



How To... Design a SAP NetWeaver - Based System Landscape

Applicable Releases:

SAP NetWeaver 7.0

IT Practice:

Unified Life-Cycle Management

IT Scenario:

Software Life-Cycle Management

Version 1.0

March 2008

© Copyright 2008 SAP AG. All rights reserved.

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

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP NetWeaver "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Document History

| Document Version | Description |
|------------------|--------------------------------------|
| 1.00 | First official release of this guide |

Typographic Conventions

| Type Style | Description |
|-------------------------|--|
| <i>Example Text</i> | Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation |
| Example text | Emphasized words or phrases in body text, graphic titles, and table titles |
| Example text | File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools. |
| Example text | User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation. |
| < Example text > | Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system. |
| EXAMPLE TEXT | Keys on the keyboard, for example, F2 or ENTER. |

Icons





| Icon | Description |
|---|-----------------------|
|  | Caution |
|  | Note or Important |
|  | Example |
|  | Recommendation or Tip |

Table of Contents

| | | |
|-----------|---|-----------|
| 1. | Business Scenario | 1 |
| 2. | Background Information | 1 |
| 3. | SAP NETWEAVER BASED SYSTEM LANDSCAPE DEPLOYMENT OPTIONS | 1 |
| 3.1 | Overview of the Roles of Systems within the 'Three System Landscape' | 3 |
| 3.1.1 | Development System (DEV) | 3 |
| 3.1.2 | Quality Assurance System (QAS)..... | 3 |
| 3.1.3 | Production System (PRD)..... | 3 |
| 3.1.4 | Background and definitions..... | 3 |
| 3.2 | Installation options available for a single system | 5 |
| 3.2.1 | Considerations with the traditional installation options available for a single system..... | 6 |
| 3.3 | Landscape options for SAP NetWeaver..... | 7 |
| 3.3.1 | 'Server virtualization'..... | 7 |
| 3.3.2 | Multiple Components in One Database (MCOB)..... | 8 |
| 3.3.2.1 | Availability | 10 |
| 3.3.2.2 | Sizing | 10 |
| 3.3.2.3 | Benefits of MCOB | 10 |
| 3.3.2.4 | Administrative considerations for MCOB system installations | 11 |
| 3.3.3 | Bundling of Usage Types..... | 11 |
| 3.3.3.1 | Availability | 12 |
| 3.3.3.2 | Benefits of bundling usage types..... | 12 |
| 3.3.3.3 | Sizing | 13 |
| 3.3.3.4 | Administrative consideration for bundled installations..... | 13 |
| 3.3.3.5 | Restrictions | 14 |
| 4. | DEFINING AN SAP NETWEAVER SYSTEM LANDSCAPE | 15 |
| 4.1 | Business process requirements | 15 |
| 4.2 | Installable Units | 15 |
| 4.2.1 | Building Blocks..... | 15 |
| 4.2.1.1 | Systems with usage types | 16 |
| 4.2.1.2 | Standalone Engines..... | 17 |
| 4.2.1.3 | Clients | 17 |
| 4.2.2 | Administration Systems | 17 |
| 4.3 | Usage Type Characteristics | 18 |
| 4.3.1 | Planning an SAP NetWeaver System Landscape..... | 18 |
| 4.3.1.1 | Sharing Usage Types | 18 |

| | | |
|---------|---|----|
| 4.3.1.2 | Decision Factors for Landscape Design | 19 |
| | Administration and Operations | 19 |
| | Business Continuity | 21 |
| | Business Requirements..... | 22 |
| | Security, Governance and Compliance | 23 |
| | Software Life-Cycle Management | 24 |
| | Interoperability | 25 |
| 4.3.1.3 | Usage Type Characteristics..... | 26 |
| | Usage Types BI and BI Java..... | 26 |
| | Usage Type DI..... | 28 |
| | Usage Type EP..... | 29 |
| | Usage Type EPC | 30 |
| | Special Case: Federated Portal Network | 31 |
| | Usage Type PI..... | 34 |
| | Usage Types AS ABAP and AS Java | 35 |
| | SAP NetWeaver Master Data Management (MDM) – Additional Capability | 35 |

1. Business Scenario

The purpose of this paper is to introduce the *SAP NetWeaver – Based System Landscapes*. This paper is based upon SAP NetWeaver **Release 7.0 (2004s)**. Target audiences for this paper will be people with some technical inclination, who want to get a better understanding of SAP NetWeaver landscape design, and what it can do for our customers. This paper relies on an understanding of SAP systems based upon release 4.6C and below. It is intended to give the reader a basic awareness of the differences, new features and issues introduced by the technology of SAP NetWeaver 7.0. **However, please note the following: (a) some portions of this paper will be fairly technical in nature; (b) this paper by no means should be considered as sufficient knowledge base for positioning or implementing SAP NetWeaver, nor can it be considered as an alternative to the RKT or training courses.**

2. Background Information

Before discussing the topic of designing a SAP NetWeaver based system landscape, it is necessary to define what the term *landscape* means. Throughout this paper references to the term “landscape” will have two distinct but separate definitions: “vertical landscapes” and “horizontal landscapes”.

A landscape is a logical grouping of systems. The grouping of landscape can be horizontal or vertical.

- *Horizontal landscapes* comprise of two or more SAP systems (system IDs – SIDs) that support “promote to production” of software for a particular piece or set of functionality – for example, the development, quality assurance and productive systems for the BI functionality is the “BI landscape”.
- *Vertical landscapes* comprise of the systems in a particular area of the landscape – for example, all of the systems that run productive services are the “production landscape”.

3. SAP NETWEAVER BASED SYSTEM LANDSCAPE DEPLOYMENT OPTIONS

As with any software implementation, and SAP NetWeaver based systems are no different in this case, the ideal software landscape to support the implementation is comprised of environments supporting three distinct needs that provide a solid ‘promote to production’ change management and change control process for all configuration and developments. These environments should provide:

- An environment where customizing and development can be performed. The environment should be representative of the productive environment and contain all product production customizing, developments and a sampling of production data. In addition new projects’ developments, customizing and data will exist in the system and this environment will be used for unit testing. This environment is used for as the initial environment for resolution of production issues and routine maintenance support.
- An isolated and stable environment for testing the customizing, developments, and maintenance support changes. The environment is representative of the productive environment and contains all product customizing, developments and in most cases production quality data. In addition this environment will also have newly completed customizing/developments that are in quality testing phase prior to productive release. The typical testing that occurs in this environment is regression and integration testing. No development tasks are performed in this environment, just quality assurance tasks. This environment may also be used for replicating and debugging productive issues.

- An isolated and stable production environment. The environment is the system of record and only contains productive customizing and developments. No development tasks are performed in this environment, just productive tasks. This environment may additionally be used for debugging productive issues.

This 'promote to production' scenario is recommended when implementing any system based on SAP NetWeaver. This is typically called a "Three System Landscape" with 1 Production system (PRD), 1 Quality Assurance system (QAS), and 1 Development System (DEV).

Recommendation

SAP recommends the three system landscape for all SAP applications, SAP xApps and SAP NetWeaver systems.

Many customers supplement a three system landscape with a fourth environment (Figure 1): A standalone sandbox environment used for destructive testing, learning, and testing. The landscape is still called a *three system landscape* as the sandbox is not part of the 'promote to production' landscape. Figure 1 below diagrammatically explains the relationship.

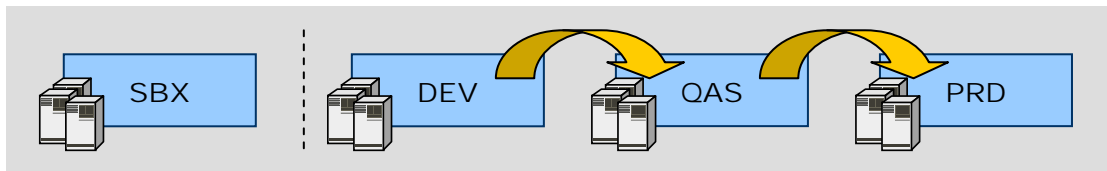


Figure 1: Typical landscape and “promote to production” process

A customer can choose to combine the above environments in a more minimal 2 system landscape – this landscape is atypical for SAP deployments and the customer must manage the additional risk and challenges of separating and isolating the different environment activities from each other and maintaining a stable and productive environment. SAP does not recommend a 2 system landscape because of the risk to successful change management when customizing, development and quality assurance testing is performed in one system.

Furthermore customers can choose to extend the three system landscape to become a 4 system landscape. This can be appropriate for customers who have:

1. Extensive distributed and parallel development teams or
2. If the customer has the need to separate the quality assurance processes into two distinct environments.

These can result in the following additions to the 'Three System Landscape' to create a '4 system landscape':

1. Additional Development system: An additional consolidation system is inserted into the landscape to consolidate distributed developments and customizing.
2. Addition of an additional quality environment into the landscape and assigning specific testing needs to each of the quality environments (typically it is seen that one system performs Application and performance testing and the second is used for integration, user acceptance and regression testing).

Typical applications of 4 system landscape are for heavily regulated or risk managed companies and for customers who use the system in a heavy Independent Software Vendor (ISV) development type model.

 Note

This paper and the examples provided assume that the customer is implementing the 'Three System Landscape'. The information provided in this paper can be adapted and extended to cover a 4 system landscape.

3.1 Overview of the Roles of Systems within the 'Three System Landscape'

3.1.1 Development System (DEV)

All customizing and development work is performed in this system. All system maintenance including break-fixes for productive processes is also performed in the system. After all the changes have been unit tested, these changes can be transferred to the quality assurance system (QAS) for further system testing.

The customizing, development and production break-fix changes are promoted to the QAS system using the change management system. This ensures consistency, management, tracking and audit capabilities thus minimizing risk and human error by eliminating manual repetition of development and customizing work in each system.

3.1.2 Quality Assurance System (QAS)

After unit testing the customizing, development and break-fix changes in the development system (DEV), the changes are promoted to the quality assurance system (QAS). Here, the configuration, development or changes undergo further tests and checks to ensure that they do not adversely affect other modules.

When the configuration, development or changes have been thoroughly tested in this system and signed off by the quality assurance team, it can be promoted to the production system (PRD).

3.1.3 Production System (PRD)

A company uses the production system (PRD) for its live, productive work. This system—containing the company's live data—is where the real business processes are executed.

The other systems in the landscape provide a safe approach to guaranteeing that only correct and tested (that is not defective) new developments and/or customizing configurations get deployed into the productive system. Additionally they ensure that **changes** to productive developments and configuration by either project enhancements or maintenance do not adversely affect the production environment when deployed. Therefore the quality of the DEV and QAS system and the implemented change management processes directly impacts the quality of the production system.

3.1.4 Background and definitions

Diving down into one of the systems within the landscape, we need to explain some of the key concepts of the underlying architecture of the systems. A system running on SAP NetWeaver's ABAP or Java runtimes is based on the three-tier architecture concept. This three-tier architecture concept is a foundation of accepted client server architecture principles ([http://en.wikipedia.org/wiki/Three-tier_\(computing\)](http://en.wikipedia.org/wiki/Three-tier_(computing))). SAP has been utilizing this concept since the release of SAP R/3 in 1992. This is a gross simplification of the concept but this helps to explain how systems within a landscape are comprised.

