

ACADEMIA DE STUDII ECONOMICE - București
Acadmy Of Economic Studies - Bucharest

FACULTY OF BUSINESS ADMINISTRATION
(Facultatea de Administrare a Afacerilor cu predare în limbi străine)

Informatics for Business Administration

-

Architectures of Business Administration
Applications - SAP

By: Professor Vasile AVRAM, PhD
- suport de curs destinat studenților de la secția engleză - master aprofundat-
(lecture notes for 1st year students of english division – Master in Business Administration)
- anul I - Zi -

București 2008

Chapter 2: Architectures of Business Administration Applications – SAP	3
2.1 Applications categories.....	3
2.2 SAP three-tiered architecture.....	3
2.3 Enterprise SOA.....	4
2.3.1 SAP NetWeaver platform	4
2.3.2 SAP and SOA	7
2.3.3 Web Services and Enterprise Services	8
2.3.4 Enterprise SOA.....	8
2.4 SAP Project basics.....	9
2.4.1 SAP project management and implementation methodology	9
2.4.2 SAP ASAP Focus - an SAP methodology for SMEs	12
2.4.3 Global ASAP – an SAP methodology for global enterprises	13

Bibliography

Books

1. SAP NetWeaver For Dummies, by Dan Woods and Jeffrey Word, John Wiley & Sons © 2004 (414 pages)
2. Upgrading SAP, by M. Sens, Infinity Science Press © 2008 (344 pages), ISBN:9781934015155
3. Sams Teach Yourself SAP in 24 Hours, Third Edition by George W. Anderson; Tim Rhodes; Jeff Davis; John Dobbins; Andreas Jenzer, Sams, August 06, 2008
4. SAP System Landscape Optimization, Andreas Schneider-Neureither [Ed.], SAP Press, <http://www.sap.com>

SAP Web Resources

1. www.sap.com
2. www.sdn.sap.com
3. www.service.sap.com
4. www.sapfans.com
5. www.asug.com
6. www.sapgenie.com
7. www.solutionbrowser.erp.sap.fmpmedia.com
8. www.sap-press.com
9. www.microsoft-sap.com
10. www.duet.com

Article I. www.oracle.com/sap

Chapter 2: Architectures of Business Administration Applications – SAP

2.1 Applications categories

We can broadly categorize, depending on deserved level, applications as:

- vertical applications, that are designed for specific industry such as banking, insurance real estate, retail, transportation, health care, government, defense etc;
- horizontal applications, that are used by all companies such as human resources, payroll, accounting, CRM, manufacturing etc.

Depending on the purpose application software can be categorized as:

- **general purpose** – is used daily by any number of business man and women (for example, word processing is maybe the most used software). The programs in that category can be used by anyone from any company or industry to handle a variety of functions. Word processors, spreadsheets, personal databases, graphics, desktop publishing all fits in that category. A characteristic of this software is that is user-friendly and designed for non-technical people;
- **specific purpose** – is for corporate use and is written or purchased and modified by computer professionals to fit the corporation's needs and business practices. Corporate application systems are referred to as either back-office or front-office.

2.2 SAP three-tiered architecture

SAP R/3 introduces a technical architecture as a three-tiered architecture (this gives the name R/3) that subdivides SAP into three architectural layers (figure 2.1) based on function:

I. [**Presentation**] User Interface Layer – front-end client or presentation layer – where front-end clients access devices such as individual PCs, laptops, workstations, mobile devices, and so on, that run a piece of software used to “talk” to the SAP applications back in the company's data center. The front-end clients are called presentation servers because they present the SAP-specific user interface to the clients. SAP specific interface can be displayed by special installation of SAP GUI (running atop the operating system GUI) or without special installation inside a web browser (SAP GUI for HTML). The SAP GUI for Microsoft Windows (called “WinGUI”) is the most popular but exist also a Java GUI for SAP for non-Microsoft-based front-end clients (both GUI interface applications are “fat client” requiring a good amount of disc space and memory to run well). The alternative, the access from inside a browser application (for example, Internet Explorer), requires less power, little disk space, and avoid the maintenance of the applications displaying the interface. The interfaces connect to the SAP application (tier II), which in turn talk to the back-end database (tier III). All SAP users have associated a user name (a user ID) and password used to establish a connection after the start of GUI and to begin a user

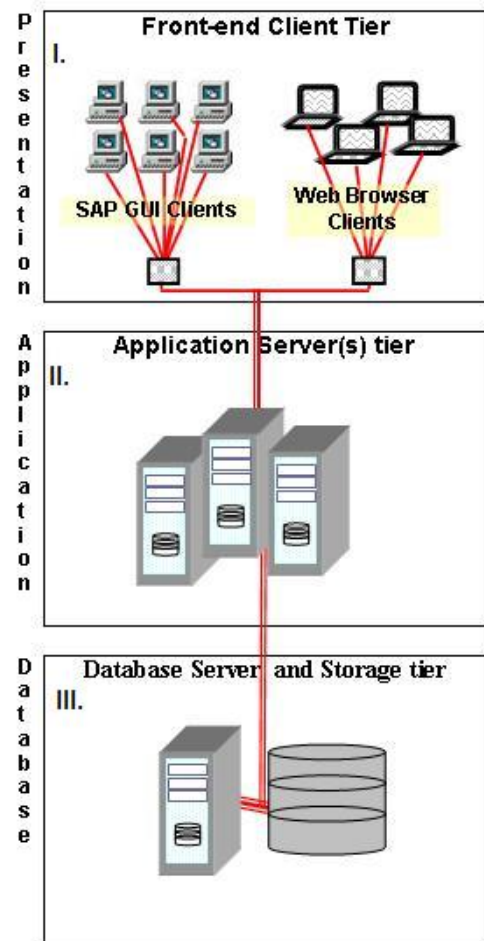


Figure 2.1 The SAP classic three-tiered architecture requires a database server, one or more application servers, and any number of front-end clients

session. It is allowed for any user to open multiple sessions (by default up to six) each of them with a particular SAP system such as SAP ERP, CRM, SCM etc (and in that way to multitask). In the world of SAP clients are essentially self-contained business entities or units within each SAP system and by using the SAP's user interfaces can log in to a client to actually access and use the system.

