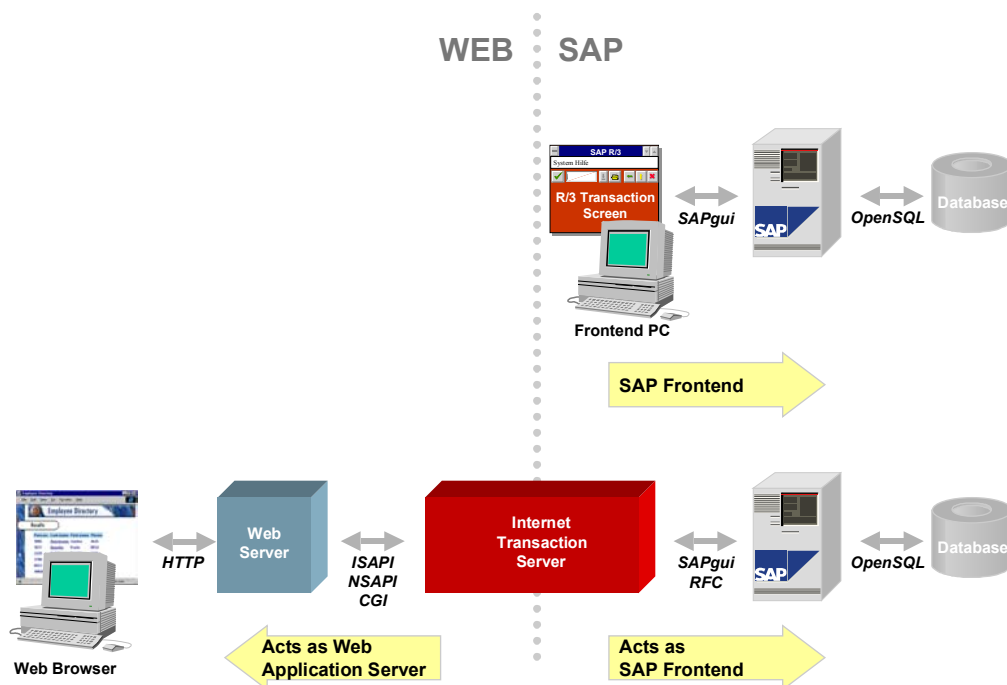
ITS Technology

ITS Technology

What is the SAP Internet Transaction Server?

The conventional method of accessing the SAP System interactively is by using the SAPgui, which displays SAP transaction screens and allows the user to interact directly with the system. It communicates with the SAP System by using a special SAPgui protocol that uses TCP/IP as its networking basis. The SAPgui has a direct network connection to the SAP Application Server via a local or wide area network. Programs can also use the SAP RFC protocol (Remote Function Call) to communicate with the SAP system.

So that you can access the SAP System with a Web browser, you need additional middleware to make the connection between SAP technology (SAP Application Servers and SAPgui) and Web technology (Web servers and browsers). This is accomplished by the ITS. It offers the services of a Web application server, however it does not do any transaction processing itself. Instead it hands all business processing to the SAP System, to which it looks exactly like a regular SAPgui. This relationship is illustrated by the following graphic:



ITS Features

Since the processing of the SAP Internet Applications Components (IACs) takes place in the SAP System, the benefits of the SAP infrastructure apply immediately to Web applications. Some of the features of ITS are:

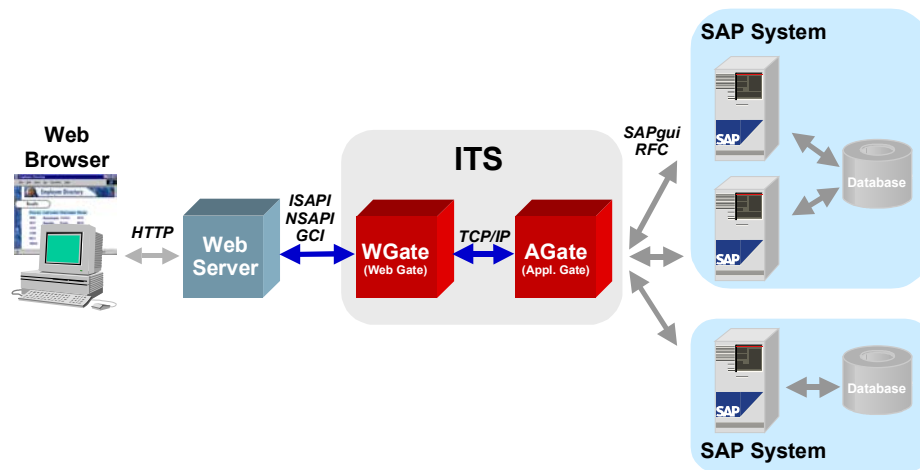
- Scalability

- Handling of logon and user sessions in the SAP System
- Transactional consistency for Web applications
- Multi-language capabilities
- Code page conversions
- Full integration with the ABAP Workbench
- Change and Transport System
- User management and authorization concept
- Scalability
- Handling of logon and user sessions in the SAP System
- Transactional consistency for Web applications
- Multi-language capabilities
- Code page conversions
- Full integration with the ABAP Workbench
- Change and Transport System
- User management and authorization concept

ITS Architecture

The Internet Transaction Server is the link between the Web and SAP. It is composed of two separate programs: WGate (Web Gateway) and AGate (Application Gateway), which may reside on the same computer or on separate computers connected by a TCP/IP network. The graphic below shows the components of ITS which are explained in detail in the following sections:

ITS Architecture



The Web Browser

Most currently available Web browsers can be used to access SAP Internet Application Components (IACs). The HTML 3.0 standard must be supported. Some of the SAP IACs use special features like Java applets or "Dynamic HTML" which may not be supported by all browsers.

The Web Server

The Web server is the interface between the Internet/Intranet and the ITS. It has the following functions:

- Accepts HTTP requests from client browsers.
- Forwards specific requests to the WGate through one of the supported interfaces and transmits the dynamically generated HTML pages back to the client.
- Delivers static information, such as pictures embedded in HTML pages, directly from the file system of the Web server machine.

The Web server automatically loads the WGate when a certain URL is called. This URL typically has the form

```
http://<server>/<path-to-wgate>/wgate/<service>/!, for example
http://www.flexicorp.com/scripts/wgate/va01/!.
```

Here the directory `scripts` is a virtual directory of the Web server that allows the execution of CGI programs or server extension DLLs. The parameters after `wgate/` are passed to the entry function of the DLL. In this case the parameters tell the ITS to start a service called `va01`.

For a list of Web servers supported by ITS see the brochure *SAP System Requirements* (available in SAPnet under <http://sapnet.sap-ag.de/SSR>).

WGate

The WGate component connects the ITS to the Web server. The WGate is always located on the same computer as the Web server. The following standard Web server interfaces are supported:

- **Microsoft Information Server API (ISAPI)** on Windows NT. The Microsoft Information Server API loads the WGate into the Web server process as a dynamic link library (DLL).
- **Netscape Server API (NSAPI)** on Windows NT. The Netscape Server API also loads the WGate into the Web server process as a DLL.
- **Common Gateway Interface (CGI)** on Windows NT, UNIX and AS/400 (as of Release 4.5A). On the UNIX and AS/400 platforms, the Common Gateway Interface starts the WGate as an external executable program.

The WGate receives requests from the Web server, establishes a connection to the AGate and forwards the requests. No data processing is done by the WGate except when an error during communication with AGate occurs. In this case the WGate generates an error message.

One WGate is configured for communication with exactly one AGate. Each WGate instance has a name which is equal to the name of the corresponding AGate.

AGate

The AGate program is implemented as a Windows NT service. It runs on the Windows NT 4.0 operating system on Intel processors. Although the AGate can be located on the same machine as the WGate, we recommend that you keep the two components on two separate machines for productive use.

The AGate is responsible for managing communication to and from SAP, including:

- Establishing the connection by using SAPgui or RFC protocols
- Generating the HTML documents for the SAP applications
- Managing user logon data
- Managing session context and time-outs
- Code page conversions and national language support

Multiple AGate instances can be installed on the same computer. Each instance has a unique name.

ITS Installation Options

The SAP System

The ITS accesses the SAP System just like any SAPgui or RFC client program would. It can use different SAP Systems for different Web services. All logon methods, such as load balancing with the SAP message server, are supported by the ITS. No special technical provisions have to be made in the SAP System to use IACs.

ITS Administration Interface

ITS features an HTML based administration tool that allows you to configure, control and monitor ITS instances. You can use it to:

- Monitor ITS instances, either individually or all together
- Monitor ITS performance
- Start or stop ITS instances or their corresponding Web servers
- Set ITS configuration parameters and security settings
- View ITS service file settings
- View registry settings
- View NLS (national language support) information

For details see the SAP Online Documentation.

ITS Installation Options

The two components of the ITS, AGate and WGate, can either be installed on a single computer (single-host installation) or on two separate machines (dual-host installation). Multiple ITS instances can be installed on one computer, just like multiple SAP system instances. These options are explained in the following sections.

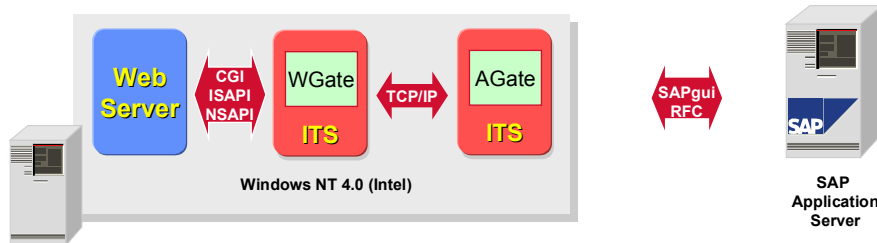
Single Host and Dual Host Installation

The simplest configuration is to install the Web server, WGate and AGate all on one computer running Windows NT 4.0 on an Intel processor. This setup is sufficient for test or development purposes with small load ("personal ITS"). It is not suited for serving the Internet.

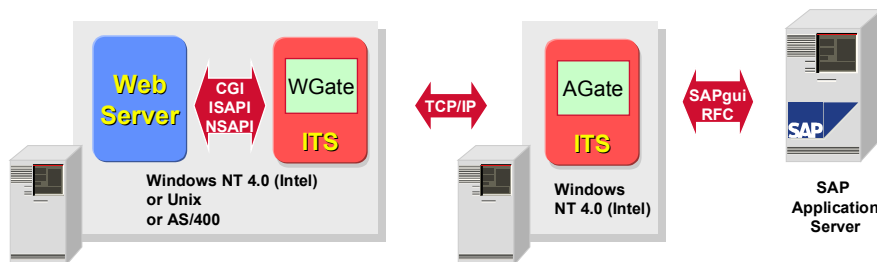
In order to cope with heavier load or to meet higher security demands the ITS functions can be split between two computers. In a productive environment this dual-host configuration is advisable. The first computer runs the Web server and the WGate. It has to be connected to the client access network (Internet and/or intranet). The second computer runs the AGate. It is connected to the WGate via TCP/IP network and handles all communication with the SAP System.

The following graphic shows the single- and dual-host installation. Since the AGate is available in a version for Windows NT 4.0 on Intel only, this is the only platform supported for the single-host installation. In dual-host installations the WGate can also run on a UNIX or AS/400 host.

Single-Host ITS Installation



Dual-Host ITS Installation



ITS Instances

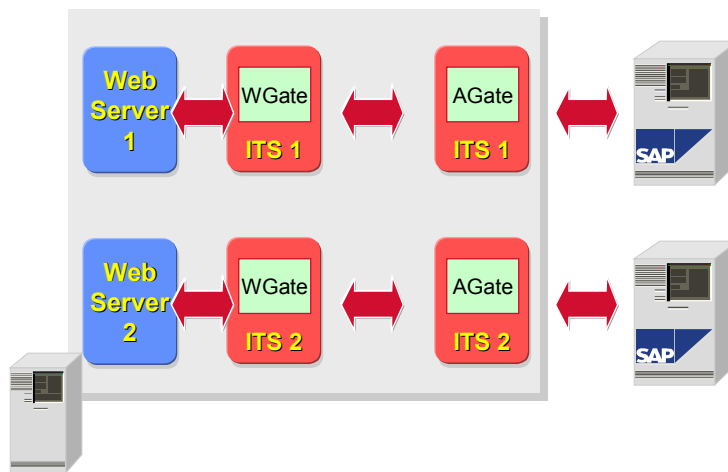
You can install more than one ITS instance (also called "virtual ITS instances") on one computer. Each instance has a unique name and runs as a separate Windows NT service. The executable files are shared with all of the AGates on a single computer allowing only one ITS version to be installed on a single computer.

Most supported Web servers also allow multiple virtual Web servers on one computer. Web server instances also have names by which they can be distinguished. There are three ways to access the different Web server instances located on one computer:

- The computer has multiple IP addresses. The addresses can be assigned to one or more Network Interface Cards (NICs).
- The Web servers use different TCP ports. For example, one server can use the standard port number 80, another server uses port number 1080.
- The HTTP header field "Host". In this case the single IP address of the Web server has multiple alias names assigned to it. Now all requests to the different server names actually reach the same computer. The Web server can still distinguish the requests by means of a field in the HTTP request which specifies the name of the server that the request is sent to. This feature is part of HTTP Version 1.1. Although most browsers do not fully implement this standard, the Host field is supported by most browsers, even older ones (Internet Explorer, Netscape Navigator and Opera versions 3.0 or higher). This feature does not work with SSL encrypted HTTP.

SAP Internet Application Components Technology**Note**

If you use multiple virtual Web servers with WGate that you want to connect to different AGate then you must make sure that these Web servers run in separate processes or memory spaces. Otherwise only one WGate would be present in memory. The Netscape Enterprise Server does this automatically. In IIS 4.0 you have to check the option "run in separate memory space" in the properties page of the Web server instance and reboot the computer. Note that this option may seriously affect the performance of the Web server.



This installation variant is useful for Web applications with a small load, such as development and test systems. The ITS administration tool (see online documentation) should also be installed as a separate ITS instance on every AGate computer.

SAP Internet Application Components Technology

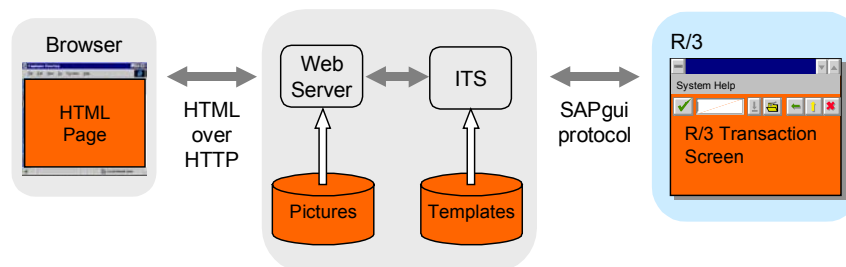
A Web application with ITS is called an SAP Internet Application Component (IAC). An IAC consists of different components. Some of them reside on the ITS server, some in the SAP system. The ITS offers the following technologies for setting up Internet Applications:

SAP Internet Application Components Technology

- **WebTransactions** are Web-enabled SAP Transactions. They are screen-based and follow a dialog flow just like normal SAP transactions. This implies that WebTransactions are usually stateful, which means that the SAP applications server holds a memory of the current application state. Subsequent dialog steps use the information of previous steps. After completion of all necessary steps, the transaction is terminated and either commits the result to the database or rejects all changes made so far.

The state of the application data is kept in the SAP system. The ITS has to keep only the state of the SAP screen in order to synchronize it with an appropriate Web page. It generates a new Web page dynamically for every SAP transaction screen. This is done with the help of templates which are written in an ITS specific programming language called "HTML Business". Additional resources such as language and theme files are used to simplify the task of multi language support and Web page design.

The following figure shows how the components that make up a WebTransaction work together. It also shows the communication that takes place between browser, ITS and SAP System.

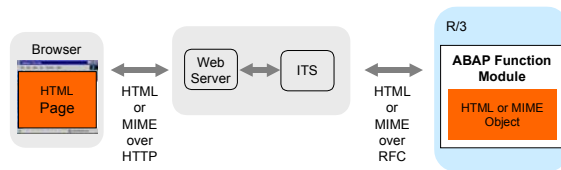


- **WebRFC** communicates directly with ABAP function modules. These function modules act like CGI scripts in that they process user input to create dynamic Web pages. WebRFC is a low-level programming interface.

Only function modules that implement a specific interface can be used by WebRFC. The function module must evaluate the Web request arguments, execute the corresponding business functions, and generate an internal table which contains the HTML page that is then sent back to the browser. Alternatively, it can create a binary MIME object and send this to the browser, for example, a graphic or Excel chart. The ITS handles logon management and error conditions for the programmer; the functionality must be implemented entirely by the function module.

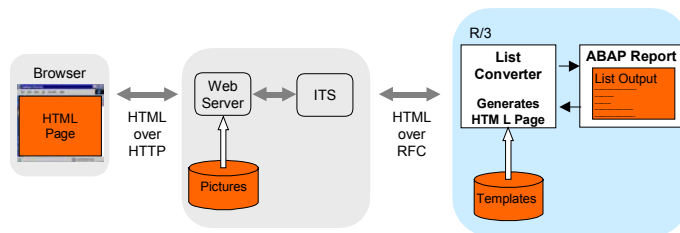
The following diagram shows how a WebRFC application works:

SAP Internet Application Components Technology



- WebReporting** allows you to call (almost) any ABAP report or list from the Web. It is based on WebRFC. The ITS calls a special ABAP function module in the SAP System to execute the report. Web pages for the selection screen are generated automatically by this function module. Then it calls the report and converts the output data to HTML and returns it to the ITS, which uses HTTP to pass it back to the Web browser. The conversion is aided by templates that are different from ITS templates in that they reside in the SAP System and do not use the HTML Business language.

The following diagram shows how an SAP report is processed for the Web:



Services

The representation of an Internet Application Component (IAC) in the ITS is called a "service". It consists of all components needed to run this application, such as templates, language resources, graphics and so on. Every service also has a "service file" that contains the settings required to connect and log on to the SAP System. Depending on the application it may contain an SAP user name and password. See [ITS Security \[Page 100\]](#) for details about service users and named users and the security issues related to these concepts. If user name and password are missing from the service file, ITS generates a login window prior before it starts the IAC.

The service is specified in the URL that the browser sends to start the application. This URL looks like

```
http://www.mycorp.com/scripts/WGate/online-shop/!
```

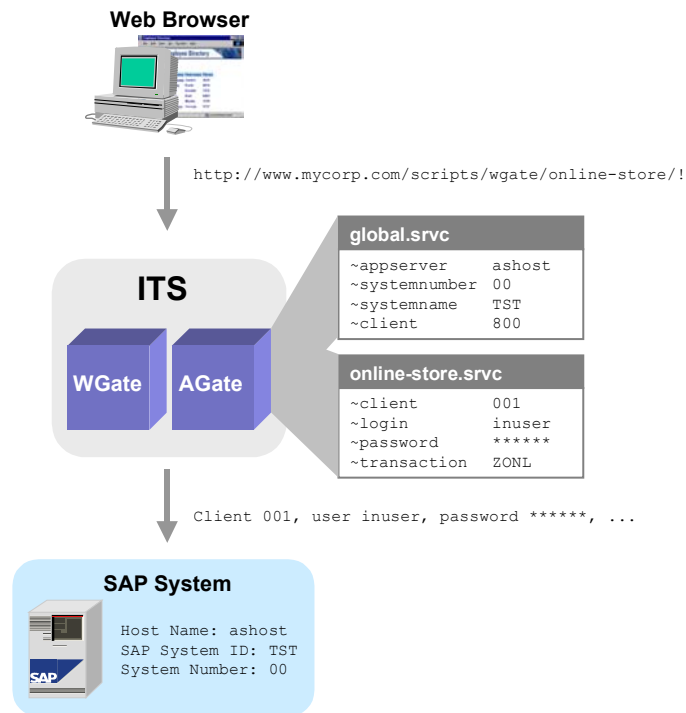
where `online-shop` is the name of the service. The ITS now looks for a file called `online-shop.srvc` for the service information. If this file does not exist, ITS issues an error message.

