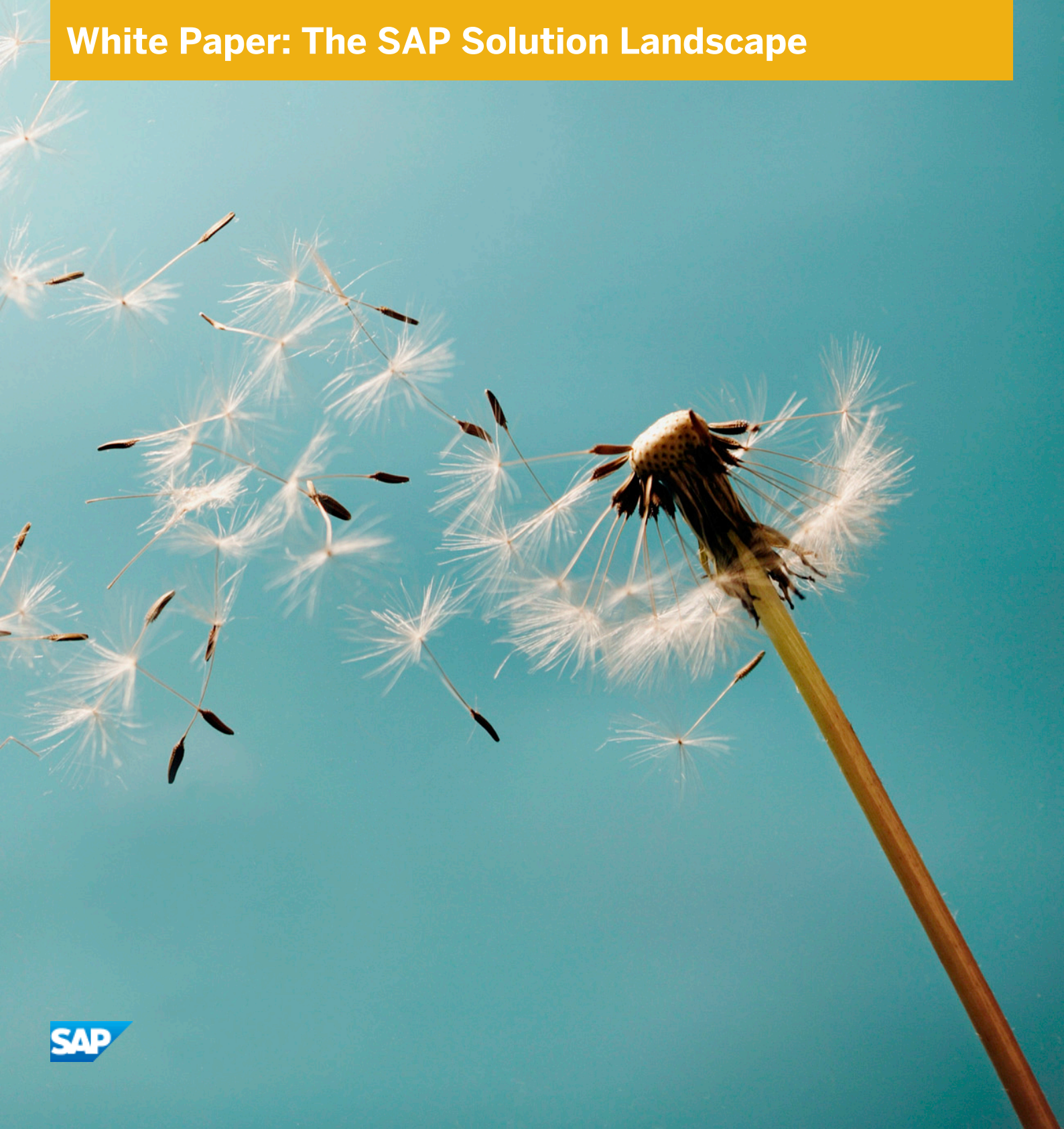


SAP Solution Manager
Document Version: 2.0 – 2016-10-04

White Paper: The SAP Solution Landscape



Content

1	About This Document.	3
1.1	SAP Solution Manager 7.1 and 7.2.	3
2	Product and Application Lifecycle.	5
3	Product Model and Landscape Model.	6
3.1	SAP Product Model.	7
3.2	Customer Landscape Model.	9
4	Usage of Entities in the Customer Landscape.	13
5	Logical and Technical Landscape.	15
6	Customer Landscape Management Activities.	17
7	Top-Down Structuring vs. Bottom-Up Guessing.	19
8	Setting Up a Landscape.	20
9	Summary.	23

1 About This Document

You find the latest version of this document at <http://go.sap.com/documents/2015/07/70621987-5b7c-0010-82c7-eda71af511fa.html>.

