K2 LEARNING™

Building Workflows with K2 Studio - Fundamentals

This learning module introduces the K2 Studio design environment and describes the fundamental concepts used when designing, assembling, deploying executing and monitoring workflow-centric K2 applications with K2 Studio.

Contents

100.IAH
Version: 2
Revision: 2

Questions, Comments or Feedback about this training module?

Please E-mail **learning@k2.com** with your comments or feedback. We appreciate any feedback that helps us to improve the quality of our learning material.

Conventions

The following table documents the conventions used throughout this module:

Convention	Description
Bold Text With Border	Represents commands, controls, items, menus, options, parameters, and file and folder paths. For example, in an instruction to click on the **OK** button; look for a control on the page with OK as its name. When you see text represented like this, you should see a control, window or field on the user interface with the same name.
`Fixed-pitch font`	Represents text (or code) that must be entered exactly as shown. When you see text represented in this style, you should be typing this text into a textbox, control, code window or other UI component.
[Italic text in Square Brackets]	Represents variables or values to be selected from the K2 Context browser
<ALL CAPITALS>	Represents a key to be pressed on the keyboard. For example, **<CTRL>** represents the Control key on the keyboard.
	Represents a best practice. Best practices are the recommended approach or guidance for specific scenarios. When you see the Best Practice icon, consider how this practice impacts your requirements or how you can implement this practice in your environment.
	Represents important information. Important information is highlighted in order to draw attention to a key piece of information, and typically serves as a warning or point you should bear in mind when using the K2 platform in your K2 solutions.
	Represents a note. Notes explain a topic or provide additional information, and can provide a summary or explanatory point about a topic.
	Explains a concept, or a provides an explanation for a step in an exercise
	Represents a hint, tip or available tools and resources that can help you during the design and development cycles of a K2 solution.
	Represents trivia. The text included next to this icon is for interest only – think of it as a quick break from learning about K2.
	Indicates an estimate time to complete an exercise, step or discussion.

Module Overview

This learning module introduces the K2 Studio design environment and describes the fundamental concepts used when designing, assembling and deploying applications with K2 Studio.

During this module, you will learn how to approach the design of workflow-centric applications, how to assemble workflows with the K2 Studio design tool and how to deploy the solution components to a K2 server environment. You will also learn about various options for User Interfaces and Data Storage, how users interact with the application and how to report on K2 workflows using a K2 Process Portals.

Target Audience

This module is intended for users who will create K2 process-centric applications with K2 Studio. Typical roles that use this design environment include SharePoint administrators, site owners and SharePoint designers. This module is also appropriate for workflow designers (such as Business Analysts and Analyst Programmers) and more technical roles like .NET developers and SharePoint developers.

This module is a prerequisite for any developer who will use the K2 for Visual Studio design tool, since the concepts are applicable to both design tools.

Outcomes

At the end of this module, participants should be able to design, build and deploy basic K2 solutions using K2 Studio. Participants should also be able to use the available K2 reporting tools to monitor workflows and understand how end users interact with a K2 process.

Complexity

This Course	Level	Definition
	100	Introduction to the topic or overview and assumes little or no expertise with the topic being covered. Typically level 100 modules cover concepts, functions, features and benefits.
	200	Covers intermediate learning materials and assumes 100-level knowledge and provides specific details about the topic and a fairly complete understanding of the features. 200-level training may discuss case studies that cover a breadth of common scenarios or explain how to use more advanced features.
	300	Covers advanced learning materials and assumes 200-level knowledge, and an in-depth understanding of product features in a real-world environment. 300-level training provides a detailed technical subset of product technologies that illustrate specific aspects of the product that are key to improving performance or interoperability and include architecture, performance, migration, development and deployment.
	400	Expert learning materials and assumes a deep level of technical knowledge and experience, as well as a detailed, thorough understanding of the topic. 400-level courses are essentially expert-to-expert sessions and the 400-level training content provides the means for customers to push products to maximum performance, achieve the broadest possible interoperability and create applications using the most advanced features.

Prerequisites and Required Knowledge

Before attempting this module, you should have covered the K2 Learning modules listed below (or have sufficient experience with the K2 platform to be familiar with the topics covered in the listed modules).

Reference	Module Name	Important topics (if applicable)
100.SEA	K2 Fundamentals	Basic understanding of the K2 platform and the typical use of the platform.

This learning module also refers to (or uses) other technologies. Because the purpose of this module is not to instruct you on these other technologies, you should be familiar with the technologies, systems or approaches listed below (to the required level of proficiency) in order to successfully complete this learning module.

Skill or Technology	Proficiency

Skill or Technology	Proficiency
Microsoft SharePoint (2007 or 2010)	Basic • Understand SharePoint terms like sites, lists and libraries • Know how to interact with list items and documents in SharePoint • Understand basic SharePoint Security concepts

Note: although this module uses SharePoint for the practical exercises, the basic concepts apply equally well to other user interface technologies. We use SharePoint in this module since it is the quickest and simplest way to create K2 applications, allowing you to focus on the core concepts of workflows with K2 Studio rather than diving into the technicalities of various user interface technologies. Other learning modules will describe the various UI technologies that are available in K2 solutions.

Compatibility

This learning module is compatible with the following K2 products and versions

These are the minimum compatibility requirements for this learning module. Earlier and later updates to the listed products may include, exclude or change the available components, screen layouts or product configuration.

Product	Version	Notes
K2 blackpearl	4.6.x or later	

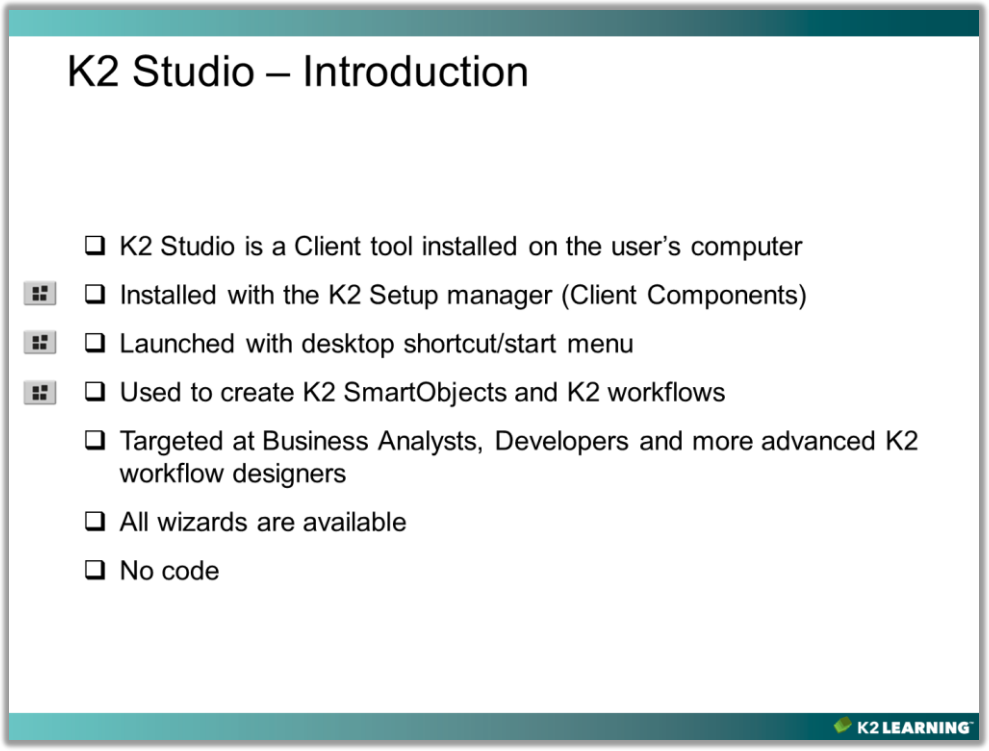
Virtual Environment Prerequisites

This learning module has prerequisites which need to be configured to support demonstrations and lab exercises. Follow the instructions below to set up the prerequisites for this learning module.

This script should be executed in the Virtual Server environment which accompanies this learning module.

Step	Instructions
Step 1: Install the prerequisites for this training module in the Virtual Server environment  *A script is provided that will create all the prerequisites for this module.*	**Tasks:** 1. Log on to the virtual environment with the following credentials: Username: `Denallix\Administrator` Password: `K2pass!` 2. Copy the file **K2 Learning-100.IAH.exe** provided with this learning module to any location in the Virtual Server environment. You may download the .exe file from the **K2 Learning Module Installers (Download)** shortcut on the VPC desktop, or directly from the following URL: `ftp://K2LearningInstallers:$k2learning!@ftp.k2.com/K2 Learning-100.IAH.exe` *Ask your instructor for help, if needed.* 3. Double-click the file **K2 Learning-100.IAH.exe** to start the extract process. You should extract the files to `C:\K2 Learning` (this value should be set correctly by default) 4. After the extract process has completed, double-click the file **C:\K2 Learning\100.IAH\Setup\SetupPreRequisites.bat** to install and deploy the prerequisites for this training module. *The installation process may take up to 5 minutes to complete* 5. Once you see the success message (**Completed setting up prerequisites for learning module…**) close the command window.

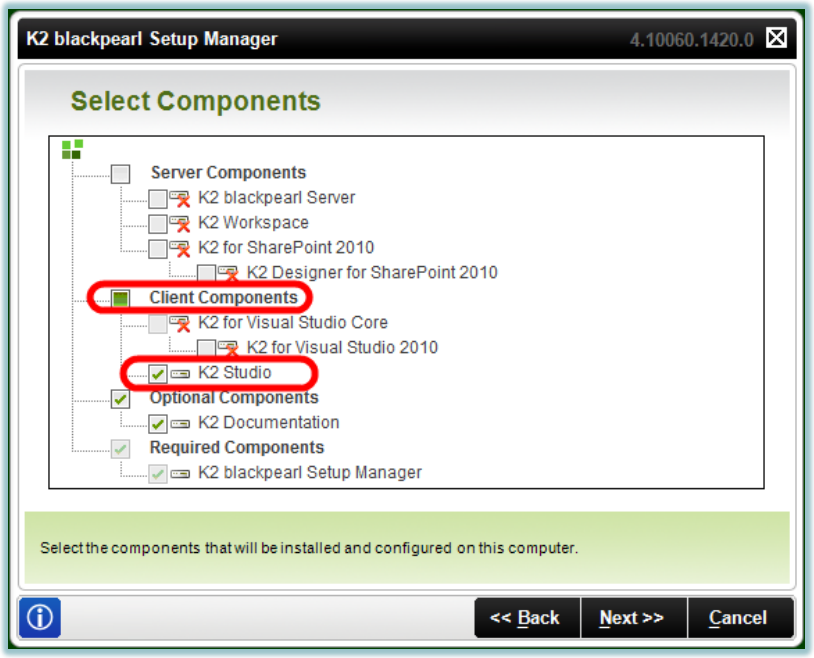
K2 Studio – Introduction



K2 Studio is a design environment used by process designers and developers to assemble and deploy K2 workflows and K2 SmartObjects. It is most often used to create more advanced workflows than are possible in the K2 Designer for SharePoint, and typical roles that use this design tool include Business Analysts and Analyst Programmers, Developers, SharePoint administrators and more advanced technical users.

K2 Studio is client tool that is installed on designers' computers by selecting the **K2 Studio** option under the **Client Components** section of the standard K2 installer, as shown below:

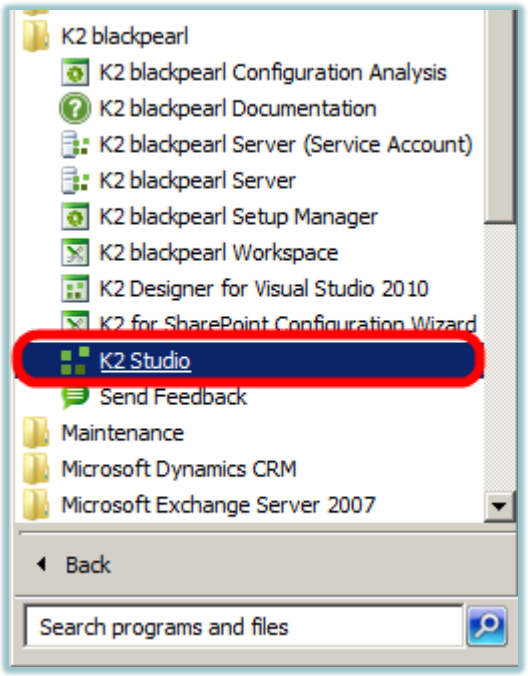
Installing K2 Studio



Once K2 studio is installed, it is launched through the start menu or through the shortcut that the installer creates on the user's desktop.

Launching K2 Studio

Start Menu



Shortcut

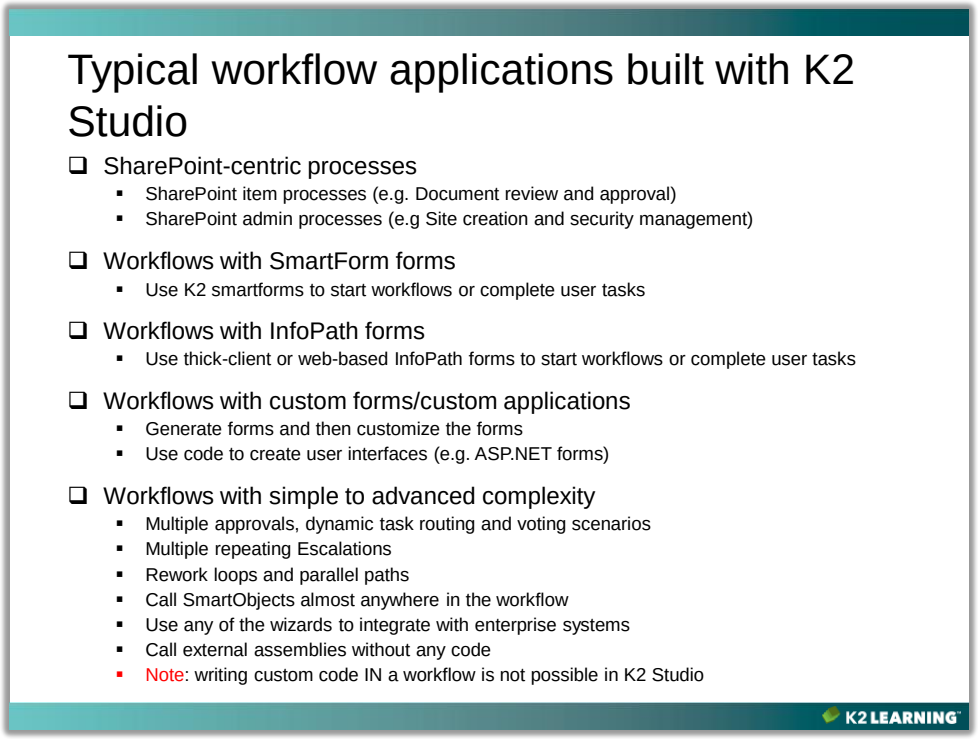
K2 Studio allows the designer to use any of the available K2 wizards and templates to create most of the workflow scenarios that are possible to build in K2. The one scenario that is not possible in K2 Studio is to modify or edit the code inside the workflow. You will need to use K2 for Visual Studio to edit the code in a workflow. It is possible to create a process in K2 Studio, open the process in K2 Designer for Visual Studio to edit the code, and then continue editing the process in K2 Studio. This can help achieve collaborative process design scenarios where less technical users build out the bulk of the process and more technical developers will use K2 for Visual Studio to write custom code in the workflow.

You may also create K2 SmartObject with K2 Studio, and we will discuss that in the learning modules that deal specifically with K2 SmartObjects.

Regarding *K2 for Visual Studio*:
In general, everything you can do in K2 Studio is also possible in K2 for Visual Studio. The two design tools are very similar in behavior and use. K2 for Visual Studio just has a more advanced project system, allows developers to write code for any item in a workflow, and has built-in integration with common development tools like source control providers and debugging tools.

This Learning Module is a prerequisite for any developer who will use K2 for Visual Studio, since the concepts are applicable to both design tools.

Typical workflow applications built with K2 Studio



Let's look at some typical workflow-centric applications that are commonly implemented with K2 Studio.

SharePoint-centric processes

These processes are usually associated with items or events in SharePoint. With K2 Studio it is possible to create workflows that interact with SharePoint content (such as list items or documents, with document review and approval as a typical example) and SharePoint administration tasks (such as interacting with sites or managing security. Automatic SharePoint site creation and permission assignment is a typical use case.

When integrating with SharePoint, K2 workflows can be created that will run when an item is created, checked-in, checked-out, deleted or modified in SharePoint, when an action is taken on a SharePoint list or library such as deleting a field on a list, or when an Item is “submitted to workflow” manually from a SharePoint list or library. That said, you can still interact with SharePoint even if the workflow was started through some other mechanism.

Workflows with SmartForms forms

If you are using K2 SmartForms to create user interfaces, you can create workflows in K2 Studio that use K2 SmartForms as the user interfaces. These forms can start workflow or be used to complete user tasks in the workflow.

Workflows with InfoPath forms

K2 processes can be associated with InfoPath forms to create rich user interfaces without requiring any code. Typically, an InfoPath form with different views is used for each of the step in the process where a human performs some work. K2 supports both thick-client InfoPath forms and web-based InfoPath forms through InfoPath Forms Services. InfoPath-based processes are described in more detail in the learning module **200.AKL - K2 and Microsoft InfoPath**

Workflows with custom forms/custom applications

You may also create workflows that use some custom form or custom application to interact with the workflow. K2 offers a Form Generation approach where the workflow will generate an ASP.NET-based user interface for user tasks. Developers can then customize and extend these generated forms. Developers may also create completely custom applications with a technology of their choice, by calling the appropriate K2 APIs. This topic is discussed in more detail in the learning module **300.NRT K2 Workflow APIs and Services - Runtime**

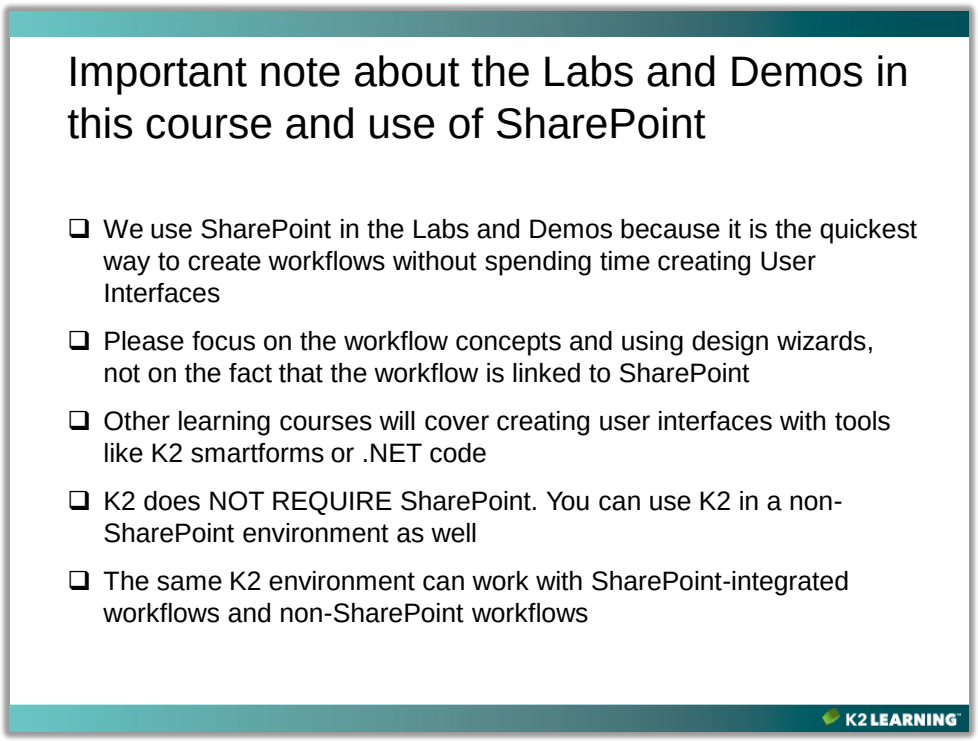
Workflows with simple to advanced complexity

K2 Studio allows users to create workflows that range from simple, sequential workflows to very complex workflows. It is possible to create processes with rework loops, parallel paths, multiple simultaneous approvals, evaluation of voting scenarios, multiple escalations on different levels of the process, customizable E-mail notifications, sub-processes and many more. This learning module will cover the basics of these workflows, while other modules (like **200.AUS Building Workflows with K2 Studio – Intermediate**) will describe how to create the more advanced workflow implementations with K2 Studio.

You can use K2 Studio to build workflows with extensive integration into enterprise systems like SharePoint and SQL and applications like Word, add references and call external services and assemblies and much more. In short, when using K2 Studio designers can achieve almost all possible workflow scenarios possible in K2, short of writing custom code.

K2 Studio can also be used to extend workflows built in K2 Designer for SharePoint. Suppose a workflow designed by an end-user in K2 Designer for SharePoint has been in production for a while. As the business evolves, this process may have more advanced rules. Instead of creating the workflow from scratch as a new process, a workflow designer can download the K2 Designer for SharePoint workflow as a .kprx file, and then open and edit the original workflow design in K2 Studio, adding as much complexity as required. Note that this is a one-way process: once a workflow is edited in K2 Studio, it cannot be opened and edited in the K2 Designer for SharePoint tool.

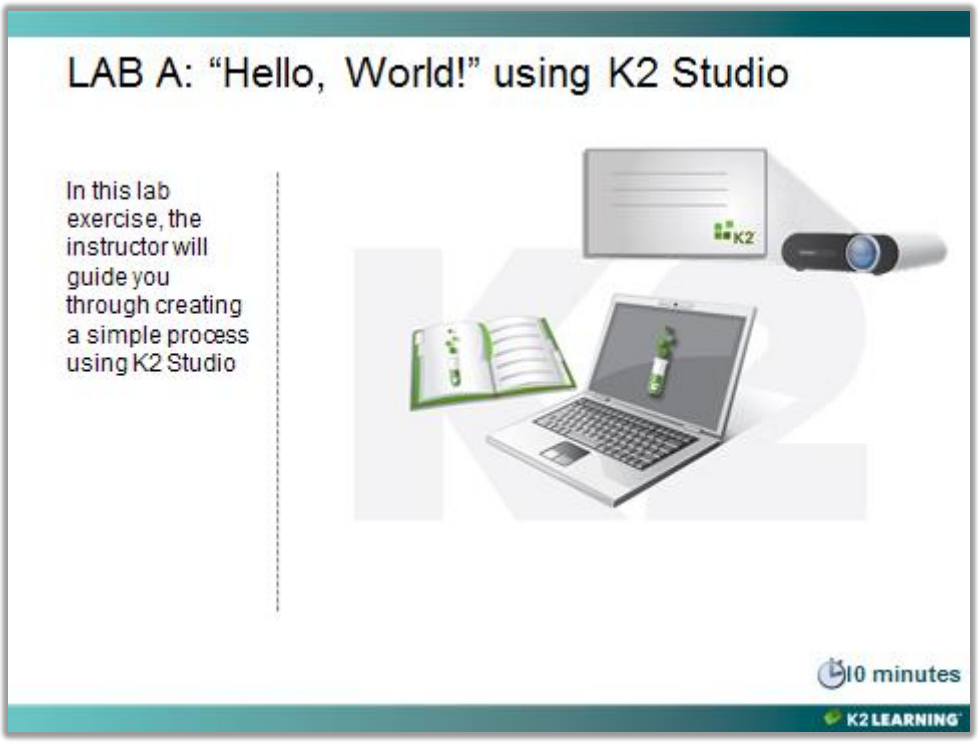
Important note about the Labs and Demos in this course and use of SharePoint



Please review the notes in the slide above. Although we are using SharePoint for the lab exercises in this module, it is important to understand that K2 does not require SharePoint to work and you can integrate with K2 workflows in many different ways. We are only using SharePoint to save time and to keep focus on the workflow concepts, rather than introducing User Interface complexities.

Later learning modules will focus on other User Interface technologies like K2 smartforms and ASP.NET pages. For now, the most important part is just understanding the workflow concepts and becoming familiar with building K2 workflows using wizards.

LAB A: “Hello, World!” using K2 Studio



Objective

In this exercise, the instructor will show you how create a very simple K2 process that just sends an email when the process is started. The objective is just to show you how to launch K2 Studio, how to create and deploy a basic process and one of the possible ways a process could be started. In the later exercises of this learning module, you will create more complex workflows with multiple steps and user tasks.

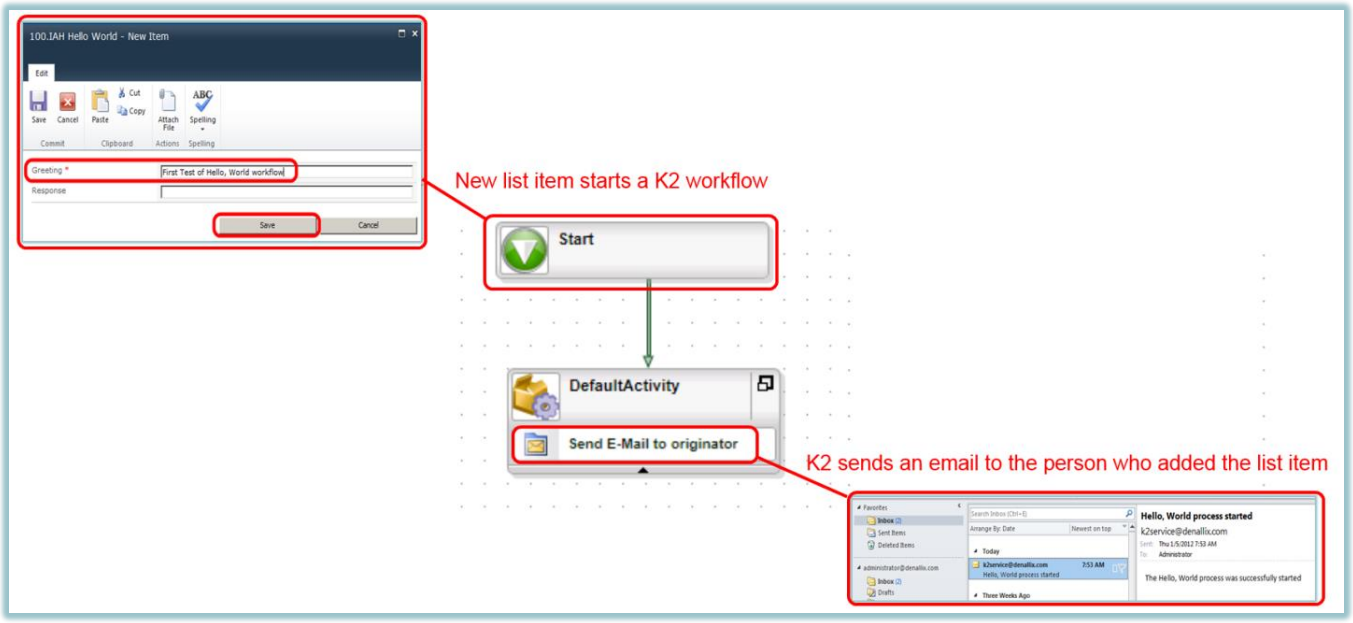
Duration

This lab should take around 10 minutes to complete. Although a guide to complete this guided lab is provided, you should preferably follow along with the instructor to complete this exercise, rather than reading the guide.

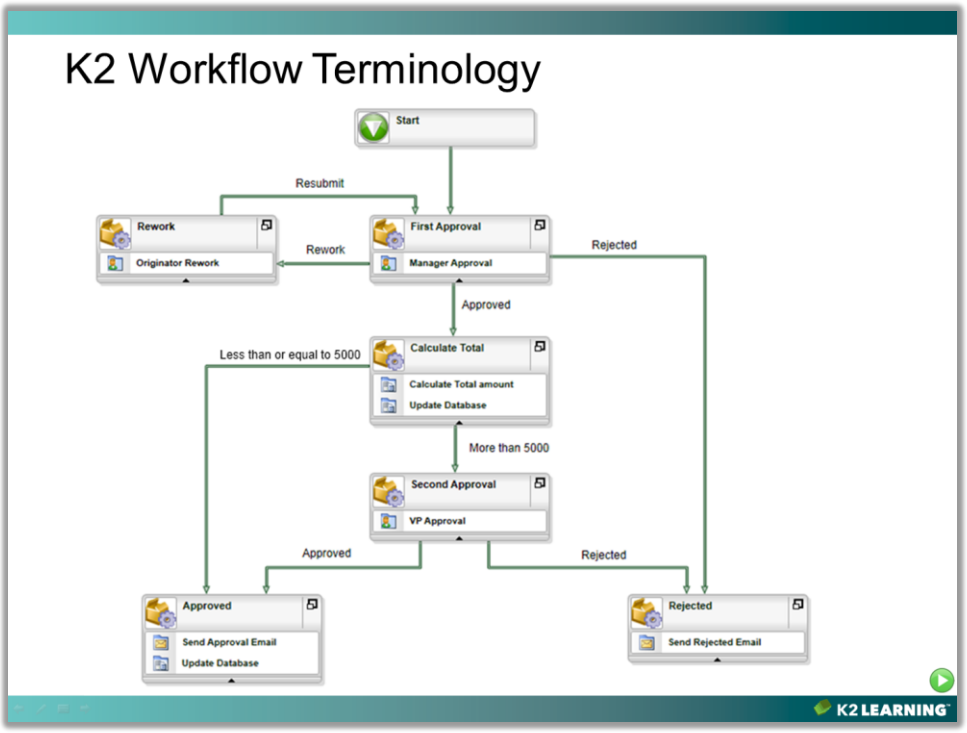
Context

Together with the instructor, you will create a simple single-step process that will send you an email if the process has started successfully. In real-world environments, you could use a simple process like this to validate whether the K2 Studio design tool and K2 server are working in a particular environment.

The process will automatically start when you create a new entry in a SharePoint list, using SharePoint Event integration. There are many other ways to start a process (for example: SharePoint workflow integration, IPC calls from other workflows, custom user interfaces, K2 smartforms and so on) but for the purposes of this module we will use a SharePoint event-based process.



K2 Workflow Terminology



Before beginning to describe the process development cycle, it is worthwhile to understand the terminology that is used when describing K2 processes.

For ease of reading, each of the K2 terms is shown on a separate page.

If you have already completed the *K2 Designer for SharePoint* learning modules, these terms may sound familiar to you. Note that *K2 Studio* uses slightly different terms to describe the same concepts, and that there are some subtle differences between workflows in *K2 Designer For SharePoint* and *K2 Studio* workflows. Most importantly, K2 Designer for SharePoint combines the concept of **Activity** and **Event** into the **Step** concept, while in K2 Studio; an **Activity** is the container for one or more **Events**.
You also have more freedom when designing Lines and Outcomes to define the possible paths in a process, full control over the layout of the process, and access to more advanced properties on the process, activities, lines, escalations and events.

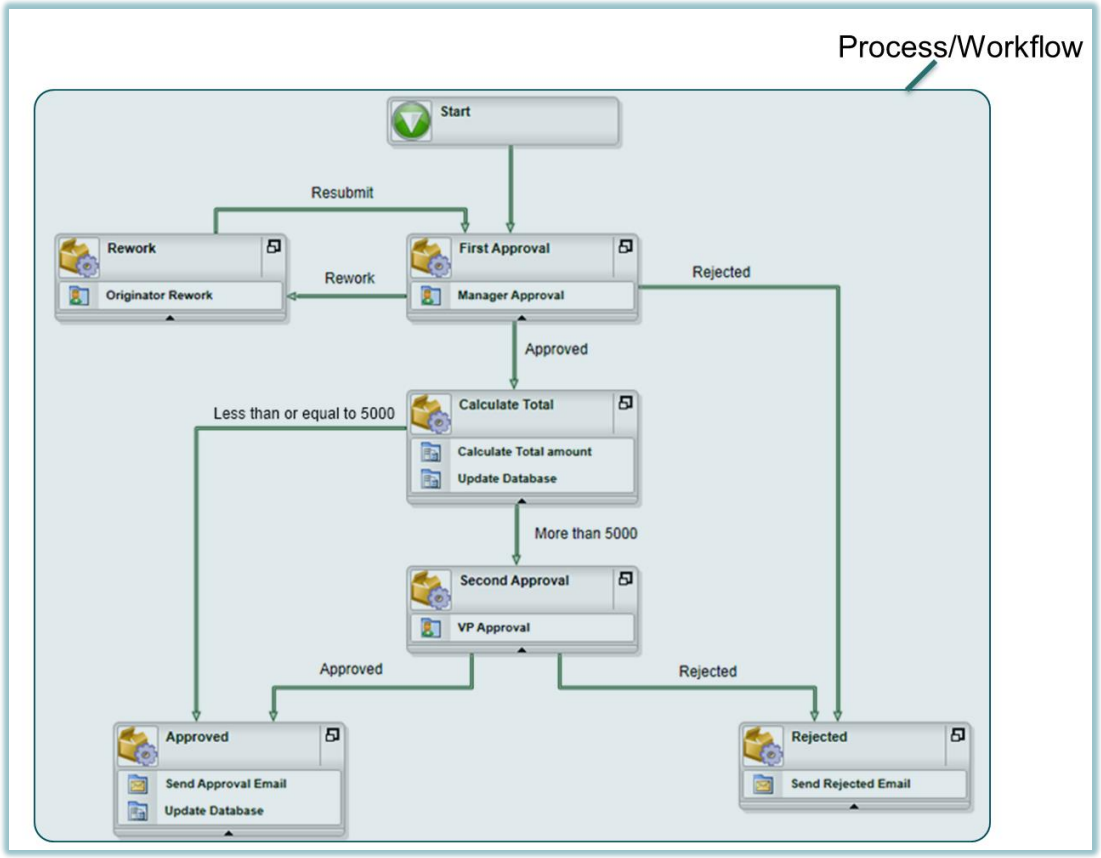
Even if you are familiar with the *K2 Designer for SharePoint* terms, you should take the time to understand the *K2 Studio* terms as well.

Process/Workflow

A **process** (also known as a **workflow**) refers to the entire process designed in the K2 design tool. A workflow will have at least a **Start** activity and one other activity (otherwise it wouldn't be a workflow, right?)

Processes usually contain multiple **Activities**, which are joined with **Lines**. The Activities and Lines generally define the flow of the process, the Events inside the activities define the work performed in the process, and the rules behind Activities define how the activities are executed.

In the diagram below, all the activities, events, lines and the rules behind each of these items comprise the process



Although a process always has a **Start** activity, processes do not need "End" activities to be valid. A K2 process will continue running until there are no more paths left to follow. When this happens, the process status changes to Complete and the process ends.
If you prefer, you can use a placeholder step in a process to model an End activity.

The simplest possible K2 process (like the "Hello, World" process you created in the guided lab) should contain at least a Start activity, another activity with at least one event, and a line that connects the two activities.