The ITS usually has one additional master service file called `Global.srvc` which contains service information applicable to all applications executed by this specific ITS. Typically, SAP

SAP Internet Application Components Technology

connection information, such as the system ID and system number, is stored here. The settings made in this file can be overridden by the service file for the specific application.

The following graphic shows the relationship between the global and the application-specific service file. A list of all service parameters can be found in the SAP Online Documentation.



ITS Network Connections

All components of the ITS are connected through TCP/IP networks. Here we describe the characteristics of these network connections (especially TCP ports). The load that IACs impose on the network connections is described in [ITS Network Load \[Page 150\]](#).

Browser – Web Server

The Web browser and the Web server use a TCP/IP network to communicate, for example the Internet or a corporate intranet. The standard HTTP protocol is used for this connection. No additional communication channel or software is needed to use the IACs delivered by SAP.

A Web server requires one TCP service. Port number 80 is reserved for HTTP and used by default by all servers and browsers. If you want to use a different port number you can configure your Web server port numbers freely. If you do this, the URL must contain this port number (in this case, 1080) in the form

```
http://server.mycorp.com:1080/index.html
```

If the communication is encrypted with SSL, a different port number is used. The default is 443. The URL for an HTTP request over SSL has the form

```
https://secureserver.mycorp.com/index.html
```

WGate – AGate

Communication between the WGate and the AGate uses a TCP/IP network connection. The WGate opens a new connection to the AGate for every incoming request. The connection uses the SAP Network Interface (NI). It is an additional protocol layer on top of TCP (since ITS version 1.1). This protocol provides two benefits:

- It can be relayed through SAProuter
- You can use SAP SNC encryption for high security demands

By default, the data sent between the WGate and the AGate is sent as clear text. You can choose a different connection type which encrypts the data with an DES algorithm and a static key. This key is not configurable; therefore, this encryption provides protection only against accidental reading of the data, but not against serious attacks. See [ITS Security \[Page 100\]](#) for details.

The WGate opens the connection to the AGate dispatcher services on the AGate host. The AGate dispatcher service name is `sapavw00_INST`, where `INST` is the name of the ITS instance. The file `\WINNT\System32\Drivers\etc\Services (/etc/services on UNIX)` defines which port number this service name is mapped to.

When you install an ITS component, 10 service ports are automatically added to the file `etc/services`:

```
sapavw00_INST      tcp/3900
```

```

sapavw01_INST    tcp/3901
...
sapavw08_INST    tcp/3908
sapavwmm_INST    tcp/3909

```



For normal ITS installations only the port `sapavw00_INST` is required. The other ports are not used and may be deleted from `etc/services`.

The ITS setup program tries to find a sequence of 10 unused ports starting with port number 3900. This procedure is repeated on each computer where an ITS component is installed. As a result the `sapavw00_INST` port number may vary for different installations. You have to check your installation to find out which port is actually used. Each ITS instance uses its own ports.



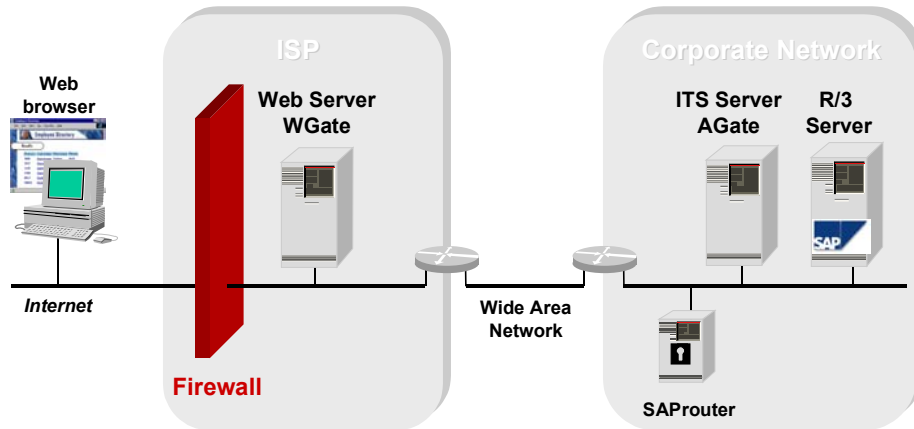
You have to make sure that the `sapavw00_INST` port numbers are identical on the WGate and the AGate host. This is not guaranteed automatically.

Outsourcing the Web Server

An Internet Service Provider (ISP) is needed if you want to offer Web services to the Internet. Some ISPs offer additional services, such as hosting your complete Web server on one of their computers. You can also use these services for running an ITS to reduce the technical effort on your side. This would also reduce the bandwidth needed for the connection between the ISP and your site, because all pictures are loaded directly off the Web server.

The WGate is located on the computer of the ISP and only the WGate–AGate connection has to be routed into your corporate network. A SAProuter can be used for security. See [ITS Security \[Page 100\]](#). The following graphic shows a possible setup:

AGate – SAP System



The ISP's infrastructure has to meet the following requirements:

- The Web server must run on one of the platforms supported by the ITS. These platforms are listed in the brochure *SAP System Requirements* (available in SAPnet under <http://sapnet.sap-ag.de/SSR>).
- The security measures must meet your standards.
- The pictures used by the IACs are located on the Web server. This means that you need convenient access to this computer for updates and maintenance.
- Using a dedicated computer for WGate is preferable to sharing one computer with other customers of the ISP.
- Since no sensitive data is located on the WGate computer, security is usually not critical. Communication between the WGate and the AGate can be secured by encryption (see [ITS Security \[Page 100\]](#)).

AGate – SAP System

The SAP connection parameters are entered during the ITS installation process. They are stored in the global service file (see [ITS Technology \[Page 80\]](#)). You can change this file at any time. All connection modes available with the SAPgui, such as load balancing or SAProuter, can be used with the ITS.

The connection between AGate and SAP system depends on the ITS programming model used:

AGate – SAP System

- **WebTransactions:** Since the AGate behaves exactly like a SAPgui, it holds one TCP session for every user session. WebTransactions keep the transaction state inside the SAP system.
- **WebRFC and WebReporting:** Usually each request opens a new RFC connection to the SAP System. Each request requires a SAP logon. The logon data is stored in the ITS so that the user does not have to re-enter it. To identify user sessions the ITS relies on a unique session ID that is encoded in every Web request. In special cases it is possible to use one RFC session for multiple Web requests.

All connections are opened by the AGate to the SAP application server and to the message server. The communication channel is identical to normal SAPgui and RFC clients. The following TCP services are used:

- **WebTransactions:** Port `sapdpXX` for the SAP dispatcher (XX = SAP instance number)
- **WebRFC and WebReporting:** Port `sapgwXX` for the SAP gateway (XX = SAP instance number)
- Port `sapmsSID` for the message server when load balancing is used (SID = SAP System name)

You can use a SAProuter to relay all communication through a single connection.

ITS Scaling and Availability

ITS Scaling and Availability

As the number of Web users increases, the ITS can be scaled accordingly. The ITS multi-tier architecture offers many options but also demands certain rules to be followed. Here we describe some of the mechanisms that can be used.

Availability is related to scalability, because in order to achieve high availability of your ITS service you avoid situations where the failure of a single component brings down the entire system. This means that you need at least two separate ITS instances. Availability issues are discussed briefly where applicable.

ITS System Load

The CPU load on the **WGate computer** depends mostly on the Web server program, because the WGate processes no data and consumes few CPU resources. For this reason the CGI WGate should not be used for high performance ITS applications. As a general rule, the computer running the Web server and WGate has much less CPU load than the one running the AGate. Of course, if the Web server runs other applications apart from SAP Internet Applications Components, no general statement can be made.

Also note that this statement only holds for the ISAPI and NSAPI Web server extensions. If the CGI WGate is used, the system load used to start a new WGate executable program for every request is very high and exceeds by far the load of the WGate itself.

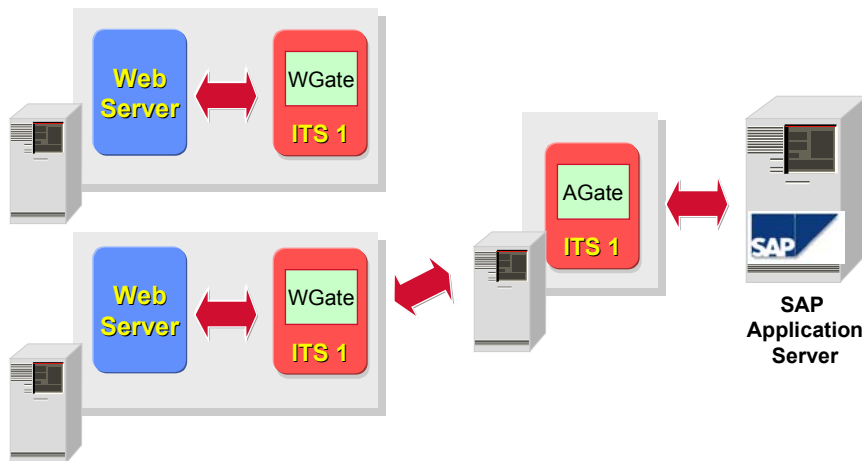
The **AGate** performs the actual processing of the Internet Application Components. For that it has to parse Web requests, merge data into SAP screens, open connections and issue requests to the SAP system, interpret and merge HTML templates and other resource files and generate new HTML pages. These tasks can be quite time consuming.

Consult the SAP hardware partners for detailed sizing information.

Scaling the WGate and Web Server

Multiple WGate running on separate Web server computers can access a single AGate. This is possible because for every request the WGate opens a connection to the AGate. The AGate sends the answer back to the WGate by using the same connection.

Conceptually, when multiple WGate are connected to a single AGate, all belong to the same ITS instance. This can be useful if one ITS serves different networks simultaneously, for example, the Internet and your corporate intranet. The setup is shown in the following diagram:



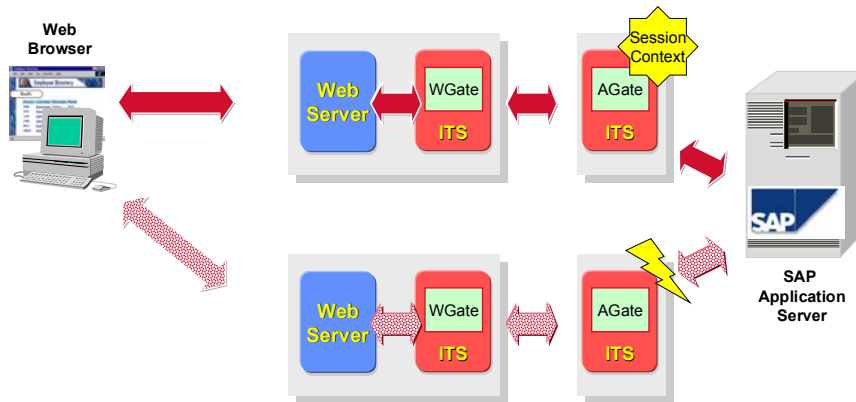
Scaling the AGate

If a single AGate machine is not capable of handling all ITS users, multiple AGate computers have to be used. In this way ITS applications can be scaled almost without limit, provided the SAP system is able to handle the load.

For every AGate computer a separate Web server is necessary, because currently the WGate can only communicate with one AGate. It can not distribute requests among different AGate computers. The Web servers are accessed by different IP addresses (if located on separate computers) or port numbers or IP names (if you use multiple Web servers on one computer). In order to give the user a single access point, a method must be implemented to distribute incoming requests among these Web servers.

There is one important requirement for this load distribution mechanism: It must make sure that the requests from one user are **always** directed to the same Web server where he or she started the session. This is necessary because the AGate keeps the state of a user session (for example, login data and the state of the SAP transaction) in the memory. If a subsequent request were routed to a different Web server (see dotted arrows in the figure below), it would end up at an AGate that does not have any information about the session. This would result in an error message.

Scaling the AGate



Most Web services are stateless, in contrast to the ITS. Many products that offer load balancing are not able to handle Web applications with a state. When you plan a large ITS service, make sure to select a technology that works with the ITS. The following sections discuss the options for achieving load balancing that fulfils the ITS requirement.

Another reason for using multiple ITS instances is to increase the availability of the service in case some components fail. We will also discuss the aspect of availability of the different methods.

Round Robin DNS

Round Robin DNS is an easy solution for achieving a limited form of load balancing for network services. The Domain Name System (DNS) is used to map the name of a server to an IP address. Round Robin DNS maps DNS requests to a defined set of IP addresses belonging to the available Web servers in a round robin fashion.

Round Robin DNS works for the ITS because the client makes only one request for the first connection and keeps this information in memory for a time which is typically longer than the duration of a user session. In the event of server failure, a method has to be set up to remove this server from the DNS. Clients will still try to connect to the failed server until the DNS information has timed out on the client machine. In the mean time the clients have no way to connect to the service. Round Robin DNS has no method of performing intelligent load balancing.

Redirection

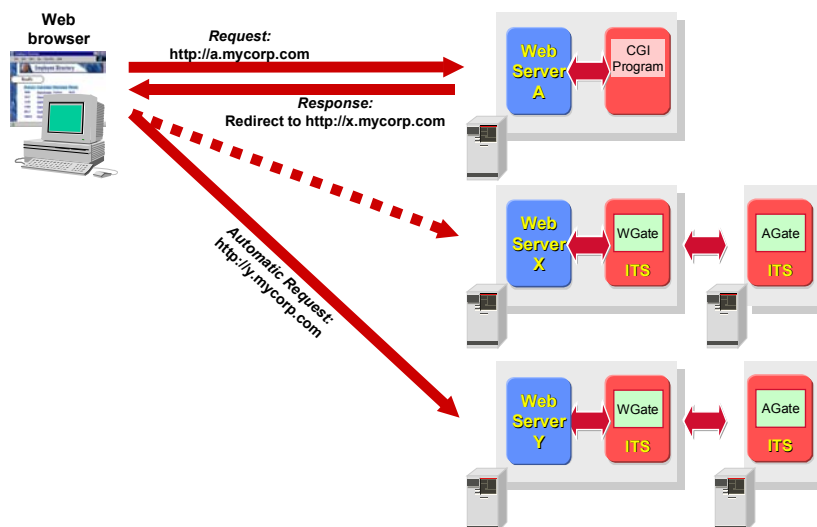
One Web server can be used to distribute requests among a set of other Web servers. To do this, users do not call the WGate URL directly but rather issue a request for a CGI program on

Scaling the AGate

the Web server. For this, a separate server or one of the WGate servers can be used. The CGI program redirects the request to one of the WGate services. A simple load distribution scheme would be to choose all available WGate servers in a round robin fashion. A more elaborate redirection program could check the availability and load situation of the servers periodically and select the server accordingly.

Redirections are supported by all modern Web browsers. It works transparently to the user in the following way: The browser makes a request to server A. Server A sends a response with a header that contains a redirection for example to server Y. When the browser receives this response it automatically repeats the request, but now to server Y, which in this case would be a WGate server.

The following graphic illustrates the procedure:



The disadvantage of this technique is that the URL displayed in the browser shows that redirection took place. Some users may create bookmarks on a transaction they use frequently and thus undermine the load balancing. To a certain degree this problem can be overcome by use of frames which hide the URL by which the ITS service is called.

Multiplexing

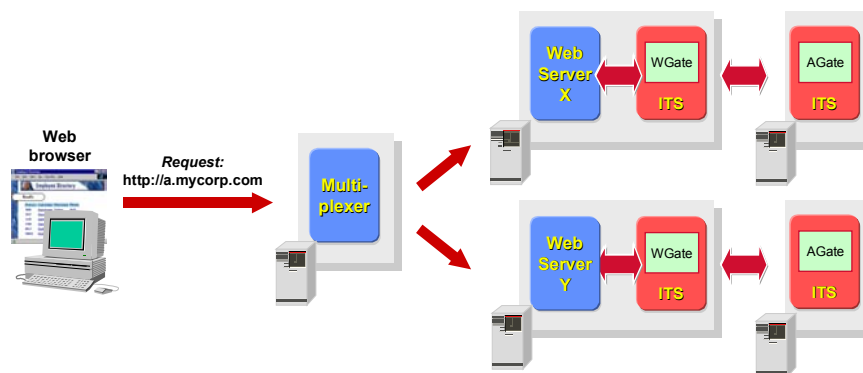
Multiplexing requests from one IP address to the servers overcomes this disadvantage, because it is completely transparent to the user. This can be realized by a network device that offers load distribution between the Web servers, for example a programmable router or a special multiplexer device.

If you are going to implement this technique you have to make sure that the device that handles network load balancing is able to fulfil the ITS state requirement described above. One option is

Scaling the SAP System

to route incoming requests based on the IP address of the client. Another criterion for selecting a product is the ability to allow intelligent schemes for load balancing and high availability.

The following graphic illustrates the technique. To the outside world, the multiplexer has the IP address A. It forwards IP packets to the Web servers X and Y which are completely hidden to the outside world.



Microsoft Windows NT Load Balancing Service

The Microsoft Windows NT Load Balancing Service (WLBS) does not work for AGate load balancing, because it "generally cannot be used to directly load-balance client requests across stateful servers" (cited from

<http://technet.microsoft.com/cdonline/Content/Complete/boes/bo/Winntas/technote/wlbsp.htm>).

Scaling the SAP System

The load that Internet Application Components impose on the SAP system is comparable to that of ordinary SAP Transactions. As in the case of normal SAP Transactions the load determines the response time of the system. Formal sizing tests are conducted by the SAP partners.

Logon Groups

ITS supports the normal SAP load balancing mechanism which uses the SAP Message Server to distribute the load between multiple SAP application servers. You should set up a separate logon group for Web access which is restricted to a set of application servers. This way you can protect internal users from possible overload from the Internet.

Scaling the SAP System

Offering a service in the Internet can lead to problems not usually present in ERP systems. One is that the number of potential users on the Internet is very large and the number of persons that actually use your service may vary quickly. To protect an internal productive SAP System from possible overload from the Internet, we recommend that you use a special login group for Internet access and restrict this login group to a subset of the Application servers.

Dedicated SAP System for the Web

Additional safety – both against overload and other security risks from the Internet – can be achieved by dedicating a special SAP System to the Internet service. This system could be coupled to the internal "main" SAP System by means of Application Link Enabling (ALE). This measure also enables you to increase security by isolating the Internet system in a separate network (for details see [ITS Security \[Page 100\]](#)).

ITS Security

This section describes those aspects of security for SAP Web applications that are related to network integration. This includes the network topology, firewalls, Web servers, communication encryption, and so on. You can find additional information, especially about application security, in the SAP Library under *BC Basis Components – Frontend Services* and in the SAP Security Guide (available in SAPnet under <http://sapnet.SAP-ag.de/securityguide>).

Introduction

When connecting to the **Internet**, you need to protect your system from unauthorized access and other intruders. Although we refer to the Internet throughout this discussion, **Intranet** applications may also require a certain degree of protection, depending on your security policy. The following information also applies to Intranets.