 Note

Three-tier architecture concept: This concept is specific to the architecture of **ONE** system in the landscape and should not be confused with the 'three system landscape' concept.

At its simplest form, a three-tier architecture system consists on a database layer, an application layer and presentation layer.

- The database layer serves to store and retrieve data for the application layer.
- The application layer provides the following; application logic, a runtime, system management and operation tools, development and change management environments and serves as an abstraction layer for the database and operating system.
- The presentation layer usually executes on the client (such as a PC or a PDA) and provides a front-end to the applications running on the application layer.

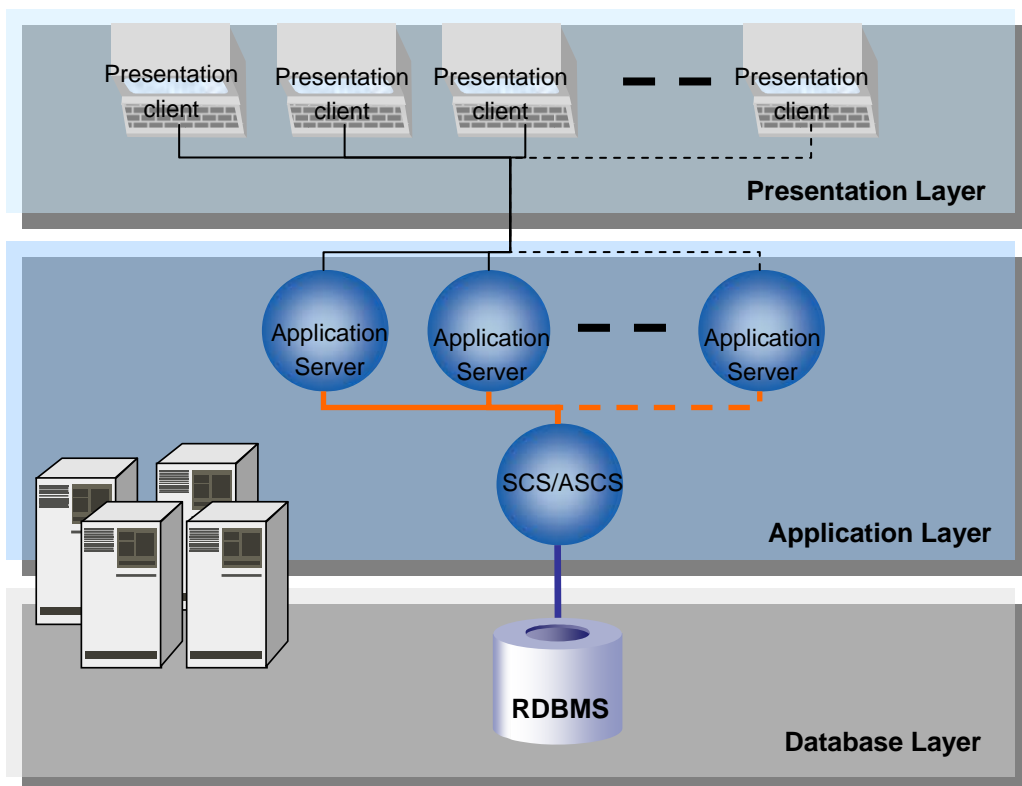


Figure 2: Three-Tier Architecture Example

What is a software based system?

A system typically is a grouping of software installed to meet an intended purpose such as an ERP system or a Data Warehouse system.

 Note

This is no different for SAP NetWeaver systems;

A pure SAP NetWeaver system serves as an integration layer that can host business applications. Depending on the intended use of the system, all SAP NetWeaver software components may be installed or activated or only some of them. After installation, the SAP NetWeaver software components in question are operated as a single unit (independent of other systems) for software life-cycle management purposes

Important

Instance - An administrative unit that groups components of an SAP system that provide one or more services. These services are started, stopped, and monitored at the same time. All components belonging to an instance are provided with parameters using a common instance profile. An instance runs on a physical host. However, there can be multiple instances on a host. An instance is identified using the system ID (SID) and the instance number. A central SAP system consists of a single instance in which all required SAP services are provided.

3.2 Installation options available for a single system

Typically the simplest system is a central system with all software required installed together on one application layer and running on a single server – another term for this is a single central application server. The presentation layer connects to the single application (server) layer.

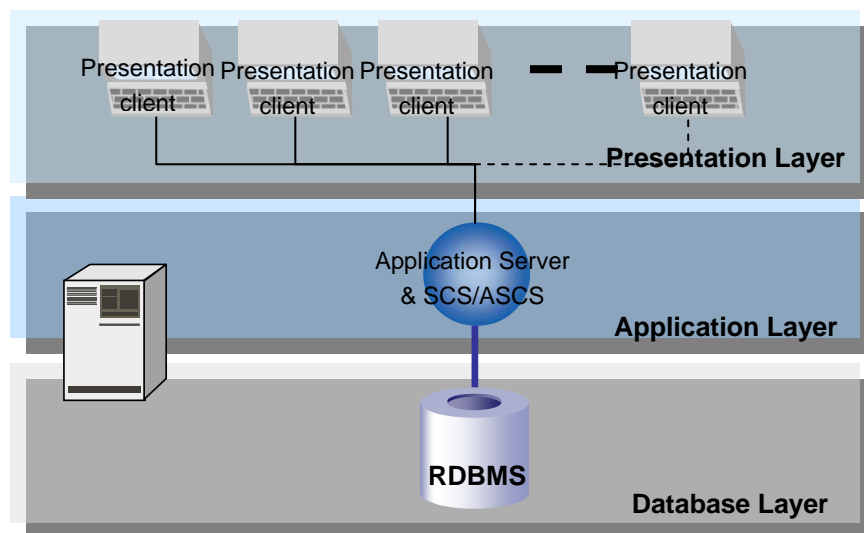


Figure 3: Single Central Application Server

The preferred installation method is to separate the critical services that are considered single points of failure (SPOF) – the enqueue server and message server – into their own instance, apart from the non-critical, redundant¹ services of the system. For Java this is known as the SAP Central Services (SCS) instance and for ABAP it is known as the ABAP SAP Central Services (ASCS) instance. Installation of an SCS is automatic for Java systems. With SAP NetWeaver 7.0, the setup of an ASCS is also possible with the installation tools and recommended. In this case, the central instance itself will no longer be a Single Point of Failure (SPOF) (nor will it be so “central,” as it will then have the characteristics of any other application server)². For high availability, then, only the SCS/ASCS need be protected by failover solutions. The application servers are not SPOFs as users may connect to another application server in the event of a failure.

This system can be scaled out by adding additional application server instances of the software to the application server layer. These additional application servers are typically being installed on additional hardware servers or on dedicated hardware allocations on one server. This provides the ability of the application server to scale across multiple server hosts. Logically this is still one system running the same business application as a single server installation – the SAP NetWeaver technology in the

¹ Application servers are considered redundant units, since more than one application server can be installed on separate machines. This type of configuration eliminates application servers as Single Points of Failure, and at least some users will be able to survive a crash of a dialog instance.

² To create (A)SCS instances from an installed central instance see SAP Note [821904](https://www.sap.com/support/821904).

application layer ensures that this works correctly. Additionally the system can be configured to load balance users and processes to specific groups of application servers – this provides administrators more flexibility in the management of resources to particular processes and users. There is only one database layer for the application layer.

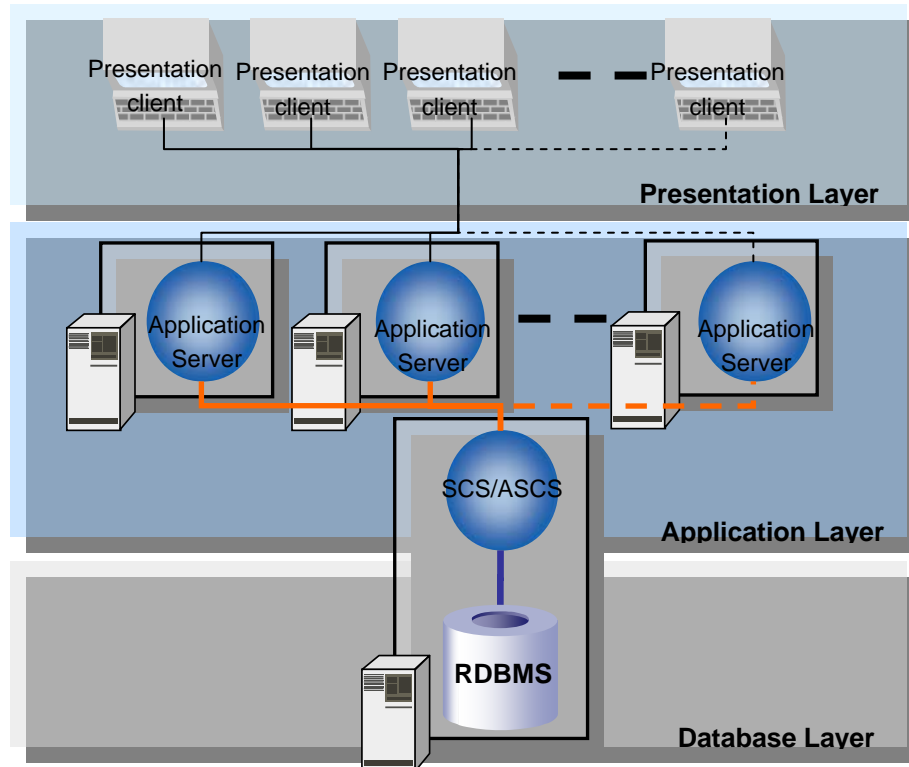


Figure 4: Scaling the Application layer with addition application servers

3.2.1 Considerations with the traditional installation options available for a single system

There are some possible challenges to adopting a “one approach fits all” model for every system installed within a landscape. These challenges are not unique to SAP software but a common challenge for all customers who have more than one system landscape and are primarily related to the areas of TCO and managing the expanding total landscape. If a three system landscape approach is used and replicated for every system deployed in a landscape, the number of individual systems quickly inflates (see figure 5 below).

