

Custom Applications and Standard SAP Solutions: What's the Difference When It Comes to Lifecycle Management?

Constantly changing processes and decidedly unique business objectives mean that updates, customization, and adaptation are a healthy and even necessary part of every company's solution landscape. However, without a strategic approach to lifecycle management, they can also be unnecessarily costly. For SAP customers, this applies both to implementations of standard SAP solutions and to your custom development scenarios. For effective lifecycle management, what tools are available to streamline your solutions — from implementation and development to updates and change management? And how do standard SAP offerings and custom solutions require *different* tools when it comes to lifecycle management?

A previous "Take Note!" column¹ clarified terms like *solution lifecycle management* and *software logistics*, with an overview of SAP tools for customers' lifecycle management needs. This article delves further into the details, offering specifics on support for the life cycles of standard SAP solutions as well as your own custom development, and takes a careful look at their distinct lifecycle management needs. It also offers a quick look at the future of lifecycle management as SAP solutions — and the entire application development arena — change and evolve.

¹ See my article "Lifecycle Management — What's That?" in the October-December 2004 issue of *SAP Insider* (www.SAPinsider.com).

Lifecycle Management for SAP Standard Solutions

Overall, the solution life cycle for standard SAP offerings can be divided into three main phases:

- The *implementation and introduction* of an SAP solution involves planning and blueprinting, configuration, testing, and preparing for go-live.
- *Productive operation* comprises monitoring and administration of hardware resources, databases, dialog user sessions, background jobs, printing, and other loads to be distributed across systems.
- *Change management* includes application of patches, custom extensions, upgrades to new releases, and anything that must be changed in the overall system due to new requirements or new facts about the behavior of the overall solution.

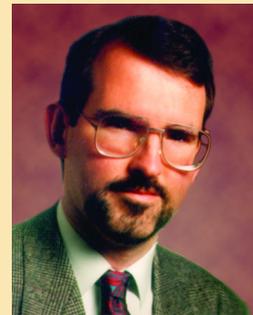
SAP offers tools and services that address software management needs for each of these phases (see **Figure 1**).

Implementation: Blueprinting, Configuration, and Testing Tools

SAP has a long history of delivering standard software that is adaptable to specific customer needs by configuration rather than programming, which provides more stability, higher speed of change, lower cost of change, and

Regular Feature

Take Note!



Dr. Franz-Josef Fritz, SAP AG

much easier upgrade to new releases. *SAP Solution Manager* provides the methodology, tools, and content to perform business-process-based blueprinting as the basis for technical setup and configuration.²

The main configuration tool is the *IMG*, SAP's implementation guide, which lists all actions required for implementing your SAP project, provides you with the direct navigation to these actions, and helps you control and document the implementation. *BC Sets* (Business Configuration Sets) are another more recent development to simplify configuration when you're dealing with connected sets of parameters in multiple, related SAP applications — for example, when you're implementing a CRM

² Please see my previous article in the October-December 2004 issue of *SAP Insider* (www.SAPinsider.com), as well as related links.

