In this reading selection, you’ll get a glimpse at the steps you’ll need to follow if you have existing workflows in your SAP system, and need to upgrade to a new release. Also, take a brief look at a new chapter in this edition, which discusses how to build a simple workflow application in SAPUI5.

Preface
“Upgrading SAP Business Workflow"
“Using SAPUI5”

Contents
Index
The Authors

Jocelyn Dart, Sue Keohan, and Alan Rickayzen, et.al
Practical Workflow for SAP
1089 Pages, 2014, $79.95/€79.95
ISBN 978-1-4932-1009-1

www.sap-press.com/3615
Welcome to the third edition of *Practical Workflow for SAP*! For those of you who already have experience with SAP Business Workflow, you know that the workflow community really believes in the value of SAP Business Workflow and the real impact it can have on your SAP implementation. The third edition will enable you, the workflow expert and evangelist, to update your workflow skills in SAP NetWeaver with new technical topics such as ABAP classes, user interfaces (UIs), inbox options, SAP Decision Service Management, SAP Fiori, and more. For those of you who are new to workflow, this book is intended for you as well; we hope to have you as an active participant in the workflow community!

**Contents of the Book**

The first edition of this book was written by SAP workflow experts who are passionate about the topic and about the impact it can have on your business. This third edition of the book stays true to the core goal of providing practical guidance on SAP Business Workflow, but also expands the first and second editions with new capabilities and new ways to use SAP Business Workflow.

This book is divided into five parts:

- **Part I: Getting Started with Workflow in SAP**
  This part of the book starts by introducing SAP Business Workflow, positioning it within the overall SAP Business Process Management (SAP BPM) strategy. It then discusses what you need to know to get SAP Business Workflow running on your system. It discusses how users get tasks from workflow, and the newer ability to use SAP Operational Process Intelligence powered by SAP HANA workflow reporting.

- **Part II: Administering Workflows**
  This part of the book walks you through all of the various tools necessary to administer your workflow environments, including advanced diagnostics, debugging, and guidance when preparing for an upgrade.
Since reporting on workflow-enabled processes has become more and more important over the years, it will also introduce you to using SAP Business Warehouse (SAP BW) for SAP Business Workflow reporting.

**Part III: Developing Workflows**

In this part of the book, we get to the heart of developing your workflows, including using the tried-and-true Business Object Repository (BOR) as well as integrating ABAP classes. Use of events and other business interfaces will help you determine how to start and control your workflows. You will learn about service-enabling your workflows so your workflow can cross system and organizational boundaries. You’ll learn how to use the BRFplus and SAP Decision Service Management tools to manage complex business rules.

**Part IV: Enhancing Workflows**

This part of the book covers various ways to enhance your workflows for newer UIs. It will enable the workflow developer to use Web Dynpro (ABAP or Java) as well as Business Server Pages (BSP) to extend the reach of workflow beyond the traditional SAP GUI. The newest UI technology, SAPUI5, is also covered.

**Part V: Using SAP Business Workflow in SAP Applications**

This part of the book covers SAP Business Workflow as used by key SAP Business Suite applications. These include SAP Supplier Relationship Management (SAP SRM), SAP Customer Relationship Management (SAP CRM), SAP ERP Human Capital Management (SAP ERP HCM), and SAP Master Data Governance. You will also be introduced to SAP Fiori, giving your users additional mobility access.

There are also two appendices, which can be referred to for tips and tricks as well as exploring new functionality that has been delivered since the second edition of the book.

With the division of the book into five major parts, you can read the different parts as you need them. Parts I, II, and III are critical for understanding how workflow works. Part IV can be read when you need to integrate one of the discussed UIs with your workflow. For Part V, you can read the chapter that corresponds to the SAP application where you want to use SAP-provided workflows, or you want to know what is unique about workflow on a particular application system.

**Target Groups of the Book**

This book is intended for business process experts (BPEs) and developers new to the topic of SAP Business Workflow. Although there is some code in this book, the bulk of the “how to build and execute workflows” content requires no programming.

It is also intended for existing workflow experts who are upgrading to SAP NetWeaver 7.0 and above, and need to upgrade their workflow skills. This target group won’t need all chapters, but there are new capabilities provided in many chapters (e.g., in the chapters on building workflows and using events), and there are new chapters on SAP Decision Service Management, SAPUI5, and other topics you may not have used on your existing release.

If you’re already a workflow expert and are starting a workflow project on SAP SRM, SAP CRM, or SAP ERP HCM, then this book will provide important insight on how to use workflow on these application systems.

**Structure of the Book**

In detail, the book consists of the following chapters:

**Chapter 1: Introduction**

This chapter provides an introduction into SAP Business Workflow by discussing what SAP Business Workflow is, when to use SAP Business Workflow, its major features, and how SAP Business Workflow fits in with the SAP overall strategy for SAP BPM. If you are an existing workflow expert, we still recommend that you read this chapter so you know where workflow fits in the SAP BPM strategy.

**Chapter 2: Requirements Gathering Strategy**

Before starting any workflow project, you need to know what is expected when the workflow goes into production. This chapter discusses questions that should be asked before starting a workflow project, how to know if the business problem is a candidate for SAP Business Workflow, and how to measure your results. This chapter is a must read before starting your first workflow project. The content also includes a number of checklists that can be downloaded from the SAP PRESS website (http://www.sap-press.com/3615).
Before you can start using workflow, the system must be configured to use SAP Business Workflow. This chapter discusses how to do the initial required configuration.

Chapter 4: Work Item Delivery
Work item delivery discusses the users’ experiences with SAP Business Workflow. This chapter discusses the inboxes that exist and the capabilities of the inboxes, and provides recommendations for delivering work items to users.

Chapter 5: Agents
When a workflow executes, it’s critical that the right person receives the right task. This chapter discusses the role of agents in the design of a workflow.

Chapter 6: Setting Up an SAP-Provided SAP ERP Workflow
Chapter 6 walks through an example of configuring a workflow template that’s provided by SAP in an ERP system.

Chapter 7: SAP Operational Process Intelligence Powered by SAP HANA
This chapter describes an SAP HANA add-on that has been developed to be used in conjunction with different workflow systems, including SAP Business Workflow. This opens a whole new set of opportunities, breaking many technical boundaries and giving business visibility and control of the underlying workflows.

Chapter 8: Workflow Administration
The workflow administration chapter provides information on what reports are available with SAP Business Workflow, and discusses how to ensure the workflows and agent assignments are kept updated to meet changing business requirements.

Chapter 9: Using SAP Business Warehouse for SAP Business Workflow Reporting
This chapter discusses the option to report on workflow execution from SAP BW. This gives the option to provide analytical reports for upper management on how often and how long processes are executing.

Chapter 10: Administration Troubleshooting Guide
This chapter provides a practical guide to troubleshooting some of the thornier problems that you may run into in the development environment.

Chapter 11: Advanced Diagnostics
Chapter 11 walks through the commonly used tools that are used to debug and resolve common workflow problems.

Chapter 12: Upgrading SAP Business Workflow
This chapter is intended to be read as part of preparation when planning an upgrade from SAP R/3 4.6x to SAP NetWeaver 7.0 and above.

Chapter 13: Creating a Workflow
Chapter 13 provides concrete examples and guidance when creating your very first workflow. It walks you through each part of workflow creation.

Chapter 14: Advanced Workflow Design Techniques
Advanced workflow design goes beyond the most simple workflow design. This chapter discusses more advanced topics such as parallel processing, containers and bindings, and other topics to enhance your workflow skills.

Chapter 15: Business Objects
Workflow functionality is based on application functionality. Business objects from the BOR link the workflow to the application functionality. This chapter discusses how objects work and how to create your own business objects.

Chapter 16: ABAP Classes
ABAP classes can be used instead of or in addition to business objects. Classes link workflow to business functionality. This chapter walks you through creating your own ABAP classes for SAP Business Workflow.

Chapter 17: Agent Determination Rules
Chapter 5, Agents, discusses how agents work in SAP Business Workflow. This chapter walks through options to determine who should get a specific task at runtime.

Chapter 18: Using Events and Other Business Interfaces
After a workflow is created, the application must tell the workflow to start and possibly when to stop. This chapter walks through how to get the workflow to respond to what is happening in the application.

Chapter 19: Custom Programs
This chapter covers the most commonly used Workflow Application Programming Interfaces (WAPIs).

Chapter 20: Service-Enabling Workflows
After you have a workflow in place, you may need the workflow to be called as a service. This chapter discusses Service-Oriented Architecture (SOA), service-
enabling a workflow, and calling the service via SAP Process Orchestration (SAP PO).

- **Chapter 21: BRFplus and SAP Decision Service Management**
  In this chapter, we’ll introduce you to business rules and the framework that supports them (BRFplus), as well as SAP Decision Service Management. Within SAP Business Workflow, you can use these technologies to control thresholds, deadlines, path selection, and much more.

- **Chapter 22: User Interface Options**
  This chapter describes required steps if you want to execute a workflow task using Web Dynpro or Business Server Pages (BSP). This chapter should be read before reading Chapter 23 through Chapter 25.

- **Chapter 23: Using Web Dynpro ABAP**
  Chapter 23 uses the user decision step type available in SAP Business Workflow to discuss using a Web Dynpro ABAP with SAP Business Workflow.

- **Chapter 24: Using Web Dynpro Java**
  This chapter covers the major steps required when using Web Dynpro Java to execute a workflow task.

- **Chapter 25: Using Business Server Pages**
  This chapter walks through an example of executing a BSP as the UI for a workflow step.

- **Chapter 26: Using Forms**
  Chapter 26 discusses the form step type in SAP Business Workflow.

- **Chapter 27: Using SAPUI5**
  You’ll take a look at SAPUI5 as it stands in relation to the other UI technologies presented in this book. We’ll then walk you through the steps to create a fully working simple workflow application that retrieves tasks from SAP, presents them in a list, allows the display of detailed information, and also gives the user the chance to approve or reject.

- **Chapter 28: ArchiveLink**
  Chapter 28 discusses the use of ArchiveLink for imaging projects with SAP Business Workflow.

- **Chapter 29: SAP Supplier Relationship Management**
  This chapter discusses how workflow is used in SAP SRM 7.0 and SAP SRM 5.0. It describes the workflow frameworks and provides examples of usage.

- **Chapter 30: SAP Customer Relationship Management**
  This chapter discusses how workflow is used in SAP CRM 7.0. It describes the SAP CRM frameworks, including the SAP CRM UI, and how workflow works in SAP CRM.

- **Chapter 31: SAP ERP Human Capital Management—Process and Forms**
  This chapter covers the major use cases for workflow in SAP ERP HCM administrative services and provides a business and technical overview of form usage.

- **Chapter 32: SAP Governance, Risk, and Compliance**
  GRC products all rely on the standard SAP Business Workflow, although some additional features are worth mentioning. In this chapter, we’ll examine GRC workflow more for where it varies from or builds upon the SAP Business Workflow standard.

- **Chapter 33: SAP Fiori and Mobility**
  In this chapter, we’ll briefly look at key issues to consider when considering mobile workflow.

- **Chapter 34: SAP Master Data Governance**
  In this chapter, we’ll quickly overview the SAP Master Data Governance application and the change request process, and then discuss the different ways you can use and implement SAP Business Workflow and rule-based (RBWF) to make changes to the process.

- **Appendices**
  Appendix A covers common tips and tricks. Appendix B gives information about new features and functionality. In addition to the appendices, there are also ready-to-use checklists for your first workflow project you can download from the SAP PRESS website (www.sap-press.com/3615), as well as downloads of the code used throughout the book.

**System Requirements**

Although many sections of the book are release independent, the content is based on workflow capabilities available in SAP NetWeaver 7.4. When discussing SAP applications, the SAP Business Suite is generally assumed. Specifically, SAP ERP (ECC 6.0), SAP SRM 7.0, and SAP CRM 2007/7.0 are the releases that are used in the application-focused chapters.
Further Training
If you want more information on workflow after reading this book, SAP Educational offers classes on SAP Business Workflow. BIT600 is an introduction class, BIT601 is a class on building workflow, and BIT610 is a class on development with SAP Business Workflow. For further information, visit the SAP Education website at http://www.sap.com/services/education.

Acknowledgments
This book could not be possible without the incredible efforts from the authors of the first edition. Alan Rickayzen, Jocelyn Dart, Carsten Brennecke, and Markus Schneider provided a very successful first edition that has been widely used and is greatly appreciated by the workflow community.

Ginger Gatling accepted the challenge to update that original “Workflow Bible” in 2009, and assembled 14 authors from around the world to donate their time and energy. These authors included additional workflow developers, consultants, partners, and customers. Ginger’s efforts cannot be overestimated. She worked with each author on their specific chapters and ensured the whole lot hung together cohesively.

Each and every author of this edition accepted and rose to unique challenges. Whether trying newer tricks in ABAP classes, reviewing more mature technologies for updates, or forging ahead with shiny new stuff, they all brought their A game. I am so fortunate to have had some association with so many bright and dedicated people.

When we started this book project, we all knew it would take time and considerable effort, but the time taken from family was certainly underestimated. We are all very fortunate in life if we can find people who love us, and when those people encourage us and support us during times when work takes front and center, as it did while working on this book, we realize how fortunate we are! So, thank you to our families and friends who provided support, encouragement, and patience throughout this project.

As a final note, I hope you have noticed that all royalties from this book will be donated to Doctors Without Borders/Médecins Sans Frontières. I remember early conversation with Ginger about what to do about author royalties for the second edition—how could you even begin to calculate one author’s updates to an existing chapter written by a previous author, versus someone writing an entirely new chapter? Well, you couldn’t. So in addition to her cat-herding skills, Ginger was also instrumental in getting the proceeds from the second edition donated to a cause that the SAP Community has supported in various ways over the years—Doctors Without Borders/Médecins Sans Frontières. At the time of this writing, the second edition had raised roughly $35,000 to support Doctors Without Borders.

Each author has individually decided to donate his or her royalties. Your purchase of this book helps us support an international medical humanitarian organization that delivers emergency aid in many countries. Thank you for enabling us to provide support to this important organization.

I hope to see you blogging about workflow on SCN after reading this book!

Susan Keohan
SAP Community Network member
Upgrade projects include workflow upgrades, even if the workflow is limited to a technical-only upgrade. This chapter provides recommendations for when you’re planning and executing an upgrade.

12 Upgrading SAP Business Workflow

By Eddie Morris

If this is the first chapter you’re reading in this book, then you probably have several existing workflows in your production system, are planning an upgrade, and are wondering how the journey will go. This chapter gets your journey started through a discussion of recommended steps to take before your upgrade begins, steps you should take during the upgrade, and what to do after the upgrade completes.

The workflow upgrade will be successful if it has a complete project plan, including a thorough understanding of your existing workflows, setting user expectations for what will happen before and after the upgrade, creating workflow test plans, executing a proof of concept, and performing significant testing. This chapter will explain how to get this all in place.

12.1 First Steps to Upgrading Workflow

Assessing, learning, and planning should be the very first thoughts you have when considering an upgrade and the impact to your existing workflows. Consider the following points:

► Your upgrade planning should begin with an inventory of your existing workflows.