i Note

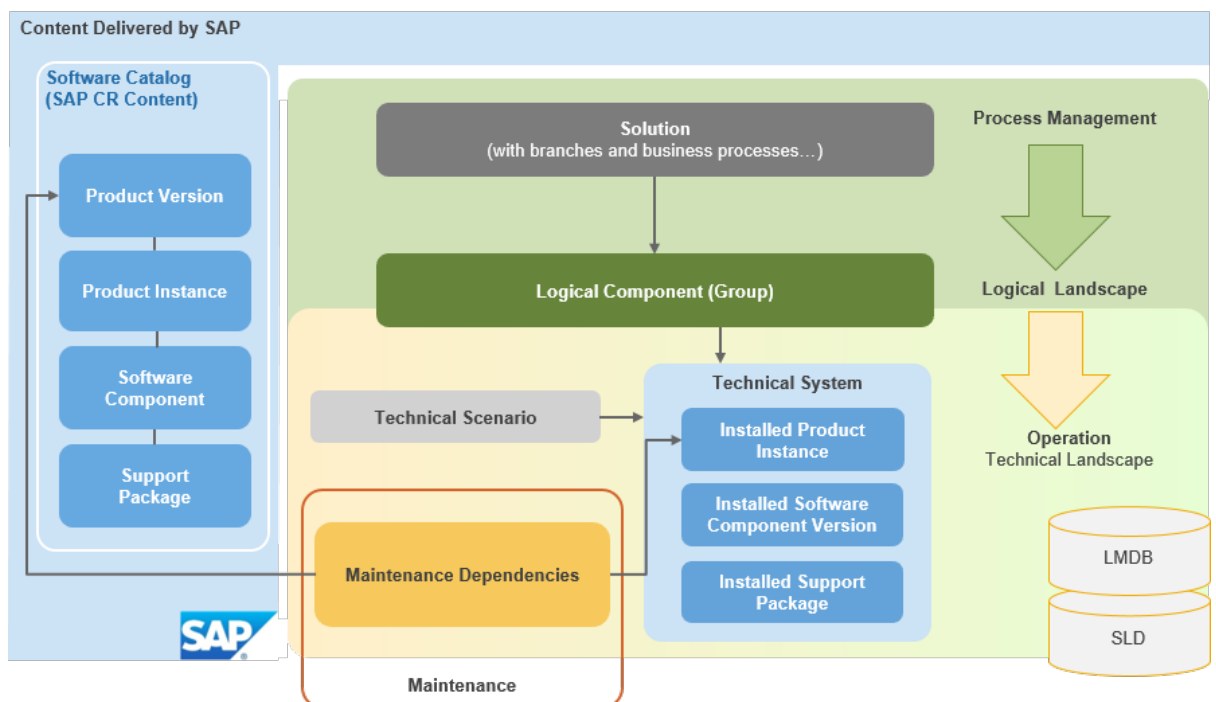
This document is valid only for release 7.1 of SAP Solution Manager.

Only the next chapter ([SAP Solution Manager 7.1 and 7.2 \[page 3\]](#)) briefly explains the basic changes in SAP Solution Manager 7.2. In this release, the new solutions model has been changed and simplified, so that this guide is no longer required.

1.1 SAP Solution Manager 7.1 and 7.2

In SAP Solution Manager 7.2, the solution concept has changed and many simplifications were established.

- The **product model** remains unchanged.
- The **landscape model** can be simplified, in comparison to the 7.1 landscape model you see under [Product Model and Landscape Model \[page 6\]](#):



- The **process management** was fundamentally changed. Main change: Logical components now have much less dependencies, for example, they are no longer used for SAP EarlyWatch Alerts and service sessions. They also do not need to be created in the landscape

management database (LMDB) any longer, to model maintenance dependencies in product systems. Logical components have moved to focus on the process management in the solution documentation. Logical components and logical component groups are documented in detail inside the solution documentation area. This affects the following areas:

- Solution documentation
- Implementation
- Test management
- Change management
- Quality gate management
- Business process monitoring and alerting

For more information, see:

- <http://help.sap.com/solutionmanager72> ▶ *Application Help* ▶ *SPS 3 or higher (select language)* ▶ *Process Management* ▶ *Solution Administration* ▶ *Basic Terms and Concepts* ▶
- [Solution Documentation Content Activation Guide](#) ▶

- **Maintenance management:** Maintenance planning and transactions are no longer prepared with the maintenance optimizer in SAP Solution Manager, but with the maintenance planner in SAP Support Portal. For more information, see <http://help.sap.com/maintenanceplanner>.

- Application operations:

- Root Cause Analysis
- Technical monitoring and alerting

For more information, see [SCN Wiki: Application Operations](#) ▶.

- Incident Management

For more information, see <http://help.sap.com/solutionmanager72> ▶ *Application Help* ▶ *SPS 3 or higher (select language)* ▶ *IT Service Management* ▶.

- **Service planning and SAP service delivery** has been simplified and now uses the technical landscape directly.

For more information, see <http://help.sap.com/solutionmanager72> ▶ *Application Help* ▶ *SPS 3 or higher (select language)* ▶ *SAP Engagement and Service Delivery* ▶.

For more information on SAP Solution Manager 7.2, check the provided links on this page.

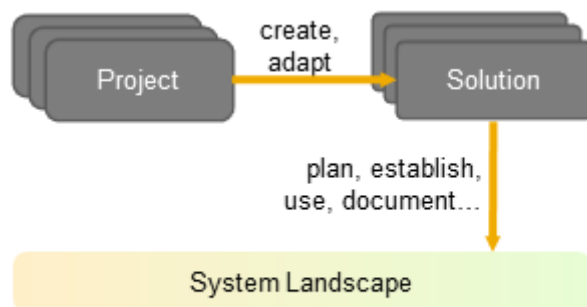
The following chapters in this document apply only to SAP Solution Manager 7.1.

2 Product and Application Lifecycle

To plan, configure, and operate SAP solution landscapes, you use several concepts of application management, technical operations and software logistics. For a general understanding the clarification of terms and definitions is crucial.

Two processes support the key lifecycle management:

- The **product lifecycle** is concerned with the design, development, production, and shipment of products. Besides product functionality, content and documentation, product lifecycle management provides a precise description of all components and their dependencies. Components are, for example products, product versions, product instances, software components, and hardware (including third-party vendors).
- The **application lifecycle** guides the enterprise from planning to the implementation and operation of a solution. A solution comprises all elements from business process to technical landscape. Implementation and maintenance **projects** help customers to organize the creation and adaption of these solutions.



Both lifecycles are based on an underlying **SAP product model**, which defines product-related data, such as the product structure, versioning, and technical dependencies between product instances and other products.

The **landscape model** defines the system and solution landscapes, consisting of the technical systems, logical components, and the solution itself.