Web Dynpro technology allows programming the user interface by providing a development and runtime environment for Web Applications, based on server side scripting.

II. **[Application]** Business Logic Layer - the "application" tier upon which SAP's programs actually execute.

III. **[Data]** Database Layer - the "back-end" RDBMS that houses all the SAP data, such as customer records, inventory status, financial information, and so on. The applications are connected to database servers where these records are maintained.

With three-levels configuration a separate server required for each service (can be one or more). Both application and data can be distributed on several machines.

The benefits of such architecture are represented by the possibility to change, when applications starts to run slowly, the structure of presentation layer (by adding new CPUs and memories to presentation servers) or to upgrade the DBMSs or add new storage without affecting the others layers.

2.3 Enterprise SOA

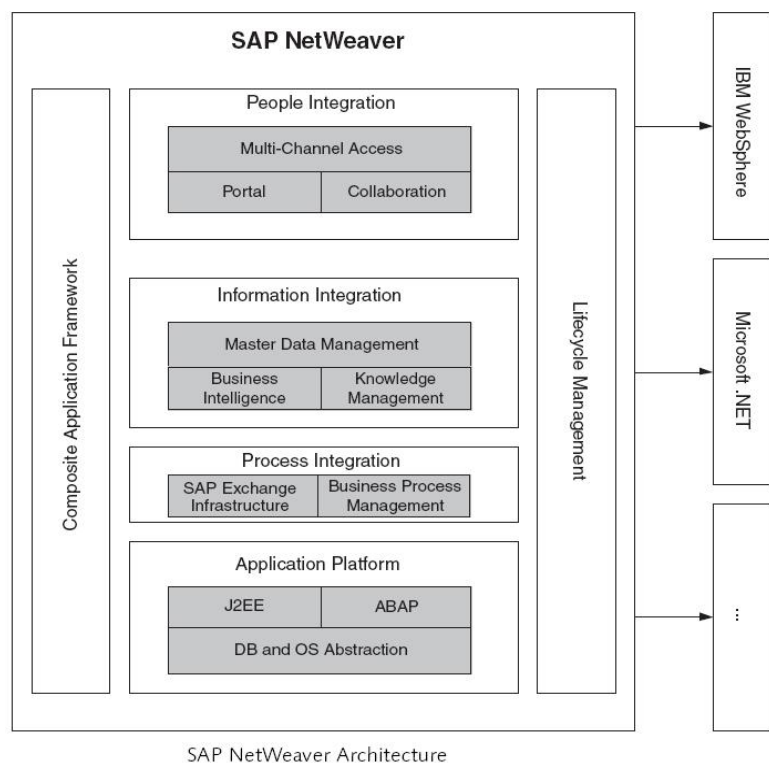
2.3.1 SAP NetWeaver platform

SAP NetWeaver (figure 2.2) provides the technology foundation for the entire SAP suite of products. It is a platform allows aligning IT with business requirements by combining composition technologies and application functionality to reduce IT complexity and increase business flexibility.

SAP NetWeaver allows composing applications using enterprise services, orchestrate business processes and events, manage enterprise information, and deliver applications and content to users more quickly and cost-effectively. SAP NetWeaver allows a scalable development and implementation by implementing crucial IT practices with SAP NetWeaver, addressing in that way the immediate IT needs, then expand the practice projects over time within a sustainable cost structure (is the successor of mySAP).

The SAP NetWeaver platform enables (<http://www.sap.com>):

- **Data unification** – Ensure that your master data is



SAP NetWeaver Architecture

Figure 2.2 SAP NetWeaver™ (<http://help.sap.com> SAP System Landscape Optimization, Andreas Schneider-Neureither [Ed.], SAP Press)

- accurate, free of duplicated records, and normalized;
- **Process integration** – Make disparate applications and business partners' systems work together;
- **Business information management** – Increase the visibility and reach of structured and unstructured enterprise data.

People Integration: SAP Enterprise Portal, Collaboration, SAP Mobile Infrastructure

Information Integration: Business Intelligence, Knowledge Management, SAP Master Data Management

Process Integration: SAP Exchange Infrastructure, Business Process Management

Application Platform: ABAP Personality, Java Personality, Business Services and Features, Connectivity, User Interface, Database and operating-system abstraction

NetWeaver provide benefits on several fronts:

- Decreased development costs.
- Easily enabled integration, thus speeding up the time necessary to deploy SAP as well as shrinking the time necessary to perform system upgrades and so on.
- Reduced total cost of ownership, primarily because maintenance and support costs are dramatically reduced in the wake of technical platform standardization (before NetWeaver, each SAP product was installed a little bit differently than its counterparts).
- Greater potential for innovation. Through NetWeaver, a company's IT organization needs to spend less time maintaining existing solutions and thus can spend more time meeting the changing needs of the business.