You should be able to answer the following questions:

► How many workflows do you have in production?
► How often do they execute?
How many instances of workflow are in error?
What custom business objects have you created?
How many custom attributes and methods have you created in a business object?
Have you done any modifications to existing objects or other workflow functions?
Have you created custom copies of SAP-provided workflow templates and customized these copies?

Plan to perform a proof of concept on the upgrade.
Your full upgrade project will probably include a mock upgrade or proof of concept upgrade on a copy of your production system. Ensure workflow is included in this effort.
Workflow testing should be performed on the most critical workflow processes. This enables you to define realistic timelines and user expectations for the project.

Consider an archiving strategy before your upgrade project.
If your existing system has been around for many years, you may have completed work items from many years ago. Archiving these work items reduces the amount of time and effort that’s involved in the upgrade.

Leverage other customers who have already completed an upgrade.
Your region should have a local SAP user community. Be sure to network with your regional SAP user community to find customers who have already performed their workflow upgrade. Some user groups may have a special interest group that’s focused on workflow. Networking with other customers who have completed an upgrade is usually very insightful.

Familiarize yourself with new workflow features.
Make you leverage what SAP delivers to the fullest extent. There is no need to create a custom feature when the feature is already provided by SAP. Doing your homework on the new workflow features can save development time.
By reading the other chapters in this book, you’ve started learning these new features. For the upgrade preparation, be sure to read Chapter 14, Advanced Workflow Design Techniques. This chapter covers new container and binding features that you need to understand when testing your workflows after the upgrade.

12.2 Steps to Take Before the Upgrade Begins
Planning, planning, and more planning are critical to your workflow upgrade. The following steps outline what should be done before the upgrade begins. Some steps can be started at the beginning of your planning process. For many of the steps, the earlier they start, the better.

12.2.1 Complete All Running Workflow Instances (Where Possible)
Maybe you can’t complete all the workflow instances, but at least complete all of the ones you can reasonably complete. Completing all running workflow instances is much easier to recommend than to actually do. Many of the customer issues appearing after an upgrade are due to existing work items in the users’ inboxes. You should perform an evaluation to know which workflows you can reasonably complete.

Be sure to complete as many workflow instances as you possibly can. If you have long-running workflows (for example, training workflows that could take months to complete), you won’t be able to complete all instances. But you should look at each workflow individually and determine which ones can and should be completed before the upgrade begins. Many short-lived approval and Electronic Data Interchange (EDI) workflows should be completed before the upgrade.

12.2.2 Create Workflow Test Plans
Your existing workflows must be fully tested. The test plans should include full executions of each workflow you have in production:

- For the critical workflows, make sure that you test each process scenario within the workflow (test each branch).
- For workflows that must be “in progress” when the upgrade happens, make sure that examples of each workflow in progress are included in your test plan.

After the upgrade completes, there should be no surprises when executing a work item that was in a Ready status before the upgrade began.
12.2.3 WF and T Tasks versus WS and TS Tasks

Depending on the release that you're upgrading from, you may have created WF and T tasks when creating your own workflows. If you're upgrading from a release where WF and T tasks were used, you should know that SAP recommends all workflows be based on WS and TS tasks. WS and TS tasks types are client-independent objects and recommended by SAP for all custom workflow development.

You aren’t required to convert your WF and T tasks to WS and TS tasks; it’s recommended you do copy the T to TS and WF to WS. However, this does not have to be done before the upgrade; it can be done over time after the upgrade.

12.2.4 Clean Runtime Tables

Cleaning the runtime tables is one of the most important things you can do to help with the speed of the upgrade, as well as to ease debugging after the workflow upgrade completes. Removing the workflow runtime tables of all unnecessary data enables the upgrade to run faster. The best option is to archive as much workflow runtime data as possible according to SAP Note 573656.

Whenever the word archiving comes up, the conversation normally gets very quiet. Many customers prefer not to archive anything. The tricky part about archiving workflow is that workflow items are normally considered one part of a total archiving project. For example, if you have SAP Supplier Relationship Management (SAP SRM) shopping basket approvals, the archiving strategy could dictate that you archive the entire shopping cart and the related purchasing documents. If your company has an archiving strategy, be sure that workflow is included. If not, work with your business stakeholders on a possible archiving project for your old work items.

There might also be runtime data that you simply don’t need any longer; for example, workflow that’s related to EDI processing, or other work items where the workflow isn’t subject to a business or system audit. If you have runtime data that isn't needed in the future, consider deleting this data via Report RSWW_WIDE (see SAP Note 49545). Be aware that once deleted, the data is no longer available. Various workflow database tables may contain unnecessary entries after you delete the work items. Therefore, it’s also a good idea to run Report RSWW_WIDE_DEP and Report RSWW_REORG_SWWUSERWI to remove these unnecessary entries. See SAP Note 1427068 for more information on Report RSWW_REORG_SWWUSERWI.

SAP Note 1068627 covers all of the notes you should use when upgrading to SAP NetWeaver Application Server 6.40 or 7.0. Ensure that all notes listed are applied to your system. Also ensure that you review any recommendations contained in the notes.

12.3 Steps to Take During and After the Upgrade

This section discusses the steps you should take during and after an upgrade.

12.3.1 Conversion of Event Linkage Tables

In release SAP R/3 4.6C, the instance linkage for workflow steps was stored in Table SWEINSTCOU. Beginning with release 4.7 (Basis 6.20) and onward, this information is stored in Table SWFDEVINST. During the upgrade, Report RSWFEVTXXPRA moves the entries from the old table into the new table. This conversion can take a very long time. However, if you can delete and archive existing work items, the event linkage tables will be cleaned up as well.

12.3.2 Basis Support Package

After the workflow upgrade, you should apply the latest Basis support package. The latest support package has all current fixes, and it’s important that you immediately upgrade to the latest support package.

12.3.3 Configure Your System for Workflow

On completion of the workflow upgrade, it’s important that you configure the upgrade system by calling up workflow Customizing (Transaction SWU3). You have to define the workflow administrator to configure the RFC destination and schedule the workflow batch jobs. You may choose to PERFORM AUTOMATIC WORKFLOW CUSTOMIZING or execute all activities manually with the EXECUTE button.

For more details on configuring, refer to Chapter 3, Configuring the System.
### Recommendation

Don’t run **Perform Automatic Workflow Customizing for section Classify Tasks as General**, because this may lead to previously unassigned work items being sent to all users’ inboxes. See Knowledge Base Article 1504215 for further information in the Support Portal.

#### 12.3.4 Workflow Definition and Binding

After you perform your upgrade, you may have issues in your workflow definition and in the binding. SAP Note 1060762 discusses container operations with date fields and time fields, providing guidance on container operations that may need to be adjusted. You might also have ABAP Data Dictionary objects in your workflow container that are no longer valid after the upgrade. This is covered in SAP Note 1058159.

Binding checks are stricter as of SAP NetWeaver 7.0. Binding definitions that did not error or cause problems may result in binding errors. Your workflow template containers may refer to ABAP Data Dictionary data types that no longer exist in the upgraded system. SAP Note 939489 covers the issue and recommends that you re-create the missing type ABAP Dictionary element. If this isn’t possible, the note also provides a program that you can execute to resolve the problem.

With release 7.10, the data binding mechanism has become even stricter. After the upgrade, running instances may suddenly go into error due to the more severe data flow inspection at runtime. See SAP Note 1787443, because after it’s applied, it makes the data binding more fault tolerant again. With the corresponding support packs of SAP Note 1787443, you can switch the binding runtime checks on or off via Transaction SWPA. See Knowledge Base Article 1732734 for more information.

#### 12.3.5 Tables SWW_CONTOB and SWW_CONT

One major change from release SAP R/3 4.7 is persistence of the workflow container. This is discussed in Chapter 14, Advanced Workflow Design Techniques. The change is the use of XML persistence. Tables SWW_CONTOB and SWW_CONT only continue to be used if you choose the “old” container persistence.

When using XML persistence in your workflow, no entries are written to these tables. You can change the settings for the persistence profile of a workflow. In the Workflow Builder, select Goto • Basic Data • Version Dependent • Control. Look in the Persistence Profile tab to change the settings.

Figure 12.1 shows the persistence options for the workflow container. XML Persistence is the new entry. The other options continue to use Tables SWW_CONTOB and SWW_CONT for container persistence.

When using XML persistence in your workflow, no entries are written to these tables. You can change the settings for the persistence profile of a workflow. In the Workflow Builder, select Goto • Basic Data • Version Dependent • Control. Look in the Persistence Profile tab to change the settings.

Entries with XML persistence are stored in Table SWWCNTP0. The only issue you'll have with the XML persistence is custom code that reads from Tables SWW_CONTOB and SWW_CONT.

#### Recommendation

Don’t change the persistence profile for existing workflows. Use the XML persistence for new workflows that are created after the upgrade.
12.3.6 Changed Locking Behavior of Runtime

The locking behavior at runtime has been changed in SAP NetWeaver 7.0 for synchronous dialog chain steps. The change was made to improve throughput.

Figure 12.2 shows the difference between the old and new locking mechanisms. In previous releases, when each step executed in a dialog synchronous chain, the work item ID was locked when the step was executing. The lock was enqueued when the step started, and dequeued when the step completed. After the first step completed, then the parent work item ID (work item ID for the WS template) was locked. The parent ID was dequeued when the next step in the chain began. Throughout the three steps, either the parent work item ID or the TS task ID was locked. For each lock acquired and released, the lock had to be enqueued and dequeued.

For customers who have very high volume and require more throughput, the design was changed so that the parent work item ID is locked throughout the entire synchronous dialog chain.

Locking throughout the entire dialog chain results in faster throughput and fewer system resources consumed. However, when a workflow executes with parallel steps, sometimes the new locking can lead to locking conflicts. When this occurs, the workflow stops until the next scheduled run of the job SWWERRE to fix workflow errors.

You have the option to configure the use of the standard locking or the new locking. You configure it in the workflow definition via the menu path GoTo • Basic Data • Version Dependent • Control. Then select the Locking properties tab. The options are shown in Figure 12.3.

**Recommendation**

Use the standard locking unless you have a specific throughput situation where you need the enhanced locking. If enhanced locking is required, use the Enhanced Locking Procedure (All Workitems) option. Refer to SAP Note 1122756 for more details.

12.3.7 Other Issues after the Upgrade

After the upgrade completes, you need to consider a few other issues:

- If you stay with the SAP GUI, you need to upgrade to the latest SAP GUI release and patch level.
After the upgrade, some users find that they can no longer display the object method by clicking the link in the preview pane of Transaction SBWP (the transaction for the inbox). This also may affect the users’ ability to view work item attachments or secondary methods. This is due to missing authorization of Transaction SWO_ASYNC, which is detailed in SAP Note 1006235 and Knowledge Base Article 1840681.

After the upgrade completes, there might be issues with deadlines, hanging workflows, and transports that were in progress before the upgrade began. Notes to assist in resolving upgrade issues are shown in Table 12.1 in the following section.

After the upgrade, you might find that some objects don’t work as normal. The solution for this is to generate the object and any subtypes that are associated with the object via Transaction SWO1.

### 12.4 Important SAP Notes

Table 12.1 lists important SAP Notes when it comes to upgrading.

<table>
<thead>
<tr>
<th>SAP Note</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1068627</td>
<td>Composite workflow upgrade note</td>
</tr>
<tr>
<td>573656</td>
<td>Collective note for archiving work items</td>
</tr>
<tr>
<td>1019080, 808790</td>
<td>Performance during upgrade</td>
</tr>
<tr>
<td>1060762</td>
<td>Container operation with date fields and time fields</td>
</tr>
<tr>
<td>1058159, 939489</td>
<td>Missing elements in the workflow container</td>
</tr>
<tr>
<td>1228836</td>
<td>Compatibility of conditions with date/time constants</td>
</tr>
<tr>
<td>1787443, 1732734</td>
<td>Stricter binding from Basis release 7.10</td>
</tr>
</tbody>
</table>

Table 12.1 Important SAP Notes (Cont.)
This chapter provides an introduction to SAPUI5, which is a UI technology with a different pattern from many of the other technologies in this part of the book. We’ll build a simple workflow application while paying particular attention to some of the SAPUI5 features that are used in the application’s construction.

## 27 Using SAPUI5

By DJ Adams

SAPUI5 is a toolkit that has been constructed to enable outside-in applications to be developed. It contains a wide selection of tools in the form of libraries, user interface (UI) controls, data management mechanisms, module management, internationalization, and more. The runtime for these outside-in applications is the web browser—more specifically, modern web browsers that support a collection of standards and technologies (predominantly HTML, Cascading Style Sheets [CSS], and JavaScript) collectively known as HTML5. The SAPUI5 toolkit contains everything you need to build modern, interactive, responsive, and platform-independent web-based applications, as described in the following list:

- **Modern**
  The UIs of SAPUI5-powered applications have a modern look and feel and can stand shoulder to shoulder with other “consumer grade” applications that have a user experience we’ve come to expect from applications used outside the enterprise context.

- **Interactive**
  The web browser is the most widespread, and arguably the most powerful and capable virtual machine (VM) for today’s applications. With this power comes a flexibility not seen before on such a universal and generic scale, and when combined with the outside-in approach to application development, fine detail interactions aren’t only possible, but become the norm.

- **Responsive**
  This is a subject large enough for its own book, but suffice it to say that today’s
consumer and business requirements often include the desire to interact with an application independent of device. There are libraries within SAPUI5 that are specifically designed to support writing an application once and having it run well on desktops, tablets, and smartphones.

Platform independent
With the majority of web browsers (and, more importantly, layout and rendering engines such as WebKit) holding no allegiance to any particular operating system, applications written natively for the web are by definition platform independent.

In this chapter, we’ll first take a look at SAPUI5 as it stands in relation to the other UI technologies already presented in this part of the book, to understand what it is, how it stands apart, and where it’s employed. We’ll then walk you through the steps to create a fully working simple workflow application that retrieves tasks from SAP, presents them in a list, allows the display of detailed information, and also gives the user the chance to approve or reject. The last section presents some pointers for further features and information.

Note
For more information on SAPUI5, see the book Getting Started with SAPUI5 by Miroslav Antolovic (SAP PRESS, 2014).

27.1 UI Applications

In this section, we’ll help you get an understanding of how SAPUI5 compares to other UI applications. The inside-out pattern refers to an approach that is taken with building applications that have a user-facing aspect. Essentially, the UI is built “inside” the system and pushed out to the user. Server-side technologies are used to define the components of what the user sees, and that definition is then pushed out to the user in a mostly static form. Minimal business and control logic exist outside the server.

But there’s another pattern to consider, which is “outside-in.” Here, we see the most extreme version of client-server application programming. The application logic, at least the logic that drives the transition between different aspects of the application’s functionality (as well as the logic that handles user interaction, data formatting, dynamic presentation of UI elements, and more) is built outside of the server context and directly for the client, where it runs independently of the server. The interaction with a server (say, an ABAP stack) is via discrete HTTP calls to retrieve and manipulate business data and execute backend business logic. Although not entirely independent of the server, an outside-in client application can be seen as an entity in its own right; indeed, with the appropriate client-side mock data services in place, it can be developed and tested completely in isolation from the server.