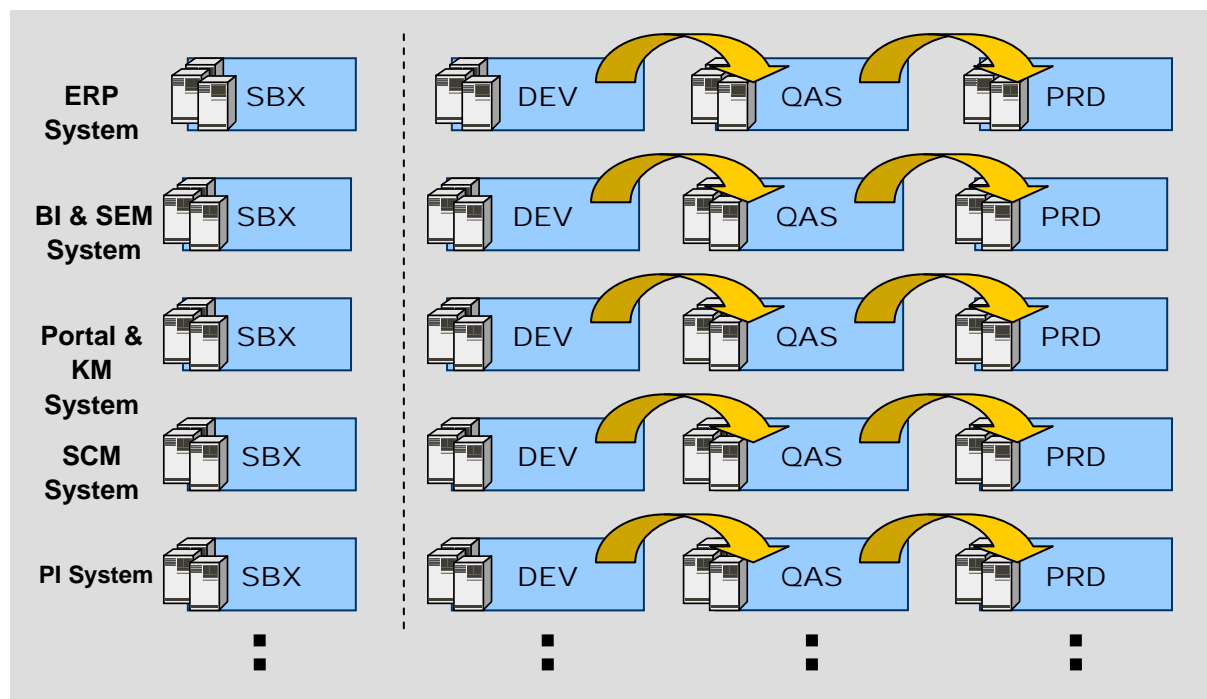


Figure 5: Multiple SAP Applications and technology landscape with promote to production paths

The design of a system landscape should incorporate the current and future impact of systems in the landscape. Considerations should be made on the areas of costs, operations, disaster recovery, high availability, system interdependence and maintenance [these choices are discussed in detail from an SAP NetWeaver perspective later in this paper]. As you can expect, it is not feasible to formulate one answer that each new system or technology will need a new system landscape. In the next section we will explore installation options available to SAP NetWeaver deployments.

3.3 Landscape options for SAP NetWeaver

3.3.1 'Server virtualization'

While not an official term at SAP, 'server virtualization' or 'server sharing' in this case simply means a process for reducing the number of individual servers utilized within a landscape and having multiple systems and instances consolidated and installed on a single large server (hardware/operating system virtualization). The server's resources (CPU, Memory) are typically available to all systems and instances. Segregation and assignment to individual systems is possible using virtualization tools delivered by the hardware or software vendor (for example HP's Superdome and Sun's System Domains to name a couple). Depending on the hardware vendor additional resource management tools may be available to support dynamic resource allocation. For more information on what might be available to you for your SAP Systems, please contact your hardware vendor directly.

In server virtualization each of the systems is viewed as independent system, each with its own database – a 1:1 relationship. In most cases the operating system is common to all systems on the server (Figure 6). This reduces the number of individual server hosts that have to be maintained, monitored, and patched, and so on independently.

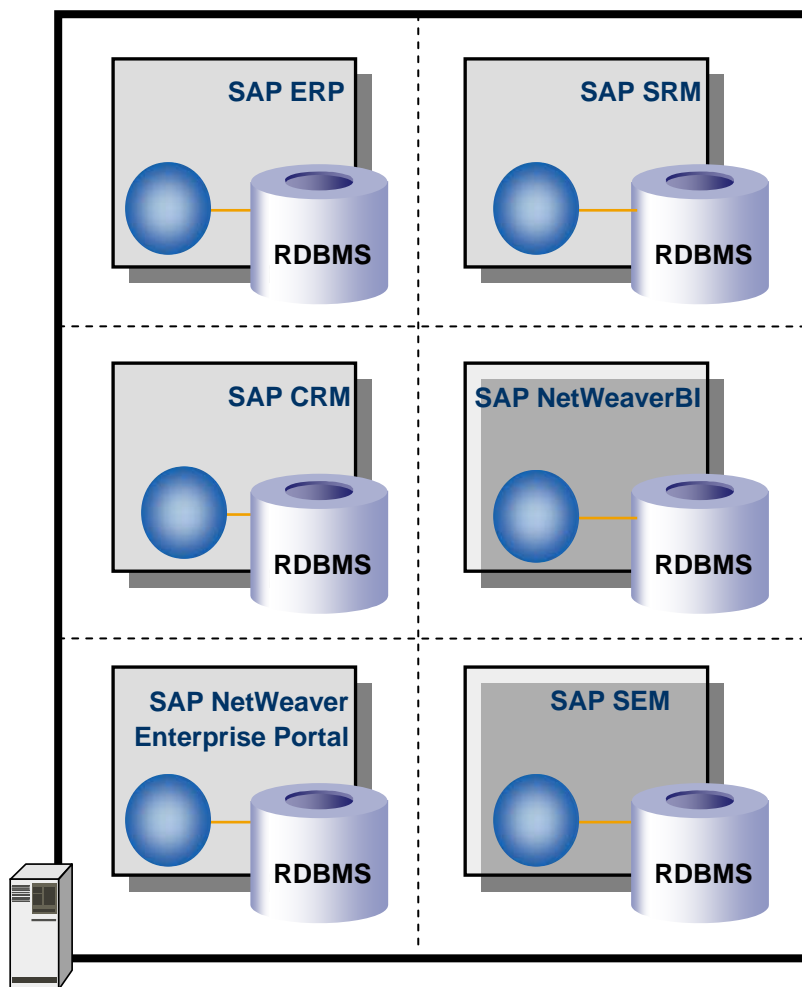


Figure 6: Multiple Systems with Applications, Multiple databases and One server

In such a configuration it is possible that one server could host all the systems for a particular application (for example SBX, DEV, QAS, and PRD for a single application), systems of similar type (for example all PRD systems for different applications) or any combination thereof. Hosting PRD systems across different landscapes on one host is quite common as high availability (HA) and disaster recovery (DR) investments can be targeted to that server. The non-PRD servers can have differing HA and DR solutions.

The SCS/ASCS and its application servers can also optionally all be installed on the same physical server (with or without virtualization) versus the traditional installation on multiple server hosts. The choice for scaling is open to the customer to determine based on their server strategy.

It should be noted that server virtualization is not limited to SAP NetWeaver based systems but to any systems in a customers' landscape. It is however recommended that the SAP systems have their own dedicated virtual server and not share with other non SAP Applications. The reason for this recommendation is that sharing adds complexity to the administration and operation of the system to customer – SAP applications are not aware of the other non-SAP applications running and therefore may conflict for resources (CPU, Memory, IO).

3.3.2 Multiple Components in One Database (MCOD)

MCOD addresses one of the challenges of system administration - managing multiple independent databases for individual systems - each system's database has to be maintained, monitored, backed-

up, etc independently. This redundant maintenance and associated operational costs can have significant impact in an environment with multiple application systems.

MCOD is a simple concept; enable SAP NetWeaver based systems (and their underlying application and software components) to be installed into and share a single database. This is visualized in Figure 8 below. Each system's data is isolated in the database (for example no table sharing) and each system's application(s) run as if they were installed independently. This is achieved by having each system assigned to a unique SAP system identified (SID) in the database – used for storage of the application's data tables. On the non-Database side each system in a MCOD configuration is administered and maintained independently to each other.

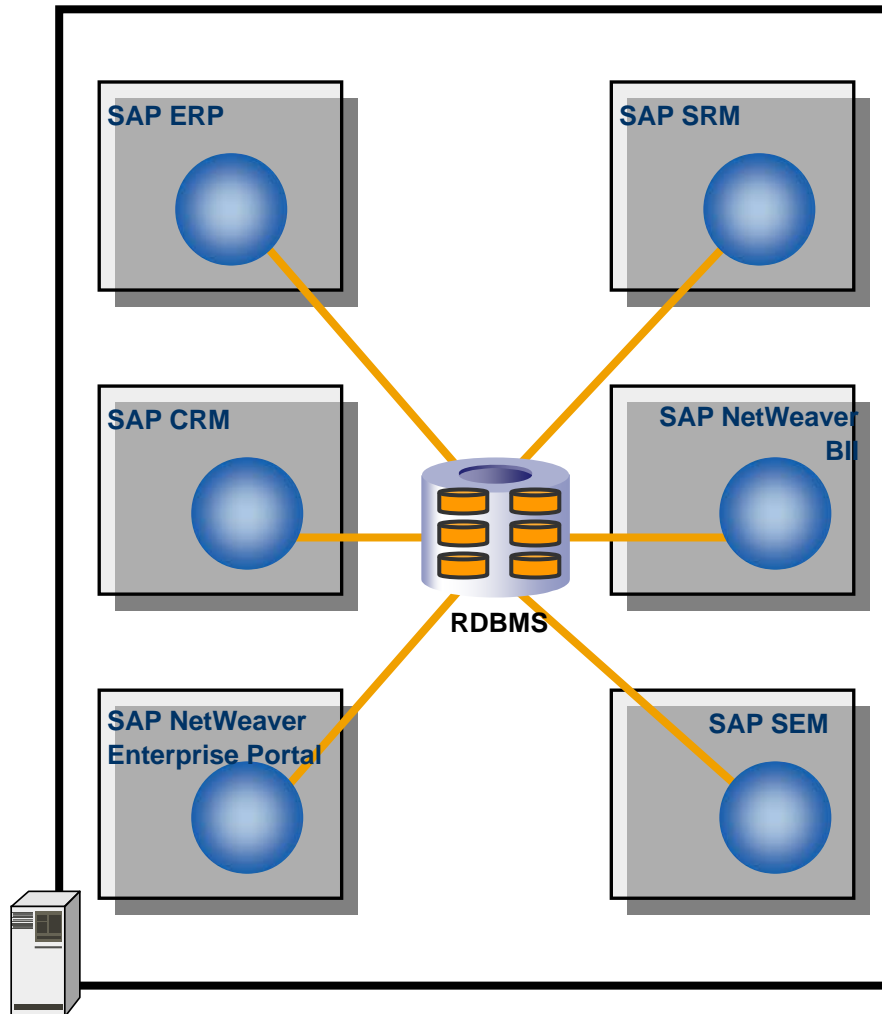


Figure 7: - Multiple Systems with Applications, One Database, One server

The MCOD option provides some major benefits to a system administrator in a number of areas apart from the obvious reduction in the number of databases to administer. As one database is utilized this configuration provides a simpler option for synchronized back-ups and recovery. All systems with the database can be recovered to one single point in the time – of course it could be countered that this single database is a single point of failure but database technology today is such that this can typically be mitigated. Additional benefits include the reduction of database related expenses such as 3rd party tools and in some cases the optimization of the use of database licenses through reducing the number of databases and/or the reduction of licensing for idle resources (ex. more optimal use of database CPUs).