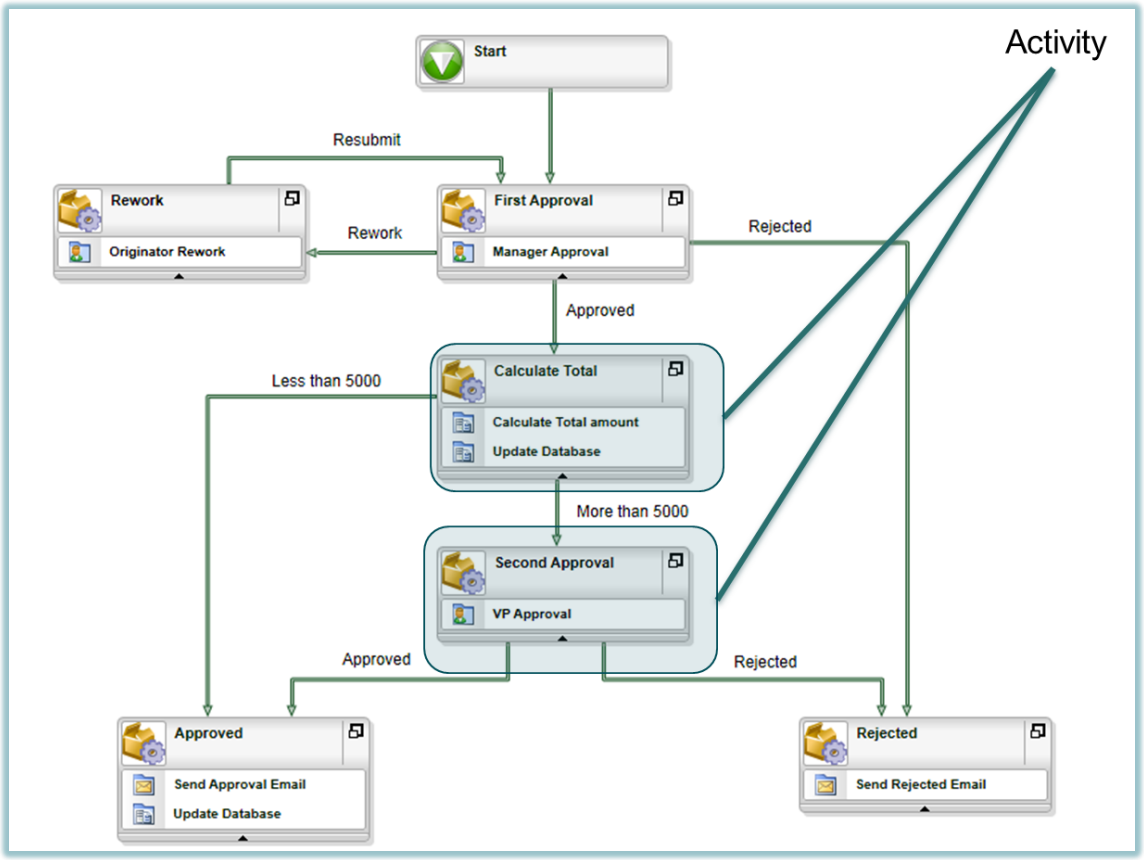
Activity

An **Activity** is a step in a process, and acts as a container for work that will be performed by a user or a system. Whenever work moves from one person to another or to the K2 server, a separate activity will be created for each of these roles. (Strictly speaking, it IS possible to perform both user tasks and server tasks in the same activity, but think of activities as indicators representing that work moves from one human/system to another human/system)

In the diagram below, two activities are highlighted. In this case, the activity **Second Approval** is performed by a user, and the activity **Calculate Total** contains server-side tasks performed by the K2 server.

It is very common to define various Rules for an activity that will control how that activity should execute at runtime. Activities may contain rules that define IF and WHEN that activity should execute, WHO will perform the task if the activity must be completed by a user and WHAT the possible outcomes for the activity could be. A process will always have at least one activity that follows from the Start activity.

Activities must always contain at least one **Event,** since the activity is only a container; the actual work is performed by the **Event**.



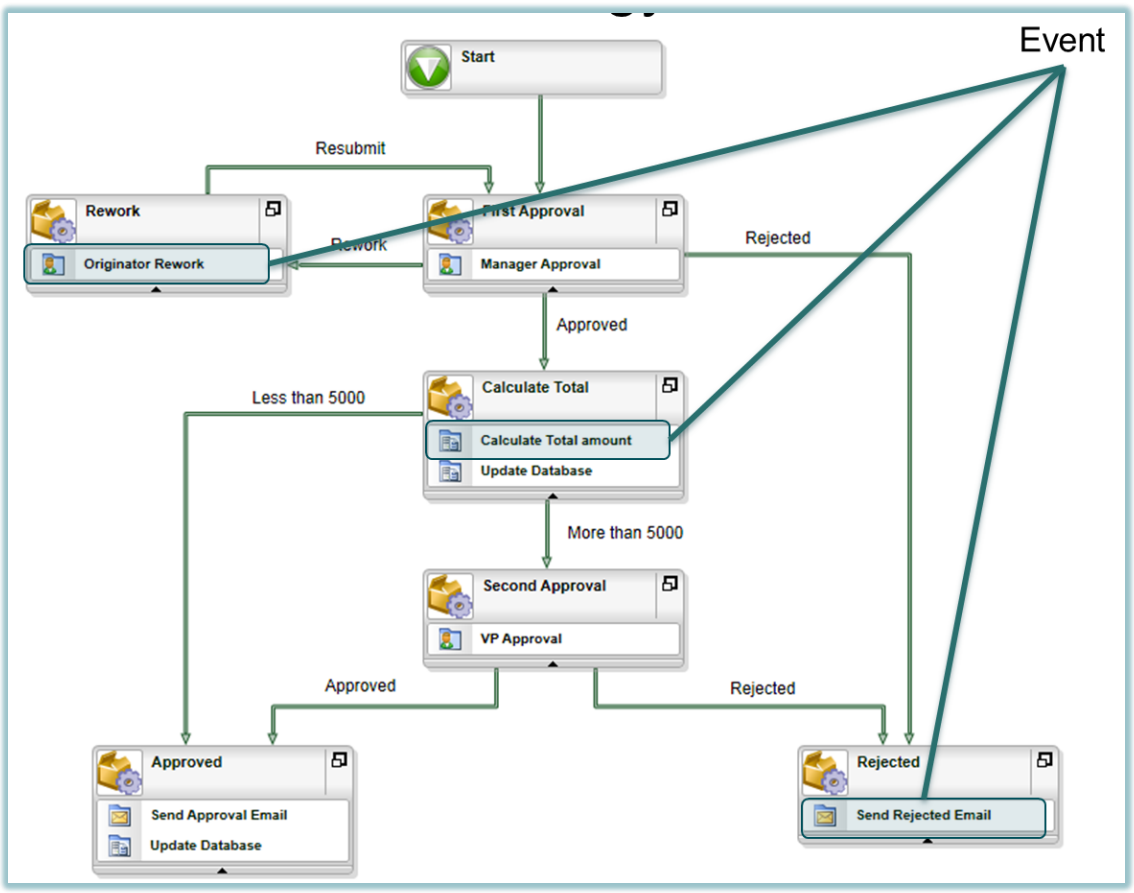
Event

An **Event** is a unit of work in a process, which is performed by a human or system. Work performed by users is known as **Client Events** and work performed by the K2 server is known as **Server Events**.

Events must always be contained inside **Activities**, and the Event will describe the work that is being performed. K2 provides many wizards to make the setup of these events very easy; for example the E-mail Event wizard which allows the designer to configure an E-mail that will be sent as part of the process. This wizard gathers input from the designer and lets them drag and drop variables or type text into the various properties on an E-mail message.

In the diagram below, three Events are highlighted. In this case, **Originator Rework** is performed by a user, and the other Events **Calculate Total amount** and **Send Rejected Email** are performed by the K2 server.

Notice that the **Calculate Total** and the **Approved** activities each contain two events. Activities can contain as many events as needed to complete the step, and for server-side processing it is very common to have multiple events in the same activity.

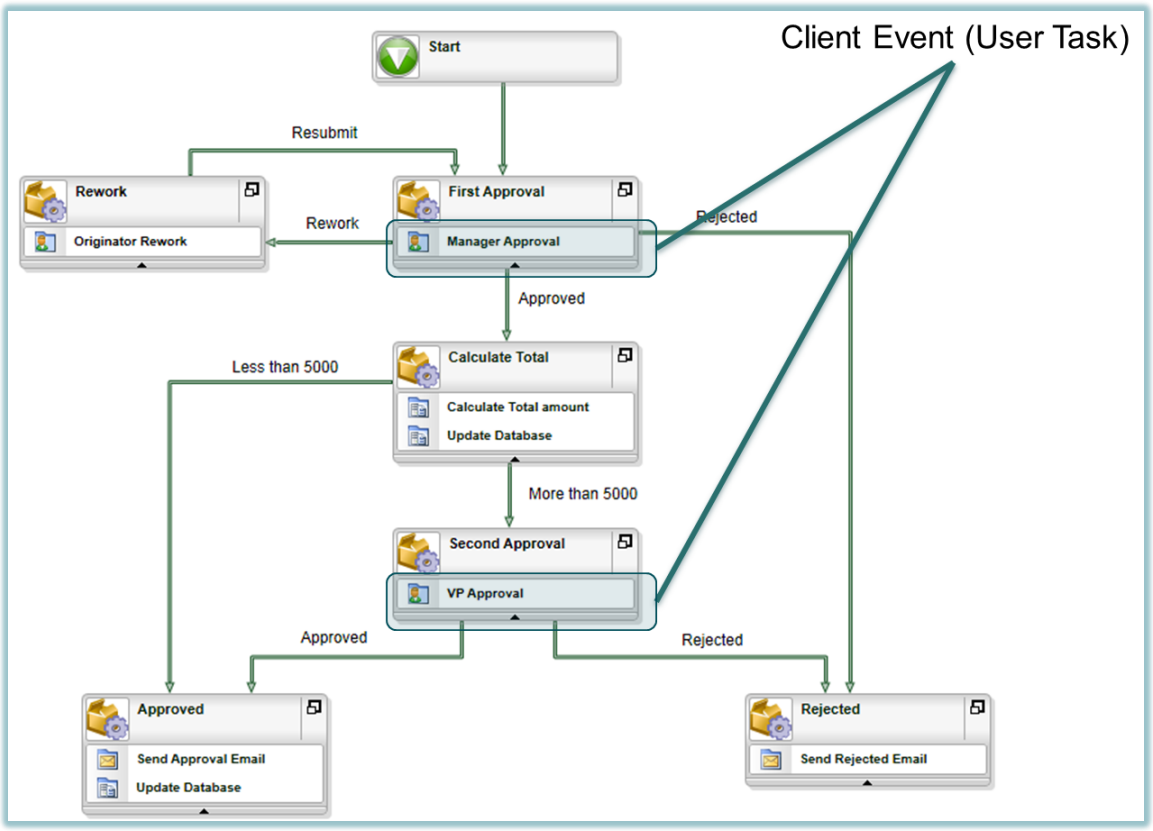


Client Event

A **Client Event** is a task which is performed by a human. All tasks that are performed by humans must be allocated to a **Destination** which is defined on the Activity level. This Destination determines who will ultimately get the K2 task on their task list.

Most often, user tasks require the participant to make some decision: perhaps they need to approve or reject a request, or send it for rework. Users do not always need to select a decision though; in some cases, the participant may only need to indicate that the task has been completed in order for the workflow to continue. The key, however, is that a person had to tell the workflow to continue. Any process that is currently at a Client Event will remain at that step until the user completes the task, or an escalation causes the task to be expired. (A process administrator may also override the process and force the workflow to go to another step. For more information, see the learning modules that deal with K2 Process Administration)

In the diagram below, two user tasks have been highlighted: **Manager Approval** and **VP Approval**. Each of these tasks will be completed by the relevant users with some or other user interface.

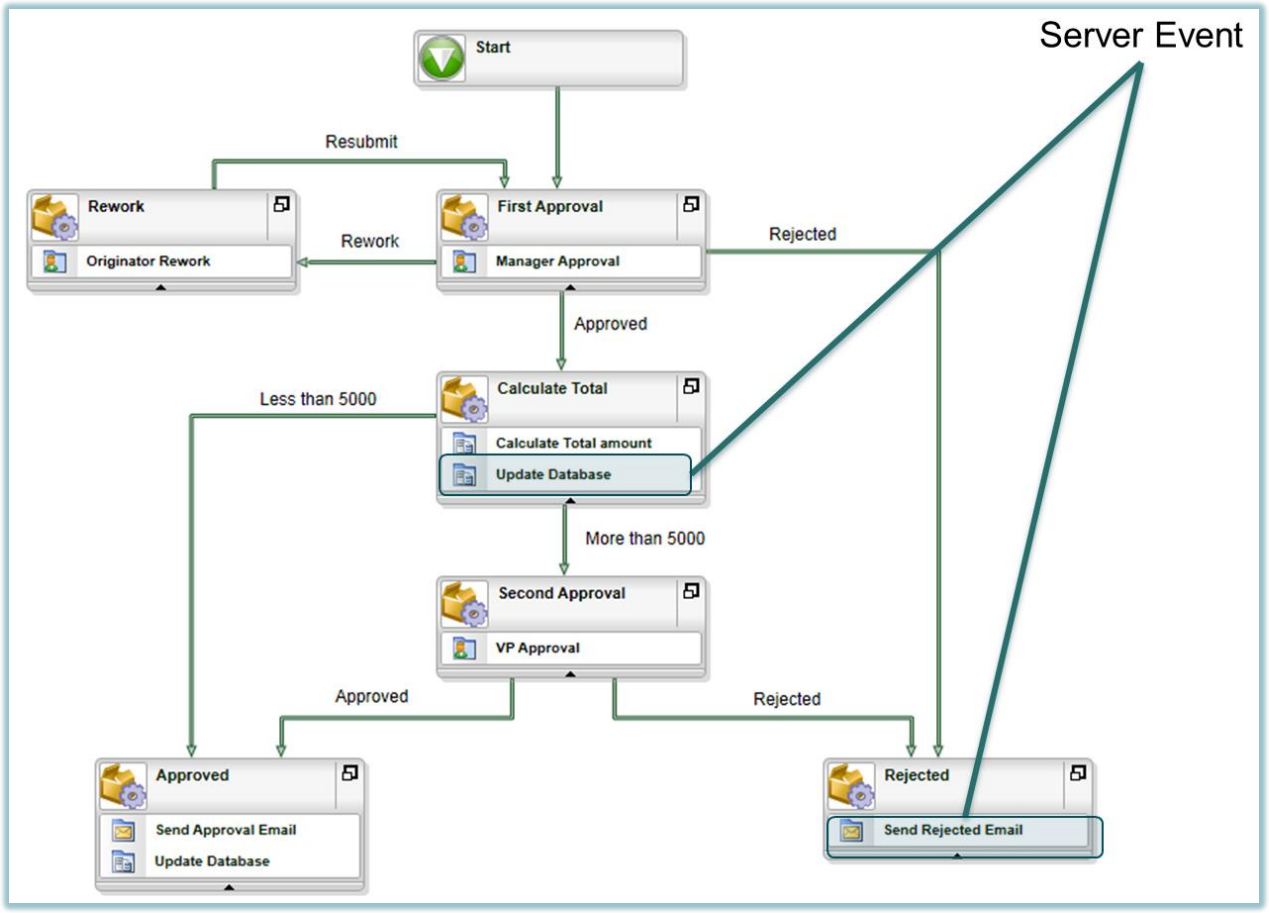


Note that all Client Events must have a Destination set on their containing Activity before the process can be deployed. After all, if a task is assigned to a user, K2 must know who to assign the task to. This Destination could be a hard-coded username, based on a group in AD or SharePoint; a role defined in K2, or could be looked up in some external system. The K2 workflows doesn't really care HOW you determine who should do the work, it just cares that some username or group name is returned that tell K2 who to assign the work to.

Server Event

Server Events are tasks which are performed by the K2 server. These tasks are usually not long-lasting and will complete in a few seconds, at most. Server Events could involve integration with some system, and the exact nature of that interaction depends on the wizard or SmartObject method being executed in the event. Server events are also used to perform some server-side processing such as performing calculations or manipulating data.

In the diagram below, two Server Events have been highlighted: in the **Update Database** event, K2 is using some code to interact with a database, and in the **Send Rejected Email** event K2 is generating and sending an email.



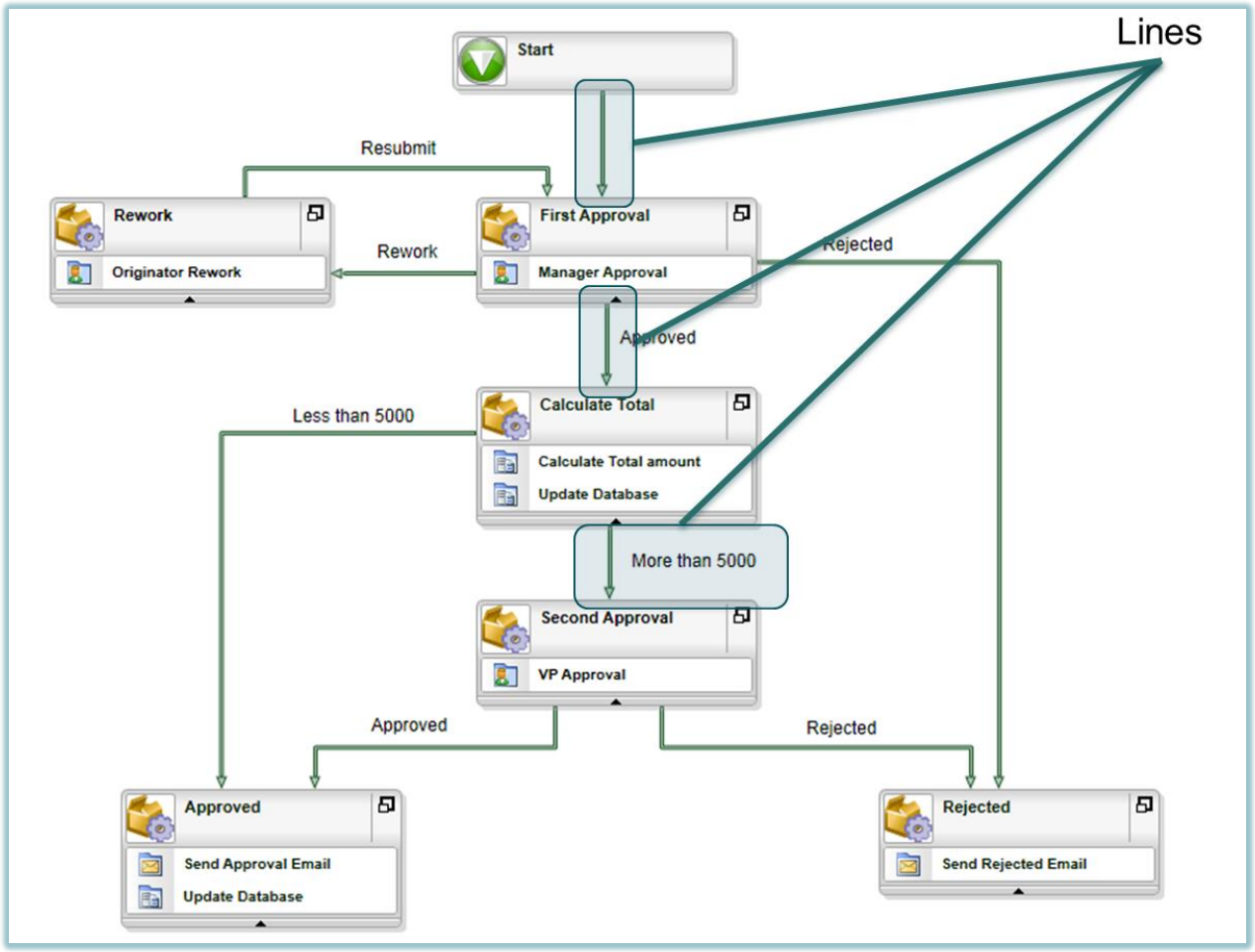
01001011 00110010	Server Events are usually executed in the security context of the K2 service account (in other words, the “Log On As” user account of the K2 Server service), but it is possible to override the security context and tell K2 to use different credentials when executing a specific server event

Lines

Lines are the possible paths in a workflow. Lines are used to join activities in a specific sequence, and effectively tell K2 how to move from one activity to another.

Lines control the "flow" of a process, and by using **Line Rules** or **Outcomes**, workflow designers can tell K2 to follow a certain path in the process based on some condition. Lines can even be used to split and merge a process if parallel work is required.

In the diagram below, three different types of lines have been highlighted. The first line just joins **Start** to the **First Approval** activity, and will always be followed since there are no Rules defined for that line. The **Approved** line is a special type of line called an **Outcome**, and is evaluating the decision made by the approver. The **More than 5000** is a Line with a **Line Rule**, which is evaluating a data value in order to make a decision about where to go next. In this example, if the amount is more than 5000 the workflow must go to Second Approval.

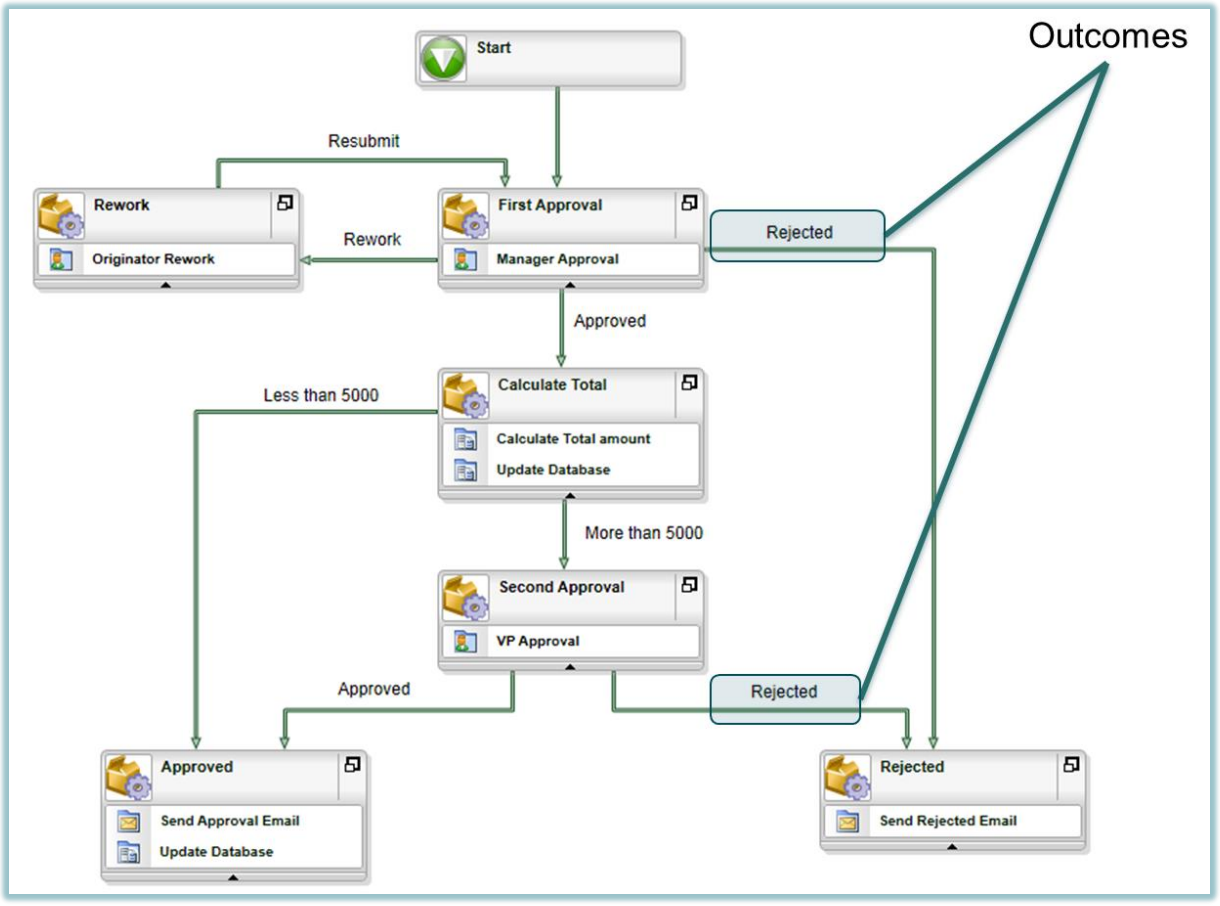


Outcomes

Outcomes are a special type of Succeeding Rule and usually surface on a workflow as Lines. Outcomes are usually based on one or more user's decisions (**Actions**) or by evaluating user actions along with other values to establish a path to follow in the process.

In the diagram below, two lines that follow certain Outcomes have been highlighted. The first **Rejected** line is one of three possible outcomes resulting from the action taken in the **Manager Approval** event, while the second **Rejected** is one of two possible outcomes resulting from the action taken in the **VP Approval** client event.

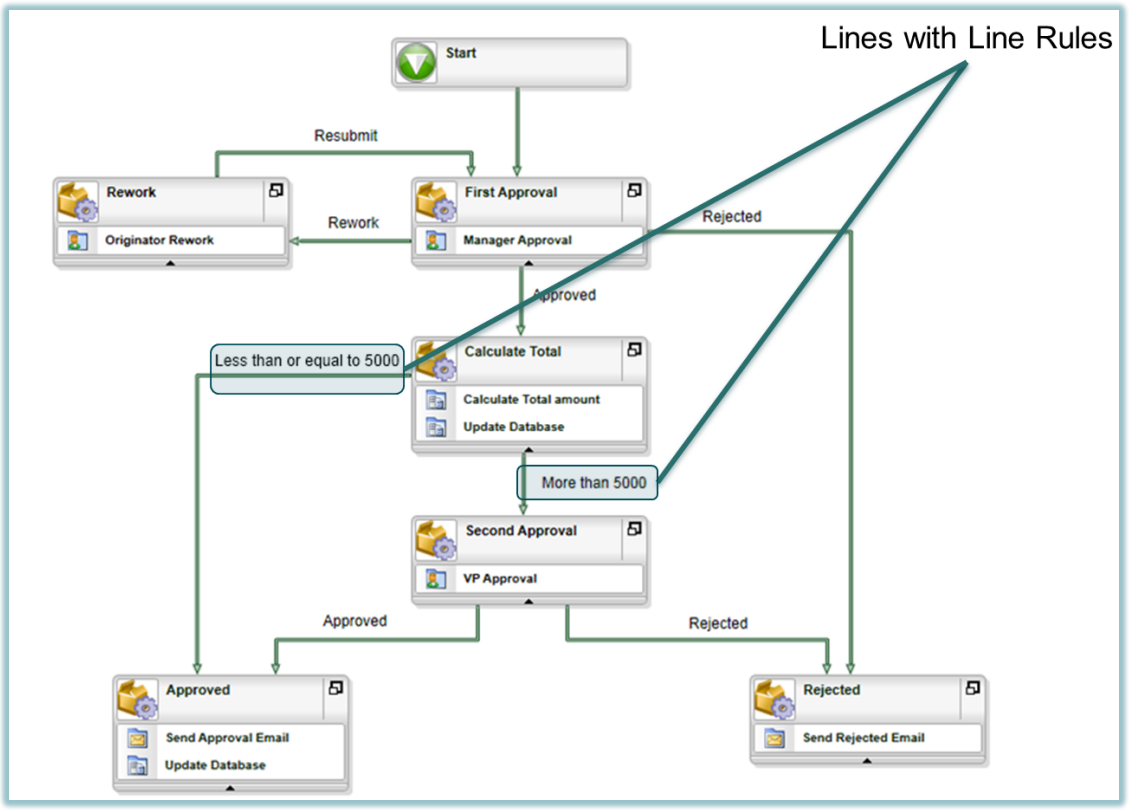
Behind the scenes, Outcomes are defined as Succeeding Rules for workflow Activities, and the resulting Outcome is saved to an activity-level datafield. The Lines that flow from the Activity evaluate this activity datafield. Even though we have indicated "Outcomes" on the diagram below as lines, strictly speaking, Outcomes are actually defined on an Activity level.



Of course, Outcomes and Lines can be a lot more complex than the example above, such as multiple parallel lines, Outcomes that evaluate both user input and data values and so on. For the purposes of this discussion, just remember that, in most cases, Actions yield Outcomes which yield Lines.

Line Rules

Workflow designers can define additional Line Rules behind Lines to control the flow of a process. Most often this is used to control the path of a process based on the result of some comparison operation. In the diagram below, two lines with Line Rules have been highlighted – each line checks the Total value of the request. If the value is less than or equal to 5000, the process goes directly to the **Approved** activity, but if the value is more than 5000, the process goes to the **Second Approval** activity.



When using Line Rules, designers should take care to consider all the possible paths. Consider the lines highlighted in the diagram above. Suppose that the first line performed the following check:
Total Amount < 5000
And the second line performed the following check:
Total Amount > 5000
What would happen if the total were exactly 5000? K2 always follows the path which evaluates to True, but in this case neither path evaluated to true, which means that the process would stop and complete after the Calculate Total activity, because there is nowhere else to go. Once a process has completed it cannot be “resurrected” and the only way to get the process back is to start a new instance of the workflow.

Bear this behavior in mind when designing processes. The recommended approach is to design your lines rules so that at least one will always be true. In the example above, you should change the first line rule to:
Total Amount >= 5000
to prevent the process from completing unexpectedly when the amount is exactly 5000.

Escalations

Escalations are tasks performed by the K2 server in response to some delay time being exceeded. Escalations can be configured for Events, Activities and the Process as a whole, and will only fire if that Event, Activity or Process has not completed within the specified interval.

Events, Activities and the Process could have several escalations that can be repeated as many times as desired. This makes it possible to design escalating levels of escalations on any level.

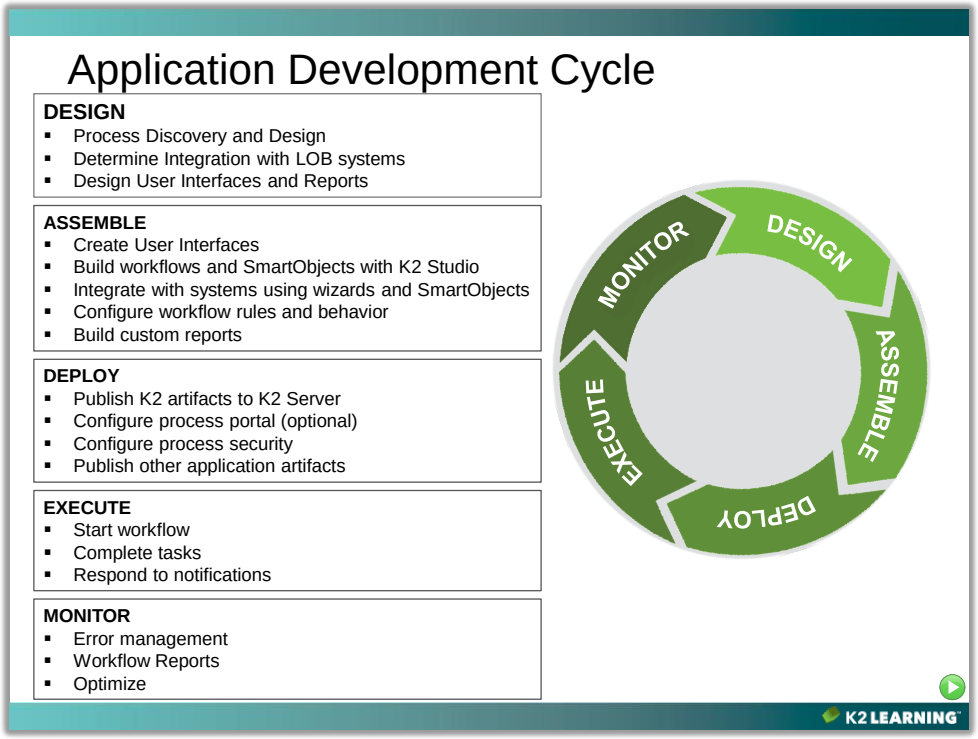
An example of an escalation is to send an E-mail to the originator of the process if the first approver has not completed the task within two days of the process being started. You could add another escalation to the first approval activity to forward the task to another user if the original approver hasn't completed the task after 3 days.

Notifications

Notifications are E-mails that are sent to users during process execution. Notifications are typically used to alert participants that they have new tasks, and are used in escalations as well. Both the styling (for example, applying custom headers and footers or fonts) and content (the message itself and the data included in the message) of the notification can be customized.

K2 update 1420 introduced SmartActions, which enables users to complete their tasks just by replying to the task notification E-mail that was sent to them. The workflow designer can decide whether to allow users to complete tasks this way, or whether they must always open the form to review the task before they can complete the task.

Application Development Cycle



The best way to understand how to use the K2 platform to implement business applications is to break down the life cycle of the application into discrete phases and then consider how K2 tools are used during each of these phases.

Design

During the design phase, the process is discovered, formalized and if necessary, optimized. You will identify the various integration points with LOB systems, and design the user interfaces and reports that are used in the application. It is not necessary to start building the process using the K2 design tools during this phase (although you may, if desired). Instead, the outcome of the design phase should be a specification for the process that can be used during the assembly phase to build the process.

While designing a K2 application, it is important to understand the capabilities of the K2 platform to know what questions to ask and how the process design would translate into K2 functionality. We will discuss some of the questions you need to ask and describe some of the capabilities of the K2 platform in this learning module. The module **200.IDO Gathering specifications and requirements for K2 projects** has a wealth of information and resources that will be useful during this phase of the process development cycle as well.

Assemble

During the assembly phase you would use K2 design tools (or perhaps other design tools) to implement the user interfaces and custom reports for the application. You can use K2 Studio to create the workflows and SmartObjects that may be required in the application, and use the no-code wizards to build integration with various LOB systems.

It may also be necessary to create the prerequisites for the solution during this phase, for example any SharePoint lists or libraries that the process needs to integrate with, or create and configure SmartObjects that expose external systems so that they can be referenced in the process.

The outcome of the assembly phase is a K2 process model in the .kprx file format, which can be saved and edited later and deployed to a K2 server for execution.

Deploy

During the deployment phase, the K2 artifacts like SmartForms, SmartObjects and workflows are published to a K2 Server. This step effectively makes the application available to users. Until the workflow is deployed to a K2 server, users will not be able to run the workflow. K2 administrators would also set up the process permissions during the deployment phase and configure the necessary reporting setup for the workflow, if required.

The K2 server is responsible for managing the versioning, storage, retrieval and execution of the process design.

01001011 00110010	K2 SmartObjects and workflows must be published to a K2 server before they can be executed. There is no capability in the product to "debug" a workflow locally on your machine without first deploying it to a K2 server.

Execute

The execution phase is where users are actually using the various user interfaces to run the workflow, or where K2 is performing server events in the process.

Instances of a workflow could be started by users (for example, creating documents or list items to start the process, or completing an InfoPath form) or based on some system event (for example starting a workflow at certain intervals). Workflows participants would then receive notifications and tasks to complete their work in the process, and the K2 server will be processing any server-side work and any escalations that may happen during the process life cycle.

Typically, no user intervention is required to keep the process running (unless an error condition occurs, which will be discussed later). If you find that a particular process requires a lot of manual override and repair by the workflow administrator, chances are that the process design is not correct and that the workflow should be tweaked and redeployed.

Monitor

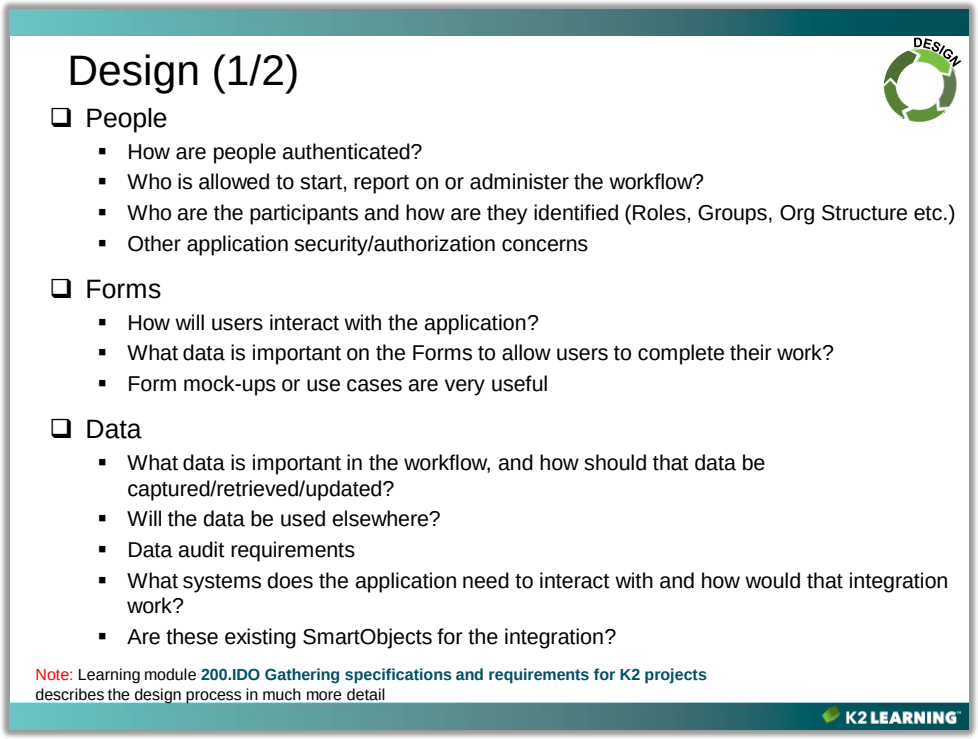
A vital phase of the process development life cycle is to monitor the processes that are executing. Monitoring may include running reports on workflows, using the K2 View Flow report to monitor the execution of a specific instance of the process, or responding to any errors that are reported while workflow instances are active.