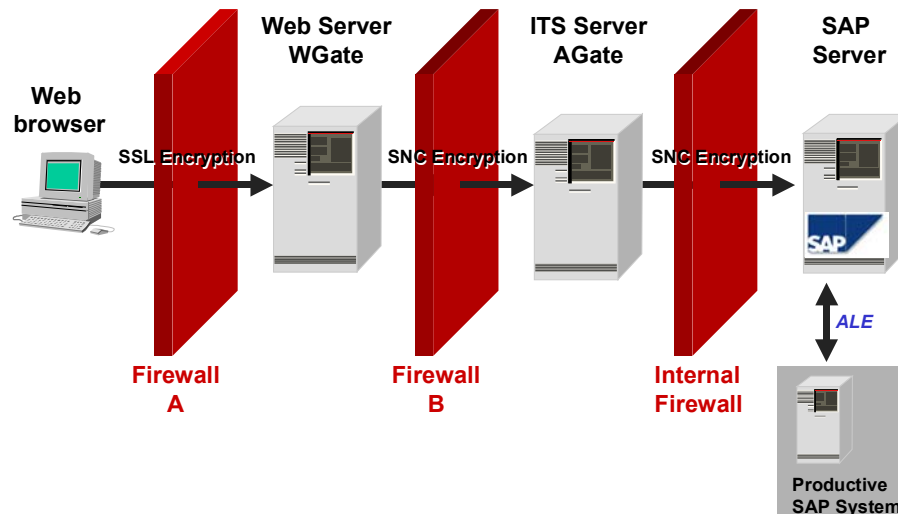
All security considerations require that you have a clear security concept. You cannot make any application absolutely secure. Instead you need to lay down the components of your application, then try to consider all possible threats (for example, unauthorized access and denial-of-service intrusions), define the severity of these consequences, and classify which are acceptable and which are not. You must have this concept in written form and refer to it for every security measure.

Secure Network Infrastructure for the ITS

The SAP Internet architecture has many built-in security features, such as being able to run the WGate and AGate on separate hosts. This is important because the WGate usually contains no sensitive information. The AGate, however, may contain information like SAP System passwords which need to be protected.

We strongly recommend setting up a network infrastructure that makes use of these features to control access from the Internet to sensitive application components and internal networks.

To do this, you can make use of specific security components, such as firewalls, packet filters and SAProuters to separate the individual parts of the network from each other. This ensures that unauthorized access is restricted to a small part of the system and cannot harm your internal network and the SAP System. The following graphic shows some of the components that you can use to build a secure network architecture when using ITS:



- Firewalls separate network segments from each other. Here, the term *firewall* is used in the broadest possible sense. It can mean anything from a commercial firewall product to an IP router configured with filtering rules.
- Cryptographic methods can be used on the whole route between the Web browser and the SAP System to protect the data from corruption and eavesdropping.
- To improve performance and reduce the amount of data available in a system opened for Internet access, you can use a separate system (coupled using Application Link Enabling, ALE) for your Internet system, instead of your productive system.

You may decide to implement some or all of these components depending on your security policy. You can find details about the components and a concrete example of a network setup in the sections below.

Security Components

Our recommended network topology consists of three separate network segments that are connected by two firewall systems. The least secure network is the Internet itself. This is where the Web browser is located.

Protecting the Web Server

The Web server is protected by the first firewall system. The Web server must be protected against any kind of network packets that are not needed for HTTP communication. To accomplish this, configure the router to pass packets to the corresponding TCP port only.

ITS Security

Usually a Web server requires one TCP service. Port number 80 is reserved for HTTP and used by default by all servers and browsers. HTTPS (HTTP plus the Secure Sockets Layer protocol), uses TCP port number 443 by default.

Configure the Web server operating system to be as closed and restrictive as possible. Disable any unnecessary network services.

Protecting the Internal Network and AGate Server

We strongly recommend taking additional measures to separate the Web server from your internal corporate network, such as using a firewall system and/or SAProuter. This prevents additional damage if an intruder manages to gain control of the Web server. We recommend keeping the AGate in your internal network.

The connection between WGate and AGate can be relayed by a SAProuter. For a detailed description of the SAProuter functionality and administration, see the SAP Library under *BC SAProuter (BC Basis Components – Client Server Technology)*. You can configure the SAProuter to relay only the WGate–AGate connection and deny all other connection attempts. See SAP Online documentation *ITS Administration Guide* on how to configure ITS with SAProuter (*BC–Basis → Frontend Services → Internet Transaction Server*).

You can use other firewall products to relay the TCP connection from the WGate to the AGate. See the documentation for your firewall for further information.

Protecting the SAP Server

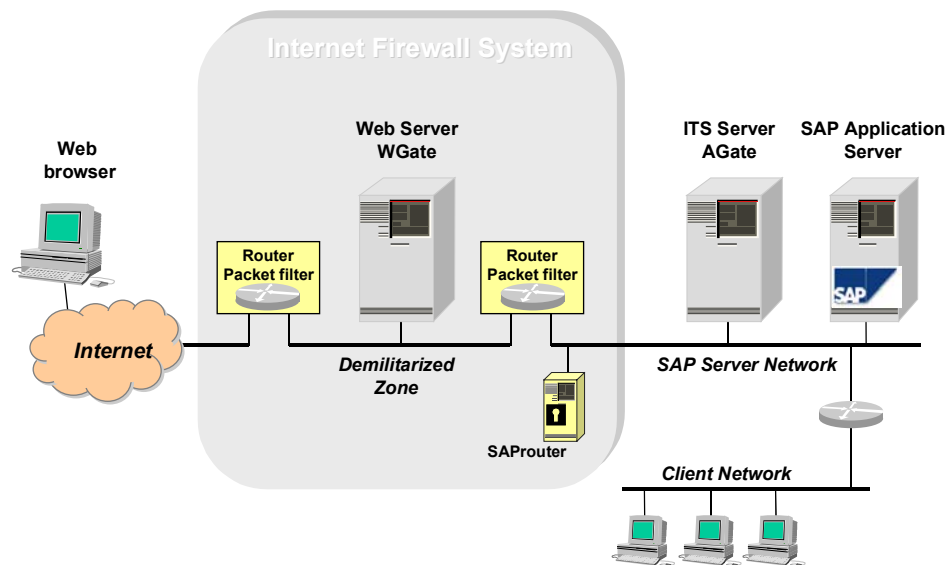
For very high security demands, the SAP server network can be protected with a firewall and SAProuter as described in [Network Security \[Page 124\]](#).

To protect the AGate from unauthorized access in the internal network, place the AGate in the server network. In this way, the AGate is protected with the same mechanisms that protect your SAP servers.

To achieve maximum protection from the Internet, place the AGate outside of the server network. Because the AGate communicates with the SAP application server in the same way as the typical SAPgui (using DIAG or RFC) no special measures are necessary for the connection between the AGate and the SAP System.

Example of a Network Configuration

The following graphic shows an example network security infrastructure suitable for Internet access to SAP using the SAP Internet Transaction Server:



The security of the network zones increases from left to right. The first router/packet filter allows direct access from the Internet to the Web server's TCP ports only. It never routes any packet directly to the second router, so no packets get directly from the Internet into the internal network. This is why the Web server network is called the *demilitarized zone*. The second router/packet filter is configured to deny any direct access from the Web server network to any host in the corporate network except for the SAProuter (TCP port and host). Therefore, the connection from the WGate to the AGate can only be transmitted through the SAProuter. In the example, the AGate is located in the SAP server network. The server network is connected to the client network through a router that may also provide access restriction.

If you want, and as mentioned in the previous section, you can also place a router/packet filter between the AGate and the SAP application server. This provides additional protection if an intruder manages to exploit any security hole of the AGate or the operating system it is running on.

In the above example, we have displayed a setup using standard network and computer components. Many vendors offer specialized firewall products for these tasks. You may use such products; however, we do not describe them in more detail here in this guide.

Encrypted Communication

You can use cryptographic methods to increase the security of all network connections involved in an ITS application.

Encrypted Communication

Browser – Web Server

All data (including passwords) is usually transmitted through the Internet without being encrypted. To maintain confidentiality for this data, you can encrypt the connection between the Web server and browser. The SAP-supported Web servers and all modern browsers support the encryption of the HTTP data stream by using the Secure Sockets Layer protocol (SSL), also known as HTTPS. HTTPS data streams are completely transparent to the ITS. For more details regarding encryption techniques, see the documentation supplied by the Web server manufacturer.

To use SSL encryption, the Web server must obtain a certificate for its public key. This certificate is referred to in this section as the **server certificate**. It is used by the browser to authenticate the server, and is issued by a Certification Authority (CA). If the browser receives a server certificate issued by a trusted CA, then the browser can verify that it is connected to the intended server. This is the precondition for the secure connection to be established.

If you want to offer a service to all Internet users, this server certificate must have been issued by an official CA that is trusted by most browsers used in the Internet. For internal users, you can set up a corporate CA and configure the browsers to trust this CA.

WGate – AGate

By default, the data sent between WGate and the AGate is sent as clear text. You can choose a different connection type which encrypts the data with a DES algorithm using a static key. This key is not configurable; therefore, this encryption provides protection only against accidental reading of the data, but not against serious attacks.

For better protection you can use SAP's Secure Network Communication (SNC) to protect the link between the WGate and AGate. SNC uses an external security product to apply encryption to communication links between components of an SAP System. It offers different levels of protection, depending on the cryptographic product you use. See [Network Security \[Page 124\]](#).

AGate – SAP System

The connection between AGate and SAP System can also be protected with SNC.

Authenticating Users

Service Users

In the **Internet** scenario, you do not necessarily know which users want to access the application data in the SAP System. In addition, due to the large number of Internet users, you normally cannot set up a separate account for each user. Therefore, if you want to make certain Internet Application Components available to anonymous Internet users, you normally set up service users with predefined passwords in the SAP System. These users should have only the permissions necessary to access the application (for example, a product catalog). The corresponding ITS service is configured with this logon information to include the password. If you need to further authenticate the Internet user (for example, when an order is placed), the application itself must perform the additional authentication. (For example, the application verifies the customer number and password using corresponding function calls.)

Encrypted Communication

For each application that is accessible over the Internet, you must create a service file, which is located on the AGate. This file contains information that is specific to the application (for example, the SAP transaction to start, the SAP client, the service user name and password to use). The ITS needs this information to run the application. The password does not appear as clear text in the service file, but is encrypted using a static key. Therefore, if you have service users with authorizations that are worth protecting, then take special care to protect the AGate service files from unauthorized access.

ITS does not store any security-relevant information on the WGate.

Named Users

In an **Intranet** or **Extranet** scenario, users can log on to SAP over the ITS using their SAP user name and password. User names and passwords are not stored permanently in the ITS in this case. User authentication takes place entirely within the SAP System. Internet Application Components are subject to the usual SAP authorization concept, just like any other SAP transactions.

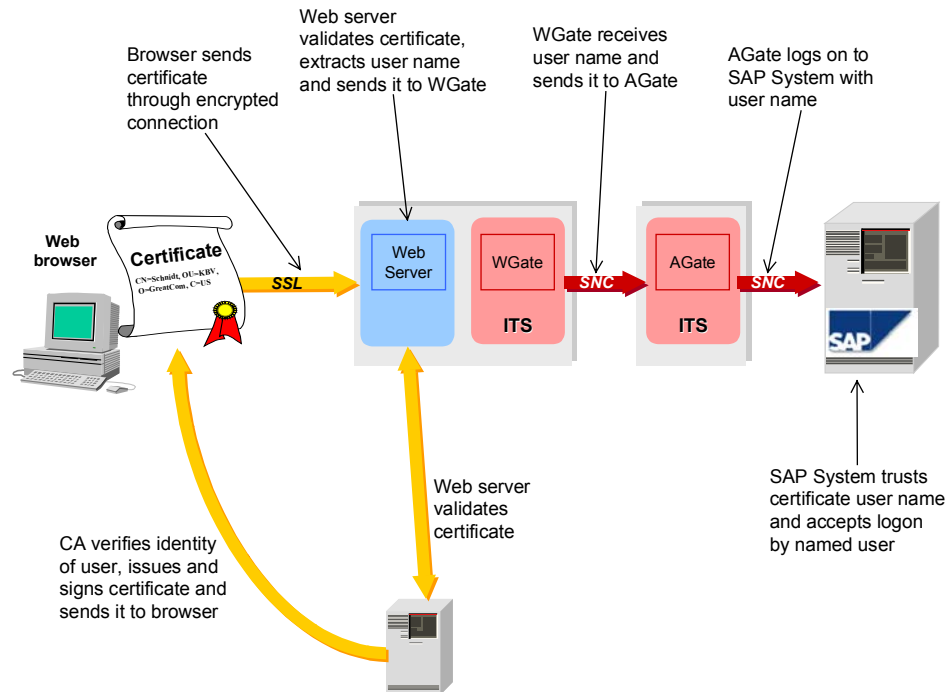
Named Users with Browser Certificates

Certificates on the client browser can also be used to identify the user in the SAP System (R/3 Release 4.5 or higher). This eliminates the need for the user to enter a logon name and password. It can be used for all IACs that require a named SAP user. A description of the basics of this technology follows. See *X.509 Certificate Logon over the ITS* for details (in SAPnet <http://sapnet.sap-ag.de/systemmanagement> → *Media Center* → *Security* → *Literature*).

The most common form of client certificates use the X.509 standard. Certificates are issued by a certification agency (CA) and installed in the browser. The Web server verifies the certificate's validity and extracts the user name (also called a *distinguished name*, or *DN*). The certificate is passed to the SAP System which logs on the corresponding user without password check. To ensure that the DN can be trusted, secure network connections have to be used along the entire path of communication.

You can either use a commercial CA (for example, Verisign Inc.) or set up your own CA.

ITS Session Integrity



Web servers can be configured to accept only connections that present a valid certificate. This can be used as an access control mechanism, especially if only certificates issued by a certain (for example, your own) CA are allowed.

ITS Session Integrity

Session Identifier

To maintain the integrity of the multi-step transactions when using IACs, the ITS issues a unique session ID number when the first request is made by the user. This session ID is sent to the browser with the first HTML page. It must be passed back to the ITS with every successive request. The session ID is used by the ITS to identify the correct session context. It also ensures that another user cannot easily take over a current session. If you need even stronger protection of user sessions, you can use HTTPS.

Cookies

Old versions of ITS rely on HTTP cookies to store the session ID. In current ITS versions a cookie is needed only for user identification across multiple services. If you do not need this feature, you can disable the sending of cookies.

Client IP Addresses

As an additional security feature, the ITS stores the client IP address along with the session ID. A possible eavesdropper, who listens in on the network connection and thus acquires the current

session ID, cannot easily issue a fake reply. (Their IP address does not match that of the original user.)

You can configure the number of significant bytes used for the network address comparison. On the AGate, the following registry key specifies a mask of the significant network address bits:

```
HKEY_LOCAL_MACHINE\SOFTWARE\SAP\ITS\2.0\<INST>\Connects\IPChecking
```

The default value is 255.255.255.255, which specifies that the entire address should be compared. Enable this value for an **Intranet** solution. For **Internet** applications, we recommend entering a value of 255.255.0.0. In this case, only the leading figures of a network address are compared. This allows clients who use multiple Web proxy servers with load balancing to access the ITS.

Robot Exclusion

Search engines are a valuable tool for every user of the Internet. They allow you to search a large portion of all existing Web pages for words or phrases to find some piece of information. In order to build their indexes, most search engines traverse the Web by following all links in all Web pages they find. A program that does this is called a "crawler", or more generally, a "robot". The crawling procedure is very effective, but there are some problems. If a link directs the robot into one of your Internet applications, the robot follows all links into the application. But since the information is dynamically generated, it is not suitable for inclusion in a static index. Also the robot's requests impose additional load on your server.

The *Standard for Robot Exclusion* (SRE) provides a method to exclude robots from certain parts or all of a servers resource space. There is no easy way to prevent a robot from disobeying this exclusion, but most robots adhere to the standard.

Before accessing a site, an SRE-compliant robot checks the file `robots.txt` in the root directory of the server (for example, `http://www.yourcompany.com/robots.txt`). This file contains rules that tell the robot which parts of the server are accessible and which are not. If you run an SAP ITS, your `robots.txt` must contain at least the following lines:

```
# This is a comment line
User-agent: *
Disallow: /scripts
```

This way the robot stays away from the Internet Application Components on your server. You could also disallow your entire server by specifying `Disallow: /`. But you may want to allow the robots access to the general part of the server so that prospective customers can find your service through the search engines.

Note that robots are not just an issue in the Internet, but also in corporate intranets where search engines are often used to build an index of corporate information.

Robot Exclusion

Web Client Connections

Your clients have to be able to connect to the Web server that runs your Internet Application Component in order to use it. This requirement is easy to meet within local networks. It requires more effort to connect clients that are not part of your organization. The following sections discuss some methods of making client connections.

Corporate Intranet

LAN

Client connections are not a big issue in a local area network (LAN) environment. Typically the network architecture contains a backbone that connects the client networks with the servers in the data center. Network bandwidth is usually not a problem in the LAN environment.

WAN

Wide area networks (WAN) are needed to connect subsidiaries or remote workstations, such as home PCs or mobile clients, to the corporate data center.

The Web servers are usually located in the data center. If clients access the system over Wide Area Networks, the Web server can also be located closer to the clients (for example, one Web server in each building or subsidiary). In this case, the client does not use the WAN, but the WGate. The WGate resides on the Web server and has to connect to the AGate through a WAN connection. Even the AGate can be in the client network. In this case the ITS accesses the SAP System over the WAN connection just like the SAPgui does. You can select the setup that best serves your needs. For details about the different network connections see [ITS Network Connections \[Page 90\]](#).

Internet

Internet Service Provider

If your site is not directly connected to the Internet, an Internet Service Provider (ISP) is needed to offer a service on the Internet. The choice of the Provider may be crucial for the quality of service that you can offer to your customers. It may also have a sizeable impact on the cost of that service.

Make sure that your provider is capable of supplying additional bandwidth in a flexible manner in case your site suddenly becomes more popular. Overloaded network connections are unsatisfactory for your customers.

Some ISPs offer additional services, such as a firewall to protect your Web server.

Extranet

Extranet

An Extranet is a connection with a defined group of external communication partners who use Internet technology, possibly even the Internet infrastructure itself. This is typically the case for Business-to-Business communication.

Web applications are a convenient and increasingly important way of connecting businesses. For example, you may want to share stock information with your partners or enable automated procurement. You can use an Internet Application Component (either supplied by SAP, or your own development) to do this, so that your partners can use the application with any Web browser and need not install any special software or other technical infrastructure.

With respect to the network installation, you need to connect your partners to your Web server so that they can use your IACs. It is usually important to authenticate the users in a reliable way so that the information is not accessible to unauthorized parties. Protection of data against eavesdropping and modification by unauthorized parties is also an important issue.

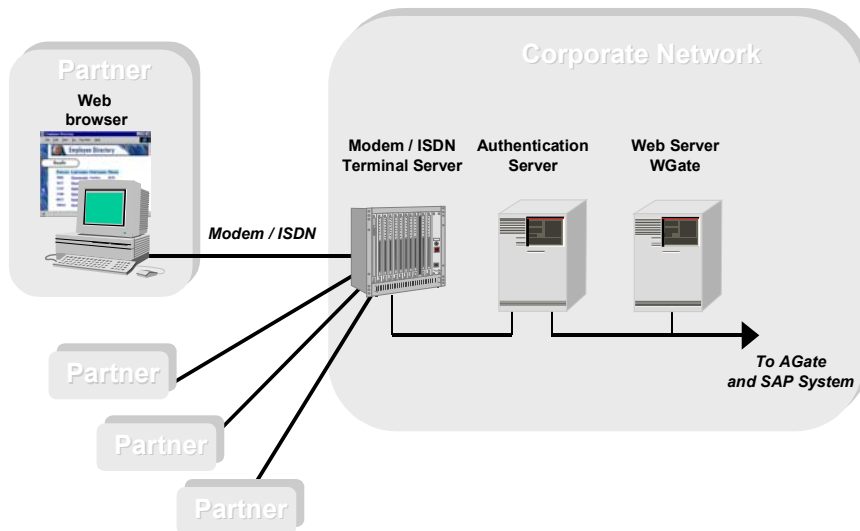
You can use any kind of network connection to give your business partners access to your Web server. An overview of connection types is given in [SAP Communication in a Wide Area Network \[Page 117\]](#). The following sections discuss some methods of connecting your business partners to your SAP Internet Transaction Server.