3.3.2.1 Availability

MCOD is available for most SAP applications based on SAP NetWeaver AS 6.10 or higher and for SAP R/3 4.6C onward. More information on availability can be found at the MCODE homepage:

<http://service.sap.com/mcod> -> Availability

(<https://service.sap.com/~sapidb/011000358700001533752002E>)

3.3.2.2 Sizing

Systems in a MCODE configuration are sized independently. For total server sizing, the sizing is additive for the shared database and server perspective – the sum of all individual system sizing. When working with the results of sizing tools like Quicksizer from SAP and with questionnaires from your hardware vendor, please ensure you indicate that you would like to size the database and server for a MCODE installation.

CAUTION

Remember each system is sized independently but the sizing for the underlying shared database and servers will be additive.

3.3.2.3 Benefits of MCODE

- Flexibility through ‘componentization’ of systems – each system is independent and therefore can have its own change management and maintenance cycles
- Reduction in the number of database instances - multiple independent and different software solutions are located in one database.
- One logical and physical database instance – for back-up and recovery and from a database administration viewpoint it is one database for operations, tuning and maintenance.
- All systems use the same release for the database – simplified patching and maintenance through standardization.
- Synchronized Disaster Recovery (backup & restore) – One back-up and recovery option for all MCODE systems.
- Separate upgrades possible – each system can be upgraded independently with independent timelines.
- High Scalability/High Availability model for all application must be considered together – since there is a single database, scaling the applications through additional application servers adds additional load on the database and affects all installed applications. Similarly, if the single database is put into a high availability environment, database failover affects all installed applications.
- Independent tuning and administration – from the SAP system perspective, each system can be tuned and maintained independently

Note

Please note that the challenges detailed above are perceived generic challenges and each customer should analyze their specific situation when considering this system installation option.

General landscape decision points are discussed later in this paper.

3.3.2.4 Administrative considerations for MCODE system installations

Please note that the MCODE configuration does not specifically introduce these challenges as the majority of them also exist for individual systems on dedicated servers.

- Maintenance of multiple applications – Duplicate systems and application administration efforts
- High availability solution costs – each system may need its own High Availability solution and it may entail differing have different levels of High Availability
- Single point of failure for all systems – The database
- CPU/Memory tuning aspects – A holistic view of all systems resource consumption and needs must be considered when tuning the server.
- Database release(s) can impact application release(s) – application upgrades may be restricted based on the database release that must be supported by all application systems

Note

Please note that the challenges detailed above are perceived generic challenges and each customer should analyze their specific situation when considering this system installation option.

General landscape decision points are discussed later in this paper.

3.3.3 Bundling of Usage Types

The ability to bundle usage types in a single system (SID) is an additional deployment option. Where MCODE addresses management of many independent database installations, bundling addresses management of many independently systems **AND** databases.

The premise is simple with bundling of usage types; it enables the option to combine many usage types into a single system.

In the past the usage types that could be installed in a MCODE implementation have typically only been available as separate systems but SAP technology has changed in recent releases: a SAP NetWeaver based system now has the ability to run many different SAP NetWeaver usage types on **ONE** system.

To sum it up, this is one system with one database, one SID and running two or more usage types.

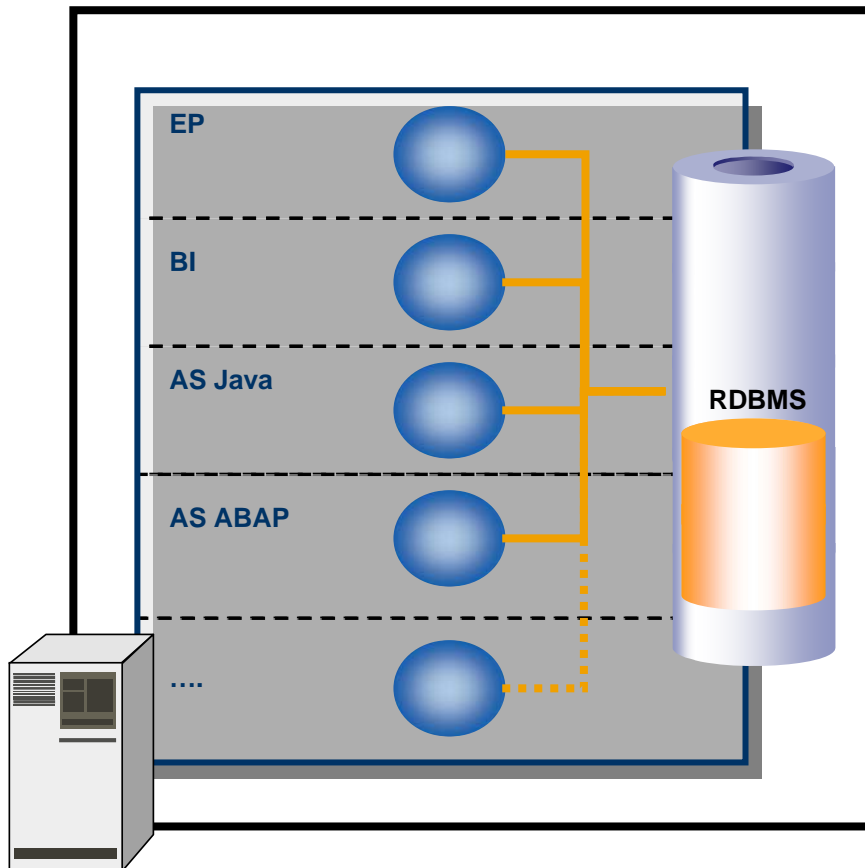


Figure 8: Multiple Applications, One System, One database, One server

3.3.3.1 Availability

For SAP NetWeaver installation bundling usage types is available from SAP NetWeaver 2004 SR1 onward.

Important

SAP applications and SAP NetWeaver will determine which of their software may or may not be combined in one system. For more information on the possible installation options you should consult the respective SAP Master Guide. These are available for download at <http://service.sap.com/instguides>.

3.3.3.2 Benefits of bundling usage types

- A single installation of different but compatible software solutions is located in one database – a single system for administration and maintenance.
- One logical and physical database instance – for simplified synchronized back-up and recovery
- From the DBA viewpoint it is one database for operations tuning and maintenance (as opposed to one database per system)
- All usage types use the same OS/DB release – simplified tuning, patching and maintenance through standardization on one release/patch set for combined SAP applications
- Synchronized DR (backup & restore) – One database and SID for automatically synchronized back-ups.

- Simplified Maintenance – one system for change management, one system to patch and one system to upgrade.
- High Scalability/High Availability – Joint protection of critical services bundled in one system is potentially a lower cost as only one High Availability solution needed
- Single tuning and administration – one system to monitor and one system to tune!

 Note

Please note that the benefits detailed above are perceived generic benefits and each customer should analyze their specific situation when considering this system installation option.

- Many of these benefits are also the same for MCOB installations (that is the database related benefits).
- Using the same OS/DB release can also be achieved with separated systems.
- For High Availability solutions, grouping the SCS/ASCS instances of separated systems within the same HA infrastructure (to save costs) is also possible.

Because of this, the most significant benefits of bundled installations are the simplified maintenance and single tuning and administration.

General landscape decision points are discussed later in this paper.

3.3.3.3 Sizing

Some brief words about sizing systems with bundled usage types. When bundled SAP NetWeaver usage types are sized independently in an additive view (independent sizing is added together to get the holistic system sizing). So for total server sizing, the sizing is additive for the shared system, the shared database and server perspective – the sum of all individual system sizing. When working with sizing tools like Quicksizer from SAP and with questionnaires from your hardware vendor, please ensure you include all sizings for the system together and indicate that you would like to size the system, the database and the server(s) for a bundled installation.

 Important

Remember each system is sized independently but you need to add all sizings together to get the holistic sizing.

3.3.3.4 Administrative consideration for bundled installations

- Maintenance of one system for all users of all applications – removes duplicate systems and application administration needs
- High availability solution – All application is a bundled usage type system will share the same HA and DR solution.
- Backup and recovery cross application synchronization is simplified via one system concept.
- Single point of failure – the database and the central instance/application server become critical
- CPU/Memory tuning aspects – A holistic view of all systems resource consumption and needs must be considered when tuning the server.
- All maintenance tasks are synchronized for all applications (that is patching, upgrades, technology changes, etc). Similarly, the downtime for all applications is also aligned (which may be less flexible).

Database release(s) can impact usage type release(s) – usage type upgrades may be restricted based on the database release must be supported by all usage types.

 **Note**

Please note that the challenges detailed above are perceived generic challenges and each customer should analyze their specific situation when considering this system installation option.

General landscape decision points are discussed later in this paper.

3.3.3.5 Restrictions

In addition to the restrictions outlined below, there are also limitations to be considered in the section “Sharing Usage Types” that should be noted and understood.

The one major restriction with installations that are shared either through MCOD or bundling is that all of the SAP software configuration must support the same database parameters configuration, database release and patch levels. In general, all the cross-database related activities have a potential effect on all systems.

For more information on recommended database parameters please consult the SAP software installation manual and the respective database parameter recommendation SAP Notes on the SAP service marketplace (<http://service.sap.com/notes> -> keywords search on your application release and database name and key word “parameters”)

For more information on the SAP software and database compatibility please consult the SAP software product availability matrix (<http://service.sap.com/pam>).

There are restrictions related to code page for the systems. These restrictions are a result of the AS ABAP usage type’s support for Unicode and non-Unicode code pages (more information on Unicode and non-Unicode code page support can be found on the SAP Service marketplace <http://service.sap.com/unicode@sap>).

| Usage Types | Unicode code page | Non-Unicode single code page | Non-unicode MDMP code pages ¹ |
|-------------|-------------------|------------------------------|--|
| AS ABAP | ✓ | ✓ | ✓ |
| AS JAVA | ✓ | - ² | - ³ |
| DI | ✓ | - ² | - ³ |
| EPC | ✓ | ✓ | - ² |
| EP | ✓ | - ² | - ² |
| PI | ✓ | - ⁴ | - ⁴ |
| BI | ✓ | ✓ | - ⁵ |
| BI JAVA | ✓ | - ² | - ⁵ |
| MI | ✓ ⁶ | ✓ ⁶ | - ⁶ |

1 - Multi-Display Multi-Processing (MDMP) is only available in the ABAP stack for SAP ERP 2004 (SAP ERP 6.0 (2005) based upon SAP NetWeaver 7.0 does not support MDMP).

2 - The AS JAVA usage type and usage types based on AS JAVA are Unicode only. This means they can connect to an AS ABAP based on this code page configuration.

3 - As above and no out of the box SAP application functionality support for MDMP

4 – PI usage type is Unicode only

5 – MDMP not supported for BI and BI Java usage types

6 –It is recommended that the MI usage type be installed on a dedicated system. For SAP NetWeaver 7.0 it is not recommended to have the SAP NetWeaver Mobile usage type (MI) deployed in a bundled system due to maintenance considerations for mobile applications. It is required that an MI system with AS ABAP has the same Unicode type as the MI back-end system: If your MI back-end system is a Unicode system, also install a Unicode MI system (that is, both AS ABAP and AS Java on Unicode); if your MI back-end system is non-Unicode, install an MI system with non-Unicode AS ABAP (that is, AS Java on Unicode and AS ABAP on non-Unicode).

 Recommendation

For new installations, it is **recommended** to install a Unicode based system for all SAP applications and SAP NetWeaver deployments that require an AS ABAP usage type based system (AS JAVA is only available in Unicode). In future all new installations from SAP will be Unicode only.