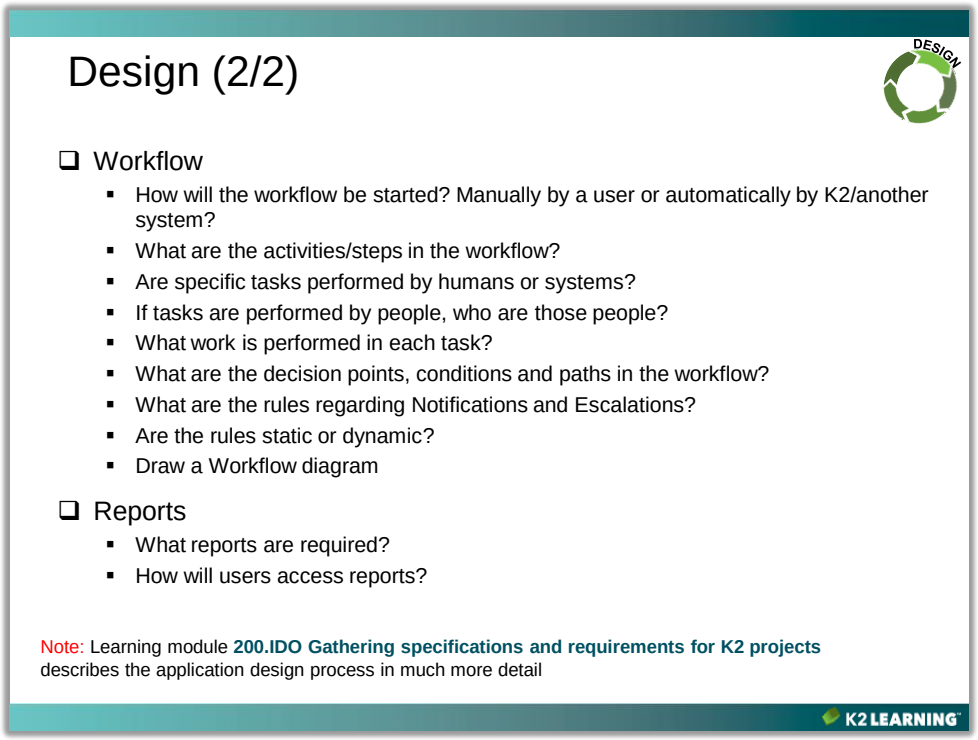
This phase also includes optimization of the application. Depending on the application design, this may involve simple modifications such as adjusting business rules or modifying group memberships to control task routing, or as complex as re-engineering the process or user interfaces.

The iterative nature of application implementation

The diagram implies that the **Design** > **Assemble** > **Deploy** > **Execute** > **Monitor** life cycle is iterative and may be performed several times. Business Processes are typically not static, and may need to evolve over time to keep up with business changes or may need to be optimized based on the reporting information stored by K2. In fact, it is very rare that a business process would be deployed once and remains unchanged for its entire lifetime, since organizations are constantly changing and business processes need to adapt to keep up. There are various ways to approach the process design to cater for changing business requirements, and we'll discuss those as we progress though this and other learning modules.

Design





During the design phase of the K2 application there are some key questions you should ask, so that you are sure to identify the components of the solutions. Let's break the design phase into main areas of a typical K2 application, and describe the kind of questions you need to ask.

The learning module **200.IDO Gathering specifications and requirements for K2 projects** goes into the design phase in much more detail. For now, we will just talk about the high-level points so that you can understand how to approach the design of a K2 application.

People

When discussing the People involved with the application, determine how these users will be authenticated (for example, through their Active Directory user accounts) and what the authorization requirements are (for example, which users can perform what actions in the application. K2 provides a workflow security model which controls the authorization for users to interact with or report on workflows. As part of the design process, these workflow security requirements should also be documented.

You should also determine how users are identified, such as using Roles, Groups or the Organization Structure to assign rights and route work to users. Finally, determine any other application security and authorization issues. In larger enterprises this could include security experts to cover the physical security of the application.

Forms

When describing the Forms for an application, we need to determine how users will interact with the application. Establish what data is important to allow the users to complete their work. While you do not need to identify the technology that will be used for the user interface during the design phase, you should be able to mock-up or otherwise describe the behavior of the user interface.

Data

As part of describing the data requirements, identify the data that is important to the workflow (for example, data points that will be used to control the path of the workflow). Determine whether the data will be used elsewhere (in which case it would be better to store the data in an external database and access it through K2 SmartObjects) and remember to include any data audit requirements. All this information will help the workflow assembler select the most appropriate data storage and forms technology for the requirements.

It is very common for other systems to be involved in a K2 application, both in the Forms and the Workflow. Integration includes system-to-system processing that needs to take place as part of a workflow, for example if the K2 server needs to update a value in a SharePoint list or create a new record in an external database.

During the design phase, it is not always necessary to determine exactly how this integration will be achieved: what is more important is to identify the points at which integration may be required, and whether assembling the workflow is going to be dependent on some other work that must be completed in the other system first. You should also be able to identify whether any existing SmartObjects can be re-used in the application to perform the integration.

Workflow

Workflow Start

Since a K2 process always has to be started somehow, you need to describe how the process will be kicked off. For example, is the workflow started manually by a user by submitting a list item to a SharePoint workflow, when a user completes and submits an InfoPath form, automatically when a document is added or modified in a specific library, or automatically by another K2 workflow? We will talk about some of the options available to start workflows in a later topic, for now it's just important to understand the use case for starting the workflow rather than selecting the specific implementation of it.

Workflow Flow and Steps

Designing the process flow and defining the steps of the process includes describing the work that is performed in the process (both by participants and the K2 server), the possible paths (or outcomes) in the process, and any business rules that affect the steps. (For example, an item may require only one approval, but if the request amount exceeds a certain threshold as defined in the business rule, the approval must be performed by a second person).

This design step is most commonly represented as a flow diagram with notes, shapes and notation that formalize the process. The technology and notation used to model the process flowchart doesn't matter; what is more important is that the workflow has been discovered and formalized into a flowchart.

The question you would ask during this phase would help you identify the work that needs to be performed in the workflow and who performs that work. You should also be able to identify the decision points, conditions and paths in the workflow.

A useful approach to gather the requirements for tasks in the workflow is to use a table like the example below and populate it for each user task or server task. You don't have to do this for simpler scenarios, but in a large, complex solution, clear documentation like this can really help the assembler to implement the solution.

System Task Template

<table>
<tr><th colspan="2">Step/Activity Name</th><th>System Task(s)</th></tr>
<tr><td>Step Name</td><td colspan="2">[TODO: Insert the step name as it would appear in the process design]</td></tr>
<tr><td>Step Description</td><td colspan="2">[TODO: give a high-level description of the task]</td></tr>
<tr><td>Expected Duration</td><td colspan="2">[TODO: if applicable, enter the expected duration of the step in days/hours/minutes.]</td></tr>
<tr><td>Error Action</td><td colspan="2">[TODO: If applicable, describe any specific actions that should be taken if the activity goes into error state]</td></tr>
<tr><td>Step Start Rules</td><td colspan="2">[TODO: define any business rules that impact IF and/or WHEN this step starts]</td></tr>
<tr><td>Repeat Step</td><td colspan="2">[TODO: indicate if this step should be repeated multiple times and how to determine how many times it should repeat. For example, the process may need to start multiple child processes for each order line item in a sales order process]</td></tr>
<tr><td>Step Escalations</td><td colspan="2">[TODO: define any escalation/SLA rules for the step. Note that it is rare for a server task to have any escalations, and is usually only when using asynchronous server events.]</td></tr>
<tr><td>Outcomes /Lines</td><td colspan="2">[TODO: describe the possible outcomes or lines from this step (the paths that flow out of this task).
<table>
<tr><th>Outcome Name</th><th>Rule</th></tr>
<tr><td>[TODO: add outcome name]</td><td>[TODO: Describe Outcome]</td></tr>
<tr><td>[TODO: Add Outcome Name]</td><td>[TODO: Describe Outcome]</td></tr>
</table></td></tr>
<tr><td>Server Task(s) and task settings</td><td colspan="2">[TODO: Describes the task(s) that should be performed, and specify any known values or settings that should be used in this server step. You could use the provided templates in Appendix B here, or just describe the step.]</td></tr>
</table>

User Task Template

<table>
<tr><td colspan="2">Step/Activity Name</td><td>User Task</td></tr>
<tr><td>Step Name</td><td colspan="2">[TODO: enter the step name as it will appear on the task list and reports]</td></tr>
<tr><td>Step Description</td><td colspan="2">[TODO: Enter the task description]</td></tr>
<tr><td>Expected Duration</td><td colspan="2">[TODO: if applicable, enter the expected duration of the step in days/hours/minutes. This value can be used to generate exception reports indicating where activities have exceeded the expected duration]</td></tr>
<tr><td>Error Action</td><td colspan="2">[TODO: If applicable, describe any specific actions that should be taken if the activity goes into error state, for example sending an email or raising an alert in another system. Any errors not explicitly handled in an activity will “bubble-up” to the process error action]</td></tr>
<tr><td>Step Start Rules</td><td colspan="2">[TODO: define any business rules that impact how this step starts, such as “wait for both the preceding parallel steps to complete” or “wait for 2 days before starting”</td></tr>
<tr><td>Participant(s)/ Destination(s)</td><td colspan="2">[TODO: Enter the person/role/group that will perform this task, e.g. “Originator’s Manage” or “Security Supervisors”. If business rules determine the possible participants describe the rules.
<table>
<tr><th>Participant/Destination</th><th>Business Rule</th></tr>
<tr><td></td><td></td></tr>
</table></td></tr>
<tr><td>Step Escalations</td><td colspan="2">[TODO: define any escalation/SLA rules for the step]
<table>
<tr><th>Escalation Name</th><th>Escalation time and business rules</th><th>Escalation action</th></tr>
<tr><td>[TODO: give the escalation a name]</td><td>[TODO: describe when the escalation should fire and any business rules that impact when the escalation should fire]</td><td>[TODO: describe the escalation action and use one of the provided escalation action templates to document the action.</td></tr>
</table></td></tr>
<tr><td>Instructions</td><td colspan="2">[TODO: enter the task description that will be displayed to the user]</td></tr>
<tr><td>Actions</td><td colspan="2">[TODO: type the available actions and descriptions for the actions, and whether the user must open the form to take the action or not, and whether they can action the task via E-Mail]
<table>
<tr><th>Action</th><th>Description</th><th>Must open form</th><th>Can action via E-Mail</th></tr>
<tr><td>[TODO: enter action name]</td><td>[TODO: Enter Action description]</td><td>[TODO: indicate with Yes/No whether the user must open the form to take the action]</td><td>[TODO: indicate with Yes/No whether the user can action the task via email]</td></tr>
<tr><td>[TODO: enter action name]</td><td>[TODO: Enter Action description]</td><td>[TODO: indicate with Yes/No whether the user must open the form to take the action]</td><td>[TODO: indicate with Yes/No whether the user can action the task via email]</td></tr>
</table></td></tr>
<tr><td>Outcomes</td><td colspan="2">[TODO: list the possible outcomes from this task as well as any rules around the outcomes]
<table>
<tr><th>Outcome</th><th>Business Rules</th></tr>
<tr><td>[TODO: enter outcome name]</td><td>[TODO: enter the outcome business rule]</td></tr>
<tr><td>[TODO: enter outcome name]</td><td>[TODO: enter the outcome business rule]</td></tr>
</table></td></tr>
<tr><td>Send email Notification</td><td colspan="2">[TODO: Indicate whether the task recipients should receive an email to notify them about the task]</td></tr>
<tr><td>Notification email content</td><td colspan="2">[TODO: Describe a custom notification, if required]
<table>
<tr><td>Subject:</td><td>[TODO: enter the message subject,</td></tr>
<tr><td>Message:</td><td>[TODO: enter the message body]</td></tr>
<tr><td>Additional Notes</td><td>[TODO: provide any additional notes about the notification]</td></tr>
</table></td></tr>
<tr><td>User Task Form</td><td colspan="2">[TODO: provide a link to the User Interfaces section below, where the user interface for the task is described. Alternatively, provide a link to a UI specification document where the user interface is documented.]</td></tr>
</table>

Workflow Rules

Workflows may contain other rules that also affect how the process should execute. These could include rules that define the Notifications or Escalations that need to be executed under certain conditions, or rules that define whether the process should even be started.

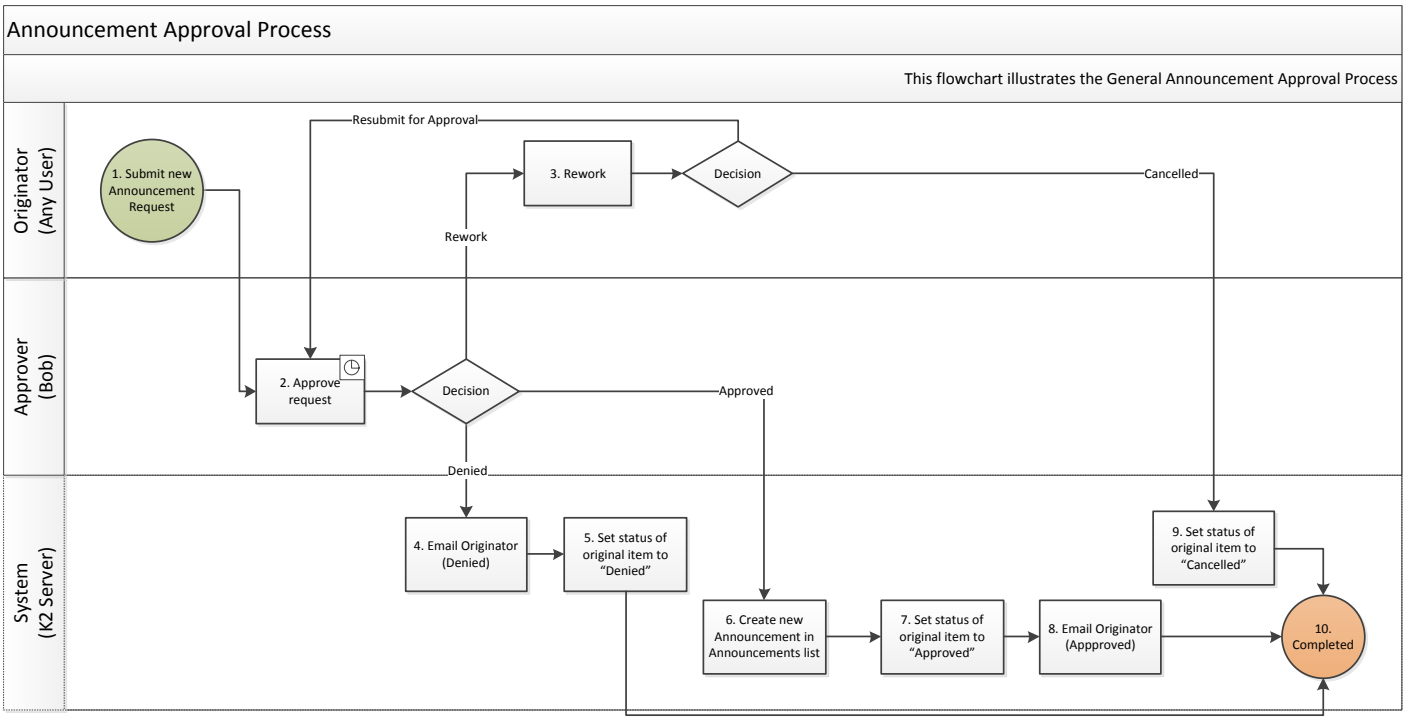
The K2 workflow engine allows these rules to be static (i.e., always apply a rule based on some specific value) or dynamic (for example, look up a value from some other system and then make a decision based on the returned value).

Workflow Diagram

The most useful design output for workflow is a workflow diagram. You can use any technology or format to describe the workflow, even a workflow drawn on paper, but most important is to describe the process flow visually.

If it is not possible to represent the process in a flowchart or process diagram, it is possible that the requirement is not a process at all. There are some design patterns (described in other learning modules) which could help to implement non-structured processes on the K2 platform, but if it is truly impossible to draw a flowchart you may need to reconsider whether implementing the solution on a workflow engine is the correct approach, or perform some process redesign and re-engineering to structure the process a little better.

Often, process diagrams appear very complicated, but workflow designers can often simplify the design with clever use of rules and outcomes. Even so, if the process flowchart is still extremely complicated with multiple outcomes and rules, it may be a good candidate for process refactoring or optimization. The image below is an example of a workflow diagram: in this case we used swim lanes and BPMN notation to describe the workflow.

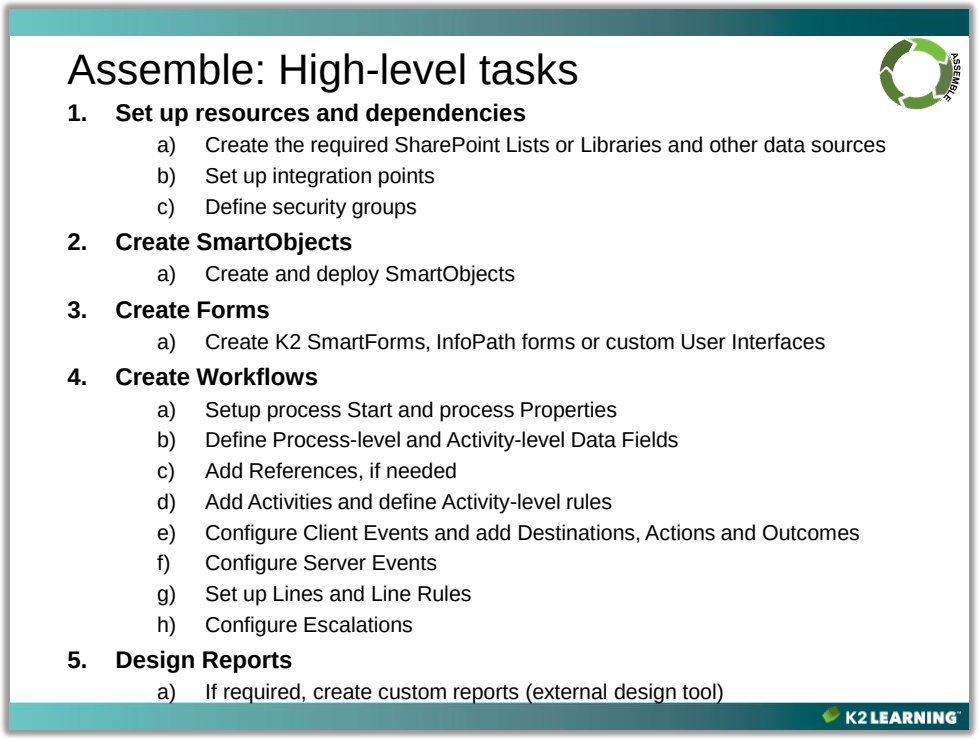


Reports

Reporting is often overlooked during the design phase, but it is an important point to consider. If there is a very particular reporting requirement, the designer should ensure that the data needed to populate the report is available, and there may be process design considerations needed in order to implement this requirement. For example, a customer report may need to include information about a customer as well as any workflows currently active for that customer. To achieve this, it may be necessary for the K2 workflow to update the customer's record in CRM so that the report can display all active processes.

The reporting design phase should also define how users will access the reports and whether any security considerations are relevant for the reports.

Assemble: High-level tasks



Once the design phase has been completed we can start to assemble the application It is better to complete as much of the process design as possible before starting to assemble the process; this will reduce the development time and the amount of development rework that may be required.

1. Set up resources and dependencies

The starting point of any K2 project is to set up the external resources that the application will use. This could include (but is not limited to):

- creating SharePoint items like Sites, Lists, Libraries and Templates that the workflow or solution will depend on
- creating or configuring data stores such as SQL databases
- creating or configuring other integration points the workflow will use to interact with external systems.
- Setting up user Roles, Groups or Active Directory artifacts that the solutions will depend on

As part of this process, ensure that the security for the prerequisite items has been configured appropriately according to the use of those systems. For example, if the workflow will interact with some system via a server event, make sure that the K2 service account has access to that system. Likewise, if you are exposing some system as a SmartObject through a service that uses the K2 Impersonation setting and the SmartObject is consumed on a form, make sure that users who will use that form have security access to the data source that the SmartObject exposes.

2. Create SmartObjects

The next step is to create and deploy the SmartObjects that will be used in the application. (If existing SmartObjects can be re-used, there Is no need to create or publish them again since SmartObjects can be re-used across different projects).

3. Create Forms

Once the SmartObjects and other data sources are available you would normally move on to creating the user interfaces for the solution. This might be a parallel step while the workflow is being developed, but it depends on the user interface technology you are using in the project. You would create the user interfaces with technologies like K2

SmartForms, InfoPath forms or custom user interfaces (Note: when using client events that generate user interfaces, it is not necessary to create the user interfaces first; they will be created by the K2 workflow automatically when the workflow is deployed.) The following Forms technologies are commonly used in K2 applications:

1. Auto-generated SharePoint Workflow Forms for processes based on SharePoint Workflow Integration (in other words, processes that are automatically or manually started from a SharePoint list item or SharePoint document)
2. Microsoft InfoPath forms, for processes based on a SharePoint Form Library or an InfoPath form template.
3. K2 SmartActions, where users can use E-Mail messages to perform actions on K2 tasks. Strictly speaking, K2 SmartActions aren't really a user interface since users just open the email and reply to the email with their decision, but you should bear this capability in mind when deciding how users will complete their tasks.
4. Auto-generated K2 ASP.NET pages. K2 blackpearl can generate ASP.NET pages that already contain the necessary code to interact with the workflow. Developers can customize these ASP.NET pages to some extent to change the behavior and style of these pages.
5. Custom forms created by developers, using the K2 APIs and Services to interact with the workflow.
6. K2 smartforms, which is a separate add-on component for K2 blackpearl allowing no-code development of web-based user interfaces.

4. Create Workflows

Once the prerequisites, SmartObjects and Forms are in place you can use K2 Studio to create a new K2 process. You would provide the process settings such as Name and Description and define any data fields that will be required in the workflow. You would also add any required references to services or assemblies at this stage, if you plan to use the Code Reference Event to interact with an external assembly.

This is also the appropriate stage to use a Process Wizard and associate the **Start** of the process with some event, if you are creating an InfoPath- or SharePoint-based process.

The majority of the work in assembling a K2 process lies in adding and configuring the various Activities, Events, Rule, Lines, Escalations and other items that make up the workflow. If the prerequisites are in place, most of the assembly of a K2 process is exactly that: assembling the process from pre-built components by running wizards or configuring properties. For example, designers can use the pre-built K2 wizards to integrate with SharePoint, or use SmartObject to integrate with some external system.

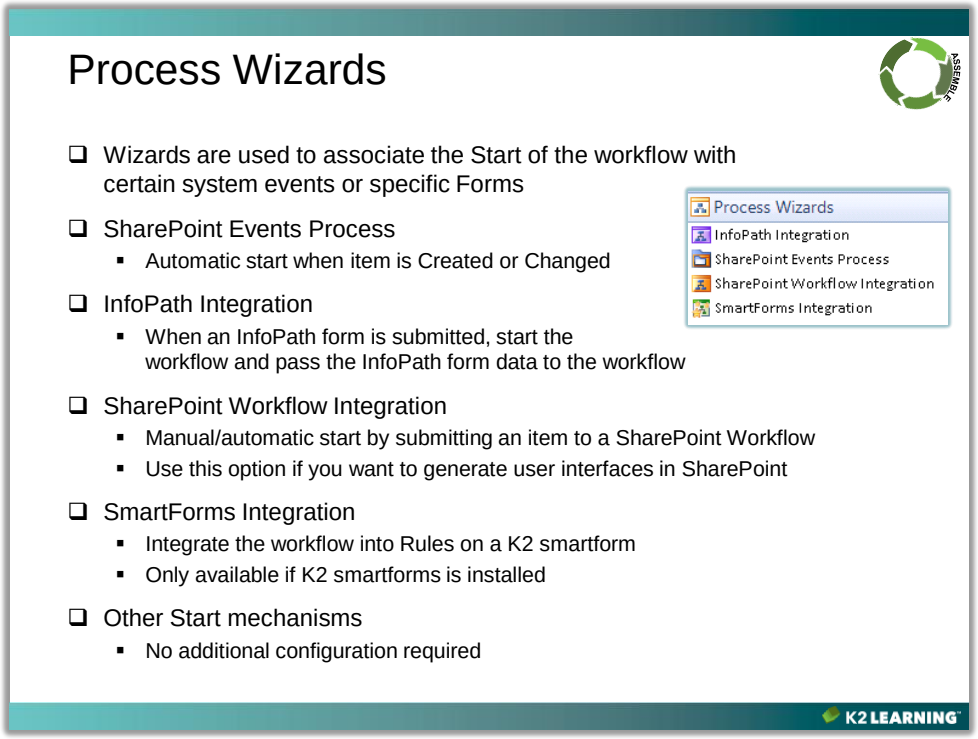
Over time, the store of available Processes and SmartObjects grows as more and more K2 solutions are deployed in an organization. A logical and controlled naming and categorization system will help reduce the complexity of a K2 environment and make it easier to manage the environment and find the re-usable components much more easily.

While you are free to select your own naming convention, the quick reference sheet QRS.023 - Naming conventions for K2 Artifacts suggests a naming convention that you may implement or modify to suit your requirements.

5. Design Reports

The final stage in assembling a K2 application is to design any custom reports, if required. You could use external report design tools to build these reports, or the built-in report design tool in K2 workspace. (Note: if you use the report designer in K2 workspace, the K2 artifacts will need to be published to the K2 server first).

Process Wizards



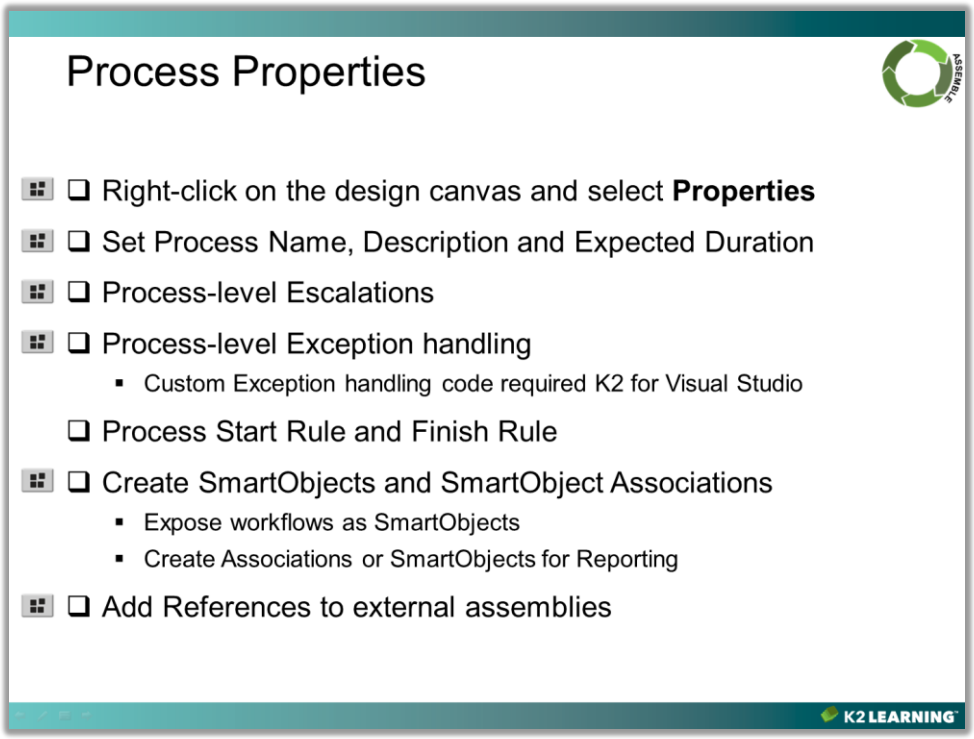
Certain workflow-start mechanisms require additional configuration so that K2 can set up the required integration points to start the process. If you are using any of the following start mechanisms, you must use the associated Process Wizard to configure the start mechanism:

Wizard	Workflow Start Mechanism	Notes
SharePoint Events Process	Automatically start a process when some event occurs in SharePoint List or Library e.g. Item Added, Document Uploaded or Field Added	Upon deployment of the workflow, K2 will add the integration points necessary to start the process automatically to the specified item. As part of the wizard, you will need to specify the list, library or content type that the process should be associated with, and the events that will cause the process to start. In advanced mode, the wizard will allow you to associate the process with events in multiple lists or libraries.
SharePoint Workflow Integration	The K2 process will be associated with a SharePoint workflow. Users will submit an item to a SharePoint workflow to start the K2 process, or the workflow will be associated with a SharePoint List, Library or Content Type. The workflow can also be automatically started when an event happens in SharePoint, for example Item Added or Item Updated.	Upon deployment of the workflow, K2 will create the necessary integration points with the SharePoint environment to register the SharePoint workflow which will start the K2 process. When using this approach, K2 will generate pages in either InfoPath or ASP.NET format for the client events.

InfoPath Integration	A user will complete and submit an InfoPath form (either through Forms Services in a web browser or through InfoPath client) to start the K2 process.	K2 will insert rules into the InfoPath form that will start the K2 workflow when the form is submitted. When the InfoPath form is submitted to K2, the form XML is saved into the process as a datafield and K2 will re-generate the form using that XML for the client events in the process.
smartform Integration	The K2 workflow will be associated with a K2 smartform.	K2 will insert Rules into a smartform so that the workflow is started when the user performs an action on the form. *Note: you don't have to use the wizard to integrate the workflow with a smartform; you can define the integration rule manually, too.*

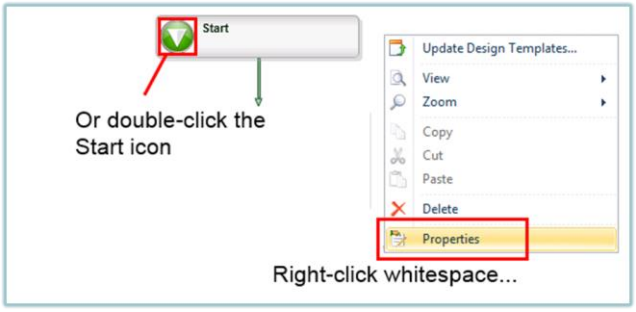
If you are starting the workflow in another way (e.g. a user uses a custom application to start the workflow, the workflows is started by another K2 process using the IPC event, or the workflow is started by a schedule, an external event or some other code) it is not necessary to run any wizard to set up the start of the process.

Process Properties

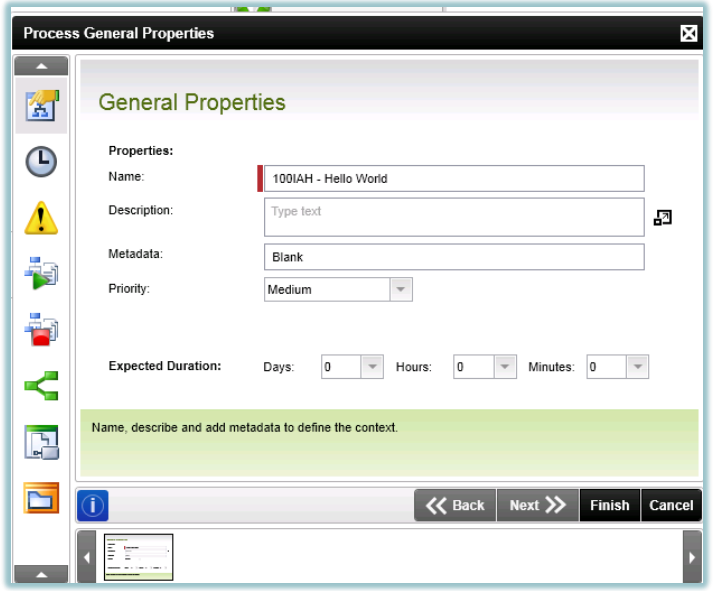


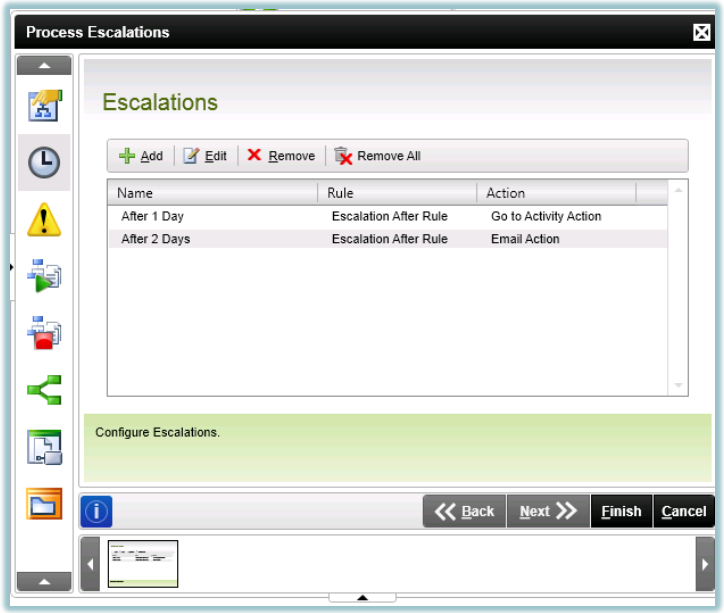
By right-clicking on the process design canvas or double-clicking the Start icon in the workflow, you can access and set additional properties for the process. This is where you could provide a name, description or expected duration for the process, define any process-level escalations or add references to the project so that you can call the references with the Code Reference Event or with custom code.

Opening Process Properties

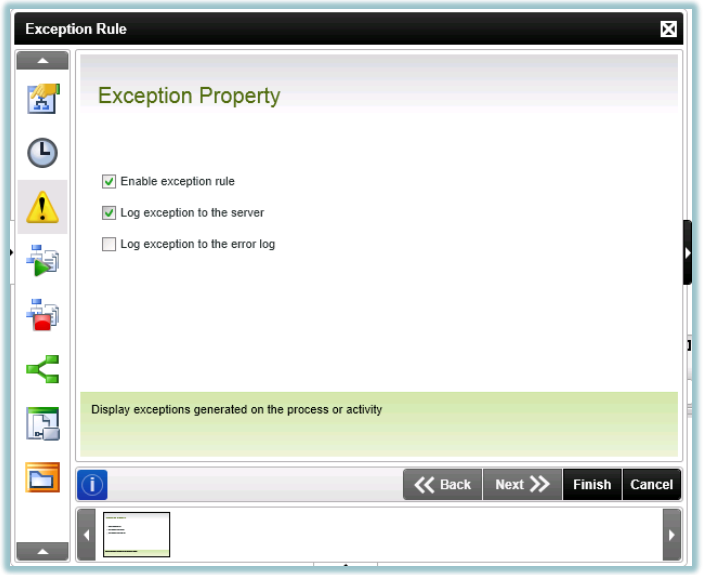


Screen	Field	Usage
General Properties  *Note: These properties cannot be modified at runtime. They are statically defined as properties of the process.*	Name	The display name of the process. Used in reports and user interfaces. This name should be unique for each process in the K2 environment.
	Description	A friendly description of the process. Not normally visible anywhere, but could be shown on custom interfaces
	Metadata	Any additional metadata for the process. This value is typically not displayed anywhere, but could be used in custom interfaces or custom code.
	Priority	The priority level of the process (High/Medium/Low). Normally used to sort tasks lists and reports based on the process priority.
	Expected Duration	How long the process is expected to be active. This value is often used on reports and other user interfaces to indicate when processes have exceeded the expected duration.
Escalations 	(Not applicable)	Used to define process-scope escalations. Use the **Add** button to add one or more escalations for the process as a whole.

