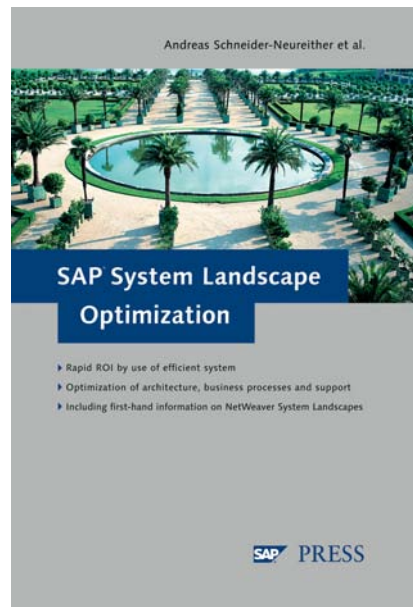


Andreas Schneider-Neureither (Ed.)

# SAP® System Landscape Optimization



SAP PRESS

# Contents

<b>Foreword</b>	<b>9</b>
<b>Introduction</b>	<b>11</b>
<b>1 Overview of SAP Solutions and Their Optimization Potential</b>	<b>17</b>
1.1 The SAP Range of Products .....	17
1.2 What Is SAP NetWeaver? .....	20
1.2.1 People Integration .....	21
1.2.2 Information Integration .....	21
1.2.3 Process Integration .....	21
1.2.4 Application Platform .....	22
1.3 The Components of SAP NetWeaver .....	22
1.3.1 The SAP Web Application Server .....	23
1.3.2 SAP Enterprise Portal .....	24
1.3.3 SAP Exchange Infrastructure .....	29
1.3.4 SAP Business Information Warehouse .....	30
1.3.5 SAP Mobile Infrastructure .....	31
1.3.6 SAP Master Data Management .....	32
1.4 The mySAP Business Suite Solutions .....	34
1.4.1 mySAP Enterprise Resource Planning (ERP) .....	34
1.4.2 mySAP Customer Relationship Management (CRM) .....	36
1.4.3 mySAP Supply Chain Management (SCM) .....	38
1.4.4 mySAP Supplier Relationship Management (SRM) .....	40
1.4.5 mySAP Marketplace .....	41
1.4.6 mySAP Product Lifecycle Management (PLM) .....	42
1.5 SAP Solutions for Midsize Enterprises .....	42
1.5.1 SAP Business One .....	43
1.5.2 mySAP All-in-One .....	43
1.6 SAP xApps .....	44
1.7 Industry Solutions .....	45
1.7.1 Industry Solutions for Banking .....	45
1.7.2 Sector Solutions for Insurance .....	48
1.8 SAP Solutions for Optimizing the SAP Landscape .....	49
1.8.1 Archiving .....	49
1.8.2 Modern User Administration .....	50
1.8.3 SAP Solution Manager .....	50

1.8.4	SAP Support Desk .....	51
1.8.5	Co-Hosting .....	51
1.8.6	Multiple Components on One Database .....	52

## **2 SAP Architecture Concepts 55**

2.1	SAP System Architecture .....	55
2.1.1	Presentation Layer .....	61
2.1.2	Application Layer .....	64
2.2	Security .....	70
2.3	Change and Transport Management .....	73
2.4	SAP NetWeaver .....	78
2.5	SAP System Landscapes .....	83
2.5.1	Example Configuration: Internet Sales and Procurement .....	84
2.5.2	Example Configuration: mySAP CRM Interaction Center .....	86
2.5.3	Example Configurations for System Landscapes and Client Strategies for Change Management .....	87
2.6	Lifecycle Management .....	91
2.6.1	Design .....	92
2.6.2	Installation .....	92
2.6.3	Implementation .....	93
2.6.4	Operation .....	94
2.6.5	Change Management .....	96
2.7	Solution Management .....	97
2.8	Summary .....	103

## **3 Optimizing SAP System Architecture 105**

3.1	Optimization Approaches .....	106
3.1.1	Process Optimization .....	107
3.1.2	Optimizing System Availability .....	108
3.1.3	Optimizing Reliability .....	109
3.1.4	Performance Optimization .....	111
3.1.5	System Landscape Optimization (SLO) .....	115
3.1.6	Cost Optimization .....	115
3.2	Implementing an Optimization Concept .....	120
3.2.1	Development Plan .....	121
3.2.2	The Topology of the Production Landscape .....	122
3.2.3	The Realization Strategy .....	124
3.2.4	The Topology of the Development Landscape .....	125
3.2.5	The Organization of Implementation and Support .....	126
3.2.6	The Creation of the Global Implementation Plan .....	126
3.3	SAP's TCO Initiatives .....	127

## **4 Optimization of Service and Support 131**

4.1	Service and Support Optimization .....	134
4.1.1	Triggers .....	134
4.1.2	Procedure .....	135
4.1.3	Factors .....	136
4.1.4	Health Check .....	138
4.1.5	Quality .....	139
4.2	Optimization Project .....	141
4.2.1	Factors and Components .....	141
4.2.2	Service and Support Processes .....	142
4.2.3	Tools and Technology .....	151
4.3	Examples and Scenarios .....	153
4.3.1	Optimization Project .....	153
4.3.2	Service Desk .....	155
4.3.3	Incident and Problem Management .....	157

## **5 Optimizing Business Processes 161**

5.1	Methods and Approaches .....	161
5.2	Performance Optimization .....	163
5.3	System Landscape Optimization (SLO) .....	167
5.3.1	Reasons for SLO .....	167
5.3.2	Key Aspects of SLO .....	168
5.3.3	Support Services and Software Tools .....	169
5.4	Business Workflow/WebFlow .....	172
5.4.1	What Is Business Workflow/WebFlow? .....	172
5.4.2	Applications and Potential Use of Workflows .....	173
5.4.3	Technical Foundations .....	175
5.4.4	Integration of Non-SAP Systems .....	179
5.4.5	Project Procedure .....	180
5.5	Reverse Business Engineering .....	181
5.5.1	Use and Benefit .....	181
5.5.2	Procedure .....	182
5.5.3	RBE Scenarios .....	184
5.6	SAP Exchange Infrastructure .....	185
5.7	Generic Business Process Redesign .....	188
5.7.1	Business Process Design as a Part of Strategy Implementation ...	188
5.7.2	Properties of Business Processes .....	189
5.7.3	Procedure for Redesigning Business Processes .....	190
5.7.4	SAP Business Maps .....	194

<b>6</b>	<b>The Way to Real-Time Enterprise</b>	<b>195</b>
6.1	Motivation .....	195
6.2	Business Process Monitoring Using Reporting .....	196
6.3	Business Process Monitoring with Real-Time Business Monitoring .....	197
6.4	Granular Key Performance Indicators .....	203
6.5	Summary .....	205
<b>7</b>	<b>Summary</b>	<b>207</b>
<b>8</b>	<b>Management Summary</b>	<b>211</b>
<b>A</b>	<b>Sources and Further Reading</b>	<b>213</b>
<b>B</b>	<b>About the Authors</b>	<b>215</b>
	<b>Index</b>	<b>219</b>

# Foreword

In the late 1990s, considerable financial investment advanced the development of information technology in enterprises. The "Best-of-Breed" approach was frequently used as a basis for IT investment decisions. This method often produces a heterogeneous, complex system landscape that is costly to implement and maintain, and has a lot of potential for optimization, which should not be underestimated.

In this book, we present various different options for optimizing an SAP system infrastructure. We do so by addressing different approaches—from business process optimization to performance analysis.

Therefore, this book is primarily intended for IT managers, system or solution architects, and senior consultants.

I would now like to express my gratitude to my co-authors Daniel Baumann, Dr. Björn Gelhausen, Johannes Hurst, Gerhard Krauss, and Achim Westermann, who, despite their grueling daily workload, found the time and energy to make valuable contributions that greatly helped with the creation of this book.

Finally, I would like to thank Galileo Press for their pleasant and constructive collaboration and, in particular, Florian Zimniak for his painstaking assistance.

I hope that you enjoy reading this book and that you find many useful suggestions for optimizing your SAP system landscape.

Heidelberg, May 2004

**Andreas Schneider-Neureither**

# Introduction

*"There is a time in the life of every problem when it is big enough to see, yet small enough to solve."  
(Mike Levitt)*

There can be no doubt that SAP applications have a very strong influence on how enterprises structure their work processes and do business today.

Even with a successful SAP implementation, however, if it is to run efficiently in the long term, there is still always a need to customize an ERP solution so it meets the demands of the market, business requirements, and the enterprise.

Many enterprises currently use SAP products to control production planning, purchasing and logistics, warehouse management and inventory management, production, vendor management, customer service, finance, personnel management, and other basic business activities. In addition, numerous enterprises also use other applications (best-of-breed) to meet the requirements of other areas. All of these different applications must be integrated and work collectively with the central SAP system.

During the last decade, the use of SAP systems and supporting applications has led to important cost savings and productivity gains in various different areas of business. SAP has provided the necessary integration and the standardized data models to help link and coordinate separate functional units in the organization. SAP implementations usually involve a high level of training due to the definition and implementation of strict procedure instructions and new business processes for adjusting the enterprise process flows to the SAP system. Many enterprises find the cultural aspects of an implementation and the constraint of adjusting to SAP processes more challenging than the actual technical tasks.

SAP has built a lot of flexibility into its solutions. If, however, the requirements and process flows of an organization don't map to the range of possible customizing settings, the results of an implementation can prove very frustrating. In trying to satisfy the requirements of a real-time enterprise, many companies today are faced with a no-win situation. They can either adjust internal process flows to reflect the processes of the SAP

**SAP Systems  
Change Business  
Processes**

application, or they can invest in solutions to integrate SAP with other enterprise applications.

**Modification of the Enterprise**

In addition to economic challenges, many enterprises must deal with a further complication. As a result of mergers and takeovers, enterprises must move from a traditional, vertically integrated organization to an enterprise made up of many different organizational parts. Consequently, they often have to concentrate on several different business areas. This complexity is heightened with the inclusion of dependencies among partner, vendor, and customer relations.

In business relations between enterprises—whereby companies aim to increase efficiency through closely linked internal systems—business processes have a direct influence on the architecture and functions required of ERP systems. Accordingly, enterprises are also increasingly forced to base the planning of ERP solutions on integrative and external requirements and less on internal or even application-specific requirements.

**"Next Generation ERP"**

The IT industry has recognized this situation and SAP also offers what is known as a "Next-Generation-ERP" or "ERP II" solution in the form of SAP NetWeaver. The integration of different ERP solutions is made possible with appropriately flexible tools and adapters and is increasingly geared toward the requirements of enterprises and their business processes.

Despite existing solutions, it is still difficult to map business processes generally across enterprise boundaries and heterogeneous systems. Quite often, however, these integration solutions are based purely on the exchange of data, so that the enterprise once again has to reflect the technical process flows. The already complex technical environment is often further complicated by partly inflexible processes that don't always cover all business requirements or dovetail completely into the existing landscape. Consequently, because of this integrative approach, the total costs of ownership (TCO)—the necessary time and capital for implementing and maintaining ERP solutions—can grow exponentially.

Many enterprises have discovered that implemented ERP solutions implemented don't always yield the added value anticipated. Based on our own experience and subsequent publications, it is apparent that in 2003, almost all companies are trying to gain more value from investments made in technology and services. This, in turn, leads to individual initiatives or complete improvement projects whose goal is to use existing resources optimally, and continually check and improve processes. We will now describe the most important initiatives.



The absolute highest priority is to improve the quality of the systems used, that is, the processes, and above all, the data in the systems. Until recently, the central data basis was the most important priority. Today, the central priority is ensuring the quality of data right from the start. This alone is not sufficient. It must be supported by high quality and consistent master data, even beyond system boundaries. Data quality is important for all systems, but it is critical for the SAP key systems and the related CRM or SCM solutions.

**System and  
Data Quality**

New business processes typically require new systems, new modes of operation, new roles, new training concepts, and targeted implementation. Because complete implementations can take up a lot of time and therefore, eat up a lot of money, the current trend is to tap the full potential of existing systems. A change to an existing system often requires only a few months, whereas a complete implementation can easily take several years. Reusability is supported by SAP applications that already provide the bases for the functions and technologies needed to implement new processes.

**Reusability of  
Solutions**

Given that a quick time-to-market is always imperative for the success of enterprises, usually, the reusability of existing systems for developing new or adjusted processes is the logical option for saving time and therefore costs.

The operation of systems and the support of business processes is evermore complex and requires an increasingly high number of systems. Ensuring availability and the speed of process flows in accordance with enterprise requirements and guidelines is becoming increasingly difficult. In the past, this was only possible if both the user and the service personnel—who was responsible for the operation—monitored the processes and intervened if necessary. With processes that go beyond system and enterprise boundaries, however, this is no longer possible. The requirements of a real-time enterprise can be met only if processes are automatically monitored for duration, integrity, and outcome, and if there are predefined roles and responsibilities for handling any problems that arise. Useful data and knowledge are a by-product of this real-time business monitoring and can provide a basis for the continued improvement of business processes.

**Real-Time  
Business  
Monitoring**

The lead times for business processes should be as low as possible. To optimize this, the work packages of a process must be exchanged quickly and completely between systems, made readily available to the person responsible and, if necessary, forwarded to other people. The Internet

**Quick Process  
Execution Times**

represents only the transport layer of the process. For this, processes often require synchronous interfaces between individual components, but frequently, however, the current forms of Enterprise Application Integration (EAI) and Electronic Data Interchange (EDI) can execute them only asynchronously. Business Services Orchestration (BSO) is one possible alternative to shorten each individual process step and therefore, reduce the runtime of the entire process.

#### **Integral Communication and Training**

Changes or continual improvements to business processes require a fully developed training concept to keep all relevant employees up to date. This will be even more difficult to achieve with future systems and cross-enterprise processes because the employees of the virtual enterprise can be in different organizations, enterprises, continents, and time zones. New communication and training concepts will have to be developed to address these challenges.

#### **Optimizing Service & Support**

One prerequisite for optimized business processes is that any problems that occur must be identified automatically, addressed quickly, and then, resolved efficiently. Support processes outside system and enterprise boundaries with scattered users and, above all, support organizations, require completely new support concepts and processes to deal with these new requirements. Not only must the data that forms the basis for the support processes exist electronically; you must be able to forward this data between different support systems to the relevant support teams, without compromising its quality. The users and the support organizations must always be cognizant of what phase the solution process is currently at.

#### **Solutions for the Virtual Enterprise**

Today, the integration of systems and different organizations is frequently achieved by the use of EDI. However, with this process, only the data that corresponds to the paper documents of old is exchanged. It is often difficult to ensure that data is complete and synchronous beyond system borders. There is a need for intelligent solutions to develop optimal systems, to integrate systems and, in doing so, save interfaces or counteract incompatibilities with central master data systems or templates.

This book addresses the aspects listed in the three main areas of optimizing business processes, optimizing system landscapes, and the associated optimization of service and support concepts.

In **Chapter 1**, we provide you with an overview of the current range of SAP products. We present the functions and the technical components involved and show the first possibilities for optimization.

**Chapter 2** introduces you to the different architectures of SAP systems. First, we deal with the conventional system architecture of SAP R/3 and show how a system landscape can be represented with SAP NetWeaver. The essential SAP concepts—Change and Transport Management, Lifecycle Management, and Solution Management—are also introduced.

**Chapter 3** provides you with the necessary foundations for optimizing existing system landscapes or implementing new ones. We describe different optimization approaches and provide explanations indicating which approaches are best used in which particular cases.

**Chapter 4** describes the optimization of your Service and Support organization. We first outline the method of optimization, later identify factors for good support, the processes that need to be optimized, and possible support tools. The Best Practices for Service & Support Processes from ITIL form the main thread in this chapter.