4. DEFINING AN SAP NETWEAVER SYSTEM LANDSCAPE

4.1 Business process requirements

Implementing an enterprise SOA enabling landscape can be very difficult for the IT department. The SAP NetWeaver platform contains many capabilities that make it challenging for the landscape designer to determine which pieces need to be implemented and where. There is the question of where to start? What systems should be service enabled? What systems need to be installed and what systems need to be upgraded? Understanding how to answer these questions is largely uninteresting to the business until there is an enterprise SOA scenario they wish to exploit.

The best approach is to let the business be the driver for the technology implemented in the landscape. What are the current industry and market trends and how does the business want to take advantage of them? Understanding the business will help create strategic objectives. Once these objectives are determined they can be translated into process requirements. The process requirements will then drive a solution approach which is then implemented with IT architecture. That is, business processes being implemented, and their underlying functionality, will drive and specify what systems need to be installed, what systems need to be upgraded, and what systems can stay in place.

4.2 Installable Units

4.2.1 Building Blocks

The system implementation of business processes is performed by installing specific groupings of building blocks of SAP NetWeaver on systems within the landscape. These building blocks are:

- Systems with usage types
- Standalone engines
- Clients
- Appliances

4.2.1.1 Systems with usage types

SAP systems are the main building blocks of SAP NetWeaver. They are identified by unique SAP system IDs (SAPIDs). A usage type determines the role which an SAP NetWeaver system plays in a landscape and the intended purpose of a system. Usage types are realized by installing *and* configuring a collection of *software components*. Usage types may also be built on and/or leverage other usage types.

- A usage type can share a system with other usage types
- In some cases, a usage type can be run separately in different systems (so interfaces between usage types must work locally as well as remotely)
- Upgrade from previous SAP NetWeaver releases may require configuration of smaller units for example, Knowledge Management and Collaboration (KMC), which is part of “EP” usage type
- A usage type (per definition) is **not** an installable unit. The actual installable units are the underlying *software components* that are installed with the installation tool(s).

There are fundamental differences between the concepts: “software component”, “usage type”, and the business process scenarios being implemented. These concepts are interrelated and their relationships are best summarized as:

- A software component is a collection of objects delivered to provide a certain capability – for example the XSS component delivers the necessary Web Dynpros to realize Employee Self Service and Manager Self Service in SAP ERP 2005. Software components are dependent on usage types – that is one cannot deploy XSS into an AS Java usage type without the EP usage type.
- A usage type encapsulates a key capability of SAP NetWeaver. This SAP NetWeaver functionality is provided in many underlying software components. The usage type serves as a placeholder for these capabilities, such that if you want to use any of this functionality, you need a system with this usage type active. By defining usage types the possible permutations and combinations of software component deployments are eliminated. This creates uniformity in the build out of SAP NetWeaver landscapes for both customers and SAP development alike.

SAP NetWeaver 7.0 has the following usage types:

- Business Intelligence (BI)
- Business Intelligence Java (BI Java)
- Development Infrastructure (DI)
- Enterprise Portal (EP)
- Enterprise Portal Core (EPC)
- Process Integration (PI)
- Application Server ABAP (AS ABAP)
- Application Server Java (AS Java)
- Mobile Infrastructure (MI)

There are also the following additional capabilities

- Master Data Management (MDM)
- Identity Management

4.2.1.2 Standalone Engines

Standalone engines are additional software units that are installed separately. Working as a standalone server they provide ONE specific (server) functionality in combination with one or more SAP systems (for example TREX). Standalone engines are not part of a usage type. They do not run on AS ABAP or AS Java and they are never considered to be full-blown SAP NetWeaver systems.

For SAP NetWeaver, the following standalone engines are available:

- Content Server
- Gateway
- SAP Central Process Scheduling application by Redwood
- liveCache
- Search and Classification (TREX)
- BI Accelerator
- Web Dispatcher

Detailed information on standalone engines may be found in the *Master Guide – SAP NetWeaver*.

4.2.1.3 Clients

Clients are additional installable programs or tools. They either reside on local front-end PCs accessed by users or on back-end systems where they act as client programs within an SAP NetWeaver system landscape.

SAP NetWeaver has the following front-end clients and tools:

- Adobe® LiveCycle™ Designer
- Business Explorer (BI)
- J2SE Adapter Engine (PI/XI)
- Mobile Infrastructure Client
- SAP GUI
- SAP NetWeaver Developer Studio
- SAP NetWeaver Developer Workplace
- Web Browser

Detailed information on Clients may be found in the *Master Guide – SAP NetWeaver*.

4.2.2 Administration Systems

In addition to the landscape building blocks above, an SAP NetWeaver landscape typically also includes shared service systems that are used for administration activities for all of the building blocks.

Candidates for these systems include:

- Support infrastructure
 - SAP Solution Manager
 - Solution Manager Diagnostics
- SAP NetWeaver Administrator

- Alert Monitor
- SAP Central Process Scheduling application by Redwood
- System Landscape Directory (SLD)

Detailed information on shared services may be found in the *Master Guide – SAP NetWeaver*.

4.3 Usage Type Characteristics

4.3.1 Planning an SAP NetWeaver System Landscape

Now that the foundation elements of SAP NetWeaver landscapes are understood, landscape planning can take place. As a guideline, SAP recommends the following approach to determine the landscape:

1. Determine the business processes of SAP NetWeaver to be implemented.
2. Determine which installable software units (that is, systems with usage types, standalone engines, and clients) are required for these processes.
3. Determine the required shared services to run in central systems in the system landscape and the landscape aspects relevant for their implementation.
4. Determine how many systems are required and how the systems are used.
5. Together with the hardware partner and technical consultant, map the required systems and standalone engines of SAP NetWeaver to hosts.
6. Implement the system landscape of SAP NetWeaver.

For each scenario, the *Master Guide – SAP NetWeaver* provides one section with further details about scenario-specific aspects for the system landscape.

This guide will help you assess the landscape-relevant specifics for the usage types that need to be considered. The following sections give an overview of factors and limitations to consider when sharing usage types. How you judge the different aspects will depend on your requirements. What is a drawback for one customer may be an advantage for another customer. No general guideline can be given, as this relies heavily on your requirements. As a result, it is mandatory to plan your system landscape well in advance according to your current and possible future requirements. It is recommended to perform this task together with a technical consultant.

4.3.1.1 Sharing Usage Types

With SAP NetWeaver you may (or may not) use a single system for multiple usage types. Accordingly, it is important to understand the technical restrictions for systems configured with multiple usage types. If you have one system with multiple usage types, SAP does not provide **standard tools** to separate these usage types to multiple systems at a later stage. For example, if you have a system with usage types A and B, you are not able to migrate it with SAP standard tools into two systems, one with usage type A and the other with usage type B.

In addition, SAP does not provide standard tools to merge multiple systems with different usage types into one system with multiple usage types at a later stage.

If you wish to perform these tasks, you require specific project support from SAP³.

³ Although you cannot merge/split existing usage types, you can install additional usage types into an existing system. See SAP Note [883948](#).

- Vertical scaling is a technique for a cluster arrangement that includes many application server instances created on one physical machine. The single machine needs enough resources to handle this configuration. We recommend that you monitor performance and memory before adding a new application server instance.
You can set up vertical scaling whenever required as special installation steps are not necessary.
- Horizontal scaling is another clustering technique that separates the application server instances across multiple physical machines. This configuration provides failover support and so ensures high availability of the application server processes. The disadvantages of this strategy are the increased installation and maintenance effort and also the cost of additional servers⁴.

With a single instance multiple components reside on the same machine. It is easy to install and maintain and is therefore appropriate for first-time installations when you want to test and familiarize yourself with the functions of the application server. At this stage, it is easy to determine what kind of topology is needed and to plan the system landscape.

Although this topology is easily and cheaply configurable, it has some drawbacks that you must consider. Since all components are located on a single server, they compete for the available resources, which can affect the performance of the processes. In addition, the components cannot be additionally secured with firewalls and high availability is not supported. This configuration can be enhanced with horizontal scaling.

Network design is also an important factor in terms of performance. For example, when systems are combined then RFC communication in a single system scenario may be faster. The communication would then be occurring within the server as opposed to over the network. This may or may not be significant depending on the performance of the network in the first place.

Administration – When systems are separate administration overhead increases. A single action may have to be repeated multiple times on separate systems. Each system has to be started and stopped individually, for example. Configuration actions – such as profile parameters, setup of printers, and so on – must be performed in each system. In a single system, actions only have to be performed once.

Monitoring – It is very easy to see that there are fewer systems to monitor in a single system. However, there is a higher degree of complexity in diagnostics. When an error does occur there are more possible causes in a single system scenario. In separate systems it is always known what application is causing the problem.

Summary

The following table sums up all aspects of this section:

⁴ Further information may be found in the *Technical Infrastructure Guide* available at <http://service.sap.com/instguides> -> SAP NetWeaver -> SAP NetWeaver 7.0 (2004s) -> Installation.

| | Stand-alone | Single host, multiple systems | MCOD | Multiple Usage Types on a Single NetWeaver System | Comments |
|--------------------------------------|-------------|-------------------------------|------|---|--|
| Administration and Operations | | | | | |
| Performance | + | - | - | - | In a stand-alone installation there is no competition for hardware resources. However, server cost needs to be factored in. |
| Administration | - | - | 0 | + | Administration actions must be repeated in each system. In MCOD installations database administration tasks are only performed once. |
| Monitoring | + | - | - | - | With multiple or combined system, monitoring is more complex. |

Business Continuity

The goal of business continuity is to keep the business running. For example **Business Continuity Planning (BCP)** is used to create a logistical plan for how an organization will recover and restore partially or completely interrupted critical function(s) within a predetermined time after a disaster or extended disruption. The logistical plan is called a Business Continuity Plan. BCP is how an organization prepares for future incidents that could jeopardize the organization's core mission and its long-term health. Incidents include local incidents like building fires, regional incidents like earthquakes, or international incidents like pandemic illnesses.

Availability – With increased automation levels the need for increased availability of business processing systems is necessary. When designing a highly available system, there is a tradeoff between the costs of availability and the costs of system downtime. The costs of downtime are not linear with respect to the duration of the downtime: with longer downtimes the increase in costs is more nearly exponential. For example, when supply chain processes are stuck for longer than three hours, the entire production may become stuck which then produces even higher downtime costs. Conversely, the measures taken to increase system availability — such as redundant components, a disaster recovery site, first-class system management tools, a skilled IT staff, system capacity planning, and proactive services — cost progressively more money to implement the more highly available a system becomes.

Because of the cost tradeoff factor, it makes sense to only make systems highly available that have a business need to be highly available. From the perspective of combining systems into a single system this then means that it is more advantageous to keep systems separate that have a high degree of availability necessary. This helps reduce complexity when systems do have a failure and keeps single points of failure separate from each other: you do not want non-mission critical systems that fail to cause mission critical systems to failover at the same time.