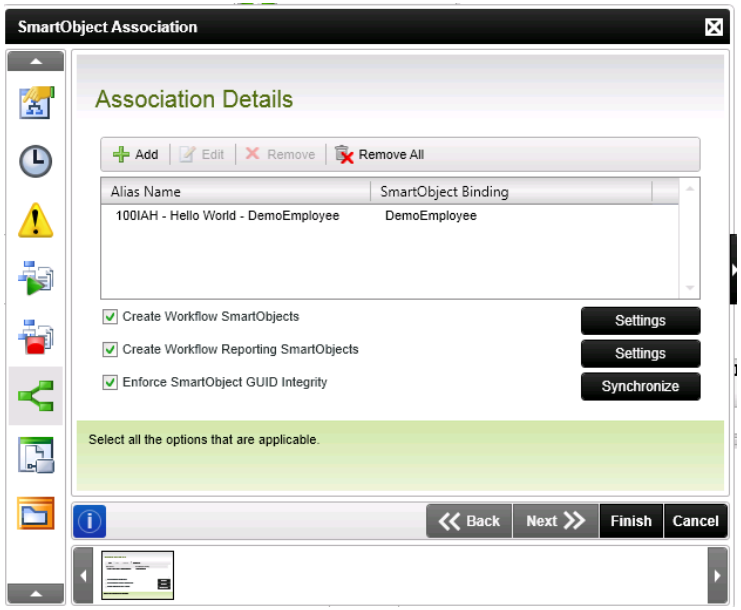


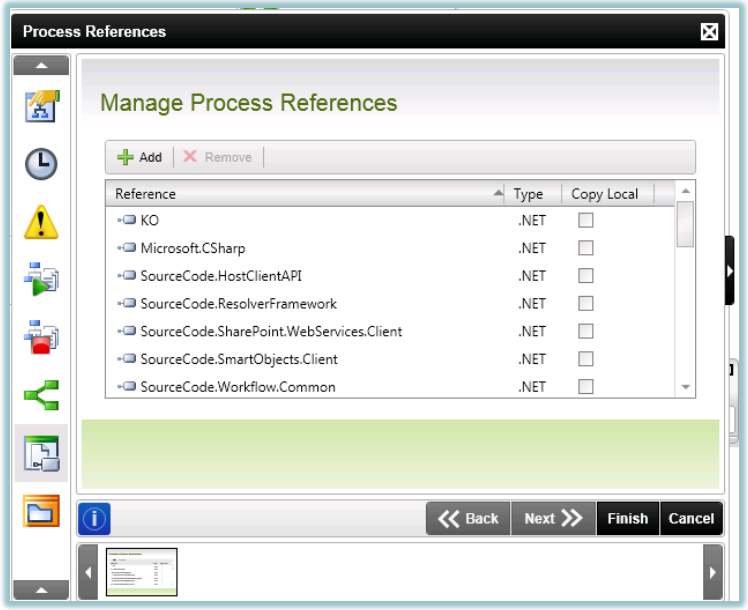


Screen	Field	Usage
Exception Property 	Enable Exception Rule	Used to enable custom error handling for the process. If this value is not selected and the process goes into error state, the process will be added to the K2 error log and be suspended until an administrator repairs the process. Click this button to enable custom code for error handling although you will need to use K2 for Visual Studio to write the error handling code.
	Log Exception to the Server	If this box is selected and the process goes into Error state, a message will be logged to the K2 log framework. Depending on the logging framework configuration, the error may then be written to the event log, text file or other target.
	Log Exception to the Error Log	If this box is selected and the process goes into Error status, the process will be added to the list of processes in error status. This list is shown on the Error Log screens in K2 Workspace and the K2 Process Portal.
Start Rule 	Enable Process Start Rule	Select this option to write custom code to determine whether the process should start. Only useful when creating processes with K2 for Visual Studio.
Finish Rule 	Enable Process Finish Rule	Select this option to write custom code to determine whether the process should finish (complete). Only useful when creating processes with K2 for Visual Studio since this requires custom code.

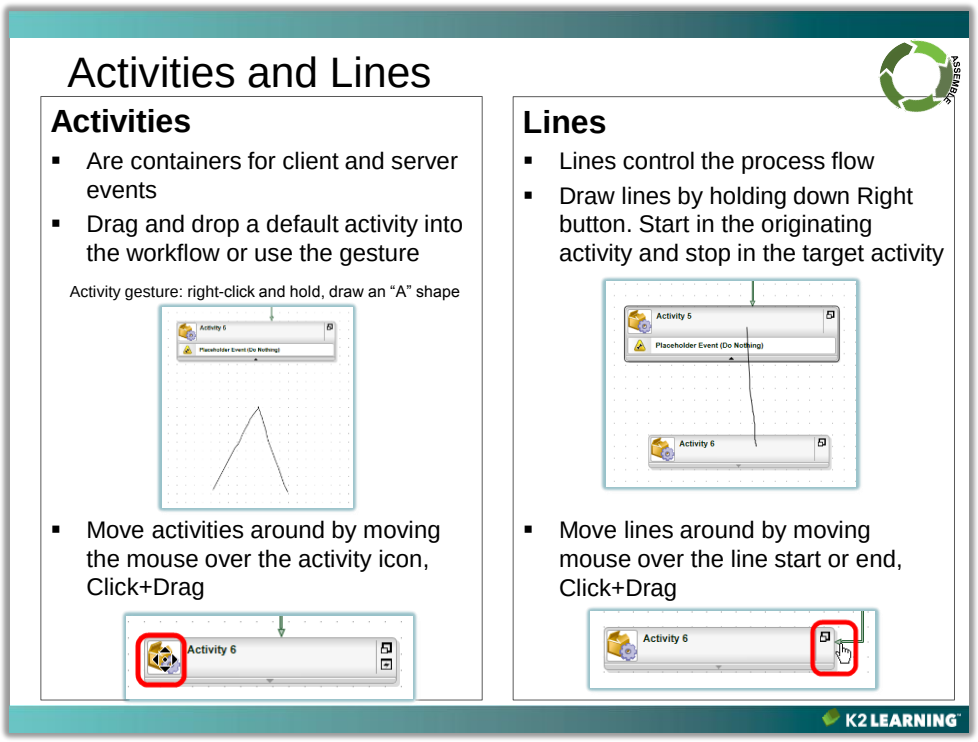


Association Details 	SmartObject Association List (**Add** Button)	Use this to associate the workflow with other SmartObjects in the K2 environment, This normally used to create relations (e.g. one-to-one, one-to-many) between the workflow SmartObjects and the target SmartObject for reporting and user interface purposes.
	Create Workflow SmartObjects	Creates a SmartObject that represents the process' methods. Most commonly, this Is used to expose the workflow "Start" event as a Method of a SmartObject so that the workflow can be called by anything capable of consuming SmartObjects. You can also expose the Client Events in the process as additional SmartObjects so that they can be consumed by other applications. This is most commonly used to allow applications to action client events though a SmartObject method call.
	Create Workflow Reporting SmartObjects	Creates Reporting SmartObjects for the workflow as well as any selected activities and events in the workflow. This is most commonly used to create SmartObjects that will be used to create custom workflow reports. Note that this is Reporting data, in other words, data such as the Activity Start Time, Event End Time and so on.
	Enforce SmartObject GUID integrity	Ensures that the GUID identifiers of the generated SmartObjects and Associations are consistent across all the environments that the process is deployed to. It is recommended to leave this option checked. The **Synchronize** button will force K2 to discover any existing SmartObjects and update the ID's in the process definition.



Manage Process References 	(Not applicable)	Use this screen to add references to assemblies (.NET and COM) and services (services with WSDL and WCF services). Once the reference is added, you may use the Code Reference Event to call constructors and methods for the added assemblies, or use K2 for Visual Studio to write code against the assembly or service.	

Activities and Lines



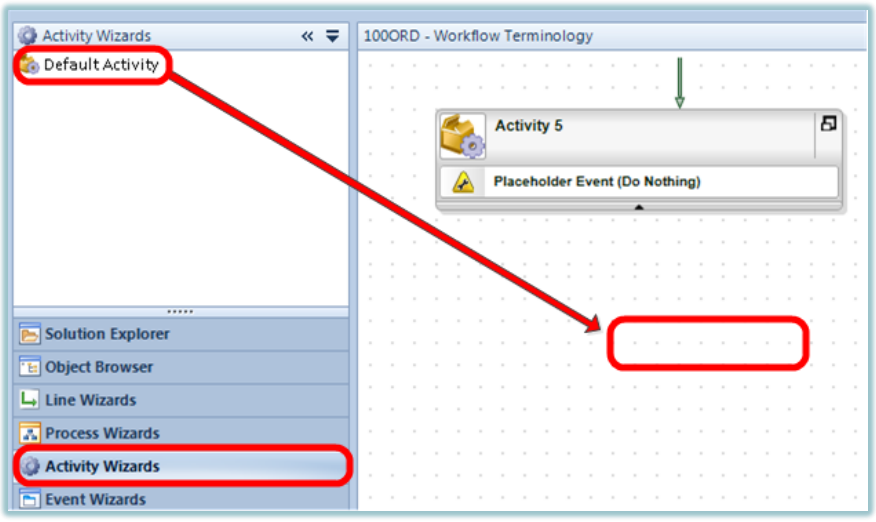
Activities and Lines are the basic building blocks used to define the steps and flow of the process. Remember that the Lines and Activities themselves do not normally perform any work; the actual work in the process is done in Events and Activities are containers for the Events.

Activities

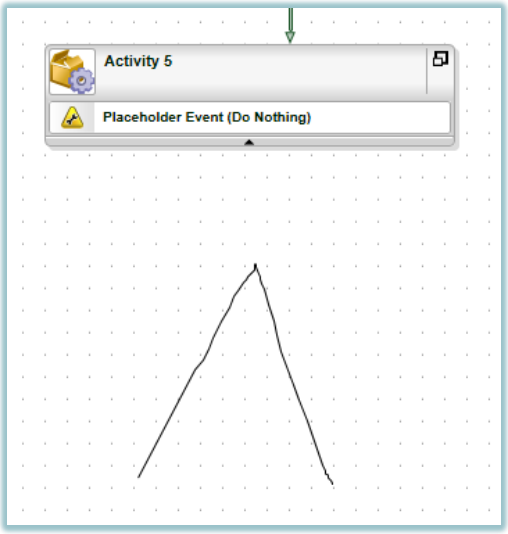
Once you have defined the basic properties of the process, you would normally start to lay out the process by dragging activities onto the design canvas, or by using the mouse gesture shortcut Λ to create a new activity on the design canvas.

To use mouse gestures, move the mouse over the design canvas, click and hold the right button on the mouse and draw a shape. The designer can recognize several different shapes and we will not those as we discuss the wizards.

Dragging the Activity Wizard onto the design canvas to create a new step in the workflow

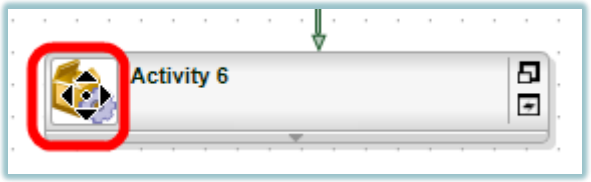


Using the mouse-gesture shortcut Λ to draw a new activity on the design canvas



You can drag Activities around by moving the mouse over the activity icon and, when the mouse changes to a positional arrow (see screenshot below), click and hold the left-hand mouse button to drag the activity around, and release the button at the desired location. Note that K2 will keep any lines that are connected to the activity intact.

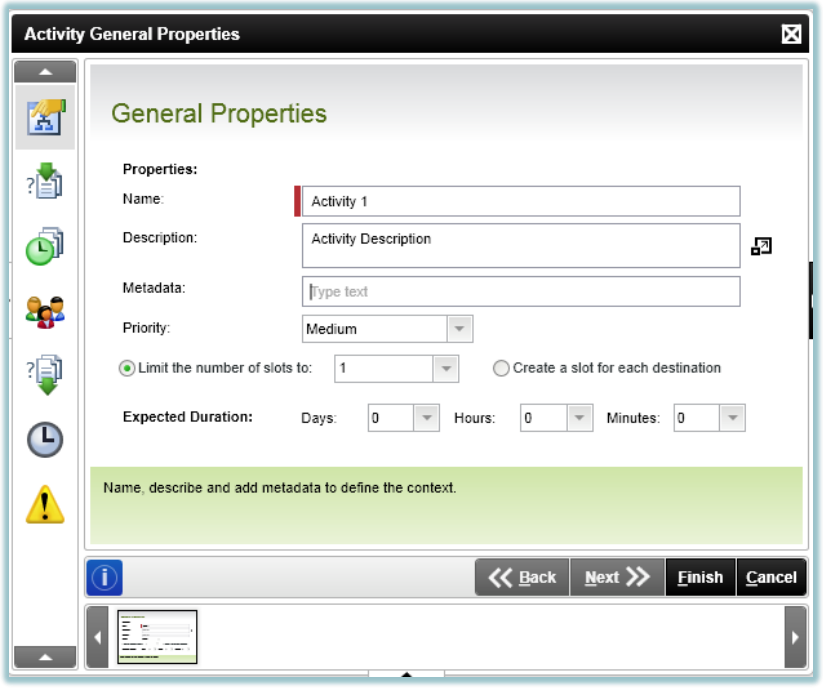
Moving an Activity around



Activity Properties

There are a few Properties you can define on an Activity. Some of these are defined as Rules which will control the execution of the activity at runtime. Right-click any Activity and select Properties to view the Rules for that activity.

The Activity Properties screen



The table below lists the various properties and rules for an activity with their use.
Note: Most of these rules are described in more detail in the module **200.AUS Building Workflows with K2 Studio - Intermediate**

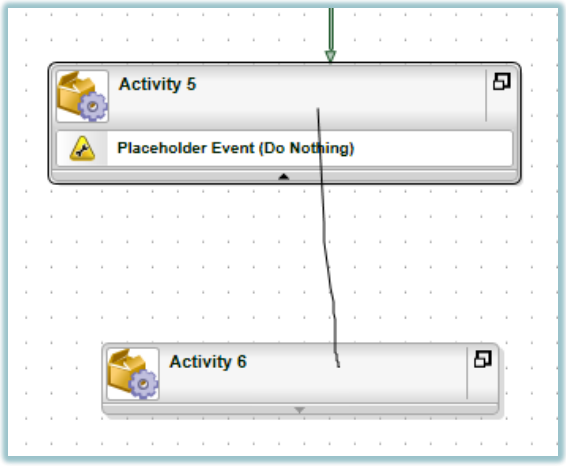
Screen	Field	Usage
General Properties  *Note: These properties cannot be modified at runtime. They are statically defined as properties of the activity.*	Name	The display name of the activity. Used in reports and user interfaces. This name should be unique for each activity in the workflow.
	Description	A friendly description of the activity. This value is often used to describe the purpose or behavior of the activity to make the process self-documenting.
	Metadata	Any additional metadata for the activity. This value is typically not displayed anywhere, but could be used in custom interfaces or custom code.
	Priority	The priority level of the activity (High/Medium/Low). Normally used to sort tasks lists and reports based on the activity priority.
	Number of Slots/Create a Slot for each destination	Determines how many active "copies" of the Activity will exist at runtime. Most often this value is set to 1, but can be dynamic for more advanced scenarios like multiple approvals or repeating activities.
	Expected Duration	How long the activity is expected to be active. This value is often used on reports and other user interfaces to indicate when activities have exceeded the expected duration.
Preceding Rule 	n/a	Determines IF the activity is allowed to start. Most often used in parallel execution environment to ensure that both preceding paths have completed before allowing the activity to start.
Start Rule 	n/a	Used to delay the start of an activity by some relative values (Days/Hours/Minutes or Seconds) or some absolute value (a specific date and time)
Destination Users 	n/a	Define the participants for the activity. In advanced mode, this configuration is used to implement more advanced requirements like multiple Destination Sets, Destination Set Rules and multiple instances of the same activity (Slots).
Succeeding Rule 	n/a	Determines WHEN the activity is allowed to finish. In more advanced cases you could use this rule to collect multiple inputs from multiple users, or expire all remaining user tasks if a certain condition is met (for example: as soon as 1 user Declines the task, don't wait for anyone else's input)
Escalations 	(Not applicable)	Determines the action(s) to perform if the Activity has not finished within a timeframe or by a particular time.
Exception Property 	Enable Exception Rule	Used to enable custom error handling for the activity. If this value is not selected and the activity goes into error state, the process will be added to the K2 error log and be suspended until an administrator repairs the process. Click this button to enable custom code for error handling although you will need to use K2 for Visual Studio to write the error handling code.

Screen	Field	Usage
	Log Exception to the Server	If this box is selected and the activity goes into Error state, a message will be logged to the K2 log framework. Depending on the logging framework configuration, the error may then be written to the event log, text file or other target.
	Log Exception to the Error Log	If this box is selected and the activity goes into Error status, the process will be added to the list of processes in error status. This list is shown on the Error Log screens in K2 Workspace and the K2 Process Portal. If this box is left un-checked, errors in the activity will not force the process into Error state

Lines

Once you have added Activities to the process, you can use Lines to join the activities together. The easiest way to draw a Line is to move the mouse over the activity where the Line should start, then click and hold the right mouse button and draw a Line to the Activity where you want the Line to end. You should always start with the mouse hovering over an activity, and preferably release the mouse over the target activity so that K2 knows where to draw the line.

Drawing a Line between two Activities

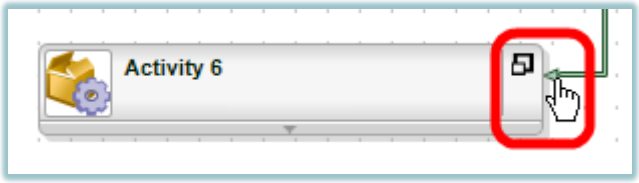


You do not need to draw lines between Activities where Client Events (user tasks) will be defined. This is because the Client Event wizard has auto-generation capability that will generate the lines and the associated line rules for you.

You will see that K2 “snaps” the Line into place, indicating that the Activities were successfully joined. You can always drag the line’s start- or end-point around to lay out the lines as you prefer. As long as the Line stays connected at both ends, it does not matter how the line is laid out. To move a line, hover the mouse over the Line’s start-point or end-point and then left-click and drag the line to the preferred place. When dragging the end point (the end with the arrow head), you can also un-snap the Line from an Activity and snap it to a different Activity, which will change the process flow.

You cannot un-snap the start-point of a Line from an Activity. If you needed to move the start of a line to another Activity, **Copy** and **Paste** the line into the new Activity, and then delete the original line.

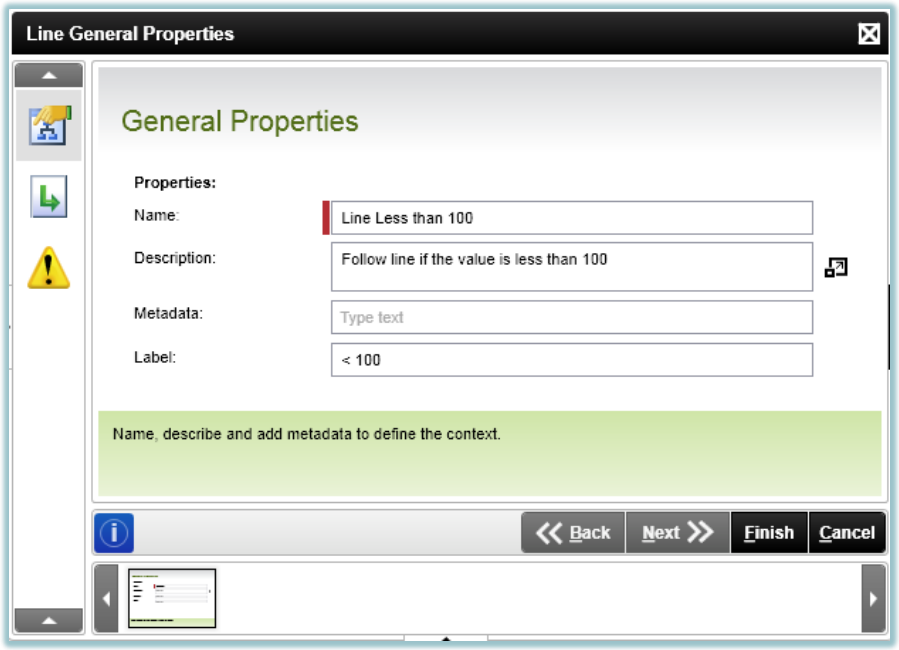
Moving a line's endpoint around



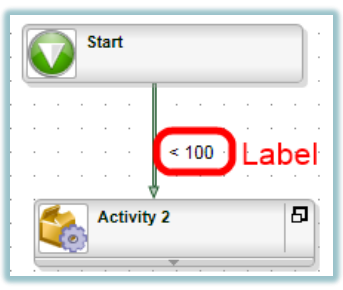
Line Properties

There are a few properties you can define on a Line. Right-click any Line and select Properties to view the Rules for that Line.

The Line Properties screen



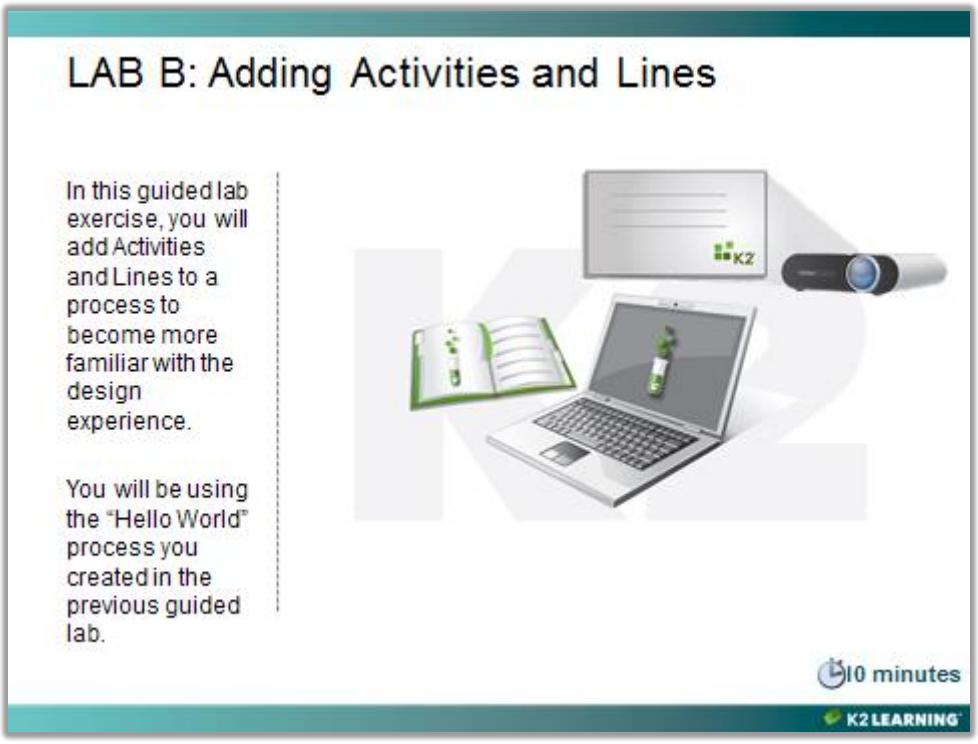
The table below lists the various properties and rules for a Line with their use.

Screen	Field	Usage
General Properties  *Note: These properties cannot be modified at runtime. They are statically defined as properties of the activity.*	Name	The display name of the Line.
	Description	A friendly description of the Line. Not normally visible anywhere, but could be shown on custom interface or used to document the behavior of the Line.
	Metadata	Any additional metadata for the Line. This value is typically not displayed anywhere, but could be used in custom interfaces or custom code.
	Label	This is a value displayed for the line on the process design canvas and view flow reports. If a line has a Line Rule, we recommend that you provide a label for the line to make the process self-documenting. 

Screen	Field	Usage
Line Rule 	n/a	Add one or more rules that define whether the line is followed. This Rule is often used in workflow implementations to control the execution path of a workflow.
Exception Property 	Enable Exception Rule	Used to enable custom error handling for the Line. The behavior is the same as for Activity- and Process-level Exception Rules.

There are many features in the K2 Designer that will help you lay out a process logically and format the layout so that it is easier to read. In the next module that deals with the K2 Studio designer (**200.AUS Building Workflows with K2 Studio – Intermediate**) you will learn how to use these features and some recommended practices to create logical, well-formatted and self-describing processes.

LAB B: Adding activities and lines



Objective

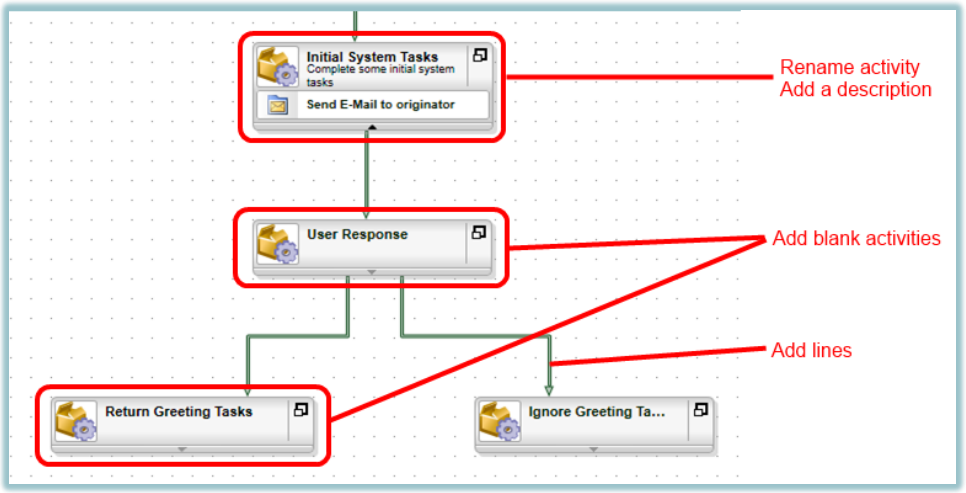
This guided lab will let you become more familiar with Activities and Lines and formatting your process by moving these items around and renaming them.

Duration

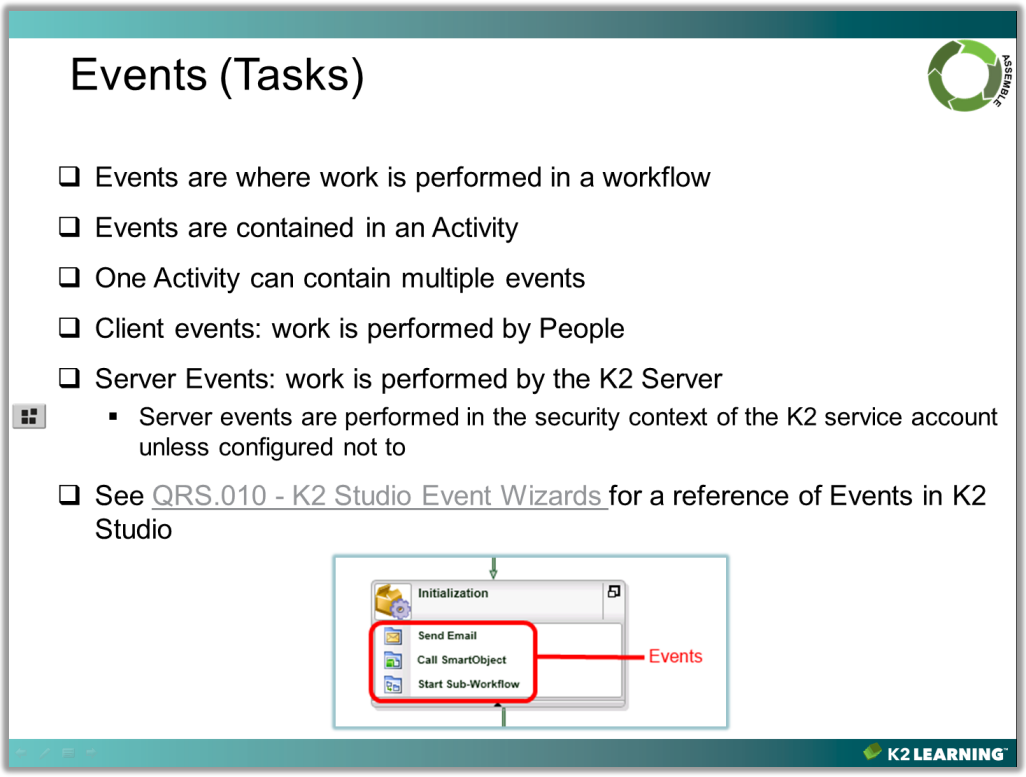
This lab should take around 10 minutes to complete. Although a guide to complete this guided lab is provided, you should preferably follow along with the instructor to complete this exercise, rather than reading the guide.

Context

You will extend the existing "Hello, World" process that you created in the previous exercise by adding more Activities and Lines and formatting the process layout.



Events (Tasks)



Events usually contain the work that will be performed in the workflow. These events could be performed by People (in which case they are known as Client Events) or the K2 server (these are known as Server Events). Events are always contained in an activity, and an activity could contain more than one event.

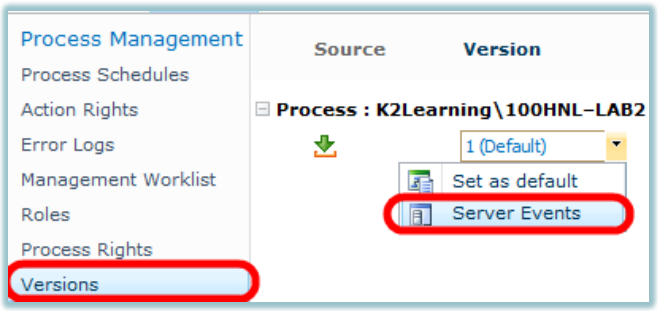
You would normally create separate activities whenever works moves between participants. For example, if you had a sequence of Client Events and then a sequence of Server Events, the Client Events would be in a separate Activity than the Server events. You can mix Client Events and Server events in the same activity if needed, though.

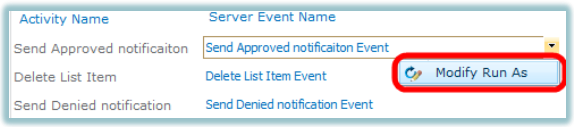
There are a wide selection of event wizards available in K2 Studio. For reference, use the quick reference sheet QRS.010 - K2 Studio Event Wizards which describes the common uses for each event wizard.

Security context of Server Events

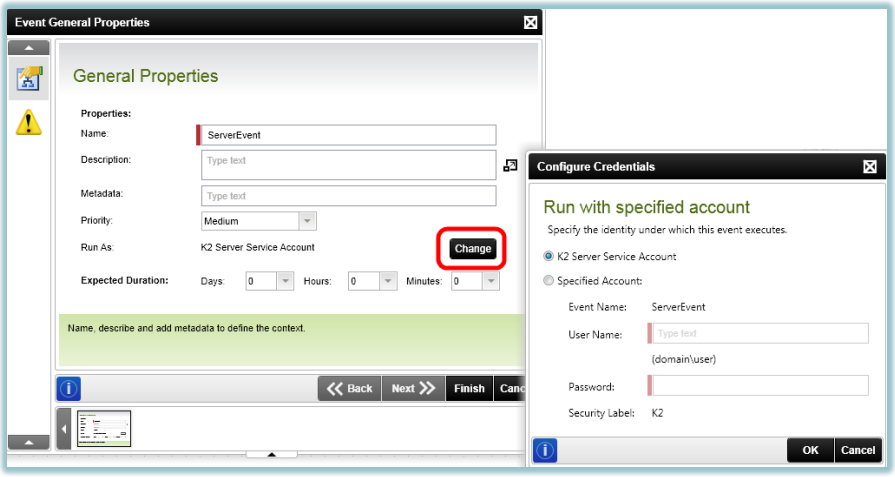
K2 workflows normally execute in the context of the K2 service account. It is possible to override the security context for a specific event in a workflow using the K2 administration console or setting the properties for the Event at design time. This feature is most often used when K2 interacts with a system that has very restricted security and the K2 service account cannot be given access to that system.

Changing the security context of a server event using K2 process portal





Setting the security context of a server event at design-time

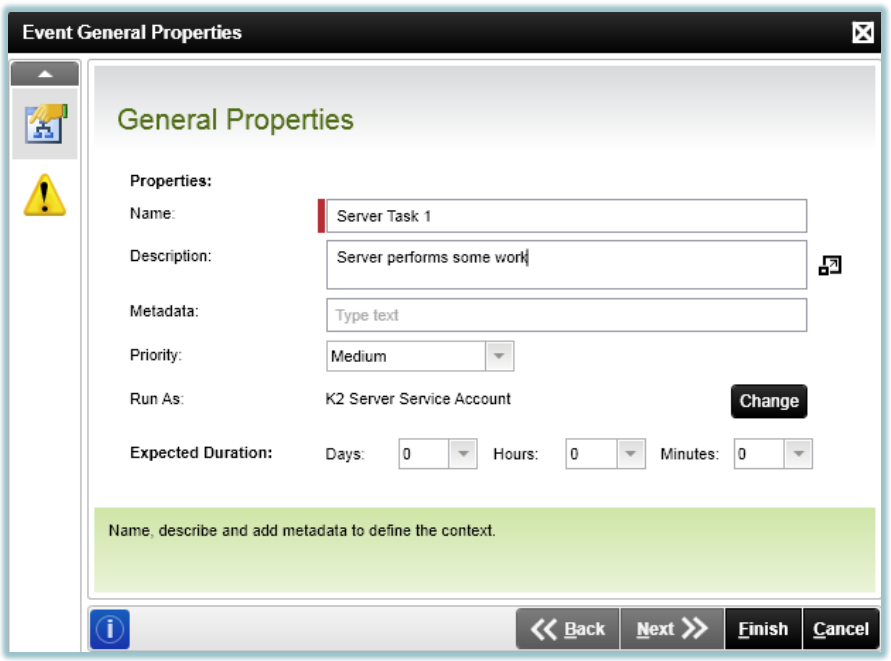


Note: It is not possible to determine the security context for an event dynamically using this configuration. The only way to do so is to write custom code in the event using K2 for Visual Studio.