3 Product Model and Landscape Model

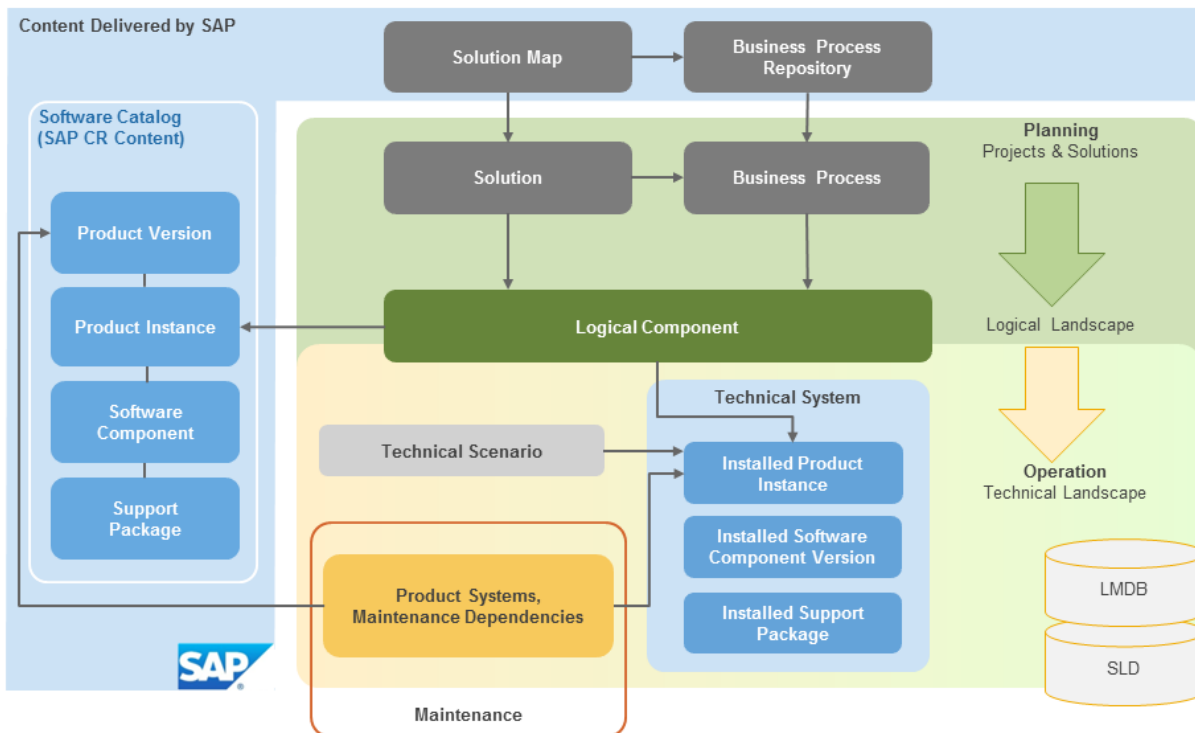
Terms like solution map, business process, products, and software components are used when talking about the SAP portfolio. This section describes these terms and their relationships.

A **Solution Map** helps customers to identify and compose their business process scenarios. It only includes business processes; there is no reference to technical systems or product names.

The **Business Process Repository** describes SAP process models of SAP solutions, with reference to products and their versions. **Processes** contain **business process steps**. A business process description is a high-level abstraction of the underlying landscape.

A **logical component** is an abstraction of a software feature, for example it represents the functionality of a CRM application server ABAP. It groups technical systems with this functionality and with different roles, such as development, test, and production systems. To be executed, a business process step uses logical components. Logical components link the abstract software feature definition to its technical implementation.

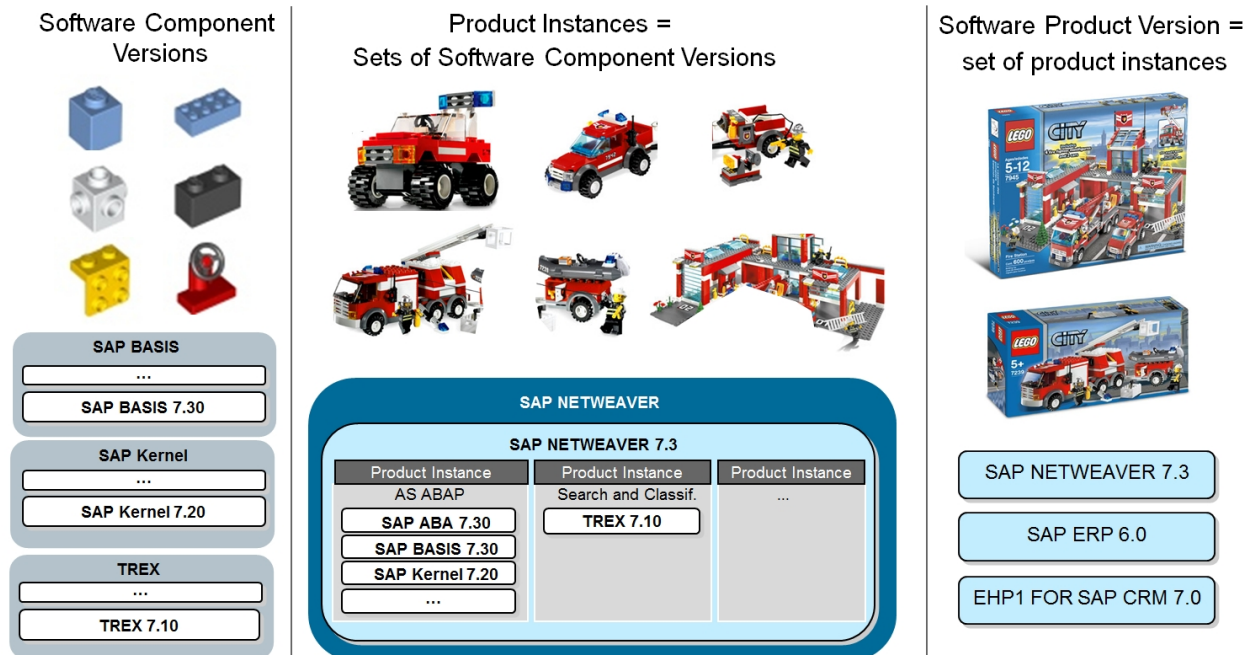
The following graphic is valid for SAP Solution Manager 7.1:



3.1 SAP Product Model

At a high level, the SAP software portfolio consists of products. Each product is a unit of license contract and shipment. When you set up the customer landscape, it is important to understand the entities of which a product consists:

- **Products** are the applications SAP delivers. **They group a set of functions to solve business needs.** The SAP product model describes all SAP products. This model defines the possible product structure. Examples: SAP ERP, SAP NetWeaver, and SAP Solution Manager.
- A **product version** is a release of a product. There can be several versions of a product. It **bundles the product instances and their software component versions** that are available at the same time for a certain function. The bundle reflects, for example, technical dependencies. Product version names typically start with "SAP..." and have characteristics like their application function and a maintenance period. In this paper, we refer to "product versions". Some other documents say just "products", for simplicity. Example: SAP ERP 6.0 and SAP ERP 2004 are versions of the product SAP ERP.
- Each **product instance** belongs to one product version. It **bundles a set of software component versions** that must be installed and updated together, to ensure a consistent state. (Although patches can be applied to individual software components.)
A product instance can contain other product instances and be re-used across product versions. A product instance (and the software component versions it contains) is always installed on one technical system and cannot be distributed among different technical systems. (An ABAP-Java dual-stack is **one** technical system; one product instance can contain both ABAP and Java software components.)
Examples: SAP ECC Server, SAP XSS (Self-Services), and TREX.
- **Software components versions** are the smallest entities that can be installed on a technical system. A software component version can be in different product instances. It can depend on other software component versions that need to be installed with it.
There are smaller shippable software entities than software component versions, such as support packages, patches and notes, which always refer to a software component version.
Examples: SRM Server 7.01, SAP BASIS 7.20, and Portal 7.0.



LEGO® is a trademark of the LEGO Group of companies, which does not sponsor, authorize or endorse this site. ©2012 The LEGO Group. All rights reserved.

Parts of the software catalog can be accessed by SAP customers in the [Product Availability Matrix](#) on SAP Service Marketplace (see figure 'Example of relationships of product version SAP ERP 6.0 in the PAM' in the next chapter), or in the [SAP Software Download Center](#).

The SAP CR content is also used to describe non-SAP products like SAP-Branded Resellers (SBR), Vendor Branded resellers (VBR) and Independent Software Vendors (ISV). The parts of the PPMS content that are relevant for SAP customers are published regularly, and synchronized with SAP Solution Manager. Technically, SAP Solution Manager receives SAP CR content in the following ways:

- SAP CR content per SLD synchronization (see SAP Note [669669](#))
- Support package shipment (software components ST-ICO and ST)
- XML attachment to SAP Notes
- Note/XML as part of Business Process Repository (BPR) and SLD

Within customer landscapes, the SAP CR content is persisted in the following repositories:

- In each SLD
- In SMSY in SAP Solution Manager 7.0
- In LMDB as of SAP Solution Manager 7.1

Products and business processes are essential to maintain the solution landscape, i.e. to plan and execute upgrades, updates etc. So it is important to have the latest information on products and processes.

3.2 Customer Landscape Model

A customer landscape is the description of all the landscape elements and their relation to each other.

To describe the landscape, entities from the SAP product models added by the following entities are used:

- **Solutions** are the core of SAP Solution Manager planning and the top of the solution landscape hierarchy. They are used to deliver SAP services, and for application lifecycle management and business process operations by customers. Planning, reporting and maintenance activities use this level. Implementation and maintenance projects adapt and evolve the solution.

To implement a business process, you first assign the business processes and applications (logical components) to a solution. Solutions group business processes and business process interfaces, with reference to logical components.

In SAP Solution Manager, a project describes the grouping and organization of business tasks, technical tasks, and organizational tasks during implementation and maintenance of SAP software.

- **Logical components** describe a function, and group the systems providing this function in roles. They are used, for example, by business process planners and software administrators, who are typically not concerned with the technical details and layout of the system implementation.
 - With a logical component you can refer to a high level view of an application, to plan and maintain it.
 - Business process descriptions refer to logical components.
 - Logical components are sets of technical systems with different roles that are synchronized during change processes, for example **development, test, and production system**.
 - They are the link between the business and technical perspectives of a landscape, and **connect the technical system landscape and the planning landscapes**.
 - A logical component can group technical systems that perform a business process (step).
 - To plan the system landscape, a logical component can already be defined without a technical system assignment. In this case, you only assign a product instance.

With logical components, you can cascade down the following entities:

► *Business Process* ► *Business Process Step* ► *Logical Component* ► *Technical System* ►

- **Technical systems** are the central element of the solution landscape in which software components are deployed, and for which operational activities, such as monitoring and alerting are performed. A technical system is a piece of software installed on one or more physical hosts. Technical systems are the basis and environment of landscape element data. They are typically created by installation, and register themselves in SAP Solution Manager, via automatic data suppliers and the System Landscape Directory (SLD). Registered data can be enhanced manually in the LMDB (Landscape Management Database) of SAP Solution Manager.

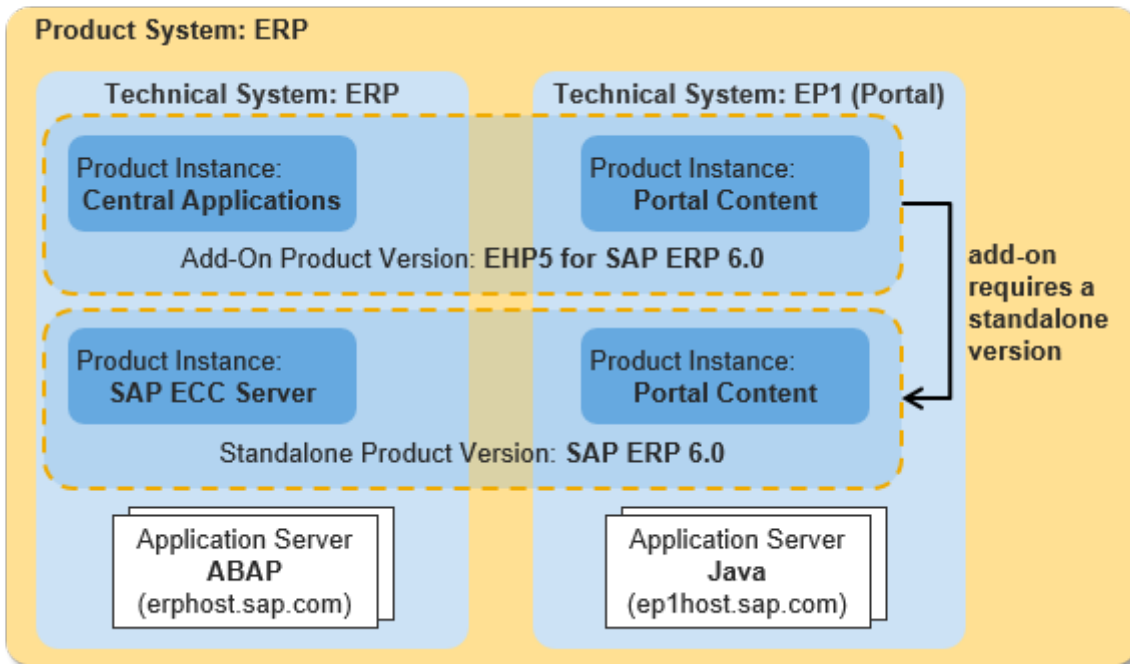
Technical systems are characterized by attributes such as SID, host name, installation number, and a technical system type (for example AS ABAP, AS Java, or TREX server). Technical systems are extended by logical systems that are views of the technical systems from different perspectives (e.g. ABAP clients).