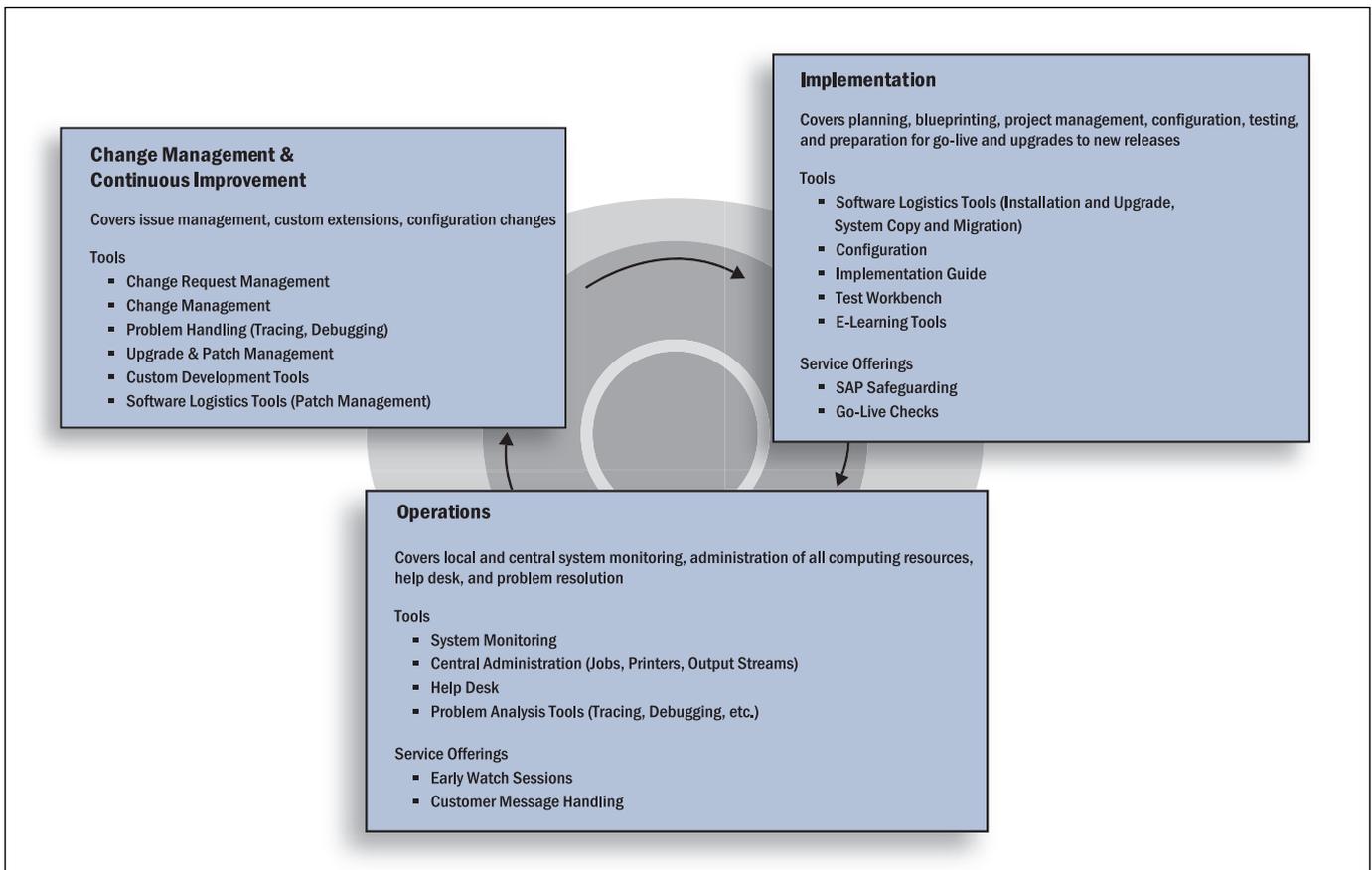


Figure 1 The Phases of the Solution Life Cycle for Standard SAP Offerings

system running in connection with an SAP R/3 Enterprise system.

Does this configuration work always happen at the customer site? The answer is no. Many generic, broadly applicable *software engines* offered by SAP — like the ones for configuration, pricing, general ledger, payroll, and so on — provide a lot of flexibility for implementing standard solutions. They can work in many different ways, and in a variety of different business scenarios and processes. But this level of flexibility is rarely needed beyond the core application development phase. Therefore, in many of these cases, SAP software engineers can and increasingly will build business-scenario-specific, predefined configurations of these engines and make them available “ready to use.”

In addition to the IMG, a new set of tools is being developed for greatly simplified creation and application of

preconfigurations; shipment is planned for the next release of SAP NetWeaver. (It’s something like the early days of TV sets, when everyone had to adjust the colors on their own individual sets at home. Now your television set comes preconfigured — nobody needs to bother to change these settings anymore.) SAP is heading toward solutions that, more and more, come with these predefined configurations to reduce the cost and complexity of the implementation phase.