Event Properties

Just like the Process, Activities and Lines you can right-click an Event to define additional Properties or Rules for the Event

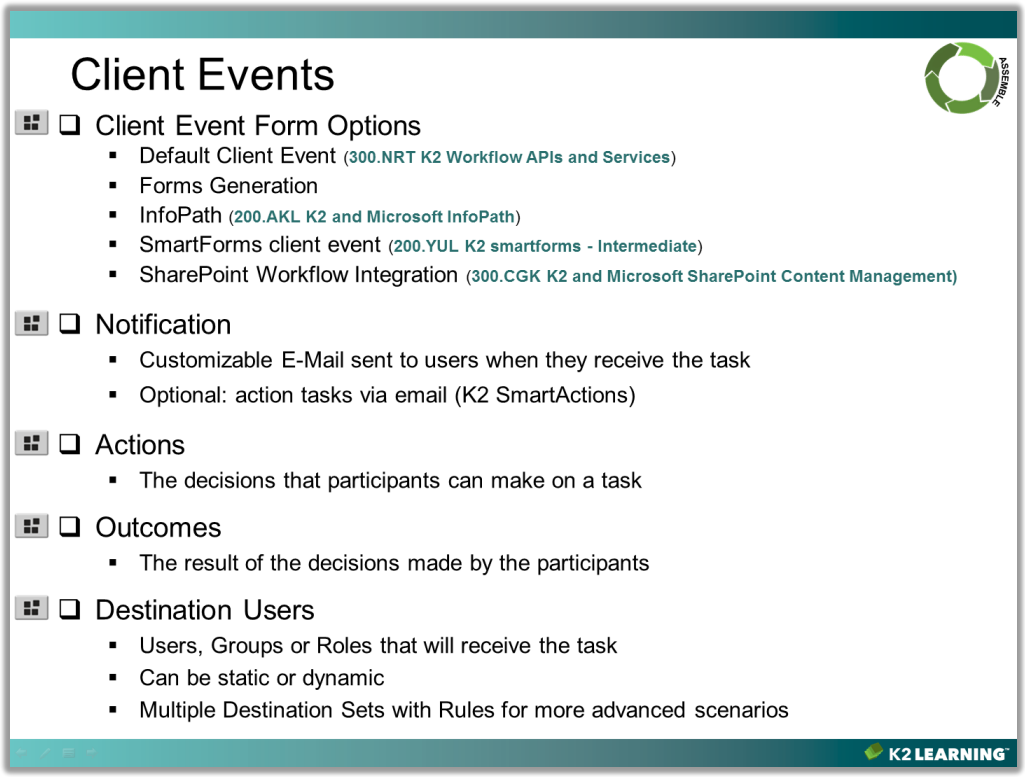
Event Properties screen



Screen	Field	Usage
General Properties 	Name	The display name of the Event. Used in reports and user interfaces. This name should be unique for each Event in the same Activity.
	Description	A friendly description of the Event. Not normally visible anywhere, but could be shown on custom interfaces or used to document the Event to make the workflow self-describing

Screen	Field	Usage
Note: *These properties cannot be modified at runtime. They are statically defined as properties of the process.*	Metadata	Any additional metadata for the Event. This value is typically not displayed anywhere, but could be used in custom interfaces or custom code.
	Priority	The priority level of the Event (High/Medium/Low). Normally used to sort tasks lists and reports based on the process priority.
	Run As	Used to change the credentials that the Event will run under. These credentials are statically defined and cannot be changed dynamically at runtime.
	Expected Duration	How long the Event is expected to be active. This value is often used on reports and other user interfaces to indicate when Events have exceeded the expected duration.
Configure Actions and Outcomes 	(Not applicable)	For client events only. Define WHICH actions a user may take and what the result of those actions is. The next topic describes these in a little more detail.
Escalations 	(Not applicable)	For client events, determines the action(s) to perform if the Event has not finished within a timeframe or by a particular time.
Exception Property 		This is used to define event-level error handling. The behavior of the event is the same as for other Exception rules. Remember, if the **Log Exception to the Error Log** box is left un-checked, errors in the Event will not force the process into Error state, which is a nice way of suppressing any Event-level errors for non-critical events and make the process more robust at runtime.

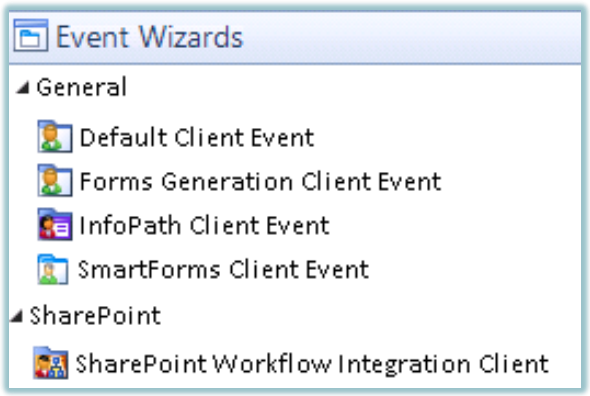
Client Events



When humans need to perform tasks as part of a workflow, Client Event wizards are used to define the task and set various properties of the task. These properties will control how the task is allocated, performed and completed. K2 Studio provides several different Client Event wizards that can be used to define these human tasks. (Note that the list of available client events may depend on any product add-ons you have installed)

The first item to consider when defining a human task step is how that task will be performed: what user interface or form the user will use to complete the task. Depending on the technology used, you would use different wizards to define the user task and the form associated with the task. Let's look at the available options and how they are most commonly used.

K2 Studio Client Event Wizards



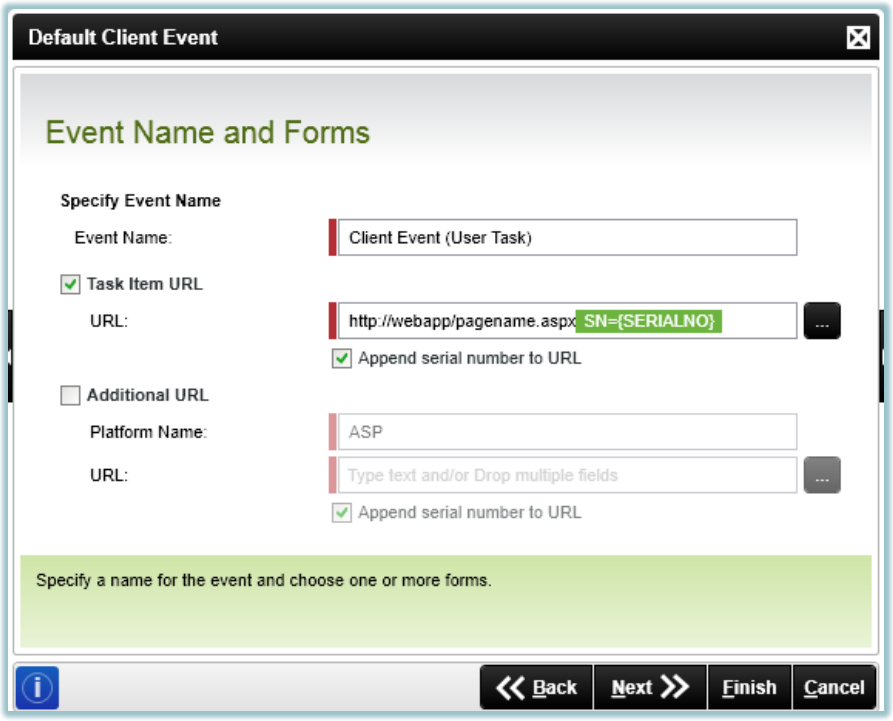
Client Event Form Options

Default Client Event

Use this event when you will manually create a custom User Interface, or when you do not need to create a custom page because the user will complete the task by replying to an Email or by completing the task directly from the K2 worklist. This wizard is typically used to direct the user to a web page; the web page contains the logic to complete the

task. The module **300.NRT K2 Workflow APIs and Services** explains how to create an ASP.NET application that uses the default client event to associate these user tasks with a custom web page.

Specifying the page URL in the Defautl Client Event

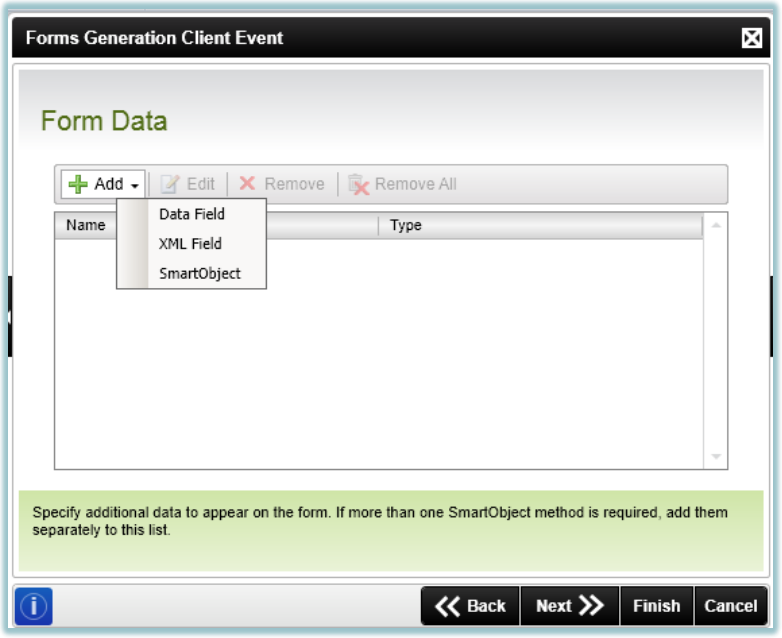


Forms Generation Client Event

Use this wizard when you want K2 to automatically generate and publish a customizable ASP.NET page for the user task. This wizard is most often used when cannot or do not wish to create a complex user interface.

When using this option, the K2 project must be deployed on the K2 server because K2 uses certain deployment methods that can only work locally on the K2 server.

Adding data to the form generated by the Forms Generation Client Event



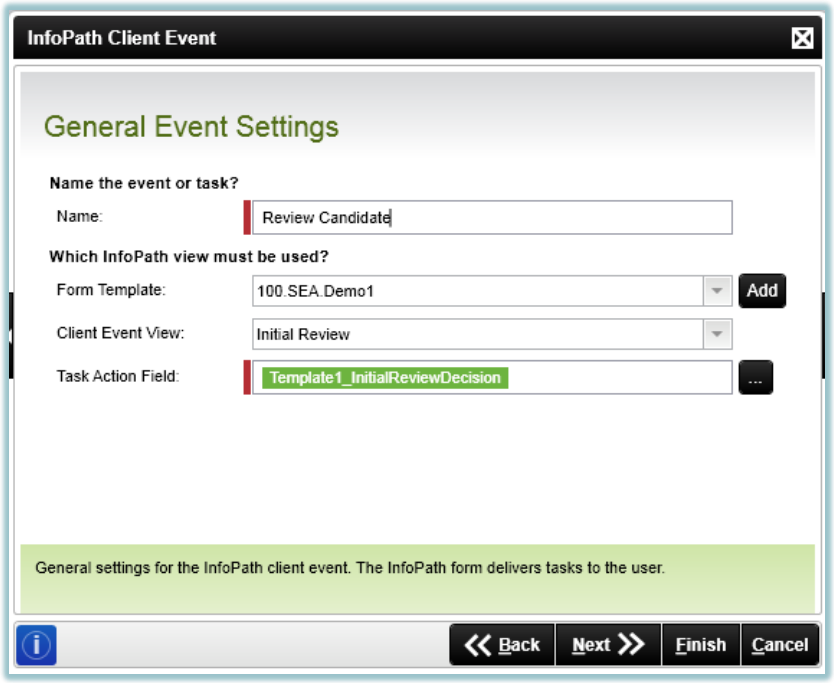
01001011 00110010	K2 creates the generated pages in a local project folder first, and this folder will be to create a web deployment project when the workflow solution is published. If you want to customize the pages generated by K2, open the folder `CSWebsiteClient\website` in the project and edit the `.css, .aspx.` and `.cs` files to customize the layout of the generated pages. When K2 deploys the project, it will use the customized pages.

InfoPath

The InfoPath Client Event is used when the user will complete the task using a view on an InfoPath form. This form could be published somewhere on the network or as a form template in a SharePoint library. When using InfoPath Forms Services on SharePoint, the form can even be rendered as a web page so that the user does not need to have InfoPath installed on their computer to open the page.

This wizard is very often used when you want to create a rich user interface with more advanced data manipulation, validation and processing rules. The biggest benefit of using this technology for the user form is that you do not need to write any code to interact with K2. K2 will automatically add the necessary logic and processing to "hook" the form up to the workflow. The K2 learning module **200.AKL K2 and Microsoft InfoPath** covers InfoPath integration in much more detail.

Specfying the form template and view in the InfoPath Client Event wizard

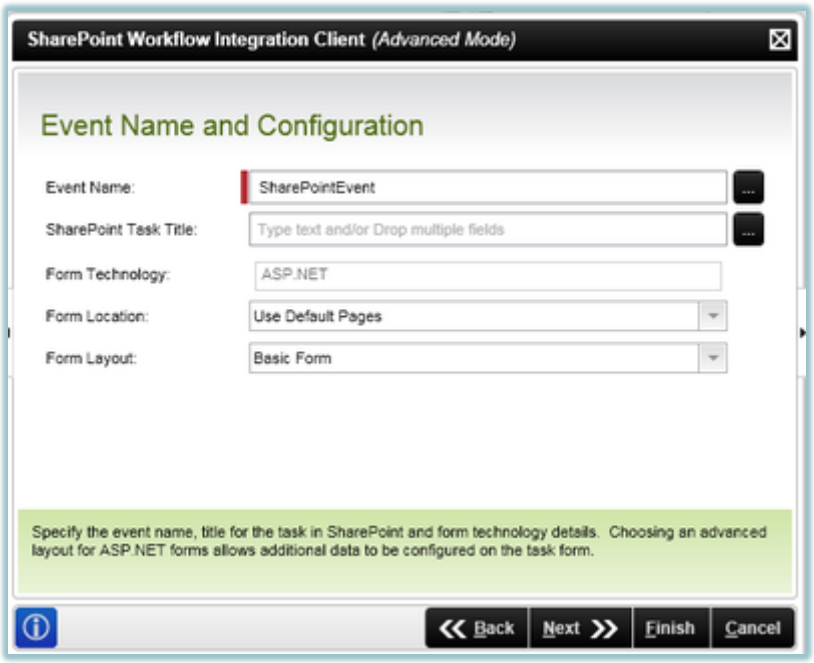


01001011 00110010	K2 does not support nor recommend using code-behind for InfoPath forms that are K2-enbled. While it IS possible to add code-behind to K2-enabled InfoPath forms, it is not a simple process and there are several step involved. Refer to the K2 KB article `http://help.k2.com/en/KB001244.aspx` for more on this topic.

SharePoint Workflow Integration

Another option for user forms is the SharePoint Workflow Integration Client Event wizard. This option is only available when you have used the **SharePoint Workflow Integration** process wizard to expose the K2 process as a SharePoint workflow. K2 will use a generated page in SharePoint to present the task to the user, creating a seamless, SharePoint-centric user experience. This event template is described more in the module **300.CGK K2 and Microsoft SharePoint Content Management.**

Setting the form options for the SharePoint Workflow Integration Client Event



01001011 00110010	The wizard can generate pages in both InfoPath and ASPX format. The ASPX format creates .ascx files which can be customized using Visual Studio. Refer to the K2 help topic **Concepts > Forms > SharePoint Forms > Form Technology, Form Location and Form Layout** for more information on customizing the layout of SharePoint Client Event pages.

SmartForms Client Event

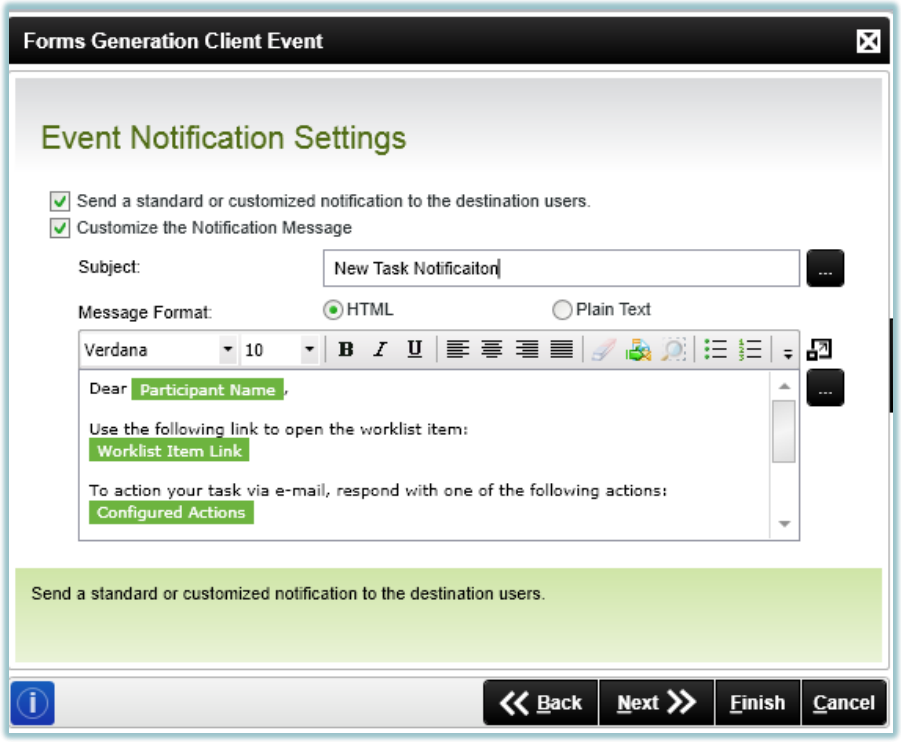
If K2 smartforms product is installed, you also have the option of using smartforms as the user interface for a client event. This option is described in much more detail in the learning modules that cover K2 smartforms, and particularly **200.YUL K2 smartforms – Intermediate.**

Notification

The K2 Client Event wizards will allow the workflow designer to use standard or customized E-Mail notifications to notify the destination users when a task is assigned to them. This E-Mail can be customized and can include variables from the process, results from SmartObject methods and can be formatted in both plain text and HTML format. In most cases, the notification E-Mail would also include a hyperlink to the task form associated with the task so that the user can click on a link in the email to open the task form and complete the task.

The notification screen is also used to indicate whether the user will be able to action the task directly via E-Mail just by replying to the E-Mail with their decision. This feature is known as K2 SmartActions and is described in more detail in later topics as well as the K2 product documentation, at **Usage > Worklist > K2 SmartActions > K2 SmartActions > K2 SmartActions Design Time.**

Configuring the notification E-Mail message for a Client Event

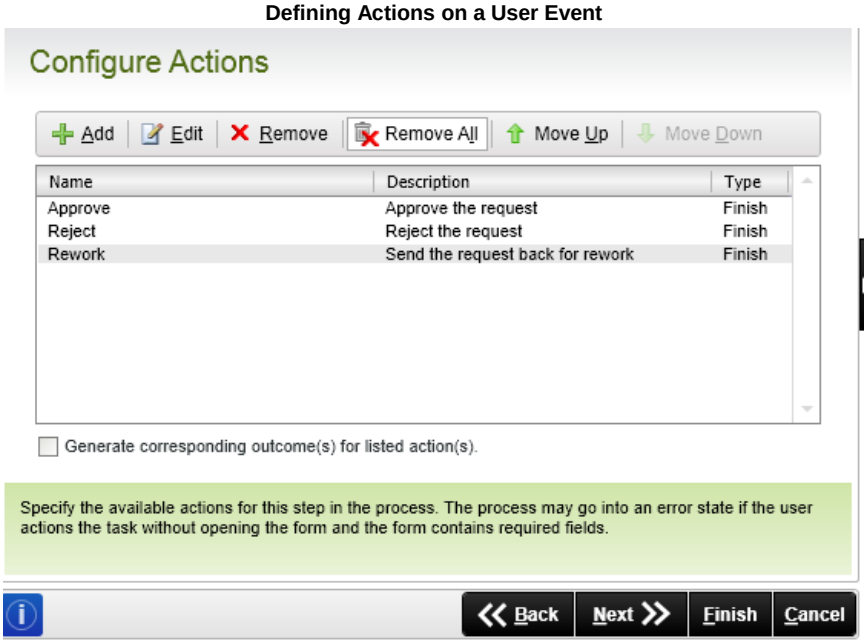


Actions

Actions define the possible decisions that users can take on a task. A user always has to take an Action on a task before the task can be completed, and is most cases the workflow will follow a particular path based on the action selected by the user. Note that K2 allows multiple actions to be defined on a task and the Actions can be anything you wish: you are not restricted to Approve/Reject actions.

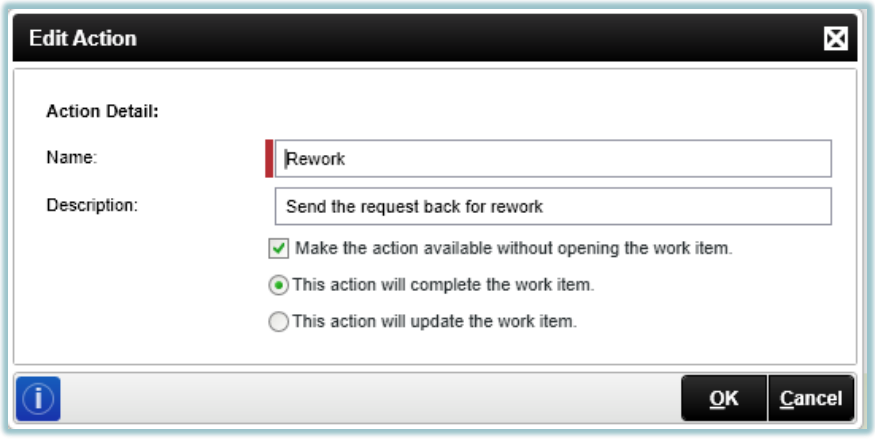
At runtime, the available actions are typically displayed in a drop-down list on the form for the task. You may also define additional instructions that can describe the task to the user.

Defining Actions on a User Event



In the screenshot above, notice the option to **Generate corresponding outcome(s) for listed actions**. This is normally selected so that the K2 designer will generate the resulting Outcomes for the activity. You may customize the generated outcomes to define more advanced rules for the Outcomes, of course.

Editing an Action



When defining an Action, notice the available options for the Action.
Make the action available without opening the work item is used to tell K2 that the user can perform the action without having to open the page associated with the task. You must select this option if the user will use K2 SmartActions to complete the task by E-Mail or if the user will just select an Action from the K2 worklist web part to complete the task. If the user must always use a form to complete the task, disable this option.
The radio button options **This action will complete the work item** and **This action will update the work item** will tell K2 whether the task should be completed if the user selects the action. In most cases, the default option to **complete** the work item is the right choice. The second option is mostly used is you want to allow the user to Save their changes to the form or data without completing the task item and letting the process continue.

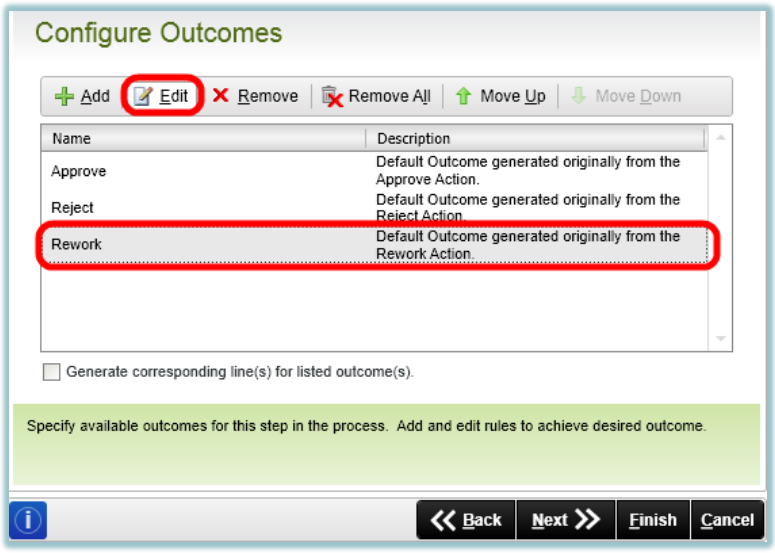
Outcomes

Outcomes define the possible results of the action selected by the user. In most cases, actions have a 1:1 mapping to outcomes: if the user selects the “Approve” outcome, the “Approve” outcome will be followed. The **Generate corresponding outcome(s) for listed actions** option on the Client Event wizard will automatically create corresponding outcomes for each action and set the outcome rule for each action. In most cases, this is all that is required and the designer does not need to make additional changes to the outcomes.

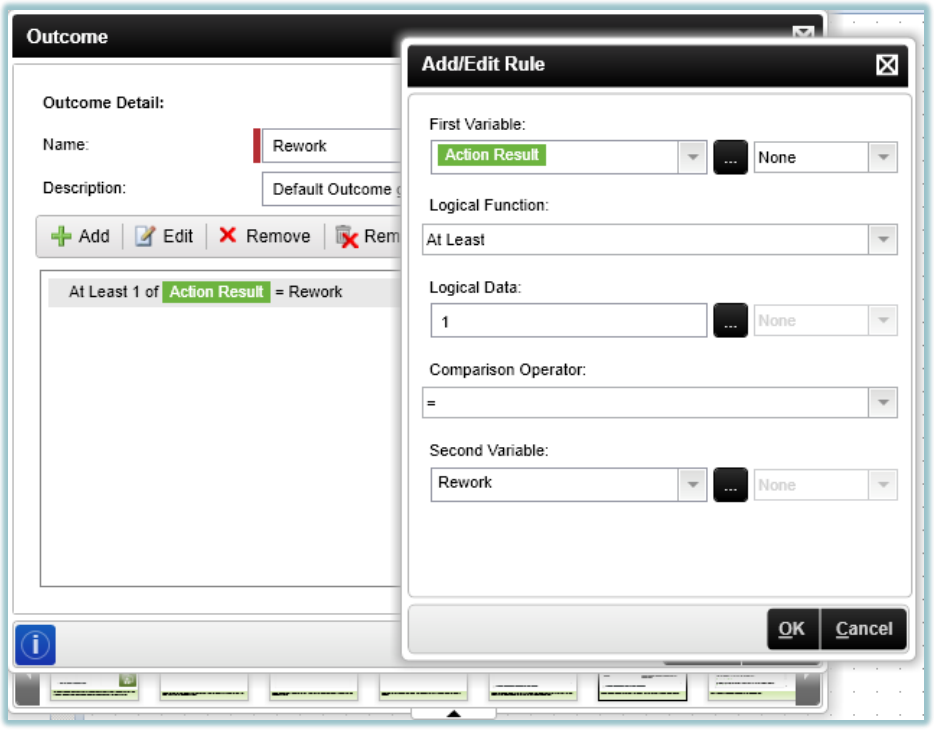
K2 can also automatically generate Lines for the Outcomes of an Activity. When the **Generate corresponding Line(s) for listed outcome(s)** (see screenshot below) option is selected, K2 will generate lines that flow out of the Activity, and you can then drag the line endpoints to whatever activities should follow from each of the possible Outcomes.

In situations where more advanced outcomes are required (for example: at least 2 approvers must approve the request), the workflow designer may manually define outcomes or edit the generated outcomes to perform some additional checks. To modify an Outcome, select the outcome in the list of available Outcomes and then click the Edit button to edit the Outcome. In the Outcome Rule editor you can then define whatever rule is required for the Outcome.

Selecting an Outcome to edit



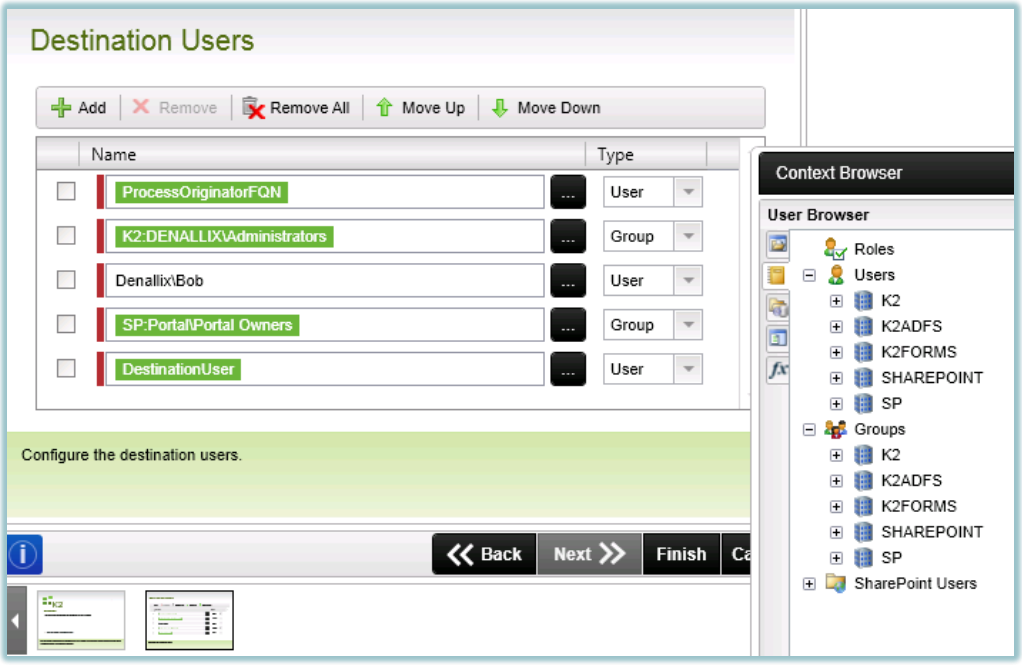
Editing the Outcome Rule



Destination Users

The last major component of any Client Event is also one of the most important: the users that the task will be assigned to, also known as Destination Users. This setting tells K2 who the human task should be assigned to, and K2 will then take care of the task allocation accordingly. There are several options for specifying destination users, and we will discuss more advanced approaches in later learning modules. For now, just know that you could define Destination Users statically by specifying Users and Groups from Active Directory, SharePoint and custom security mechanisms as well as Roles defined within K2. You can also dynamically determine the Destination Users by returning a username, group name or collection of usernames from a SmartObject method. Finally, you can also use Destination Sets to define different sets of destination users and the rules associated with each Destination Sets.

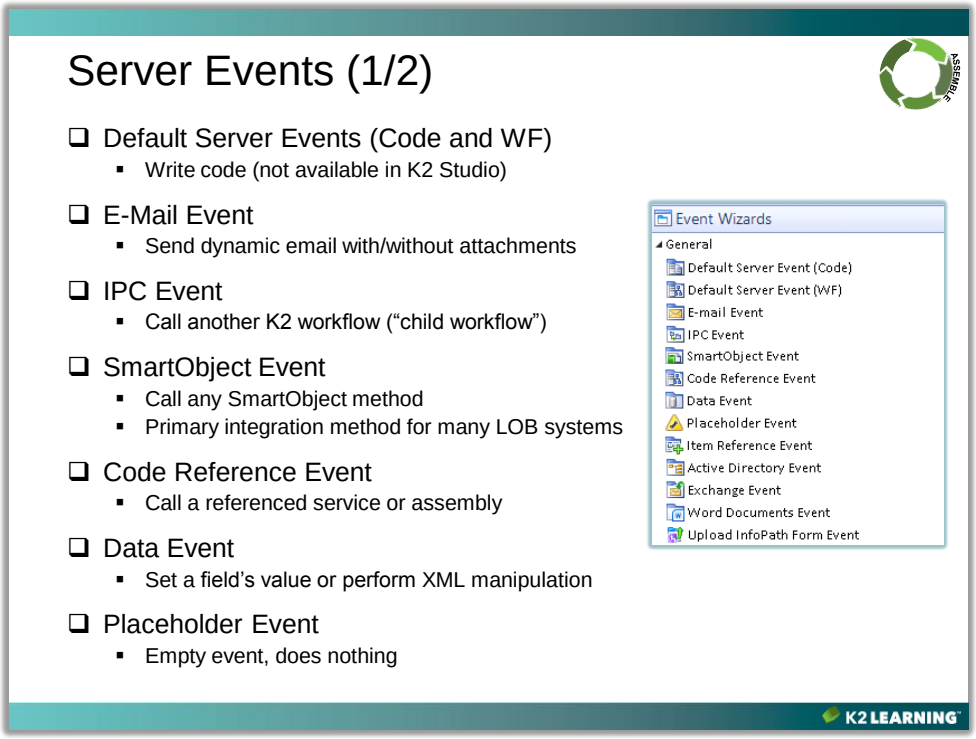
Setting Destination Users for an Activity with a Client Event

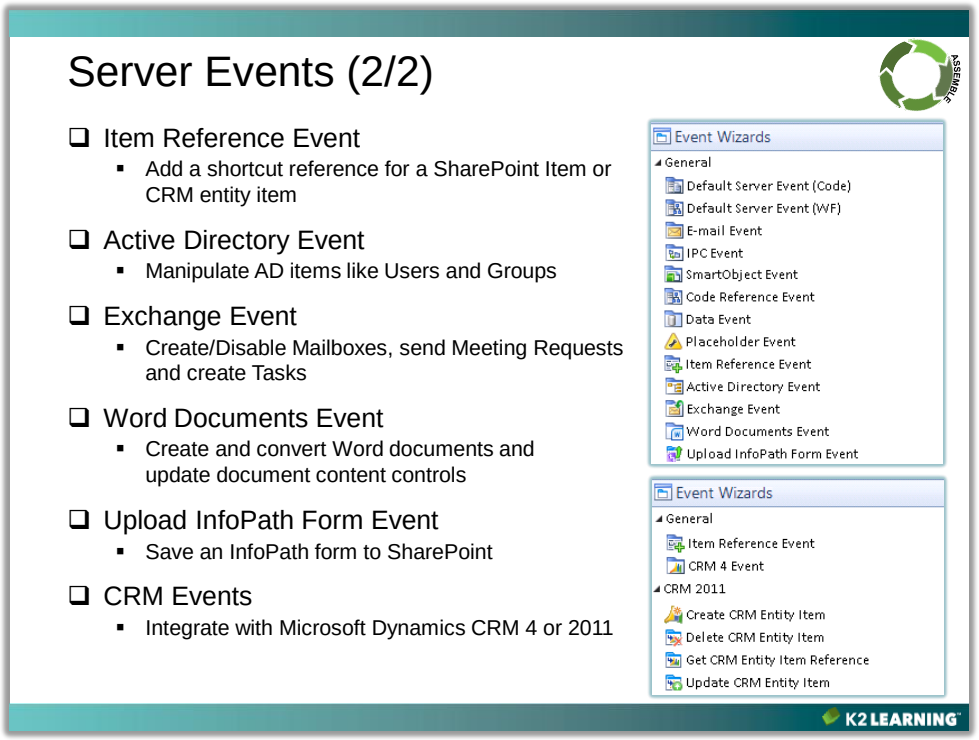


Advanced settings for destination rules are also available, such as indicating whether K2 should create a task for each user or one task for everyone, whether the task should be assigned to all the users at the same time or in sequence and whether to limit the number of tasks created. We will discuss these advanced approachs in the learning module **200.AUS Building Workflows with K2 Studio – Intermediate.**

01001011 00110010	The Destination Users property is actually set on the Activity level, not the Event level. This allows you to perform some more advanced processing in a workflow, such as repeating the same event for each user that the task is assigned to. You can also use Destination Users on sever-only events to implement an approach where K2 repeats a set of server-side events for each value (not necessarily a user) in some collection. We will discuss this approach in later learning modules.

Server Events





There are several wizards which make it easy to integrate with common enterprise systems such as Active Directory, Exchange and Microsoft Dynamics CRM. These wizards could be used to implement powerful and flexible processes that manage, for example, employee onboarding and off-boarding and CRM-centric processes.

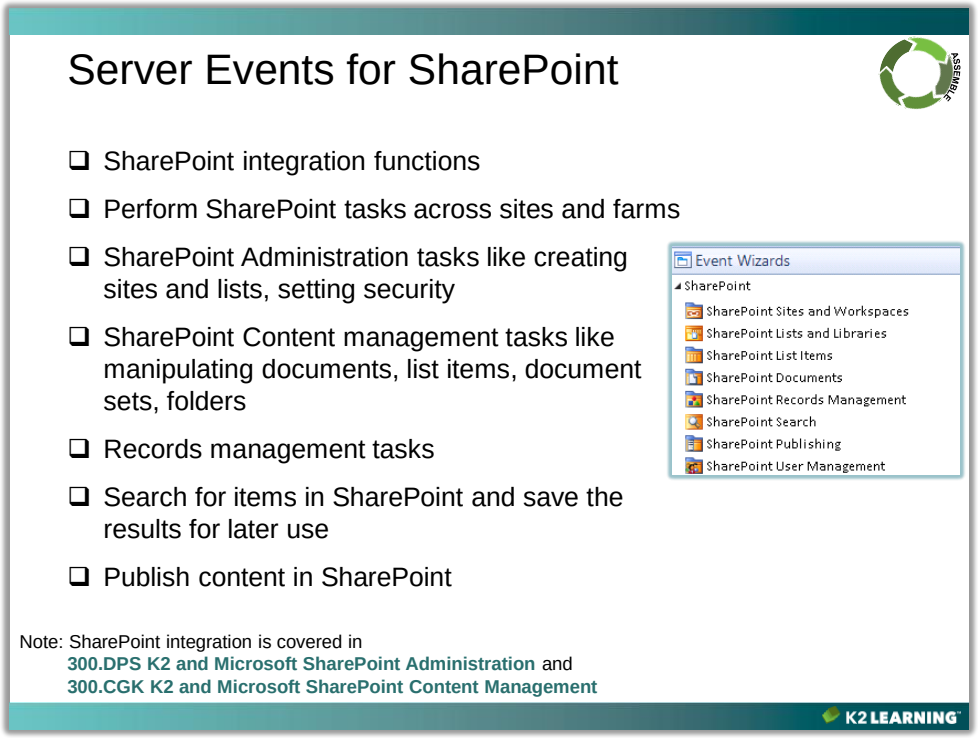
Depending on the installed version of K2, the available wizards may be different to the wizards described here. You can always refer to the product documentation to see which wizards are available for your installed version.