Basis layer. The SAP technology stack was simply referred to as the Basis layer. Today describes administrators and technical consultants that architect, support and monitor SAP systems and includes common system administration tasks such as performance monitoring, tuning, security, internal and external systems connectivity, faxing capabilities, email broadcasting, and more. It also includes the development repository of SAP's ABAP programming language and database objects. In this way, the Basis layer provided the necessary development, communication, and monitoring infrastructure necessary to meet the needs of the client/server era. In 90s the Basis extended to allow applications to adapt to the Web by the SAP Web Application Server (Web AS). SAP Web AS allows also integrate the SAP product SAP Internet Transactions Server (ITS) into technology core and later on added SAP Java as strategically platform independent model for web development. SAP Web AS provides an environment in which all applications, in addition to the underlying operating system, can be developed, managed and executed. Now SAP Web AS offers a platform for efficient development and for using Web applications and is a component of all mySAP installations. SAP Web AS is platform-independent and ensures a high degree of scalability of an SAP installation.

The components of SAP WebAS (figure 2.3 and 2.4) and their tasks are described below:

- Internet Communication Manager (ICM) sets up the connection to the Internet, process both server and client Web requests (can behave as client and/or as server), and supports protocols such as HTTP, HTTPS, SOAP, and SMTP. Output is in standard format such as HTML, XML, or XSLT. It uses threads to communicate via Internet. If the incoming HTTP requests are processed by a work process data exchange occurs using what is known as Memory Pipes in the shared memory.
- Dispatcher distributes the requests to the work processes if these free or store them in the dispatcher queue if processes occupied. 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.

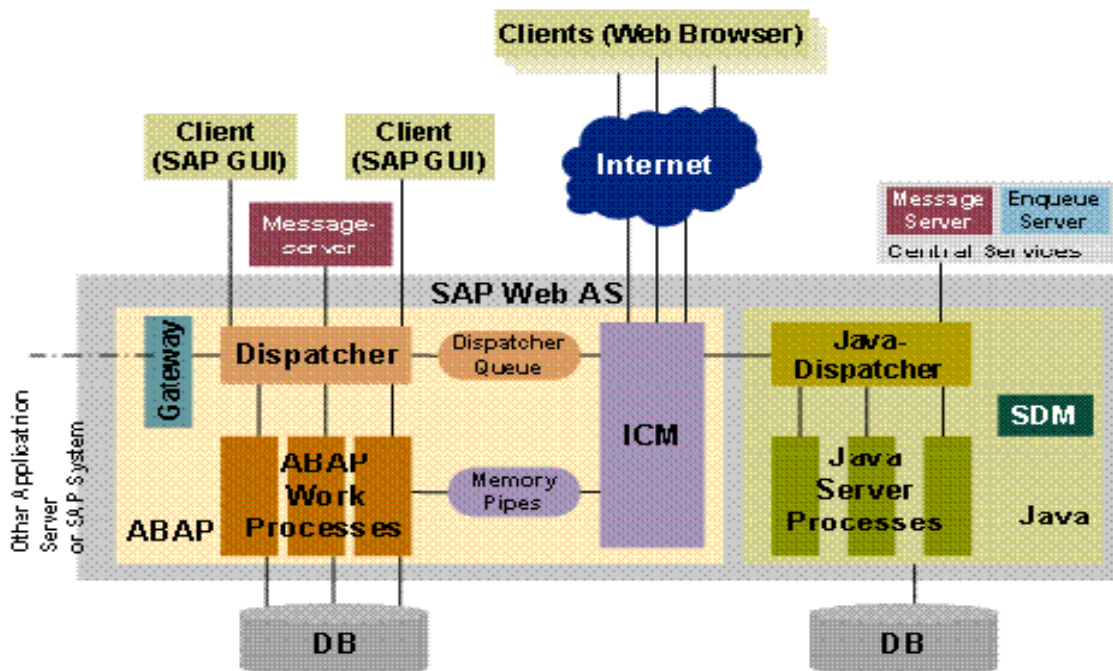


Figure 2.3 Components of Web Application Server (<http://help.sap.com>)

- The ABAP workbench is the application object development environment for SAP Web AS. All development objects in the ABAP Workbench are stored in the central repository and are client-independent. ABAP Work Process executes the ABAP code.
- SAP Gateway makes the RFC interface between the SAP instances available (within an SAP System and beyond system boundaries; systems like R/3, R/2, external systems and so on).
- Message Server exchanges messages and balances the load in the SAP System (communication between dispatchers of a SAP system).
- Java component of the SAP Web AS composed by Java Dispatcher, Server Process, and Software Deployment Manager.
- The database layer of SAP Web As ensures optimal access to data via ABAP environment (Open SQL). It ensure database independence via JDC (Java Database Connectivity) and by transferring the functions for Open SQL for ABAP to an Open SQL for Java and a range of APIs.