Remote Access Server

One way of connecting your partners is a remote access server where users connect by using modems or ISDN. Every partner opens a separate connection to your server. This means that your server needs as many ports as the maximum number of partners that access your system at the same time.

You can use strict authentication methods with one-time passwords to control access to the server. SAP access then needs an additional password for user authentication.

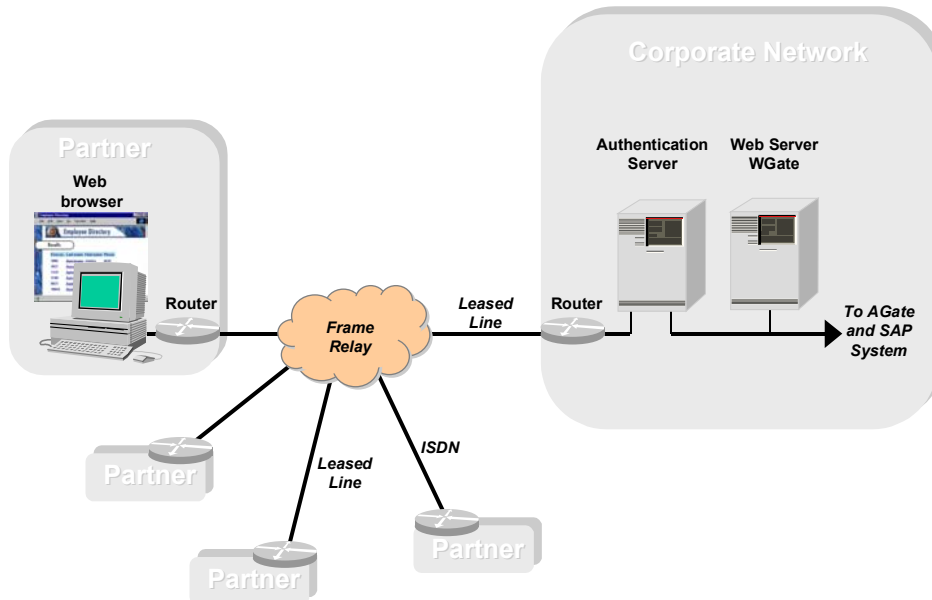
This solution has the advantage that you rely only on the phone system as a carrier. The technology needed to access your system is inexpensive, widely available and easy to configure. Security devices for one-time passwords can be easily distributed to your partners. The following graphic shows a possible setup:



Frame Relay

If you need reliable access with greater network bandwidth you could use frame relay to connect your partners. Another advantage is that you need only one connection to the network provider. All communication partners need a connection to the frame relay provider. A virtual connection with every partner has to be established. For authentication of the communication partner and protection of the data you can use routers that support encryption, for example, the IPsec standard. The following graphic shows a setup of this type:

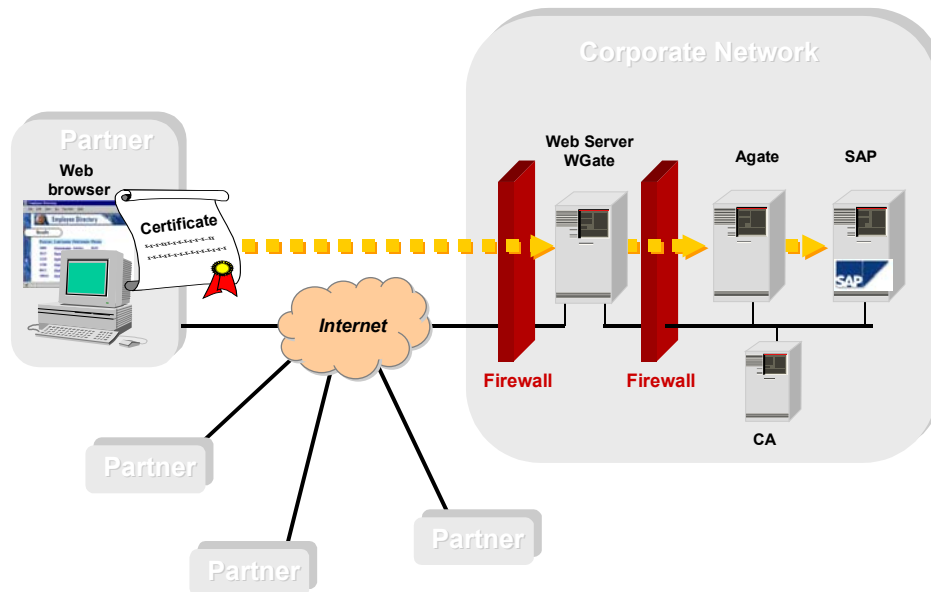
Extranet



Internet

You can also use the public Internet to offer a service to your business partners. Your communication partners need only an Internet Service Provider (ISP) to connect to the Internet temporarily. You need a permanent connection in order to offer the service on the Internet.

Security considerations are crucial when you use the Internet to connect external clients to your data center. You could use browser certificates to identify users. No other access control is used, which saves the need for a special authentication server. To achieve this, the certificate is used both to control access to the Web server (only users with valid certificate are allowed) and for authentication in the SAP System. You can either trust a commercial certification agency (CA) to authenticate your partners and issue the corresponding certificates or you can install your own CA and distribute the certificates yourself. The following graphic shows a setup of this type:



The certificate is received by the Web server which verifies its validity and extracts the unique name of the user. It passes the name over the WGate to the Agate, which sends it to the SAP System. The user does not have to enter any authentication, such as a name or password. This technology requires SNC-protected communication between the WGate, AGate and the SAP System.

Alternatively, you can also use other security mechanisms, such as IPsec, to set up a secure network connection (called Virtual Private Network, VPN) over the potentially unsafe Internet.

ITS System Landscape

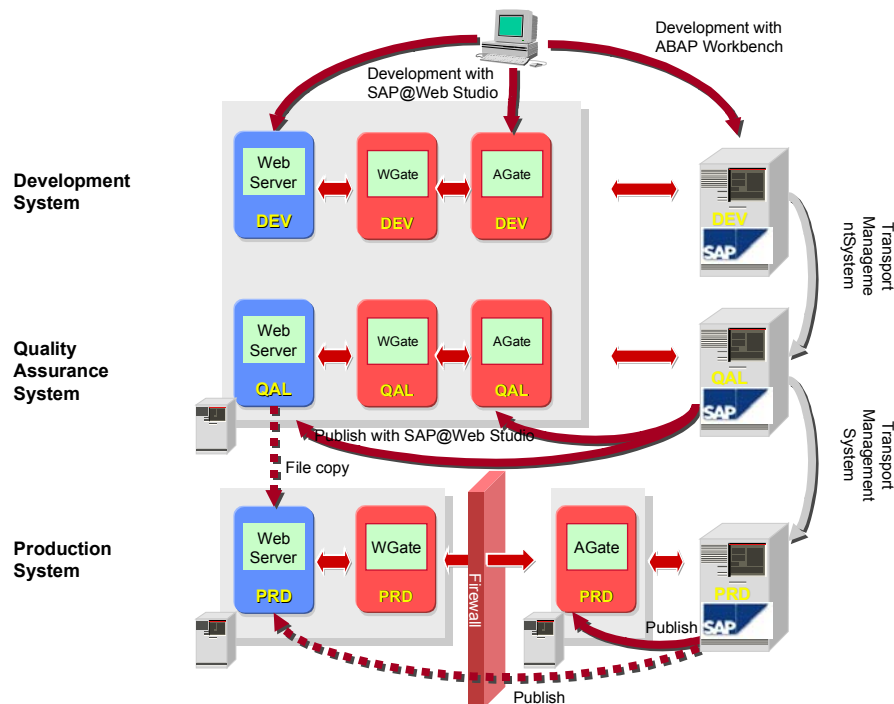
ITS System Landscape

SAP development usually takes place in a dedicated development system. This applies to new developments as well as changing the SAP standard applications. All objects that belong to an application are managed by the Transport Management System (TMS). Once development is complete, all changed or created objects are transported into a quality assurance system and on to the productive system.

The Internet Transaction Server (ITS) is completely integrated with the SAP Transport Management System. Web objects, such as HTML templates or graphics, are stored in the SAP System. They can be checked out to the local file system for development purposes. This is done with the SAP@Web Studio desktop application. All transports are made within the SAP System to prevent Web objects and business application from becoming inconsistent.

The Studio also gives you the option of copying a set of Web objects from the SAP database to the ITS and to the Web server on which you want to run the application. This is necessary because the Web server and the ITS read Web objects from the file system only.

As a general rule, every SAP System needs a separate ITS instance. For stability reasons, productive SAP systems and ITS instances should always be located on dedicated computers that have no other functions. The following figure shows a possible system landscape for the development of Web applications:



In this example, the system load of the development and test systems is low, so the Web servers and ITS instances are located completely on a single computer. The Windows NT 4.0 operating system must run on this computer. The entire ITS infrastructure consists of only three computers,

two for the production system and one for development and test. For large scale tests, for example performance analysis, you may need additional resources.



Note

In this setup the development and test ITS always have to have the same release, because when being installed on the same computer they share the same executable programs. This is not usually a severe problem, because of ITS backward compatibility, but you should keep this fact in mind when planning the setup.

On their desktop PCs, the developers use SAP@Web Studio for developing the Web objects and SAPgui for application development in the SAP System. SAP@Web Studio needs an RFC connection to the DEV SAP system. Development is done in ITS instance DEV. For this, all developer desktops need to use NetBIOS Network Shares to access files on the server.

All developed objects are transported to the quality assurance system QAL by the Transport Management System. After this, the Web objects can be copied to the server remotely by using network shares, or locally with the SAP@Web Studio.

You can install an additional ITS instance for the ITS administration interface on the development and test computer. This is not shown in the graphic.

The production ITS is a dual-host installation for security and load reasons. Transports into the SAP System and the AGate computer work in the same way as in the development system. The Web server, however, is usually located behind a firewall system and is not accessible from any other computers. You can copy Web objects to the production Web server in one of two ways, depending on your security policy:

- File copy from the QAL Web server. If no other way is allowed by the firewall, you can use removable media. Advantage: no additional communication channel is needed.
- If the Web server runs the Windows NT operating system, SAP@Web Studio can be used on this computer to fetch the objects from the SAP System. For this you need an RFC connection to SAP which has to go through the firewall as well. Advantage: more convenient.



Note

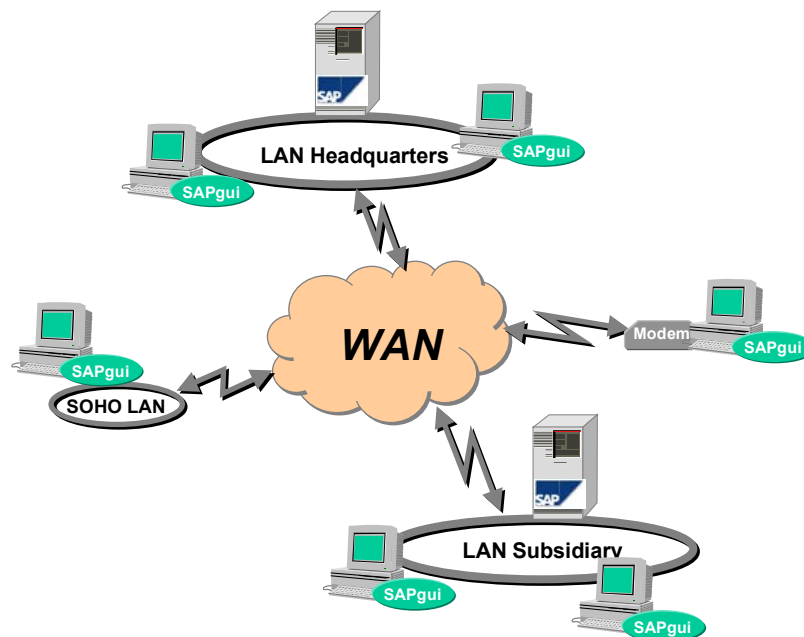
Service files (see [ITS Technology \[Page 80\]](#)) contain ITS configuration information, such as the SAP System that the service must connect to. On the other hand, they are also Web objects and as such are subject to the Transport Management System. This means that the service file from the test system could be transported into the productive system which would then connect to the development system instead of the production system. To avoid these problems you may have to edit the files manually after a transport, or exclude them from transport altogether.

ITS System Landscape

SAP Communication in a Wide Area Network

With the increasing globalization of companies, greater mobility of employees and flexibility of working conditions (for example, working from home), wide area networks (WANs) are becoming ever more important. WANs enable access to IT resources in locations that are outside the local network.

WAN connections are used for all types of communication: for SAP frontends, for program-to-SAP communication and for communication between SAP Systems. The following graphic shows three typical scenarios in which communication partners use WAN connections to communicate with the SAP System:



- Single PCs use modems, ISDN or other methods of transmitting data to connect to the center. This variant is used mostly by home workstations or portable terminal PCs.
- A large group of communication partners, in a subsidiary for example, usually has its own local network. This LAN is linked to the headquarters by a WAN connection. The LAN-LAN link is also often used for communication between different SAP Systems.
- A "Small Office/Home Office" (SOHO) is a level between the first two variants. It comprises a small office or home workstation that has its own local network. This network often consists of just the PC itself and a router that connects to the corporate network through ISDN or other services.

The following briefly describes the most important types of connections (modem, ISDN, leased lines, frame relay, X.25 and the Internet). We also discuss the criteria you should consider when you choose a connection:

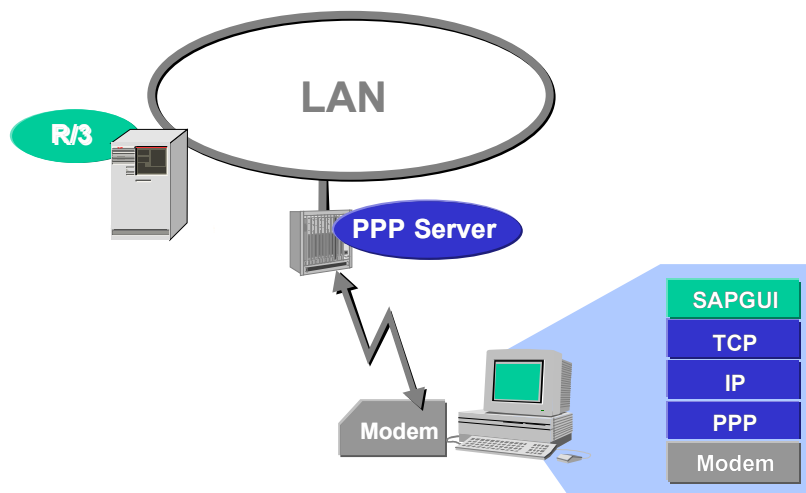
- Transfer rates

SAP Communication in a Wide Area Network

- Line costs (these may depend on the exact location of your communication partners)
- Availability
- Security against downtime
- Technical ease-of-use and ease of securing the connection

Modems and Telephone Lines

You can use modems and analog telephone lines to connect PCs to a remote network. Here as well, the SAP frontend software uses the TCP/IP protocol to communicate with the R/3 System.



The modem can be connected directly to the SAP server, or to a special terminal server. We recommend the second option for security reasons. Another protocol is needed in the data link layer to transport IP data packets with the modem. You could use SLIP (Serial Line Internet Protocol) or PPP (Point-to-Point Protocol) here.

SLIP was the first protocol that allowed IP packets to be sent using a serial line. This protocol is outdated and is used rarely due to its awkward configuration.

PPP has become the standard instead of SLIP, since it contains improvements in configuration and transmission security. Unlike SLIP, PPP is also able to communicate with other protocols in the transport layer and is not limited to TCP/IP.

You can use both protocols to connect SAP frontends. PPP and the software that controls the dialing process of the modem (*dialer*) are usually contained in modern TCP/IP software.

Ensure that modem lines always run on a dedicated machine, such as a specialized terminal server that can serve a large number of lines. The number of available modem ports determines the maximum number of parallel partners that you can connect to.

SAP Communication in a Wide Area Network

A switched telephone line is not suitable for making a LAN-LAN connection due to its narrow bandwidth. Instead choose either a fixed connection (frame relay), or another connection type (for example, ISDN or X.25).

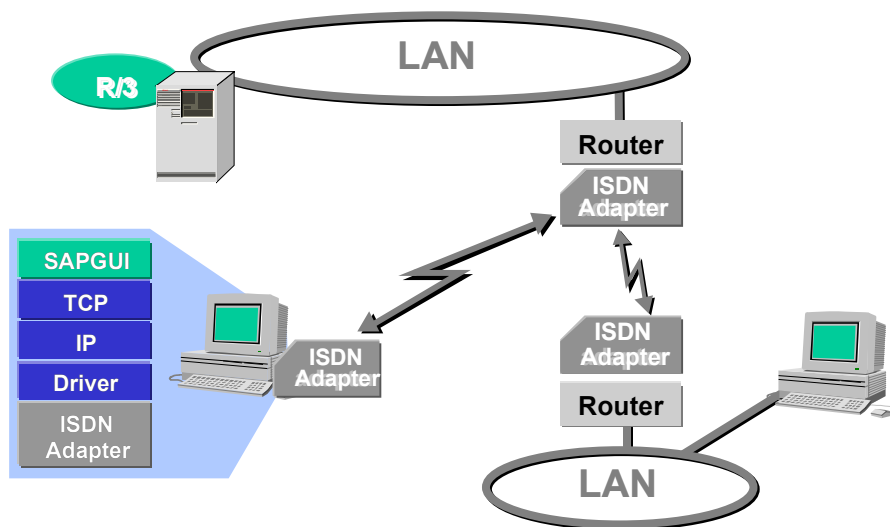
The cost of the lines for modem connections are the same as those for a telephone connection, and therefore depend on the distance and time used, but are independent of the volume of data transmitted.

ISDN

Integrated Services Digital Network (ISDN) is a digital variant of the classic analog telephone connection, and the configuration is therefore similar. An ISDN adapter is used as the terminal instead of the modem.

You can connect the ISDN adapter to the SAP server directly, or to a special terminal server. We recommend the second option for security reasons. The number of available ISDN channels limits the number of parallel, existing connections (each connection usually uses one channel).

The advantages of using ISDN in an R/3 environment are the better quality of the transmission lines, and higher transfer rates (64 kbps for each channel). Therefore, ISDN is well suited to a LAN group between multiple locations. Routers with ISDN adapters are used here. Many ISDN routers offer additional functions such as bundling several channels to increase the bandwidth, or a temporary hold of connections during longer periods of inactivity (*Short Hold Mode*). The costs of ISDN are like those for an analog telephone connection, which are dependent on the distance and the time used, but independent of the volume of data transmitted.