The table below briefly describes the available Server Event wizards and how they may be used and mostly serves as a reference that you can refer back to later. It is not necessary to cover the table in detail.

Event	Description
Default Server Events* (Code and WF)	Server Event for writing custom C# code. Developers can extend these events by writing custom Microsoft .NET C# code or Windows Workflow schedules using Visual Studio.
E-mail Event	Server event used to send a customizable e-mail. In the advanced mode, this wizard allows the designer to add attachments to the email being sent.
IPC Event	Server event used for inter-process communication (IPC) in parent-child or sub-process scenarios. This event creates and starts an instance of another process, and either waits for the sub-process to complete, or starts the sub-process and immediately continues with the parent process. As part of the wizard, you can also pass values back and forth between the Parent and Child processes.
SmartObject Event	Almost any system can be exposed as a K2 SmartObject and workflow designers can use the SmartObject event to interact with SmartObjects that are available in the K2 environment. This wizard allows workflow designers to interact with these systems without writing any code and without having to understand how to call a method in a referenced assembly.
Code Reference Event	Server event used to set properties, instantiate, and execute methods on a .NET class from any .NET assembly or service that is included as a referenced assembly in the process. Only simple types are currently supported. This wizard makes it possible to integrate with almost any system without needing to define a SmartObject for the external system, but it may require a developer to create the .NET assembly or service first
Data Event	This event is used to set data fields in the process, and can perform XML manipulation to transpose values from one XML field to another XML field.
Placeholder Event	The placeholder event is an empty event which does not perform any action. This step is useful when the process designer does not have the necessary detail to implement a specific step yet, when the step will be implemented in a later version of the process, when the designer wants to save the process design and let another user implement the step, or when an empty event is needed during a process, for example an empty End Process even
Item Reference Event	Used to add a “shortcut” reference to a SharePoint list item, SharePoint document or CRM entity. This will create a datafield in the process that will allow you to read values from the referenced item at runtime without needing to locate the item in the source system again. This wizard can also be used to add a reference to a collection of items (for example, a collection of documents created by a particular user). This collection can then be passed to another wizard (for example, the Convert document wizard) to repeat the same action (convert) on each document referenced in the collection.
Active Directory Event	Used to create, update, delete and edit Active Directory users and groups. These wizards are very useful in employee onboarding, transfer and off-boarding processes When using the Active Directory wizards, it may be necessary to configure additional security settings outside of K2 to allow these wizards to work. See the following topic in the K2 Product Documentation for more information `Concepts > Integration > Active Directory > Active Directory Event Wizard - Overview`
Exchange event	Used to create and disable exchange mailboxes, send meeting requests and add tasks to the Exchange task list. When using the Exchange wizards, it may be necessary to configure additional security settings outside of K2 to allow these wizards to work. See the following topic in the K2 Product Documentation for more information `Concepts > Integration > Exchange > Exchange Wizard - Overview`

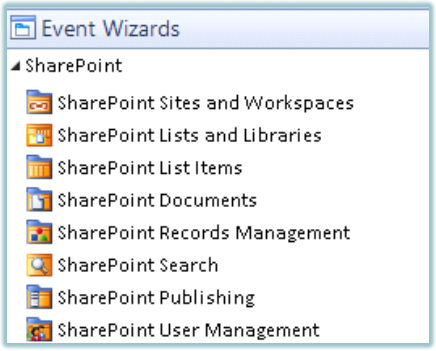
Event	Description
Word Documents Event	Used to convert existing word documents to a different format, create new documents, or update document content using content controls. The **Convert Documents** option leverages Word Automation Services in SharePoint 2010 to convert a Word document into another format. The supported formats include : PDF, XPS, Document, DocumentMacroEnabled, Document97, Template, RTF, Word XML and more This option is most commonly used to convert a Word document into a PDF or RTF document. As an example, during a customer contract process, users may change the content of a Word document. At some stage during the process, the Word document should be converted to a read-only PDF document and E-mailed to the customer. This wizard can perform the PDF conversion. The **Create Document** option is used to create a new Word document automatically. As part of this wizard, the content type for the new document must be specified. This wizard is commonly used together with the Update Document Content Control or Update Document steps, since the content of the new document may need to be modified. Document Content Controls are a useful feature of Microsoft Word documents that allow for the creation of updateable document templates. (See the Microsoft topic http://msdn.microsoft.com/en-us/library/bb386290.aspx for a more in-depth discussion on Content Controls). The **Update Document Content** option can be used to update content controls in a document as part of a process. For example, during a customer onboarding process, a welcome letter should be generated for a new customer. The welcome letter may be based on a specific template, and content controls are used to populate the customer address and salutation at runtime. As part of a K2 process, this wizard could be used to update the content control using a variable or a value returned from a context field before the document is E-mailed to the new customer.
Upload InfoPath Form Event	This step is used to upload the current InfoPath form for the workflow to a specified SharePoint form library. Remember that, when InfoPath forms are K2-enabled and submitted to a workflow, K2 stores the InfoPath form internally, not in a SharePoint form library. You can use this wizard if you wanted to save a copy of the InfoPath form in SharePoint so that users could access the form. A good example would be to save a “snapshot” of the InfoPath form once the process is completed into a library where users only have Read access. This “snapshot” can act as the record of the approval process and the data that that was approved and captured during the workflow
CRM Event	Used to create, delete and update Microsoft Dynamics CRM entities (requires the CRM SDK 4.0). An “Entity Item” in CRM just means a record in CRM. For example, an Account is an entity item; an Order is another entity item; and a Product is an entity item. These wizards allow process designers to interact with many entities that exist in CRM. See the K2 product documentation for a list of the supported CRM entities and required security configuration to allow these wizards to work: `Concepts > Integration > CRM > CRM Requirements and Supported Entities`

Server Events for SharePoint



K2 provides several wizards out-of-the-box that allow process designers to interact with SharePoint artifacts and services, just by configuring steps in a workflow using a wizard-based interface. Typical uses of these wizards include updating or creating list items during a process, moving or deleting a document during a process, creating new sites and lists in SharePoint and modifying SharePoint security. The screenshot below shows the available SharePoint wizards (as of K2 4.6.5)

SharePoint wizards in K2 Studio



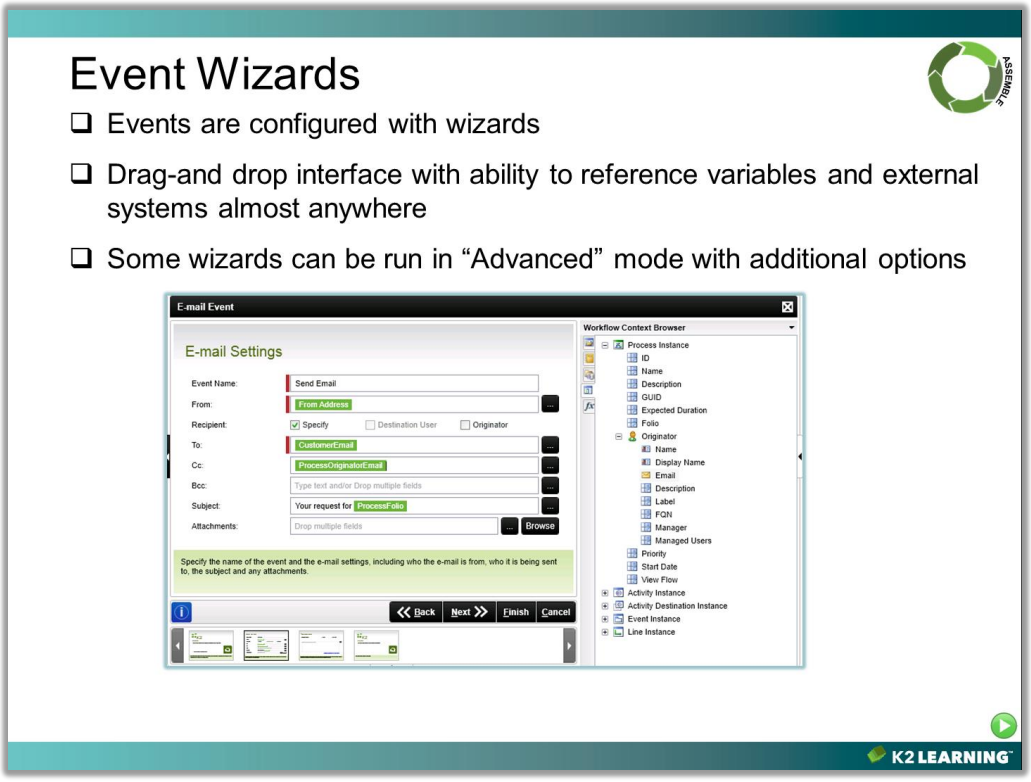
The table below briefly describes each wizard and some additional information about the Events. It is not necessary to cover this table in detail, the purpose here is just to give you a point of reference to return to when you need to know which wizard should be used how to perform a particular SharePoint task in a workflow.

Event Wizard	Description
SharePoint Sites and Workspaces	Create, update, or delete SharePoint sites and workspace across any SharePoint site collection, web application, or farm. The wizard can accept dynamic values for the target site and template to use, but it does rely on the specified template already being deployed in the target SharePoint environment. *Example: as part of a project management process, create a new team site in SharePoint based on the standard Team Site template*

Event Wizard	Description
SharePoint Lists and Libraries	Create, update, or delete SharePoint lists and libraries across any site collection, web application, or farm. *Example: as part of a contract collaboration process, create a new document library in a SharePoint site so that all documents relating to the process can be stored in the same document library.*
SharePoint List Items	Create, update, copy, or delete SharePoint list items across any list in any site collection, web application, or farm. *Example: as part of an Announcement approval process, create a new list item in a SharePoint Announcements list using dynamic data fields for the item's properties.*
SharePoint Documents	Upload, download, copy, move, update, check in, check out, undo check out, or delete SharePoint documents across any library in any site collection, web application, or farm. *Example: as part of a document approval process, move a document from one SharePoint library to another library on a different site once it has been approved.* Notes: ▪ The **Move** operation is actually a Copy-And-Delete operation, so the new document will not retain the version history of the source document. ▪ The **Check-In** and **Check-Out** options can be used to "lock" a document for editing during a workflow. While a document is checked-out, no other users can make any changes to a document until it is checked-in ▪ When using the **Delete** option, the document does NOT get moved to the SharePoint Recycle Bin. If the delete document step is executed, the document is deleted for good. ▪ The **Update** options can update metadata for a specific document. The available metadata fields that can be updated are based on the columns defined in the SharePoint document library of the target document. Custom columns can also be updated using this wizard. ▪ You can use this wizard to create a Document Set. As part of the **Create Document Set** option, the document set content type can be selected, if document set Content Types have been defined in the SharePoint environment. This wizard is useful in ECM scenarios where documents are grouped and categorized using Document Sets. For example, as part of a Sales Proposal process, all documentation generated for the proposal (such as Word Documents, Spreadsheets, PowerPoint slides) should be grouped together into a Sales Proposal content type, which inherits from the Document Set content type. Process designers can use the wizard to create the document set in a library, associate the document set with the Sales Proposal content type and then set metadata (for example, the prospect name or sales manager) for the document set. Once the document set has been created, the Move Document" and Copy Document wizards can be used to move documents into the new document set
SharePoint Records Management	Manage records, or manage holds within SharePoint Records Center across any site collection, web application, or farm. *Example: as part of a customer investigation process, declare the customer's contract document as a record so that it cannot be modified or deleted while the process is active.*
SharePoint Search	Used to run a search in SharePoint across multiple lists or libraries across any site collection, web application, or farm. The search results may be stored and used throughout the process. The results of this wizard are commonly used in conjunction with another wizard. *Example: as part of a Freedom-Of-Information process, search for all documents relating to a specific topic in a specific library, and save the resulting list of items in a process field. In a separate event, use the list of returned values from the search wizard in a SharePoint document wizard to download and attach the documents to an email which is sent to the requestor.*
SharePoint Publishing	Server event used to manage SharePoint publishing, including creating a publishing page across any site collection, web application, or farm. *Example: as part of a Press Release approval process, create new content in a publishing page which is publicly available.*

Event Wizard	Description
SharePoint User Management	Server event used to manage membership, manage groups, and manage permissions in SharePoint at a granular level across any site collection, web application, or farm. You could create groups, modify group members and set permissions on almost any SharePoint entity using this wizard. *Example: as part of a document approval process, once a document is approved give all users in the organization Read-level access to the document.*

Event Wizards



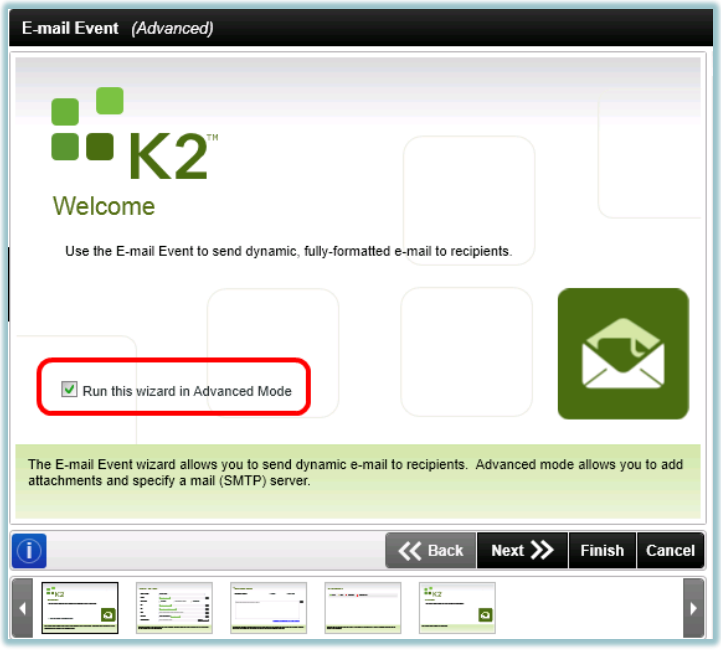
The event wizards provided by K2 help designers to assemble workflows with ease. When you drag and drop an event into the design canvas, a wizard will launch and you will configure the wizard by dragging and dropping values into the wizard fields, or by typing values in. Designers can even drag and drop contextual fields ("variables" into the wizard to build a very flexible and dynamic wizard.

You can always re-run an event wizard top reconfigure or change the settings in the event, and you can copy and paste Activities and Events in the designer as well.

Launching the event wizard again

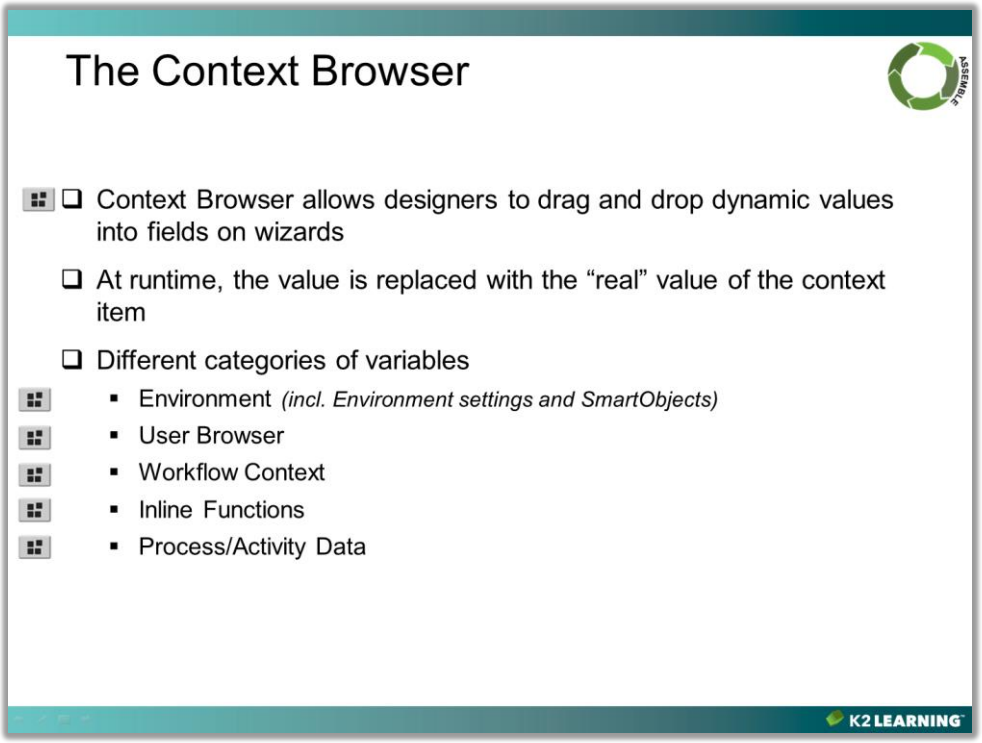
Note that some wizards allow you to run the wizard in "Advanced" mode, which exposes additional settings in the wizard. To run in advanced mode, navigate to the first screen of the wizard with the navigation bar and check the advanced mode checkbox, as shown below.

Running a wizard in advanced mode



01001011 00110010	To prevent the wizard from popping up when you drop an event into the design canvas, hold down the <CTRL> key when you drop the event. This way, you can quickly lay out your workflow. You will not be able to deploy the workflow until you have configured all the events, however.

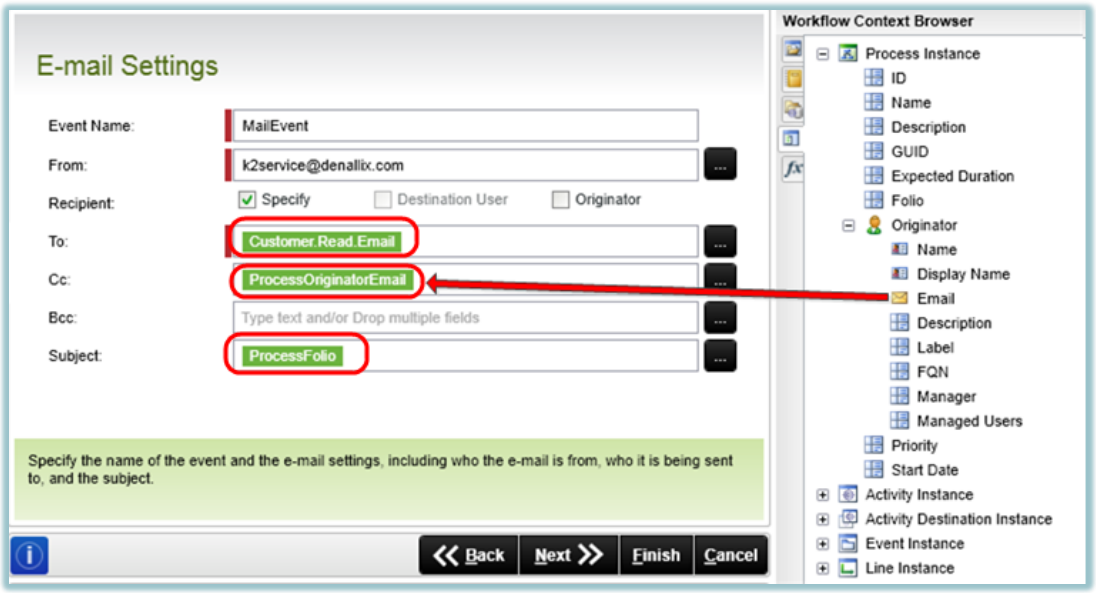
The Context Browser



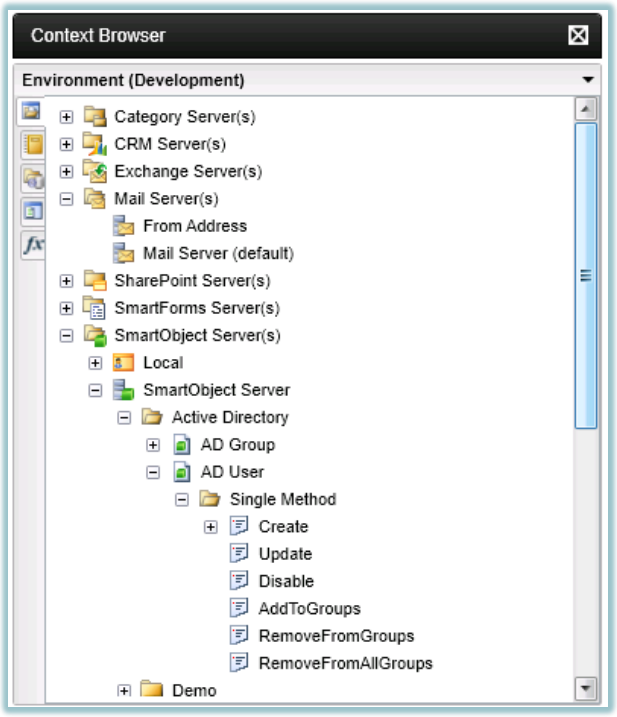
The K2 Context Browser plays a vital role during the process assembly phase because it allows process designers to select variables when designing a process. At runtime, the K2 server will replace those variables with the evaluated (or real) values when the process is executed.

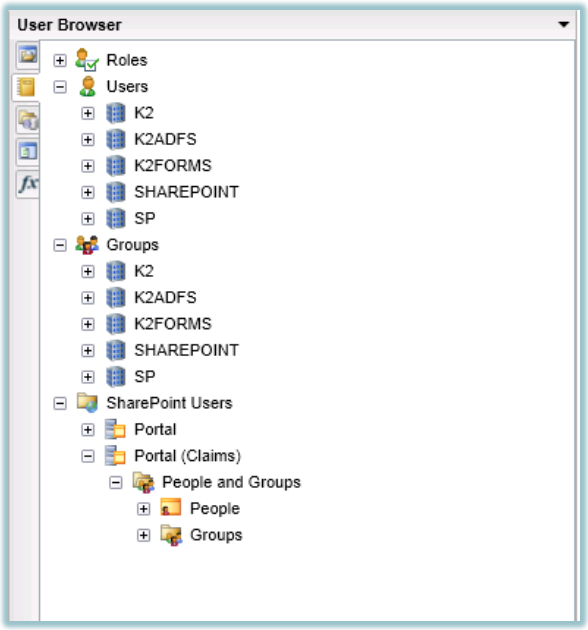
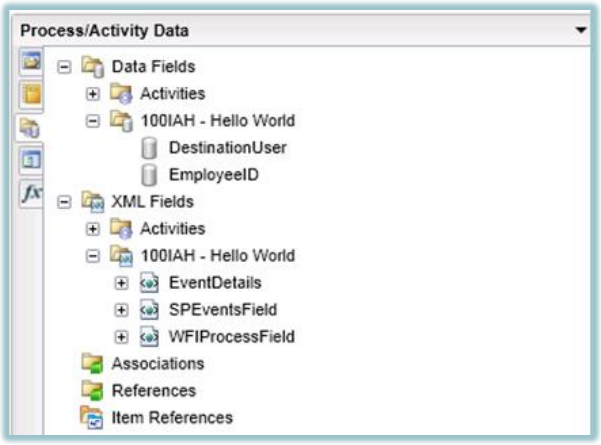
Consider the example below. Here, the E-Mail event wizard is being configured, and the designer has dragged values from the Context Browser into the E-Mail wizard. The context fields are highlighted in green, indicating that they are variables that K2 will replace at runtime. In this example, the workflow designer has dragged values from the context of the workflow (The Originator's Email and the Process Folio) as well as the result of a SmartObject method (the customer's E-Mail address in the "To:" line).

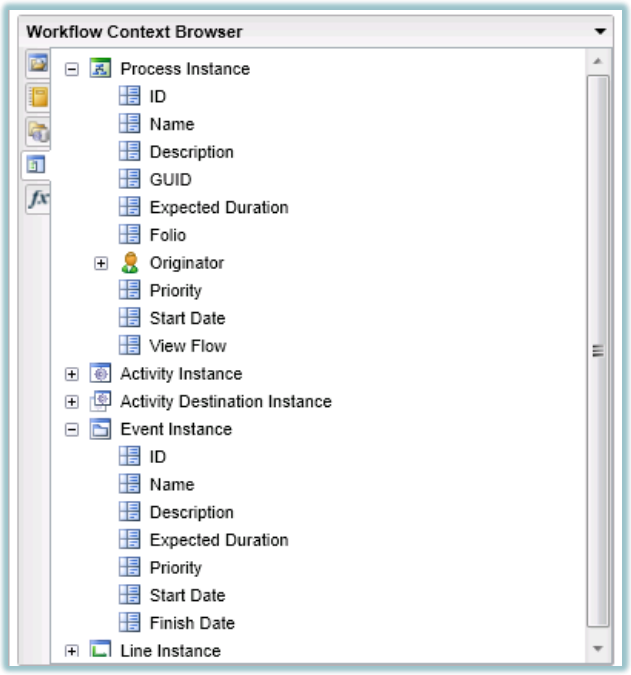
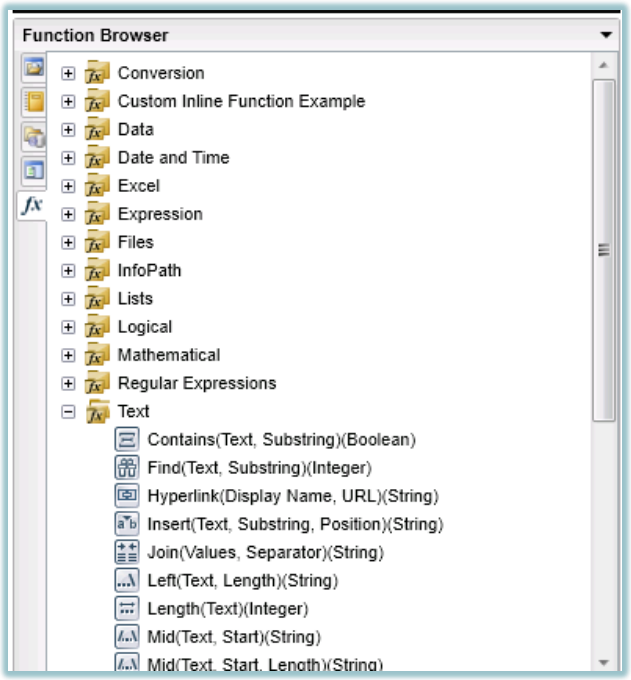
Populating a wizard by dragging values from the context browser



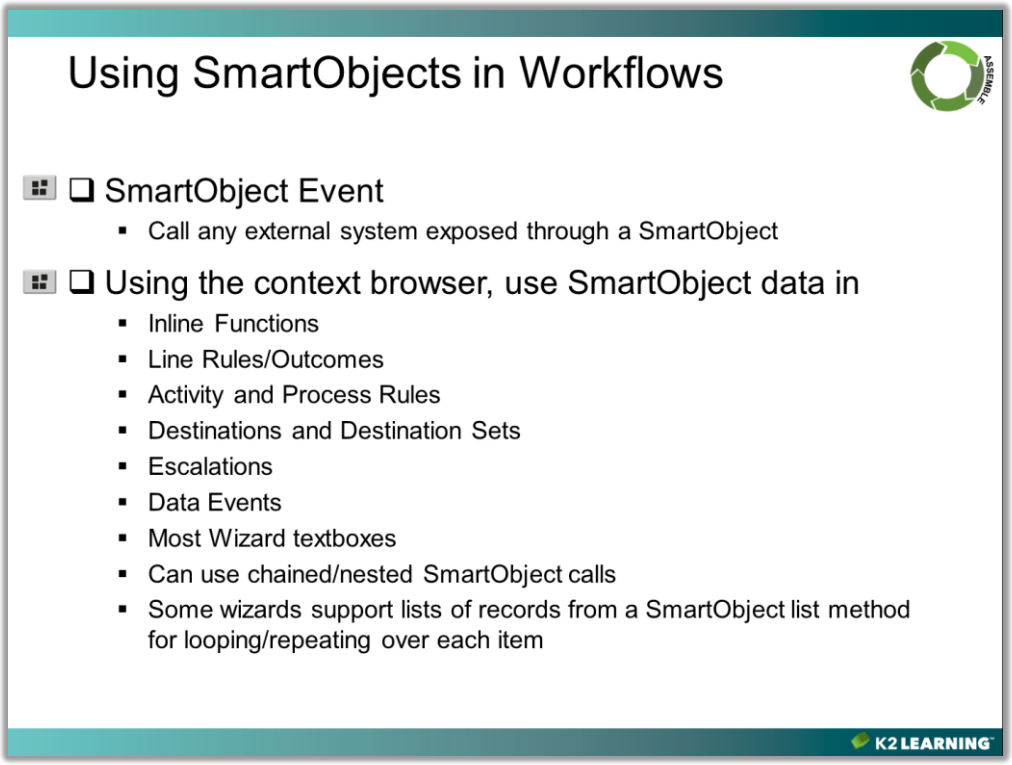
The real power of the context browser lies in the different categories of context values that are available. Let's look at the available categories. (Notice the tabs on the left-hand side of each screenshot: the different categories are accessed by clicking on the tab. Once the category has opened, you may browse for the value you want to use by browsing the tree structure shown on the right)

Category	Values
Environment 	This category exposes server-level configuration settings, both standard settings (such as the SharePoint server URL) as well as custom settings. The ability to add custom settings is especially important when you want to deploy solutions across different environments. Here is an example: suppose that each of the environments at the organization (Development, Test and Production) have a different address that should be used when sending E-Mails as part of a workflow. If users receive an E-Mail from "testworkflow@denallix.com", they know that this is an E-Mail form the test environment. When the workflow is deployed to Development, Test or Production, it would be tedious to replace the "From" email address for each E-Mail event in the process before it is deployed. A better approach would be to define a custom field for the Environment called "FromEmailAddress". The value of this field would be different for each environment, and the workflow designer just has to drag the "FromEmailAddress" context field into the E-Mail wizard. When the workflow is executing in Development, K2 will use the development email, and when the workflow is executing in Production, K2 would use the production email. Fields from the Environment Library can be used in K2 workflows as well as K2 smartforms. Notice the **SmartObject Server** folder in the context browser. This node will allow the designer to browse the available SmartObjects on both the K2 server as well as any SmartObjects in the current project, so that they can drag a property or method from the browser into a wizard textbox.

Category	Values
User 	The User category allows the designer to browse users and groups from the current security providers (most commonly, Active Directory and SharePoint) as well as any Roles defined on the K2 server. These values are most often used in Destination Rules, but as with all values from the Context Browser, could be used in almost any wizard textbox.
Process/Activity Data 	The Process/Activity data category exposes any data fields or XML fields defined in the process or the activities in the process. Additionally, it will expose any SmartObject Associations defined for the process. The References node allows the workflow designer to define a “shortcut” reference to any SmartObject method. This is especially useful if the same SmartObject method is used several times in the workflow.

Category	Values
Workflow Context 	The workflow Context category exposes various properties of the current process, for example the process folio, start date, the originator of the process and several of the originator's properties as well as context-sensitive items such as the current activity's start date. This category is very often used to obtain the originator of the process so that a task or email can be sent to that person. Note that the context-sensitive values may not always be available: in an activity, for example, none of the Line Instance values can be used because the Line is not in context when the Activity is being run.
Inline Functions 	The final and most extensive category in the Context Browser is the Inline Functions category. K2 provides many functions that can be used in wizard textboxes as well as SmartObject methods to perform processing on data before or after it is used. The available functions are divided into different types depending on the function being performed, such as Date/Time processing, Mathematical functions or Text functions. In K2 blackpearl, developers can even create and register custom functions to extend the list of available functions.  The Conversion function is especially useful to convert data types. Suppose that a particular wizard expects a string value, but the data field in the process was defined as an Integer. Using the conversion functions, you can convert the data field from integrer to string and then pass the result into the wizard. Easy!

Using SmartObjects in Workflows



SmartObjects allow K2 to integrate with almost any other technology and expose that technology as a business object that can be used by a Workflow, Form or Report. They are probably one of the most important features of the K2 platform and allow you to create very flexible, powerful and dynamic applications.

Effectively, SmartObjects act as a layer between consumers (for example K2 processes, Forms and Reports) and some underlying back-end system that stores the information. Those underlying systems could be based on almost any technology – as long as there is a connector for the target system, it can be exposed through SmartObjects.

The SmartObjects layer can also act as a central repository for business objects to allow solution designers to find all the available business objects in one place. Designers can use the same tools and wizards to interact with those business objects, and they don't need to know anything about the differences between say SQL and SharePoint to be able to use the data in either system.

From a workflow designer's perspective, SmartObjects allows them to integrate with external systems using visual wizards without needing to know anything about the external system's interfaces or APIs.

Note that SmartObject communication is two-way, meaning it is possible to read and update data in the underlying system using SmartObjects. SmartObjects support Create, Read, Update and Delete-type methods (commonly referred to as CRUD methods), and in addition support Execute (perform a specific action, for example "Disable Account") and List (list a collection of objects, for example "Get a list of employees") methods as well.

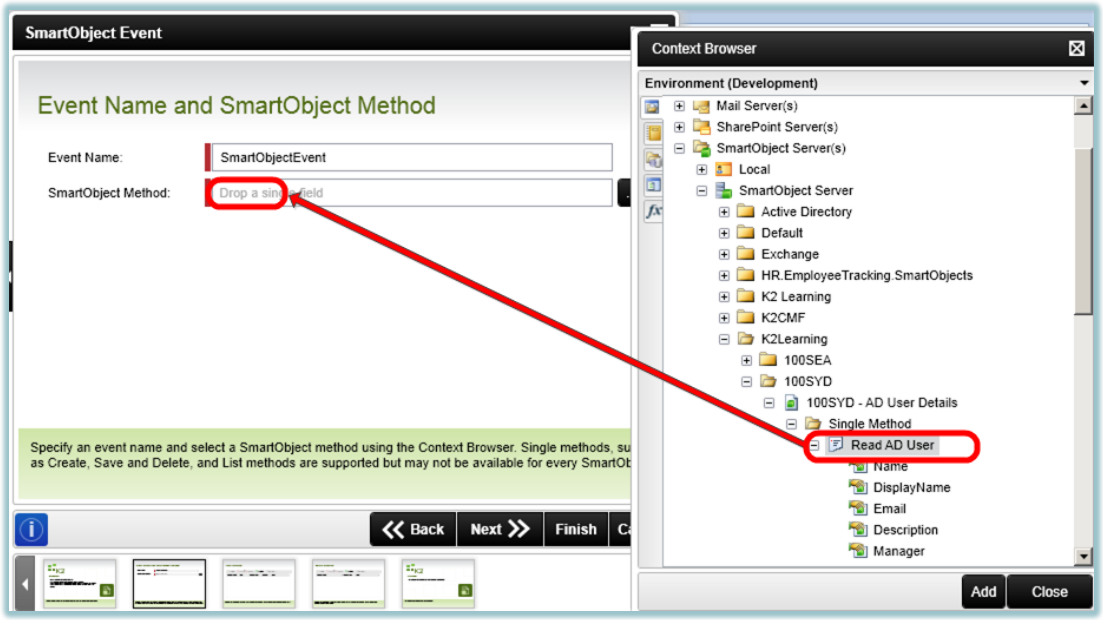
K2 design tools and utilities allow designers to create SmartObjects, but it may be necessary for IT resources to create SmartObjects (depending on the complexity of the underlying system.) The point is, once these SmartObjects are created, they can be used and re-used by various technologies and different projects.

SmartObjects are explained in more detail in the following modules:
100.SYD K2 SmartObjects – Fundamentals
200.PER K2 SmartObjects – Intermediate

The SmartObject Event Wizard

When using the SmartObject Event wizard, designers can drag and drop a method from any available SmartObject into the wizard. The wizard will then discover the parameters and input properties for the method and allow the designer to drag variables or type values for the parameters or methods. If the SmartObject returns values, the designer can save these values into process datafields or pass them into Inline Functions or other SmartObject methods.