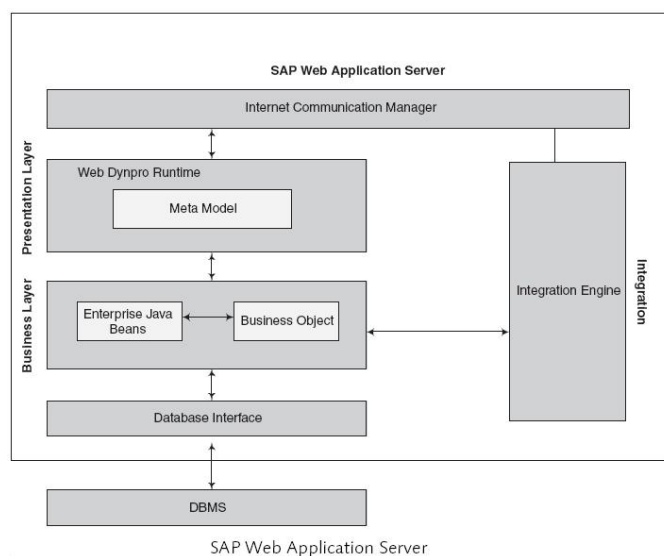


Figure 2.4 SAP Web Application Server

(<http://help.sap.com> SAP System Landscape Optimization, Andreas Schneider-Neureither [Ed.], SAP Press)

The services that run on a server are merged and managed together by an administrative unit, the *instance* (figure 2.5). Depending on the configuration the instances can carry out different tasks. In large SAP systems the application layer can be distributed over several instances improving in that way the performances. Each of the instances offers *buffer* areas for different objects (programs, dictionary objects, screen structures, or tables contents).

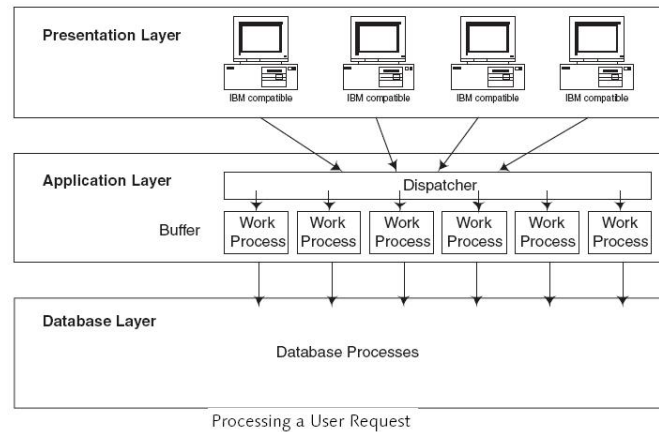


Figure 2.5 Instance and buffer (<http://help.sap.com> SAP System Landscape Optimization, Andreas Schneider-Neureither [Ed.], SAP Press)

2.3.2 SAP and SOA

Service-Oriented Architecture (SOA) is nothing more than an approach to designing a more innovative computing platform that takes advantage of reusable services to build powerful business processes (figure 2.6). SOA provides for definitions and methods of building an IT infrastructure that makes it possible not only to exchange data between different systems and other data repositories, but to build and extend business processes. As such, SOA is necessarily tied to both the underlying operating systems as well as the development/programming languages native to an application. But the link is a loosely coupled one; in this way, it is possible to support different operating environments spanning traditional UNIX offerings, Linux, Windows, and even mainframes.

To easily "extend" SAP's business processes to other business applications or sources of data, SAP has ensured its latest round of applications support the industry architecture Service-Oriented Architecture (SOA). SOA provides a more robust blueprint for designing an adaptable enterprise computing solution. Though it requires a "hub" of SAP's latest and greatest NetWeaver stack, an SOA-compliant architecture can support older SAP business applications such as R/3 and everything in between.

The idea is that applications are treated and thus connected to one another similar to how you might personally access an everyday service or utility. At the end of the day, if an application can share data via industry-standard XML (eXtensible Markup

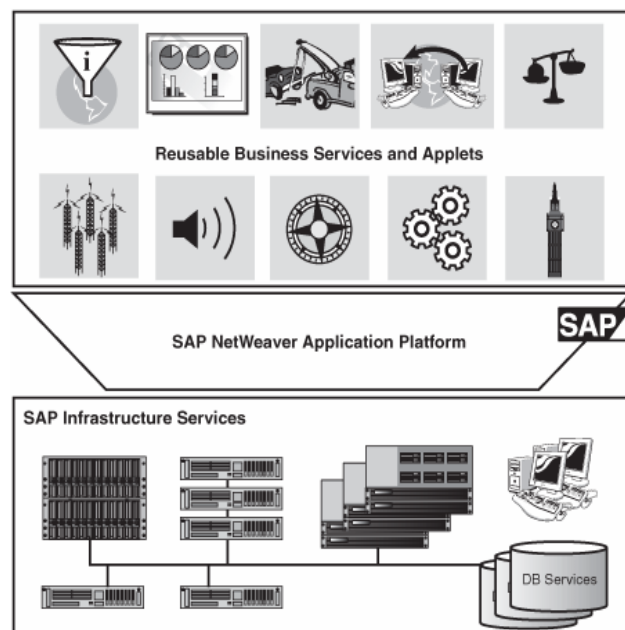


Figure 2.6 SOA allows SAP to be easily extended to take advantage of readily available service-based applications and utilities (Source: Sams Teach Yourself SAP in 24 Hours, Third edition by George W. Anderson, Tim Rhodes, Jeff Davis, John Dobbins, Andreas , August 06 2008)

Language, a standard "open" method of structuring and sharing data between different computers or systems via the Internet among other mechanisms), it can be connected to SAP and thus easily used and accessed like a shared reusable service. Such services might include pricing engines or web applets that manage orders or customer account information. The idea is for a company to eventually publish a list of all services available to its business (an applications services portfolio, if you will), making it even easier to change and flexibly grow the company's enterprise computing platform based on changing needs. SAP NetWeaver merely provides the vehicle for connecting such application services; it provides the platform.