27.2 Inside SAPUI5

Now that we’ve been introduced to SAPUI5, let’s take a look at its major component parts, with a particular focus on what is required to build our simple workflow application. We’ll cover various libraries, and pick one in particular that will help us create an application that has a responsive design. We’ll also look at how we interact with the server, as well as examine features of SAPUI5 that support nontrivial application design, such as the Model View Controller (MVC) concept and component-based code organization.

27.2.1 The Libraries

The libraries are collections of features, predominantly UI controls, which can be used to construct the UI of your application. There are many libraries, far too many to cover in this chapter, but we’ll cover a few important ones.

Perhaps the most important library for building modern multiplatform applications is the sap.m library, which contains a wealth of visible UI elements. As the part of the library name intimates, it was designed for mobile first, so the controls are responsive from the ground up.

A great showcase of sap.m controls is the sap.m Explored application that’s available in the SAPUI5 SDK, itself written using sap.m controls. A coffee break or two getting to know this Explored application and what it showcases will be time very well spent. Note that the source code is available for all the sap.m samples presented in the Explored application—just select the appropriate button to see it.
27.2.2 Model View Controller

The Model View Controller (MVC) concept has been around for a very long time. It allows a developer, or a team of developers, to separate out the different concerns of an application (presentation of visible elements, handling of logic, and integration to the data sources) so that each concern can be independently built and maintained.

At its core, SAPUI5 supports the MVC concept and makes it easy to build applications with loosely coupled models, views, and controllers. It’s not opinionated, nor does it dictate any particular pattern. Rather, it makes the construction of MVC applications an accessible and first-class approach to development.

There are a couple of aspects of the MVC support that are of particular interest for us: the model support for various data sources, and the array of possibilities when it comes to defining views.

Model Support

SAP’s MVC implementation includes support for different data source types: arbitrary XML, JavaScript Object Notation (JSON), and OData. Support can be either with one-way binding (from the model to the view) or two-way (from the model to the view, and vice versa). The OData model support is of particular interest to us for our simple workflow application, because it allows us to reach back to the server and enjoy read/write access to SAP’s workflow system. In our case, the server is an ABAP stack with SAP Gateway components, although it could be any SAP system that speaks OData.

The OData model mechanism is known as a server-side mechanism, in contrast to, for example, the JSON model mechanism, which is client-side. That’s not to say that the JSON model can only be instantiated with data on the client side; the instantiation of a JSON model object can specify a URL from where JSON data can be loaded. The difference is in where the nominal postinstantiation “home” of the data lies.

With JSON, all of the data is loaded onto the client side (into the browser). With OData, only the data required to satisfy the current bindings is pulled from the server. The rest of the data remains on the server until required. Furthermore, we can use OData operations to write back to the server.

Sketching Out the Application

Now that we’ve covered some of the fundamentals and some of what SAPUI5 provides, it’s time to start considering our application design. In this section, we’ll sketch out and construct a simple UI using SAPUI5 features that we’ve explored previously. We’ll be using a standard SAP Business Workflow service to access items, and the purpose of the application is to display items in a list, allow selection and the viewing of more detail, and finally offer the user a chance to approve or reject. We want to have the application run on desktops, tablets, and smartphones, so we’ll use controls from the sap.m library. We want to access the workflow features in an ABAP stack system, so we’ll use the OData model (this will come in the next section), and we’ll use the MVC approach.

27.3.1 The sap.m.SplitApp Control

Now that we’ve picked the sap.m library to provide the core of our UI elements, we still have to make a decision on how our application will look overall. There are a number of design patterns that some of the controls in the sap.m library represent; one of those is the “Master Detail” pattern, which presents a list of items in a master view, and details of a selected item in a detail view.
On desktops and tablets, the master view is normally shown in the left third of the screen, and the detail view is normally shown in the right two-thirds. On smartphones and other small devices, either the master or the detail is shown at any one time, and transitions are used to hide or reveal information as appropriate.

In the sap.m library, the SplitApp control is exactly what we need to achieve this design pattern.

### 27.3.2 MVC and XML Views

We’ll use MVC to separate out the different concerns of our application. Specifically, we’ll define our views declaratively using XML. You may be wondering at this point exactly how we’re going to use views, how many we’ll have, and where they fit in, especially as we’re going to use the SplitApp control to coordinate the rest of the UI. The SplitApp control has a couple of aggregations `masterPages` and `detailPages`, which suggest that the content of these aggregations should be, well, `sap.m.Page` controls.

This isn’t a requirement, however, and to achieve full-on MVC, it’s possible, and indeed common practice, to have these aggregations actually contain views with corresponding controllers. This is what we’ll do. Figure 27.1 shows a schematic of how the views and other controls will fit together in a typical SplitApp constellation. Note that we’re also going to be using the component approach, with a `UIComponent`, which is also shown in the schematic.

**Figure 27.1** Relationship of `UIComponent`, `SplitApp`, and Views

### 27.3.3 The Component

As you’ll see when you explore the SAPUI5 documentation, the component concept is very powerful and flexible. We’ll just scratch the surface in our application, but it’s good practice to think and build in terms of component objects.

Now that we’ve looked at our building blocks for the application’s UI, let’s look at piecing it together.

### 27.3.4 What We’re Aiming For

Let’s take a look at what the master detail pattern translates to in real terms. Figure 27.2 shows our application UI rendered on a tablet. It would look very similar on a desktop.

**Figure 27.2** Our Application UI on a Tablet
The master detail pattern is clear to see in the way the UI is divided into a one-third (master) and a two-thirds (detail) section. In the master, we see a simple list of workflow tasks awaiting action. In the detail, we see information for the selected task (the first in the list) and also a couple of action buttons to either Approve or Reject the task.

Before moving on, let's have a look to see how this UI is rendered on a smartphone in Figure 27.3.

As we've mentioned, only the master or the detail view is shown at any one time, and there's a navigation (back) button in the detail view to allow the user to return to the list in the master. Note also that the layout of the Basic Details (WorkItem ID, Type, Priority, etc.) has automatically adjusted itself, with values popping under the field names to fit the real estate available.

27.3.5 UI Construction: Step by Step

Let's take the UI construction step by step, and we'll soon have what you've seen in the previous figures.
After the meta tags that declare the character set and also give Internet Explorer hints that it should use the most up-to-date rendering (we don’t have to do this for other browsers), we have the SAPUI5 bootstrap—a script tag that points to SAPUI5’s core. There are various parameters that we pass:

- `data-sap-ui-theme`
  The `sap_bluecrystal` theme is the one that is used most often in conjunction with the `sap.m` controls, especially in the SAP Fiori user experience (UX) context.
- `data-sap-ui-libs`
  Normally, we’d declare the library loading requirements in the component, but because we’re using a `Shell` control from the `sap.m` library here in the `index.html`, we want to have `sap.m` specified in the bootstrap.
- `data-sap-ui-xx-bindingSyntax`
  To allow us to do things such as mix literal strings with curly brace model property placeholders in our UI control properties (e.g., in the `ObjectListItem`’s `intro` property in the master view), we need to declare that we want to use “complex” binding syntax.
- `data-sap-ui-resourceroots`
  With this parameter, we can easily match up folder locations with control namespaces; in this case, we’re saying that controls (the component and views, mostly) whose names start `practicalworkflow` are to be found in the current folder, which is represented by the path specification `"./"`.

After we’ve declared the SAPUI5 bootstrap, it just remains for us to instantiate the component within a `ComponentContainer`, wrap that in a `Shell` control from the `sap.m` library (the `Shell` gives us some control over the application’s visible width), and place the `Shell` into the HTML document.

### 27.3.6 Component Definition

Our component definition comes next. It’s in a `Component.js` file in the same folder as the `index.html` (and will be found automatically through a combination of the earlier `resourceroots` declaration and the default name `Component.js` for the module that’s loaded).

The definition is a very simple one, as this is a simple application, and is shown in Listing 27.2.

```javascript
jQuery.sap.declare("practicalworkflow.Component");
sap.ui.core.UIComponent.extend("practicalworkflow.Component", { createContent : function() {
    // create root view
    var oView = sap.ui.view({
      id : "idViewApp",
      viewName : "practicalworkflow.view.App",
      type : "XML"
    });
    // set data model on root view
    return oView;
  } });
```
var sWFServiceURL = "http://<server>:<port>/sap/opu/odata/IWWRK/WFSERVICE/";
var oModel = new sap.ui.model.ODataModel(sWFServiceURL);
oView.setModel(oModel);
oModel.attachEventOnce("requestCompleted", function(oEvent) {
sap.ui.getCore().getEventBus().publish("data", "ready");
});
// done
return oView;
}
}
Listing 27.2 Component.js

We inherit the UIComponent from sap.ui.core and declare a single method createContent, which SAPUI5 calls to ask for the visible controls that should be rendered.

We’re using the MVC concept and so instantiate and return a top-level, or “root” view. All our views are XML views, and this one is no exception. The name practicalworkflow.view.App resolves to the file App.view.xml in the view/ folder in our application directory.

At this stage, we’ll gloss over the code that deals with the OData connection as we’re just focusing on the UI for now; we’ll come back to it in the next section.

27.3.7 Views and Controllers

Now it’s time to have a look at that App view that we instantiated in the component, as well as its controller, and the other views and controllers for our application.

App View and Controller

The App view that was returned as the content for the component is declared in the view/App.view.xml file and is shown in Listing 27.3.

Listing 27.3 App.view.xml

It’s very straightforward, and actually only contains a single control: sap.m.SplitApp. But let’s first take a quick look at how the view itself is declared; this will be the same for all the views that follow.

The XML declaration is directly followed by the XML’s root element—a View control from the sap.ui.core.mvc library. Note that we’re using XML namespaces for the library prefixes; the convention in our sample application is that the default namespace (the one without an explicit prefix declaration) represents the library that we’re going to use the most, sap.m. For the controls from other libraries (such as the View control itself), we declare a prefix; in this case, there’s just one, mvc, for sap.ui.core.mvc.

With the namespace declarations out of the way, we can declare the controls in a clean and lightweight way, as we’ll do here for the SplitApp.

As you’ve already seen, the SplitApp has a couple of aggregations (masterPages and detailPages), and we specify a View for each of these: the practicalworkflow.view.Master view for the masterPages aggregation, and the practicalworkflow.view.Detail view for the detailPages aggregation.

Note

You can see that the XML element names are either capitalized or not. The capitalized names (such as SplitApp) represent the controls, whereas the noncapitalized names (such as masterPages) represent aggregations.
Using SAPUI5

The App view’s controller is declared with the `controllerName` attribute and points to a name that happens to be the same as the view. But in this case, the default suffix of `.controller.js` means that this resolves to the file `App.controller.js` in the same folder (view) as the view. Refer to the application structure in Figure 27.4 for a reminder of where things are if you need to.

The App view’s controller is shown in Listing 27.4.

```javascript
sap.ui.controller("practicalworkflow.view.App", {
    onInit : function(oEvent) {
        this.oSplitApp = this.getView().byId("idSplitApp");
        // Have child views use this controller for navigation
        var that = this;
        this.oSplitApp.getMasterPages().forEach(function(oPage) {
            oPage.getController().navigation = that;
        });
        this.oSplitApp.getDetailPages().forEach(function(oPage) {
            oPage.getController().navigation = that;
        };
    },
    navTo : function(sPageId, oContext) {
        var sPrefixedPageId = this.getView().byId(sPageId).getId();
        this.oSplitApp.to(sPrefixedPageId);
        if (oContext) {
            this.oSplitApp.getPage(sPrefixedPageId).setBindingContext(oContext);
        }
    },
    navBack : function() {
        this.oSplitApp.backMaster();
    }
});
```

Listing 27.4  App.controller.js

The App’s controller concerns itself mostly with navigation. In the controller’s `onInit` event, we’re sharing the navigation functions of the `SplitApp` control with the other views by making the functions of this controller available via a navigation property of the master and detail views. For a simple application, this may be all you need, and is certainly all we need here.

There are two navigation functions that are used to navigate to and back from a detail view:

- `navTo`
  - This is called from the master view’s controller; it receives a page ID (`idViewDetail`), works out the fully qualified ID (in XML views controls are automatically prefixed), gets the `SplitApp` to navigate to it, and sets the data binding context appropriately.

- `navBack`
  - This is called from the detail view’s controller and simply gets the `SplitApp` to navigate back to the master.

Note that the page ID `idViewDetail` is taken from the ID given to the view when it was assigned to the `detailPages` aggregation of the `SplitApp` (refer to Listing 27.3).

Master View and Controller

The master view is where we define our list of workflow tasks and is shown in Listing 27.5.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<mvc:View
    controllerName="practicalworkflow.view.Master"
    displayBlock="true"
    xmlns:mvc="sap.ui.core.mvc"
    xmlns="sap.m">
    <Page title="Workflow Tasks">
        <content>
            <List
                id="idList"
                items="!/WorkflowTaskCollection"
                mode="SingleSelectMaster"
                select="onListSelect">
                <ObjectListItem
                    intro="Created by {created_by}"/
                    title="{task_name}"
                    number="{note_count}"
                    numberUnit="notes">
                    <attributes>
                        <ObjectAttribute text="{created_at}" />
                    </attributes>
            </ObjectListItem>
        </content>
    </Page>
</mvc:View>
```

Sketching Out the Application
In this view, we're declaring a `Page` control from the `sap.m` library. A `Page` is a common UI building block for applications, especially in the `sap.m.SplitApp` and `sap.m.App` context. It has a header with a title (as you can see in Figure 27.3, where the title in the master is `Workflow Tasks`) and the default content aggregation is where our task list is declared.

The task list is an `sap.m.List` control, which has an items aggregation that is bound to the OData entityset `WorkflowTaskCollection`. For each entity in this entityset, we display an `ObjectListItem`, also from the `sap.m` library, which presents various properties from the data model in a useful and concise way. You can spot the property bindings in the control by looking for the curly braces. You can read more about the data model properties in the next section.

The master view's controller must handle some navigation, but it's also used to cause the first item in the list of tasks to be preselected. Listing 27.6 shows the master view's controller.

```javascript
sap.ui.controller("practicalworkflow.view.Master", {
  onInit : function(oEvent) {
    sap.ui.getCore().getEventBus().subscribe("data", "ready", this.selectFirstItem, this);
  },
  selectFirstItem : function() {
    if (!sap.ui.Device.system.phone) {
      var oList = this.byId("idList");
      var oFirstItem = oList.getItems()[0];
      oList.setSelectedItem(oFirstItem);
      this.navigateToDetail(oFirstItem);
    },
    onListSelect : function(oEvent) {
      this.navigateToDetail(oEvent.getParameter("ListItem"));
    },
    navigateToDetail : function(oItem) {
      this.navigation.navTo("idViewDetail", oItem.getBindingContext());
    }
  }
});
```

Listing 27.6 Master.controller.js

We'll cover the `onInit` function in the next section. Suffice it to say that at the appropriate time, the `selectFirstItem` function is called to preselect the first item in the list. It does this unless the device is a smartphone, in which case, we don't want any item preselected or any navigation to the item's detail because the user would see the detail view first rather than the master view.

The following is a summary of the functions:

- `selectFirstItem`
  The `List` control is identified, the first item object is retrieved, and then the `List` control's `setSelectedItem` is called to highlight that first item. Finally, the controller's `navigateToDetail` is called to cause navigation to happen.

- `onListSelect`
  This is the handler for the `List` control's select event (refer to `select="onListSelect"` in Figure 27.1) and calls the controller's `navigateToDetail` with the object that represents the item that was selected.

- `navigateToDetail`
  Via the navigation connection to the `App` view's controller, this causes `SplitApp` navigation to the detail view to happen, passing the selected list item's binding context so it can be applied to the detail view after navigation.

Detail View and Controller

After a task has been selected from the `sap.m.List` control in the master view, the detail can be displayed for it in the detail view. Listing 27.7 shows the detail view.