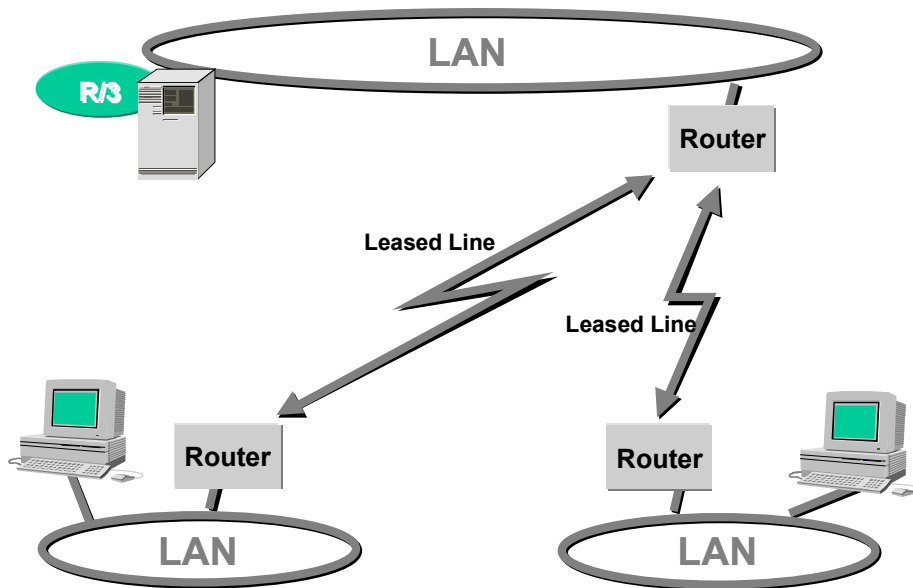


Leased Lines

Leased lines connect two local networks. In most countries, you can lease lines from various network providers. Special routers that connect to the leased lines of the network provider are used as the terminal. The leased lines open a point-to-point connection between two routers at a

SAP Communication in a Wide Area Network

specific data rate, for example, 1.5 mbps (million bits per second). The leased bandwidth cannot be exceeded. The price of leasing lines is high and depends on the transmission rate. Therefore, you must use the available bandwidth effectively.



Frame Relay

Frame relay is a single protocol in the data link layer that transmits data packets over a long distance network. It does not offer error recognition or flow control. The advantage of frame relay compared to leased lines is that point-to-point connections do not have to be made to all communication partners. You only need one line to the network provider, which relays the data packet to the receiver. The receiver ID is in the header of the data packet. To do this, *Permanent Virtual Circuits* (PVCs) are set up. A PVC corresponds to a virtual leased line for one communication partner.

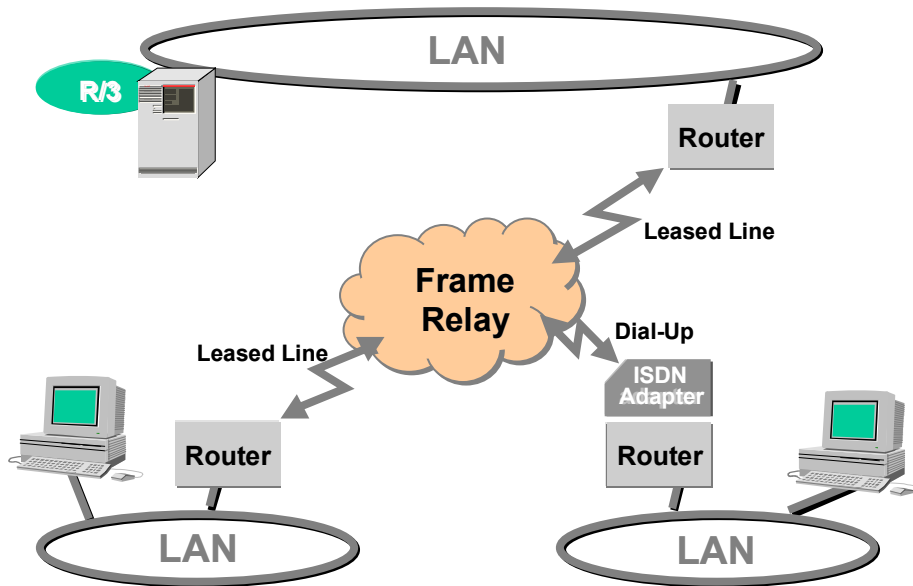
Switched Virtual Circuits (SVCs) let users of frame relay extend their PVC networks dynamically, and if needed, set up logical network connections that lead to end points in the same network, or that pass through gateways leading to end points in other networks. SVC is a relatively new technology and is still not widely used.

Frame relay networks are usually accessed over leased lines or ISDN. Special routers that can use the relevant network interfaces are used as the terminals (*Customer Premise Equipment*, CPE).

For each virtual circuit, frame relay guarantees a specific data rate known as the *committed information rate* (CIR). For example, the transfer rate for a connection between two large branches can be 1.5 Mbps, but the connection of an external site is only 128 kbps. The data shares the same line going to the frame relay service provider. A higher data transfer rate than the CIR can often be reached, but this is not guaranteed.

SAP Communication in a Wide Area Network

Frame relay is an efficient and relatively inexpensive way of transmitting data and has become increasingly popular in the last few years. The costs of frame relay depend on the CIR, the actual volume of data, and other factors.



X.25

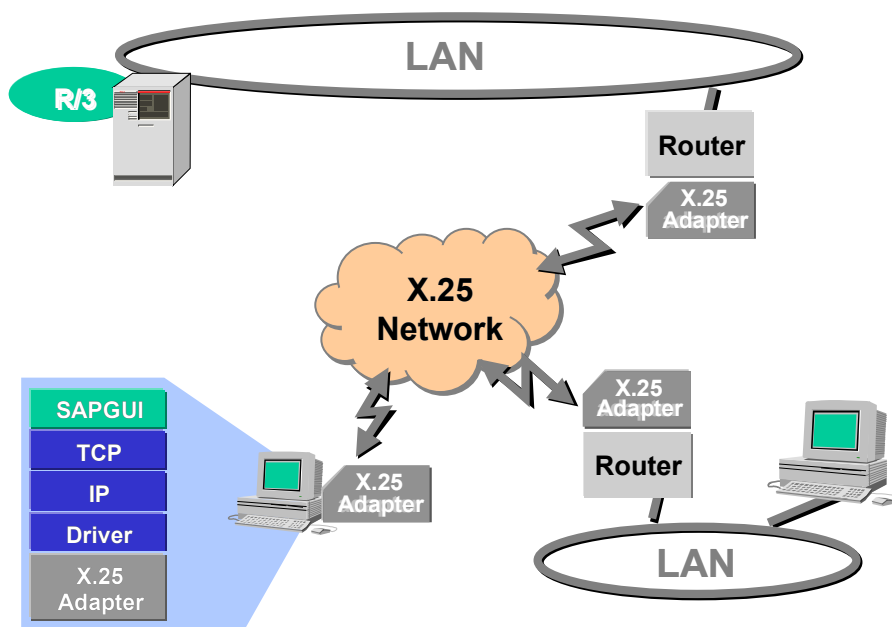
X.25 (Datex-P in Germany) is a packet-oriented service. The costs depend on the volume of data transmitted, and therefore on the number of packets transmitted. Unlike modem or ISDN connections, X.25 is not dependent on time or distance. X.25 can be seen as a forerunner of frame relay although it has many more options and is considerably more complex. The higher costs of X.25 have meant that it has lost in importance to frame relay.

X.25 lets you have two different connection types:

- Dialed connections (SVC, switched virtual circuits)
- Permanent connections (PVC, permanent virtual circuits)

SVCs enable a connection to be made to different partners. With PVCs, the connection is always made with the same partner. The advantage of this connection type is that establishing the connection is faster. The transmission costs depend on the volume of data transmitted, the transmission speed, and the connection type (SVC, PVC).

SAP Communication in a Wide Area Network



Internet

You can also use the Internet to connect to an SAP System. The Internet has a number of advantages: it can be accessed from many locations and it has low costs. On the other hand, both its security and availability are subject to limitations. These problems are mainly due to the fact that the Internet is an open network that is not administrated centrally.

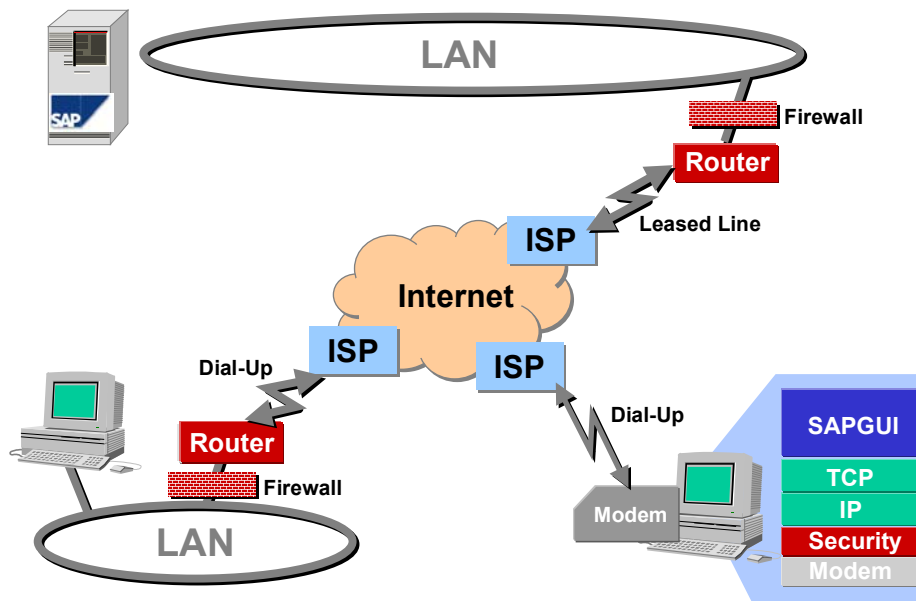
To connect your network to the Internet, you need an Internet Service Provider (ISP) that makes the connection. To offer a service, you need a fixed IP address in the Internet and a permanent Internet connection. The clients can then use a local ISP to dial into the Internet.

If you want to use the Internet to communicate with an SAP System, you must make sure that the system is protected from unauthorized access. If required by the application, you must also secure communication between the SAPgui and the SAP System. There are special measures that you can take to make your system secure. This requires a lot of technical work and administration, however you can give this task to a suitable third party (such as your ISP). The different ways of securing your SAP Internet connections are described in the section [Network Security \[Page 124\]](#).

The security of your network connection against failure is a general problem, since an ISP can usually only guarantee those network connections that it can influence. You must decide how critical it is for your application if your network connection were to be broken temporarily.

The following graphic shows a scenario in which SAP frontends are linked through the Internet. It includes examples of some of the security measures you should consider when planning to connect to your SAP System over the Internet:

SAP Communication in a Wide Area Network



Network Security

Network Security

Your network infrastructure is of great importance for the security of your system. It has to allow your application to communicate, while at the same time preventing unauthorized access.

A well thought out network topology is a basic requirement for using effective security mechanisms. It may be difficult to revert some fundamental decisions, so you must include security aspects in your planning right from the start, even if they are not of immediate importance.

When you plan your security mechanisms, it is important that you have a clear picture of the resources you want to protect, and can estimate the possible risks. When you think out your security strategy you need to weigh the possible damage that unauthorized access could cause to your system against the expense of avoiding this danger. You must be fully aware of your own security strategy and make sure that it exists in written form.

Your security strategy must cover all layers of your system:

- **Application:** The SAP System offers a range of mechanisms, such as authorization checks.

This documentation does not discuss the security aspects of the application. For more information see the *R/3 Security Guide*.

- **Server:** The operating system configuration of the SAP servers must be secure. This includes all network services (deactivate any services that are not needed, such as `sendmail`), remote login, password administration, the Network File System (NFS) and the Network Information System (NIS).

This documentation does not discuss the server configuration in any more depth. For more information see the *R/3 Security Guide*.

- **Network:** The network must allow authorized access to resources, but must prevent unauthorized access. You must protect all SAP servers in a system where security is critical. The database server contains all application data and is particularly important. However, the application servers are also just as relevant, since data can also be accessed here.

The security of the SAP servers is usually part of a comprehensive strategy for securing resources in the network. The section [Controlling Access \[Page 126\]](#) tells you how to integrate the protection of SAP servers into this strategy, and which tools are provided by SAP.

The network connection between the SAP frontends and the application servers transmits all the data that the user enters or displays. This data clear text, however it can be extracted. There are several different methods of encrypting this data if you want it to remain secret. The same applies to other SAP communication connections, such as CPI-C, RFC and the Internet Transaction Server.

Encryption is usually mandatory for connections across open networks, such as the Internet. However, it may also be necessary to use encryption in internal networks for authenticating users and protecting sensitive data.

The section [Encrypting SAP Network Connections \[Page 133\]](#) describes the ways in which you can make sure that your communication connections are secure.

Controlling Access

Controlling Access

Access Control Functions

Access control for SAP connections must perform different functions depending on the SAP component being used:

- The SAPgui and other frontend programs use the dispatcher, gateway and the message server. Frontend computers in the frontend network must only be able to open connections to these TCP ports on the application servers.
- You must prevent network connections from the frontend network to the SAP servers. This protects the servers against unauthorized logins with Telnet or from other sources.
- If the database is on a dedicated computer, other frontend computers must not be able to set up a connection to it. These connections are not necessary anyway, since the database is addressed only by the application servers.
- Printers are usually located in the frontend network. To print, the SAP server must set up a TCP connection with the printer. More complex output management systems may have other requirements.
- Access control must permit the SAP System to connect to any other network services that it needs to use in other parts of the network. These could include the Domain Name System, mail, directory services, and so on.

You can use a management console to administrate the computers and resources in the server network. This console must be located in a secure room in the computing center. It has unrestricted access to all computers and network components in the server network.

Multilevel Controls

If you want to place strict access controls on your system, then you must do this on a multilevel basis. In this way you can make sure that the failure of a single control does not leave the whole system unprotected. A series of security systems makes it more difficult for any potential threats to get through to protected systems. Another advantage of multilevel security is that the system is secure, even if an individual component is incorrectly configured. Poor configurations are one of the main causes of unauthorized access to systems.

You can control network connections into and out of the server network in several different ways. The following discusses three options, of which only the third is multilevel and can be recommended for use in area where security is critical:

- Packet filter
- SAProuter
- Firewall with SAProuter

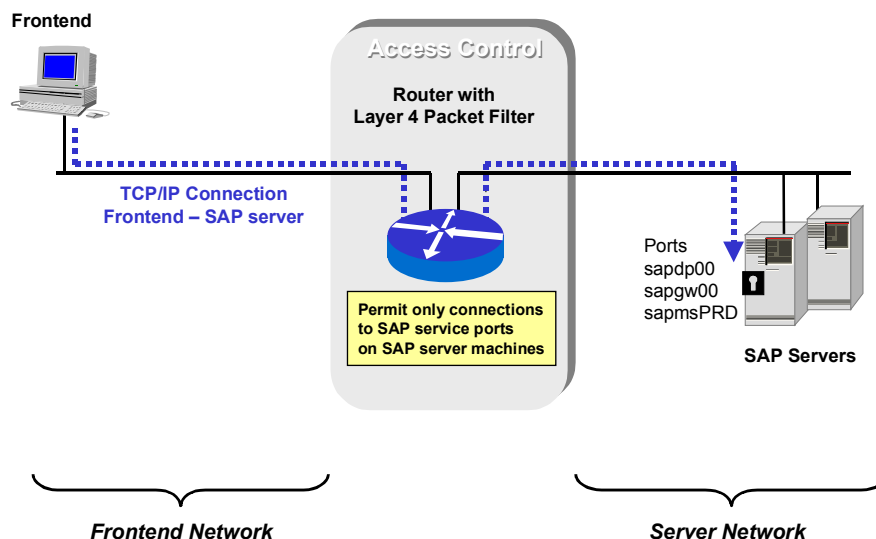
Packet Filters

The most simple method of controlling access is to connect subnets through a router that can filter network packets according to certain criteria. Some of these devices work in the network layer (IP protocol, layer 3 in the TCP/IP protocol stack), others filter in the transport layer (TCP protocol, layer 4).

Routers that filter in the network layer can allow or reject packets between two computers, or between two IP networks. This is the most simple variant, however it does not provide much protection. It does not filter out external intruders who use IP spoofing to change their IP address temporarily to an allowed address.

Devices that work in the transport layer can also look for TCP information in the network packets. You can filter them for certain services (TCP ports) and also according to which partners are allowed to set up or accept a connection.

The TCP ports required by SAP services are described in [Communication Connections of the R/3 System \[Page 13\]](#). The following graphic shows a simple form of access control that uses a router to support filtering in the transport layer. The frontend computer sets up a direct TCP connection to the application server through this router. The router only allows frontend computers to connect to the SAP services of this SAP System. The example shows the TCP ports for an SAP instance called PRD with instance number 00.



SAProuter

SAProuter is a program that relays SAP data (SAPgui, Remote Function Call, CPI-C, SAPlpd,...) between IP subnets or within a single subnet. SAProuter cannot replace a conventional router, instead it is intended to relay SAP network connections directly and securely from your own network to another. These types of programs are also known as application level gateways.

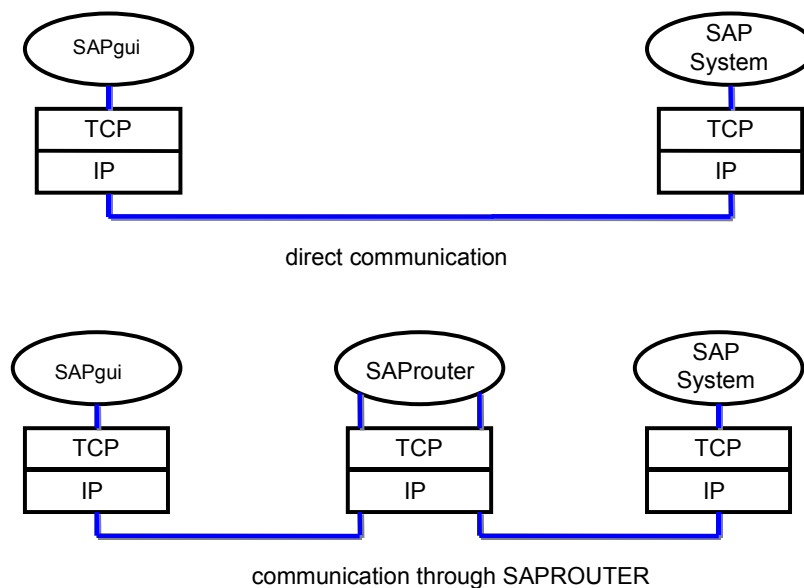
Controlling Access

How SAProuter Works

SAProuter examines any incoming data packets, checks the route and authorization and then sends them to their specified targets. The partner that sets up the connection specifies the route (from SAProuter to SAProuter and to the target). This procedure is known as source routing. If you want a connection, from a SAPgui to an R/3 System for example, to run through a SAProuter, then you need to enter the route in the SAPgui command line.

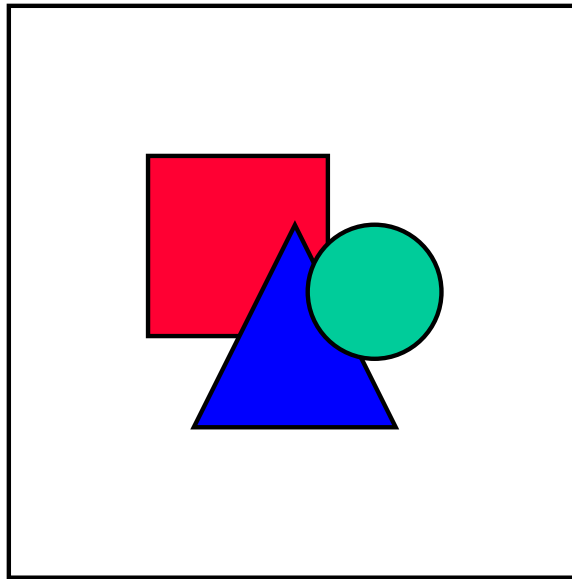
You can use SAProuter if there is no direct IP connection between the communication partners. SAProuter must be running on a host that is connected to both IP networks. It can then receive data from the SAPgui in one IP network and relay it to the R/3 server in another IP network, and vice versa. Normal IP routing is used between the application programs and the SAProuters.