On top of these well-tested scenarios, preconfigured by SAP, companies develop their own corporate standards and templates for enterprise-wide rollouts (see **Figure 2** on the next page). This corporate governance should further limit the flexibility of configuration models at the business-analyst and end-user layers, and at the same time simplify the task of local domain administrators. Of course, end users will want some flex-

ibility to personalize their applications, but again this needs to be reigned in to conform to company standards and rules.

For testing, the *Test Workbench*, provided in SAP Solution Manager, in conjunction with *eCATT* (extended Computer Aided Test Tool), provides a powerful and easy-to-use environment to test applications (user interfaces, function modules, and many more) in a distributed environment. SAP delivers test scripts for easy and comprehensive regression testing.

✓ Note!

If, in exceptional cases, SAP preconfigurations need to be adjusted because of extreme and very specific requirements, they should only be changed by a highly expert “guru” — not by a typical consultant or user.

Operation: Administration and Monitoring Tools

For many years, SAP has provided CCMS (Computing Center Management System) for easy, centralized monitoring and administration of distributed landscapes. Monitoring agents were hooked into the various components of a system landscape. Monitoring data was collected and aggregated centrally, providing alerts and capabilities to drill down into the components for problem resolution. For the Java stack, the *Visual Administrator* has been added in SAP Web Application Server 6.20 to provide a similar toolset.

SAP is currently combining all these tools into a Web-based administration facility, which allows central (or even remote) administration and monitoring of all components in a solution landscape. This capability will be made available for SAP NetWeaver 2004 and all products built on top of it, like mySAP ERP. Pilot versions have already been shipped, and general availability will be reached by Q2 2005.

Another important capability is the supported *management processes* for analysis and problem management — again this is a synergetic combination of tools at the customer site and services provided by the SAP support organization. The analysis tools are built in such a way that they allow Customer Competence Center specialists or even SAP support personnel working remotely to analyze difficult situations and problem reports.

Change: Change Management and Release Upgrade Tools

Recently *Change Request Management (ChaRM)* functionality has been added to SAP Solution Manager, SAP's standard application management solution.³

Available as of SAP Solution Manager

³ See <http://service.sap.com/solutionmanager> for more on SAP Solution Manager functionality.