→ Remember

Guiding Principle: Avoid manual data entry; use automatic data suppliers. Sometimes, no data supplier can be used, for example, for security reasons.

- **Product systems** bundle technical systems on which one or several product versions are installed. Note: Only one version of a product can be installed at one time on one technical system. For example it is not possible to install SAP ERP 6.0 and SAP ERP 2004 at the same time on one technical system. A product system is

needed for maintenance and planning. It defines dependencies between technical systems and their installed product instances. One technical system can be used in several product systems.



The example above shows an SAP ERP 6.0 product system with EHP5. There is more detailed description in the *Maintenance Planning Guide* at <https://wiki.scn.sap.com/wiki/x/gEW7Gg>. You can check the relationship of product versions in the Product Availability Matrix (PAM) at <http://support.sap.com/pam>. The following screenshot shows the product version SAP ERP 6.0 and its relationship to other product versions.

The screenshot displays the SAP Product Availability Matrix (PAM) for SAP ERP 6.0. The central focus is the 'SAP ERP 6.0' product version, which is highlighted with a dashed orange border. Four arrows radiate from this central box to four surrounding panels:

- Add-On Product Versions:** Lists various enhancement packages including EHP4 FOR SAP ERP 6.0, EHP4 FOR SAP ERP 6.0 / NW7.01, EHP5 FOR SAP ERP 6.0, EHP6 FOR SAP ERP 6.0, ENERGY CALC GAS CONSUMP 1.0, and FIREFIGHTER 4.0_700.
- Upgrade from:** Lists previous product versions such as SAP ERP 2004, mySAP ERP 2004, SAP FI-CA 4.64, SAP FI-CA 4.71, SAP FI-CA 4.72, and SAP HR-PS 4.62.
- Upgrade to:** States 'No successor product versions available'.
- Required Product Versions:** States 'No product versions required'.

The interface also features a top navigation bar with 'SAP Product Availability Matrix', user information 'Welcome Stephanie Janssen (Settings) External View', and 'SAP ERP 6.0'. Below the main content, there are tabs for 'General Information', 'Technical Release Information', and 'Details & Dates' (with 'Related Product Versions' selected). A footer contains links for 'FAQ', 'PAM Documentation', 'Feedback', 'Copyright', 'Impressum', 'Privacy Statement', 'Legal Disclaimer', and the URL 'https://service.sap.c'.

- **Technical scenarios** group technical systems. They are the primary working element for operations. For example, monitoring and alerting and services like the Early Watch Alert (EWA) focus on the technical system as their central object of interest.

A landscape model is more than a technical layout of landscape data. To interpret landscape data or change the configuration, you must take account of most of the model's core elements and their relations.

The following table shows the landscape model entities and their relationships. Note that crosses are only set for direct relationships.

		Solution	Project	Process	Logical Component	Product System	Product Version	Product Instance	Software Component	Technical System	Technical Scenario	Transport Route
Solution	uses:		X	X	X							
Project	uses:	X			X							
Process	uses:				X							
Logical Component	uses:							X		X		
Product System	uses:						X	X		X		
Product Version	uses:							X				
Product Instance	uses:								X			
Software Component	uses:									X		
Technical System	uses:							X	X			
Technical Scenario	uses:									X		
Transport Route	uses:				*					X		

* There is no check if the technical systems in a logical component are connected by transport routes. Technically, a transport route is not mandatory, but it is logical to have one.

4 Usage of Entities in the Customer Landscape

Landscape description plays a key role in several software logistic processes:

The entities in the customer landscape serve as navigation nodes for almost all processes in planning, maintenance, and operations. The landscape model and its nodes provide the pattern for navigation and element selection.

All SAP Solution Manager applications and use cases should be able to understand and interpret landscape information, even if **not all processes need all elements**. Although not all functional parts of SAP Solution Manager need all landscape entities for their tasks, each entity should serve uniformly as a navigation and grouping node, across all scenarios.

For example, Incident Management works based on technical systems because incidents are created for a selected technical system. Test Management works on a higher level, based on solutions, projects, and logical components, because tests are carried out for a group of technical systems that host the same business functions of a solution (for example, production, test, and development system).

The key question is: What needs to be part of a process and what landscape information does it need? Maintaining **all** information leads to bloated scenarios and unnecessary complexity. Maintaining **less** information decreases the options of a uniform landscape description.

Even worse, maintaining your own “universe” with entities and groupings to be independent will lead to extra maintenance effort and lack of conceptual understanding with the customer, undermining the intention of consistent and uniform landscape management.

The entities are used in several application lifecycle management (ALM) processes.