In **Chapter 5**, we provide you with different methods to use in order to optimize your business processes: general performance optimization, system landscape optimization, business workflow, Reverse Business Engineering (RBE), SAP XI, and generic business process redesign.

**Chapter 6** deals with real-time business monitoring. The possibilities and limitations of RBE are compared with the alternative of using business intelligence tools for analysis.

**Chapter 7** provides a summary of the individual chapters.

In **Chapter 8**, we provide you with an overview of the optimizing strategies presented.

Lastly, in the appendices, you'll find a list of references and suggested further reading, as well as a brief biography of the authors.

## 2 SAP Architecture Concepts

*The central task of IT infrastructures in enterprises is to provide seamless support for value-added chains, within the company and beyond system boundaries.*

The information necessary for the operative business and strategic development of a company can usually be found in different IT-supported areas, and is stored and processed in different systems. However, in many large firms, an exchange of data between the central IT systems and those of customers, business partners, or external employees is not uncommon.

You must consider these different requirements when designing system architecture, which should be flexible enough to allow for changes that can be integrated without too much effort.

### 2.1 SAP System Architecture

Many enterprises currently use SAP products to control production planning, purchasing and logistics, warehouse management and inventory management, production, vendor management, customer service, finance, personnel management and other basic business activities. As mentioned in Chapter 1, the mySAP Business Suite includes various business modules and numerous sector and cross-sector solutions.

A distributed system of this nature requires a common infrastructure, which is provided by SAP NetWeaver and, in particular, by the SAP Web Application Server (Web AS)—the successor to SAP Basis. This type of system architecture is subdivided into three layers, which make up the basic services of a business application system: the presentation layer, the application layer, and the database layer. Each of these layers performs certain functions and constitutes a part of the overall system landscape.

System  
Architecture

The *presentation layer* enables the user to interact with the relevant application. This interface is typically called the GUI (*Graphical User Interface*) and it is used to execute the application logic, utilizing the other layers in the combined infrastructure. Applications are executed in the *application layer* while the data to be processed is managed by the *database layer*.

The distribution of system functions over several layers means that an SAP system is extremely scalable. The separation into layers means that the

overall load can be distributed over several servers. The functions of a single layer can also be distributed over multiple servers. This is made possible by, among other things, the network communication of the participating servers.

### Presentation Layer

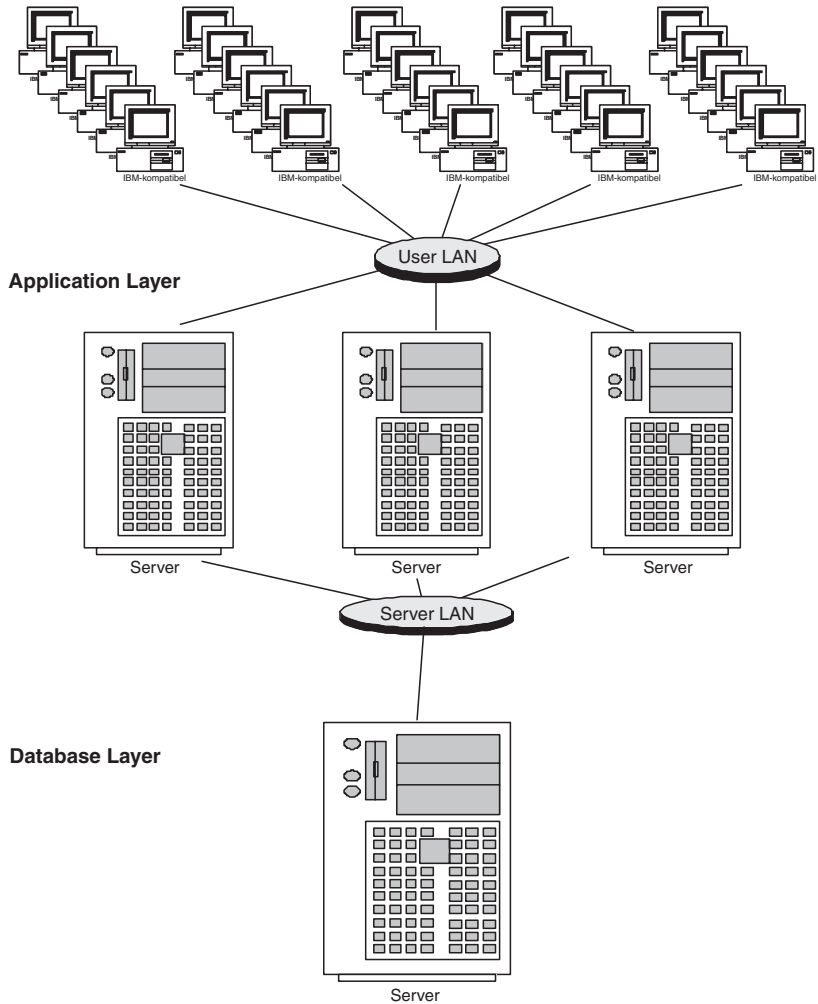


Figure 2.1 Three-Layer Model

In the days of SAP R/3, we spoke of one-, two-, or three-level architecture, depending on whether the services of the SAP R/3 system ran on one, two, or more servers. With one-level system architecture, all processing tasks occur on a single server. Two-level configurations are usually

implemented with special presentation servers, which are responsible for preparing only the graphical interfaces. This is usually performed in such a way that the end users use their workstation (usually a Windows PC) as a presentation server via which they can access data that is stored and processed on a separate server. With a three-level configuration, a separate server is used for each service, that is, the presentation layer, the application layer, and the database layer run on different machines. The application layer, in particular, can be distributed over several application servers to counteract a possible high load caused by end users, data-exchange processes, or background processing.

Another prerequisite for the layers is that they should offer a high degree of flexibility with regard to hardware and operating-system environments. mySAP Business Suite is a standard software, so you must be able to use it on different platforms. Because the SAP system architecture supports a wide range of operating systems and hardware platforms, SAP tries to remain as platform-independent as possible. A Java version has been written for the SAP GUI, for example, which means that it can be run on almost all operating systems.

The application logic of mySAP components can run on numerous operating systems and hardware platforms. It should be noted, however, that the application programs cannot be run on different platforms without some amount of customization. An environment must be created that will allow for programs to be independent of the operating system. This is achieved using the SAP Web AS, or its predecessor, SAP Basis.

SAP Web AS is a later development of the SAP Application Server technology. New technologies were added to process or forward HTTP requests, for example. This is the task of a new process—the *Internet Communication Manager* (ICM). It can communicate via the Internet as a server and as a client.

**SAP Web  
Application Server**

Figure 2.2 shows the architecture of the earlier SAP Basis compared with the new SAP Web AS architecture. You can see the Internet communication enhancement in the form of the ICM.

Like SAP Web AS, SAP Basis provides the runtime environment for all SAP applications and ensures that the application is optimally embedded in the system environment. It defines a stable architecture framework for system enhancements and includes tools for the administration of the entire system. Additional tasks include the distribution of resources and providing system interfaces.

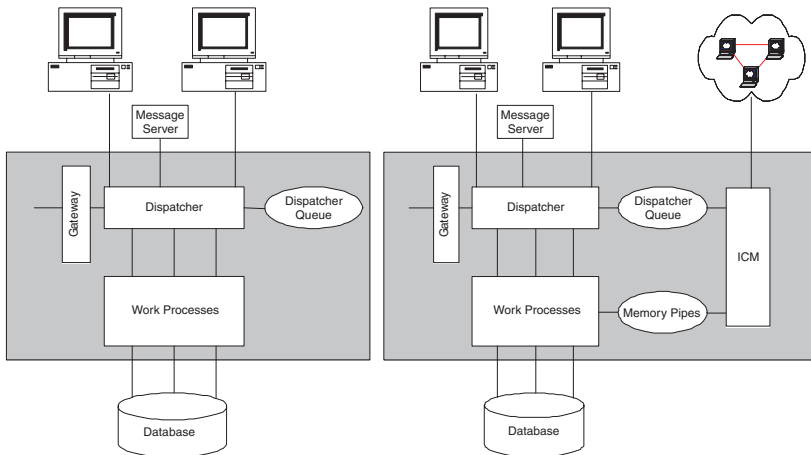


Figure 2.2 The Architecture of the SAP Web Application Server

**Processes** In contrast to the operating system, this runtime environment is a mass of parallel cooperating processes. On each application server, these processes include the dispatcher and a number of work processes, depending on existing resources. Special work processes can be defined according to the services for which the application is to be used, for example, an application server can be used exclusively for dialog processing. Therefore, we will define only dialog work processes on this server. You should use a different application server for background processing. Batch work processes are defined there for background processing. Communication between the dispatchers of an SAP system is done via the message server. There is always only one message server per SAP system. Communication between systems (R/3, R/2, external systems, and so on) is made possible thanks to the *SAP Gateway*.

The *SAP Internet Transaction Server* (ITS) is used for communication between an SAP R/3 system and a Web server. The ITS changes SAP screen images into HTML format and provides specific Web applications, which are known as *Internet Application Components* (IACs). You can also use SAP ITS as a gateway between a mySAP component and a Web server.

As previously mentioned, since the introduction of the SAP Web AS, new technology has been used to process HTTP requests (or other protocols) directly from the Internet, or send them to the Internet as HTTP client requests. To support this new development, another process has been added to the SAP kernel, namely, the ICM, which we already described.

Therefore, the definable processes available for the SAP Web AS are the message server, the dispatcher, work processes, the gateway process, and the ICM.

Standard Web protocols such as HTTP, HTTPS, Web DAV, SOAP, or SMTP are supported with the ICM. Output is in standard formats such as HTML, XML, or XLST. Requests and responses from the Internet with a specific URL (server/port combination) are processed with the ICM. The ICM process uses what are known as *threads* for parallel processing. Different threads have different tasks that can be carried out simultaneously. Thus, for example, the `Control` thread takes incoming requests and forwards them to the `Worker` thread, which processes a connection's requests and responses.

The services that run on a server are merged and managed together by an administrative unit, the instance, which is formed for this very purpose. Instances can carry out different tasks, depending on the configuration. If only one instance is connected to the database of an operative SAP system, it is referred to as the *central instance*. The central instance must be able to carry out all necessary services and should be configured appropriately. If an application server is defined with dialog work processes, this server is referred to as a *dialog instance*.

**Instances**

In large SAP systems, you can distribute the application layer over several instances. By distributing the load over several servers, you can improve the performance. Alternatively, you can run time-critical or load-intensive tasks on separate servers. If, for example, several hundred end users will be working with the SAP system, you can define different dialog instances to be used to achieve an evenly distributed load via logon groups. Should you need to output certain printouts—such as account statements or dunning letters by a certain date or time because they are a prerequisite for further procedures—you can define an application server so that it is available only for these tasks.

Each of these instances offers *buffer* areas for different objects, such as for programs, dictionary objects, screen structures, or table contents. These buffers are built up while the instance is being used and are constantly being optimized. Different algorithms are used to organize the buffer so that it can store frequently needed data. If the buffer is not large enough (a typical scenario), existing entries are removed from the buffer to accommodate new entries. Instances are also used to configure high-availability solutions, for which you will need special HA software (High-Availability software, provided by third-party vendors) that automatically

**Buffer**



stores services and functions on another instance if the central instance fails, thereby ensuring the continued operation of the SAP system.

The best way to see how the different layers (presentation, application, database) are interconnected is to take a look at the process flow of an end-user request.

**Request Process Flow**

The *dispatcher* is the central process on the application layer for processing SAP transactions. Its main functions are to distribute the request load to the individual work processes, establish connection to the presentation layer, and organize communication between work processes. When an end user makes processing entries from his or her workstation, using his or her user menu (presentation layer), these entries are converted into a special format (SAP GUI protocol) and forwarded to the dispatcher. The dispatcher places this request in the dispatcher queue. The queue is then processed following the "first-in, first-out" principle. The dispatcher distributes queries to free work processes that carry out the processing. SAP transactions are usually implemented in several worksteps. A query is created for each workstep and is assigned to a free process, enabling each transaction to be processed by several work processes until that transaction is complete. The result of the processing is then returned to the presentation layer via the dispatcher, whereby the SAP GUI interprets the data received and generates the output screen.

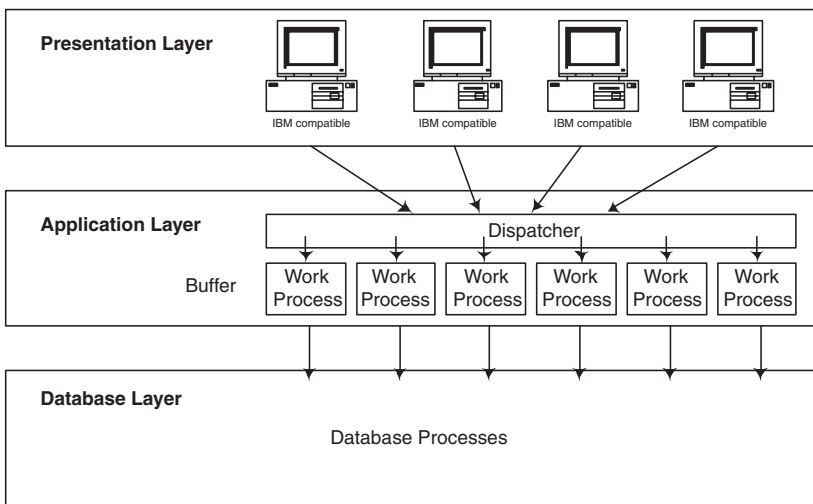


Figure 2.3 Processing a User Request

After we present an overview of the SAP system's architecture, we will examine the individual steps in greater detail.

## 2.1.1 Presentation Layer

The *presentation layer* allows the end user to interact with the relevant application. This layer is presented in the form of the SAP GUI. The *SAP GUI* is a universal client for accessing SAP R/3 or mySAP functions. The SAP GUI works as a browser. It receives information on the "what, when, where, and how" from the SAP server and displays the content on the user interface.

SAP GUI

The SAP GUI offers easy access to all SAP functions including application transactions, reports, and system administration functions. Thanks to the clearly defined menu structure, navigation is simplified. The SAP GUI is available in over 20 languages, which is integral to the development of international business processes today.

SAP has optimized the protocol between the presentation and application layers with regard to network load. When a user makes an entry, only the necessary data is sent via the network to the client. This also means that communication between the clients and the application server can also be produced within a WAN without additional loss. Therefore, for example, companies with several subsidiaries can use a central SAP system to handle business processes and users can access the system from their local sites.

The SAP GUI family comes in three different formats, each of which has its own unique selling point and is suited to a particular user group.

### ► SAP GUI for Windows

The SAP GUI for Windows has been specially designed for Windows operating systems. It provides an environment similar to that of Windows applications and also offers integration with other applications in the same way as the Windows environment (using OLE interfaces or ActiveX controls). In addition to the standard functions offered by all members of the SAP GUI family, SAP GUI for Windows offers extensive integration with Microsoft Office and links with mySAP components such as mySAP SCM.

Extensive  
Integration

### ► SAP GUI for HTML

The SAP GUI for HTML allows access to certain transactions using a Web browser. For this access, however, you will need the SAP Internet Transaction Server (ITS), which controls the actual conversion to HTML. Using the services of the SAP ITS and HTML templates, screen elements of SAP transactions are converted to HTML. Not all elements can be converted to HTML, but all standard elements such as tables, lists, menu trees, and so on can be displayed. The use of SAP GUI for HTML is therefore best suited for occasional users. Access to SAP functions using the

Access Using the  
Web Browser

SAP GUI for HTML doesn't require a separate client on the end-user's desktop, just a Web browser, for example, Microsoft Internet Explorer. To develop a business process with the help of a Web browser, you need *Internet Application Components* (IAC). IACs are SAP transactions that have been specially developed for execution via the Internet or intranet. They are based on *Business Application Programming Interfaces* (BAPIs), which use objects and methods to access SAP processes and data. IACs are addressed by the SAP ITS with the help of the SAP GUI protocol.