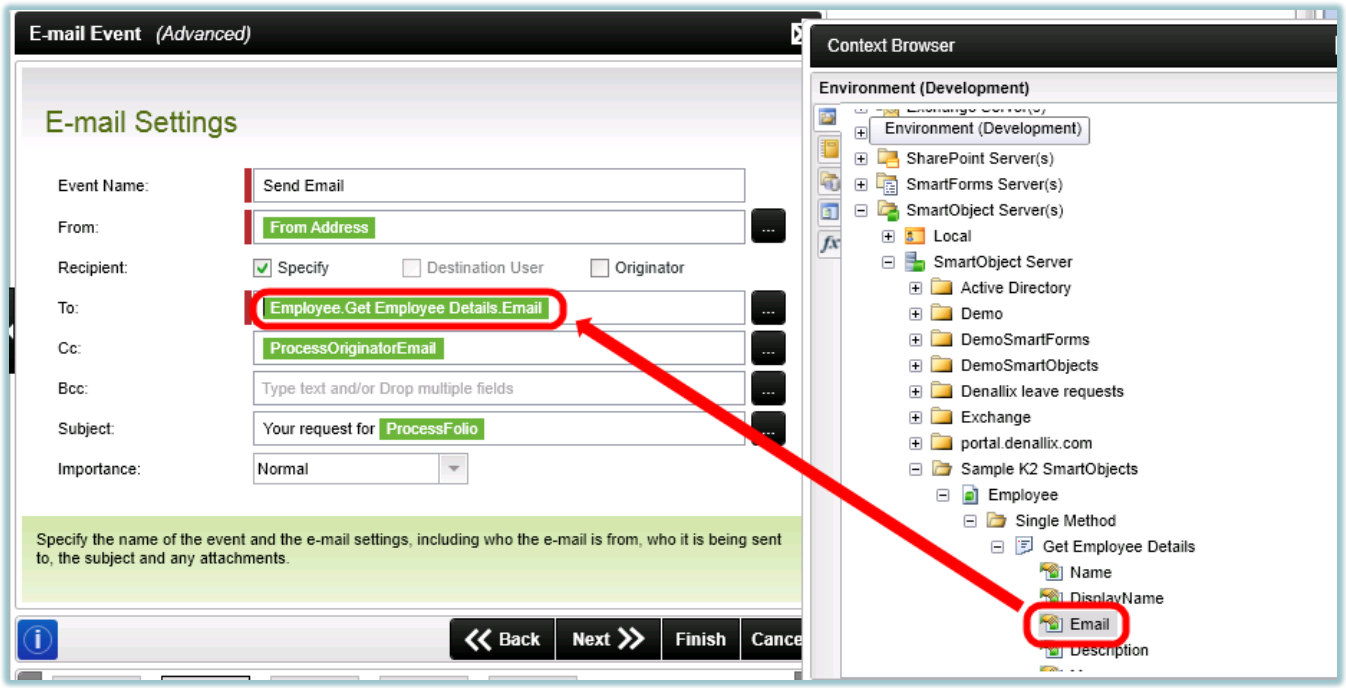
Dragging a SmartObject method into the SmartObject Event wizard



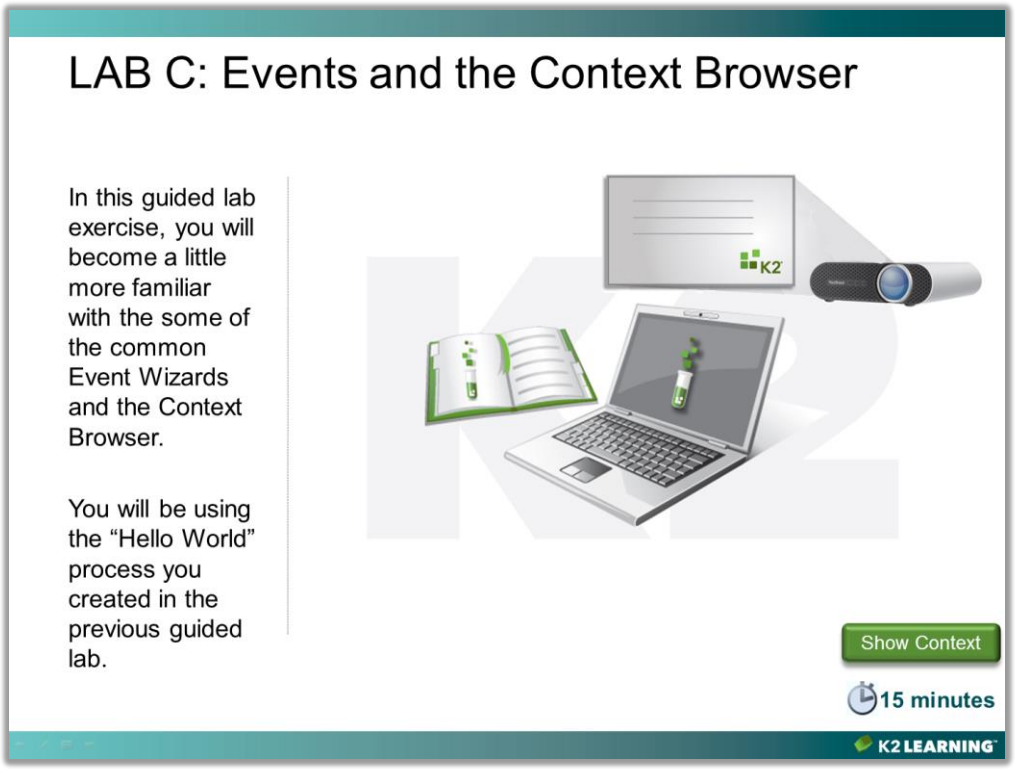
Using SmartObjects from the context browser

Apart from the SmartObject event wizard, workflow designers can drag properties from a SmartObject method into almost any textbox in any wizard by using the context browser. Designers can also use properties from SmartObject methods in rules, in Inline Functions and can even nest multiple SmartObject method calls to arrive at some value or perform some task.

Using the context browser to drag a SmartObject property into a wizard



LAB C: Events and the context browser



Objective

In this exercise you will learn how to add and configure Events in a process. You will configure both Client Events (User Tasks) and Server Events (Server Tasks).

Duration

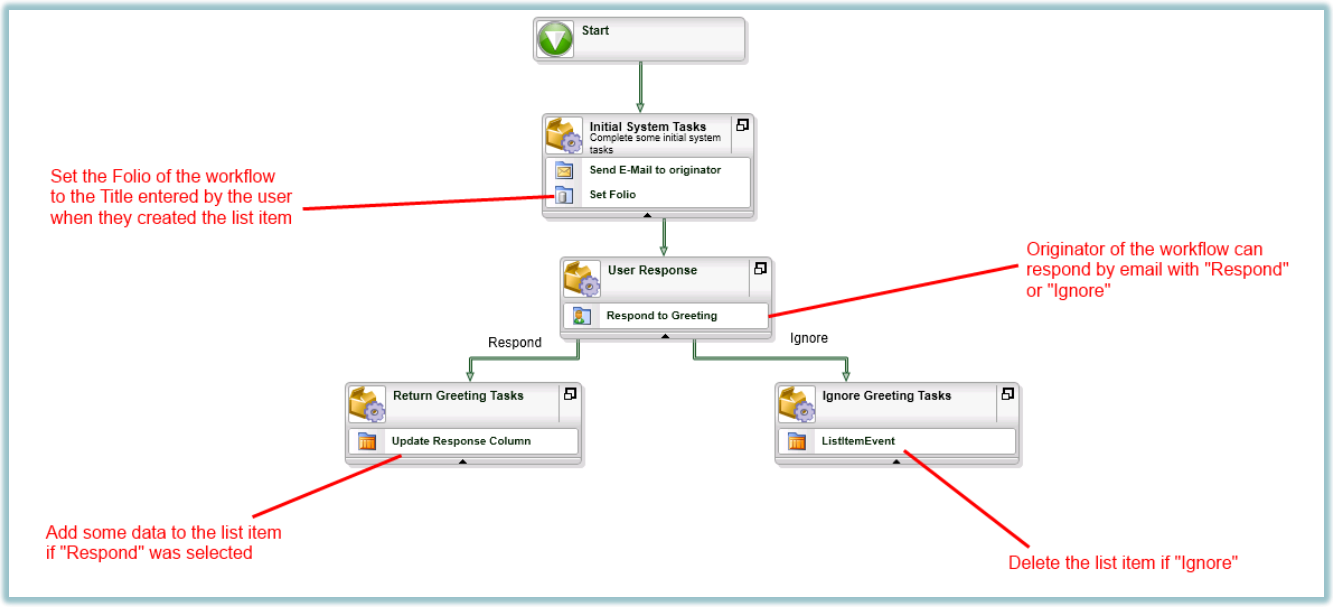
This lab should take around 15 minutes to complete. Although a guide to complete this guided lab is provided, you should preferably follow along with the instructor to complete this exercise rather than reading the guide.

Context

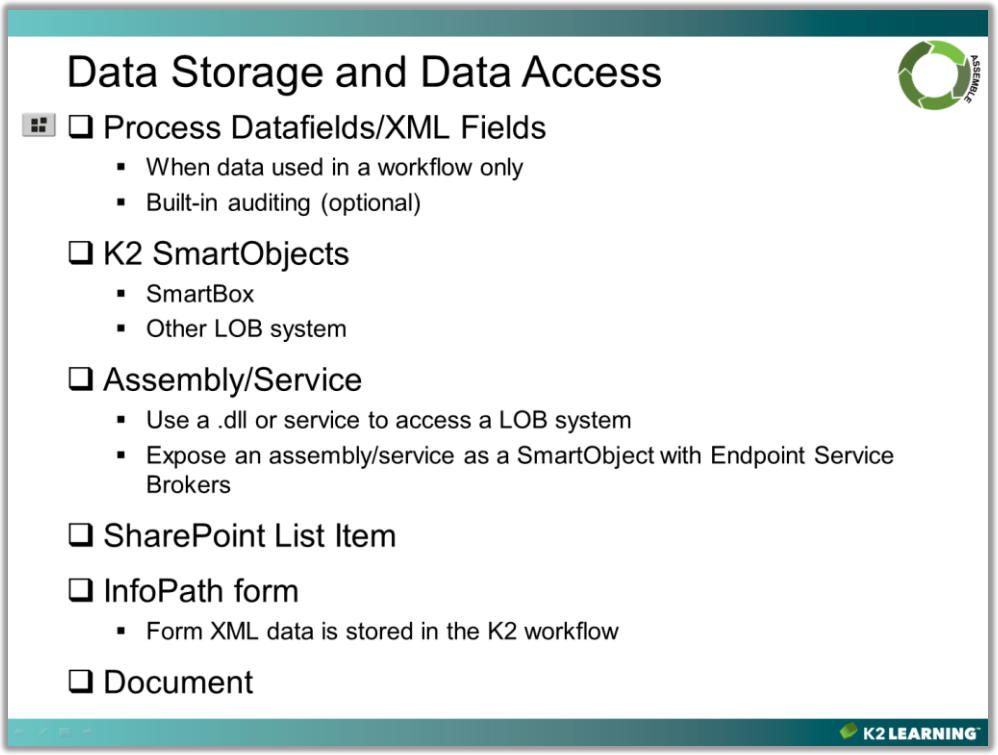
You will extend the existing “Hello, World” process that you created in the previous exercise by adding Events to the blank Activities. What we want to the process to do is to deliver a task to the user who started the process and give them a choice whether to respond to the greeting crated in the SharePoint list, or ignore the greeting. If they decide to respond to the greeting, the workflow should update the Response column in SharePoint. If they decide to ignore the greeting, the workflow should delete the list item in SharePoint. We won’t be using a specific form to capture the user’s decision; instead, we will allow them to make the decision via E-mail with K2 SmartActions.

We will also add and event to set the reference or "Folio" of the process instance to the Title of the list item entered by the user. This way, we can easily associate the workflow instances with the list items in SharePoint that started the workflow.

The screenshot below shows the desired behavior:



Data Storage and Data Access



There are a few common approaches used for storing data in K2 solutions, and each has its own considerations when it comes to ease of use, features and which forms/user interfaces that data can be consumed on.

When designing the data and data storage for workflows, you should consider the scope of the data as well as the available data storage options to select the best approach. Note that it is possible to mix data sources in the same solution; for example if there is a datafield that is only relevant to the workflow and not required in a form or in a report, it is easier and quicker to define the datafield in the workflow rather than in some external system. The rest of the data could still come from the external data source.

Scope of Data

A K2 application may require data for different purposes:

1. **Data for users**
   This refers to data that must be provided by users (humans) in the workflow or data required for the users to interact with the workflow. Most workflow tasks will need to provide some additional information to the user so that they can make the appropriate decision in the workflow. This data is usually displayed on a form or included in a notification email.

2. **Data for the K2 process**
   This refers to data that is required only in the K2 process. Perhaps the workflow must make a decision on routing the task based on the value of some data. This data may never be displayed to a user, but it is still required by the workflow to make the routing decision.

3. **Data for Reporting**
   This aspect is often overlooked during the workflow design phase. Some additional data may be required so that users can report on the workflow. While the data is not required in the process or on a form, at some stage the workflow may have to update this reporting system with data so that reports are accurate.

Data Storage options

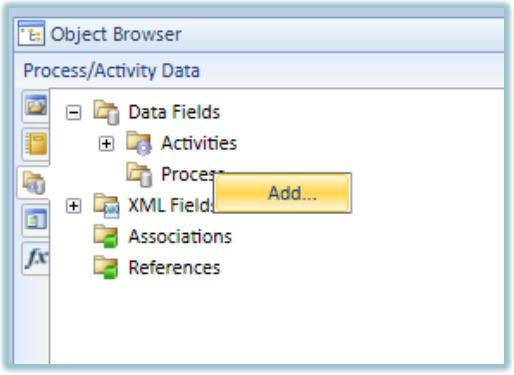
K2 process-level datafield/xml field

With this approach, the data is declared and stored inside the workflow itself. K2 supports two different kinds of data fields: Data Fields, which are single-value fields of different types (such as numbers, dates, strings and binary values) and XML Fields, which can store XML structures (for example, when an InfoPath form is associated with a process, the form data is saved as an XML data field).

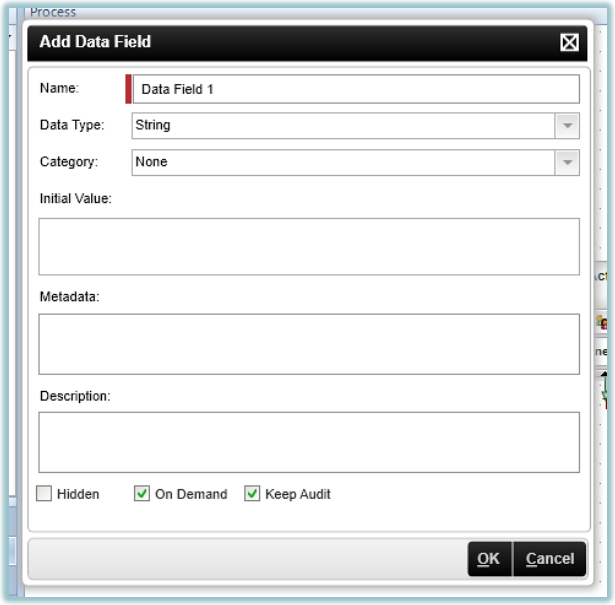
Process-level datafields and XML fields are most often used for data that is only relevant for the workflow. They are the easiest to work with in the workflow wizards, and all the K2 forms technologies can easily interact with these datafields.

The workflow designer can define fields or XML structures to store data in the workflow. See the screenshots below that illustrate how to declare datafields in a K2 process

Adding a datafield using the context browser



Setting a datafield's properties



The biggest advantage of this approach is that the data fields are managed by K2 and it is easy to use the data in K2 workflows and user interface, and you can optionally leverage the built-in data audit capability provided by K2 for these fields (as long as the "Keep Audit" option was selected). The biggest disadvantage of this approach is that the data is not easily accessible from other consumers outside of K2: you would have to create SmartObjects that expose the workflow and then consume these SmartObjects from the external application.

SmartBox-based SmartObject

With this approach, you will use K2 tools to define and deploy SmartObjects that use the K2 SmartBox database as the data store. This approach at makes the data a little more accessible from other applications, and the data is still easily usable in K2 workflow. The drawback of this approach is that you will need to use SmartObject events whenever you need to interact with the data in the workflow, and SmartObject access not supported in all the available workflow Form technologies.

External system exposed as a SmartObject

This approach requires a designer to create SmartObjects that wrap some external system. If the external system is supported by K2 out-of-the-box (e.g. SQL database) this is a simple process, but if it is not, creating an adapter is an advanced developer task. The advantage of this approach is that existing data stores can be consumed by workflows and certain K2 forms, but the drawback is increased complexity, additional development and the additional SmartObject event steps required in the workflow to interact with the data.

Remember that the external system could be almost anything: SharePoint lists and libraries, a table in a SQL database, a table in some other system and so on.

External System accessed through an assembly/service
If your organization already has assemblies and services that expose external systems, these can be added as references to a K2 process and used in a workflow though the Code Reference Event. This is a slightly more technical approach, may not be easily usable by workflow designers and can add complexity and troubleshooting overhead to a solution. In addition, no K2-generated interfaces will be able to support external references without customization and development work.

SharePoint List Item
Data can be stored in the SharePoint item that started the workflow, or in any items added to the process as a reference. The benefit of this approach is that users can define the data structures in SharePoint and use SharePoint interfaces to interact with the data. The drawback is that interacting with the data is slightly more complex from the workflow perspective (although not nearly as complex as the Code Reference event) and that none of the K2-generated UI's will be able to reference the data from the list item without custom development work, unless those list items are exposed as SmartObjects or you are using data form the list item in a SharePoint-integrated workflow.

InfoPath form
An InfoPath form is really just a XML data structure, and when an InfoPath form is workflow-enabled, the form data is saved as a XML datafield in the process. Workflow designers can easily interact with data in this XML field in workflow wizards, and can use the Data Event wizard to manipulate and retrieve values in the XML field without any code.

The drawback of this approach is that the data is not easily consumed by other applications, and it is challenging to consume the data in the InfoPath form in a report. You could define data in external data stores and then add these as data sources to the InfoPath form, but then the workflow would still need a SmartObject to get at the same external data store: K2 cannot leverage data connections in an InfoPath form natively.

Some organizations prefer to keep the data in an InfoPath form for the duration of the workflow, and at the end of the process extract the data from the form and insert it into a reporting database. This could be useful for two main reasons: it is easy to work with the data during the workflow, and the data can easily be consumed by reporting applications once the workflow has completed. In addition, the XML datafield in the process effectively becomes the audit trail and point-in-time proof of the data that was approved during the workflow.

Document
The last mechanism commonly used to store workflow data is to use a document (e.g. a Microsoft Word document or Excel spreadsheet) and then "attach" this document to a workflow process. The benefit of this approach is that the document can take any format and layout, and it is easy to create document templates to capture user content. The drawback of this approach is that it is very challenging to extract or manipulate the data in the document (unless the document uses content controls or custom properties exposed on a SharePoint list as columns). It is also very difficult to report on the data in the document.

Note that the document itself is not stored in K2: normally, the document is saved to a SharePoint library and K2 will read values from the document using the link to the document and any properties or content controls in the document.

Selecting the right data storage option
The table below lists the benefits and drawbacks of each data storage approach.

1. **K2 Process DataFields and Process XML Fields**
   This is a good option if the data values are only relevant to the K2 process and do not need to be retrieved by external systems or reports once the workflow has completed

Benefits	Drawbacks

Benefits	Drawbacks
▪ Easy to define and use when designing a workflow ▪ No additional development or programming required to access the data from a K2-generated page or the workflow ▪ Built-in audit capability ▪ Can be hidden from workflow reports ▪ Data can be displayed on the native K2 applications for blackberry and iPhone ▪ Can easily be retrieved in code once you have a handle on the process instance object ▪ Data can only be changed from the context of the current workflow	▪ It is difficult to access this data from another system ▪ Data storage is not optimized for large-scale environments and it can take some time to retrieve the data from the K2 data store if the store is very large ▪ Process datafields cannot be displayed on the native K2 task list ▪ Data types are restricted ▪ Potential for impacting the K2 server performance if the data fields are binary or XML and large objects are stored in them ▪ No field-level security is available ▪ Data can only be changed from the context of the current workflow

2. SharePoint Lists/Libraries and SharePoint Item metadata

This option is most appropriate if the data is already stored in SharePoint, or if it is necessary for users to manipulate the data outside of the process. In addition, this option leverages SharePoint's built-in UI generation capability, so it is not necessary to create custom user interfaces so that users can interact with the data.

Benefits	Drawbacks
▪ Easy to create custom lists in SharePoint ▪ Users can maintain the lists and lists items independently ▪ It is easy to read list item metadata from workflow, IF the workflow is reading information from the same list item that started the process. Reading information form another list item is a little more complex and either requires the ID of the target list item or a SmartObject that exposes the list. ▪ Data is available to other applications fairly easily ▪ Data can be protected based on SharePoint permissions	▪ No built-in audit trail ▪ The data can be modified outside of the workflow (unless the workflow locks the item through check-out or permissions) ▪ It is a little more difficult to access the data for another list item. Will require the target list item ID or a SmartObject for the target list. ▪ Accessing the data from an external application will require code against SharePoint APIs ▪ Storage is no optimized for complex or large-scale reporting requirements

3. **K2 SmartObjects**

This option is most appropriate when the data must be accessed from other systems as well as workflow, and where the data will be re-used in multiple K2 projects. This approach also makes it easy for InfoPath and custom form developers to consume data from the data source without needing to know anything about the data source itself.

Benefits	Drawbacks
▪ SmartObjects can be consumed easily in workflow ▪ Can be easily added as a data source in InfoPath forms ▪ Can be used by other applications such as SSRS and SharePoint BCS ▪ Can be re-used in different K2 solutions ▪ Can be abstracted into simpler objects that are easy for designers to use ▪ Can expose existing data stores	▪ No built-in audit trail ▪ The data can be modified outside of the workflow ▪ The processing time added by K2 to query the data can add overhead to calls. This is normally not a problem in workflow server event, but could be in large, complex reporting scenarios ▪ Do not support complex data types ▪ Can be complex to troubleshoot ▪ If there is no available adapter for the target system, custom code will be required to create an adapter that exposes the target system as a SmartObject service.

Benefits	Drawbacks

4. Assemblies and Web Services

External data stores can be exposed through .NET assemblies, and these can then be consumed by K2 workflows using the Code Reference Event wizard. This approach is most appropriate when the data is only required in the process very few times and will not be used in many processes.

Benefits	Drawbacks
▪ Can leverage existing services and assemblies (provided they use simple types) ▪ Give developers full control over the access to the external system	▪ Not very user-friendly for workflow designers ▪ The K2 wizards may not support all complex types ▪ Maintenance and deployment become more complex ▪ Not easily exposed on K2–generated pages or InfoPath forms.

5. Documents

Data could also be stored inside of documents. For K2-based solutions, this usually means that the data is stored in a Microsoft Excel spreadsheet or Word Document and that document is associated with a workflow via SharePoint. Another option is to store the data in a Microsoft InfoPath document, in which case the data is stored as a XML structure and is rendered as a form though InfoPath.

Benefits	Drawbacks
▪ The forms are effectively the document itself, which means it can be developed fairly easily and consumed easily by users ▪ Using InfoPath, non-developers can create fairly complex and rich user interfaces without writing code	▪ When using Word or Excel, accessing the data in the document can be complex ▪ Adding code to Word, Excel and InfoPath documents to implement more advanced UI features is technically challenging ▪ The data in a word document or excel spreadsheet cannot be structured easily or accessed easily from other systems.

Forms and Data compatibility and complexity

It is also useful to consider the complexity of consuming the various data storage approaches in the user interfaces for a solution and in the workflow itself.

The table below should give you a high-level understanding of what data storage can be used in which Client Event types and how easy it will be to display that data on the user interfaces.

Consumer (Client event/Workflow) Data Store approach	SharePoint workflow Integration Client Event	InfoPath Client Event	Forms Generation Client Event	Default Client Event	K2 SmartForms	K2 SmartActions and Workflow
K2 process-level datafield/xml field	✓	✓ May need to use the data event wizard to transpose process-level datafields into fields on the InfoPath form	✓	⅄	✓	✓
SmartBox-based SmartObject	⅄	⌑	✓	⅄	✓	✓
External system exposed as a SmartObject	⅄	⌑	✓	⅄	✓	✓
External System exposed via an assembly/service	⅄	⅄	⅄	⅄	⅄	⌑
SharePoint Item	✓	⌑ If the SharePoint item is exposed as a SmartObject	⌑ If the SharePoint item is exposed as a SmartObject	⅄	⌑ If the SharePoint item is exposed as a SmartObject	✓
InfoPath form	n/a	✓	⅄	⅄	n/a	✓
Document	⌑ If the data is exposed as a custom property in a SharePoint list	⌑ If the data is exposed as a content control or custom property in a SharePoint list. Documents can also be attached to the InfoPath form.	⅄	⅄	⌑ If the data is exposed as a custom property in a SharePoint list	✓ If the data is exposed as a content control or custom property in a SharePoint list

Legend:

✓ - support through wizards, no code or customization required

⌑ - requires configuration- or wizard-based customization, simple

⅄ - requires developer to write code, more complex

Selecting the right form and data approach for a K2 solution

Selecting the right approach for the data storage and forms technology depends on many factors, more than we can cover in this learning module. It is not possible to give a hard-and-fast rule about which data store to use in general, because it depends a lot on where the data is required and how the data will be used. When selecting a data storage approach for K2 solutions, take the scope of data into account along with the various benefits and drawbacks of the different data storage methods.

Some very general guidelines for selecting a data storage approach:

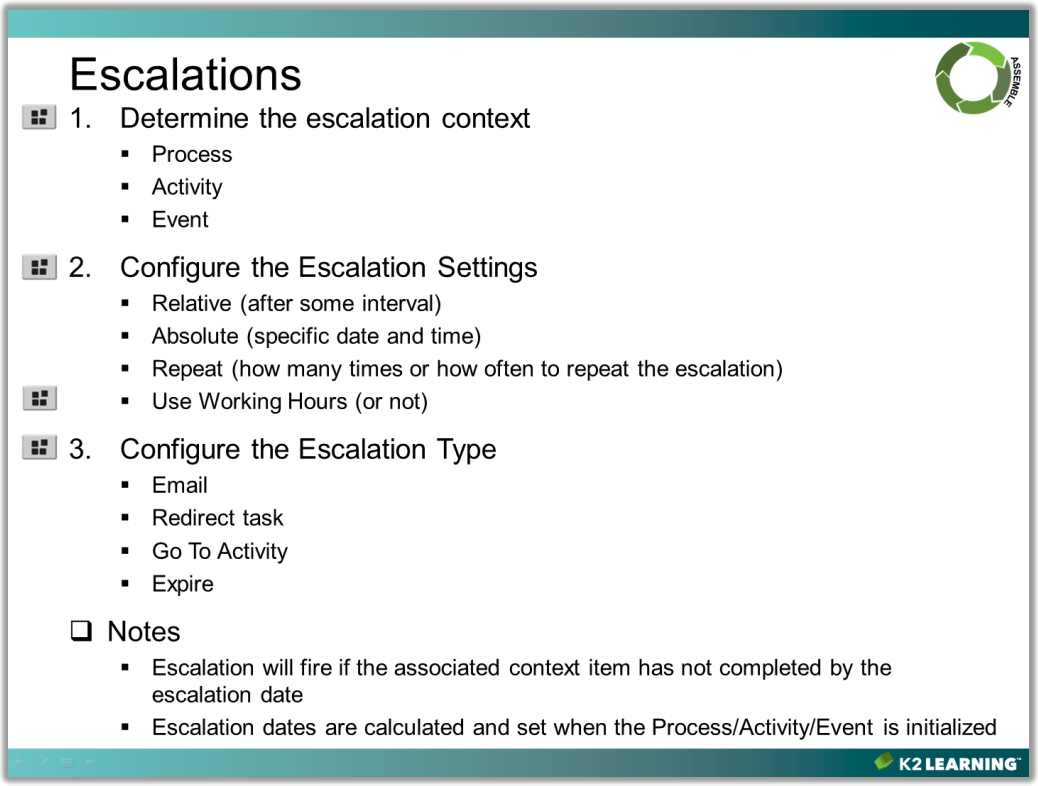
- If the data is only required in the workflow and nowhere else, the quickest and easiest approach is to define the data as a datafield in the process
- If the data is already available, create a SmartObject to expose the data rather than a service or assembly. Only use a service or assembly if the data is read once or twice in a workflow and not really needed in other workflow solutions
- If the data is required in reports, forms or applications after the workflow has completed, store the data in an external store like a SQL database and expose it as a SmartObject for the workflow
- If the data store will grow significantly over time, use an appropriate storage mechanism like SQL server
- Do not store large objects like documents, images or InfoPath forms with document attachments in a K2 process. Really, don't do it.

There can be long-term benefits to exposing data sources as SmartObjects, so take this into account when evaluating the potential time it may take to create the SmartObjects

Based on typical usage of the K2 platform and requirements, the following list provides suggestions (suggestions only, this is not a definitive or absolute list of scenarios and options) on the appropriate data and form technology to use for sample requirements.

Scenario	Data Storage	Process Wizard	Client Event form technology
Simple approval item without complex UI validation or lookup requirements. Simple reporting requirements, no large sets of data	SharePoint List	SharePoint workflow Integration	SharePoint workflow Integration Client Event
Simple to medium-complexity web-based Forms with data connections, rules and validation	SmartObject	SmartForms	SmartForms Client Event
Item with more complex data storage and more advanced form logic such as conditional validation and repeating tables	InfoPath	InfoPath	InfoPath Client Event
Existing Data store, complex reporting requirements, complex user interface requirements	SmartObject	Custom Form	Default Client Event (custom form)
Existing data store, simple or complex reporting requirements, simple user interface requirements	SmartObject	Custom Form	Forms Generation Client Event
Data captured on a template, long narrative or multiple images and tables. Very simple reporting requirements	Document	SharePoint workflow Integration	SharePoint workflow Integration Client Event

Escalations



Escalations are often used in processes to perform some action if a certain interval has passed and the process or a step in the process has not completed yet.

An example of an escalation may be that if an approval step in the process is not completed within 48 hours, the approval task should be re-assigned to the approver's manager and an E-mail sent to both the original approver and the originator of the process to inform them that an escalation has occurred. If the approval task is completed within 48 hours, the escalation will never execute.

K2 escalations consist of a time-based component and an action component. The time component defines WHEN the escalation should fire, and could be a relative date-time (for example, fire the escalation 48 hours after the Activity is started) or an absolute date-time (for example, calculate the date-time 2 days before the item's due date, and then use this date as the Escalation Rule for the Escalation). The action component defines what action K2 should take when the escalation time is reached. It is possible to define multiple escalations with different time and action components on the same item, which is useful if you have a "escalating levels of Escalation" requirement.