```javascript
Listing 27.7 Detail.controller.js
```
Sketching Out the Application

This is very similar to the declaration of the master view, in that it starts out with
an sap.m.Page control, and everything else is within the Page's default content
aggregation.

The Page control is also similar to the one in the master view (it has a title and a
footer with a Bar in it), but there are a couple of additions:

- A handler (onNavButtonPress) is declared for the navButtonPress event.
- The Bar has buttons on the right-hand side (in the contentRight aggregation).

As you'll see shortly in the controller, the navigation (or back) button is shown
(it's the left-facing arrow in the header of the detail view's Page header in Figure
27.3), but only if the device is a smartphone. This is so that the user has some way
of triggering the navigation back from the detail view to the list in the master
view. If the device is large enough for the SplitApp to show both master and
detail views simultaneously (as you can see in Figure 27.2), then the navigation
button isn't needed.
The rest of the XML view declaration is straightforward; we use a number of sap.m controls, including an IconTabBar containing a couple of IconTabFilters to represent the Basic Details and Task Description data for the selected task. You can see an example of how the description is displayed in Figure 27.5.

Perhaps the most interesting thing in the detail view declaration is the use of an XML fragment to contain the declaration of the BasicDetails layout. This is in a separate file BasicDetails.fragment.xml, also in the view/ folder, and is shown in Listing 27.8.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<core:FragmentDefinition
xmlns:l="sap.ui.layout"
xmlns:f="sap.ui.layout.form"
xmlns:core="sap.ui.core"
xmlns="sap.m">
<l:Grid
defaultSpan="L12 M12 S12"
width="auto">
<l:content>
<f:SimpleForm
minWidth="1024"
maxContainerCols="2"
editable="false"
layout="ResponsiveGridLayout"
title="Task Information"
labelSpanL="3"
labelSpanM="3"
emptySpanL="4"
emptySpanM="4"
columnsL="1"
columnsM="1">
  <f:content>
    <Label text="WorkItem ID" />
    <Text text="{workitem_id}" />
    <Label text="Type" />
    <Text text="{type}" />
    <Label text="Priority" />
    <Text text="{priority}" />
    <Label text="Subject" />
    <Text text="{subject}" />
  </f:content>
</f:SimpleForm>
</l:content>
<l:Grid>
</core:FragmentDefinition>
```

The layout of the Basic Details screen (WorkItem ID, Type, Priority, and Subject) is done using a combination of a responsive Grid control from the sap.ui.layout library and a SimpleForm control from the sap.ui.layout.form library. The declaration of the span and column properties allows the form to reshape itself sensibly, depending on the screen real estate available.

Buttons

Before moving on to look at the detail view’s controller, we should examine the buttons in the footer’s Bar. These sap.m.Button controls represent the OK and Cancel buttons shown in Figure 27.5, and later in Figure 27.6. The button colors are declared indirectly and semantically via the Button’s type property, and both buttons share a single handler for their press events. This handler is onAcceptRejectButtonPress in the controller.

We’re almost there! Let’s have a look at the detail view’s controller, which is shown in its entirety in Listing 27.9.
Using SAPUI5

27

Sketching Out the Application

27.3

jQuery.sap.require('sap.m.MessageBox');
jQuery.sap.require('sap.m.MessageToast');
sap.ui.controller("practicalworkflow.view.Detail", { 
mDecKeys : {  "Accept" : "0001",  "Reject" : "0002",  "Forward" : "0003" },
onInit : function() {  var oPage = this.getView().byId('idDetailPage');  oPage.setShowNavButton(sap.ui.Device.system.phone);  var oDescriptionFilter = this.byId('idIconTabBar').getItems()[1];  oDescriptionFilter.bindElement('Description'); },
onNavButtonPress : function(oEvent) {  this.navigation.navBack(); },
onAcceptRejectButtonPress : function(oEvent) {  var sActionType = oEvent.getSource().getType();  sap.m.MessageBox.confirm("Are you sure you wish to take this action?", jQuery.proxy(function(sConfirmation) {  if ("OK" === sConfirmation) {  this.processAction(sActionType);  } }, this), "Please confirm" );  },
processAction : function(sActionType) {  var sDecKey = this.mDecKeys[sActionType];  if (sDecKey) {  var oView = this.getView();  var sWorkItemId = oView.getBindingContext().getProperty('workitem_id');  var oModel = oView.getModel();  oModel.callFunction("ApplyDecision", "POST", {  "workitem_id" : sWorkItemId,  "dec_key" : sDecKey
});
},

to the top of the Detail.controller.js file. We have a couple of require statements, to bring in static classes sap.m.MessageBox and sap.m.MessageToast, functions from which we’ll be using when we handle the actions triggered by the OK and Cancel buttons.

We also have a static map of decision keys that’s located in the controller definition. This is hard-coded here; we could have taken the possible decision keys from the server, but we’re doing it this way to keep things simple.

Here are the functions one by one:

1. **onInit**
   In our function to handle the view’s initialization, we set the navigation button’s visibility depending on whether our device is a smartphone or not. We also explicitly call `bindElement` on the `IconTabFilter` representing the task’s description, which we’ll explain in the next section.

2. **onNavButtonPress**
   If a navigation button is visible, and it’s pressed, this handler is called and uses the SplitApp’s navigation mechanisms (via the `navigation` property) to go back to the master view.

3. **onAcceptRejectButtonPress**
   When this handler is called, we look at the event’s source (via the standard `getSource` function from the `event` object that is passed to the handler) to determine what was pressed. We’re using a simplistic mechanism here by
looking at the button’s semantic type definition, which will either be Accept or Reject. We then ask the user for confirmation by showing a confirmation message box (see Figure 27.6). If OK (rather than Cancel) is pressed, then we call the processAction function to carry out the task decision process.

- processAction
  
  This is where the decision is made, and the task is approved or rejected as directed. As this is a connection to the server, you can find the description in the next section.

![Figure 27.6 The Confirmation Message Box in Action](image)

### 27.4 Frontend, Meet Backend

Now that the application’s UI is defined, it’s time to turn our attention to the workflow data itself. This is on the SAP backend, along with functions to manage it. Being an outside-in application, this is exactly where we expect it to be; our application logic that is used to interact with the user runs on the client, but the key business data is in SAP.

In the following sections, we’ll look at how to access workflow data in the SAP backend from our application’s UI.

#### 27.4.1 SAP Gateway and OData

The integration layer of choice today for ABAP stacks is SAP Gateway. If you have a 7.40 system, the SAP Gateway components will be available by default. If not, you can install SAP Gateway either directly in your enterprise system or on a
Contents

Foreword ................................................................................................................................. 29
Preface .................................................................................................................................. 33

PART I: Getting Started with Workflow in SAP

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td>45</td>
</tr>
<tr>
<td>1.1</td>
<td>SAP Business Workflow: Defined</td>
<td>45</td>
</tr>
<tr>
<td>1.2</td>
<td>SAP Intelligent Business Operations</td>
<td>48</td>
</tr>
<tr>
<td>1.3</td>
<td>How SAP Business Workflow Fits Into the SAP Overall Business Process Management Strategy</td>
<td>51</td>
</tr>
<tr>
<td>1.4</td>
<td>When to Use SAP Business Workflow</td>
<td>52</td>
</tr>
<tr>
<td>1.4.1</td>
<td>Factor 1: Human Involvement</td>
<td>53</td>
</tr>
<tr>
<td>1.4.2</td>
<td>Factor 2: Repetition</td>
<td>53</td>
</tr>
<tr>
<td>1.4.3</td>
<td>Factor 3: The Fruits of Success</td>
<td>54</td>
</tr>
<tr>
<td>1.4.4</td>
<td>Factor 4: The Cost of Failure</td>
<td>56</td>
</tr>
<tr>
<td>1.5</td>
<td>What Can Be Achieved Using SAP Business Workflow</td>
<td>57</td>
</tr>
<tr>
<td>1.6</td>
<td>Features of SAP Business Workflow</td>
<td>58</td>
</tr>
<tr>
<td>1.6.1</td>
<td>Robust Integration into SAP Applications</td>
<td>58</td>
</tr>
<tr>
<td>1.6.2</td>
<td>Workflow Builder</td>
<td>59</td>
</tr>
<tr>
<td>1.6.3</td>
<td>Graphical Status Visualization</td>
<td>60</td>
</tr>
<tr>
<td>1.6.4</td>
<td>Routing Mechanisms</td>
<td>61</td>
</tr>
<tr>
<td>1.6.5</td>
<td>Escalation Management (Deadlines)</td>
<td>62</td>
</tr>
<tr>
<td>1.6.6</td>
<td>Synchronization Management</td>
<td>62</td>
</tr>
<tr>
<td>1.6.7</td>
<td>Integration with Mobile Devices</td>
<td>63</td>
</tr>
<tr>
<td>1.6.8</td>
<td>Integration with Groupware Products</td>
<td>63</td>
</tr>
<tr>
<td>1.6.9</td>
<td>Ad Hoc Process Enhancers</td>
<td>63</td>
</tr>
<tr>
<td>1.6.10</td>
<td>Desktop Integration</td>
<td>64</td>
</tr>
<tr>
<td>1.6.11</td>
<td>Process Analytics and Dashboards</td>
<td>64</td>
</tr>
<tr>
<td>1.6.12</td>
<td>Robust Extensible Architecture</td>
<td>66</td>
</tr>
<tr>
<td>1.7</td>
<td>Some Terminology</td>
<td>66</td>
</tr>
</tbody>
</table>

2 Requirements Gathering Strategy ........................................................................... 69

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Introduction</td>
<td>69</td>
</tr>
</tbody>
</table>
2.2 Understanding the Business Process ............................................ 71
  2.2.1 Business Drivers for the Process ........................................ 71
  2.2.2 Expected Benefits of Workflow ........................................ 73
  2.2.3 How the Process Works in Theory ...................................... 75
  2.2.4 How the Process Works in Practice .................................... 75
  2.2.5 Unions and Workers’ Councils ......................................... 78

2.3 Gathering Object Data ........................................................... 78
  2.3.1 Data Needed by the Workflow ........................................ 79
  2.3.2 Where to Look for Data .................................................. 80
  2.3.3 What Data Already Exists ............................................. 81
  2.3.4 Making the Most of the Data Collected .......................... 81

2.4 Determining the Agents .......................................................... 84
  2.4.1 Who Are Your Agents? .................................................... 85
  2.4.2 Criteria for Finding Agents ............................................. 86

2.5 Determining Work Item Delivery Routes .................................. 89

2.6 Confirming the Workflow Design .......................................... 90

2.7 Workflow Administration ........................................................ 92

2.8 Planning for Production Start ............................................... 93

2.9 Planning for Future Enhancements ....................................... 94

3 Configuring the System ............................................................ 97
  3.1 Quick Start: Configure the SAP Business Workflow Engine
    in Seconds ............................................................................. 99
    3.1.1 Automatic Configuration of the Workflow Engine .......... 99
    3.1.2 Specifics of Automatic Customizing ................................. 101
    3.1.3 IMG Activities for Authorization Management .......... 104
  3.2 Task-Specific Customizing ....................................................... 105
    3.2.1 Maintaining the Organizational Plan ............................... 106
    3.2.2 Agent Assignment for Tasks ......................................... 106
    3.2.3 Activating the Triggering Events for a Workflow or
      a Task ............................................................................. 106
  3.3 Transport and Client Copy ....................................................... 107
    3.3.1 Transport of Tasks ...................................................... 107
    3.3.2 Transport of Settings for Tasks and Workflows .......... 107
    3.3.3 Transporting a Workflow Definition ............................ 108
  3.4 Customizing for the SAP Business Workflow Engine ............. 108

4 Work Item Delivery ................................................................. 111
  4.1 The Human Factor ................................................................. 111
  4.2 SAP Business Workplace ....................................................... 114
    4.2.1 Work Item Text .......................................................... 115
    4.2.2 Attachments ............................................................. 117
    4.2.3 Reserving and Replacing a Work Item ......................... 119
    4.2.4 Executing a Work Item .............................................. 119
    4.2.5 Synchronous Dialog Chains ........................................ 121
    4.2.6 Forwarding a Work Item .......................................... 122
    4.2.7 Priority ................................................................. 123
    4.2.8 Resubmission .......................................................... 124
    4.2.9 Logs ........................................................................ 124
    4.2.10 Outbox .................................................................... 126
    4.2.11 Rejecting Work Items .............................................. 126
    4.2.12 Substitution ............................................................ 127
  4.3 Universal Worklist ................................................................. 127
    4.3.1 Major Features ............................................................ 128
    4.3.2 Getting Started ........................................................... 130
    4.3.3 Working with Tasks .................................................... 131
    4.3.4 Personalization Options ............................................. 132
    4.3.5 Customizing Options ................................................ 133
    4.3.6 Action Handlers ......................................................... 133
    4.3.7 UWL Configuration Wizard ....................................... 134
    4.3.8 Universal Worklist Connectors and API and the
      SONIC Adapter .............................................................. 136
  4.4 Examples of Universal Worklist Customizing ......................... 137
    4.4.1 How to Add/Remove Columns .................................... 138
    4.4.2 How to Create Custom Columns ................................. 139
    4.4.3 How to Make a Mandatory Rejection Memo ............... 141
  4.5 Extended Notifications .......................................................... 141
    4.5.1 Types of Notifications ................................................ 142
    4.5.2 Features of Extended Notifications .............................. 142
    4.5.3 Configuration Required for Extended Notifications .... 142
  4.6 Business Workflow Work Center .......................................... 144
  4.7 Mobile Workflows ............................................................... 145
  4.8 External Users .................................................................... 146
  4.9 Other Considerations ........................................................... 147
## 5 Agents

5.1 Understanding Agent Assignment .................................................. 150
   5.1.1 Possible Agents ............................................................... 151
   5.1.2 Responsible Agents ........................................................... 154
   5.1.3 Excluded Agents ................................................................. 155
   5.1.4 Recipients .......................................................................... 156
   5.1.5 Actual Agents ..................................................................... 157
   5.1.6 Assigning Multiple Agents per Work Item .......................... 158
   5.1.7 Deadline and Notification Agents ...................................... 159

5.2 Agent Assignment Using the Organizational Structure ............... 160
   5.2.1 The Basic Organizational Plan ........................................... 160
   5.2.2 The Extended Organizational Plan .................................... 165
   5.2.3 Doing without an Organizational Plan ................................. 166