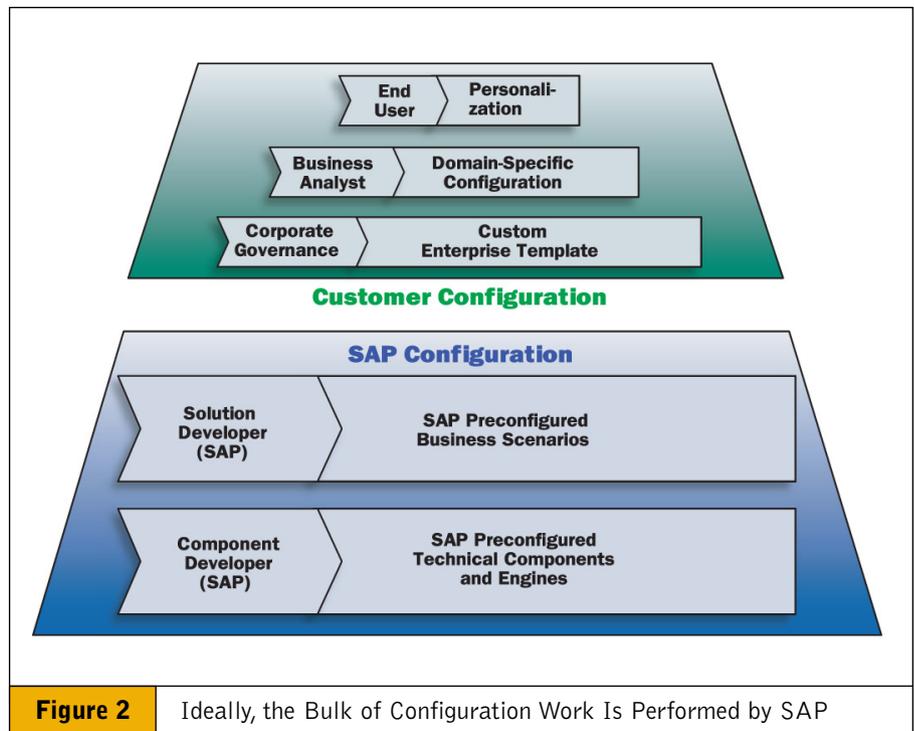


Figure 2 Ideally, the Bulk of Configuration Work Is Performed by SAP

Not Just Tools — SAP Support and Services, Too

Lifecycle management for SAP solutions not only covers the tasks handled by customers autonomously, it also extends to collaboration with SAP service and support teams across this life cycle.

During the implementation phase, for example, SAP support can provide safeguarding by analyzing the overall hardware, software, and configuration setup to determine inefficiencies and risks before the solution goes live.

Once the solution is live, SAP support services can remotely detect, analyze, and solve problems in the most efficient way to minimize the impact of problems and maximize the availability of the solution.

In both phases, your systems can be directly linked with SAP support systems to generate automated analysis reports and allow in-depth inspections by suitable SAP personnel, if the situation requires it.

Release 3.2, ChaRM features provide all the needed workflow functionality for approval and tracking of change requests.⁴ Likewise, SAP has developed a *change management component* as part of Release 3.2, which deals with the project management aspects as well as the needed software logistics/software transport operations.

⁴ For more information on Change Request Management and other application management tools available in SAP Solution Manager, see "Looking for Ways Your IT Organization Can Contain Costs Without Sacrificing Services? An Introduction to SAP Solution Manager Tools" by Cay Rademann in this issue of *SAP Insider* (www.SAPinsider.com).

Furthermore, an *issue-tracking component* (as part of SAP Solution Manager) is under development to provide the same end-to-end view of any administration and problem management issues that need to be driven to complete resolution as efficiently as possible.

An important goal is to upgrade and change productive systems with minimal downtime. SAP has developed and delivered lots of *innovative approaches to reduce downtime* over the years, including:

- Repository Switch Upgrade technology, which allows the bulk of the work for the upgrade process to be done on a “shadow repository” while the system is up and running for productive use. This helps significantly reduce downtime and risk during upgrades.
- ABAP Modification Assistant, to merge customer modifications with SAP-delivered software changes.
- Incremental migration for tables, which also helps to migrate whole systems from a non-Unicode to a Unicode environment.

In this area, SAP Solution Manager supports the planning of upgrades by delivering roadmaps, tools for comparing actual and target landscapes, and project management facilities.

Managing Custom Development with SAP Tools

When it comes to extensions to SAP solutions or new applications built on top of SAP NetWeaver, SAP delivers a

whole set of tools (see **Figure 3**) to support lifecycle management of *custom development* as well:

- Development tools and supporting development infrastructure services
- Software distribution tools (software logistics)
- Monitoring, administration, and analysis tools for the productive operation
- Tools to handle change requests coming from the productive use of the software

On the production and change management side, many of the same tools are at work for both homegrown and SAP standard solutions. In the areas of development and distribution, however, there are some distinct challenges for custom development that SAP helps you meet.