Backup and Recovery – When data is distributed over multiple systems within a landscape – as it is with any service oriented architecture – backup and restore capabilities present an increased challenge to the IT department to ensure that there is little to no data loss in the case of any system

crash⁵. When systems are sharing the same database there are advantages and restrictions that impact landscape design:

Advantages

- Consistent backup very easy
- Consistent recovery to every point in time
- Easy copying of complete landscape

Restrictions

- All systems affected by a crash or restore
- No point-in-time restore for single components

Summary

The following table sums up all aspects of this section:

| | Stand-alone | Single host, multiple systems | MCOD | Multiple Usage Types on a Single NetWeaver System | Comments |
|-----------------|--------------------|--------------------------------------|-------------|--|---|
| Availability | + | + | + | – | <p>Separate systems may be started and stopped individually.</p> <p>HA with “Single server, multiple systems” and “MCOD” complex.</p> <p>Separate systems increase availability and hardware costs.</p> <p>All systems in a bundled installation share the same availability.</p> |
| Backup/recovery | – | – | 0 | + | See Advantages and Disadvantages above. |

Business Requirements

The requirements of the business organization itself play a major role in landscape design and should not be overlooked by the IT department. When systems are separated a certain amount of flexibility is gained. For example, it is then easier to outsource services and/or sell off businesses. However, when the systems are separated, the challenge then exists to provide a consolidated view of the operations.

Organizational – How the business is organized can have a significant impact on the landscape design. Does the company operate in a single country only, or is it a global operation? A system may require configuration flexibility when there are global operations. For example, there may be local versus global business processes and even differences between business processes in each locality.

⁵ For more information on Backup and Restore concepts with SAP systems, see the Advanced Technology Group at <http://service.sap.com/atg> -> Backup & Restore.

When there is a single system the granularity necessary for differing configuration of business processes may either be lost or be very complex to configure. When there are separate systems it is easier to configure differing business processes.

Political – Like the organizational requirements above, if the corporation operates in many different lines of business it may be more advantageous to have systems separated by lines of business. This gives the flexibility to have separate business processes (and configuration) for each line.

Legal – There may be legal reporting and/or regulatory reasons to have systems separated. Many customers separate Human Resources (HR) into its own system, for example, because of the particular data security requirements and the steady legal changes that necessitate the frequent application of patches to the system.

Summary

The following table sums up all aspects of this section:

| | Stand-alone | Single host, multiple systems | MCOD | Multiple Usage Types on a Single NetWeaver System | Comments |
|------------------------------|-------------|-------------------------------|------|---|---------------------------------|
| Business Requirements | | | | | |
| Organizational | + | + | + | – | Separate systems more flexible. |
| Political | + | + | + | – | Separate systems more flexible. |
| Legal | + | + | + | – | Separate systems more flexible. |

Security, Governance and Compliance

Complying with regulations and auditing requirements is critical for any business. How the servers are secured and licensed from both an infrastructure and application perspective presents its own set of tradeoffs in terms of landscape design.

Infrastructure Security – With a single system the ability to place firewalls in between systems is lost. If the server is externally facing this can be a critical factor in maintaining data security. If the server completely resides within the corporate network and no external access is given, this may be less of a concern.

The tradeoff with having more or fewer servers in the infrastructure then comes with providing rights management between the servers. Secure communication models must be set up and replicated to each server which presents more administration overhead.

Application Level Security – Similar kinds of tradeoffs exist at the application level. With fewer systems there are fewer user accounts to manage. This comes with the tradeoff, though, that all the data is then combined in a single database. It is possible that this is prohibitive from an audit or regulatory standpoint. For example, often human resource or pharmaceutical systems are kept separate from other systems to provide different data access and security models.

Licensing – Depending on how third party tools are licensed there is an opportunity for reduced licensing fees with a single system. If tools – such as operating system and database tools, monitoring tools, scheduling tools, back-up tools, high availability solutions, print solutions, and so on – are purchased and licensed by server then implementing on fewer servers will result in lower costs for the tools.

Summary

The following table sums up all aspects of this section:

| | Stand-alone | Single host, multiple systems | MCOD | Multiple Usage Types on a Single NetWeaver System | Comments |
|--|-------------|-------------------------------|------|---|---|
| Security, Governance and Compliance | | | | | |
| Infrastructure security | + | - | + | - | Firewalls only possible in separate systems (with increased administration). |
| Application level security | 0 | 0 | 0 | 0 | Fewer user accounts, but combined data may not be preferred. |
| Licensing | - | - | + | + | Fewer servers and/or database instances can reduce licensing costs for 3rd party tools. |

Software Life-Cycle Management

When systems are combined they essentially become bound from a software life-cycle management perspective. This permanent relationship has a significant impact on system landscape design.

SAP Software Maintenance (Installations, Upgrades, Support Packages and Support Package Stacks) – All the components in a single system must be patched and upgraded together. This can be very a very substantial effort from a project perspective. For example, if two usage types share the same system, then an upgrade must cover both usage types. This requires more effort not only for the technical upgrade, but also for the system testing after the upgrade, as both usage types must then be verified.

This relationship can also affect the rollout of software. For example, in a global implementation where parallel release environments are used – that is release N and release N -1 being rolled out to separate parts of the organization – the usage types that are combined are patched and upgraded together. So system consolidation may not necessarily occur if more than one version of a usage type is necessary for development and production support.

Operating System and Database – Not all SAP applications, products and capabilities are available on all SAP supported platforms. That is, some capabilities have limited platform availability. Clearly this can affect what capabilities can be installed together in a single system or even a single server. During the landscape design process, consult the SAP *Product Availability Matrix* (PAM) on the Service Marketplace (<http://service.sap.com/pam>) to determine the available platforms for the systems and capabilities in the landscape.

Summary

The following table sums up all aspects of this section:

| | Stand-alone | Single host, multiple systems | MCOD | Multiple Usage Types on a Single NetWeaver System | Comments |
|---------------------------------------|-------------|-------------------------------|------|---|---|
| Software Life-cycle Management | | | | | |
| SAP Software Maintenance | + | + | - | - | Combined systems are upgraded/patched together. For MCOD, this may require an upgrade of the database which affects all installed applications. |
| OS & DB | + | - | 0 | - | Some capabilities have limited platform availability. |

Interoperability

The cross-interoperability of systems and scenarios has a significant impact on landscape design. For the landscape planner, it means understanding the impact that a landscape change on a single system – such as an upgrade – has on the landscape as a whole and the scenarios it supports.

Design-time and Runtime

A design-time system should not have a higher release as corresponding runtime system(s), and it is recommended to have design-time and runtime systems on the same release.

Release Strategy

To assist customers with planning, SAP publishes a Release Strategy:

- The first release of a new SAP application can be integrated with a defined set of releases of other SAP applications. A successor release always supports at least the same set as its predecessor release, as long as the application releases within this set are in mainstream or extended maintenance. After an upgrade of an SAP application, all previously used scenarios (referring to the scenarios shown on the [industry solution map](#)) are still available. However, if you want to take full advantage of the latest version of a scenario or of a new scenario on an industry solution map, you might need to upgrade more than one SAP application within your landscape. For further information see *SAP's Release Strategy For All Major Shipments from 2005 Onward* at <http://service.sap.com/releasestrategy>

{tc "2.1.1.1 Systems with Usage Types (boirs Zarske)" \ 4MERGEFORMAT}Definition of SAP NetWeaver Hub

For landscape architecture modeling purposes, it is best to consider if a system is providing its services to multiple systems and scenarios in the landscape (1:many), or if it is providing its services to a dedicated system/scenario in the landscape (1:1).

An SAP NetWeaver hub is an SAP NetWeaver system whose purpose is to connect multiple systems in a system landscape and to provide services centrally for these systems.

For example, a BI system running as an enterprise data warehouse for several systems in a landscape would be an SAP NetWeaver hub. In contrast to this, the BI Accelerator appliance has a one-to-one relationship with one BI system and does therefore not function as a hub.

The advantages of central services are that they reduce redundancy, help to manage complexity, and reduce time to implement changes.

4.3.1.3 Usage Type Characteristics

With SAP NetWeaver it is possible to both combine and distribute usage types amongst systems within the landscapes. Whether or not to do so is up to the landscape designer and there are characteristics about each usage type and interdependencies that play a role in determining what is considered by the designer to be an optimal setup for their landscape. This section discusses each usage type and the landscape relevant factors that have an impact on landscape planning.

Usage Types BI and BI Java

BI requires AS ABAP as a prerequisite in the same system. Usually, scenarios running on usage type BI also require usage type BI Java, and there is a 1:1 relationship between BI ABAP and BI Java. Optionally, it can be combined with other usage types in one system as well. A full installation of BI-based scenarios includes the ABAP stack (for usage types AS ABAP and BI) and the Java stack (for usage types AS Java, EPC, EP, and BI Java). You need BI Java including EP if you want to use the new SAP NetWeaver 7.0 (2004s) integrated planning functionality⁶. Without BI Java and EP, you can only use the SAP NetWeaver BI 3.5 (previously SAP BW 3.5) Business Planning and Simulation functions in combination with the 3.x runtime.

Please note the following landscape aspects for BI-based scenarios:

- There is a 1:1 relationship between EP and BI:
 - BEx Web Java Runtime only reads Web Template from one system – this limits Java to ABAP to one
 - BEx Information Broadcasting supports only one Portal – this limits ABAP to Java to one
 - You can install BI and BI Java together in one system, as well as in separate systems.
 - Each BI Java usage type requires EP (see above), but there may be additional portals in the landscape without BI.
- For BI, you should reserve a dedicated client for reporting purposes. In this client, activate SAP NetWeaver 7.0 (2004s) BI Content Add-On as required.
- BI scales with AS ABAP.
- BI Java scales with AS Java.

There are different options where to run BI in your SAP NetWeaver landscape. You can install BI and BI Java together in one system (one system ID), as well as in separate systems (two system IDs). In addition, BI can be optionally combined with other usage types. Each option has different advantages and disadvantages. You can use the following points as a basis for your decision making:

⁶ BI Java also adds other new functionality in addition to integrated planning.

- ABAP+Java dual stack installation with one system ID

This is the cheapest way to run BI. This option is recommended for small and midsize systems.

| Factor | Explanation |
|----------------|---|
| + Cost | You do not have to operate and maintain an additional server (although the single server cost could be prohibitive). |
| + Availability | Availability is the same as all components are started and stopped at the same time |
| – Scalability | Server itself must scale; split is not possible |
| + Effort | The system is patched and upgraded as a whole. Since BI and BI Java are usually both required for a scenario, a single system is easier to maintain. Although, if other usage types are also combined, this can reduce flexibility. |

- Two standalone installations on a single server with two system IDs

Recommend for small and midsize systems with potential growth.

| Factor | Explanation |
|------------------------|---|
| – Cost | You have to operate and maintain an additional server. |
| – Availability | The systems can be stopped and started individually. For example, if you plan the downtime of one stack, you have to consider the availability requirements of the other as well. |
| + Scalability | Each server can scale itself. |
| – Release dependencies | Since BI and BI Java are usually both required for a scenario, a distributed system is more difficult to maintain. |

In addition to the option whether to install one or two systems, there is another important option regarding your system landscape. You can install one or two portals depending on your user groups.

A Federated Portal Network (FPN) is a network of independent portals which share content. Each portal can use content exposed by another producer portal, but at the same time expose its own content for use by other consumer portals. Further information about FPN can be found later in this paper.