5.3 Implementing and Maintaining the Structure in Practice .......... 168
   5.4.1 Strategies for Organization Plan Maintenance .................... 168
   5.4.2 Transporting Organization Plans ....................................... 169
   5.4.3 Reassigning Work Items from One User to Another .......... 171
   5.4.4 Substituting Users ............................................................ 171
   5.4.5 When to Refresh the Organizational Environment ............. 173
   5.4.6 Training and Encouraging Agents .................................... 174

5.4 Implementing and Maintaining the Structure in Practice .......... 168
   5.4.1 Strategies for Organization Plan Maintenance .................... 168
   5.4.2 Transporting Organization Plans ....................................... 169
   5.4.3 Reassigning Work Items from One User to Another .......... 171
   5.4.4 Substituting Users ............................................................ 171
   5.4.5 When to Refresh the Organizational Environment ............. 173
   5.4.6 Training and Encouraging Agents .................................... 174

## 6 Setting Up an SAP-Provided SAP ERP Workflow .................... 177

6.1 Finding Workflows Delivered by SAP ........................................ 178
   6.1.1 Using the SAP Help Portal to Find SAP Documented Workflows ................................................................. 178
   6.1.2 Finding Workflows within Workflow Builder .................... 179
   6.1.3 Technical Workflows .......................................................... 180

6.2 Business Scenario ..................................................................... 180

6.3 SAP Workflows Involved .......................................................... 181

6.4 Starting the Workflow ................................................................ 182

6.5 Activating the Scenario ............................................................. 183

6.6 Commonly Used SAP ERP Workflows ..................................... 185

## 7 SAP Operational Process Intelligence Powered by SAP HANA ......................... 187

7.1 The Objectives of SAP Operational Process Intelligence ........... 188
   7.1.1 Process Transparency ...................................................... 188
   7.1.2 Limitless Scenario Visibility ............................................ 190
   7.1.3 Insight to Action ............................................................... 190

7.2 Prerequisites for Operational Excellence .................................. 190
   7.2.1 SAP HANA ................................................................. 191
   7.2.2 SAP Operational Process Intelligence for SAP HANA ....... 192

7.3 space.me Dashboards ............................................................... 192
   7.3.1 The Phase Diagram ....................................................... 195
   7.3.2 Measurements ................................................................. 196
   7.3.3 Key Performance Indicators ............................................ 196
   7.3.4 Tasks ........................................................................... 197
   7.3.5 Checklists ................................................................. 199
   7.3.6 Additional Context Data ................................................. 200

7.4 SAP HANA Studio ................................................................. 200
   7.4.1 Context ................................................................. 201
   7.4.2 Processes ................................................................. 201
   7.4.3 Measurements, Indicators, and Everything Else ............. 203
   7.4.4 Activating the Scenario ................................................ 204

7.5 Next Steps ............................................................................ 205

## PART II: Administering Workflows

## 8 Workflow Administration .......................................................... 209

8.1 Reporting on Workflows ......................................................... 210
   8.1.1 Reporting on Workflow Progress .................................... 210
   8.1.2 Reporting on Workflow Performance ............................... 212
   8.1.3 Work Items by Processing Duration ................................. 213
   8.1.4 Work Items with Monitored Deadlines ............................ 214
   8.1.5 Reporting on Agent Behavior .......................................... 214
   8.1.6 Identifying and Justifying Potential Improvements .......... 215

8.2 Error Resolution ..................................................................... 216

8.3 General Techniques for Resolving Runtime Errors ................... 219
   8.3.1 Basic Settings for Error Monitoring ................................. 219
8.3.2 Finding and Fixing Work Items ........................................ 220
8.3.3 Diagnosis of Workflows with Errors ............................... 220
8.3.4 Work Item Selection ..................................................... 222
8.3.5 Work Item Frequency .................................................. 223
8.4 Working with the Work Item Display .................................. 224
8.4.1 Work Item Display: Standard View ............................... 226
8.4.2 Work Item Display: Technical View ............................... 227
8.4.3 Working with the Work Item Container Display ............... 230
8.5 Working with the Workflow Log ........................................ 231
8.5.1 User View ............................................................... 232
8.5.2 Technical View .......................................................... 234
8.6 Working with the Classic Workflow Technical Log ............... 236
8.7 Resolving Agent Determination Errors ............................... 239
8.7.1 Fixing Work Items with No Agents or Wrong Agents ...... 239
8.7.2 Preventing Agent Determination Problems from Recurring .. 242
8.7.3 Support Tools for Agent Determination Problems .......... 242
8.7.4 Resolving Buffering Errors ........................................... 243
8.7.5 Fixing Work Items with Buffering Problems ..................... 244
8.7.6 Preventing Buffering Problems from Recurring ............... 245
8.7.7 Support Tools for Buffering Problems ............................ 245
8.8 Other Support Tools ....................................................... 246
8.9 Help Desk in the Intranet .................................................. 247
8.9.1 Web-Based Help Desks ................................................. 247
8.9.2 Self-Help ................................................................. 248
8.10 A Day in the Life of a Workflow Administrator .................... 249
8.10.1 Workflow Administrator Assignment ............................. 250
8.10.2 Who Makes the Business Decisions in the Event of Failure ... 252
8.10.3 Crisis Response ........................................................ 254
8.10.4 Day-to-Day Monitoring .............................................. 256
8.10.5 Periodic Reality Checks .............................................. 258
8.10.6 Housekeeping and Archiving ...................................... 259
8.10.7 Making the Most of a Precious Resource ....................... 260
9 Using SAP Business Warehouse for SAP Business Workflow Reporting .................................................. 263
9.1 SAP Business Warehouse Basics for the Workflow Expert ....... 264
9.2 Standard Workflow Analysis with SAP Business Warehouse .......... 266
9.2.1 Data Flow and Overview of Provided Business Content .................................................. 267
9.2.2 InfoCube: Aggregated Process Data .................................. 268
9.2.3 Aggregated BPA and HR Data ....................................... 269
9.2.4 Using Standard BI Content ............................................ 269
10 Administration Troubleshooting Guide .................................. 271
10.1 A Workflow That Doesn’t Start ......................................... 271
10.1.1 Did It Really Fail to Start? ........................................... 272
10.1.2 Is the Triggering Event Set Up Correctly? ......................... 272
10.1.3 Check the RFC Queue to See if the Event Raising Has Aborted Midway ...................... 273
10.1.4 Check the Consistency of the Workflow Definition .......... 274
10.1.5 Simulate the Event ...................................................... 275
10.1.6 What’s Preventing the Workflow from Starting ............... 275
10.2 A Workflow That Stops in Mid-Track .................................. 276
10.3 The Most Likely Causes (and How to Avoid Them) ............... 276
10.3.1 Background Work Items Aborting in the Middle .......... 278
10.3.2 Business Object Doesn’t Exist ...................................... 279
10.3.3 Inconsistent Definition ............................................... 280
10.3.4 Workflow Step Must Be Confirmed Manually ................. 281
10.3.5 Condition Set Up Incorrectly ...................................... 281
10.3.6 Binding Problems with the Workflow Trace .................. 281
10.3.7 Why Duplicate or Multiple Workflows Are Triggered ....... 282
10.3.8 Why an Agent Doesn’t Receive a Work Item ................. 283
10.3.9 Why the Wrong Agent Receives the Work Item ............. 285
10.3.10 Why the Work Item Follows the Wrong Route .............. 286
10.4 The Administrator’s First Aid Guide .................................. 286
10.4.1 Resolving Work Item Errors ........................................ 286
10.4.2 Resolving Workflow Instance Errors ............................. 291
10.4.3 Finding Workflows That Seem to Have Disappeared ....... 294
10.4.4 Resolving Event Linkage Errors ................................. 295
11 Advanced Diagnostics ....................................................... 299
11.1 Troubleshooting Workflows ............................................ 299
11.2 Diagnosis Logs ............................................................ 301
11.2.1 Workflow Log ........................................................ 301
11.3 Debugging with the ABAP Debugger ............................................. 308
11.3.1 Workflow Trace .............................................................. 309
11.3.2 When to Use the Workflow Trace ......................................... 314
11.3.3 Additional Trace Functionality ............................................. 314
11.3.4 Supportability Tool .......................................................... 316
11.4 Preparing a Task to Go to Production ......................................... 317

12 Upgrading SAP Business Workflow ............................................. 323
12.1 First Steps to Upgrading Workflow ............................................. 323
12.2 Steps to Take Before the Upgrade Begins .................................. 325
12.2.1 Complete All Running Workflow Instances (Where Possible) ................................................ 325
12.2.2 Create Workflow Test Plans .............................................. 325
12.2.3 WF and T Tasks versus WS and TS Tasks ............................ 326
12.2.4 Clean Runtime Tables .................................................. 326
12.3 Steps to Take During and After the Upgrade ............................ 327
12.3.1 Conversion of Event Linkage Tables ................................. 327
12.3.2 Basis Support Package .................................................. 327
12.3.3 Configure Your System for Workflow ............................... 327
12.3.4 Workflow Definition and Binding .................................... 328
12.3.5 Tables SWW_CONTOB and SWW_CONT ....................... 328
12.3.6 Changed Locking Behavior of Runtime ............................. 330
12.3.7 Other Issues after the Upgrade ....................................... 331
12.4 Important SAP Notes .......................................................... 332

PART III: Developing Workflows

13 Creating a Workflow ............................................................. 337
13.1 Workflow Builder Basics ...................................................... 338
13.1.1 Look and Feel of the Workflow Builder ............................... 338
13.1.2 Building Your First Workflow ......................................... 342
13.1.3 Saving, Activating, and Testing ....................................... 345
13.2 Enhancing Your Workflow ................................................... 347
13.2.1 Deadline Monitoring ....................................................... 347

14 Advanced Workflow Design Techniques .................................... 381
14.1 Step Conditions .................................................................. 381
14.1.1 Step Condition: Create Work Item .................................. 384
14.1.2 Step Condition: Complete Work Item ............................... 384
14.2 How to Implement Parallel Processing ................................... 384
14.2.1 How to Implement Table-Driven Dynamic Parallel Processing .................................................. 386
14.2.2 How to Implement a Fork .............................................. 390
14.3 Reusing Workflows as Subworkflows ................................... 391
14.4 Modeled Deadlines ............................................................ 392
14.4.1 Notify Recipients via Email That a Deadline Has Been Missed ................................................ 393
14.4.2 Skip the Work Item If a Deadline Has Been Missed .......... 393
14.5 Containers .......................................................................... 394
14.5.1 Containers and Bindings in Depth ................................... 395
14.5.2 Reimplementation of the Container ................................. 399
PART IV: Enhancing Workflows

20 Service-Enabling Workflows .......................................................... 627
  20.1 Workflow in the Context of a Business Process ....................... 627
  20.2 Web Services and Service-Oriented Architecture ..................... 629
    20.2.1 Web Services Defined .................................................... 629
    20.2.2 Web Services Description Language ............................... 630
    20.2.3 Web Services and SOAP ............................................... 631
  20.3 SOA at SAP ............................................................................. 632
  20.4 Implementing the Service Interface for a Workflow ................... 636
    20.4.1 The Function Module .................................................... 637
    20.4.2 Service-Enabling in the Application (Inside-Out Approach) ........ 638
    20.4.3 Service-Enabling from the Enterprise Services Repository (Outside-In Approach) .................................................. 643

21 BRFplus and SAP Decision Service Management .......................... 659
  21.1 Introduction to Business Rules and Decision Services .............. 660
    21.1.1 Business Rules Scenarios in Workflows ......................... 662
    21.1.2 Business Rules Engines in SAP ........................................ 662
  21.2 Overview of Decision Services in SAP DSM/BRFplus ............... 664
  21.3 Calling Decision Services from SAP Business Workflow .......... 667
  21.4 Example: Decision Service for Adaptable Deadlines .............. 669
  21.5 Example: Decision Service for Agent Assignment .................. 673

22 User Interface Options ................................................................. 681
  22.1 Inbox and the User Interface .................................................. 682
  22.2 Containers, Bindings, and Data Flow ...................................... 684
    22.2.1 Where the Customized UI Fits in with the Data Flow When Using UWL ................................................................. 684
    22.2.2 Common Steps When Using Web Dynpro and BSP with UWL ................................................................. 686

23 Using Web Dynpro ABAP .............................................................. 689
  23.1 Example Based on User Decision ............................................ 690
    23.1.1 Standard User Decision .................................................. 690
    23.1.2 User Decision with Note in Universal Worklist .................. 691
    23.1.3 User Decision with Web Dynpro ABAP ............................ 694
  23.2 Required Settings in Web Dynpro ABAP .................................. 697
    23.2.1 Web Dynpro ABAP Application ...................................... 698
    23.2.2 Window ................................................................. 701
    23.2.3 Main View .................................................................... 702
  23.3 Configuration for the Universal Worklist ................................ 705
  23.4 Ensuring the User Only Executes the Work Item in the Universal Worklist .................................................. 706
    23.4.1 Implementing a Message by the Execution from SAP Business Workplace .................................................. 707
    23.4.2 Hiding the Work Item from SAP Business Workplace ....... 707

24 Using Web Dynpro Java ............................................................... 709
  24.1 Real-World Example of Web Dynpro Java and Workflow .......... 710
  24.2 Setting Up the Java Development Environment .................... 712
  24.3 Tasks on the Java Side ........................................................ 716
    24.3.1 Presentation Layer ....................................................... 716
    24.3.2 Model Layer ............................................................. 718
  24.4 Tasks in the Universal Worklist ............................................. 721

25 Using Business Server Pages ...................................................... 725
  25.1 Building the Example BSP and Workflow .............................. 725
    25.1.1 Laying the Groundwork .................................................. 725
    25.1.2 Setting Up the WebFlow Service Handler ........................ 733
  25.2 Launching the BSP and Transferring Data ............................. 734
    25.2.1 Defining the Workflow Service and Creating a Task .......... 734
    25.2.2 Creating a Test Workflow .............................................. 736
    25.2.3 Launching a BSP Application from the SAP Business Workplace .................................................. 738
    25.2.4 Launching a BSP Application from Another Web Application .................................................. 738
PART V: Using SAP Business Workflow in SAP Applications

28 ArchiveLink ........................................................................................................... 787
  28.1 What Is ArchiveLink? ...................................................................................... 788
  28.1.1 Technical View of ArchiveLink ............................................................... 788
  28.1.2 Business View of ArchiveLink .................................................................. 790
  28.2 ArchiveLink Standard Scenarios ..................................................................... 791
  28.2.1 How SAP Business Workflow Integrates ArchiveLink ......................... 792
  28.2.2 Workflow-Based Document Scenario: Store for
       Subsequent Entry (Previously “Early Archiving”) ..................................... 792
  28.2.3 Workflow-Based Document Scenario: Store for
       Subsequent Assignment (Previously “Late Archiving”) ........................... 793
  28.3 BOR Object Types ......................................................................................... 794