The SAP ITS acts as a gateway between the HTTP, the SAP GUI, and the Remote Function Call (RFC) protocol. Basically, the ITS is like a layer between the presentation layer and the application layer in SAP architecture, which means that the three-layer architecture becomes a four-layer architecture.

**Platform-Independent and Direct**

► **SAP GUI for Java**

The SAP GUI for Java was developed to achieve platform-independence in the area of the SAP frontend. The SAP GUI for Java is a uniform SAP frontend for many platforms, providing front-end functions with the help of JavaBeans. The technology behind it is a combination of Java and C++. The ActiveX controls, which are used in the world of Windows, are written as JavaBeans. Existing C/C++ libraries on the desktop in question are used by SAP GUI for Java to provide functions such as network communication, data transfer, or others. The SAP GUI for Java is installed on the desktop as an application (browser-based installation) and communicates directly with the SAP Web AS.

**Summary** The presentation layer of the SAP system implements the GUI and therefore allows the end users to work with the application software. We can differentiate between different implementations of the user interface: SAP GUI for HTML, SAP GUI for Windows, and SAP GUI for Java. With SAP GUI for Java, you can use the GUI on almost all platforms, thanks to the Java Virtual Machine (JVM). The SAP GUI for HTML also enables you to interact with the application logic, regardless of which platform is used, provided the client is a current Web browser. The SAP GUI for HTML is implemented on the server side, however, it can be used only by end users. Only SAP GUI for Windows and SAP GUI for Java provide a development environment. Another restriction with the Java or HTML execution is that there is no integration with MS Office.

**Web Dynpro** Web Dynpro technology is a new development. It can be used for programming the user interfaces. Web Dynpro technology provides a development and runtime environment for Web applications, based on server-

side scripting. Java Server Pages (JSPs) and a tag library are used for the interfaces. The Web Dynpro technology closes significant gaps between the typical Web development tools and the need for a cost-efficient, easy to adjust, manageable professional user interface for business solutions. It adheres to guidelines that also apply to SAP dynpro technology.

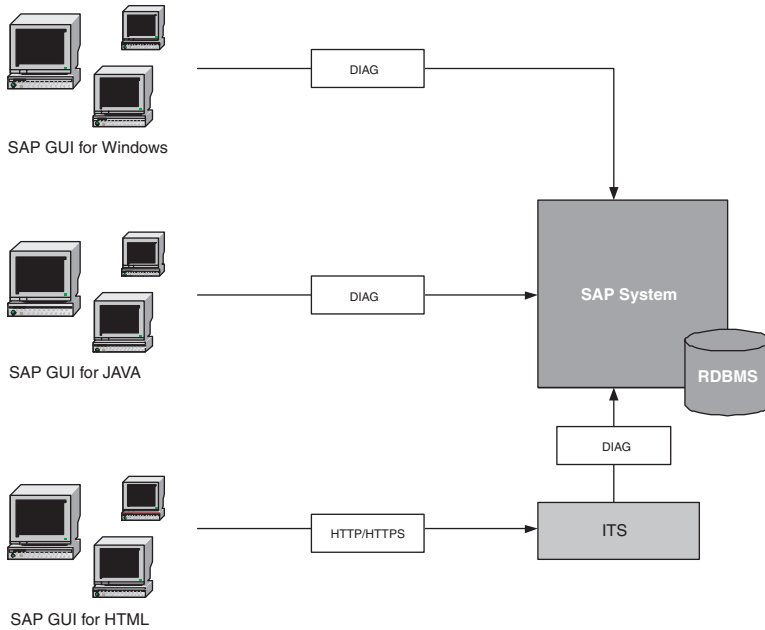


Figure 2.4 SAP GUI Family

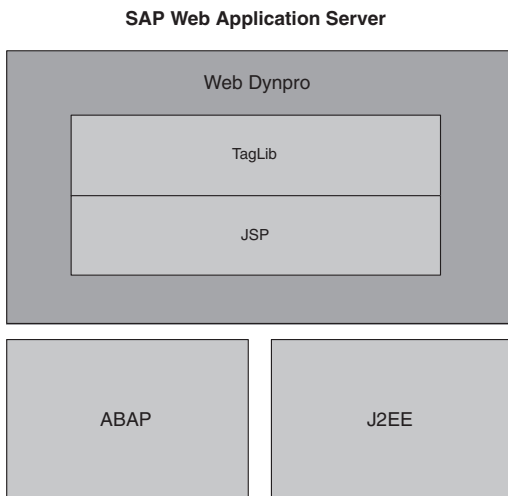


Figure 2.5 Web Dynpro

## 2.1.2 Application Layer

In recent years, the development of business software has grown to unforeseen proportions. Increasingly, you can integrate more business processes and show how business processes can work outside the realm of the company. This means that the potential usefulness of this type of software to an enterprise has grown enormously. Higher demands are made on planning, maintenance, and administration, however, mainly due to the integration of the Internet. Requirements such as platform-independence, security, development versioning, multi-user possibility, internationalization, and so on, are becoming increasingly important, and therefore, the underlying system environment must conform to these requirements.

### SAP Web Application Server

The *SAP Web Application Server* (Web AS) provides an environment in which all applications, in addition to the particular characteristics of the underlying operating system, can be developed, managed, and executed. SAP Web AS offers a platform for efficient development and for using Web applications and is a component of all mySAP installations. The server contains the presentation and application layers in its environment and ensures the independence of the database system used. It combines reliability, scalability, international language support, and connection to the *Change and Transport System* (CTS) with open standards such as HTTP, HTTPS, SMTP, WebDAV, HTML, XML, and server-side scripting. With SAP Web AS, SAP offers a uniform infrastructure for J2EE- and ABAP-based applications (*ABAP Personality* and *Java Personality*).

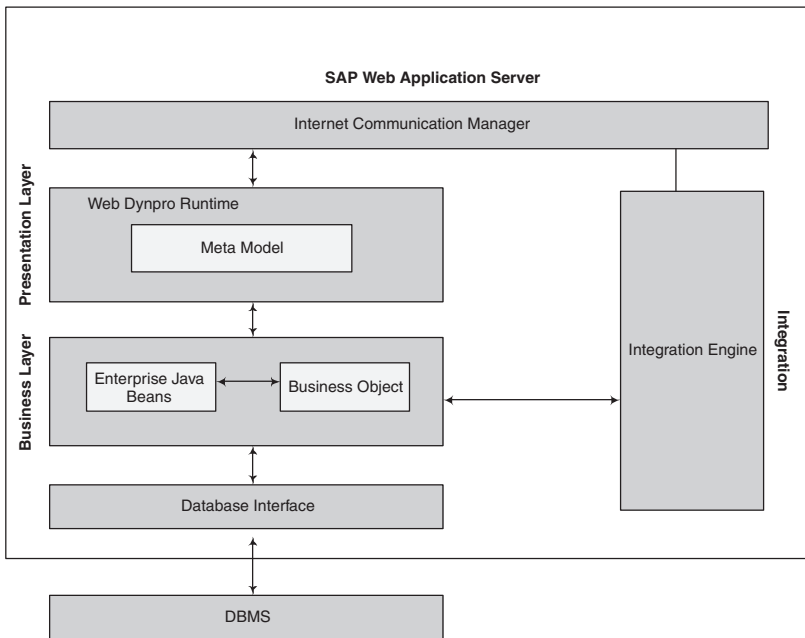
### Internet Communication Manager

The SAP Web AS enhances SAP's application server technology (SAP Basis) for the Internet world and is therefore fully integrated into the SAP environment (development environment, user administration, authorization concept, system monitoring, communication protocols). Another process added to the SAP kernel is the *Internet Communication Manager* (ICM). It uses threads to communicate via the Internet. If the incoming HTTP request is processed by a work process, data exchange occurs using what are known as *Memory Pipes* in the shared memory. The ICM distributes user requests to the presentation layer or sends Web service requests to the integration layer. If the ICM is addressed by an application or a client, it acts as a server. If the SAP Web AS initiates communication, however, then the ICM acts as a client.

Within a work process that oversees different tasks in the application layer, the *Internet Communication Framework* (ICF) provides the environment for handling HTTP requests. The ICF is the layer between the ICM

and the processing of requests in the work process. This makes it possible for work processes to generate Web-compliant content, which is forwarded to a browser via the ICM. One option for creating this content is to use applications with *Business Server Pages* (BSPs), which have been developed in the SAP system using the *Web Application Builder*.

To ensure a high degree of flexibility, the SAP Web AS is functionally divided into different layers.



**Figure 2.6** SAP Web Application Server

In the presentation layer, you can develop user surface for a Web application directly using JSPs or BSPs and the corresponding JSP tag library or using the new Web Dynpro technology. The underlying business layer provides content in Java or ABAP.

The most important function of an application is the *business process logic* that processes business-critical data. This logic runs in the application layer of the SAP Web AS and can transfer the results to a user interface or an external application. It can be written in ABAP or in Java, based on the J2EE standard. This offers greater development flexibility. The application layer of the SAP Web AS makes up a fully functional J2EE-certified runtime environment that processes the requests sent by the ICM and delivers dynamic results. New application programs, based on a component

**Business Process Logic**

model for application logic, can be developed with *Enterprise Java Beans* (EJBs). You can also access application objects written in ABAP. The conventional functioning of BAPIs is further supported. In addition, the SAP Web AS supports standards such as XML, SOAP, WDSL, UDDI, and so on to allow for the development of and access to Web services. These Web services can be used within an enterprise or outside the realm of the company, with different SAP or non-SAP applications and entirely platform-independent. Enterprises can therefore use Web services to help define business processes that affect several business partners with different application systems. To help you react quickly and flexibly to market requirements, you can use the following tool—ABAP Workbench.

**ABAP Workbench** The development environment of the SAP Web AS, the *ABAP Workbench*, is a strongly integrated environment for developing application objects. Over the years, the ABAP Workbench has proven to be extendable and reliable, so the concepts underlying this development environment are also applicable to the Java environment. This ensures that the robustness and reliability integral to developing enterprise-critical application programs is also inherently true for the Java personality.

**Database Layer** The *database layer* of the SAP Web AS offers many outstanding features. Thanks to Open SQL, it ensures optimal access to data via the ABAP environment. Optimizing functions, such as buffering SQL statements or data buffering, are used. This interface ensures that the business processing logic can be developed independently from the underlying database and the operating system. The database interface monitors the conversion of Open SQL statements—from the ABAP statements to the appropriate database statements. When interpreting the Open SQL statements, the interface carries out another syntax check and ensures that the capacity of the SAP buffer, which is used to process any statements in the SAP system, is used optimally. Then, SAP transfers experience gained in the ABAP world to the J2EE development environment. Database independence is also ensured, for example, via the use of open standards such as *Java Database Connectivity* (JDBC). The functions of Open SQL for ABAP are transferred to an Open SQL for Java and a range of *Application Programming Interfaces* (APIs) is provided for Java development.

The SAP Web AS forms the basis of any mySAP component system, be it mySAP Customer Relationship Management (mySAP CRM), mySAP Supply Chain Management (mySAP SCM), or others. Besides the open, integrated development environment, concepts such as connection standards, platform-independence, scalability, high availability, globalization,

manageability, change management, and even security and performance are also considered critical.

## Connection Standards

Several connection standards, such as J2EE Connector Architecture or Microsoft.NET Connectivity, are supported. In this way, you can mix different products and applications so that non-SAP applications can also be integrated with the SAP Web AS. To ensure that it's possible to communicate with all possible applications, many protocols and formats are supported, such as COBRA, FTP, SMTP, or RFC as SAP standard protocol. XML documents can also be interpreted.

The technical foundation of the Business Framework consists of the different connection technologies provided by SAP Basis or the SAP Web Application Server. These standards mean that you can integrate the different SAP functions, programs, objects, and so on, outside system and platform boundaries. *Remote Function Call* (RFC) technology and *Intermediate Documents* (IDoc) form the basis for executing SAP functions in distributed systems. This technology is embedded in the business process scenarios via the Application Link Enabling (ALE) distribution model, and via the definition of SAP business objects, known as BAPIs, which we already introduced and will elaborate on below.

Connection  
Technologies

*RFC* is the SAP variant of the *Remote Procedure Call* (RPC). It allows functions to be executed in other programs. There are different remote function call procedures to satisfy the different requirements of individual applications and their business processes:

Remote Function  
Call (RFC)

- ▶ With *synchronous RFC* (sRFC), function modules in external systems can be called in realtime. With this type of RFC, the client waits until the server has completed processing.
- ▶ Long-running processing can be split into different function calls using *asynchronous RFCs* (aRFC), which are then processed in parallel.
- ▶ If errors occur during the processing of aRFC or sRFC, it is impossible to detect if the error occurred before or after processing. It is also impossible to establish whether or not the RFC was even executed. With *transactional RFC* (tRFC), data consistency and unique execution is ensured. Transactional RFC is asynchronous and assigns a *Transaction Identifier* (TID) to ensure that if data is sent numerous times because of network problems, it is recognized by the server. This helps to avoid data being processed several times, which can lead to erroneous information



in the application. One condition of asynchronous processing, however, is that parameters can only be transferred from the client to the server. The direct return of information or status information is not possible.

- ▶ The *queued RFC* (qRFC) is an enhancement of the tRFC. With qRFC, the calls are placed in queues. This ensures that function calls are processed in a fixed sequence.
- ▶ Web-enabled exchange of information with XML is carried out by *HTTP-RFC* (xRFC). Function calls are sent via HTTP to an SAP system in the form of an XML document based on the SOAP standard.

**ALE and IDocs** *Application Link Enabling* (ALE) is an interface technology developed by SAP to link application systems (SAP systems and non-SAP systems) to each other via a distribution model. ALE enables systems on different platforms to communicate with each other. Data is exchanged asynchronously using *IDocs* (Intermediate Documents), a data format defined by SAP. IDoc is an SAP standardized document format that allows you to link different application systems via a message-based interface. It is similar to a container used to exchange data between SAP systems, or between SAP systems and non-SAP systems, because it features a neutral data structure. IDoc types are assigned different message types that are linked to business objects from the *Business Object Repository* (BOR). Thus, IDoc types are assigned to specific business processes.

**BAPI** Another important SAP interface is the *Business Application Programming Interface* (BAPI). BAPIOs are interfaces based on real business process scenarios. The idea of the BAPI is based on an object-oriented modeling concept, which is converted in the implementation of BAPI as RFC-enabled function modules in SAP applications. Using SAP Business Connectors,<sup>1</sup> however, BAPIOs can also be called via the Internet.

**Telephone, Fax, and Email** SAP architecture offers many different functions to help provide fast and efficient communication for optimizing business processes within a company and between companies. The Internet is also increasingly being used to develop and optimize processes. There is also a need, however, for the exchange of unstructured information using media such as the telephone, email, and fax, and exchanging structured data such as XML documents, worksheets, or electronic forms. Many different services and interfaces are required for this, which are provided by SAP architecture via APIs, special user interfaces, or interfaces to partner products. These services can include:

---

<sup>1</sup> The *SAP Business Connector* is a middleware product that permits the integration of SAP business process operation using open and non-proprietary technologies. It uses the Internet as a communication platform and the data format used is XML/HTML.