You can either use an SAP NetWeaver 2004 portal or an SAP NetWeaver 7.0 (2004s) system with usage type EP as a central portal. Be aware that different methods are used to connect BI capabilities of SAP NetWeaver to the central portal:

- To connect an SAP NetWeaver 7.0 (2004s) system with usage type BI Java to an SAP NetWeaver 7.0 (2004s) system with usage type EP, you use the federated portal feature remote role assignment.
- To connect the following systems, you use application integrator iViews:
 - SAP NetWeaver BI 3.x (previously SAP BW 3.x) to SAP NetWeaver 2004 portal
 - SAP NetWeaver BI 3.x (previously SAP BW 3.x) to SAP NetWeaver 7.0 (2004s) usage type EP

- SAP NetWeaver 7.0 (2004s) usage type BI Java to SAP NetWeaver 2004 portal

Note the following recommendation:

- Two portals – One portal is used for BI only and one portal is used as Enterprise Portal.

| Factor | Explanation |
|------------------------|--|
| + Users | Different user groups (in general Enterprise Portal user group is larger than BI user group) |
| – Integration | Two portals could be integrated with Federated Portal. |
| + Release dependencies | BI Portal (BI Java) could be synchronized with BI (Enterprise Portal could be independently upgraded). |

- One portal – One portal is used for both BI portal and Enterprise Portal⁷

| Factor | Explanation |
|------------------------|---|
| – Users | Common user group (identical users) |
| + Integration | Implicit |
| – Release dependencies | BI might require upgrade or application of a Support Package of the one portal (combined Enterprise Portal and BI Portal) because BI is closely related to BI Java (BI portal). |

Usage Type DI

DI requires AS Java as a prerequisite in the same system. Optionally, it can be combined with other usage types in one system. Depending on the scenario, EP and AS ABAP may be optionally used. Please note the following landscape aspects for DI based scenarios:

- SAP strongly recommends that Java Web Applications are integrated in Enterprise Portal (EP) on the UI level. For this, you require an EP system in your landscape (SAP Enterprise Portal 6.0 SP2 or higher).
- Optionally, you can install DI and EP in two separate systems (one system with DI and AS Java, one system with EP and AS Java).
- AS ABAP is optional. You require it if you want to use business workflow. You can install it either in one system with DI and EP or as separate system.
- The functions of DI may run in a single server or be distributed across multiple servers as necessary.
- In Java development (Web Dynpro), you can only connect to one ADS installation at a time.
- Development Infrastructure inherits the high availability from AS Java and scales with AS Java

There are different options where to run DI in your SAP NetWeaver landscape. You can install DI in one system (one system ID), as well as in separate systems (two system IDs): one DI configured for the Design Time Repository (DTR) and Change Management Service (CMS); and one DI configured for the Component Build Service (CBS). To scale, the following recommendation is given: enhance each NWDI as necessary by adding CBS/DTR servers. In addition, DI can be optionally combined

⁷ This option is only possible with a single BI system. With two or more BI systems this will not work.

with other usage types. Each option has different advantages and disadvantages. You can use the following points as a basis for your decision making:

- Java only installation with one system ID

This is the leanest way to run DI.

| Factor | Explanation |
|------------------------|--|
| + Cost | You do not have to operate and maintain an additional system (although with larger systems, the single server cost could be prohibitive). |
| + Availability | Availability is the same as all components are started and stopped at the same time. |
| – Scalability | Although the server itself can scale, it is not possible to individually separate and scale the CBS. |
| + Release dependencies | The system is patched and upgraded as a whole. A single system is easier to maintain. Although, if other usage types are also combined, this can reduce flexibility. |

- Two standalone installations on a single server with two system IDs

| Factor | Explanation |
|------------------------|--|
| + Cost | You do not have to operate and maintain an additional server (although with larger systems, the single server cost could be prohibitive). |
| – Availability | The systems can be stopped and started individually. For example, if you plan the downtime of one application, you have to consider the availability requirements of the other(s) as well. |
| + Scalability | Server itself can scale or split to two servers possible. |
| – Release dependencies | A distributed system is more complex to maintain. |

- Two standalone installations on a two servers with two system IDs.

Recommended for large systems

| Factor | Explanation |
|------------------------|--|
| – Cost | You have to operate and maintain an additional server. |
| – Availability | The systems can be stopped and started individually. For example, if you plan the downtime of one application, you have to consider the availability requirements of the other(s) as well. |
| + Scalability | Each server can scale itself. |
| – Release dependencies | A distributed system is more complex to maintain. |

Usage Type EP

EP requires AS Java and EPC as a prerequisite in the same system. Optionally, it can be combined with other usage types in one system. Please note the following landscape aspects for EP based scenarios:

- A federated portal network is a network of independent portals which share content. Each portal can use content exposed by another producer portal, but at the same time expose its own content for use by other consumer portals.
- You can either use an SAP NetWeaver 2004 Portal or an SAP NetWeaver 7.0 (2004s) system with usage type EP as a central portal. Be aware that different methods are used to connect BI capabilities of SAP NetWeaver to the central portal:
 - To connect an SAP NetWeaver 7.0 (2004s) system with usage type BI Java to an SAP NetWeaver 7.0 (2004s) system with usage type EP, the federated portal features Remote Role Assignment, Remote Delta Links and Remote Application Integration (currently only for BI reports) may be leveraged, depending on the administrative scenario.
 - To connect the following systems, you use application integrator iViews:
 - SAP NetWeaver BI 3.x (previously SAP BW 3.x) to SAP NetWeaver 2004 Portal
 - SAP NetWeaver BI 3.x (previously SAP BW 3.x) to SAP NetWeaver 7.0 (2004s) usage type EP
 - SAP NetWeaver 7.0 (2004s) usage type BI Java to SAP NetWeaver 2004 Portal
- Enterprise Portal inherits the high availability from AS Java and scales with AS Java.

There are different options where to run EP in your SAP NetWeaver landscape. You can install a single EP for the entire landscape (centralized EP infrastructure), or multiple EP usage types for each scenario (Federated EP infrastructure). Each option has different advantages and disadvantages. See the section “Special Case: Federated Portal Network” for a comparison.

Usage Type EPC

As of SAP NetWeaver 7.0 (2004s) Service Release 2 (SR 2), usage type Enterprise Portal (EP) has been separated into two closely related usage types: EP Core (EPC) and Enterprise Portal (EP).

- Usage type *EP Core* covers the core portal capabilities (for example portal runtime and administration)
- Usage type *EP* now includes all the portal add-on capabilities (for example Knowledge Management, Collaboration, Visual Composer)

A lean entry point and runtime for applications (EPC) can now be implemented. This allows more flexibility to use SAP NetWeaver in scenarios, where only basic portal capabilities are required (for example no knowledge management, no collaboration, no Visual Composer capabilities are needed). The result is improved overall performance, reduced memory consumption and TCO.

The main use cases for EPC are:

- Application Runtime for ERP C-Roles:
 - ERP provides starting with SAP ERP 6.0 Enhancement Package 1 especially developed PFCG roles as a repository for the complete SAP ERP functionality. These PFCG roles will be uploaded to a portal via the Role Upload tool and presented in a “Best Fit” way. The application runtime for these uploaded roles shall be a lightweight portal with improved performance, reduced memory consumption and reduced TCO.
- Portal Platform for ERP Work Centers:
 - ERP provides with SAP ERP 2005 SP8 changed versions of all Work Centers where the usage of KMC capabilities has been removed. These Work Centers have to run on the

same lightweight portal as the uploaded C-roles described in high level requirement “Application Runtime for ERP C-Roles”.

- SAP NetWeaver Standalone
 - Although basically intended for ERP usage, standalone SAP NetWeaver can benefit from a lean portal implementation, saving installation/administration, and so on costs for those Customers that only require pure Portal use (no KM, or Collaboration, or VC, ...)

Special Case: Federated Portal Network

The distributed nature of systems based upon service-oriented architecture may call for multiple portals, or multiple portal runtimes, within the landscape. For example, a global organization may have multiple regional portals. Or functionality (ERP, CRM, and so on) may be developed and implemented in separate portals based upon this functionality. A federated portal network (FPN) allows organizations with multiple portals, SAP and non-SAP, to share content between the portals. By implementing a federated portal network and sharing content between portals, organizations can provide users at each location with a single portal access point. From each portal configured as an access point, the users are able to access information, services and applications distributed on portals throughout the entire organizational network.

Within SAP NetWeaver 7.0 (2004s) SPS 10 and above FPN is available, which allows to set up different portal landscapes.

Like in previous releases an important landscape option is to create one central portal. All users connect to this portal and all content and connections to backend systems are available there. If this architecture is possible and suits the organizational and technical requirements, it offers a lean setup and central administration.

However, if this architecture does not serve the needs of the company, the federated portal network is a configuration option that allows the creation of more complex landscapes in which portals can share content:

- The most common setup for a federation would be setting up one central consumer portal⁸ serving as the entry point for all end users in the organization. It can link to content residing on multiple producer portals⁹ - they might contain portal content, access to backend systems and be administered by different authorities.
- Multiple consumer portals can be set up as access points for different groups, for example if there are different autonomous subsidiaries in the corporation. If some content should be provided centrally, for example by headquarters, a central producer portal can be set up providing content for various consumers. Thus, the content does not have to be copied to individual portals due to organizational requirements, but is provided only once centrally.
- In theory, a full network of portals can be defined, that means multiple consumer portals interacting with multiple producer portals. However, it is recommended to keep the landscape as “light” as possible, since the administrative effort will increase with increasing complexity of the federation.
- Of course, if there are multiple isolated portals within the organization that don’t need to be integrated for business reasons, there is no requirement to set up a federation. It is still a valid setup to run multiple portals without any connection between them.

⁸ Consumer: in a federated portal network, a portal that accesses and uses remote content provided by other producer portals in the network.

⁹ Producer: in a federated portal network, a portal that exposes its content for consumption by other consumer portals in the network.

Currently only direct producer – consumer relationships are supported. “Transitive content sharing” or a multi-level federation such as: producer → consumer/producer → consumer is currently not supported.

Overall, the Federated Portal Network allows the sharing of content between portals. The content is deployed only once, but users from other portals can also access this content. It is not used for scenarios such as synchronizing portal content or transporting content to various instances.

The Federated Portal Network supports 3 content sharing modes:

- Remote Role Assignment: Assigning remote roles to users directly → available since SAP NetWeaver 7.0 (2004s) SPS09 through standard UME role assignment tools (only for SAP NetWeaver portals).
- Remote Delta Links: Copy content and paste as delta link from remote portals to the local Portal Content Directory of the consumer → adjust and integrate this content, for example iViews / pages, and so on, on the consumer portal. Available since SAP NetWeaver 7.0 (2004s) SPS10 (only for SAP NetWeaver portals).
- WSRP content sharing: In order to integrate standard-compliant content coming from Non-SAP portals, a new iView template is available in the Portal Content Studio. Consuming and producing content is supported since SAP NetWeaver 7.0 (2004s) SPS09.