29 SAP Supplier Relationship Management ......................................................... 797
  29.1 Approval Processes ......................................................................................... 800
  29.1.1 Simple Approval Processes ...................................................................... 800
  29.1.2 Complex Approval Processes ................................................................. 800
  29.1.3 Key Concepts and Requirements .............................................................. 801
  29.2 SAP SRM Approval Frameworks ..................................................................... 806
  29.3 Process-Controlled Workflow in Detail .......................................................... 808
     29.3.1 Technical Background ......................................................................... 811
     29.3.2 Basic Assumptions for Document Processing ....................................... 813
     29.3.3 Configuration of Approval Levels ......................................................... 813
     29.3.4 Business Rules Framework ................................................................... 816
     29.3.5 Agent Determination ............................................................................ 817
     29.3.6 BAdI Implementations for Agent Determination ................................... 819
     29.3.7 Approval by Document Owner ............................................................... 829
     29.3.8 Shopping Cart Modifications during Sourcing ....................................... 830
     29.3.9 Parallel Approval with Overlapping Responsibility ............................. 832
     29.3.10 Approval Process Overview: User-Focused Process
            Visualization .......................................................................................... 833
     29.3.11 Process-Controlled Workflow: Advanced
            Troubleshooting ...................................................................................... 837
  29.4 Real-World Example of a Process-Controlled Workflow ............................... 841
     29.4.1 Simplest Scenario: Execute without Approval ....................................... 841
     29.4.2 Configuring Two-Step Approval ......................................................... 842
     29.4.3 Details on BRF Integration .................................................................. 843
### 29.5 Application-Controlled Workflow in Detail

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.5.1 One-/Two-Step Approval</td>
<td>847</td>
</tr>
<tr>
<td>29.5.2 n-step BAdI</td>
<td>848</td>
</tr>
<tr>
<td>29.5.3 SAP-Provided BAdls and Workflow Templates</td>
<td>849</td>
</tr>
<tr>
<td>29.5.4 Line Item-Based BAdl</td>
<td>850</td>
</tr>
<tr>
<td>29.5.5 Approvers Changing Documents in Application-Controlled Workflow</td>
<td>850</td>
</tr>
<tr>
<td>29.5.6 Configuring Approval Processes</td>
<td>850</td>
</tr>
<tr>
<td>29.5.7 Agent Determination</td>
<td>853</td>
</tr>
<tr>
<td>29.5.8 Ad Hoc Agent Assignment</td>
<td>854</td>
</tr>
<tr>
<td>29.5.9 Implementing Dynamic Processes</td>
<td>856</td>
</tr>
<tr>
<td>29.5.10 Tracking Workflow Instances</td>
<td>861</td>
</tr>
<tr>
<td>29.5.11 Application-Controlled Workflow Advanced Troubleshooting</td>
<td>861</td>
</tr>
<tr>
<td>29.6 Real-World Example of Application-Controlled Workflow</td>
<td>863</td>
</tr>
<tr>
<td>29.6.1 Without Approval</td>
<td>863</td>
</tr>
<tr>
<td>29.6.2 Two-Step Approval</td>
<td>863</td>
</tr>
<tr>
<td>29.7 Inbox and UWL</td>
<td>865</td>
</tr>
<tr>
<td>29.8 Offline Functionality</td>
<td>866</td>
</tr>
<tr>
<td>29.8.1 Outbound Processing</td>
<td>867</td>
</tr>
<tr>
<td>29.8.2 Offline Approval by the Recipient</td>
<td>867</td>
</tr>
<tr>
<td>29.8.3 Inbound Processing</td>
<td>868</td>
</tr>
<tr>
<td>29.9 Deadline Monitoring</td>
<td>868</td>
</tr>
<tr>
<td>29.10 Recommendations When Getting Started</td>
<td>870</td>
</tr>
<tr>
<td>29.10.1 Upgrading a System</td>
<td>871</td>
</tr>
<tr>
<td>29.10.2 Archiving</td>
<td>873</td>
</tr>
<tr>
<td>29.10.3 New Installation</td>
<td>873</td>
</tr>
</tbody>
</table>

### 30 SAP Customer Relationship Management

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>30.1 Introduction</td>
<td>878</td>
</tr>
<tr>
<td>30.1.1 One Order Concept</td>
<td>879</td>
</tr>
<tr>
<td>30.1.2 Business Transaction and SAP CRM Business Objects</td>
<td>881</td>
</tr>
<tr>
<td>30.1.3 Transaction History</td>
<td>883</td>
</tr>
<tr>
<td>30.1.4 SAP CRM UI Framework</td>
<td>884</td>
</tr>
<tr>
<td>30.1.5 SAP Enterprise Portal Integration</td>
<td>887</td>
</tr>
<tr>
<td>30.2 Customizing and Workflow Administration</td>
<td>887</td>
</tr>
<tr>
<td>30.2.1 Workflow Customizing</td>
<td>887</td>
</tr>
<tr>
<td>30.2.2 Workflow Administration</td>
<td>887</td>
</tr>
</tbody>
</table>

### 31 SAP ERP Human Capital Management—Processes and Forms

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>31.1 Workflow within the HCM Processes and Forms Framework</td>
<td>936</td>
</tr>
<tr>
<td>31.1.1 Special Features of the Workflow Templates</td>
<td>936</td>
</tr>
<tr>
<td>31.1.2 Special Features of the Standard Tasks</td>
<td>937</td>
</tr>
<tr>
<td>31.1.3 Requirements for an HCM Processes and Forms</td>
<td></td>
</tr>
<tr>
<td>Workflow Template</td>
<td>938</td>
</tr>
<tr>
<td>31.2 Standard Workflows in HCM Processes and Forms</td>
<td>939</td>
</tr>
</tbody>
</table>
Index

A

ABAP class, 352, 408, 471, 567, 574, 797, 883, 885, 894, 896, 897
BI_PERSISTENT~LPOR, 495
business class, 492
CL_CRM_MKTPL_APPL_BASE, 889
CL_HRASR00_PROCESS_OBJECT, 938
CL_SWF_FORMABSENC, 472, 515
constant, 478
events, 517
exception class, 506
FIND_BY_LPOR, 498
functional method, 490
GUID, 497
instance management, 501, 527
private, 483
protected, 483
public, 483
public static constant, 514
SAPSRM/CL_WF_PDO_P0, 798
static, 478
structured key, 497
utility class, 479, 485, 490
ABAP Data Dictionary, 328
data type, 365
reference, 365
ABAP Debugger, 308
ABAP expression, 490
ABAP Objects — see ABAP class
AC_CONTAINER, 551, 552
Access sequence, 577
Action, 954
handler, 723
Activation, 217
event, 183
linkages, 100
SAP-provided workflow, 177
Active version, 377
Activities, 369
log, 99
step, 350

ACTOR_TAB, 552
Ad hoc
agent, 542, 835
agent assignment, 404, 405, 537, 854
anchor, 369, 854, 1049
anchor step, 405
approver, 836
features, 404
process, 799
process extension, 803
workflow, 63, 125, 233, 404, 593
workflow extension, 405
Ad hoc workflow
add reviewers, 855
additional approvers, 854
Adaptive RFC model, 712
Administrative standard workflow, 901
Administrator, 86, 168, 209, 594
default, 103
skills, 250
Adobe
Designer, 748
Advance
with dialog, 347, 357, 376
After method, 1040
Agent, 67, 70, 84, 221, 300, 386, 964
actual, 157
assignment, 158
assignment for tasks, 106
deadline, 159
determination, 157
determination for governance, 1005
excluded, 150, 155, 284, 285, 862
parallel processing in SAP SRM, 832
people, 85
possible, 106, 107, 150, 151, 535, 594, 610
possible, assignment, 153
report on behavior, 214
responsible, 150, 154, 818, 864
restrict assignment, 1038
selected, 150, 156, 535, 800
troubleshoot, 284
Agent (Cont.)
troubleshoot, doesn't receive work item, 283
wait times, 212
Agent assignment, 908
ad hoc, 854
Agent determination, 76, 84, 227, 239, 282, 283, 285, 317, 318, 817, 864, 887, 889, 946
BAdI implementations, 819
error, 218
error handling, 239
prevent problems, 242
rule, 889, 894
SAP SRM, 853
support tools, 242
Agent determination rule, 409, 535, 536, 537
container, 552, 558
defining, 537
exception, 552
expression, 559
function module, 551
manual selection, 542
result, 542, 543, 544
using, 542
Alert, 128, 267
Alloy, 113
Anchor
ad hoc, 369
step, 1049
API, 608
Application
service enabling, 638
Application programming interface → API
Application-controlled workflow, 806, 847
advanced troubleshooting, 861
Approach
inside-out, 638
outside-in, 643
Approval
by document owner, 829
delivered scenarios, 852
two-step, 842
workflow, 958
Approval app, 984
add, 987
Approval level, 803
configuration, 813
Approval Procedure wizard, 1029
Approval process, 46, 799
configure, application-controlled workflow, 850
job role, 802
overview, 840
reviewer, 836
user-focused process visualization, 833
Approve All user interface, 988
Archive Development Kit, 791
ArchiveLink, 787, 1041
equivalent configuration, 789
tasks, 790
third-party software, 789
Archiving, 93, 259, 791, 873
workflow, 324
Association, 414
Asynchronous, 438, 587
method, 353, 458, 589
task, 121, 280, 358
Attachment, 81, 115, 117, 118, 120, 131, 1034
SAP business object, 118
Attribute, 227, 410, 412, 478, 486, 505
add to user decision step, 361
based on object references, 454
buffer, 452
class, 487
create, 434
custom, 466
database, 435
database, add, 437
details, 419
instance-independent, 434
multilang, 452, 455
programming, 450
virtual, 419, 446, 450
vs. functional method, 505
Audit trail, 124
Auditor, 87
Authorization, 102, 104, 151, 239, 240, 321
HR structural, 278, 284
problem, 258
role, 105
Authorization (Cont.)
structural, 105, 166
Automatic forwarding, 220
Automatic workflow customizing, 97, 101, 102, 303
active version, 101
plan version, 101
B
Background
method, 276, 278, 459
processing, 287, 319, 375
step, 300, 949, 950
task, 103, 223, 600, 1003, 1031
work item, 224, 288, 602
Background job
missing deadline, 103
BBP_WFL_SECURE_BADI, 849
BAdI, 287, 571, 615, 810, 819, 862, 864, 889, 924
agent determination, 818, 1005
application-controlled workflow, 849
BBP_WFL_APPROVE_BADI, 949, 845, 857, 859, 860
BBP_WFL_SECURE_BADI, 849
dynamic approval processes, 857
example configuration, 849
n-step, 848
SAPSRM/BD_WF_DEADLINES, 869
SAPSRM/BD_WF_RESP_RESOLVER, 819
SAPSRM/BD_WF_REVIEWER_F4, 837
SAPSRM/BD_WF_REVIEWER_RULE, 837
SAPSRM/BD_WF_RESP_RESOLVER, 819
WF_Core_RESP_RESOLVER, 819
BAPI, 410, 438
Basic data, 338
Batch job, 614
cleaning up, 259
customizing, 108
monitoring deadline, 103
BC Set
activate for process-controlled workflow, 874
customization, 842
customize, 875
one-step approval, 843
Before method, 1040
Benefit, 111
BI content, 269
Binding, 67, 310, 311, 314, 324, 328, 337, 338, 363, 381, 395, 397, 434, 542, 582, 584, 603, 630, 720, 938
automatic, 1034
definition, 363
definitions within workflow, 604
detailed explanation, 395
error, 221, 292
incorrect, 279
problem, 287, 297
problem with workflow trace, 281
UI, 684
BOL, 879, 884
framework, 893, 895
browser, 415, 416
BUS1006, 353, 355, 566
BUS2002, 353
object method, 896
one-/two-step approvals, 853
SOFA, 749
WF_DOC, 748
BOR object, 140
add interface, 432
add method to, 440
association, 414
attribute, 412
BUS1001, 428
BUS2000, 904
BUS2001, 904
BUS2001, 904
BUS2001, 904
BUS2012, 440
BUS2121, 466, 468
composition, 414
create, 426
create as subtype, 430
create from scratch, 429
create, example, 428
create, methods, 427
customizing, 466
capsulation, 410
event, 412
external, 427
external via delegation, 431
Index