2.3.3 Web Services and Enterprise Services

As NetWeaver evolves in response to a changing world, SAP has bet the bank on a number of technologies and approaches intended to minimize integration headaches, total cost of ownership, and so on. These include Web Services and Enterprise Services.

Though not synonymous, Web Services represents the vehicle that makes Enterprise Services possible. They are beginning to become the new standard for inter-application communication. The idea is that Web Services are "open" rather than tied to a particular type of technology or hardware platform or software vendor. This platform independence, therefore, makes it possible to communicate between vastly different technological platforms. This makes Web Services a valuable attribute of any NetWeaver solution—particularly those that wish to get a jump start on new SOA-enabled solutions.

Enterprise Services, on the other hand, extend the use of Web Services by combining them into a usable context as they relate to a business process. In this way, they become reusable components that may be shared across partners and organizations to foster real innovation. SAP takes this a step further by packaging these Enterprise Services and making them available in the new Enterprise Services Repository (included in SAP NetWeaver 7.1) and the Enterprise Services Workplace out on SDN (the SAP Developer Network, accessible to registered developers and others at <https://www.sdn.sap.com>).

2.3.4 Enterprise SOA

Enterprise SOA is a blueprint for an adaptable, flexible, and open IT architecture for developing services-based, enterprise-scale business solutions. With SAP NetWeaver as a technical foundation, enterprise SOA moves IT architectures to higher levels of adaptability – and moves companies closer to the vision of real-time enterprises by elevating Web services to an enterprise level.

An enterprise service is typically a series of Web services combined with business logic that can be accessed and used repeatedly to support a particular business process. Aggregating Web services into business-level enterprise services provides a more meaningful foundation for the task of automating enterprise-scale business scenarios.

Enterprise SOA enables composite applications to be built by assembling enterprise services (similar concept to Web Services, though more generic in nature), while SAP NetWeaver's Enterprise Services Repository (ESR) serves as the central building block for creating SAP Enterprise SOA applications.

SAP describes five principles of Enterprise SOA:

- **Abstraction** - Serves to mask unnecessary or otherwise puzzling details.
- **Modularity** - An essential property of enterprise services that enables the development of reusable components or building blocks by breaking down services into fundamental units.
- **Standardized connectivity** - Necessary to describe and enable data sharing and triggering, which in turn can be used to build and combine flexible services into full-fledged enterprise business processes and business solutions or scenarios.

- **Loose coupling** - Another necessary property that enables individual services to grow and evolve without requiring a rewrite. Loose coupling preserves reusability as well as integration and connectivity between services.
- **Incremental design** - The ability to enable changes to a service's composition and configuration without requiring a rebuild.

Enterprise SOA-enabled SAP applications built from a mix of reusable and extensible services deliver the flexibility, agility, and uniformity required by enterprises seeking to spend less time maintaining code and more time innovating. Building SAP applications from a single suite of enterprise services makes this not only achievable but less costly than preceding client/server and monolithic architectural approaches to deploying enterprise applications.

Underlying and making possible these enterprise services is metadata that describes each enterprise service's characteristics and the data that comprises and passes between them. XML is typically used extensively to create and package the data, which in turn is wrapped and then described by WSDL and communicated about by the SOAP protocol. To be sure, these standards are open and subject to change.

Here are a few things to remember about SOA and Web Services:

- Services are standalone; by definition, a service is not required to know how an application functions.
- In the same way, applications are not required to know how a particular service operates; they are completely independent from one another.
- The interface definition ascribed to a particular service hides its unique programming language and logic details; again, an application does not need to know anything about the service's internal operations.

2.4 SAP Project basics

2.4.1 SAP project management and implementation methodology

All SAP projects are designed, realized, deployed, and implemented by using the SAP project management and implementation methodology called "AcceleratedSAP(ASAP)". This methodology includes 5 phases (figure 2.7), as follows:

1. **Project Preparation.** This is the first phase of ASAP, which is used to collect all resources required during the project: staff, equipment, budget, procedures, documentation, and overall project planning. In this phase, you lay the groundwork: You organize your project team; define project standards; document your project plan in detail; solidify the project scope; resolve issues; and define processes. Now's the time for you to identify and prepare everything that may be needed along the way for your project.
2. **Business Blueprint.** This second phase is used to analyze the SAP system and prepare it for the next phases. All business processes are analyzed in such a way that they can be prepared for the new functionality. Therefore, both the functionality of the SAP target release and current company operation way are analyzed. In this phase, you define your solution in minute detail. Your team digs about, gathering information on the business processes that project must integrate, and then your eager team members determine what the final product will look like. Along the way, you can use questionnaires to determine how many roles the team needs, what reports have to be configured, what customization you need to do, and much more.
3. **Realization.** Time for the blueprint created in phase two to start becoming a reality. In this phase, your team custom-codes the basic software to meet your company's specific needs.

Also, you execute conversions and create ports. This phase is a lot of work: Realization is typically the longest phase of the ASAP methodology. This phase is used to physically realize the entire SAP system landscape. It includes all technical work required to realize all software products, including SAPGUI on PCs.

4. **Final Preparation.** In this phase, you ensure that your new system executes properly and that your team has fully tuned its performance. You also stress-test your system to make sure that all applications are fully integrated. If stress-testing reveals any problems, now's the time to fix them. This is also when you make sure you train your end users. Finally, handle all change-management activities now, including establishing a support desk.
5. **Go-Live and Support.** This phase includes all steps that are involved to run and maintain the SAP system during productive use.