- ▶ Communication via the SAP Enterprise Portal
- ▶ Sending and receiving documents via email or fax using SAPconnect
- ▶ Integration of groupware products
- ▶ Telephone integration using SAPphone
- ▶ Calendar and scheduling overview
- ▶ Generic object services such as SAP Business WebFlow or SAP ArchiveLink for integrating scanned documents with the relevant business processes.
- ▶ XML documents

### **Platform-Independence**

The SAP Web AS is platform-independent. Consequently, enterprises enjoy great flexibility in the choice of their hardware or operating-system platforms, depending on their requirements or the existing IT infrastructure. To ensure this flexibility, SAP adheres to a strategy that supports all platforms that are persistent and have an adequate market share. Consequently, numerous combinations of hardware, operating systems, and database are possible. In discussing the SAP GUI family, we already described the necessity for the presentation layer to be platform-independent.

### **Scalability**

Application server technology, which SAP has further developed as the SAP Web AS, ensures the high degree of scalability of an SAP installation. Consequently, the scaling of SAP systems—according to growing business requirements—is ensured and can process a growing number of business procedures. If you distribute the application layer over several application servers, offering additional resources to support the growing load, then you can easily expand your SAP landscape. There are also software-based possibilities to distribute the load evenly over several instances. Load distribution using the SAP Web Dispatcher can also be supported to process HTTP requests. The SAP Web Dispatcher acts as a "software-web switch" between the Internet and the SAP system, made up of one or more Web application servers. In this way, there is only one entry point for HTTP requests in the system. The SAP Web Dispatcher also provides load balancing by forwarding requests to the application server with the current highest capacity.

## **Globalization**

The latest professional Web applications should satisfy different language and business requirements. To meet the requirements of globalization, the SAP Web AS offers a range of functions that support the expansion of business process flows. These functions address concepts such as internationalization, currency support, time zones, security aspects, and much more, with country-specific conditions and language differences also taken into account. The implementation of Unicode, for example, means that several languages can be supported concurrently.

## **2.2 Security**

In a business environment in which there is an increasing use of open networks and cross-enterprise business relationships and processes, secure data transfer between the Internet and the company intranet is integral to doing business. The SAP Web AS provides an infrastructure offering far-reaching functions in the area of security in heterogeneous environments. Business processes and their data in applications and Web services are protected against unauthorized access and misuse. Access authorizations and roles can determine who may view and process data. In addition, security aspects regarding the operating system and the database can also be considered separately. The SAP Web AS offers various guidelines to ensure technical system security, which can be converted into enterprise requirements with the help of a security plan. The most important aspects of security that must be addressed include:

### **▶ User Authentication**

Only authorized users can have access to the system; unauthorized persons cannot copy an identity. Users can execute only those tasks for which they are authorized.

### **▶ Protecting Integrity**

No unnoticed changes can be made to data.

### **▶ Protecting Confidentiality**

Data and communication must be protected from unauthorized reading and eavesdropping.

### **▶ Non-Repudiation (liability)**

Reliability and legal liability must be ensured.

### **▶ Checking and Logging**

Activities and events must be recorded so that they can be accessed at a later point in time (for example, audits).

There are a range of mechanisms available for user authentication in the SAP system.

► **User IDs and Passwords**

To save passwords, the system converts a user's plaintext password with a one-way hash routine into an appropriate hash value that is stored in the database. The one-way hash routine guarantees that the original plaintext password cannot be evaluated from the hash value.

► **Secure Network Communications (SNC)**

SNC is available for user authentication when the SAP GUI for Windows or RFCs are used. SNC uses an external security product for authentication between the communication partners instead of the conventional authentication (i.e., user name and password). If, for example, the security product Public Key Technology is used, then a *Public Key Infrastructure* (PKI) will be needed.

► **SAP Logon Tickets**

After authentication on the SAP system, a user can issue himself or herself an SAP Logon Ticket to provide Single Sign-On (SSO) for several systems. The ticket can then be presented to other systems (SAP or non-SAP systems) as an authorization token. The user won't need to enter a user ID with a password, but can access the system once the logon ticket has been checked.

► **X.509 Client Certificate**

As an alternative to user authentication with user ID and passwords, when using Internet applications via the *Internet Transaction Server* (ITS), the user can present X.509 client certificates. Here, user authentication is executed on the Web server using the Secure Sockets Layer protocol (SSL), where the transfer of passwords isn't required.

The individual functions can be supported by additional features. User administration can be supported via the use of a central directory service that would allow the company to maintain user roles and authorizations efficiently. In a distributed system landscape with several SAP systems, this aspect is also supported via the use of a central system for user administration. The SAP Web AS offers these *Central User Administration* (CUA) functions. They allow for several systems and their users, along with authorizations and roles, to be maintained centrally, and then automatically distributed to the systems in the system group using ALE. In addition to a defined organizational management and the Directory Service mentioned above, a certain automaticity of functionality can also be recorded.

**Network** The network infrastructure is of paramount importance to the security of a system. The network must support the communication needed by the company and its requirements excluding unauthorized access. A clearly defined network topology can help to avoid security risks based on software errors (at the operating-system level and at the application level as well), or access to the network, such as unauthorized eavesdropping.

The most important SAP services to ensure network security are the SAProuter and Secure Network Communications (SNC).

**SAProuter** The *SAProuter* is an SAP program for transmitting SAP system connections through firewalls. The SAProuter is a proxy in the network interface layer (NI layer)<sup>2</sup> of the SAP system. It can also log connection activities at various levels of detail. The SAProuter, alone, cannot control access to the network. Rather, it must be used in connection with a firewall system.

**Secure Network Communications (SNC)** SNC is a software layer in the SAP system architecture that offers an interface to an external security product. You can increase the security of the SAP system with SNC via implementing additional security functions not directly offered by SAP systems (for example, by using smart cards for user authentication). SNC offers security at application level. This means that a secure connection between the components of the SAP system (for example, between the SAP GUI and the SAP Web AS) is ensured, independently of the communication connection or the transport medium. Therefore, you can rest assured that the network connection between two communication partners protected with SNC will be secure.

**Digital Signatures** SAP applications can use what are known as *Secure Store and Forward (SSF) mechanisms* to ensure the integrity, authenticity, and confidentiality of data. If these mechanisms are used, then the actual data itself is protected if it leaves the SAP system. SSF uses digital signatures and digital envelopes to protect data. The *digital signature* is a unique identifier for the signer. It cannot be forged and it protects the integrity of the data. (The digital signature is not valid for any changes made to data after signing.) The digital envelope ensures that the data content can be read by the intended recipient only. SSF needs a security product to carry out its functions. Since Release 4.5, SAP delivers the *SAPSECULIB* (SAP Security Library) with SAP systems as the standard library of the SSF service. SAPSECULIB is a software solution especially devoted to digital signatures. An additional SAP-certified external security product is necessary to support cryptographic hardware (for example smart cards, security boxes, and so on) or digital envelopes.

---

<sup>2</sup> NI is SAP's abstract network protocol based on TCP/IP.

## 2.3 Change and Transport Management

All enterprise applications require intensive maintenance throughout the software lifecycle. In order to customize an application to reflect business processes, the company must usually create its own application settings. To support this maintenance effort and stay current with all the changes, specially developed tools for the particular configuration settings are used and continual updates are made to the system landscape. However, for security reasons, changes should not be implemented in the business-critical productive environment; instead, they should be made in a separate development system of the productive system. In this way, changes can be tested before they are actually put into use. SAP recommends that you use a separate system or independent layer for each sub-task in the development or change procedure. Generally, a three-system landscape is constructed for each component system in a system group, consisting of a development, quality (test), and productive system. Changes or developments are executed in the development system and SAP mechanisms are used to transfer them to the test system for extensive testing. After successful testing, the changes are transferred to the productive system to be used for the relevant business procedures. For this, the SAP architecture offers a complete Change Management and Transport service that fulfills business-critical requirements.

The special data structure of the SAP system is decisive for handling changes. In addition to business settings (customizing), which are only relevant for certain SAP clients,<sup>3</sup> the SAP system always contains cross-application settings and objects, that is, client-independent customizing and the Repository.

All development objects in the ABAP Workbench are stored in the central repository and are client-independent. These are mainly programs, tables, screens, function modules, classes, and so on, which are organized in *packages* (previously referred to as *development classes*). The packages contain semantically-related development objects. Changes can be made to the repository in different ways. Any necessary company enhancements to the standard repository are implemented with customer developments, therefore, new objects are created. The company uses the customer name range for these objects to differentiate them from the standard SAP objects, which is important for software upgrades. Modifications or customer enhancements, on the other hand, are changes to

ABAP Workbench

---

<sup>3</sup> A *client* is a self-contained unit within an SAP system, under commercial law and from an organizational and data systems point of view.

the repository via which standard SAP objects are changed or expanded with customer-developed objects (Customer Exit).

**Customizing** Via customizing, you can adjust the business settings in the system to meet the requirements of the business procedures. These settings must be carried out when an SAP system is first installed and can, if necessary, be adjusted during operation due to enterprise-specific or, for example, legal changes. Examples of customizing settings are company codes, plant, warehouses, and so on.

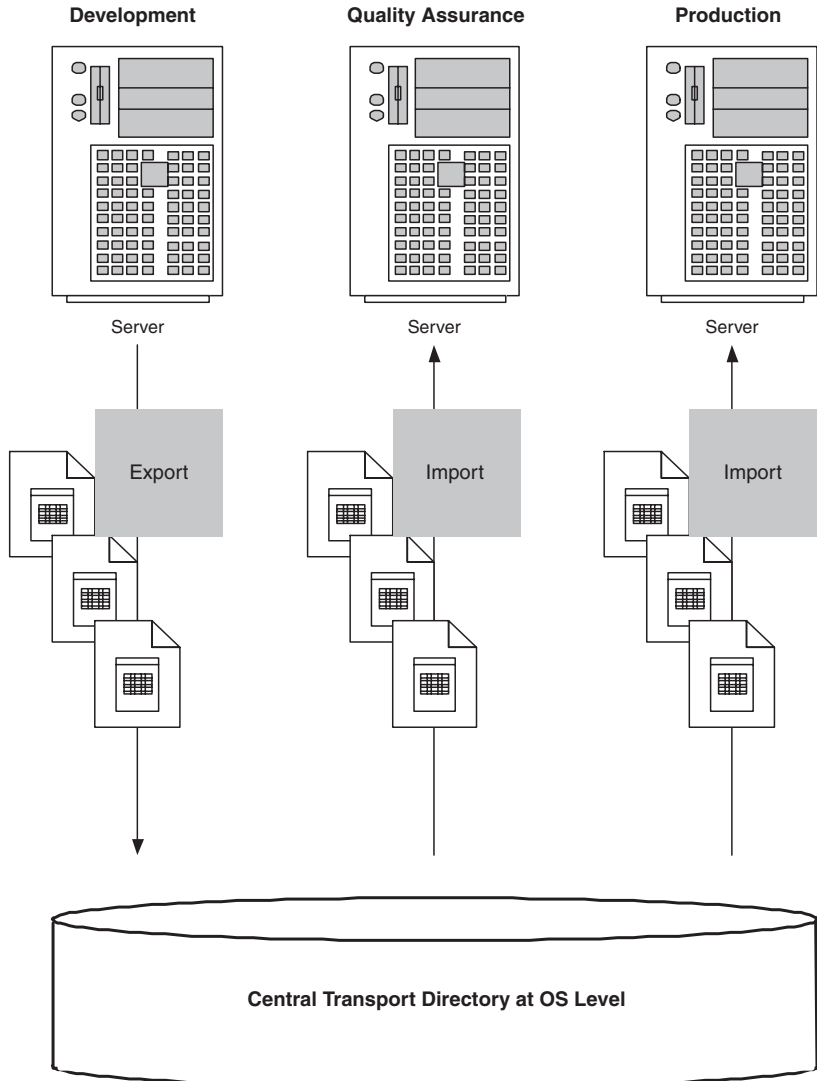


Figure 2.7 System Landscape with Transport Directory

Change Management and the necessary transport services, provided by SAP architecture, support the need for a company to execute, test, and use changes and developments on different levels. The recording and transport of object changes and settings is controlled by the *Change and Transport System (CTS)*. Therefore, changes are transferred from one system to another system using transport and the export and import phases. The *Transport Management System (TMS)* is used for importing into the target system. It manages all import queues. Technically, a central transport directory is used for individual exports from and imports to the corresponding systems. All systems in the landscape have access to this directory, which contains the data and structural information of the changes.

In the SAP system, changes to the repository or customizing are usually recorded in transport requests. This provides the following advantages, in the context of SAP software logistics:

**Transport Requests**

- ▶ Change history at object level
- ▶ Centralization of project-specific changes to different objects
- ▶ Distribution of changes/settings/modifications beyond system boundaries

The distribution of transport requests takes place along defined transport layers. All development projects carried out in an SAP system and transported via the same route to the target system (or systems) are brought together in a transport layer.

**Transport Layers**

Transport layers are created before the start of the first development in the transport route editor in the TMS. This transport layer is assigned to the development system as a standard transport layer. The allocation of development objects to the relevant transport layer is done using the previously mentioned packages. Objects supplied by SAP belong to the SAP transport layer. Additional transport layers are then usually needed only if new development systems are integrated into the system group.

We can differentiate between two types of transport routes:

**Transport Routes**

▶ **Consolidation routes**

To make changes transportable, you need a consolidation route for each relevant transport layer. The starting point of this consolidation route is the development system and the destination is the quality assurance system (in a two-system landscape, the production system). Changed objects, for which a consolidation route has been set up in the transport layer, are recorded in transportable change requests.



Once the request has been released from transport administration, the changed object can be imported into the consolidation system.

If objects are changed and no consolidation route has been defined for their transport layer in the current SAP system, these changes are automatically recorded in local change requests (or in customizing requests without transport destination). You cannot forward them to other SAP systems.

There can be one consolidation route only per SAP system and transport layer.

► **Delivery routes**

Once developments have been imported into the quality assurance system, they must be adopted in the production system or in several SAP systems (for example, additional training systems), for which delivery routes must be defined. These delivery routes have a source system and a target system.

By setting up a delivery route, you ensure that all change requests imported into the source system of the delivery route are automatically earmarked for import into the target system of the route.

Depending on the existing system landscape, you can set up several delivery routes with the same source system and different target systems (parallel forwarding), or attached one after the other (multi-step forwarding).

Transport control in the CTS ensures that all requests from the development system for all other SAP systems are earmarked for import in the same sequence in which they were exported. The same repository object or the same customizing setting can be contained in different requests at different development stages and consequently, you can avoid a new version being overwritten by an old one.

**Transport Request  
Types**

Because changes to different objects are handled differently, they are also recorded in separate transport request types. As a result, there are workbench requests that are mainly used for recording changes to repository objects in the ABAP Workbench and to client-independent customizing. There are also customizing requests that record the client-dependent customizing settings from just one client (i.e., the source client of the request). Additional request types such as copies or relocations are used to transport complete objects to a specific system. With copies, the original system of the objects remains the same; with relocations, the target system becomes the new source system.

Separate tasks can be assigned to a transport request for different project employees. Each employee can release his or her assigned task after completion of the development work (customizing work). Once all tasks have been released, the transport request can be released by the request holder. If, for example, a repository object is processed by a developer and recorded in a change request, it is reserved exclusively for this change request for processing. This object can be changed only by developers who are also participating with a task in this change request. Until the change request has been released, the development or maintenance of development objects is locked for all other developers not working on it. These developers can view the objects in question only.

To simplify transport administration, the transport system can also be workflow- or project-driven. If the transport system is workflow-driven, a workflow, which controls the work process, is automatically started on release. The agents of the individual steps are automatically determined and a work interface is provided to allow them to carry out their tasks directly.

If the transport system is project-driven, developments and customizing activities are planned in project structures of IMG project management. Changes that are independent of one another can be separated into different projects and imported into the next system independently.

The *Transport Organizer* offers functions for the creation, documentation, and release of change requests during the customizing and development process and for reorganizing the development landscape. The Transport Organizer tools are primarily intended for the development team and those in charge of a development or implementation project.