These processes are managed via the SAP Solution Manager work centers, which is illustrated in the following table. An “X” indicates a mandatory direct relationship; an “O” indicates an optional one.

entity \ work center in which the entity is used	Solution Documentation	Implementation	Test Management	Change Management	Quality Gate Management	Incident Management	Root Cause Analysis	Technical Monitoring & Alerting	Business Process Monitoring & Alerting	Job Management	Maintenance Management	Service Planning & Service Delivery
Solution	X		X			O			X		O	X
Project		X	X	X	X	O					O	O
Process	X	X				O			X			O
Logical Component	X	X	X	X	X				X	X	O	X
Product System											X	
Technical System						X	X	X				
Technical Scenario								X		X		
Transport Route				X	X							

For more information, see [SAP Solution Manager Processes](#) and the SCN Wiki [AppOps](#).

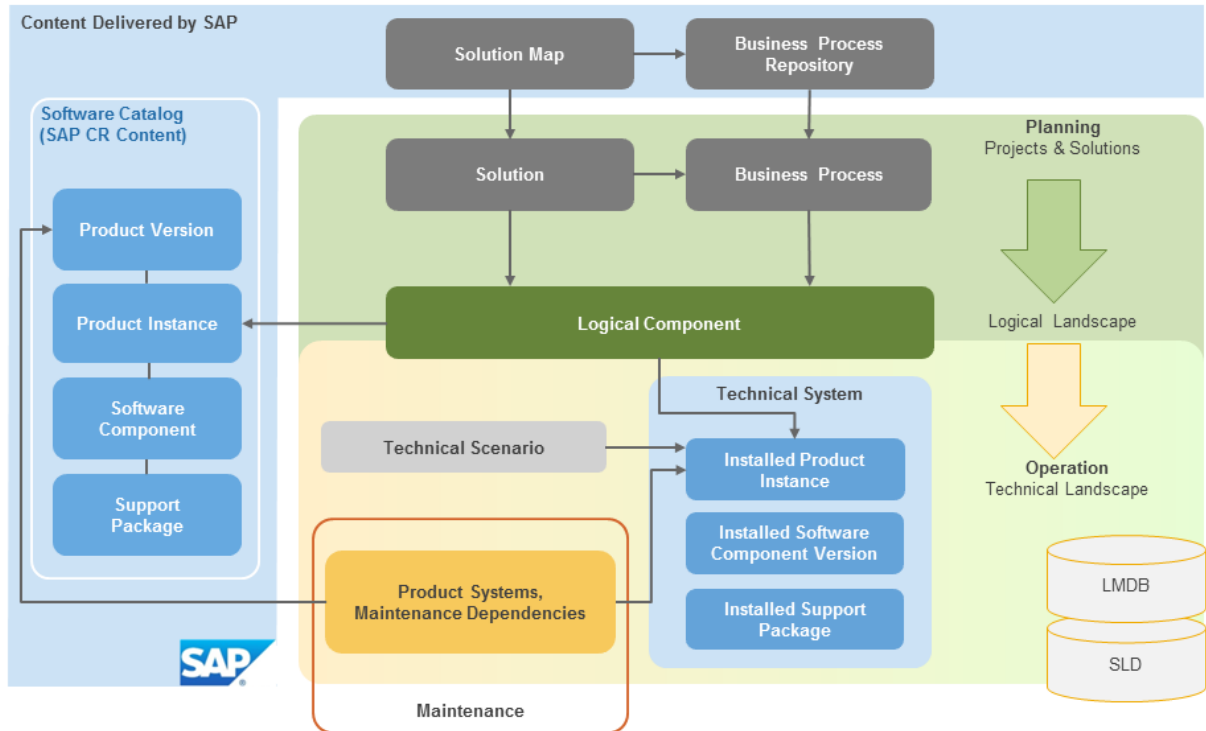
➔ Remember

Guiding Principle: A simple and intuitive landscape model allows for easy navigation and reference to an object. Although not all functional parts of SAP Solution Manager need all landscape entities for their tasks, each entity should serve uniformly as a node for navigation and grouping, across all scenarios.

Do not store data that is not needed, especially if it has to be maintained manually. Every entity of landscape description should have a known consumer.

5 Logical and Technical Landscape

SAP Solution Manager maintains landscape information repositories. The elements and high-level grouping of the repositories and some of the relationships between the managed entities are shown in the following graphic:



SAP provides support packages through the SAP Service Marketplace, and solution maps, business process-related data, templates and product information, on demand. This information is used to plan, maintain and operate management processes.

The **logical landscape** comprises solution, projects, business processes and logical components. Logical components are links to the technical landscape. You create and change the logical landscape manually.

The **technical landscape** comprises product systems, technical systems, product instances, software components and support packages. Technical systems are created during the installation process, and register themselves in the SLD, which forwards the information to the LMDB of SAP Solution Manager. Product systems are created manually in the LMDB because they group product instances according to product dependencies, and the planned technical landscape layout.

➔ Remember

Guiding Principle: We try to store all technical landscape information, and all grouping constructs of the technical landscape, such as technical scenarios and product systems, in SLD and LMDB. The illustration focuses on technical landscape elements and groupings. For navigation and reporting, grouped entities can be

referred to from throughout the functional groups of SAP Solution Manager. External repositories, such as a configuration management database (CMDB), can be integrated into the SAP Solution Manager landscape by:

- Context-based navigation links between the tools and applications
- External APIs (e.g. Web services) to access data on demand
- Synchronization of data. Data consistency must be considered when data is duplicated, so this is usually very expensive in Total Cost of Development (TCD) and Ownership (TCO).

6 Customer Landscape Management Activities

There are three main categories of landscape management activities, which each use different parts of the landscape model: planning, build, deploy & test, and operations.

A customer landscape often performs all of these activities at the same time, for different parts of projects and daily operations.

Planning

Setting up or adapting a landscape involves two planning activities:

1. **Business process planning** translates the business requirements into a process to be implemented or adapted. The primary sources of information are the *solution map*, *process repositories*, and the *business requirements*. Applications are represented by *logical components* and their *product instances*; a logical component describes at a high level the vehicle on which the processes will run.
2. **System landscape planning** is concerned with the physical setup of *application components* and *technical systems*. **Solution landscape planning** refers to *solutions*, *processes*, and *logical components*. Technical systems may not initially be linked to a logical component. **Technical landscape planning** refers to the physical layout of the solution landscape. It sets up *technical systems*, which host *product instances* for operations and maintenance. As in business process planning, you need to decide what must be newly allocated and what can be reused, by considering operational and maintenance perspectives.