The SAProuter has a separate TCP connection to each communication partner. It does not relay any IP packets directly. The following graphic shows a SAProuter connection between a SAPgui and an SAP System:



You can use SAProuter to completely disconnect IP networks or subnets. You may want to do this for the following reasons:

- You want to protect certain subnets for security reasons (example: firewall, see below).
- Two programs that need to communicate with each other cannot do so because of the network structure. Examples: There is no direct IP routing, there are no registered IP addresses...
- You do not want R/3 processes to use slow WAN connections. SAProuter has a separate TCP connection to each communication partner which means that the R/3 System only communicates directly with SAProuter at the network level. SAProuter then communicates with the WAN when it relays the data.



SAProuter is also used when you connect to SAPnet or to an SAP Service Center to allow remote access to your system.

Security Functions of SAProuter

You can use SAProuter to improve your network security by doing the following:

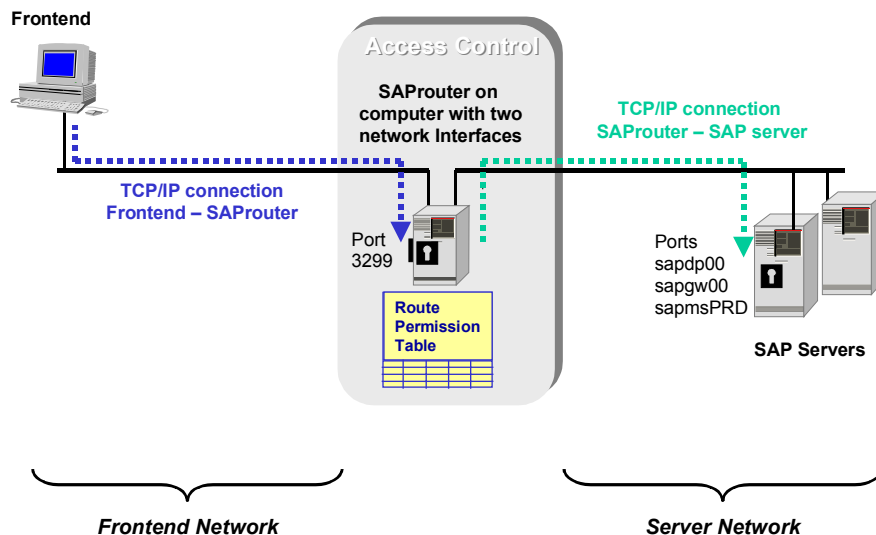
- Only allow connections between certain frontend computers or to certain server computers
- Only allow connections to certain SAP services
- Only allow access from certain other SAProuters
- Use a password to protect your SAProuter from unauthorized external access
- Log the connections to your SAP System in different levels of detail

The SAProuter configuration is recorded in a file called the *Route Permission Table*. Use this file to specify the IP addresses and address templates of computers that are allowed access to your SAP Systems, and to specify services that these computers can connect to. You can also log important SAProuter activities, such as setting up and ending a connection.

For more information on configuring SAProuter, see the SAP Online Documentation *BC - SAProuter* or Note 30289.

The following graphic shows the route that the network connection takes through SAProuter. The connection is made up of two separate TCP connections, with SAProuter forming the connection in the application layer. The arrows show the direction in which the connection is set up. The TCP ports in this example are for an SAP instance called PRD with the instance number 00. You need to allow access to these ports by making appropriate entries in the *Route Permission Table*.

Controlling Access



If you use the message server for load balancing, you need to open the port `sapms<sid>`, as well as the dispatcher port `sapdp<nn>` and the gateway port `sapgw<nn>`.

Firewall with SAProuter

If you want to take particularly heavy security measures to protect certain areas of the network, for example when you want to protect access to and from an open network such as the Internet, then you need to implement a multilevel security system. This type of system is usually known as a firewall. You can also use firewalls to secure critical areas in your internal network, such as an SAP System that contains highly sensitive data.

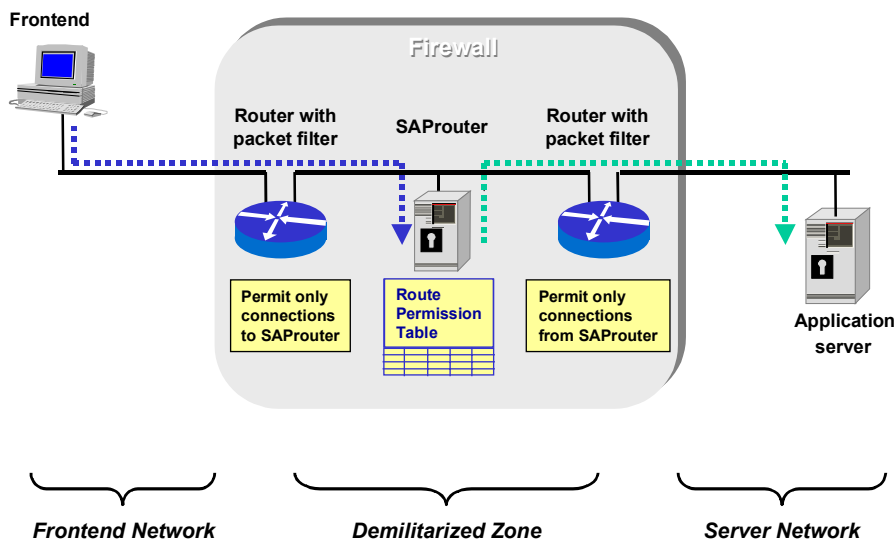
A firewall is a system made up of hardware and software components that only permits authorized connections and blocks all others. The components in a firewall system work in different network layers:

- **IP packet filter:** Authorized communication partners are identified by their IP address
- **Filter at the TCP port layer:** All TCP ports are blocked, except for those that are assigned to the required applications (usually all standard services are blocked). This means that an external program can only communicate with an authorized application. In this case the applications need to take additional security measures to identify the external communication partners (see below).
- **Application level gateways** provide an enhancement to TCP port filtering. A program (application level gateway) is installed in the firewall for each application with an authorized TCP port. This program receives the incoming data, uses this application to identify the source of the data, if possible, and then relays the data into the internal network if it passes this check. This means that there is no longer a direct TCP connection between the computers in the internal network and the outside world. SAProuter provides this function for SAP data. It uses the *Route Permission Table* to check authorizations.

Many vendors offer complete firewall packages that include all the required hardware and software components in pre-configured form. Some firewalls support user authentication with one-time passwords that are generated by token cards. This type of procedure is far superior to the password mechanism implemented in SAProuter. For more information, see the documentation provided by the vendor. Since traditional firewalls do not contain special function for SAP network traffic, you need to handle SAP connections in the same way as normal TCP connections. We recommend that you do not accept multiple TCP connections (dispatcher, gateway) for each application server. Your configuration will be simpler if you use a SAProuter that bundles these connections. Contact the vendor of your firewall to find out how you can integrate SAProuter into a complete firewall product.

There is no reason why you cannot construct a firewall system yourself, using normal components such as routers and computers. We cannot describe the implementation of a specific vendor here. Instead, the following describes a system that is constructed from individual components. This also makes it easier to see the connections between the components.

In the lower layers, routers and packet filters block IP packets according to simple, predefined rules. SAProuter plays the role of an application gateway for SAP connections. The following graphic shows an example of how to secure an SAP network with a firewall of this type:



The server and frontend networks have direct access only to the computers in the 'demilitarized zone'. The routers/packet filters are configured so that they only accept connections for certain network services on the computers in the demilitarized zone. They never accept connections from the other router. This ensures that each network connection must pass through the SAProuter (or another application gateway) in the firewall. Any other application gateways, such as Web proxies, are also located in the demilitarized zone.

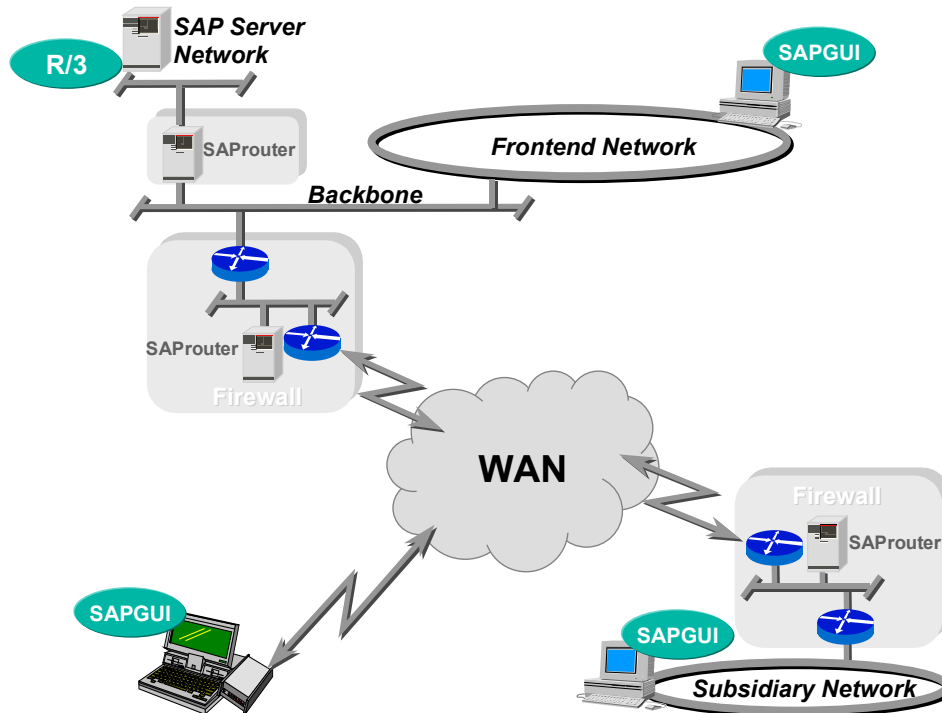
If you use SAProuter, you only need to connect the packet filter of the router in the frontend network to the port on the host where SAProuter is running (this is usually the port 3299). All SAPgui and RFC connections must pass through this port. The second router makes sure that,

Controlling Access

even if an intruder takes control of SAProuter, other network services on the SAP server computer cannot be accessed directly.

Example of Usage

The following graphic shows you an example of how to use firewalls with SAProuter:



The example includes SAP frontends in the internal corporate network, in a subsidiary network connected by WAN, and in laptops using dial-in connections. The corporate network is protected by a firewall containing a SAProuter that relays all SAP connections. The firewall can also contain other application level gateways, for example, for web access, but they are not shown here. The internal subsidiary network is also protected against access from the WAN by a firewall.

All SAP frontends access the central SAP System. The SAP servers are located in a separate high-speed network (only one server is shown here). A SAProuter connecting this server network to the backbone in the internal network provides additional protection for the systems.

Encrypting SAP Network Connections

The protection of network connections includes three levels of security:

- **Authentication**
Authentication only identifies the communication partner. The data itself is not protected.
- **Integrity**
This protection mechanism allows the system to recognize changes made to data while it was transmitted.
- **Confidentiality**
Data is encrypted to ensure its confidentiality, making it pointless to try and read it without authorization.

Cryptographic methods are used in all three levels of security. Authentication is the lowest level of security, confidentiality the highest. If you use one of the higher levels, you probably use those below it as well.



Note that the use of cryptographic methods can be subject to national restrictions.

You can use cryptographic methods at different layers in the network protocol stack:

1. **Physical Layer**
The physical layer is a passive component that does not offer encryption of data. Electronic data security is not discussed in this piece of documentation.
2. **Data Link Layer**
The network packets in the data link layer (the Ethernet, for example) can be encrypted directly by the network card. This method is complicated by the need for special hardware and is rarely used in computers.
3. **Network Layer**
There are several standards for securing data at the network layer (the IP layer, for example); the best known is IPsec, which will in future be part of the Internet protocol standard IPv6.
4. **Transport Layer**
There are also a number of standards in the transport layer (the TCP layer, for example. The Secure Sockets Layer (SSL) is used particularly often in the Internet.
5. **Application Layer**
Most applications at this layer define their own security protocols. SAP does not have its own security procedure, instead it uses Secure Network Communication (SNC) as an interface to the security products of other vendors. See below for more information.

Encrypting SAP Network Connections

Securing Data Below the Application Layer

You can secure the transmission of data in any layer below the application layer. The mechanism you use must be completely transparent to the SAP System, since SAP does not support these methods directly.

There are a range of products that add security functions to the TCP/IP protocol in the operating system, while remaining transparent to the application. For example, you can use an external program to authenticate each TCP connection in a firewall system without this being seen by the communication partner. Some vendors use security functions to modify the TCP/IP protocol stack directly. You can use these products together with the SAP System, however you must carefully check their compliance with the system, the SAPgui and any other SAP components you use. SAP does not test these security functions itself.

If you want to connect separate local networks through a non-secure open network such as the Internet, you could use Virtual Private Networks (VPNs). VPNs use suitable network devices (such as routers) to encrypt all network connections between two local networks. This process is completely transparent to the applications, which means that you can run your SAP connections through a VPN.

Procedures that go below the application layer can only authenticate users between network components. The user must still log on to the SAP System with a user name and password. An exception to this is web access through the Internet Transaction Server that has been encrypted with Secure Sockets Layer (SSL). Here you can use the X.509 certificate used for the encryption for authentication in the SAP System as well. For more information, see [ITS Security \[Page 100\]](#).

Securing Data in the Application Layer with SNC

SNC (Secure Network Communication) is an interface in the SAP architecture that lets you use external encryption products to secure SAP communication. SAP does not implement any encryption methods in its own software, instead it lets the user choose an encryption procedure and infrastructure, such as key distribution. SAP software is not subject to country-specific restrictions on encryption software and is always kept up-to-date. The security product can also use other security functions not offered directly by SAP, such as smart cards or biometrics. A variety of products has already been certified for use with SAP. The product you use determines whether SNC supports all three levels of security.

This piece of documentation only describes those areas where SNC differs from other methods of securing the transmission of data. For a detailed description of SNC, see the SAP Online Documentation.

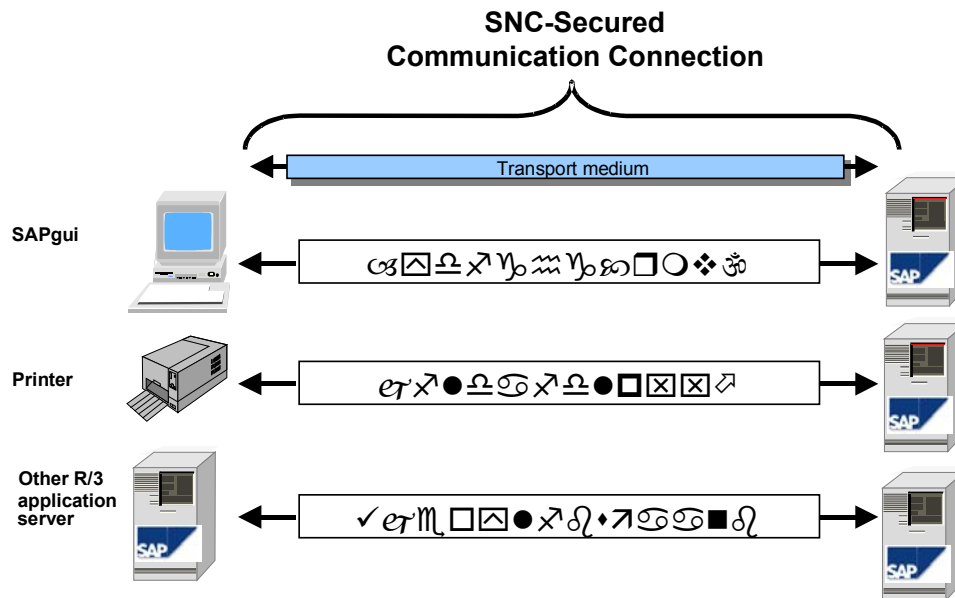
SNC secures data in the application layer. This guarantees a secure connection between SAP communication nodes (for example, between the SAPgui and the SAP application server), independently of the communication connection or transport medium used. This data security applies to SAP data traffic only.

You can use SNC for all types of external SAP communication:

- SAPgui
- Printing (together with the SAPlpd printer server)

Encrypting SAP Network Connections

- Communication between SAP Systems
- Communication with external systems using RFC and CPI-C
- Internet Transaction Server

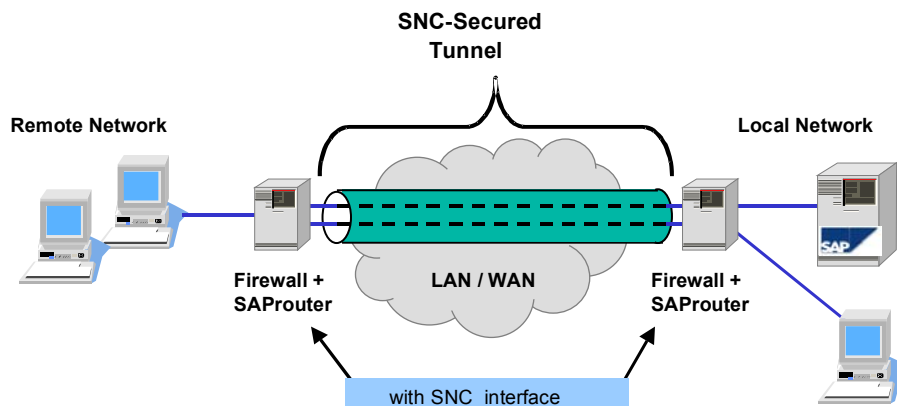


A user who logs on to the SAPgui through an SNC connection is automatically authenticated in the SAP System. The user does not need any other logon information such as user name or password.

You cannot use SNC to secure the connection between the SAP application servers and the database. For this reason, SAP recommends that you operate the application and database servers in a secure network that you can protect with appropriate network tools (see [Controlling Access \[Page 126\]](#)).

You can also use SNC between two SAProuters to set up a secure tunnel between networks, as in a Virtual Private Network (see the following graphic). This infrastructure secures your connections even if some of your components have an older SAP Release. SNC is supported for all external connections of the SAP System as of Release 4.0A (full Internet Transaction Server functions are supported as of Release 4.5A). You require the most up-to-date SAProuter version.