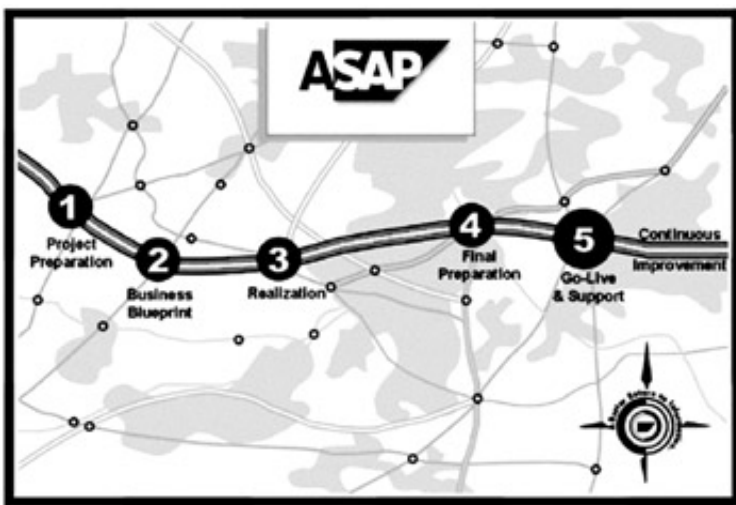


Figure 2.7 Accelerated SAP phases. ©Copyright SAP AG.
(Source: Upgrading SAP, by M. Sens, Infinity Science Press © 2008)

A new SAP implementation project reflects several common themes including:

- business requirements;
- SAP application functionality;
- underlying or enabling technologies;
- implementation project management concepts.

All of these can be viewed in terms of road maps: for every new SAP project, key decisions and factors are related to how business, functionality, technology, and overall project management requirements are weaved together and addressed. In simplest terms, a business road map is the path or process by which an organization's vision and business requirements are translated into a set of fundamental business processes, which in turn are married to SAP (and other) applications, products, and components. A business roadmap provides the highest level of abstraction surrounding the implementation effort. In that view the technologies, and the applications atop those technologies, are simply tools used to navigate and fulfill the roadmap's purpose.

From an SAP project perspective must be addressed four primary "views":

- Business view (*why?*) - to completely describe the system's purpose;
- Functional (application) view (*what?*) - to completely describe the system's functions;
- Technical view (*how?*) - to completely describe the system's technical underpinnings;
- Implementation (project management and oversight) view (*with what?*) - to completely describe the system's implementation details.

Each view requires attention in terms of vision, stakeholder buy-in, staffing and other resource requirements, time lines, and much more.

Once exists an SAP infrastructure platform any SAP business process must follow the phases of designing, configuring, and testing before deploying atop of that infrastructure. After deploying has been completed, an SAP end user must be trained so that finally is able to actually use the system in his daily routine (for example, moving and tracking inventory, reporting on financials, supporting the sales team by helping to manage order status, and so on).

(i) The Business View

The first critical step in project realization is developing a sound business view that articulates *why* a particular problem needs to be solved or *why* an opportunity needs to be explored. Developing a firm's unique business view mandates addressing the following:

- The identification of business-relevant stakeholders;
- Long-term strategy enablement;
- Short-term business objectives;
- Core competencies;
- Competencies other than the core competencies (and thus opportunities for partnering, developing alliances, or contracting out specific services);
- Procurement and other sourcing strategies (and how those strategies and relationships might change over time);
- SAP globalization and localization realities (where the emphasis tends to shift back and forth between enabling global consistency and roll-up financial reporting, for example, along with addressing the currency and language requirements for local user communities).

The most relevant stakeholders at this level are those involved in strategically aligning and executing the actual business processes (sometimes called work-streams) necessary for survival. Developing and communicating the business view requires the involvement in the project team of business executives and other officers of the board, along with functional managers and team leads, business analysts, and any other line-of-business leaders if necessary.

(ii) Functional View

The functional view is the easiest to grasp for individuals intimately familiar with how to run a business. This view addresses the *what* surrounding a solution. Not how, or when, or with what, but simply what. What will a particular business process do? In this way, the functional view does the following:

- It describes or communicates the flow of work (workflow) in a stepwise fashion. (What steps are necessary to execute a business process and thereby achieve a particular end state?)
- It describes the properties or qualities to be exhibited by the business process. (What characteristics or properties should the business process encompass, and to what degree?)
- It addresses the preceding from a technology-independent and SAP-independent perspective.

As you can imagine, the key stakeholders for such a view are the end users who will ultimately execute the business process as part of their daily work. Business process designers, line-of-business leaders, and others involved with the functionality to be embodied by the solution are important stakeholders, too.

(iii) Technical View

The technical view addresses the *how* part of the solution equation. It gives legs to the functional view in that the technical view describes how the business solution will work. Important considerations include the following:

- Focusing on the key dimensions of the system; how the system will deliver the performance, availability, scalability, security, agility, systems manageability, and so on required by the business;
- Describing the solution's overall components in terms of business applications and other SAP components, data and relevant dependencies, interface requirements, underlying technical infrastructure, and all the inter-component relationships and integration points necessary to enable the previously described functional view;
- Providing to the extent possible a technology-independent perspective as to how all this is accomplished.

Primary technical view stakeholders include solution developers and programmers, infrastructure and other technology specialists, and other technology-focused suppliers, vendors, and partners.