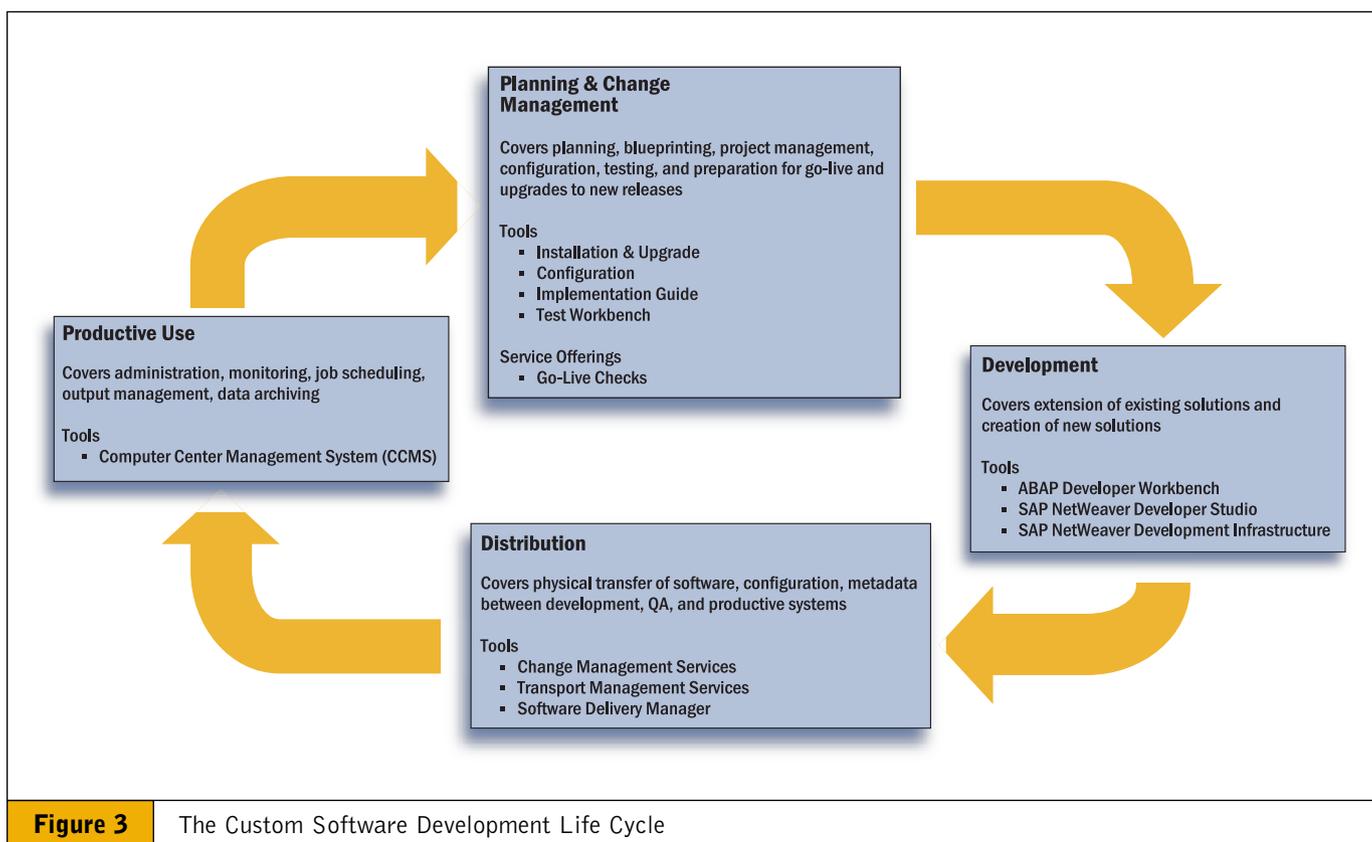
Development: SAP NetWeaver Development Infrastructure

With SAP Development Workbench (for ABAP development) and SAP

NetWeaver Developer Studio (for Java and all the newer capabilities in SAP NetWeaver), SAP delivers a comprehensive set of tools for custom development in all areas of SAP NetWeaver — in fact, the very same set of tools used at SAP for application development. These developer tools are accompanied and supported by a set of server-side tools for productive development in larger groups and projects: the *SAP NetWeaver Development Infrastructure (NWDI)*. Most important are the Design-Time Repository for managing various versions of source code, metadata, and all kinds of other development objects, and Component Build Services, for automated creation of executables out of the right combination of source code and required building blocks.