Encrypting SAP Network Connections



Comparison of Data Security Methods

The following table summarizes the main differences between securing data in the network layer and in the application layer (SNC):

Network Layer	Application Layer (SNC)
External product	External security product required
Transparent for SAP	Certification by SAP
Authentication in the network layer	Authentication in the SAP System
Secures all network data traffic	Secures SAP network data traffic only

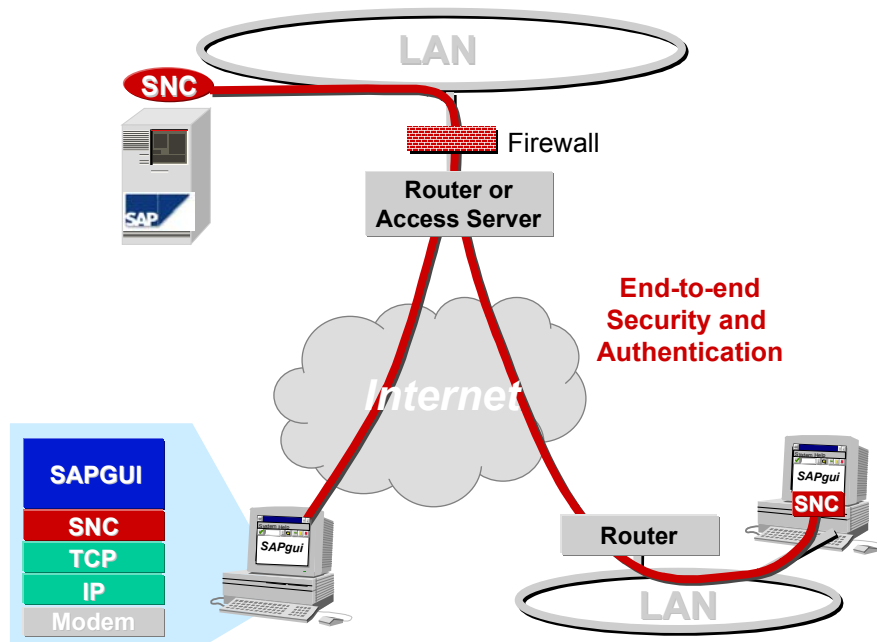
Example of Usage: SAPgui over the Internet

The Internet is being used increasingly for conducting business transactions. These business applications are based mostly on Web technology and therefore use a Web browser as a frontend. With the Internet Transaction Server (ITS), SAP gives you the option of operating an SAP System through a Web browser. For more information on security issues in this area, see [Internet Transaction Server: Technology \[Page 80\]](#). You can also log on to the SAP System directly through the Internet and the SAPgui. This does not present any technical problems, since the Internet works with the same network protocol as the SAPgui, namely TCP/IP. However, because the Internet is largely an open network, neither availability nor security is guaranteed.

As well as security for the transmission of data, you also need a firewall that rejects unauthorized access and so guarantees the security of your internal network and the SAP System. For more information, see the section [Controlling Access \[Page 126\]](#).

Securing Data in the Application Layer

The following graphic shows you how you can use SNC to secure a SAPgui connection over the Internet. Communication is secured between the end nodes SAPgui and SAP System.

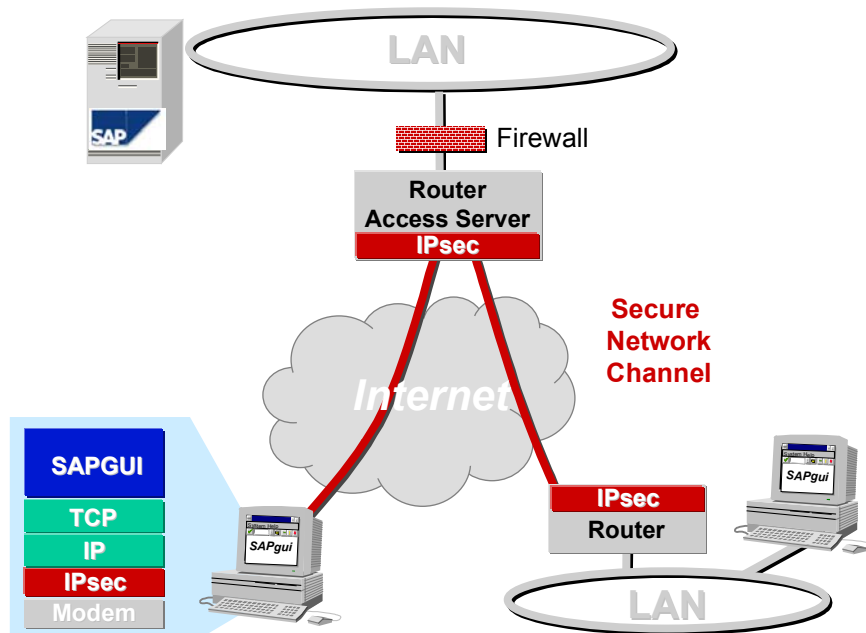


The security software takes care of authentication in the SAP System. You also need an authentication at the firewall to restrict access to the corporate network. Only SAP data is encrypted; access to files or the exchange of e-mail with the corporate network is not secure.

Securing Data in the Network Layer

The following graphic shows you how you can secure the transmission of data in the network layer. The data is secured between the access points to the open network, but not in the internal network.

Encrypting SAP Network Connections



A Virtual Private Network (VPN) is used to connect to a network in a remote subsidiary. The network components set up a secure connection to the end nodes of the Internet Service Provider that gives you access to the Internet. The network packets between the local networks are then tunneled through this connection. The connection uses a secure protocol such as IPsec. You can make use of these security functions either from devices in your network, or from an Internet Service Provider.

Dial-up connections from an individual computer can use a *Virtual Private Dial-Up Network*. In this case, either the network software on the frontend host or the Internet Service Provider must take responsibility for securing the data. The graphic shows in which part of the network stack the security function takes effect.

The type of data security you choose depends on your requirements. If you want to secure end-to-end data transmission, and use cryptographic methods for authentication in the SAP System, then you need to use SNC. It is particularly worthwhile to set up a corresponding infrastructure with a security product certified by SAP if you want to use SNC throughout the whole company. However, if you only want to secure the transmission of the data, a VPN is probably the simplest solution. You can then use generally available network components with security functions that secure all data traffic between the subsidiaries.

Network Load

Frontend Network Load

Type of Communication

The communication protocol used between the SAPgui and the R/3 application servers is **block-oriented**. This means that the entries made by the user are not sent to the application server immediately. The user enters and edits the data locally on the terminal until it is released. Only then is a request with a data packet sent over the network to the application server. The server processes this request and then sends the response back to the SAPgui. This action is known as a *dialog step*.

As well as the dialog steps, other communication also takes place, for example:

- You issue a request when you open SAPgui menus, since only then are the menus fetched from the application server.
- The application server controls the progress display in the SAPgui status bar. You can deactivate it if the connection is poor.
- Keep-alive data packets are exchanged if the SAPgui is inactive for long periods of time. These packets only consist of a few bytes.
- Frontend printing through the SAPgui can place a heavy load on the network (see *Network Load Due to Printing*).

A communication basically consists of a request initiated by the frontend, to which the application server reacts with a response. Keep-alive data packets are an exception, since the application server sends them to the SAPgui without being requested to.

Amount of Data in each Dialog Step

The communication between the SAPgui and the R/3 System has been optimized to keep the load on the network as low as possible. Only the contents of those elements that actually appear on the screen are transmitted. The graphics are formatted on the frontend. In addition, the data is compressed.

It is difficult to make a general statement on how much load the SAP frontend places on a network. The load depends on the SAP transactions used, the speed of the user, and to a lesser extent, on the application data.

SAP uses SAP standard benchmark measurements for the applications FI, MM and SD to determine the amount of data in each dialog step. These benchmarks simulate typical actions in these applications. The network load does not differ greatly between the applications and stands between 0.5 KB and 3 KB for each dialog step. For R/3 Release 4.0 **an average of 1.7 KB is transmitted across the network for each dialog step**. Other applications may return different values, however, experience has shown that the standard benchmark value is a good approximation of actual requirements. For more exact planning, however, it is advisable to carry out detailed tests with the actual user profiles.

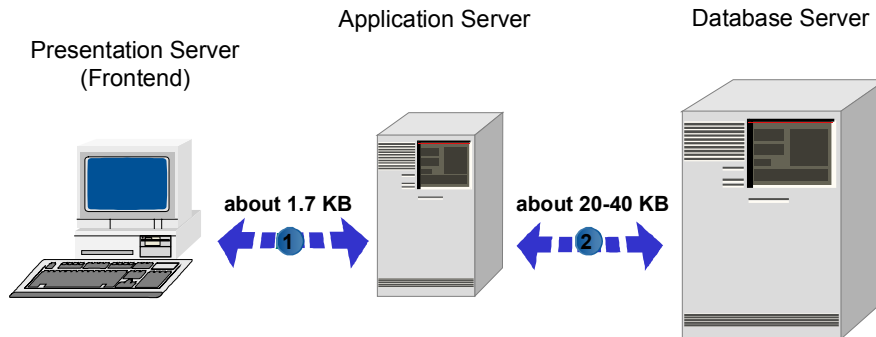
We cannot make a statement on the network load caused by downloading large tables (for example, for external table calculations), since the data involved depends entirely on the application data.

Transactions that use the new SAP control technology can differ greatly from traditional transactions in the amount of network load they cause. The amount of data that has to be transmitted depends more strongly than before on the data being processed by the user. This

Frontend Network Load

can, for example, be due to the fact that an editor control always has to transmit the entire text to the frontend for processing. This means that the amount of data that has to be transmitted depends on the size of the text. On the other hand, however, the user can now edit the text on the frontend and data is not continually being exchanged with the application server. This leads to fewer dialog steps in an application, and more data in each dialog step. When you attempt to estimate the network load produced by transactions that use controls, you must use actual user profiles for your analysis. Only in this way can you be sure that your planning data is reliable.

Regardless of the type of transaction, the raw data from the SAP database tables is always processed on the application server. A consequence of this is that there is about 10-20 times more data flowing between the application server and the database than between the application server and the frontend. The following graphic depicts typical data traffic between the frontend application server and the database:



Formula for the SAPgui Network Bandwidth

You can use the following formula to estimate the network bandwidth needed between the application server and the frontend for N users. It takes into account the average amount of data traffic in each dialog step, the number of users, the time a user takes to process a screen ("thinking time") and the response time of the system. This formula has been tried and tested in practice, however it is still only a first estimate of the actual bandwidth requirements.

The parameters of the formula are:

C	Network bandwidth required for SAPgui data [bit/sec]
N	Number of users working simultaneously
T _{Think}	Thinking time needed by a user to process a screen [seconds]
T _{Resp}	Response time needed by the system until it can display the next screen [seconds]

Frontend Network Load

The required network bandwidth is estimated as follows:

$$C = 16000 * N / (T_{\text{Think}} + T_{\text{Resp}}) \text{ bit/sec}$$



Note the following:

- Experience has shown that working at the theoretical maximum capacity of a connection can lead to a strongly peaked distribution of network delay times. This is because the work steps of the users are not distributed evenly over time. The fewer users working in parallel, the more frequent are these peaks.
- There is no safety reserve built into the formula. We recommend that make your bandwidth at least double the result of the formula.
- A bandwidth of 9600 bit/sec is the minimum acceptable bandwidth when working with one user.
- This formula only applies to SAP dialog transactions (without controls) up to and including R/3 Release 4.5.
- The most imprecise part of the formula is the thinking time assumed for the user. The most reliable way of estimating this is to observe the way users work productively over an extended period of time.
- The formula only gives the requirements for SAPgui data traffic. You must also take load caused by other applications (printing, web browser, access to file server and so on) into account when you estimate your required bandwidth.

Network Load with Windows Terminal Server

The Microsoft Windows Terminal Server (WTS, also known as "Hydra") is software that lets multiple users work interactively and simultaneously on the same Windows NT computer. The actual program runs on the server, however the user interface appears on another computer, the client. The client is connected to the server through a network. The client computer needs a program that relays the entries made by the user (with keyboard and mouse) to the server and displays the graphical output of the program in a window. The WTS uses special compression methods to transmit the graphical output as a bitmap.

You can use SAPgui with WTS since Release 4.5A. (Remember that you can have only one version of the SAPgui installed on the computer at a time.)

Using typical SAP transactions (SD benchmark), we have measured network traffic of about 20-40 KB for each dialog step in a local network, about 10 times more than a SAPgui. The network traffic has a different character than with the SAPgui. Instead of large network packets, the WTS sends many smaller packets (about 20 times more than the SAPgui). A window size of 800*600 was used in the test, with the SAPgui maximized within the window. However, the size of the window does not influence network traffic greatly, due to the use of data compression.

It is difficult to compare this load with the network load of the SAPgui. WTS network traffic depends greatly on how many keystrokes, mouse moves and waits are used for each dialog step. The SAPgui always produces the same amount of data in each dialog step, regardless of the speed of the network. This can vary for WTS clients, depending on whether the connections are fast or slow. We made our measurements in a fast LAN.

The Citrix MetaFrame Client (an additional product) produces about 30% less network traffic than the WTS client from Microsoft. Citrix states that their client can also be used comfortably with slow modem connections (28.8 Kbps).

Network Load in the SAP Server Network

Network Load in the SAP Server Network

The data traffic between the SAP servers is not as easy to describe as the traffic between SAP frontends, since it depends more heavily on the application. The following sections describe several basic "rules of thumb" for the different types of data traffic. These rules can help when you make an initial estimate of the required bandwidth in the server network.

To obtain more accurate data, you must make measurements from a real system. For a useful measurement, you must have realistic data in the system and all data flows must be considered.

Interactive Users

In the three-tiered architecture of the SAP Systems, data is mainly processed on the application servers and the database. The thin frontend only receives the data sent over the network that you want to display. The amount of data between the SAP application servers and the database is at least one order of magnitude greater than between the application server and the frontend. This is because the system must transmit not only the data that is displayed but also data that is *not* displayed.

The average data transmitted in different, typical applications (standard SAP benchmarks) between the application server and database is approximately 20-40 KB in each dialog step. This amount can vary greatly depending on which transaction is executed and how much data is in the system.

In addition to the required network bandwidth, the runtime of packets in the network is also very important for system performance since a lot of queries and responses are sent back and forth ("*round trips*") when the database is accessed. Therefore, choose a network technology with the shortest delay time, for example, layer 2 switches. "Slow" network components such as routers usually should not be placed within the server network.

The data traffic between the application servers of an SAP System is much lower than between the application servers and the database. However, short network runtimes are also important for this type of data traffic.

Background Jobs

In addition to the load generated by interactive applications, you must also consider batch jobs. They generate a similar load for each processing step as the interactive applications. The difference is that there is no delay between the processing steps as with interactive users. This is why the network load of a background job is much higher than the load an interactive user generates.

External Communication

External programs that use CPI-C or RFC to communicate with the SAP System also generate network load. It is difficult to make a general statement about network load. If the external program executes a similar action to an interactive SAP transaction, the data traffic between the application server and the database is generally comparable to this transaction since the data must be updated in the same way. Therefore, the network traffic between the application server and the database is also one order of magnitude greater for external programs than for data traffic between external programs and the application server.

Database Backup

Along with the SAP System data traffic, there are other applications necessary for operating the SAP System that generate load in the server network, for example, management functions and file access. The most important of these functions is backing up the database. The data traffic generated by this action has the same order of magnitude as accessing the database. In large installations, you may have to back up the database using a separate network adapter and a special backup network (for example, a **Storage Area Network**).

Other Applications

If other applications are running on the same network or even on the same servers, you must consider this when you determine the network bandwidth. In a large SAP System, we recommend that for each system you set up a separate network that is solely reserved for the data traffic between the application server and the database.

Network Load from Printing

Network Load from Printing

Printing from the SAP System

The printing format and the data transmission to the printer or the spool system normally occurs in the **spool** work process of the SAP System. The amount of data and the destination chosen by the spool work process for the print data are both factors in network load.

The amount of data to be printed depends on:

- The type of printing format (printer device type for example, HP-PCL or postscript)
- The protocol of the data transmission (for example, Berkeley lpd or SAPlpd)

The route over which the data is sent depends on the access method (local or remote printing).

Access Method for Local Printing

L = Unix local printing with lp/lpr

C = local printing with an NT operating system

E = different output management system

Access Method for Remote Printing

U = Printing to an LPD host using Berkeley protocol

S = Printing to SAPlpd host using SAP protocol

F = Printing on the frontend host

Network Load when Printing

How Size of Print Jobs Affects Network Load

Small print jobs of only one page result in an overhead of approximately 100% due to the print job administration in relation to the transmitted data. Print jobs of 10 pages lower the administration overhead to approximately 10%.



If possible, ensure that print jobs are as large as possible. Avoid printing only single pages.

How Access Methods Affect Network Load

The access method that uses the Berkeley protocol transmits the print data uncompressed over the network. The network load corresponds to the number of characters to be printed and the print job administration.

The SAPlpd access method differs from the Berkeley protocol in that the print data is compressed. For full body text pages, the compression rate reaches up to 3:1.

The frontend printing access method causes a greater overhead due to processing through the connection to the SAPgui frontend. Therefore, only use this method in exceptional cases.



If possible, use access method **S** (SAPlpd).

How Output Devices Affect Network Load

Postscript devices generate approximately 20% overhead for body text.

HP PCL devices generate approximately 10% overhead for body text.

SAPWIN devices generate approximately 8% overhead for body text.



Output devices only have a marginal effect on network load.

Determining the Customer-Specific Network Load

In LAN environments, printing usually causes no problems regarding network load. For business locations connected through a WAN, a large amount of print data can lead to problems. To best determine the network load caused by printing, you must determine the size of the print-formatted documents. You must consider the documents that are printed most often at the relevant locations.

Proceed as follows:

1. Call Transaction SPAD.
2. Switch to change mode and choose an output device.
3. Choose *Edit* → *Debugger*.
4. Select the option *Retain print file*.

After printing, the print data is stored in the `/usr/sap/<SID>/<instance>/data` directory. You can view the size of the data in the file system.

Measurement Results

Components when Measuring

R/3 Release 4.0B

SAPlpd Version 4.08

Conditions for Measurement Results

Only body text pages without formatting were used for the measurements. Three measurements were made for each configuration.

1. Measurement: 1 page (65x80) = 5200 characters
2. Measurement: 10 pages (65x80) = 52000 characters
3. Measurement: 100 pages (65x80) = 520000 characters

The transmitted data bytes were measured here.

Measurement Results**Results**

Berkeley lpd resulted in the following ratios of transmitted data to user data

Print job, 1 page body text with Postscript	234%
Print job, 10 pages body text with Postscript	130%
Print job, 100 pages body text with Postscript	122%
Print job, 1 page body text with HP PCL	160%
Print job, 10 pages body text with HP PCL	112%
Print job, 100 pages body text with HP PCL	110%
Print job, 1 page body text with SWIN	145%
Print job, 10 pages body text with SWIN	110%
Print job, 100 pages body text with SWIN	108%

SAPIpd resulted in the following ratios of transmitted data to user data

Print job, 1 page body text with Postscript	146%
Print job, 10 pages body text with Postscript	53%
Print job, 100 pages body text with Postscript	38%
Print job, 1 page body text with HP PCL	133%
Print job, 10 pages body text with HP PCL	50%
Print job, 100 pages body text with HP PCL	36%
Print job, 1 page body text with SWIN	128%
Print job, 10 pages body text with SWIN	47%
Print job, 100 pages body text with SWIN	35%

Frontend printing resulted in the following ratios of transmitted data to user data

Print job, 1 page body text with SWIN	376%
Print job, 10 pages body text with SWIN	235%
Print job, 100 pages body text with SWIN	258%

ITS Network Load

ITS Network Load

This section discusses the amount of data passed between the components of the ITS when it processes Web applications.

Statistics are given for typical WebTransactions delivered by SAP, for example the WebTransaction *Online Store*. We selected a few items from a short product list and placed an order. No pictures were associated with the products. Other ITS technologies (WebReporting and WebRFC) are not covered here because the amount of data depends entirely on the application data or the way in which a self-written WebRFC application is programmed.

The network load can be computed directly from the amount of data in each dialog step, multiplied by the number of users, multiplied by the number of dialog steps in each time interval.

Browser – Web Server

The network traffic is usually dominated by images embedded into the HTML pages. There are two kinds of images: General images that are used for buttons and similar features, and application images, such as pictures of products. Our measurements can only include the general images, but no application images, because we cannot make an assumption about their size.