Index
BOR object (Cont.)
FORMABSENC, 465
hierarchical view, 416
IFARCH21, 794
IMAGE, 795
inheritance, 410
inheritance relationship, 413
interface, 412, 415, 794
key, 412, 433
link with business documents, 790
macros, 448
method, 412
overview, 409
polymorphism, 410
predefined, 464
program, 422
programming, 447
relationships, 412
SELFITEM, 464
status, 428
subtype, 414
STSTEM, 465
text, 422
used for training, 417
USR01, 465, 472, 493, 512
view existing, 417
WF_TASK, 465
BOR object component
create, 431
create from scratch, 432
BPM → SAP Business Process Management
Branch, 369
Breakpoint
external, 309
BRF, 807
evaluation ID, 844, 846
evaluation results, 841
event, 817
expression, 817
integration with SAP SRM, 843
process level, 809
SRP SRM, 816
BRFplus, 956, 994
business rules, 662
decision services, 664
BFplus (Cont.)
decision tables, 1007, 1023
enhance context, 1015
integration with SAP Business Workflow, 667
MSMPS, 958
process and user determination, 1010
rules-based workflow with SAP Master Data Governance, 1007
BFplus Workbench, 664
BSP, 682, 696, 725, 726
application, 725, 729, 731, 734, 738, 740
importing parameters, 739
launch app from another app, 738
view results, 736
Buffer
error, 243
fix problems, 244
prevent problems, 245
support tools, 245
Buffering, 243, 245, 558
error, 218
problem, 549
Business Add-in → BAdI
Business Application Programming Interface → BAPI
Business applications
object-oriented view, 409
Business Communications Services, 625, 867
Business Configuration Set → BC Set
Business document, 787
integrate with SAP application data, 787
Business driver, 71
Business intelligence → B1
Business object, 78, 308, 338
ASPBURLST, 864
BUS20001, 880, 881
BUS2010022, 902
BUS20323, 388
BUS21211, 797, 821, 844, 851, 852
BUS22200, 828
BUS22201, 822
BUS22202, 827
BUS22203, 824
BUS22205, 826
BUS4101, 849
collect data, 82
Business object (Cont.)
collect data into, 81
doesn’t exist, 279
FORMABSENC, 1049
Selfitem, 1031, 1032
SOFM, 1035
STSTEM, 1038, 1040, 1042, 1043
STSTEM.GenericInstantiation, 1043
USR01, 1037
WEBSERVICE, 740
Business Object Builder, 415, 573
collect data program, 447
generate macro code, 457
interfaces, 432
Business Object Layer → BOL
Business Object Repository Browser, 416
Business Object Repository → BOR
Business object type, 583, 589
Business process, 71, 78, 252, 563
engine, 49
model, 965
owner, 253
Business rules
generators, 660
generators in SAP, 662
scenarios in workflow, 662
Business Server Pages → BSP
Business transaction, 879
event, 518
Business transaction event (BTE), 571, 578
Business user, 352
Business Workflow Explorer, 58
Business Workflow Workplace inbox, 113
Business workspace, 346
C
Callback
dialog, 733
handler, 733
URL, 730, 738, 741
Canceled status, 606
Change
management, 95
Change document, 518, 573
CDHDR, 574
Change request, 995
actions, 1017
activate, 1004
blueprint process, 1015
chek, 1004
code background activities, 1015
create workflow, 1020
define step flow, 1011
dialog tasks, 1002
disable custom workflow template, 1001
implement process, 1019
status, 998, 1017
step type, 997, 1018
Change request type, 996
BFplus application, 1010
predefined, decision table content, 1011
step, 996
workflow template, 996
Characteristics, 268
Check function module, 273, 283, 297, 306, 320, 617, 618, 619, 620
Checklist, 199
Cinderella effect, 244
Circular Distribution wizard, 1030
CLASS.CONSTRUCTOR, 488
Clearing task, 259
Client copy, 107
Commit, 614
COMMIT WORK, 272, 610
Common problems, 278
Complete manually, 224
Complete workflow, 613
Complete transaction, 354
Complex data type, 397
Composition, 414
Condition, 312, 320, 369, 577, 587, 617
alias, 1010, 1018
multiple, 392
record, 577
simulation, 320
start, 306
start condition, 584
Confirm
end of processing, 375, 481, 485
end of processing flag, 897
<table>
<thead>
<tr>
<th>Page</th>
<th>Consistency check, 274, 322</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CONSTRUCTOR, 488</td>
</tr>
<tr>
<td></td>
<td>CONSTRUCTOR method, 494</td>
</tr>
<tr>
<td></td>
<td>Container, 67, 289, 324, 338, 394, 395, 603, 604, 620</td>
</tr>
<tr>
<td></td>
<td>_ATTACH_OBJECTS, 1034</td>
</tr>
<tr>
<td></td>
<td>agent determination rule, 538</td>
</tr>
<tr>
<td></td>
<td>definition, 363, 364</td>
</tr>
<tr>
<td></td>
<td>detailed explanation, 395</td>
</tr>
<tr>
<td></td>
<td>displaying, 230</td>
</tr>
<tr>
<td></td>
<td>element, 546, 551</td>
</tr>
<tr>
<td></td>
<td>event, 395</td>
</tr>
<tr>
<td></td>
<td>method, 395, 1040</td>
</tr>
<tr>
<td></td>
<td>modifying, 231</td>
</tr>
<tr>
<td></td>
<td>operation, 328, 369, 600</td>
</tr>
<tr>
<td></td>
<td>operation step, 366</td>
</tr>
<tr>
<td></td>
<td>purpose, 363</td>
</tr>
<tr>
<td></td>
<td>read current values, 612</td>
</tr>
<tr>
<td></td>
<td>rule, 395, 396</td>
</tr>
<tr>
<td></td>
<td>runtime, 463</td>
</tr>
<tr>
<td></td>
<td>task, 395, 397</td>
</tr>
<tr>
<td></td>
<td>type, 395</td>
</tr>
<tr>
<td></td>
<td>UI, 684</td>
</tr>
<tr>
<td></td>
<td>WAPI, 699</td>
</tr>
<tr>
<td></td>
<td>workflow, 395</td>
</tr>
<tr>
<td></td>
<td>Container element, 234, 366, 397, 589, 1032, 1035</td>
</tr>
<tr>
<td></td>
<td>_RULE_RESULT, 1035</td>
</tr>
<tr>
<td></td>
<td>_WIL_Group_ID, 227</td>
</tr>
<tr>
<td></td>
<td>_WIL_Object_ID, 227, 398</td>
</tr>
<tr>
<td></td>
<td>change, 366</td>
</tr>
<tr>
<td></td>
<td>multiline, 386, 387</td>
</tr>
<tr>
<td></td>
<td>Content server, 792</td>
</tr>
<tr>
<td></td>
<td>Context data, 191</td>
</tr>
<tr>
<td></td>
<td>Contract, 823</td>
</tr>
<tr>
<td></td>
<td>Cost saving, 55</td>
</tr>
<tr>
<td></td>
<td>Create event, 320</td>
</tr>
<tr>
<td></td>
<td>Create task, 351</td>
</tr>
<tr>
<td></td>
<td>Crisis response, 254</td>
</tr>
<tr>
<td></td>
<td>CRM -&gt; SAP CRM</td>
</tr>
<tr>
<td></td>
<td>Cross-component Business Process Management, 267</td>
</tr>
<tr>
<td></td>
<td>Custom code, 329</td>
</tr>
<tr>
<td></td>
<td>Custom process, 951</td>
</tr>
<tr>
<td></td>
<td>Custom relationship, 556</td>
</tr>
<tr>
<td></td>
<td>Customizing, 97, 303, 327</td>
</tr>
<tr>
<td></td>
<td>activities, 108</td>
</tr>
<tr>
<td></td>
<td>automatic, 97, 99, 303, 327</td>
</tr>
<tr>
<td></td>
<td>SAP-provided workflow, 177</td>
</tr>
<tr>
<td></td>
<td>task-specific, 105</td>
</tr>
<tr>
<td>D</td>
<td>Dashboard, 188</td>
</tr>
<tr>
<td></td>
<td>space.me, 192</td>
</tr>
<tr>
<td></td>
<td>Data management, 751</td>
</tr>
<tr>
<td></td>
<td>needed for workflow process, 82</td>
</tr>
<tr>
<td></td>
<td>quality, 76</td>
</tr>
<tr>
<td></td>
<td>to include in workflow, 84</td>
</tr>
<tr>
<td></td>
<td>Data Store Object (DSO), 265, 267</td>
</tr>
<tr>
<td></td>
<td>Data type, 644</td>
</tr>
<tr>
<td></td>
<td>complex, 397</td>
</tr>
<tr>
<td></td>
<td>Database attribute, 436, 453, 455</td>
</tr>
<tr>
<td></td>
<td>reorganization, 259</td>
</tr>
<tr>
<td></td>
<td>Deadline, 62, 73, 213, 229, 322, 332</td>
</tr>
<tr>
<td></td>
<td>add to process, 349</td>
</tr>
<tr>
<td></td>
<td>agent, 159</td>
</tr>
<tr>
<td></td>
<td>customizing, 103</td>
</tr>
<tr>
<td></td>
<td>defining, 347</td>
</tr>
<tr>
<td></td>
<td>job, 290</td>
</tr>
<tr>
<td></td>
<td>latest end, 348</td>
</tr>
<tr>
<td></td>
<td>latest start, 348</td>
</tr>
<tr>
<td></td>
<td>message, 350</td>
</tr>
<tr>
<td></td>
<td>modeled, 348, 392, 1047</td>
</tr>
<tr>
<td></td>
<td>monitoring, 213, 214, 347</td>
</tr>
<tr>
<td></td>
<td>monitoring for work item, 246</td>
</tr>
<tr>
<td></td>
<td>notification, 258</td>
</tr>
<tr>
<td></td>
<td>reporting, 214</td>
</tr>
<tr>
<td></td>
<td>requested end, 348</td>
</tr>
<tr>
<td></td>
<td>requested start, 229, 348</td>
</tr>
<tr>
<td></td>
<td>specific type, 349</td>
</tr>
<tr>
<td></td>
<td>work item, 159, 223, 602</td>
</tr>
<tr>
<td></td>
<td>Deadline monitoring</td>
</tr>
<tr>
<td></td>
<td>SAP CRM, 868</td>
</tr>
<tr>
<td></td>
<td>Deadline object, 1043</td>
</tr>
<tr>
<td></td>
<td>use in workflow, 1047</td>
</tr>
<tr>
<td></td>
<td>Debugger ABAP, 308</td>
</tr>
<tr>
<td></td>
<td>Debugging, 318</td>
</tr>
<tr>
<td></td>
<td>Decision service, 660, 667</td>
</tr>
<tr>
<td></td>
<td>adaptable deadlines, 669</td>
</tr>
<tr>
<td></td>
<td>agent assignment, 673</td>
</tr>
<tr>
<td></td>
<td>call from SAP Business Workflow, 667</td>
</tr>
<tr>
<td></td>
<td>Decision set, 806, 827</td>
</tr>
<tr>
<td></td>
<td>Decision table</td>
</tr>
<tr>
<td></td>
<td>single value, 1011</td>
</tr>
<tr>
<td></td>
<td>Decision types, 815</td>
</tr>
<tr>
<td></td>
<td>Default agent-determination rule, 154</td>
</tr>
<tr>
<td></td>
<td>Default rule, 354</td>
</tr>
<tr>
<td></td>
<td>Default workflow administrator, 103</td>
</tr>
<tr>
<td></td>
<td>Delegation, 414, 431, 696</td>
</tr>
<tr>
<td></td>
<td>Delete work item history, 260</td>
</tr>
<tr>
<td></td>
<td>workflow, 260</td>
</tr>
<tr>
<td></td>
<td>Delta pull, 131</td>
</tr>
<tr>
<td></td>
<td>Design phase, 352</td>
</tr>
<tr>
<td></td>
<td>process-oriented, 633</td>
</tr>
<tr>
<td></td>
<td>Destination logical, 102</td>
</tr>
<tr>
<td></td>
<td>Decision view declaration, 770</td>
</tr>
<tr>
<td></td>
<td>Development component (DC), 713</td>
</tr>
<tr>
<td></td>
<td>Diagnosis buffer synchronization, 243</td>
</tr>
<tr>
<td></td>
<td>log, 301</td>
</tr>
<tr>
<td></td>
<td>transaction, 302</td>
</tr>
<tr>
<td></td>
<td>utility, 167</td>
</tr>
<tr>
<td></td>
<td>Diagnostics, 299</td>
</tr>
<tr>
<td></td>
<td>log, 301</td>
</tr>
<tr>
<td></td>
<td>tool, 300</td>
</tr>
<tr>
<td></td>
<td>Dialog tasks, 1002</td>
</tr>
<tr>
<td></td>
<td>work item, 223, 287, 603, 605, 606</td>
</tr>
<tr>
<td></td>
<td>Dialog chain synchronous, 121, 594, 613</td>
</tr>
<tr>
<td></td>
<td>Document archive, 788</td>
</tr>
<tr>
<td></td>
<td>electronic, 793</td>
</tr>
<tr>
<td></td>
<td>finder, 789</td>
</tr>
<tr>
<td></td>
<td>from template, 369, 744</td>
</tr>
<tr>
<td></td>
<td>incoming, 791</td>
</tr>
<tr>
<td></td>
<td>link business objects, 790</td>
</tr>
<tr>
<td></td>
<td>processing, 813</td>
</tr>
<tr>
<td></td>
<td>storage, 789, 791</td>
</tr>
<tr>
<td></td>
<td>template, 341, 747</td>
</tr>
<tr>
<td>E</td>
<td>ECL, 789</td>
</tr>
<tr>
<td></td>
<td>Electronic document, 793</td>
</tr>
<tr>
<td></td>
<td>Email, 114, 287, 393, 596, 1031</td>
</tr>
<tr>
<td></td>
<td>attachment, 1034</td>
</tr>
<tr>
<td></td>
<td>customizing send, 1036</td>
</tr>
<tr>
<td></td>
<td>interfaces for sending, 625</td>
</tr>
<tr>
<td></td>
<td>notification template, 964</td>
</tr>
<tr>
<td></td>
<td>notification/rejections, 1037</td>
</tr>
<tr>
<td></td>
<td>send mail, recipients, 1035</td>
</tr>
<tr>
<td></td>
<td>Encapsulation, 410</td>
</tr>
<tr>
<td></td>
<td>End date/time latest, 869</td>
</tr>
<tr>
<td></td>
<td>requested, 869</td>
</tr>
<tr>
<td></td>
<td>End of processing confirm, 375</td>
</tr>
<tr>
<td></td>
<td>Engineering Client Viewer -&gt; ECL</td>
</tr>
<tr>
<td></td>
<td>Enterprise Service, 628</td>
</tr>
<tr>
<td></td>
<td>bundle, 635</td>
</tr>
<tr>
<td></td>
<td>create from function module, 638</td>
</tr>
<tr>
<td></td>
<td>mediated calls, 643</td>
</tr>
<tr>
<td></td>
<td>runtime, 638</td>
</tr>
<tr>
<td></td>
<td>test, 642</td>
</tr>
<tr>
<td></td>
<td>trigger workflow, 627</td>
</tr>
<tr>
<td></td>
<td>Enterprise Service Bus, 633</td>
</tr>
<tr>
<td></td>
<td>Enterprise Services Builder, 645</td>
</tr>
<tr>
<td></td>
<td>Enterprise Services Repository, 635, 643</td>
</tr>
<tr>
<td></td>
<td>service interface, 643</td>
</tr>
<tr>
<td></td>
<td>Entityset, 775</td>
</tr>
<tr>
<td></td>
<td>binding, 782</td>
</tr>
<tr>
<td>Document (Cont.)</td>
<td>viewer, 788</td>
</tr>
<tr>
<td></td>
<td>workflow-based, 793</td>
</tr>
<tr>
<td></td>
<td>Documentation, 182, 300, 376, 422</td>
</tr>
<tr>
<td></td>
<td>workflow definition, 376</td>
</tr>
<tr>
<td></td>
<td>DSM Workbench, 664</td>
</tr>
<tr>
<td></td>
<td>Duet, 113</td>
</tr>
<tr>
<td></td>
<td>Dynamic loop, 386</td>
</tr>
<tr>
<td></td>
<td>Dynamic parallel processing, 159, 385, 386, 1041</td>
</tr>
<tr>
<td></td>
<td>Dynamic parameter, 722</td>
</tr>
<tr>
<td></td>
<td>Dynamic process, 803</td>
</tr>
<tr>
<td></td>
<td>adaptation, 860</td>
</tr>
</tbody>
</table>
Error
administration, 216, 219
agent determination, 218
buffering, 218
event linkage, 218
determination, 218
handling, 218
monitoring, 219
real-world examples, 254
resolution, 216
resolve in production system, 286
status, 217
types, 445
work item, 218

Error Overview report, 216

Escalation, 62, 79, 81, 94, 348
Evaluation path, 537, 540, 541, 556
Event, 106, 412, 479, 517, 564
activation, 106
binding, 274, 297
business partner workflow, 583
business transaction event, 578
cache, 307, 351, 352
change document, 352
container, 395, 573
container element, 573, 582
create, 320, 446
creation, 571
creator, 369, 565, 581
definition, 570
definition, 422, 446, 564, 570
generate with WAPI, 609
handler, 519
initiator, 297, 564, 582
instance linkage, 359
linkage, 101, 409
local, 590
manager, 620
master data change, 577
material change, 574
message control, 576
name, 564, 589
parameter, 564, 573
parked, 591
queue, 275, 280, 295, 303, 304, 590,
592, 613
raise via function module, 614