Distribution: Software Logistics Tools

Distribution and deployment tasks are covered by a set of tools for handling installations, upgrades, and patches — and many of the same tools apply for



software from SAP, partners, or for custom and homegrown software. These *Software Logistics* tools are linked back to the development tools so that, for every new or changed piece of software, you can define the component or layer it belongs to and how it needs to be distributed once development is finished. Transport routes for software are defined from development to integration testing/QA to production.

How Lifecycle Management Is Changing

As SAP solutions evolve, and with SAP NetWeaver technology serving as the underlying integration and application platform for all new products and releases, lifecycle management will necessarily evolve and change as well. We are all witnessing the following general shifts in IT landscapes and application development:

- From software coding to *model-driven development*
- From programming language source code to *business packages* with metadata, data, process and UI templates, and many more related building blocks
- From application components to *end-to-end solutions* across components
- From monolithic systems to *adaptive system landscapes*
- From manual operation to automated and *autonomous operation* with strong support for handling of exceptional events (collecting the relevant context information automatically and offering effective ways to resolve)
- From proprietary software and system management mechanisms to *standards-based management* concepts and tools

To address changes like these, SAP introduced the *Enterprise Services Architecture (ESA)* over the last year or so. While ESA was originally designed to streamline solution architecture, it is also expected to play a key role in lifecycle

management.⁵ With ESA, Web services-based communication between components allows for a logical and physical decoupling of the building blocks of ESA — those components that act as the “providers” of services and the “consumers” of these services, which receive these communications. As a result, the software components that provide services can be flexibly deployed on a very broad scale — from one small machine on up to multiple big machines. ESA’s metadata-based service definitions are key building blocks for creating solutions with much less coding than was required previously — or even without any coding at all.

ESA supports and even enforces the decoupling of the various layers of your solution architecture, thereby reducing and hiding complexity in business processes. It not only brings huge benefits for applications, but also has great potential for lifecycle management tools themselves. Initiatives like Web Services Distributed Management (WSDM) not only provide concepts for *management of Web services* (MOWS), but also for *management using Web services* (MUWS). These initiatives are still in an early stage, but the direction is very clear: *the ESA architecture will also apply directly to lifecycle management.*

By taking a services-based approach, ESA helps reduce the costs associated with handling multicomponent software. In keeping with this approach, the following goals have been defined for lifecycle management as supported in future versions of SAP NetWeaver:

- ☑ Reduce complexity with:
 - Simplified and coherent software logistics tools that will handle installations, upgrades, and patches for the ABAP and Java side, as well as metadata-based deliverables like Web

⁵ For a primer on ESA, see my column, “When Does a Web Service Become an Enterprise Service? An Introduction to the Principles of Enterprise Services Architecture (ESA)” in the April-June 2004 issue of *SAP Insider* (www.SAPinsider.com).

Dynpro definitions, portal iViews, BI InfoCubes and reports, XI mappings, and others, which together make up a solution or a business scenario

- Preconfiguration and template-based reconfiguration of components, thereby vastly reducing configuration effort and taking much risk out of changes
- Consolidated operations infrastructure, for a much more unified technology and tools across all layers and components
- ☑ Centralize operations and software logistics for a whole system landscape by:
 - Supporting central description and management of the whole distributed solution landscape, including non-SAP components
 - Adding a single point of control for all SAP NetWeaver lifecycle tasks
 - Providing consistent methodology and tools for all parts of SAP NetWeaver and applications on top of SAP NetWeaver
 - Bringing together business processes and their required underlying software components in both software logistics and operations

For additional information on current and evolving SAP solution lifecycle management offerings, as well as lifecycle management offerings for your custom development projects, please refer to www.sap.com/netweaver, <http://service.sap.com/solutionmanager>, and www.sap.com/services. ■

Franz J. Fritz has a Ph.D. in mathematics and 30 years of experience in all areas of IT.

Workflow and business process management have been particular areas of interest for much of his life. He has worked for SAP since 1993 as Program Director and Vice President with responsibility for the Business Process Technology and Internet-Business Framework departments. Recently, he took over responsibility for SAP NetWeaver product management within SAP AG as Vice President of NetWeaver Product Management and Standards.