**Transport Organizer**

Whereas the Transport Organizer is helpful for the development team in the software development process until the release of a change request, the *Transport Management System* (TMS) supports the administration when importing the request into the target system. The TMS is started using Transaction STMS.

**Transport Management System**

The TMS enables you to organize, execute, and monitor transports between different SAP systems. Because all necessary information and functions are reproduced in the SAP system, in most cases, no additional user action is required at the operating-system level.

The TMS offers the following functions:

- ▶ Configuration of transport routes with a graphical editor
- ▶ Display of import queues in all SAP systems in the transport domain
- ▶ Import of all requests in an import queue
- ▶ Import of all requests for a project
- ▶ Import of individual requests
- ▶ TMS quality assurance
- ▶ Transport workflow
- ▶ Transport between SAP systems that don't share a transport directory

## 2.4 SAP NetWeaver

Integrated business processes constitute part of SAP core competencies. In heterogeneous system environments, however, and in situations where many different solutions are replaced by a single integrated solution, there is a need for technology that supports enterprises with the integration and development of their software products. In a heterogeneous IT environment, you can now define integrated, self-contained business processes outside department and enterprise boundaries. You can also use an existing infrastructure that protects current investments and adds value to applications already in place. The key here is *Enterprise Services Architecture* (ESA), an SAP concept for designing business applications, which is based on Web service standards. SAP offers an open integration and application platform for the implementation of this concept: SAP NetWeaver.

### Comprehensive Integration Approach

SAP NetWeaver uses the approach of integrating users, information, and business processes across technologies and organizations. It provides different technologies to support this integration. SAP NetWeaver is the successor to mySAP Technology and it expands on the functions of the latter, especially when it comes to *integration*. With this type of solution, all possible influencing factors associated with the areas of people, information, or technological processes such as infrastructure or heterogeneous components, must be considered.

The key words associated with SAP NetWeaver are People Integration, Information Integration, Process Integration, Application Platform, and Solution Lifecycle Management.

### ► **People Integration**

The integration of employees in the enterprise process flows is made possible via different technical components:

- ▶ *SAP Enterprise Portal* for uniform, personalized and role-based access to heterogeneous IT environments used for linking people and information
- ▶ *Collaboration* for the advancement of communication and collaboration between different teams in real time. It has functions such as collaboration room, team messages, team calendar, task assignment, and so on.
- ▶ *SAP Mobile Infrastructure*, which makes it possible to link up with the enterprise system via voice, mobile, or radio technology

### ► **Information Integration**

The following areas contribute to the integrated processing of information:

- ▶ *Business Intelligence* for the integration, analysis, and distribution of business-critical information to allow for informed decision-making
- ▶ *Knowledge Management* for access to unstructured data such as text files, presentations, or audio files
- ▶ *SAP Master Data Management* for data integration in heterogeneous landscapes—master data can be consolidated and aligned system-wide

### ► **Process Integration**

Business processes can be unified and given an integrated design thanks to:

- ▶ *SAP Exchange Infrastructure* for communication between application components of different origin; software components, interfaces, mappings, and content-based routing rules are defined
- ▶ *Business Process Management* for the design and operation of business processes in a dynamic IT environment

### ► **Application Platform (i. e., SAP Web Application Server)**

Important features of the application platform, which we addressed earlier in this chapter, are:

- ▶ *ABAP Personality* for supporting ABAP
- ▶ *Java Personality* for supporting J2EE
- ▶ *Business Services and Features* for supporting business applications and procedures

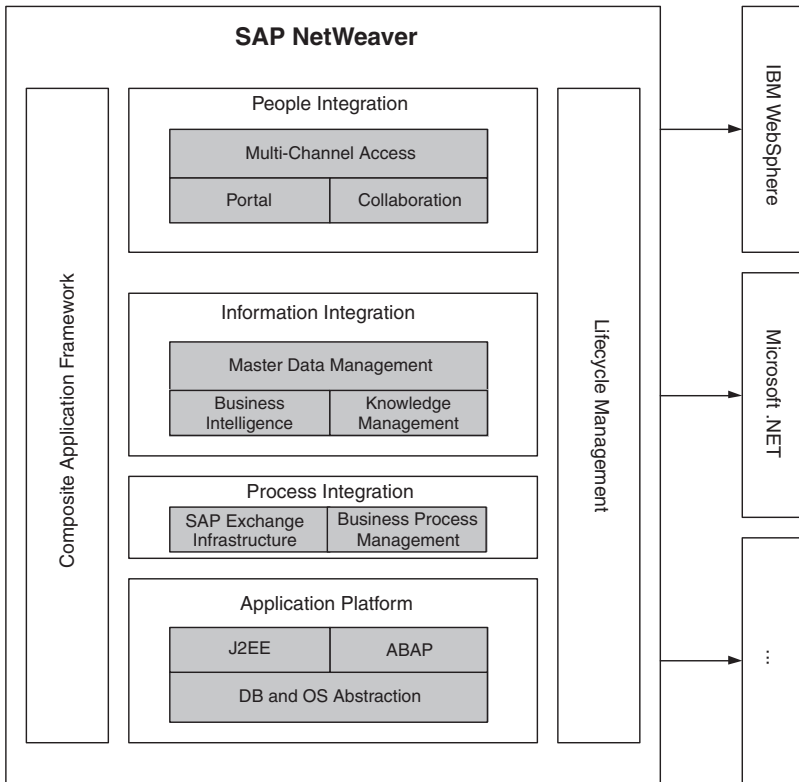
- ▶ *Connectivity* for supporting connection standards and Web services
- ▶ *User Interface* for linking the presentation layer with the application
- ▶ *Database and operating-system abstraction* to ensure the independence of existing databases and operating systems
- ▶ **Solution Lifecycle Management**

Solution Lifecycle Management supports the individual phases of the software lifecycle such as design, development, implementation, testing, or operation with the following core functions:

  - ▶ Customizing and testing
  - ▶ Data archiving
  - ▶ Hardware sizing
  - ▶ High availability
  - ▶ Installation
  - ▶ IT landscape
  - ▶ Performance
  - ▶ Software change management
  - ▶ System management
  - ▶ Technical infrastructure
  - ▶ Upgrade technology

SAP NetWeaver is the basis for different SAP solutions today. In future, all SAP solutions should be developed based on the concept of SAP NetWeaver or ESA.

The individual SAP NetWeaver components can also be integrated into the existing IT landscape by themselves. For example, SAP XI can be used as a solution for application integration or the SAP Business Information Warehouse (SAP BW; as the technical base of mySAP Business Intelligence) can be used for the integration and analysis of business-critical information. The SAP XI, for example, is a platform for the interaction of different application systems and therefore ensures the consistency of transactions.

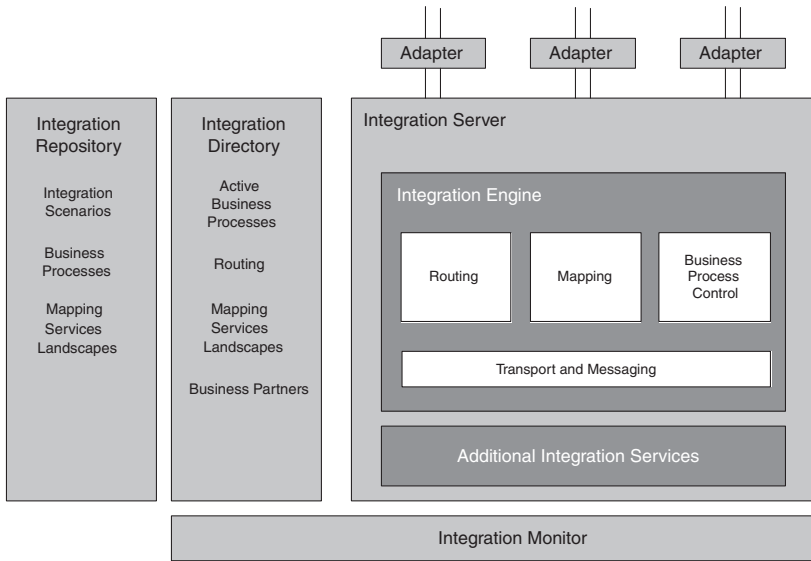


**Figure 2.8** SAP NetWeaver Architecture

As an integration and application platform, SAP NetWeaver provides predefined integration and business content in one solution. This preconfigured content helps to accelerate integration and in this way, SAP NetWeaver helps to reduce costs. Business processes are becoming increasingly complex and are planned beyond system and technology boundaries. Therefore, SAP is currently providing a technology platform that offers the services necessary to meet these new requirements. Business process flows can be designed flexibly because open technologies, such as Web services, are used.

**Predefined Integration**

In the Internet age of cross-solution and cross-enterprise business, closed solutions are being replaced by an open IT architecture in which end-to-end business processes are represented by different Web services running on several distributed component systems. Each of the component systems is specialized in particular services. The use of different components helps to provide optimal support for enterprise business, but it results in complex IT system landscapes.



**Figure 2.9** SAP Exchange Infrastructure

**Adaptability** It is essential that the IT architecture ensures the reliable, cost-effective operation of distributed landscapes, while simultaneously offering the enterprise the option to adapt the IT landscape to reflect the changing business requirements. The architecture must be open and flexible to meet these requirements.

SAP NetWeaver offers possibilities ranging from simple, compact IT systems installed on one server to highly available, scalable, and secure configurations for critical applications.

The optimal technical infrastructure for a given business requirement is determined by factors such as landscape type (development, quality assurance, production), main business processes, transaction volumes, security, and availability.

**Key Attributes of SAP NetWeaver** To ensure that key figures such as availability, performance, and *total cost of ownership* (TCO) represent the highest standards, SAP NetWeaver has the following key attributes:

- ▶ Multi-component architecture
  - ▶ Component-specific infrastructure
  - ▶ Component-specific upgrade, tuning, and availability strategies

- ▶ mySAP components are platform-independent
  - ▶ Component-specific performance improvement and scalability
  - ▶ Protection of existing investments in hardware, software, and IT know-how
- ▶ Platform topology specification
 

Can range from maximum consolidation (single server, one database) to maximum distribution (multiple servers, multiple databases on different platforms).
- ▶ The ability to run several clients on one physical system
 

This is particularly important for *Application Service Providers (ASPs)* that want to host independent customers on a single system.
- ▶ Migration tools to support platform changes
- ▶ Complete system management tools

A key factor for reducing overall operating costs over the entire lifecycle of an e-business solution is the ability to continuously provide an optimal IT infrastructure at each stage of the cycle and to customize the configuration to meet changing business requirements. Ultimately, SAP NetWeaver provides tools for the design, implementation, operation, and maintenance of IT landscapes. Predefined sample landscapes describe possible IT landscape configurations. Enterprises can use them as templates for designing their own IT landscape and then configure this IT landscape to best reflect their own specific business needs.

## 2.5 SAP System Landscapes

Several questions arise when planning system landscapes:

- ▶ What services do I need to execute business processes? What application components can I use to support these services?
- ▶ What type of data will be shared between these components?
- ▶ What additional technical components will I need?

You must answer these and many other questions during the first phase of an implementation project in order to define the system landscape. Depending on the enterprise- and business-specific requirements, these landscapes can differ significantly in structure.

In the next section, we present different example landscapes to illustrate the possible alternatives and their effect on availability, performance, scalability, and TCO.



## 2.5.1 Example Configuration: Internet Sales and Procurement

The infrastructure of SAP Internet Sales is based on an SAP CRM system and the SAP J2EE Engine 6.20. Almost all business processes are processed on the CRM system and some business applications also run in Java applications. mySAP CRM runs on the basis of the SAP Web Application Server (Web AS).

*SAP Internet Pricing and Configurator (IPC)* is used to configure products, process configurations, and determine prices. Special services for the area of e-business, such as search functions, are provided by the *TREX program* (Text Retrieval & Information Extraction). With TREX, simple text and HTML documents can be indexed and searched.

An *OLTP system* (Online Transaction Processing System), such as *SAP R/3*, is used as a back-end system for materials and finance applications. The *SAP BW* can be used to analyze business and draw up reports. It is an optional component, however, as is the *SAP Advanced Planner and Optimizer (SAP APO)*, which is used for queries regarding availability in the e-selling scenario, in Supply Chain Management, Demand Planning, and other areas.

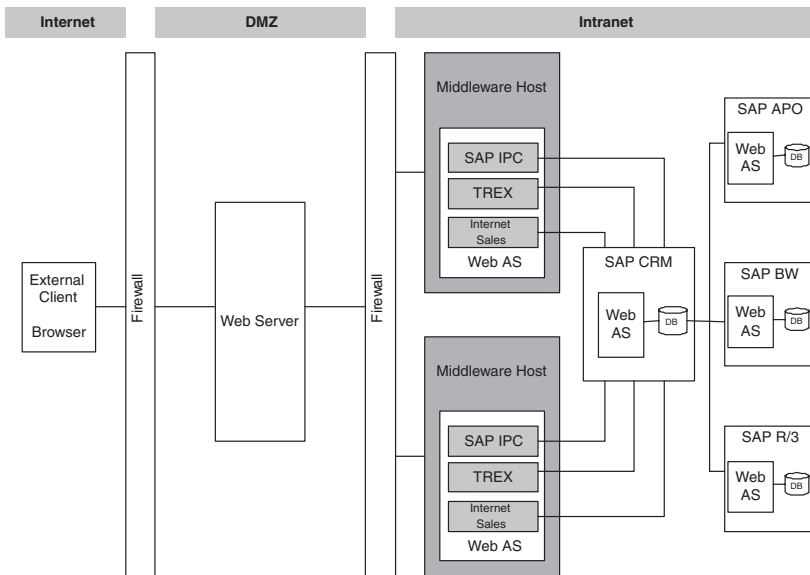


Figure 2.10 Internet Sales

### Defining Requirements

The design of the technical system infrastructure has a direct influence on the cost-effectiveness of the implementation. For this reason, the

demands made on the system must be clearly defined. You must consider different factors, the most important of which is what purpose will the system serve. This means that you must know if the system on which the critical business processes are to be run is a development system, a test system, or an actual production system. You should already have determined how many end users will work with the system. If you cannot calculate this figure, you must find some appropriate means in which to do so. Availability requirements must be defined, so that the necessary provisions can be made. You should note, however, that ensuring high availability is very costly. System security is another subject that you must consider as it too has a large influence on the infrastructure. The separation of different software components on a network and the use of firewalls does, of course, make the landscape more complex, something that also has an effect on landscape administration. When planning an IT landscape, a high degree of flexibility should also be a priority. Planning should be done in such a way that the landscape can be extended at any time and adjusted to fit new requirements.

The *Customer Relationship Management (CRM)* and *Enterprise Resource Planning (ERP)* systems are installed separately to ensure optimal performance and the appropriate choice of platform. The separation of Web services from the other components improves performance and scalability. A separate Web application server provides the runtime environment for these Web services.

A firewall separates the internal components from the external Web server, which can be accessed via the Internet using HTTP. Consequently, the application systems are protected from possible external access.

Each of the physical server platforms can be scaled independently. Companies can develop a separate high-availability solution for each software component, database, or host. It is important that all of the hardware and software is included in a high-availability concept. For each component, there is a special technology to provide for high availability. A redundant construction of network topology may be necessary for the network, while cluster technology can be used for the server and the operating system. Databases can be given high availability with the help of replication, for example, while the disk mirroring and SAN technology may play a role in the area of memory management. If there is a need for load balancing, you can do so using several application servers for middleware and the individual SAP component systems as well. In this way, the functions of the dispatcher or the SAP Web Dispatcher are used to distribute the load.

**Availability**

The overall e-business solution runs on separate servers and possibly on different operating system and database platforms. SAP XI, which forms part of the SAP NetWeaver, provides the system connections.