(iv) Project Implementation View

The project implementation view is probably the simplest to comprehend. It answers the question *with what* shall the solution be built, and over what time period leveraging what resources? The project implementation view, or more simply the implementation view, accomplishes the following:

- Describes and details the deployment plan, which in turn encompasses organizational and third-party resources, timelines, constraints (business, functional, technical, and other), and so on;
- Describes the SAP products and components to be used to fulfill the functional vision/view, and how they will be configured.

Common implementation view stakeholders are project managers and coordinators, technical specialists, developers/programmers, testers, business process owners, executives, business leaders, power users, and more.

2.4.2 SAP ASAP Focus - an SAP methodology for SMEs

Every company, no matter its size, needs efficient, streamlined processes, insightful business intelligence, and systems that are flexible enough to grow with changing business needs. But the solutions that deliver these benefits can put a strain on the budgets of even large companies. For businesses with limited time, IT personnel, and finances, the power of comprehensive enterprise solutions has seemed out of reach.

A traditional approach to IT implementation with exhaustive as-is and to-be studies, an in-house project team, and generous timeline might be the norm for large multinational companies. But midsize enterprises typically need to move more quickly on a tighter budget while minimizing disruption of ongoing operations. They need an approach that accelerates the return on their investment while reducing costs, implementation time, and risk.

To address those needs, SAP ASAP Focus is specifically designed for a safe, predictable, and affordable implementation of solutions based on SAP Best Practices offerings, such as qualified mySAP™ All-in-One partner solutions and packaged solutions. The methodology thus ensures lower project risk and reduced time to value. An SAP solution implemented with the SAP ASAP Focus methodology combines the following advantages:

- A road map designed to implement a predefined business solution in the shortest possible time;
- Complete predefined and pretested processes for specific business activities, supported with ready data conversion tools that allow the project team to validate the solution with your data right from the start;
- Ready-made documentation covering business processes, and configuration that allows the project team to focus on getting the organization prepared;
- Predefined reports, print forms, authorization roles, and test scenarios that ensure a cost-effective implementation;

- An approach that allows you to accomplish certain tasks before the SAP project team arrives on-site, further reducing your costs.

2.4.3 Global ASAP – an SAP methodology for global enterprises

Global enterprises faced to maintain consistency across the IT infrastructure (since disparate systems with intermediaries for information exchange can bring slower agility). Net-centric operations offering worldwide standardization of business processes, comprehensive business intelligence, and enhanced flexibility are mainstays for successful, multinational organizations. Through a combination of advanced implementation tools and SAP expertise the Global ASAP methodology can be used to plan, design, deploy, and implement SAP solutions for global enterprises. "Global ASAP is a systematic, process-driven methodology geared specifically for global, multisite SAP implementation projects. Global ASAP is comprised of a set of tools, road maps, documentation, and activities designed to help you implement SAP® solutions in your global enterprise."¹

What sets Global ASAP apart from single-site implementation methodologies are the following key elements:

- A global template (figure 2.8) and global infrastructure to support and jump-start all your local implementation projects;

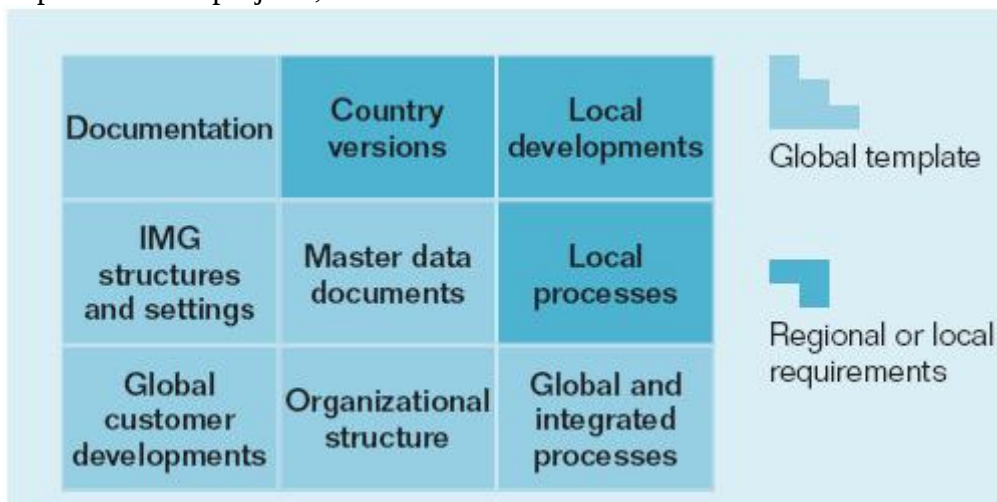


Figure 2.8 Global Template with Local Requirements¹

- A global rollout strategy (figure 2.9) that forms the basis of the entire implementation program
- Incorporation of local business requirements into the global template for rollout and go-live at every site.

The methodology combines the technical and application work around an SAP implementation with other work streams such as program and project management, organizational change, risk management, training, and benefits realization. Global ASAP ensures that all required aspects of an SAP project come together in a cohesive planning framework to meet your business requirements.

Global ASAP allows attaining the following objectives:

- Align business processes across international borders;
- Tackle complex integration and standardization issues;
- Account for unique local processes during implementation rollouts;
- Increase the value, benefits, and productivity of your SAP solutions;
- Implement SAP solutions across multiple countries and multiple sites;
- Enable change management across all implementation sites.