Exchange Infrastructure → SAP Process
Orchestration

Event (Cont.)
READY_FOR_RELEASE, 841
READY_FOR_WORKFLOW, 812
receiver, 519, 565, 583, 588
simulation, 275, 283, 320
snapshots, 605
start, 583
start condition, 581, 584
start workflow, 337
start workflow with, 613
status change, 575
terminating, 580, 587, 588, 617
trace, 273, 303, 304, 305, 520, 568,
574, 593
triggering, 180, 386, 580, 581
triggering step, 369
type linkage, 588
wait for, 589
waiting for, 370
work item, 602
workflow API, 580
Event linkage, 183, 272, 275, 283, 295, 296,
297, 298, 565, 582, 583, 588, 589, 590,
592, 887, 902, 938
advanced, 619
error, 218
not activated, 818
processing in table, 620
table, 327, 566
Event raising, 571
business transaction event, 578
change document, 571
human resources, 577
message control, 571
SAP ERP Financials, 578
status management, 571
Event type linkages, 1000
Exception, 442, 444, 460, 853
send to workflow, 460
Exception class
CX_BO_ABORT, 507
CX_BO_ERROR, 507
CX_BO_TEMORARY, 507
Exception handling, 391
Exchange Infrastructure → SAP Process
Orchestration

Excluded agent, 284, 285, 862
Execute
rule for work item, 242
without agent check, 152
work item without agent check, 243
Export parameter, 444
Expression, 116, 338, 349
agent determination, 155
Extended notification, 113, 141, 142, 143
Extended organizational plan, 165
External breakpoint, 309
External user, 146

Factory calendar, 491, 673, 1043, 1044
Flowchart, 90
Fork, 369, 385, 390
Fork definition
n of m logic, 390
Form, 369, 746
electronic, 743
PC document, 747
simple, 744, 745
start, 747
Forum, 94
Forward task, 898
Forwarding, 115, 122, 150
work item, 122
Function module, 418, 537
call with Java code, 721
check, 273, 283, 306, 320, 409, 585, 617,
618, 619, 620
exceptions, 461
receiver type, 585, 619, 621, 622
remote enabled, trigger workflow, 628
BH_GET_ACTORS, 1035
rule, 551
SAP_WAPI_READ_CONTAINER, 603, 711
SAP_WAPI_START_WORKFLOW, 711
SAP_WAPI_WORKITEM_COMPLETE, 711
SWE_EVENT_CREATE_UPD_TASK,
617, 620
SWE_EVENT_CREATE_UPD_TASK,
616
SWE_EVENT_MAIL, 276
trigger workflow, 636

G
General task, 157, 166, 183, 285, 353, 736, 908
attribute, 152
Generic Interaction Layer → GenIL
Generic Object Service → GOS
gen component, 910, 912, 932
framework, 878
implementation class, 910, 911
Going live, 93
GOS, 338, 516, 564, 593, 623, 788
Graphical workflow log, 125, 233, 1049
GRC, 955, 958
applications, 956
SAP Access Control, 956
SAP Process Control, 956
GUID, 477, 944
format, 211

H
HCM Processes and Forms, 935
agent determination rules, 946
interactive tasks and subworkflows, 947
program exits, 944
specific workflow techniques, 940
standard workflows, 939
template requirements, 938
UWL configuration, 951
workflow templates, 936
Help desk, 247, 251
web based, 247
Hide task, 100
HR Administrative Services, 935
HR infotype, 578
HR master data, 577
HR structural authorization, 278, 284
HTML control, 789
Implementation guide

Import

Hybrid container apps, 983
Human-centric process, 47
Human role, 258

Include

InfoCube

Instantiate, 501
Instance Linkage option, 229
Infotype, 165, 578, 949
Inheritance, 410, 413

Interface

Interface (Cont.)
Improve, 477
IFDISPLAY, 432
IPOBJECT, 476
Intranet, 247
Invoice, 826, 876
Item type, 954
iView, 717

Java

class, 718
code, 721
Java EE 5, 711
JavaScript Object Notation → JSON
Job, 161, 163
SWWCOND, 384
SWWDIHEX, 213
SWWPERE, 278
Job role, 802
JSON, 777
model object, 754

Key

Key field
add, 433
create, 433
details, 418
object declaration, 448
Key figure, 268
KPI, 196

Latest end, 348, 350, 492
date/time, 869
text, 354
Latest start, 348
text, 354
Latest start date/time, 869
Launch handler, 733
Library
sap.m, 755

Link table, 788
Local object, 345
Local Persistent Object Reference → LPOR
Locking, 330, 331
Log, 99
technical, 301, 341
Logical destination, 102
Logical expression, 585
Logical unit of work, 614
Logically deleted, 390
Looper
until, 370
Lotus Notes, 63, 113
LPOR, 477, 495, 499, 513
persistent, 495
transient, 495

Macro, 448

SWC_CALL_METHOD, 463, 516
SWC_CREATE_OBJECT, 450, 454
SWC_GET_ELEMENT, 459, 551
SWC_GET_PROPERTY, 450, 451, 452, 458, 516
SWC_GET_TABLE, 459, 552
SWC_GET_TABLE_PROPERTY, 450
SWC_SET_ELEMENT, 463
SWC_SET_TABLE, 452
Mail, sending, 394
Managed system, 665
Mass deletion, 243
Master controller, 779
Master data
initiate/process changes, 995
Master data governance, 993
application framework, 994
change request, 995
change request (business object), 1000
define scenarios with workflow, 994
Floor Plan Manager, 994
Measurements, 196
Message
control, 576
type, 644, 649

Method, 318, 353, 410, 412, 483
add to BOR object, 440
after, 1040
asynchronous, 353, 421, 438, 589
background, 446
before, 1040
container, 282, 395, 397, 1040
create, 437
create object reference, 460
custom, 468
default, 227, 440
define as asynchronous, 441
define as synchronous, 441
details, 421
dialog, 438
diagnosis, 422, 442, 446
instance-dependent, 439
parameter, 421, 439, 442
programming, 457
result, 439
return errors, 460
secondary, 1039, 1040
synchronous, 353, 458
testing, 318
virtual, 446
WF_DOC_CREATE_SOFM, 748
without dialog, 353

Metrics, 74
Microsoft Office, 64, 748
Outlook, 63, 113
Midnight magic, 244
Mobile device, 90
Mobile fit, 982
Mobile workflow
approaches, 983
Model layer, 718
Model View Controller → MVC
Modeled deadline, 348, 392, 1047
monitoring, 374
notification, 1048
Modeling Support wizard, 1029
Monitoring, 256
deadline, 347
Queue

WORKFLOW_LOCAL_xxx, 307

Quote, 827, 876

Ready status, 605

Receiver

function, 583

function module, 320, 409, 583, 585, 590, 619, 621, 622

Recipient, 150, 156

Refresh

buffer, 244

organizational environment, 245

Relationship

between BOR objects, 413

workflow data points, 83

Replication, 52

Replace, 115, 605

manually, 224

Report

RSWWIDE, 326

RSWWIDE_DEP, 326

standard, 210

Reporting, 64, 94, 210, 263, 612

agent workload, 214

deadline target, 214

diagnosis of workflow with error, 212, 220

orphanned work item, 544

processing duration, 213

queued workload, 215

work item by processing duration, 213

work item by task, 223

work item selection, 222

work items by task, 213

work items with monitored deadline, 213, 214

diagnosis of workflow with error, 212, 220

orphanned work item, 544

processing duration, 213

queued workload, 215

work item by processing duration, 213

work item by task, 223

work item selection, 222

work items by task, 213

work items with monitored deadline, 213, 214

workflowing, 108

workflow for object type, 212

workflow performance, 212

workload analysis, 214

workload analysis for the past, 215

Requested end, 348

text, 354

Requested end date/time, 869

Requested start, 348, 605

deadline, 280, 480

Requirements gathering, 69

agent determination, 84, 86

data, 78

future enhancement, 94

suitability, 72

Reserve, 115, 606

Responsibility, 538, 546, 804

determination, 810

overlapping, 832

split/shared, dynamic process, 859

Responsibility rule, 88, 242, 537, 544,

545, 549

priority, 547

Responsible agent, 818, 821, 828, 864

Restart

after error, 223

workflow, 613

Resubmission, 120, 124

Result parameter, 439, 443

Return on investment, 54, 74, 407

Reusability, 410

Review workflow, 406, 856, 1050

Reviewer, 406, 802, 855

RFC, 305, 307, 613

configure destination, 101

error, 314

model, 718, 721

queue, 273, 282, 305, 613

RFC destination, 712, 718

setup, 712

RFC workflow destination

configuring, 101

Workflow_Local, 102, 307

RFx, 828

RH_GET_ACTORS, 542, 1035

RH_SAP_ORG_OBJECT_RELATE, 550

Risk Management, 974

Role, 105

SAP_BC_BMT_WFM_ADMIN, 105

SAP_BC_BMT_WFM_CONTROLLER, 105

SAP_BC_BMT_WFM_DEVELOPER, 105

SAP_BC_BMT_WFM_PROCESS, 105

SAP_BC_BMT_WFM_SRV_USER, 103

SAP Business Workflow Engine, 599

SAP Business Workplace, 100, 112, 114, 115,

116, 284, 350, 706, 738, 840

SAP BW, 64, 263, 265

SAP Content Server, 789

SAP CRM, 877

action, 888, 916

action condition, 928

Alert inbox, 891

assignment block, 883, 915, 931

BOL, 897

BOL entity, 884, 915, 920

BOL framework, 884

BOL model, 910

BOL object, 885

BOL root object, 898

BTOrder, 912

business document, 878

business object type, 882

business transaction, 881

campaign automation, 902

change history, 883

claims and funds management, 877

component controller, 913

Component Workbench, 885

CRM inbox, 891, 923, 930

CRM_ORDER_MAINTAIN, 880

CRM_ORDER_READ, 880, 889

CRM_ORDER_SAVE, 880

customer interaction center, 877

dynamic mapping, 894

dynamic navigation, 894, 910, 920

enhancement set, 886

event callback, 880

event handler table, 880

follow-up document, 884

GenLL, 884

GenL component, 884

inbound plug, 915

lead, 903

mapping class, 895

marketing, 877
Workflow (Cont.)
container, 133, 140, 281, 338, 341, 349, 365, 395, 396, 582, 583, 684, 685, 739, 950
continue after system crash, 246
create, 337
create for SAP ERP HCM processes, 946
create/execute, 343
customizing, 887
data needed, 79
definition, 67, 303
design, 90, 218
developer, 114, 252
development, 1029
documentation, 347
event, 292
event-based, 974
final standard, delivered, 58
fine-tune, 968
generic enhancers, 1030
header event, 590
ID, 304, 600
Index, 898
initiator, 81, 85, 344, 345, 559, 736
instance, 67, 212, 231, 310, 325, 600
interface with governance framework, 1000
item, 602
log, 301
macro, 448
master data governance, 994
mobile, 145
modify routing logic, 957
object, 233
outbox, 126, 342
patterns, 94
planer-based, 974
process-controlled, 807, 869
report, 94, 263
rerousing escalation, 961
restart, 613
restart after error, 246
route as subworkflow, 391
review, 406, 856, 1050
role, 105
rule-based, 1006
Workflow (Cont.)
runtime system, 102, 280
start condition, 586, 887
start event in SAP Master Data Governance, 1001
starting, 563, 594, 596
subworkflow, 405, 600, 603
templates, 222, 1029
templates and tasks, 966
terminology, 66
test plan, 325
toolbox, 119, 624
trace, 309, 310, 312, 861
transmit email, 1037
trigger externally, 627
trigger with Enterprise Service interface, 627
triggered by incoming document, 791
troubleshooting, duplicate/multiple, 282
upgrade, 323
version number, 95, 600
visualization for business user, 188
WF_VERIFY, 180
wizard, 341, 394, 405
WS10000011, 904
Workflow API, 682, 686, 741, 980
Workflow Application Programming Interface
→ WAPI
Workflow Builder, 48, 59, 90, 179, 275, 282, 329, 337, 355, 542, 902
container definition, 341
frame, 338
look and feel, 338
start, 342
test workflow, 345
Workflow container, 1047
element, 1037
Workflow customizing, 97
automatic, 101, 102
Workflow definition, 147, 274, 342, 357
activating, 345
basic data, 338
binding, 366
branch, 369
conditional branch, 369
container, 364
Workflow definition (Cont.)
deadline, 347
documenting, 376
fork, 390
form, 369
saving, 345
syntax check, 345
testing, 345
transport, 108
Workflow email
configure settings, 867
Workflow Engine
customizing, 108
Workflow Instance Error
support tools, 293
Workflow instance Error, 291
fix, 292
Workflow log, 115, 120, 124, 210, 224, 231, 276, 294, 410, 593, 595
agent view, 233
approval display, 862
chronicle view, 232
filtering, 375
formats, 231
graphical view, 233
object view, 233
technical view, 234
view list of agents, 284
Workflow step, 338
activity, 369
ad hoc anchor, 369
advance with dialog, 376
branch, 369
case, 370
condition, 370
conditional branch, 369
container operation, 366, 369
event creation, 369
fork, 369
loop, 370
manual confirmation, 375
outcome, 372
process control, 370
result, 372
Workflow step (Cont.)
send email, 370
simple form, 369
type, 368
user decision, 344, 370
waiting for event, 370
web activity, 371
WFXML, 371
Workflow template, 107, 222, 351, 352, 354, 940, 952
change request process, 1020
HCM Processes and Forms, 936
process variations, 1006
rule-based, 1007
use for change request, 1001
WS10000034, 864
WS10000606, 847, 863
WS1000129, 847
WS14500015, 859
WS1790260, 949
Workflow_Local, 307
Worklist
access from mobile device, 63
for mobile device, 980
Workflow analysis, 171, 214
World Wide Web Consortium → W3C
WSDL, 630
WSHANDLER, 738, 739, 741
WS→Workflow template, 107
X
XML, 140, 312, 603, 604, 710, 721, 943, 944, 952
code to work item ID, 723
customization with UWL, 691
declaration, 763
downloading process data, 840
execute WebFlow Service, 739
fragment, 770
load customized, 693
message type, 644
persistance, 329
view, 756
XML Schema Definition Language, 644
Jocelyn Dart is a long-time SAP employee who currently majors in solutions in the SAP Intelligent Business Operations Powered by HANA bundle, and minors in user experience technologies and design thinking. Since joining SAP Australia in 1994, she has worked directly with more than 70 organizations in Australia, New Zealand, and internationally, across a diverse range of industries. She is currently a Platinum Consultant, Design Thinking coach, and was recently appointed as an SAP Mentor by the SAP Community Network, where she blogs on a number of topics, workflow included.

Sue Keohan is a senior application developer at MIT Lincoln Laboratory. She has worked with SAP since 1995, when MIT began its implementation. She has been designing and implementing processes with SAP Business Workflow since 1997, starting with SAP R/3 release 3.1C. She was instrumental in founding the ASUG Workflow and BPM Special Interest Group as well as the SAP-WUG mailing list. In 2008, Susan achieved one of her highest professional honors—to be named an SAP Mentor.

Alan Rickayzen is senior product manager in HANA SAP BPM development. He has been with SAP since 1992 and in data processing since 1988. In 1995, he joined the SAP Business Workflow group, performing development work as well as consulting for various blue-chip U.S. customers. In his pursuit of interoperability, he became one of the principle authors of the Web standards BPEL4People and WS-HumanTask, and he was instrumental in the integration between SAP Business Workflow and IBM Notes.