This clear differentiation between application and technical platforms ensures that companies can change or expand the platform without affecting the overall functional integration.

This setting can be used for productive systems with up to a few thousand intranet and Internet users.

### **2.5.2 Example Configuration: mySAP CRM Interaction Center**

The interaction center is a key area for executing business procedures with customer relationships and offers a complex platform for the interactive processing of these business processes. In the interaction center, agents create the links between the business partner and the business processes. The customer information necessary for this is provided there. The following scenarios, among others, can be reproduced in the interaction center:

- ▶ **Inbound telesales**  
for customers who want to place an order; they can contact the interaction center
- ▶ **Outbound telesales**  
for the support of sales campaigns
- ▶ **Information help desk**  
for processing customer queries
- ▶ **Service**  
for processing service queries and requests
- ▶ **Complaint handling**  
for processing complaints

The mySAP CRM Interaction Center is based on the CRM server, which contains the *Interaction Center Framework* (ICF) and the components SAPphone and SAPconnect. With *SAPphone*, you can integrate telephone functions into the SAP application and with *SAPconnect*, email functions (with the help of a mail server). Additional external products are also required to integrate telephony functions, such as the telephony gateway, the telephony server, and the interface to the telephone system for *computer telephony integration* (CTI).

To create, check, and distribute messages and announcements, which a supervisor can delegate to individual agents, you will need the *Broadcast Messaging Server*. The messages will then appear at the appropriate agents' work centers. You can define the priority, language, and expiry date of each message. Products can be configured, configurations processed, and prices determined with the help of the SAP IPC. The TREX program supports flexible document searches and the structuring of accumulations of documents using automatic document classification. With TREX, simple text and HTML documents, providing necessary customer information, can be indexed and searched. As in the first example (Internet Sales), an OLTP system (such as SAP R/3) can be used for materials, management, and financial applications and to evaluate key figures from the different scenarios of SAP BW. SAP APO is necessary to monitor the availability of products.

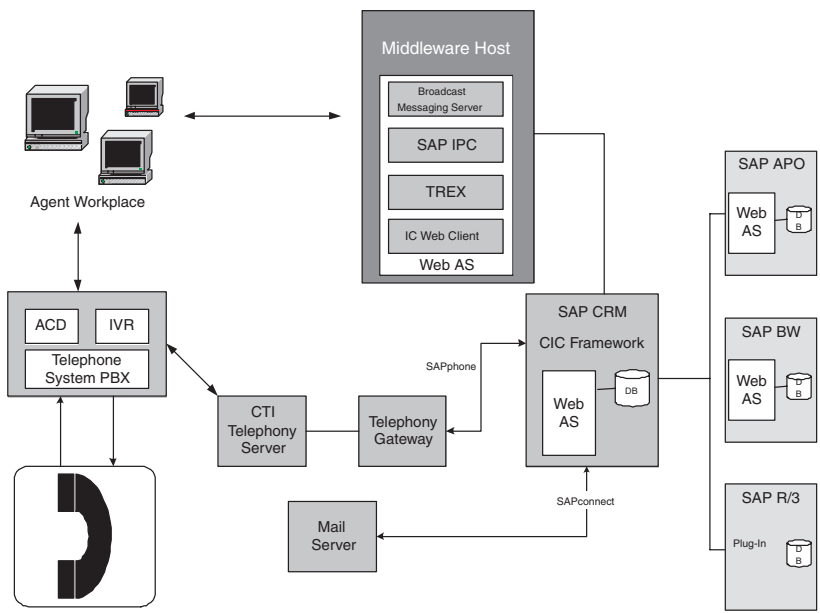


Figure 2.11 mySAP CRM Interaction Center

### 2.5.3 Example Configurations for System Landscapes and Client Strategies for Change Management

A system landscape always comprises all application systems involved and their corresponding SAP clients. Software logistics are used to coordinate and distribute changes made to SAP software. When planning this landscape, you must consider different requirements that include:

Multi-System Landscape

- ▶ The security of productive data
- ▶ Quality assurance
- ▶ Optimal system performance
- ▶ Development standards
- ▶ Upgrade strategy

Additionally, there are factors such as hardware and maintenance costs and administrative effort.

When putting together a system landscape, SAP recommends that customers plan for a three-system landscape, with separate systems for the development, quality assurance, and productive environments. Companies with a large international system implementation may need more systems for the different functions. There may also be, for example, training, load-testing, or sandbox systems. The three-system landscape applies to each component system within the system group, which means that a three-system landscape should be built for an SAP R/3 or R/3 Enterprise and also for each additional component system such as for each SAP BW or each CRM system.

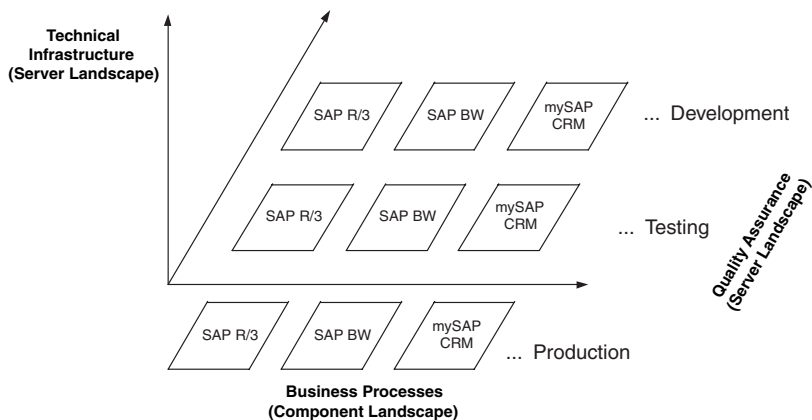
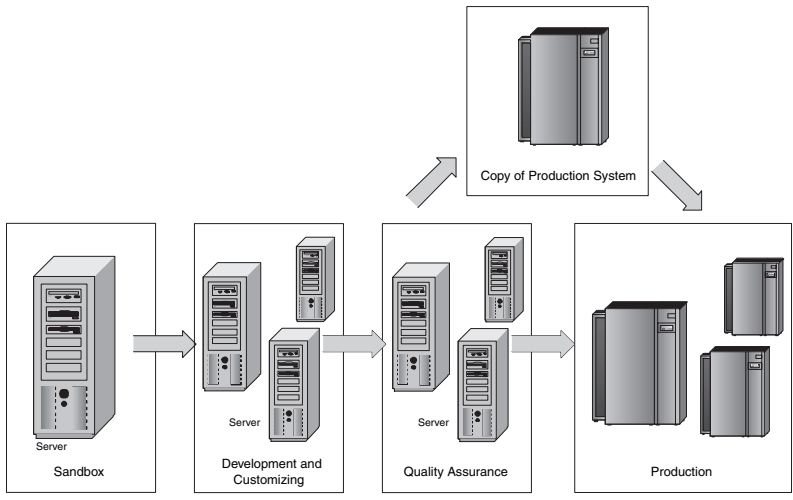


Figure 2.12 SAP System Landscape

Figure 2.13 shows a common landscape for quality assurance. The landscape consists of a sandbox system, a development and customizing system, a quality assurance system, a copy of the production system, and a production system. The sandbox system is used to test simple new functions without affecting development processes or planned test procedures. The development and customizing system is used for the actual development work or customizing settings, so that they are executed

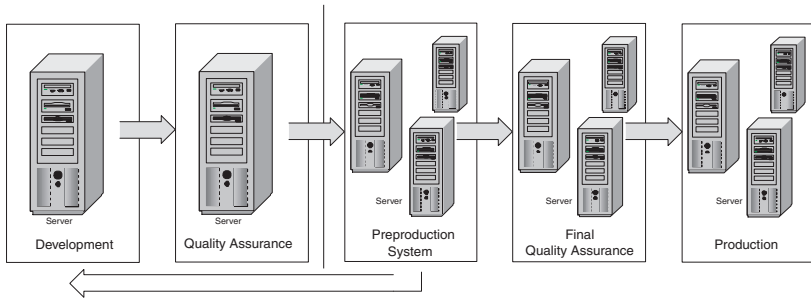
centrally according to their specification. These settings are transported to the adjacent quality assurance system where technical and functional tests are carried out on the new developments or settings. A copy of the production system is inserted to allow for integration tests in a production-like environment and under production conditions (interface tests, far-reaching changes, and so on). It is provided with all new developments to ensure that it is at the same development stage as the production system itself. Far-reaching tests can be devised that can, for example, also be used as load tests if you know that a high load will be generated, so you can see how the system reacts to certain load peaks. If all tests are successful and the settings or new developments generate the required functions, you can then import the changes into the production system where they will be available for the actual business processes.



**Figure 2.13** Quality Assurance System Landscape

You can differentiate between changes such as new developments or settings for new sub-processes and changes arising via end-user support. In the course of production support, small adjustments are frequently made to customize settings, or program errors are removed. This is necessary to ensure unrestricted production operation. These changes must be made available to the end users and at relatively short notice for their daily work. New developments or new sub-processes, however, are often scheduled for the long term, and are the result of temporal planning as part of the development and test process. They are transferred to the production system only at a specified time after the successful completion of development and test work. This scenario can occur with a graded multi-

system landscape so that the different change processes can be executed independently of each other. A development thread made up of a development and a quality assurance system is used for new developments that provide the thread for production support: a preproduction system, final quality assurance, and a production system. This is referred to as a two-phase system landscape.



**Figure 2.14** Two-Phase System Landscape

**Client Concept** The functions related to the Change Management process can also be distributed on separate levels with the help of a suitable client concept. Clients can take on different functions, such as:

- ▶ Sandbox
- ▶ Blueprint demo
- ▶ Customizing master
- ▶ Quality assurance master
- ▶ Quality assurance test
- ▶ End-user training

SAP systems are client-enabled, therefore, multiple clients can be defined. This means that business units can be defined as self-contained, logical subsystems with their own master and transaction data. As you can see in the previously mentioned client functions, clients can also be used for technical and organizational purposes. Consequently, a client concept can be designed to determine how different clients are distributed to the development, quality assurance, and production systems. This makes it possible to separate the different sub-processes in an implementation technically and organizationally, by uncoupling development clients and customizing clients, or defining several clients on the quality assurance system for different locations or applications. On the production system, however, you should apply only those clients that are used productively.

Multiple production clients require increased administration. Therefore, try to implement the least number of production clients possible.

## 2.6 Lifecycle Management

Distributed software solutions require more systems than do closed solutions. As IT landscapes grow, consideration of the TCO becomes an increasingly important factor. To lower the TCO, the IT landscape should be serviced, maintained, and extended throughout its entire lifecycle. Going live is not the key objective. A well balanced, well defined design prior to implementation and the constant improvement of the solution after implementation are equally important. The primary goal is for the solution to maintain a high value during its entire lifecycle.

Lifecycle

The lifecycle of an IT landscape can be broken down into different phases:

1. Design
2. Installation
3. Implementation
4. Operation
5. Change management

While the goal of a software environment is to support the business processes of an enterprise, it undergoes all of these phases. As the system landscape grows, the phases become exponentially more complex.

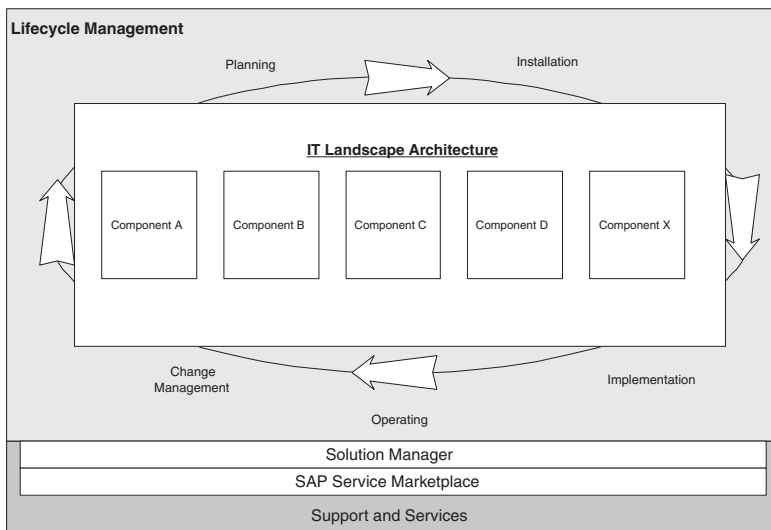


Figure 2.15 Lifecycle Management



## 2.6.1 Design

### Formulating Requirements

During the design phase, the requirements that will shape the solution are discussed. These requirements are primarily determined by the business processes to be supported by the solution. To optimally translate the requirements, the appropriate component systems or non-SAP applications must be selected, the communication routes set down, and the structure of the system landscape planned. You must consider to what extent any existing system landscape can be extended. The platforms to be used are also selected. Issues such as the appropriate security strategy or high-availability requirements must also be considered.

### Sizing

Once you have identified the necessary components to be defined for the system landscape, sizing must be carried out for the system. In sizing, you will determine how the individual systems should be designed so that the requirements for supporting business processes can be fulfilled in all situations. Realistic sizing must factor in hotspots and load peaks. This includes daily load distribution of, for example, end-of-day processing, monthly and yearly close, and seasonally conditioned load. During the sizing phase, you must estimate hardware requirements such as network bandwidth, main memory layout, processor performance, and so on. You should keep in mind various factors such as the number of users needed to work with the different applications, or the volume of data that will be created by this work or that must be distributed outside the realm of system boundaries. The estimated number of users should be determined by how much you think they will work with the system. Estimations made during this phase, which will serve as a basis for determining the hardware and network resources required, should be verified again at a later time with performance and load tests.

## 2.6.2 Installation

### New Systems and Upgrades

After the design phase, you must install the defined component systems. Both new installations and system upgrades must be taken into account. It is also important that there is as little as possible user access and also limited system downtime (for upgrades). The launch of the SAP Web AS saw the introduction of new upgrade technology which, for example, greatly minimizes downtime during an upgrade to SAP R/3 Enterprise (latest version of SAP R/3) because different phases such as repository import or activation can be carried out while the system is online. Furthermore, the release strategy of SAP R/3 Enterprise is designed in such a way that the technological basis of the system—the SAP Web Application

Server—can be updated separately from the application functions (application core and enterprise extensions), thereby reducing the complexity of an upgrade.

In heterogeneous landscapes with different applications, it is critical that these applications can be integrated without too much effort. For this reason, SAP NetWeaver offers an *Application Programming Interface* (API) that can be used by other installation routines.

### 2.6.3 Implementation

The installation of individual component systems alone does not yet deliver the complete application functions needed to establish the business processes of an enterprise. To do this, the individual steps and functions must be adapted to the circumstances of the enterprise. With heterogeneous landscapes and cross-system business processes, you must map the process flow in software and in the system, and coordinate the individual sub-steps with each other precisely. During this step, external applications must be connected and the necessary data flows must be implemented using interfaces. It is often necessary to formulate extra developments to achieve the desired functionality.

The success of this implementation phase depends to a large extent on how well the business processes are defined and documented. It is important to clarify how different procedures should run and what areas need to be considered during them. The more precisely these processes can be defined and the enterprise-specific conditions described, the more accurately the business process requirements can be reproduced in software components and program flows. New components and integration into existing systems must, of course, also be considered.

The implementation phase will be more complex if business processes that surpass system and enterprise boundaries are designed. With these scenarios, generally applicable requirements and component- or enterprise-specific prerequisites must all be considered. To a large extent, these settings depend on each other; they also influence the success of the software execution of business processes in detail. Settings for each component must be made depending on other components and in the event of adjustments; you must always remember that changes can affect other application areas and therefore, the business processes themselves.