¹ MULTISITE IMPLEMENTATION OF SAP® SOLUTIONS FOR GLOBAL ENTERPRISES, Multisite_Implementation_of_SAP_Solutions_for_Global_Enterprises_.pdf, www.sap.com

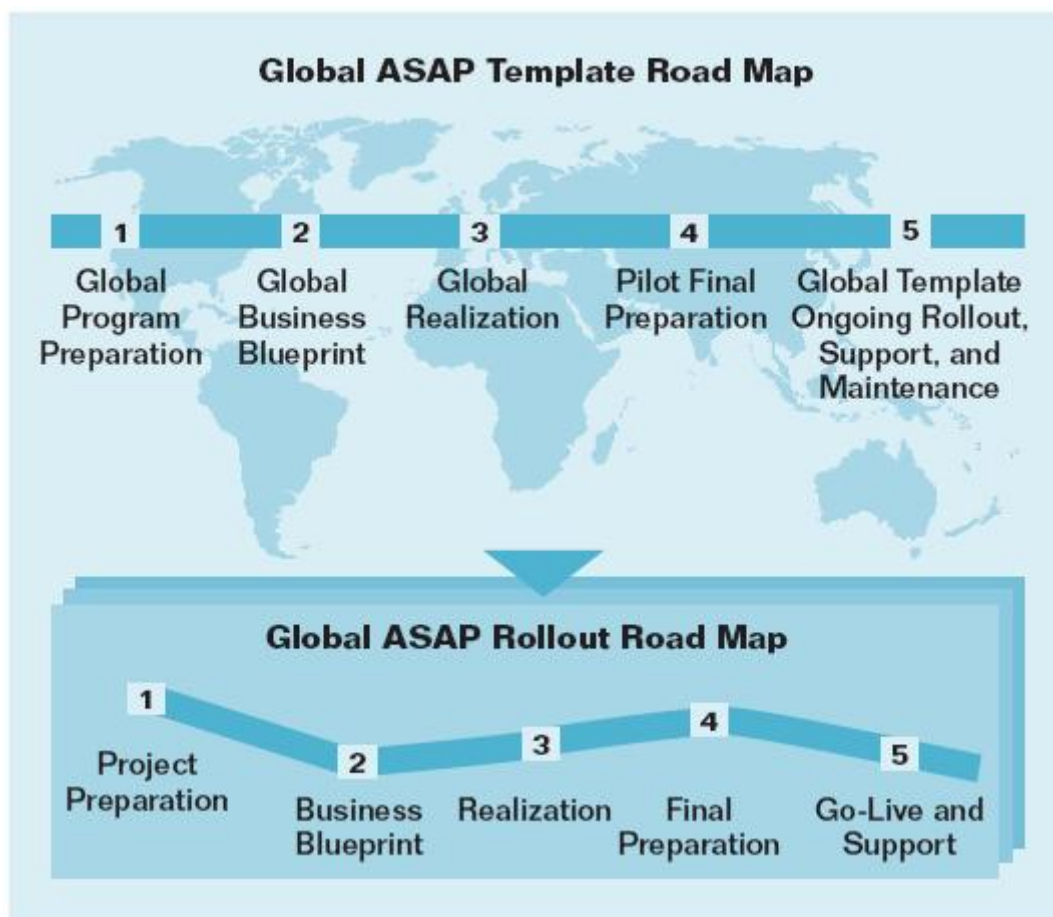


Figure 2.9 Global ASAP Methodology and Global Template

The methodology phases

Phase 1: Global Program Preparation

Every successful implementation begins with planning and preparation. During this phase, you define project goals, establish your high-level scope, and develop your program and project plans. This in-depth strategy-definition process helps you examine all the key aspects you need to consider before starting your implementation. You assess your degree of readiness, current business processes, risks, opportunities, and more. As this process unfolds, you have a valuable opportunity to align business processes across international borders in a way that can yield significant cost reductions and enterprise-wide benefits. During this first phase, you initiate and plan your template project and rollout program.

Phase 2: Global Business Blueprint

Once you have defined your global rollout strategy, your team of SAP implementation experts creates a global business blueprint containing details of all your implementation requirements – including software configuration, reporting, master data, interfaces, and conversions. This blueprint forms the basis of the global template. At this time, your SAP project team begins blueprinting your pilot site–specific local requirements, processes, and development activities. The pilot site is the first implementation site and forms the basis for future rollouts.

Phase 3: Global Realization

During the global realization phase, the implementation team uses the business blueprint to develop the global template. This enterprise-wide implementation template consists of a rollout package and a solution package, both geared toward improving the efficiency and consistency of your multisite deployment. The rollout package consists of all your project guidelines, standards, and procedures.

The solution package consists of a predefined set of processes, called scripts, which support and streamline software configuration, development activities, and data migration routines.

Phase 4: Pilot Final Preparation

During phase four, you finalize cutover preparation and bring the pilot site live. By going live with the global template at a pilot site, you gain valuable experience that is useful as you move on to the next rollout sites.

Phase 5: Ongoing Rollout, Support, and Maintenance of the Global Template

Phase five of the Global ASAP methodology guides you through fine-tuning the global template before conducting subsequent rollouts. This phase emphasizes the support of regional or local units. At the local level, the project team uses the global template as the basis for implementation; however, it is supplemented by local, site-specific requirements identified during the blueprinting phase. Subsequent rollouts are guided by the rollout road map that is populated during template development with specific processes, templates, tools, and content. The rollout road map facilitates quick, easy, and repeatable deployment of the global template at the local level.

The experiences gained during rollouts are incorporated into the global template and made available for future deployments of SAP software. This ensures that your template is up-to-date and ready to use for software upgrades, implementation of additional applications, or rollouts to new sites. The template is updated continuously for optimal performance to keep your business processes running smoothly.