It is generally advisable to have network load considerations in mind when you design a Web application. Otherwise you may irritate users who have slower network connections.

In our test application, each dialog step (mouse click) resulted in an average of 4 requests to the Web server. This includes images used for buttons. Some dialog steps update more than one browser frame and so request more than one HTML page to be loaded. Each dialog step created average network traffic of about 23 KB.

The amount of data in each dialog step depends on the SAP transaction that is called by the ITS and also strongly on application data. This dependency on the application is at first surprising to anybody who is familiar with SAPgui network traffic, because the SAPgui exhibits almost no such dependency. This is because the SAPgui only receives the screen elements that are actually displayed on the screen. Any scrolling results in a new communication step. In contrast, the ITS has to fetch the entire list from the SAP System and send it to the browser for local scrolling. This results in more data for each screen, but fewer screens that have to be transmitted.

WGate – AGate

Because the WGate itself processes almost no data, the network traffic between the WGate and the AGate consists mainly of HTTP requests that are produced by the browser and the HTML pages generated by the AGate. The protocol overhead is minimal. Pictures are not sent over this connection but loaded directly from the Web server. The same applies to static pages, embedded Java applets and so on.

The size of the HTTP requests is usually small (a few hundred bytes). This means that the total load is dominated by the size of the HTML pages (usually a few KB). It also depends on the SAP

AGate – SAP System

transaction and on application data. The number of requests to the AGate is exactly the same as the number of requests to the WGate URL on the Web server.

In our test application, each dialog step resulted in an average of 3 requests from the WGate to the AGate and created an average network traffic of about 16 KB.

Using the built in DES encryption algorithm results in an overhead of about 10 percent.

AGate – SAP System

For WebTransactions, the network traffic between AGate and SAP system is just normal SAPgui traffic. However the amount of traffic in each dialog step is larger than for normal SAPgui users, since current versions of ITS do not support compression of the SAPgui traffic. So network traffic between the AGate and SAP is about two times higher than that between the SAPgui and SAP for the same transaction.

The amount of data transmitted between the AGate and the SAP System again depends on the SAP transaction, as well as on application data. Note that multiple frames (each of which result in one Web request) do not result in multiple requests to the SAP System because they are all contained on one SAP screen. So the number of requests to the SAP system may actually be lower than the number of Web requests to the WGate.

WebRFC and WebReporting use RFC to send the HTML text of the generated Web page from the SAP system to the AGate. The amount of data depends on the size of this page. Since RFC uses data compression, the actual network load is smaller than the HTML page size. For details see [Network Load from RFC Data Transmission \[Page 153\]](#).

In our test application, each dialog step resulted in an average of 2 requests to the SAP System and created average network traffic of about 5 KB.

Network Load from CPI-C Data Transmission

Network Load from CPI-C Data Transmission



The following information is valid as of R/3 Release 4.0.

General Information on CPI-C Data Transmission

The CPI-C protocol transmits data in character format in packets of up to 30 KB. This protocol does not compress the data. The application is responsible for the conversion of numeric or text content.

Statistics for the Required Network Bandwidth for CPI-C Data Transmission:

Since there is no data compression with the CPI-C protocol, the network load during data transmission is proportional to the amount of transmitted user data.

The relationship of network load to transmitted user data was determined by measuring transmissions with varying amounts of data and packet sizes.

This led to the following results:

- CPI-C connection: 2KB
- Network data/user data: 1,1:1

This data led to the following formula for calculating the network bandwidth:

$$B = \frac{16384 * V + 8192 * D * 1,1}{T} \quad in \quad \frac{Bit}{s}$$

Legend:

B	Required network bandwidth in Bit/sec.
T	Time period in seconds
V	Number of connections in time period T
D	Amount of user data (in KB) to be transmitted in time period T

Network Load from RFC Data Transmission



The following information is valid as of R/3 Release 4.0.

Overview

Network load is influenced by several factors when you use the SAP RFC protocol to transmit user data.

There are different types of RFC data transmission:

- **Synchronous RFC (sRFC):**
In a synchronous call, the RFC client waits for the result of the call before the program is continued.
- **Asynchronous RFC (aRFC):**
This is an RFC variant within the SAP System where the function module is started in the background. The client does not wait for the result of the call. This variant creates the same network load as synchronous RFC.
- **Transactional RFC (tRFC):**
Transactional RFC ensures that the call is processed only once in the target system, even if a connection is terminated, and the client repeats the call. This variant creates only very little additional protocol overhead compared to synchronous or asynchronous RFC.

The data that is transmitted is compressed before it is sent. Since the compression factor of the data depends on the content, a general statement about the required bandwidth of an RFC transmission cannot be made. There are different data categories, for example:

- **Text data** (example of a file with characters generated randomly)
- **IDOCs with IDES material data**
- **Binary data** that was already compressed

This data is used to estimate the required network bandwidth. You still need to make an exact estimate of the data to be transmitted.

Statements about transmitting data are limited to using tables, and not simple or structured parameters.

Furthermore, only connections between external programs and an R/3 System are considered. No statement is made about RFC connections between two R/3 Systems.

General Guidelines for Optimizing Data Transmission Using RFC:

To enable RFC communication, you must take the following points into consideration regarding the required network bandwidth:

1. **Establishing the connection:**
Each time a connection is established, logon data and other system parameters, such as character code pages, are exchanged. This causes a system load of approximately 2.5 - 3 KB on the network each time a user logs on to the SAP System. If several function modules are called one after the other, an existing connection should be used again if possible, since

Network Load from RFC Data Transmission

a new logon to the SAP System is not needed.

An RFC connection from the SAP System to an external RFC server program behaves much the same way. When IDOCs are transmitted, for example, sending a group of previously assembled IDOCs causes much lower network load than directly sending them one at a time.

2. **Size of the data units:**

Since the largest amount of overhead for an RFC transmission occurs when making the connection (or calling a function), sending the largest possible data blocks is the best method. Depending on the application, you need to make a compromise between the size of the data block and the manageability of the data. Tests have shown that for data of approximately 100 KB and greater, the overhead when making the connection or calling the function is negligible. Large amounts of data only produce additional improvements in performance if you have a very good data compression program.

3. **Registering external RFC servers:**

For RFC servers that are pre-registered on the gateway, there is a minor increase in protocol overhead (compared to the programs started on the gateway). If you do not take the actual registration procedure (1238 bytes) into consideration, approximately 3 KB more of data is transmitted. This is also true for registered RFC servers that the overhead is marginally low when the amount of data is 100 KB.

4. **Synchronous RFC or Transactional RFC:**

If you use transactional RFC, there is a marginally greater overhead when calling function modules (approximately 600 bytes for each call). This difference is small however, when sending large amounts of data.

Calculating the Required Network Bandwidth for RFC Data Transmission:

The following influence the required network bandwidth:

- Compression behavior of the data
- Size of the data units
- Time frame for sending the data

The required network bandwidth is determined by the amount of data sent over the network for each unit of time.

The data sent over the network consists of the application data that is transmitted, which is compressed, and the RFC protocol overhead.

(RFC protocol overhead means in this case the amount of data transmitted that results from making the connection, calling the function module and data coding, compression, flow control and backup.)

Since the compression factor cannot be generally specified, we cannot conclude that there is a proportional connection of application data and data on the network.

One estimate of the amount of data under certain conditions appears as follows:

Scenarios

1)

- The user data is already compressed in another way and cannot be further compressed.
- There is a mass data transfer (more than 100 KB per connection).

Network Load from RFC Data Transmission

→ The data load on the network approximately corresponds to the user data + 10%.

2)

- The user data is text data sent to tables. Table rows are filled with empty characters up to the end of the row (corresponds to the representation of IDOCs in the SAP System).
- There is a mass data transfer (more than 100 KB per connection).
- In the analysis, only 'non-empty characters' were counted as user data.
- The estimate is based on measurements with IDES data.
- The result depends heavily on the content of the text data and should be seen as an approximation.

→ The data load on the network is approximately 30 - 40% of the user data.

3)

- The user data is text data that does not conform to a specific model.
- The estimate is based on measurements with generated text files with random characters.
- There is a mass data transfer (more than 100 KB per connection).

→ The data load on the network is approximately 90 – 100 % of the user data.

Measuring RFC Data Transmission:

You can measure a data transmission using RFC with little technical effort. You need a host with a network monitor in the segment of the network containing a communication sending point. You can also install the network monitor on one of the target hosts.

To avoid having to build up an isolated measuring environment, you need to specify a data filter on the network monitor so that only the data belonging to the connection you want to measure is analyzed.

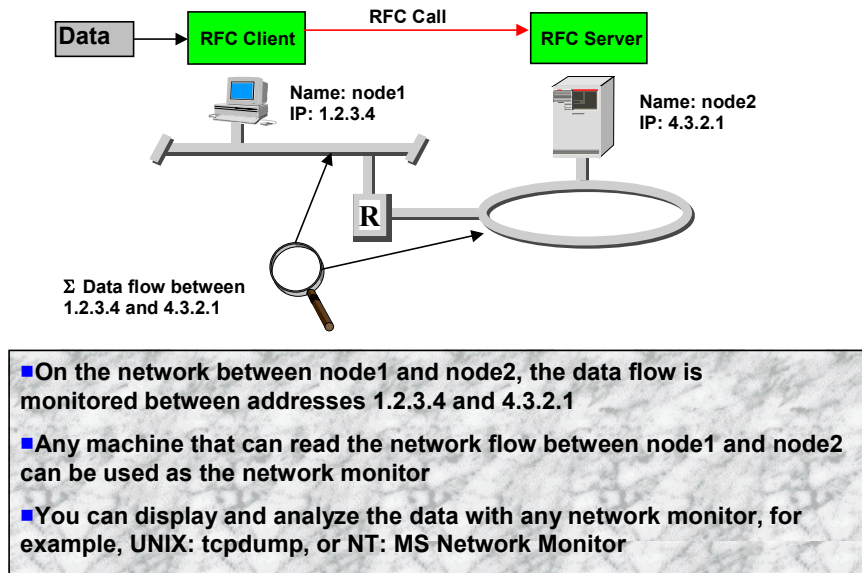
The easiest way to filter is to display only the data traffic between two specific hosts. If there is no connection between these two hosts other than an RFC connection that you are testing, then this filter is sufficient. The simplicity of this method makes it the most advisable.

However, if there are other connection types between the specific computers at the time of measurement, you must have additional filtering to limit it to the RFC connection (for example, at the port level or the protocol level).

Procedure

User data with a specific size and various contents was transmitted between an external system and the SAP System. The transmitted data was measured on the network using a network monitor that monitors the traffic between the two hosts.

Network Load from RFC Data Transmission



Result

If you consider the statements made above, you can determine that the network load is a maximum of 1.1 times that of the transmitted data (for mass data transmission).

If you use this as a worst case scenario, you can determine the maximum required bandwidth.

To make more accurate statements about the actual network load that is generated, you need to take measurements using the data you want to transmit. Here you can determine how densely the data can be compressed before transmission. The most difficult part of measuring is selecting representative data in order to determine the network load most accurately.

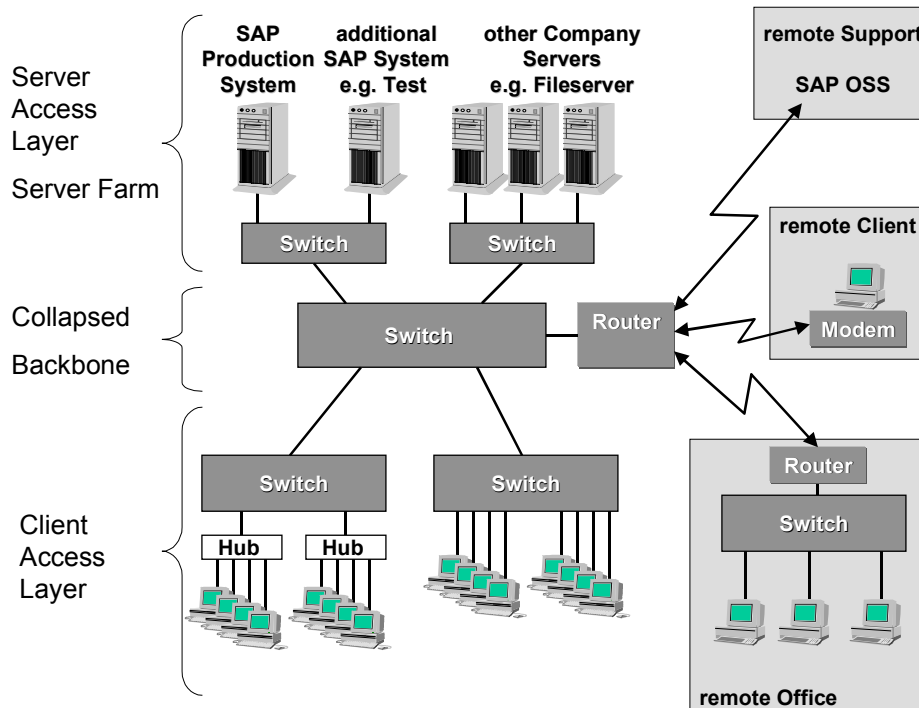
Appendix: Examples of R/3 Network Configurations

Overview

The following network configuration examples show how you can set up your network for small, medium, and large R/3 Systems. All these configurations serve as examples and are not compulsory. However, if you set up different networks, you must still follow the SAP guidelines for network configuration.

Small SAP Systems

For smaller R/3 Systems, the R/3 server uses only one network adapter for communicating with the SAPgui frontends. Since the network load is very low through SAPgui frontends, you do not need to connect the R/3 server using several, physically separate subnets (network adapters). Dividing up the rest of your network into subnets is possible and also useful. High data traffic only occurs locally on the R/3 server between the database and the central instance.



Midsize and Large R/3 Systems

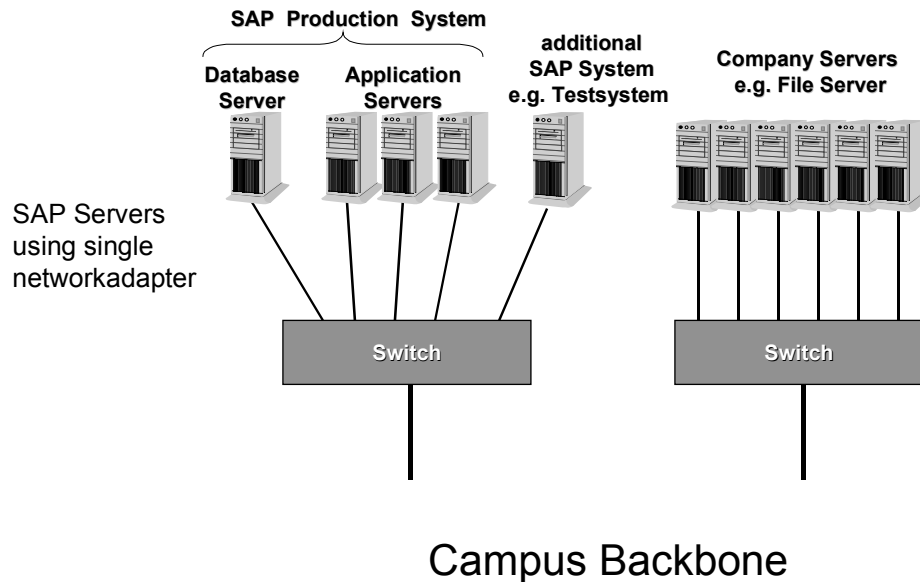
If your R/3 System consists of several servers, a high network load occurs between the database server and the application servers. A high-speed switch (at least 100 MBit) is suitable for a simple and cost-effective connection of the servers.

You can use one or two network adapters for each server. The differences are explained below.

Two Network Adapters for each Server (Separating Data Traffic)

One Network Adapter for each Server

The entire data traffic of a server runs over one network adapter only. Therefore, you need a network adapter with sufficient transfer speed (at least 100 MBit). If the servers connect using the switch, ensure that the entire bandwidth is available to all ports. The host name of the servers must match the IP name for the respective network adapter.



Two Network Adapters for each Server (Separating Data Traffic)

In very large R/3 Systems, the network adapter can become a bottleneck. In this case, you can install a second network adapter for each SAP server. It is only used for the communication between the R/3 servers (database server, central instance and dialog instance) (server network).

This type of configuration also lets you separate the data traffic between the application servers and the database server from the rest of the data traffic. If you need to, you can install several network adapters on the database server.

Two possible configurations are described in the following section.

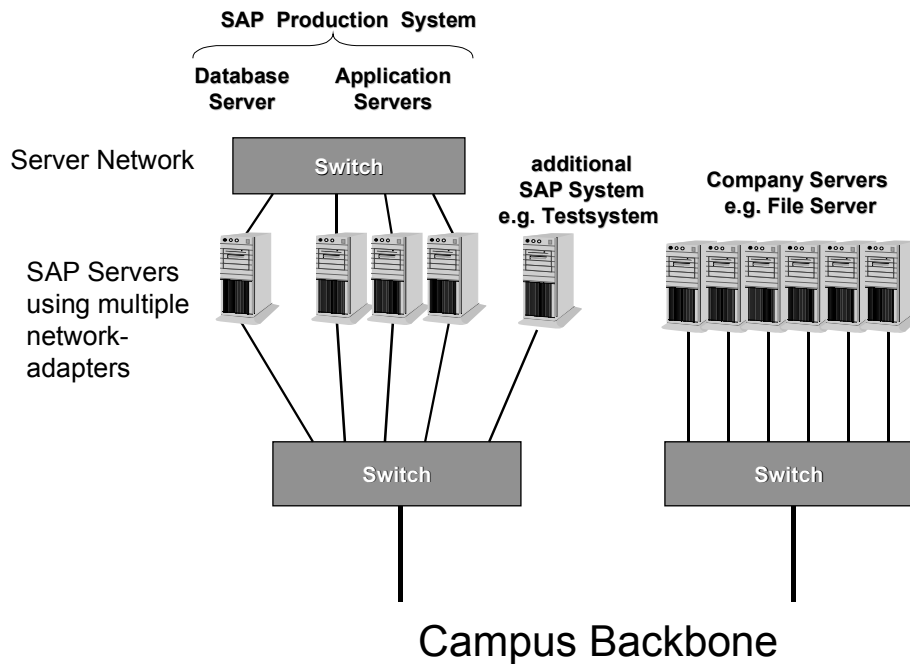
Connecting All SAP Servers to the Corporate Backbone

If an SAP instance also runs on the database server, you must connect this host to the backbone and the server network. When you assign the host name to an adapter, the same guidelines apply as with the application servers: The host name is assigned to the backbone connection.

In this configuration, routing entries must already exist on the application servers, ensuring that the database traffic flow is actually running over the server network.

Two Network Adapters for each Server (Separating Data Traffic)

When you are deciding if you want to install an additional SAP instance on the database server, consider performance and tuning aspects.



Connecting the Application Server to the Corporate Backbone

In this configuration, all the application servers are connected to the corporate backbone and to the server network. This complete isolation of the database server has advantages and disadvantages. Being separated from the rest of the corporate network guarantees high security against unwanted access to sensitive data, and against accesses that can hurt availability and performance on this important server. In this case, an SAP instance cannot run on the database server.

If a bottleneck occurs in the network adapter that connects the database server to the server network, you can establish an another connection to the server network with an additional network adapter.

For performance reasons, do not back up the data on the server, in particular the database server, over the server network. If you need to back up the data over the network, we recommend using a separate network for the backup (SAN = Storage Area Network). In this case, the database server must be equipped with an additional network adapter to connect to the SAN.

Two Network Adapters for each Server (Separating Data Traffic)