A landscape setup or adaption can originate from either of the two types of planning, neither has priority over the other, and they need to “know about” each other.

For example, when you plan to implement a new business process, you need to decide how to set up logical components, product instances and technical systems. Although you start top-down, you need knowledge about the overall actual landscape.

Maintenance

Maintenance deals with support package deployment and upgrade tasks. The dependencies between technical systems, installed product instances, and installed software component versions are the primary source of information that guide the processes. Product systems identify these dependencies.

Operation

Technical operations usually refer to technical landscape entities such as hosts, technical systems, technical scenarios, relational database management systems (RDBMS), and the connections between them.

For example, **technical operations includes the technical monitoring**. Beside other capabilities, it monitors system availability, checks whether jobs are running on time and in time. If you have set up solutions and logical components, you can also maintain **business process monitoring**.

Supporting all three scenarios requires one consistent view and description of the landscape. This view needs to contain existing and planned landscape elements. Future elements are primarily used in planning, existing elements primarily in maintenance and operations.

➔ Remember

Guiding Principle: The landscape model must support all three categories equally. Separate views of the landscape support an end user who only needs to know and use what is needed for his task. Other information may be shown, but must be consistent across all scenarios.

An end user only sees a projection of the landscape data: The projection contains those parts that are needed for his task in particular.

7 Top-Down Structuring vs. Bottom-Up Guessing

The key challenge in a landscape design is the balance between what needs to be configured and what can be discovered.

- A **model that relies on configuration** ('top-down structuring') requires detailed knowledge and probably high manual effort. Landscape data configuration errors are likely, which leads to an instable and often incomplete landscape description.
- In a **model that relies on landscape element discovery** ('bottom-up'), it is difficult to form the higher-level landscape relations. The description must be enriched with additional data during deployment, or the discovered data will not be sufficient to deduce the exact nature of the components, and discover dependencies and links between them. A lot of 'guessing' is required.
A further challenge is the continuous changing of the solution landscape. You will rarely start from scratch, and you will rarely perform just maintenance. A solution will always be in planning, maintenance, and operation. The landscape setup and maintenance model needs to reflect this and provide a way of discovering what already exists.

➔ Remember

Guiding Principle: You need to find a balance so that planners and maintainers are not overloaded with too many unrelated details, and administrators and operations are not overloaded with high-level information that they do not need for their daily activities. There must be an understanding of what each part will create, and how the two directions meet and connect. An existing landscape needs to be “discovered” with minimal manual effort.

There is a balance point between the “logical” and “technical” worlds. The logical domain focuses on the manual top-down approach, and the technical world gathers physical information from the technical elements. Both areas rely on a consistent concept for the entire landscape. For example, you cannot create and run processes and logical components without considering both technical setup and operational aspects.

8 Setting Up a Landscape

We can refine and drill down into the landscape details almost without limit, but at a higher level there are only four key activities to set up landscape elements.

The term 'setup procedure' implies that the steps have a defined sequence, which is only partially true because, as described above, a solution landscape is always in a planning, maintenance, and operation. Nevertheless, the four steps are part of each of these stages, and there is a certain sequence. Once a solution landscape exists, a change might only concern the technical part, leaving the business process untouched, or vice versa.

1. Step: Business Process Plan

When a business process and technical landscape map exist, you first define the deployment requirements for the new or adapted process. Logical components need to be allocated or re-used. You can already redefine or reuse product systems, depending on the process requirements.

➔ Remember

Guiding Principle: A business process plan should refer to an existing technical landscape plan, or be created at the same time as the overall solution landscape plan. Product systems are the key prerequisite for maintenance and should be allocated and planned now, not later.

2. Step: Technical Landscape Plan

After high-level planning, define logical components and product instances to form the technical landscape. Define which **product instance** must be installed on which **technical system**. This mapping is defined by **logical components**.

You need to consider many aspects for this mapping: Performance and availability requirements as well as what should be combined to operate and maintain the solution, so this step requires an overview of the overall landscape, and collaboration between the business process and technical landscape planners.

This step is followed by the physical installation of the software.

➔ Remember

Guiding Principle: A technical landscape plan should refer to an existing business process plan, or even be created at the same time. Groupings of technical operations, software logistics, and product version maintenance are central parts of this plan.

3. Step: Deployment

The hardware and product instances and their software components are installed according to the plans made in the first two steps. The installed technical systems register system information automatically in SLD (forwarded to LMDB).

➔ Remember

Guiding Principle: If the automatically provided technical system information is incomplete or incorrect, issues have to be solved at the source of the problem. Manually correcting this kind of information in secondary stores must be a rare exception.

4. Step: Configuration

Once the software components are installed, several activities can take place. To realize the business process, you can now configure and customize applications on the target technical system.