Different tools are provided for the individual SAP components to support settings made during the implementation phase—from technical set-

**Defining Business Processes in Detail**

**Implementation Tools**

tings in the system to recommendations and templates for carrying out different processes. Among other things, all configuration settings are automatically connected to the previously mentioned Transport Management for software distribution. In addition, with the SAP Solution Manager, SAP provides a central platform for efficient solution management that supports individual processes from the implementation phase through system operation. The concept of SAP Solution Management will be described in greater detail at a later stage.

**Testing Settings** Settings that go beyond the system landscape must be tested very carefully to ensure the precise functioning of the processes and the integrity of individual landscape components. This should be done after every partial implementation. It is important that the sub-processes of the individual application components are correctly planned, that interfaces to other systems—whether SAP or non-SAP systems—can ensure the desired data exchange, and that the process sequence as a whole yields the desired results. To ensure that these tests run smoothly, well defined test scenarios, which can always be restarted and easily adjusted to new constraints, must always be in place.

#### **2.6.4 Operation**

After converting the relevant requirements into an integrated, cross-system software solution during the implementation phase, new requirements are inevitably made on the IT infrastructure. This type of infrastructure should offer the following characteristics:

► **High Performance**

A good overall system performance is a prerequisite for user satisfaction and for the effectiveness of the entire system solution.

► **High Availability**

The systems (SAP and non-SAP) form the business framework of an enterprise. Each system failure inevitably leads to an interruption of business.

► **Low costs**

The total cost of operation (TCO) of a system must be taken into account to keep the running costs of an application as low as possible.

The goal of running this type of infrastructure or its component systems is to support business processes; as such, it is essentially a service provided by the IT department of an enterprise for the other areas of business. It involves relevant processes and tasks, required on a daily basis, to ensure

that the characteristics listed above are maintained. The individual levels of the system should be considered separately. Operation can be divided up between the administration and monitoring of SAP components and the underlying operating systems and databases, the middleware used for communication and data exchange, and the areas of network and non-SAP applications.

The service to be provided by the IT department for other departments is usually described in *Service Level Agreements* (SLAs). An SLA should have a clear definition of the services to be provided and the criteria to gauge the quality of services. In addition, the method and type of reporting on service performance and necessary escalation levels—in the event of non-fulfillment of agreements—should also be defined.

**Service Level Agreement**

A generally applicable administration concept should be defined to support and align the individual tasks in systems operation. Such a concept would cover the organization of the tasks that form part of systems operation for an SAP solution and contain all necessary information for the successful administration and maintenance of the entire system infrastructure. It is necessary for you to differentiate between routine tasks carried out regularly to identify possible problems early and remedy them; reactive tasks that crop up in the event of failures; and proactive tasks that monitor and analyze certain key figures in an effort to avoid imminent resource bottlenecks, or, using support packages in order to avoid known problems in advance.

**Operating Concept**

Necessary tasks for the execution of systems operation for an SAP solution may include the following:

- ▶ Monitoring the hardware resources of the entire infrastructure
- ▶ Monitoring system resources to ensure performance
- ▶ Checking manufacturer's notes on known problems with hardware and operating systems
- ▶ Maintenance and monitoring of the database and database backup
- ▶ Checking the SAP operating parameters such as work processes, buffers, security settings, and so on
- ▶ Checking SAP Notes
- ▶ Implementing and monitoring output management (printing, fax, email, and so on)

## 2.6.5 Change Management

Due to the fast moving, flexible market requirements, adjustments to the systems, the infrastructure, and the application-specific process flows are unavoidable. Every change is not without its risks, however, which can have a deleterious effect on the operation of the software solutions. Badly prepared changes are the primary cause of operation problems and can even lead to system breakdowns. It is therefore vital that each change to be executed is carefully planned. Matters such as the expected effects of the change, necessary tests, or emergency strategies should be included in these plans. Predefined Change Management can help you to minimize the risk of inconsistencies that can arise due to changes in an operative system.

### Defining Processes

To reduce the potential for inconsistency, you should define the following processes:

#### ► Approval Processes for Planned Changes

Approval processes should, among other things, help to centralize Change Management so that all departments can check the influence of a change on their processes.

#### ► Risk Appraisal and Emergency Strategy

Possible risks must be defined from the outset and appropriate measures must be implemented in the event that these risks become real problems.

#### ► Informative Test Cases

Changes should not be executed directly on a production system; otherwise, you run the risk of permanently damaging productive operation. Adjustments must therefore first be transferred to a test system and tested thoroughly.

#### ► Documenting Changes

In order for changes to be carried out at any time and by anyone, you must log each correct setting.

SAP offers an integrated Change Management concept for SAP solutions containing different specifications of the software delivery: upgrade, support packages, and SAP corrections.

### Support for Upgrades

Completely new release versions are delivered in the form of upgrades, featuring new or functionally enhanced versions of SAP standard programs or other repository objects. With SAP Web AS, SAP provides new system-switch technology that minimizes the system downtime required

during an upgrade. In addition to the productive system, a temporary instance of SAP Web AS is installed that contains the new repository object. This means that most upgrade phases can be implemented while the productive system is online for the execution of business processes.

Errors in different software components are corrected with *support packages*. Support packages contain corrected SAP objects that replace erroneous objects. These packages are made available by SAP at certain intervals and can be implemented into the SAP system online, although this is not always advisable.

**Support Packages**

*SAP corrections* contain corrections for errors in individual objects, for example, ABAP programs that have been prepared by SAP because of problems in some application processes. These corrections can be implemented in certain actual cases.

**SAP Corrections**

With all methods of the SAP Change Management concept, modifications made by customers to SAP objects will be taken into account, because modifications come under the system change control and can therefore be harmonized. The Change Management process is supported in SAP systems by the Change and Transport System (CTS), as described in an earlier part of this chapter. The services offered by the CTS are extended with the SAP Web AS for the area of Java.

## **2.7 Solution Management**

Due to the fact that the phases in the lifecycle of an IT infrastructure become more laborious to execute as openness and complexity increase, there is a need for functions and tools that will support these processes. System landscapes become increasingly more complex as the dependencies of different applications grow in importance, as a result of the pronounced dispersal of business processes beyond system boundaries. Increased integration due to technical and business-critical demands must also be taken into account, as well as the increased significance of the Change Management processes or support. Consequently, in addition to the tools provided by SAP NetWeaver, SAP also offers a solution management platform that supports the administration of complete IT landscapes.

The *SAP Solution Manager* provides tools, services, methods, technical, and application-related documentation and templates that can be used for implementation, for support and for the operation of complete IT solution landscapes. It should be used as the central infrastructure for

**SAP Solution Manager**

solution management and provide the link between technical key values and the critical business processes.

SAP uses the following approaches in its Solution Management strategy:

► **Standardization**

Consistency and clarity can be ensured by standardizing services, methods, and technologies. This is important to keep the total operating costs to a minimum.

► **Solution orientation**

Since IT solutions (e-business solutions) are no longer made up of one system or one application, but of multiple applications of differing technologies, SAP uses a cross-system, solution-oriented approach to manage the entire infrastructure. The system solution is therefore considered to be a collection of integrated systems so that the dependencies of the individual systems and applications can be considered automatically.

► **Process Orientation**

To achieve complete control over the system solution, you must look at the main business processes that are integral to the complete value chain of an enterprise. Therefore, the focus is on the most important business processes and their individual technical steps. Among other things, monitoring these processes is organized more flexibly and the correlation of systems and applications to problems is more easily identified.

► **Openness**

To fully map the SAP administration solutions to the complete enterprise infrastructure, SAP targets its Solution Manager openly at all SAP and non-SAP applications. This ensures a complete overview of the software landscape to be tended to.

► **SAP integration**

With the help of the SAP Solution Manager, SAP follows the approach of setting up a continuous support infrastructure, by integrating itself into the customer's support process. Thanks to a constant connection of the customer system to SAP, services can be updated and executed, or customer support messages can be forwarded to SAP for processing.

With the help of the SAP Solution Manager, you should be able to have full control of the entire IT infrastructure to support the phases of implementation, operation, and support over the entire lifecycle of a software solution. This necessary and extensive standardized help to implement

complex administration strategies will enable the enterprise to reduce the TCO, or, at least, to keep it low.

SAP subdivides its concept of Solution Management into different main areas, each with a different bias (see also Figure 2.16):

► **Global Strategy and Service Level Management**

This covers the complete strategic management of an organization. It provides know-how and services to support the strategic specification of operating processes, monitoring and reporting on SLAs. It is intended for decision-makers.

► **Business Process Management**

This guarantees the smooth and optimal flow of daily business processes and offers help with business process monitoring, interface management, or job management. It is primarily intended for those responsible for business processes and application support.

► **System Management**

System Management provides a central administration environment for the technical management of the IT infrastructure. Here, the technical tasks of system maintenance can be processed, such as performance management, monitoring, and so on. It is intended for technical support personnel and system administrators.

► **Software Change Management**

The purpose of Software Change Management is to support the implementation of software changes such as release management or test management. It is used to standardize and automate software distribution and maintenance and test cases for a complex, distributed software landscape. This area is for project and development teams and application support.

► **Support Desk Management**

The Support Desk provides a support environment. By integrating end users, application support, partners, and SAP support, it enables you to define an efficient, integrated problem-solving process.

The SAP Solution Manager is a central platform that provides all the necessary tools for implementing an SAP software solution, monitoring it, and ensuring system operation and support. It is completely solution-oriented, which means that it not only allows for the support of individual systems, but for the entire IT landscape. It covers the entire lifecycle of a system solution (see Figure 2.17).



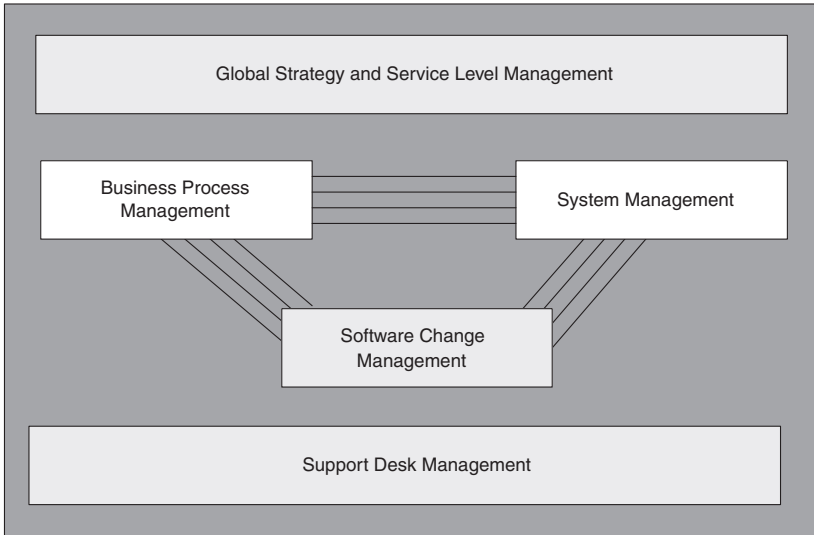


Figure 2.16 The Areas in SAP Solution Management

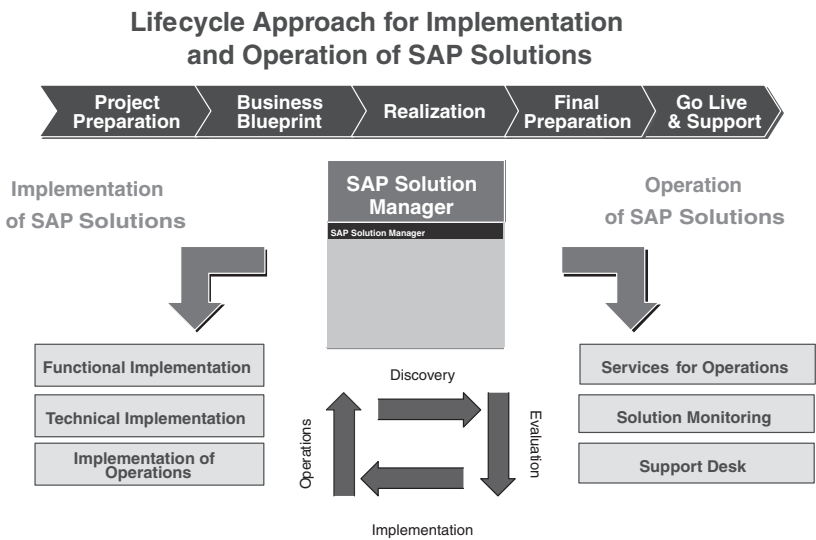


Figure 2.17 SAP Solution Manager Throughout the Lifecycle

**SAP Solution Manager for Implementation**

Basically, the SAP Solution Manager is divided into the SAP Solution Manager for Implementation and the SAP Solution Manager for Operation. The *SAP Solution Manager for Implementation* is an extension of the *ASAP method* (Accelerated SAP) and it offers functional and technical roadmaps for the support of implementation activities from the blueprint phase through system operation. With the help of what are known as Best Prac-

tices documents—documentation on specific application scenarios or configuration guidelines—content is provided that may be useful for the planning and implementation of software solutions and business processes. Customizing Distribution and the Test Workbench help you to set up application configurations and test cases in a process-oriented way.

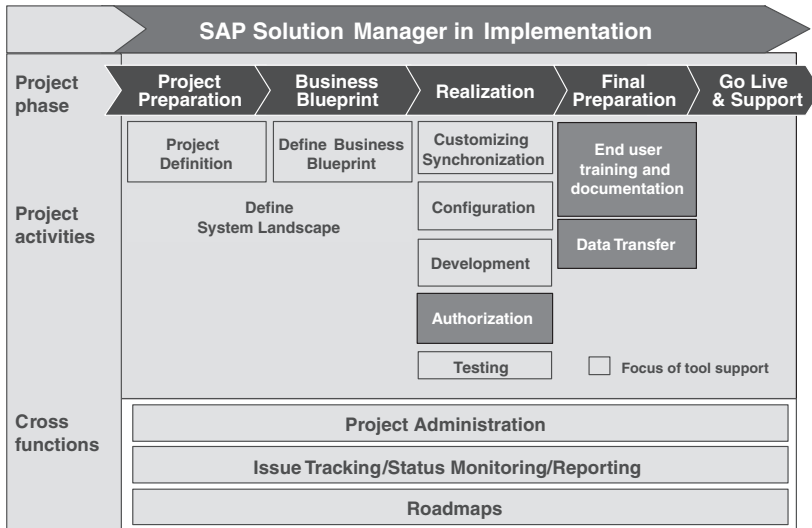


Figure 2.18 SAP Solution Manager for Implementation

The *SAP Solution Manager for Operation* offers functions for executing the complete system and application operation. It provides a platform for the provision and execution of SAP services for optimizing or securing SAP solutions and for monitoring the entire solution. Solution Monitoring is divided into Service Level Management, System Monitoring and Business Process, and Interface Monitoring. With the help of System Monitoring, the Solution Manager provides a central platform for monitoring the complete system landscape. This is based on the *SAP CCMS Monitoring Infrastructure* (Computer Center Management System). This infrastructure offers an object-based framework for monitoring components. Thanks to the CCMS, you can monitor all participating application systems because it provides a central view of the current system status. In addition to these mechanisms, analysis and reaction methods are also integrated. Business process monitoring maps the technical monitoring functions to business-critical processes via breaking these processes down into individual sub-processes and assigning them to software process flows. Therefore, you can assign a complete business process to different applications (cross-system) and monitor it centrally (interface monitoring).

### SAP Solution Manager for Operation

SLAs, which define the services to be performed by an IT department, can be controlled via Service Level Monitoring. Defined key values in individual component systems are analyzed at certain intervals and summarized in reports.