The two content sharing modes Remote Role Assignment and Remote Delta Links are appropriate for different business scenarios. The following approaches can be used:

- “Bottom-Up-Approach”: Autonomous units create and maintain content on their own producer portals. They build fully configured roles, which should be integrated into a central consumer portal with only little central administration effort. This approach is supported by Remote Role Assignment.
- “Top-Down-Approach”: Due to various reasons the central department decides to separate content to individual producer portals (for example XSS applications are deployed onto a separate producer portal). The administration effort on the producer portals is minimized, since the content is only deployed to the producer and adjusted and maintained on the consumer portal. For this approach Remote Delta Links would be the right choice.
- Mixed scenarios: Especially for portal federation mixed scenarios can also be considered. Autonomous portal content is created in a portal; however, this will be reused and adjusted on other consumer portals. Here Remote Delta Links or Remote Role Assignment (using role merging) might be suitable depending on the specific use case.

Thoroughly analyze which content sharing mode to be used in which scenario and how the overall administration concept in the federation should look like. It will have impacts on the administration effort required on consumer and producer portals.

The following table summarizes the aspects of the options:

| | Remote Role Assignment | Remote Delta Links |
|----------------------------------|---|--|
| Shared Content | 0 Roles | + iViews, pages, worksets, roles |
| Administration effort/TCO | – Producer: create and maintain portal content and applications | 0 Producer: create and maintain portal content and applications (could be minimized) |
| | + Consumer: assign users to roles | 0 Consumer: copy portal content, adjust and integrate portal content |

| | | |
|--|---------------|---|
| Modification possible on consumer | – Merge roles | 0 Adjust content and integrate into local content |
|--|---------------|---|

In order to make the decision whether a federated portal network would be an appropriate portal landscape, thoroughly evaluate the pros and cons of this solution. One central productive portal implementation causes a lot less administrative effort than a federated landscape with multiple portals, which have to be monitored, administered and configured. However, a federated portal network can offer other benefits that can make it desirable: multiple portals exchanging content running on different releases, SPS levels and service-level agreements. Moreover, business units can own autonomous portals and create content (the delegated content administration sometimes is too restricted for organizational requirements). Thus, decide between the tradeoff of whether organizational or technical requirements demand a federated portal landscape versus one central portal.

You can use the following points as a basis for your decision making:

- Central landscape (1 portal)

| Factor | Explanation |
|------------------------------|---|
| + Administration effort | One portal to be administered |
| – Release dependencies | Requires maintenance coordination across scenarios. Individual flexibility for scenarios is reduced: all scenarios are patched, upgraded, and so on together. |
| 0 Testing | All functionality tested in one portal |
| 0 Autonomy of business units | Typically central IT department plays leading role, delegated administration concept |
| – Availability | Single portal determines availability and SLAs |

- Federated Portal Network (2-n portals)

| Factor | Explanation |
|--------------------------------------|---|
| – Administration effort | Administration performed in multiple portals with different scopes |
| + Release dependencies ¹⁰ | Upgrades performed on different portals independently and is aligned with application system it is designed for |
| 0 Testing | Integration and landscape tests required |
| + Autonomy of business units | Autonomous content, application creation and maintenance |
| 0 Availability | Availability requirements can vary, but consumer portal must remain available as a user entry point |

¹⁰ In general, the central consumer portal should be on the highest level in the landscape to ensure smoothest operation. See the restrictions at http://help.sap.com/saphelp_nw70/helpdata/en/45/76322bf0020dece1000000a11466f/frameset.htm.

Usage Type PI

PI requires AS ABAP and AS Java as prerequisites in the same system. Optionally, it can be combined with other usage types in one system.

Nevertheless, we recommend that you have a dedicated PI system. For compatibility of the PI/XI release with the release level of connected systems in the landscape, see SAP Note 103047. For example, SAP XI 3.0 is compatible with SAP NetWeaver 6.40 and 7.0 application systems, since the same XI protocol is used. If you want to upgrade or install an application in your system landscape, you first have to make sure that the PI system is on a compatible release level. If required, you have to upgrade the PI system first to the new or a higher release. With a dedicated PI system, this can be accomplished with a minimum of downtime. Also, the PI system would not be affected by the downtime of other usage types running in the same system.

Please note the following landscape aspects for PI based scenarios:

- You can use the J2EE Adapter Engine (PI/XI) that is part of your PI system as a central J2EE Adapter Engine (PI/XI). Optionally (for performance reasons), you can also install a non-central J2EE Adapter Engine (PI/XI) separately as a system with AS Java and parts of the usage type PI on a separate host.
- J2SE Adapter Engine (PI/XI) can be installed in a non-SAP Java environment. However, it only hosts a subset of the adapter functionality, is only supported for compatibility reasons, and should only be used if this is a precondition in your environment.
- We recommend that you run the business process engine (that is, the PI system) and the workflow engine (that is, the AS ABAP system) in two different systems.
- We recommend that you run the application or service provider (that is, the AS ABAP system) and the integration broker (that is, the PI system) in two different systems.
- PI requires a dedicated client, due to special PI requirements concerning the client pipeline. Whereas application client pipelines only send and receive messages, the central PI pipeline can also map messages to different formats before sending them.
- The default installation variant for a Process Integration system is a single host installation; here all the central components, namely the central Integration Server, Integration Builder, and System Landscape Directory (SLD) are installed on one host.
- Process Integration scales with AS Java and AS ABAP.

There are different options where to run PI in your SAP NetWeaver landscape. You can install PI as a standalone system, or combined with other usage types. However it is strongly recommended to run PI as a standalone system. You can use the following points as a basis for your decision making:

- PI standalone system

| Factor | Explanation |
|------------------------|--|
| + Release independency | Usage types are not affected by the upgrade of your PI system, as PI is backward-compatible. |
| + Flexibility | As there are no direct interdependencies between the PI system and other usage types, it is easier to plan and perform changes – for example, it is easier to plan the downtime of an application system, as you do not have to consider PI availability requirements. |

– Cost You have to operate and maintain an additional system.

- Running PI integrated with other usage types¹¹

| Factor | Explanation |
|----------------------|---|
| + Cost | You do not have to operate and maintain an additional server (although with larger systems, the single server cost could be prohibitive). |
| – Availability | Availability is the same as of the application, and the systems cannot be started and stopped individually – if the application system is up, PI is also available. If the system is down, PI is not available. |
| – Release Dependency | Applications are affected by the upgrade of PI and vice versa. |
| – Flexibility | PI depends on the application and vice versa – for example, if you plan the downtime of an application system, you have to consider PI availability requirements as well. |

Usage Types AS ABAP and AS Java

AS ABAP and AS Java provide the technical foundation for all the scenarios and their installation options are parts of the scenarios that they support. For the scenarios that use Adobe Document Services (ADS) there are the following considerations:

Landscape Considerations for ADS

- Using One ADS Installation from Several Systems
To optimize the use of your system landscape, it is possible to connect several ABAP-based SAP systems to one SAP NetWeaver AS Java running the ADS for the generation of SAP Interactive Forms output formats. In general, this setup is suitable for the use of SAP Interactive Forms in interactive scenarios.
- Ensure that the system running the ADS has at least the Support Package level of the systems connected to it.
- Adobe document services currently do not run on all platforms supported by SAP NetWeaver.
- For more information, see SAP Note [925741](#).

SAP NetWeaver Master Data Management (MDM) – Additional Capability

As a standalone engine, MDM provides a central repository of master data and catalog functionality. An MDM repository utilizes a database to store master data and associated related information that consist of single primary table and numerous support tables. The additional tables are used to define characteristics, taxonomies, and hierarchies as references to entries in the main table.

Rich clients are supported on Microsoft Windows systems. Maintenance functions are available through the Enterprise Portal usage type. Several APIs are available to connect to SAP and non-SAP systems. The following notes should be taken into consideration:

- Multiple language support must be considered when sizing the system.
- Usage type PI required for communications to some remote systems.

¹¹ Each application system can only be linked to one PI system; it is not possible to connect against a local PI and against a central one.

- BI direct connection supported as of MDM 5.5 SP05.
- Portal business package available for user access to the MDM repositories.
- COM and Java APIs available for connections to non-SAP systems.
- Plug-ins need to be installed to utilize MDM extractors.
- System will scale with master and slave server installations.
- MDM Web Services available via the Java API (AS Java usage type required).

Although technically not MCOD, MDM can be in a technical manner be installed with other SAP NetWeaver usage types in an existing database with other SAP NetWeaver Usage Types. Repositories are loaded into memory on the server. For this reason, MDM servers with large repositories and/or multiple languages, it is strongly recommended that MDM be installed on a standalone system with a dedicated database.

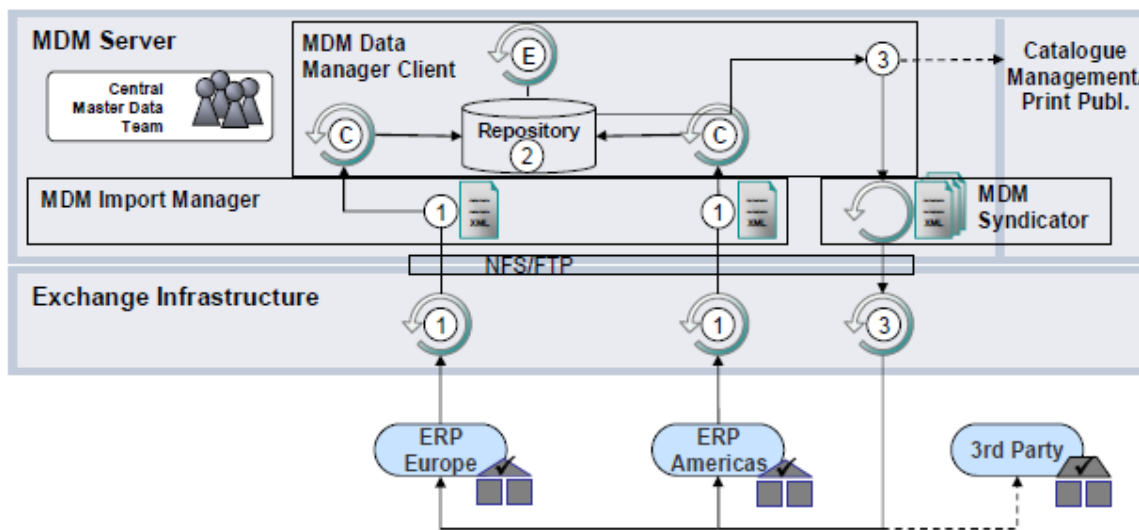


Figure 10: MDM Architecture

- MDM standalone system

| Factor | Explanation |
|------------------------|---|
| + Release independency | Usage types are not affected by the upgrade of your MDM system. MDM is backward-compatible. |
| + Flexibility | MDM not affected by database maintenance. |
| + Performance | No memory competition with other applications yields greater performance. |
| - Cost | You have to operate and maintain an additional system. |

- Running MDM on a system with usage types

| Factor | Explanation |
|---------------|---|
| + Cost | You do not have to operate and maintain an additional server (although with larger systems, the single server cost could be prohibitive). |
| -Flexibility | Shared database outages affect the MDM system. |
| - Performance | MDM performance can drop due to a lack of memory resources. |

www.sdn.sap.com/irj/sdn/howtoguides