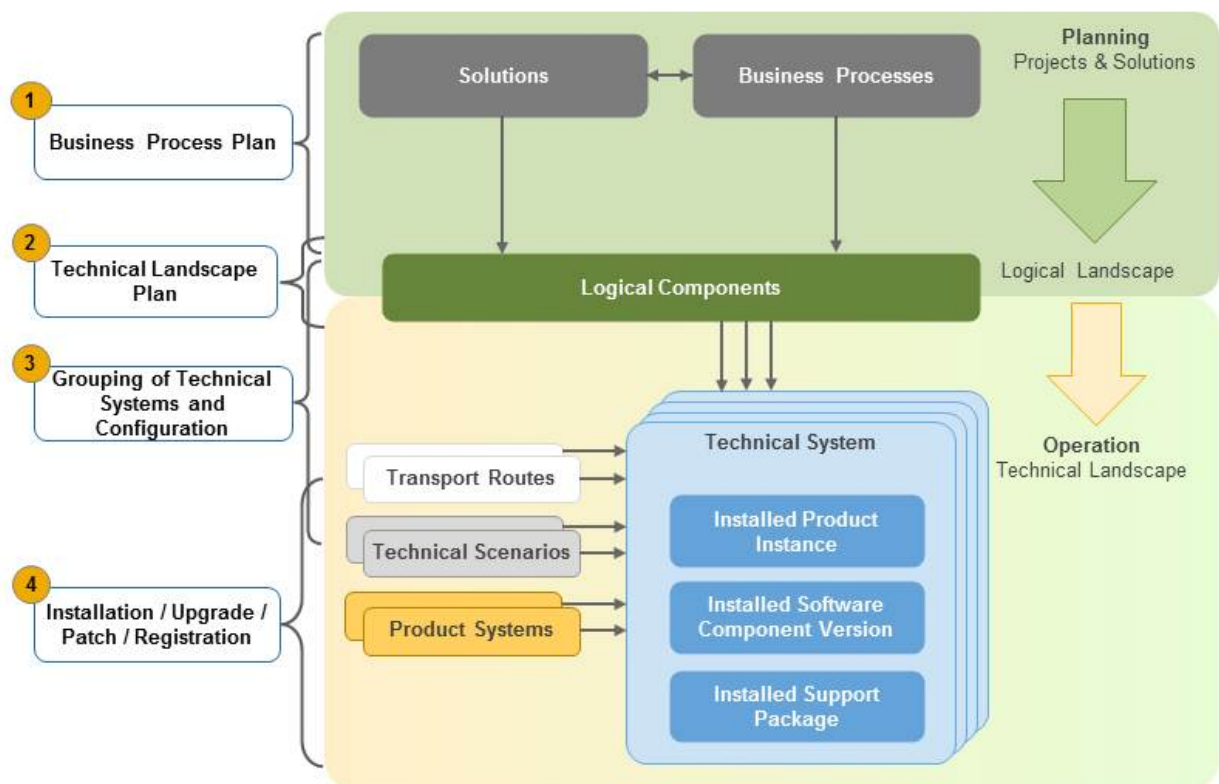
- For operations: Configure **technical scenarios**. They group technical systems for operational activities such as technical system monitoring.
- For maintenance: Create **product systems** and assign installed product instances. A product system is the unit of maintenance planning across the entire landscape; it defines which installed product instances belong together and need to be updated and patched as one unit.
- For transport management: Configure **transport routes and clusters** in the technical landscape.

➔ Remember

Guiding Principle: Since landscapes are unique, connecting and grouping cannot be completely automatic. You must validate the consistency of all linkages and groups among all repositories, such as LMDB and SLD.

Grouping should be intuitive and have the same process and structure throughout all repositories. We may have different types of groups, but we should not have different implementations. This configuration approach can be combined with agile system development.

The following figure shows the steps and their relations in the landscape, simplified to the key task. The plans result in a set of logical components and technical systems. Most of the activities in landscape management focus on the connection and grouping of these elements.



The process starts with initial solution planning and continues with the configuration of the managed systems, in a **top-down approach**. This is the preferable way of managing the various entities.

The reality is often different: There is already an installed system landscape with various technical systems. A lot of software based on many different technologies (such as ABAP, Java, or Microsoft .NET) can be installed. SAP Solution Manager might not be set up correctly to manage the various systems, because prerequisites are not

fulfilled. For example, several of the entities, such as solutions, product systems, logical components and technical scenarios – which structure the landscape – are missing or incomplete. In this case, we speak of a **bottom-up approach**.

For example, an installation, upgrade, or patch requires a well-defined landscape definition in SAP Solution Manager. Tools like **Maintenance Optimizer** can calculate the correct installation procedure. Depending on the use case, you define only the parts of the landscape that are needed for the installation, upgrade, or patch - not the whole landscape.

9 Summary

SAP is aiming for a model and processes that allow for a simple view, largely self-maintaining entities, and very little need for user input. Product complexity is only exposed to the degree that is necessary for the task. Customers should focus on the high-level operations to maintain their landscape.

However, many landscapes do already exist and cannot be re-created because we changed the model or a process, even if a new creation would be an improvement.

This white paper provides a strategy and direction for customers and SAP development in this challenging environment.

The SAP model must support the following activities for landscapes:

- *Solution planning* deals with solutions and logical components.
- *Maintenance* deals with product systems and technical systems with installed software for which updates, upgrades must be deployed.
- *Operations* deal with technical scenarios and technical systems.

It is a challenge that all activities are related to each other. If we proceed top-down, i.e. from solution to technical system, we rely on a documented and configured technical landscape to assign technical systems to logical components. If we proceed bottom-up, identification and descriptive means do not always allow for an unambiguous prediction of which product instances are installed on a technical system. Manually entered data is not reliable.

➔ Remember

Guiding Principle: A UI works on as high an abstraction level as possible: UIs only offer what is needed for a selected activity.

Rules for data maintenance:

- The landscape tools automatically handle settings that are technically known.
- Where uncertainties exist, the landscape tools must offer short, weighted selection to support manual input by the user.
- The system should allow an automatic setting or selection to be overruled manually.
- The user must be able to distinguish manually entered data from automatically provided data.

SAP Solution Manager helps you to fulfill the guiding principles and supports you in optimizing the end-to-end lifecycle management.

For more information, see the SCN Wiki [Documentation for Landscape Management Database – LMDB](#). Here, you can also leave a comment to give feedback and get the latest version of this white paper.

Important Disclaimers and Legal Information

Coding Samples

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 to 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, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Accessibility

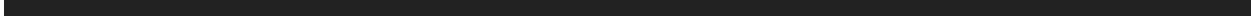
The information contained in the SAP documentation represents SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of wilful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or wilful misconduct. All links are categorized for transparency (see: <http://help.sap.com/disclaimer>).





**go.sap.com/registration/
contact.html**

© 2016 SAP SE or an SAP affiliate company. 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 SE or an SAP affiliate company. The information contained herein may be changed without prior notice.
Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.
These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company 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.
SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.
Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.