**Support Desk** In addition, the Solution Manager also offers the functions of the *Support Desk*. The Support Desk provides the environment of an extensive call center or message processing system. It offers functions for the efficient processing of internal company support messages, and evaluating and resolving them. SAP Notes are a big part of the Support Desk and are implemented with the help of the SAP Note Assistant. Complete message processing is also optimized with subjects such as escalation management, workflow integration, or solution database. Support does not end at the enterprise boundaries, however. If SAP support in the company is not in a position to resolve an application problem, then SAP Global Support (at SAP) is automatically involved via the Solution Manager/Support Desk.

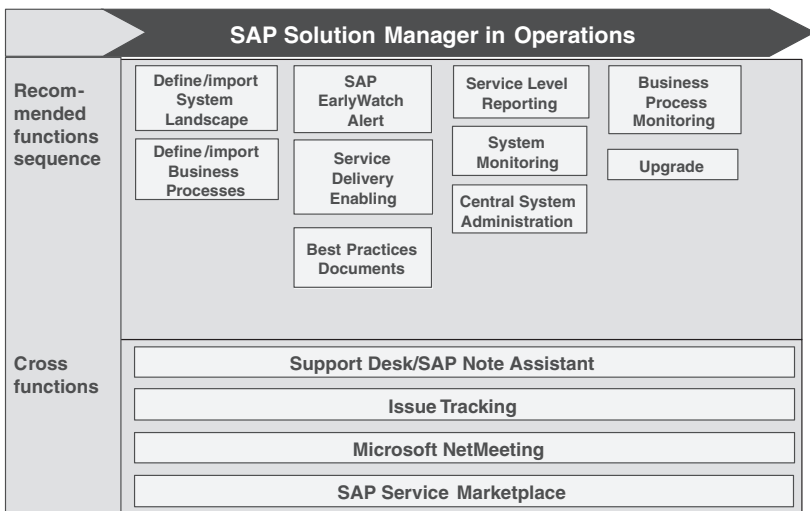


Figure 2.19 SAP Solution Manager for Operation

With the functions listed above, the Support Desk offers support for the construction of a user-driven support organization. This organization can be accessed directly from all connected SAP component systems. One particular advantage of the message creation is that the relevant system data for the system in question is automatically attached to the query, so that the message processor does not have to waste any time on the

lengthy compilation of this data. Figure 2.20 shows a diagram of a possible workflow using the Support Desk.

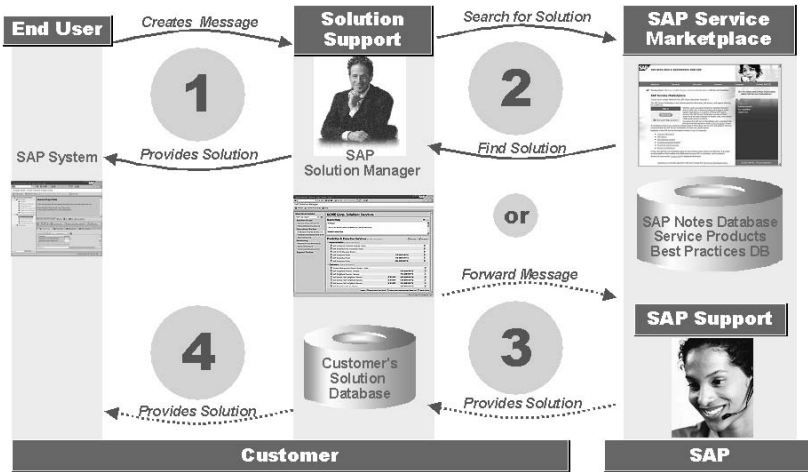


Figure 2.20 Support Desk – Workflow

## 2.8 Summary

SAP, when coupled with the different component systems (mySAP Business Suite), offers an integrated software system that addresses nearly all the business requirements of an enterprise. Software solutions that comprise multipart business-critical applications are designed in such a way that they implement and optimize the necessary business processes technically. Consequently, you must ensure the availability and performance of these solutions throughout the entire lifecycle. Different options are provided, based on the architecture of the SAP systems, to respond flexibly to the enterprise-specific and business requirements of a company.

The most important points regarding SAP system architecture are:

- ▶ The high scalability of SAP systems is ensured thanks to the use of client-server architecture (three- or four-layer model).
- ▶ All important component systems in the mySAP Business Suite are based on the SAP Web Application Server. This satisfies general requirements such as platform-independence, scalability, standardization, globalization, and so on.
- ▶ With the integration of new technologies, new standards are achieved in this Internet-driven world.

- ▶ With the advent of SAP NetWeaver, SAP offers an open integration and application platform for implementing the concept of Enterprise Services Architecture.
- ▶ SAP system landscapes are designed to meet the demands of the implementation of business processes, performance, availability, data security, and quality assurance.
- ▶ The SAP Solution Manager is used to support lifecycle management for an SAP solution. Its functions are used for all phases in the lifecycle—from the design phase through system operation.
- ▶ The functions of the SAP Solution Manager are solution- and process-oriented, resulting in the offering of services that address the entire software solution so attention can be focused on the critical-business processes.

# Index

## Numerics

- 2-phase system landscape 90
- 3-layer model 56
- 3-system landscape 88

## A

- ABAP Personality 23, 64, 79
- ABAP Workbench 66, 73
- Accelerated SAP 100
- Account Management 45
- Adapter 187
- Adapter Engine 187
- Advanced Planner and Optimizer 39
- ALE 67, 68
- Application adapters 29
- Application layer 55, 60, 64
- Application Link Enabling see ALE 67
- Applications Management 150
- Architecture 35
- Archiving 49
- ASAP 100
- Authentication 70, 71
- Availability Management 147

## B

- Background service 110
- Bank analyzer 46
- Bank Customer Account 45
- Basel II 46
- Bell Laboratories 139
- Best Practices, ITIL 144
- Broadcast Messaging Server 87
- Buffer 59
- Business Application Programming Interface 62, 68
- Business Information Warehouse 21
- Business Intelligence 79
- Business Maps 209
- Business Object 190, 191
- Business Object Repository (BOR) 68, 175, 177
- Business packages 28
- Business Process Management 21, 79
- Business Process Monitoring 113, 166

- Business Process Optimization 161
  - methods 161
  - performance 162, 163
  - process redesign 163
  - RBE 163
  - Reverse Business Engineering 181
  - SLO 162
  - Workflow 163
  - XI 163, 185
- Business Process Redesign 188
- Business Process Reengineering 107
- Business processes
  - architecture 190
  - description 189
  - properties 189
  - redesign 190
  - strategy analysis 190
  - task chain diagram 192
- Business Server Pages 65
- Business Services Orchestration 14
- Business Workflow/WebFlow 163

## C

- Capacity Management 147
- CATT 129
- CCMS 51, 101, 113, 166
- Central instance 59
- Central user administration 50
- Change and Transport Management 73
- Change and Transport System 64, 75
- Change Management 87, 132, 135, 145
  - client concept 90
  - processes 96
- Client concept 90
- Code Inspector 129
- Co-hosting 51
- Collaboration 79
- Comparing 181
- Computer Installation and Acceptance 149
- confidentiality 70
- Configuration element 145
- Configuration Management 145

- Continuity Management for IT
  - Services 148
- Conversion Workbench (CWB) 170
  - application areas 172
- Cost driver 118
- Cost factors 119
- Cost optimization 115
- CRM-Server 86
- Cross Applications 18
- CUA 50, 71

## D

- Data 181
- Data conversion 170
- Data quality 13
- Database layer 55
- Deming quality circle 139
- Development class 73
- Development landscape 125
- Development plan 121
  - sub-components 122
- Dialog instance 59
- Dialog service 110
- Digital signatures 72
- Direct costs 119
- Dispatcher 59, 60
- Dispatcher Queue 60
- Dispatching 201
- Downtime 109
- Drag & Relate 24

## E

- eCATT 129
- Electronic Data Interchange 14
- Enqueue service 110
- Enterprise Application Integration 14
- Enterprise Services Architecture 78,  
80, 106
- Environmental Management 149
- Event-driven process chain (EDPC)  
192

## F

- Financial key figures 195
- Financial Management for IT services  
146
- Forecast 200

## G

- Gateway process 59
- Gateway service 110
- Generic process redesign 163

## H

- Health check 138
  - content 138
  - objectives 138
  - process 138
- High availability strategy 110

## I

- IDoc 67, 68
- Implementation plan 126
- Incident Management 145, 157
  - Key Performance Indicators 159
- Indirect costs 119
- Industry solutions 45
- Instance 59
  - central 59
  - dialog 59
- Integration Broker 21
- Integration Directory 29
- Integration Repository 29
- Integration Server 29
- Integrity 70
- Intermediate Documents see IDoc 67
- Internal Accounting Standards 46
- Internationalization 70
- Internet Application Components 58,  
62
- Internet Communication Framework  
64
- Internet Communication Manager 64
- Internet Transaction Server 58
  - X.509-Authentifizierung 71
- IRR 117
- Issue Management 198
- IT Infrastructure Management 148
- ITIL, Best Practices 144
  - IT Infrastructure Management 148
  - service delivery 146
  - service support 144

## J

J2EE Connector Architecture 67  
Java Database Connectivity 66  
Java Personality 23, 64, 79

## K

Knowledge Management 27, 79  
KPI monitoring 198

## L

LDAP 50  
Liability 70  
Lifecycle management 22, 91  
    change management 96  
    design 92  
    implementation 93  
    installation 92  
    operation 94

## M

Management 109  
Management of Local Processors 149  
Master Data Management 21, 79  
Memory Pipes 64  
Message server 58  
Message service 110  
Microsoft.NET Connectivity 67  
Migration 170  
Migration Workbench (MWB) 170, 172  
Mobile Infrastructure 79  
Multiple Components on One  
    Database 52  
Multi-system landscape 87  
mySAP 34, 35  
mySAP All-in-One 17  
mySAP Analytics 35  
mySAP Business Suite 17, 34, 106  
mySAP Corporate Services 35  
mySAP CRM Interaction Center 86  
mySAP Customer Relationship Management 36  
mySAP Enterprise Resource Planning 34  
mySAP Financials 35  
mySAP Human Capital Management 35  
mySAP Marketplace 41  
mySAP Operations 35

mySAP Product Lifecycle Management 42  
mySAP Supplier Relationship Management 40  
mySAP Supply Chain Management 38  
mySAP Technology 78

## N

NetWeaver  
    Application Platform 22  
    Information Integration 21  
    People Integration 21  
    Process Integration 21  
Network Services Management 148  
Next Generation ERP 12  
NFS service 110  
Note Assistant 51

## O

Object type 175  
Open SQL 66  
Operating concept 95  
Operations Management 149  
Organizational Structure Change  
    Consulting 169

## P

Package 73  
Performance 108, 111, 162, 207  
    application optimization 114, 163  
    technical optimization 114, 163  
Performance monitors 163, 165  
Plain HTTP Adapter 187  
Presentation layer 55, 61  
Problem Management 145, 157  
    Key Performance Indicators 159  
Process optimization 107  
Production landscape 122  
Proxy-Generierung 186

## Q

Quality assurance 89  
Quality assurance system 125

## R

Real-time business monitoring 13, 197, 209  
    controlling 202



- data extraction 199
- distribution channel 201
- escalation 202
- forecast 200
- KPIs 203
- logging 202
- visualization 199
- Real-time enterprise 11
- Release Management 145
- Reliability 108, 109
- Remote Procedure Call 67
- Reporting 196
- Repository Switch Upgrade 129
- Reverse Business Engineer (RBE) 163, 181
  - analysis possibilities 181
  - scenarios 184
  - use 181
- Reverse Modeling Engineer 182
- RFC, Remote Function Call 67
- ROI 116

**S**

- Safeguarding Check 210
- SAP APO 84
- SAP Business Connector 187
- SAP Business Information Warehouse 30, 80
- SAP Business Maps 194
- SAP Business Objects 190
- SAP Business One 18
- SAP Business Workflow 107
- SAP CCMS Monitoring Infrastructure 101
- SAP Conversion Service 170
- SAP corrections 97
- SAP Enterprise Portal 18, 24, 79
- SAP Exchange Infrastructure (XI) 29, 79, 163, 185, 210
  - Architecture 81
  - configuration phase 186
  - design phase 185
  - runtime phase 186
- SAP Gateway 58
- SAP Global Support 102
- SAP GUI 61
  - for HTML 61
  - for Java 62
  - for Windows 61
- SAP Internet Pricing and Configurator 84
- SAP Internet Sales 84
- SAP J2EE Engine 22, 84
- SAP logon tickets 71
- SAP Master Data Management 32
- SAP NetWeaver 12, 20, 22, 55, 78, 106, 128
  - architecture 81
- SAP Note Assistant 102, 129
- SAP Notes 102
- SAP One Server 128
- SAP RBE Explorer 182
- SAP Reverse Business Engineer 209
- SAP Security Library 72
- SAP Services 110
- SAP Solution Manager 50, 94, 97, 98, 99, 113, 129, 153, 154, 155, 162, 166, 167
  - for Implementation 100
  - for Operation 101
  - Support Desk 102
- SAP Solution Maps 194
- SAP Support Desk 51
- SAP system architecture 55
  - development plan 121
  - implementation plan 126
  - optimization 120
  - realization strategy 124
- SAP System Landscape Planning 83
- SAP system landscape
  - Change Management and QA 87
  - Internet sales 84
- SAP Web Application Server 22, 23, 55, 64, 79
  - architecture 57
  - connection standards 67
  - connection technology 67
  - layers 65
- SAP Web Dispatcher 69
- SAP xMA 45
- SAP xRPM 44
- SAP-Basis 55, 57
- SAPconnect 86, 180
- SAPforms 179
- SAPphone 86

SAProuter 72, 110  
SAPSECULIB 72  
Sarbanes Oxley Act 195  
Second Level Support 143  
Secure Network Communications 71, 72  
Secure Store & Forward 72  
Security Management 148  
Service & Support 14, 131, 208  
    Change Management 135  
    degree of maturity 139  
    health check 138  
    influencing factors 136  
    optimization 141  
    optimization factors 141  
    optimization project 153  
    procedure 135  
    process optimization 142  
    quality 139  
    quality assurance 141  
    quality care 141  
    Tools for optimization 151  
    Trigger for improvement 134  
Service Delivery 144, 146  
Service Desk 144, 151, 155  
Service Level Agreement 95  
Service Level Management 146  
Service Level Reporting 113, 166  
Service Management systems 151  
Service Support 144, 151  
Shared Collaboration Knowledge 186  
Single point of Failure 109  
Sizing 92  
Software Lifecycle Support 150  
Solution Lifecycle Management 80  
Solution Management 97, 99, 112  
Solution Management Optimization 210  
Solution Monitoring 112, 166  
Spool service 110  
SSL 71  
Support Desk 102  
Support organisation 143  
Support package 97  
Support tools 154  
System availability 108  
System Landscape Change Consulting 169

System Landscape Directory (SLD) 186  
System Landscape Optimization (SLO) 115, 162, 167, 208  
System Monitoring 101, 111

## T

TCO 117, 127  
    Service Map 127  
Testing an IT Service for Operational Use 150  
Threads 59, 64  
Total Cost of Ownership 105  
Transaction Identifier 67  
Transport layer 75  
Transport Management System 75, 77  
Transport Organizer 77  
Transport request 75  
Transport request types 76  
Transport route editor 75  
Transport routes 75  
TREN 21, 84, 87

## U

Unicode 70  
Unification 26  
Update service 110  
Upgrade 92, 96, 129

## W

Web Dynpro 62, 65  
Work process 58  
    dialog 58, 59  
Workflow  
    application area 173  
    definition 176  
    functionality 173  
    Integration of non-SAP systems 179  
    Inter workflow communication 179  
    project 180  
    SOAP 179  
    technical foundations 175  
    Wf-XML 179  
Workflow Builder 176  
Workflow/WebFlow 107, 209

## X

X.509 client certificate 71  
xApps 18, 44