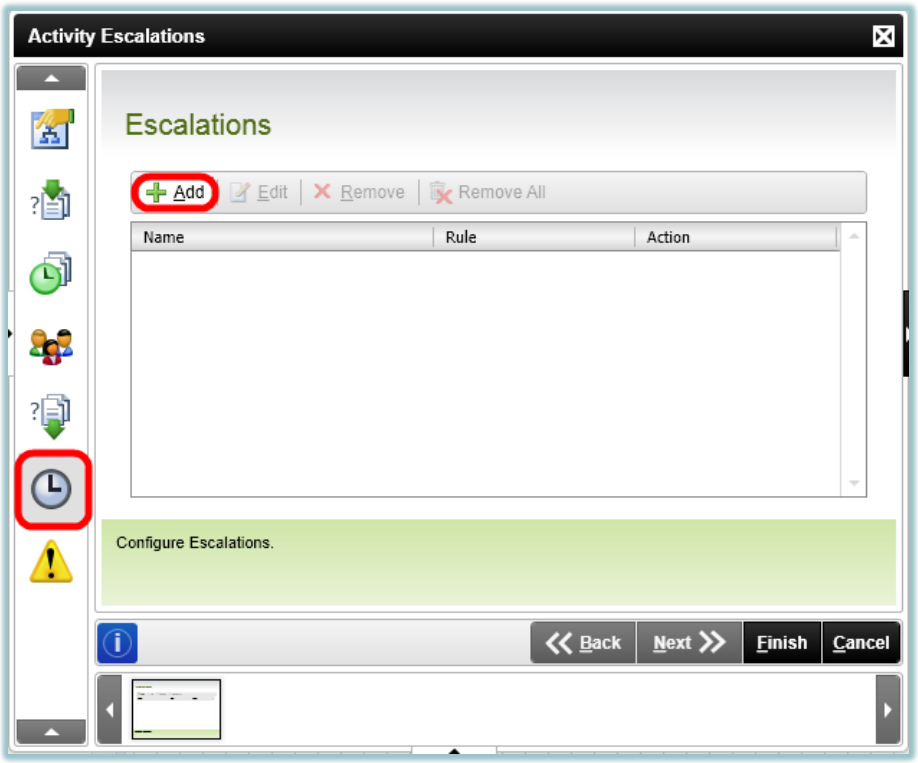
When using K2 Studio to design workflows, escalations can be defined in the following contexts:

Context/Scope	Evaluation Logic	Available Escalation Actions
Process	Fire of the process/workflow has not completed within the target time.	▪ Send E-Mail *Send a customizable E-Mail to any mail address.* ▪ Go To Activity *Force the process to go to another step, and expire one or all currently-active steps in the process* ▪ Custom Code (K2 for Visual Studio only) *Write custom code to execute when the escalation fires*

Context/Scope	Evaluation Logic	Available Escalation Actions
Activity	Fire if the activity has not completed within the target time	▪ Send E-Mail *Send a customizable E-Mail to any mail address.* ▪ Go To Activity *Force the process to go to another step, and expire one or all currently-active steps in the process* ▪ Expire Activity *Expires the current activity and continues with the process* ▪ Redirect *Redirects the task to another user's K2 worklist.* ▪ Custom Code (K2 for Visual Studio only) *Write custom code to execute when the escalation fires*
Event	Fire of the Event has not completed within the target time	▪ Send E-Mail *Send a customizable E-Mail to any mail address.* ▪ Custom Code (K2 for Visual Studio only) *Write custom code to execute when the escalation fires*

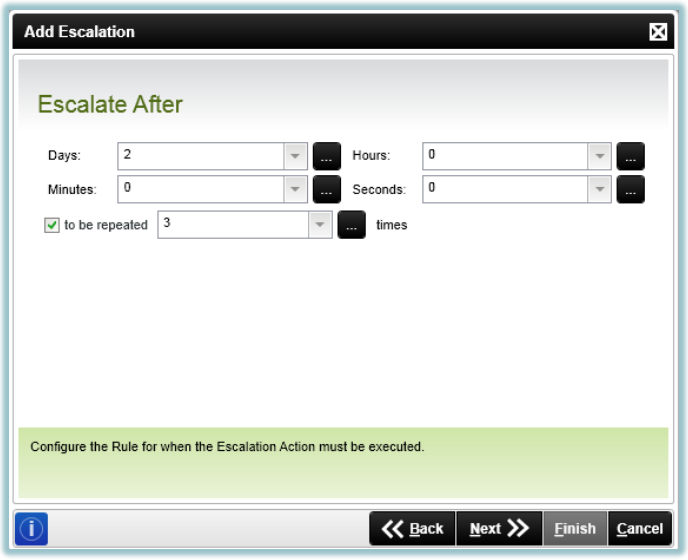
The sequence of screenshots below illustrates how you may add an Escalation to an Activity from the Activity Properties screen. (Right-click an Activity and select **Properties** to open the Activity Properties screen)

Adding an Escalation to an activity

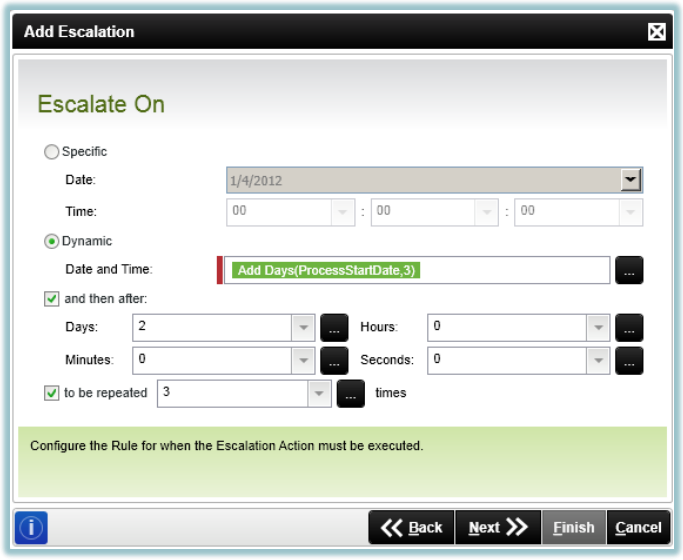


Defining the Escalation Rule

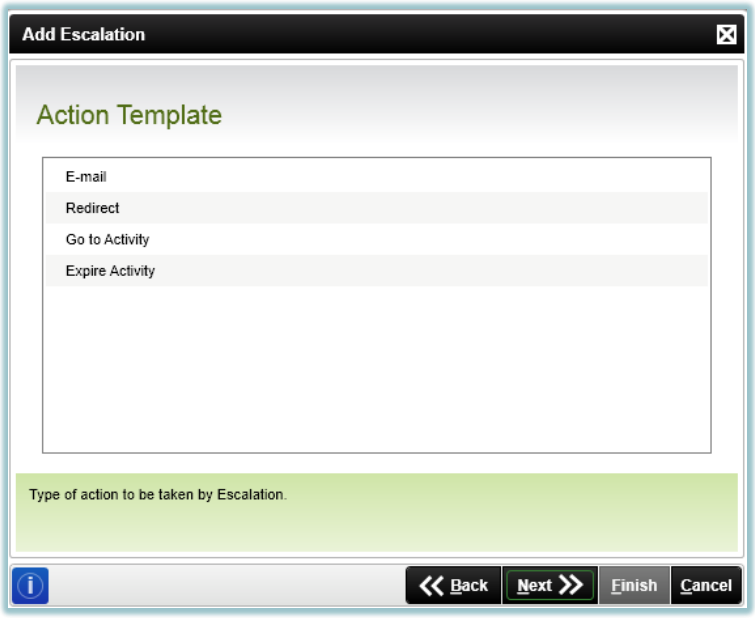
Escalate After (Relative)



Escalate On (Absolute)



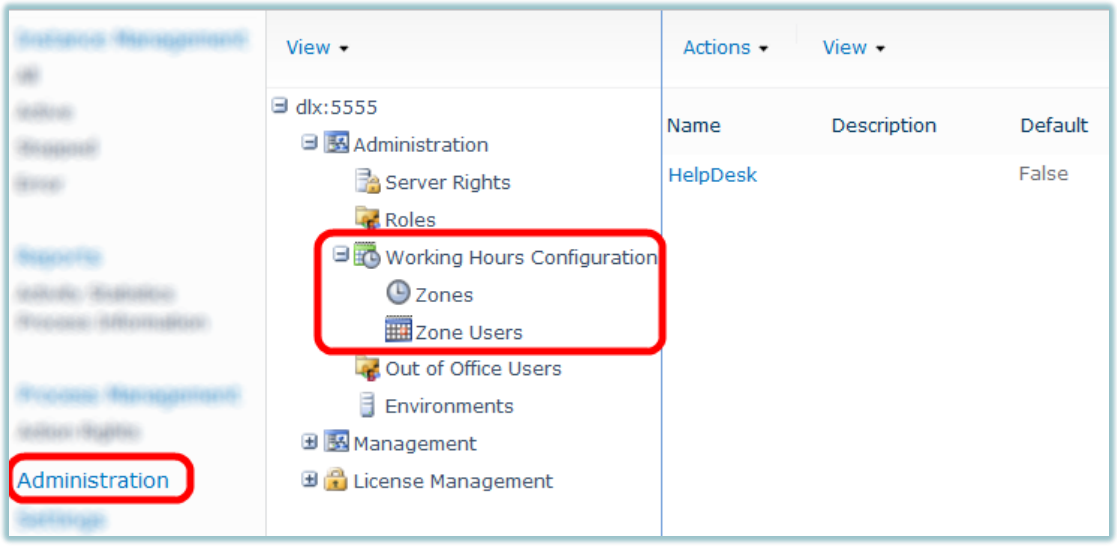
Selecting the Escalation Action



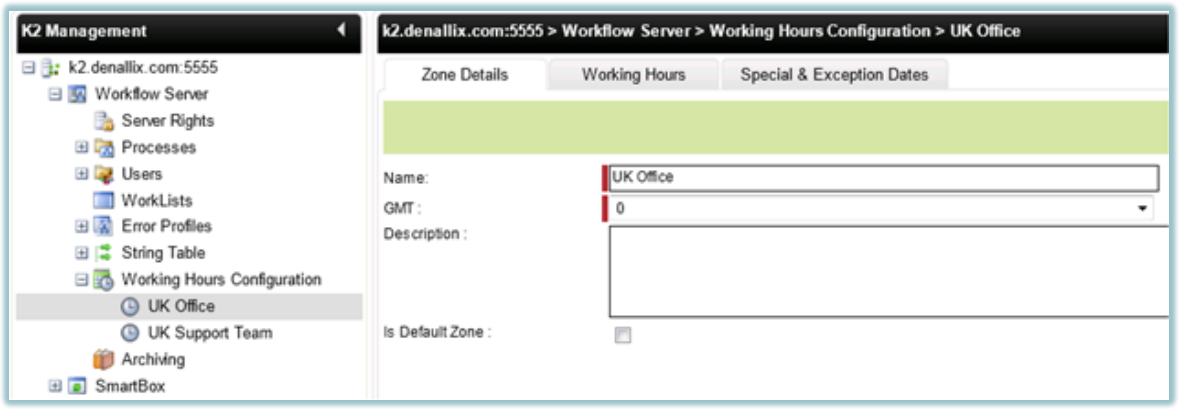
Escalations can also consider working hours. Suppose a corporate helpdesk process has a “Return call within 6 working hours” policy for helpdesk requests. If a helpdesk request is logged at 4 p.m. on a Friday afternoon, the escalation may need to take into account that the helpdesk is only operational until 5 p.m. and then starts again at 8 a.m. on Monday morning. By telling the escalation to consider working hours and then defining those working hours, you can ensure that the escalation is executes at the correct time: 2 p.m. on Monday instead of 10 p.m. on Friday).

For reference, the screenshots below show where working hours can be defined and how they are included in an escalation.

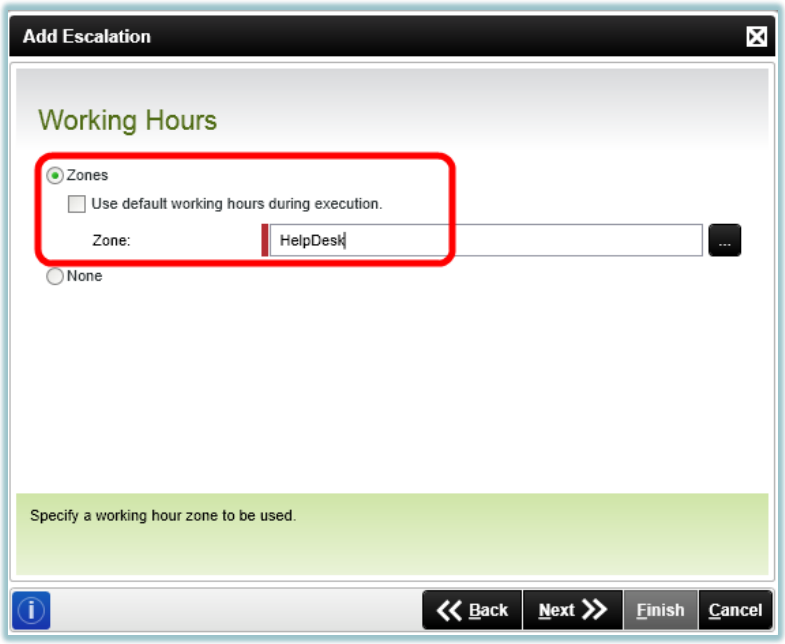
Defining and Managing Working Hours in a K2 Process Portal site



Managing Zones in K2 Workspace



Taking working hours into account in an Escalation



Just because it is possible to define multiple repeating escalations on multiple items in a process doesn't mean that you should use this feature all the time. It is easy for business process owners to get carried away with the number of escalations that can be defined in a process.

Try to keep the number of escalations to a realistic and reasonable level to avoid spamming users with too many E-Mails, and avoid intricate escalations that are so complex that no-one really knows what is happening to the process, when.

If a requirement implies multiple escalations with multiple actions at multiple steps in a process, it may be an indication that the process needs to be refactored. Escalations are not a cure for an inefficient or poorly-controlled process.

01001011
00110010

Escalation Start dates are calculated when the Process/Activity/Event is started and then saved into an internal "To-Do" table in the K2 database. If the Process/Activity/Event completes before the end date is reached, the escalation is not fired and is deleted from the internal "To-Do" database table.

Security

During the assembly phase you may need to configure or set up security for the application as well. As with any solution or project, consider the security concerns while assembling the solution, both from an authentication perspective and an authorization (permissions) perspective. Note that security may be application-specific (such as setting rights on specific artifacts of the application) or configure globally (such as setting permissions on the K2 server or defining global roles in the authentication provider).

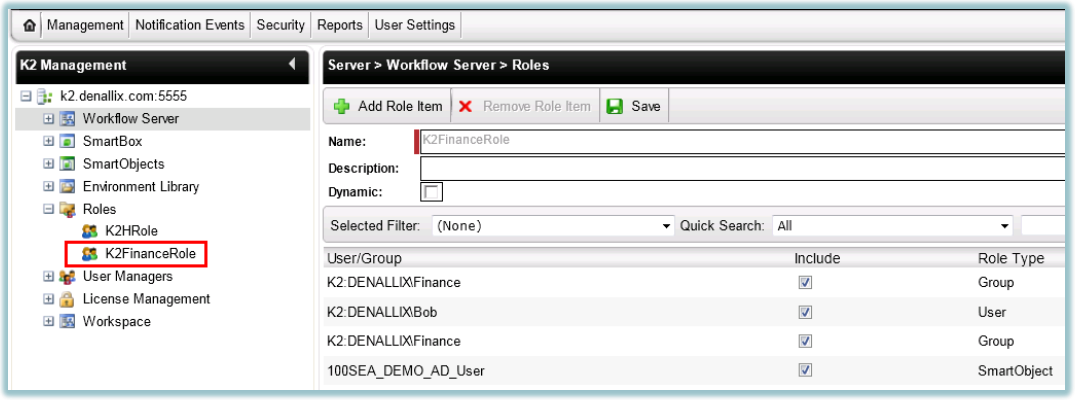
Security is a large topic which we discuss throughout other K2 learning modules, so we will just look at the basics of K2 application security for the moment.

Security Groups/Roles

K2 supports different mechanisms for defining security groups or roles. You can easily reference groups defined in Active Directory, SharePoint for permissions and task assignment, and K2 also provides a mechanism called K2 Roles if you prefer to define these groups in K2 itself. (This is discussed in more detail in the module **200.DUB K2 Workspace – Administration)**

You can also define groups in some external mechanism. Very often, organizations will maintain groups of users in some database, and if that data source is exposed as a K2 SmartObject it is very easy to use the data source in a K2 destination rule for task allocation or in a K2 Role. If you wanted to use this external data source to define K2 security and handle authentication, however, you would need to implement a custom security provider if you do not wish to use K2's built-in SQL user manager component.

Defining Roles in K2 Workspace



Authentication Mechanisms

K2 supports multiple authentication mechanisms, and you may use different authentication mechanisms in parallel. This is more a function of system configuration rather than application assembly though, and by the time you are assembling K2 applications the authentication mechanism should have been configured already.

In most scenarios, organizations leverage K2's built-in support for Active Directory to manage the authentication of users. This means that K2 can re-use the user's security credentials whenever they interact with K2 and they will not need to provide a username/password again. K2 can also leverage security group and role information defines in both Active Directory and SharePoint for process security and task allocation.

In some solutions users outside of the organization's Active Directory may need to interact with K2, or the organization may not even have an Active Directory implementation. For these scenarios, K2 supports custom security providers, which means that developers can create and register a custom security provider mechanism on the K2 server and create custom forms to authenticate these non-AD users. K2 also has support for SharePoint Claims-Based Authentication (CBA) for Incoming Claims only, which means K2 can leverage the work already done in SharePoint to authenticate users with CBA credentials. Note that CBA requires some configuration on the K2 server to enable support for CBA.

We will discuss authentication in more detail in other K2 learning modules, so for now just remember that you can use multiple and non-AD authenticator mechanisms.

Implementing Security

While assembling your application you may need to perform tasks in different systems to set up the security. This may involve configuring the authentication mechanism and setting up groups or roles in that system. Once the groups are defined, you can start to use them in your K2 workflows or K2 security configuration.

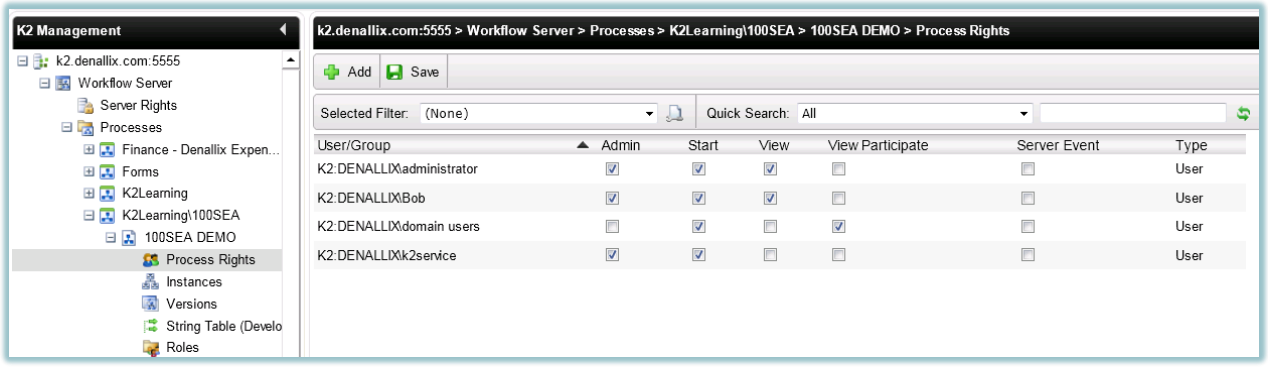
Once you have deployed the K2 application artifacts, you can configure security for those artifacts. We will talk more about this in an upcoming topic, but for now just consider the three most common K2-specific security configurations:

K2 Process Security

K2 processes have different permissions which affect what users will be able to do with the workflow. These permissions are normally configured with K2 workspace or a K2 process portal, and controls who is allowed to Start, report on or administer a workflow definition. Note that these workflows security settings are not tied to workflow versions. We'll talk more about these rights in the topic **Workflow Security and Rights**.

Users do not require any permission on the workflow to be able to **action tasks** that were assigned to them as part of a workflow. The fact that the task was assigned to the user implies that they have permission to perform the task.

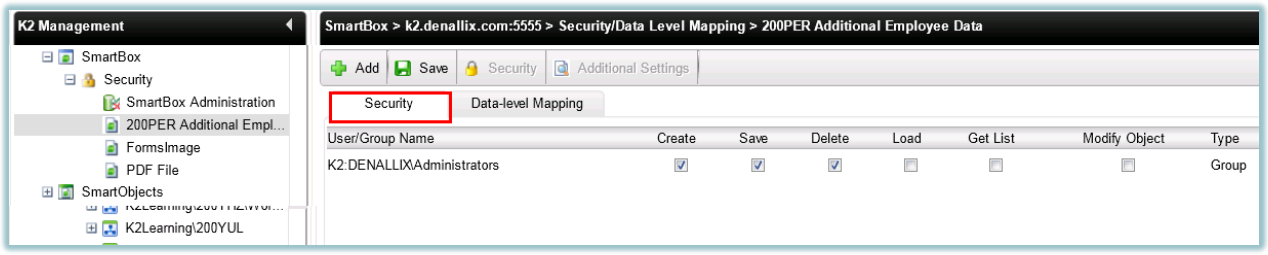
Managing Process Security IN K2 Workspace



K2 SmartBox SmartObject Security

If you have used K2 SmartBox to define a SmartObject, you can use K2 workspace to control who may execute methods of these SmartObjects. We'll talk more about this topic in the learning modules that deal with K2 SmartObjects.

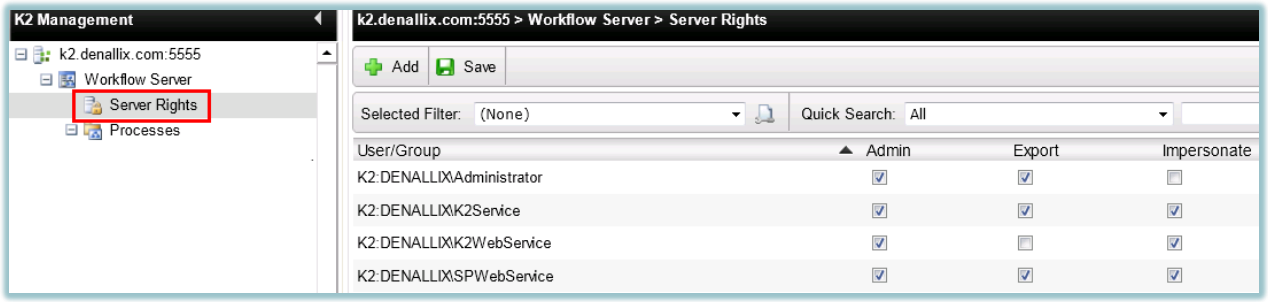
Managing SmartBox-based SmartObjects' security in K2 Workspace



K2 Server Security

Security on the K2 server is normally defined to control who is allowed to deploy new application artifacts to the K2 server and who is allowed to administer the K2 environment. Normally, these security settings are configured for a K2 environment before any applications are assembled, but depending on what your application does, it may be necessary to modify these rights. We discuss these security considerations further in the module **200.DUB K2 Workspace – Administration.**

Managing Server Rights in K2 Workspace



The quick-reference sheet QRS.002 - K2 Securable Components lists the various security options and settings supported by K2.

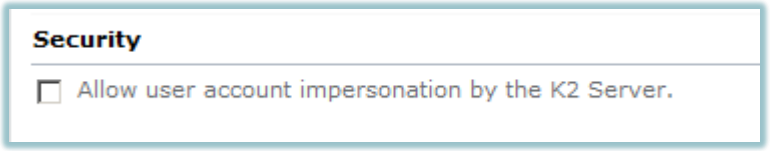
SharePoint Security

SharePoint security options can also play a role in securing a process. Access rights can be configured on SharePoint sites, lists and libraries that users use to start workflows or to report on workflow, or you may configure security in a SharePoint list/library that has been exposed as a K2 SmartObject.

01001011
00110010

Lists and Libraries that are exposed as SmartObjects can also be protected from unauthorized use by setting the appropriate security option. The **Allow user account impersonation by the K2 server** setting will always pass the currently-connected user's credentials to the SharePoint list so that any existing SharePoint security is maintained. If this setting is not checked, K2 will always use the K2 service account's credentials when it interacts with the SharePoint list through a SmartObject.

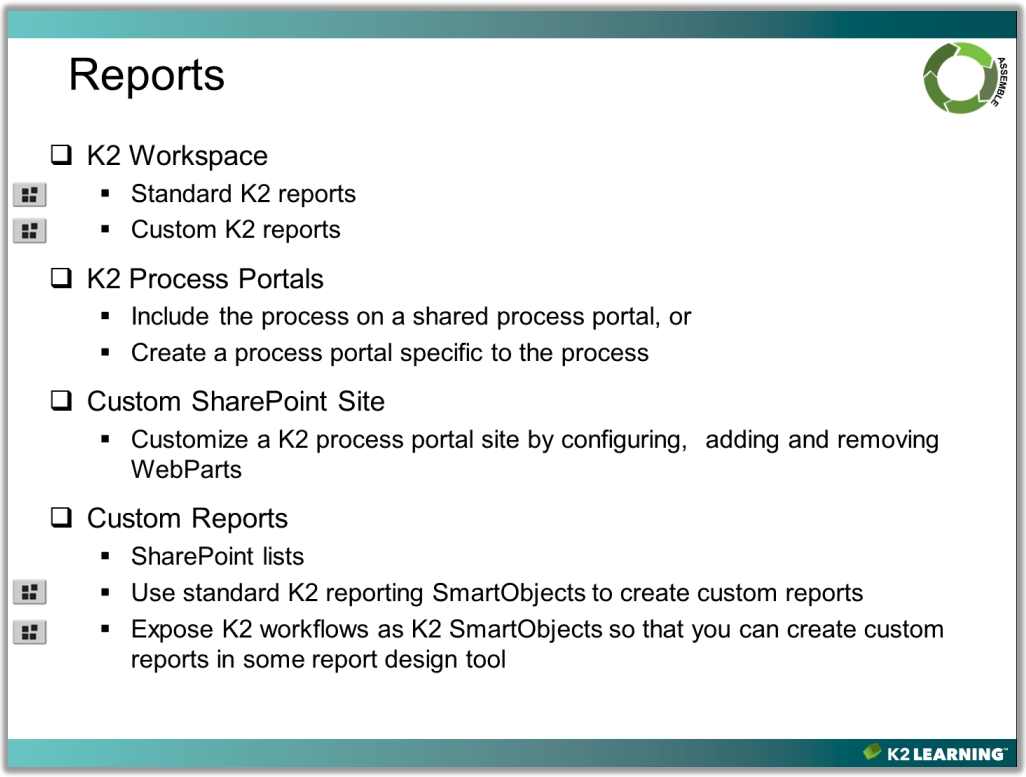
Setting the security option for SmartObjects based on Lists and Libraries in SharePoint



Other Security

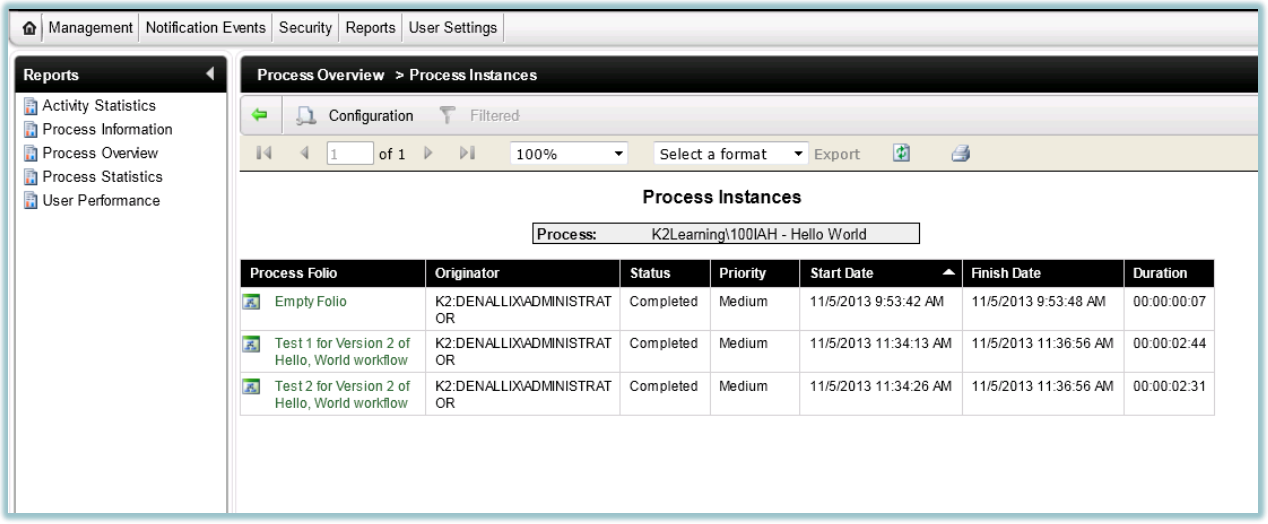
You may need to configure or set up authorization for other systems as well. Most often this involves setting up the security settings for services that interact with the back-end LOB systems that have been exposed as K2 SmartObjects, and we will discuss that more in the modules that deal with K2 SmartObjects, **100.SYD K2 SmartObjects – Fundamentals** and **200.PER K2 SmartObjects – Intermediate.** For now, just note that you may need to configure security in an external system as part of the application assembly process.

Reports



During the assembly phase you may need to create custom reports for your application, or leverage the standard reports that are provided with K2. K2 workspace allows you to run these standard reports and allows you to create custom reports with a report design tool. When using this report design tool, you can create custom reports that combine workflow reporting data with business data to create relevant and useful BI reports.

Accessing the standard reports in K2 workspace



The Learning Module **100.BRU K2 Workspace – Reporting** goes into more detail on each of the available standard K2 Reports

If your organization uses SharePoint, you can also use K2 Process Portals to report on workflows. You would normally create a new SharePoint site using a Process Portal as the Site Template, and then add the deployed workflow to the site. Depending on your organization's security and administration requirements, you could create separate Process Portal sites that contain specific workflows or use a shared process portal site with multiple workflows. You could also create a custom SharePoint site and add the available K2 reporting web parts to the site to create a custom

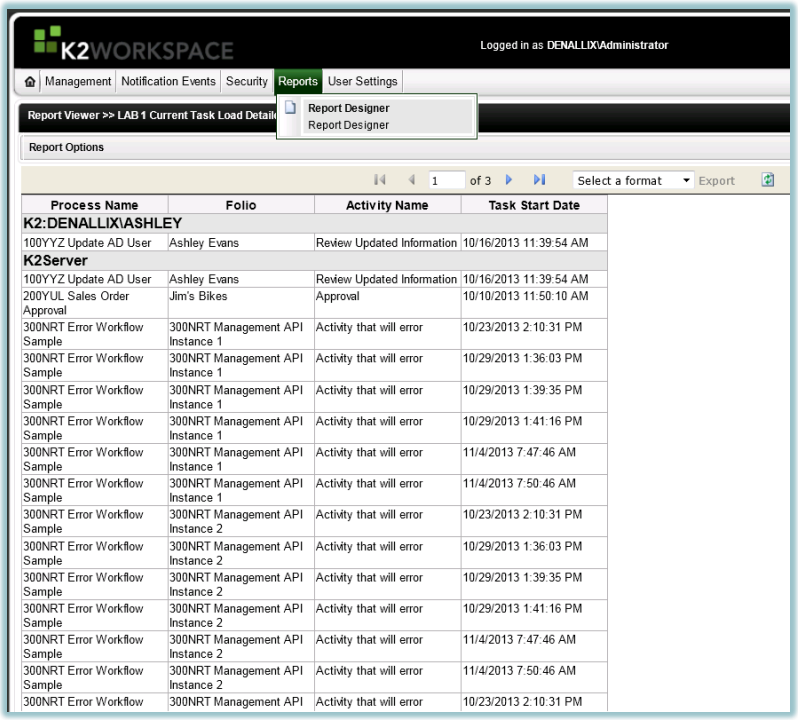
"dashboard" solution for your application. The learning module **200.CPH Process Portals and Web Parts** covers Process Portals and Web Parts in more detail.

Custom reports

A popular approach for reporting on processes is to define SharePoint lists, and then to update items on those lists during the process execution. For example, a column could be added to the list item that the process is started from, and that column can be updated by the K2 process with the latest status of the process. This gives users a quick report on the status of their submitted item. This approach could also be used to update a separate SharePoint list with reporting information.

You may also create custom reports that combine workflow reporting data with business data in other report design tools. Normally, this is achieved by creating SmartObjects for your Workflow along with SmartObjects for the business data. Once this is done, you can use tools like the Visual Studio SSRS report designer to combine methods from these SmartObjects to create relevant business reports. The learning module **300.UKY Creating Custom Reports with Report Design Tools** discusses this in more detail.

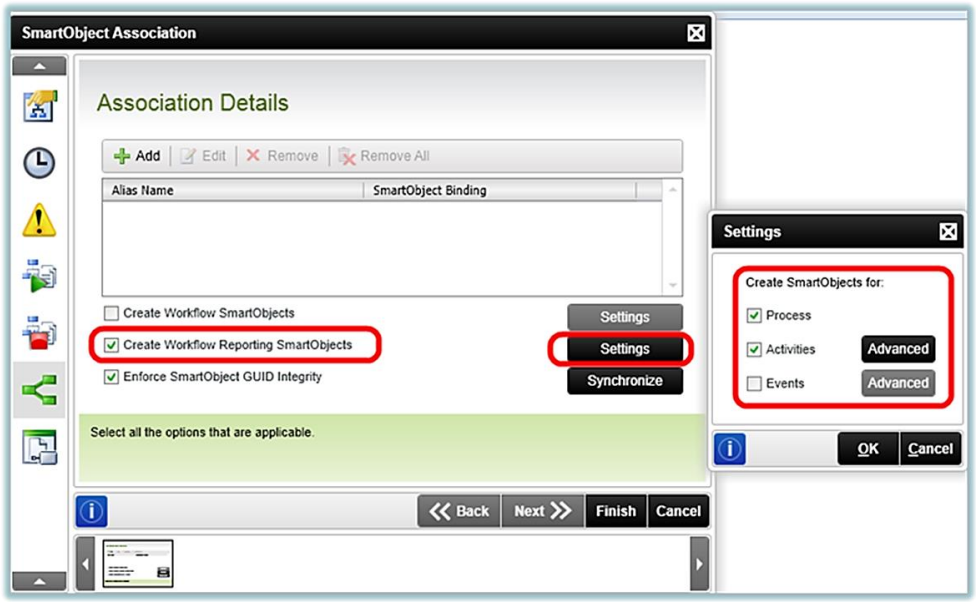
A custom report designed in K2 Workspace Report Designer



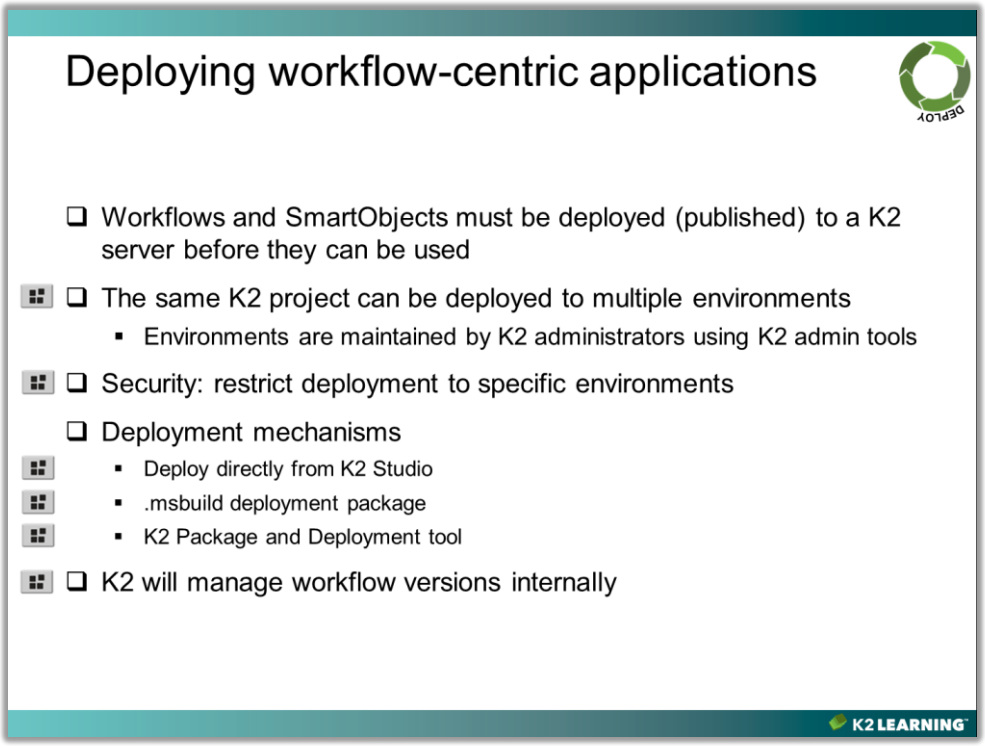
Process Name	Folio	Activity Name	Task Start Date
K2:DENALLIX\ASHLEY			
100YYZ Update AD User	Ashley Evans	Review Updated Information	10/16/2013 11:39:54 AM
K2Server			
100YYZ Update AD User	Ashley Evans	Review Updated Information	10/16/2013 11:39:54 AM
200YUL Sales Order Approval	Jim's Bikes	Approval	10/10/2013 11:50:10 AM
300NRT Error Workflow Sample	300NRT Management API Instance 1	Activity that will error	10/23/2013 2:10:31 PM
300NRT Error Workflow Sample	300NRT Management API Instance 1	Activity that will error	10/29/2013 1:36:03 PM
300NRT Error Workflow Sample	300NRT Management API Instance 1	Activity that will error	10/29/2013 1:39:35 PM
300NRT Error Workflow Sample	300NRT Management API Instance 1	Activity that will error	10/29/2013 1:41:16 PM
300NRT Error Workflow Sample	300NRT Management API Instance 1	Activity that will error	11/4/2013 7:47:46 AM
300NRT Error Workflow Sample	300NRT Management API Instance 1	Activity that will error	11/4/2013 7:50:46 AM
300NRT Error Workflow Sample	300NRT Management API Instance 2	Activity that will error	10/23/2013 2:10:31 PM
300NRT Error Workflow Sample	300NRT Management API Instance 2	Activity that will error	10/29/2013 1:36:03 PM
300NRT Error Workflow Sample	300NRT Management API Instance 2	Activity that will error	10/29/2013 1:39:35 PM
300NRT Error Workflow Sample	300NRT Management API Instance 2	Activity that will error	10/29/2013 1:41:16 PM
300NRT Error Workflow Sample	300NRT Management API Instance 2	Activity that will error	11/4/2013 7:47:46 AM
300NRT Error Workflow Sample	300NRT Management API Instance 2	Activity that will error	11/4/2013 7:50:46 AM
300NRT Error Workflow	300NRT Management API	Activity that will error	10/23/2013 2:10:31 PM

To create reporting SmartObjects for a workflow, access the workflow properties and then use the SmartObject Associations tab to Create Workflow Reporting SmartObjects, as shown in the screenshots below. You can use the Settings button to select which parts of the workflow should be exposed as SmartObjects. You can also use the standard workflow reporting SmartObjects if you prefer.

Creating workflow reporting SmartObjects for a specific workflow



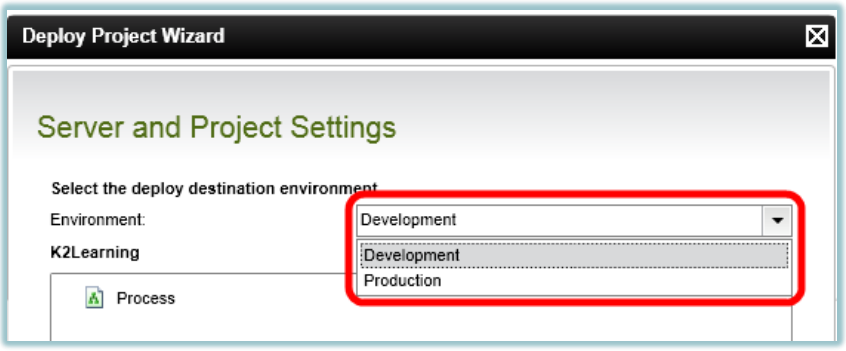
Deploying workflow-centric applications



Because K2 workflows and SmartObjects are executed by the K2 server, these items will not be available until they have been deployed (published) to a K2 server. Note that K2 deployment only deploys K2 artifacts or some artifacts referenced by the solution such as InfoPath forms. If you are referencing external systems, using custom user interfaces or custom reports, there will be additional deployment considerations for your application.

Many organizations have separate development, testing and production environments for K2. When a K2 project is deployed a specific target environment must be selected, as shown in the screenshot below:

Selecting the environment where the K2 project will be deployed

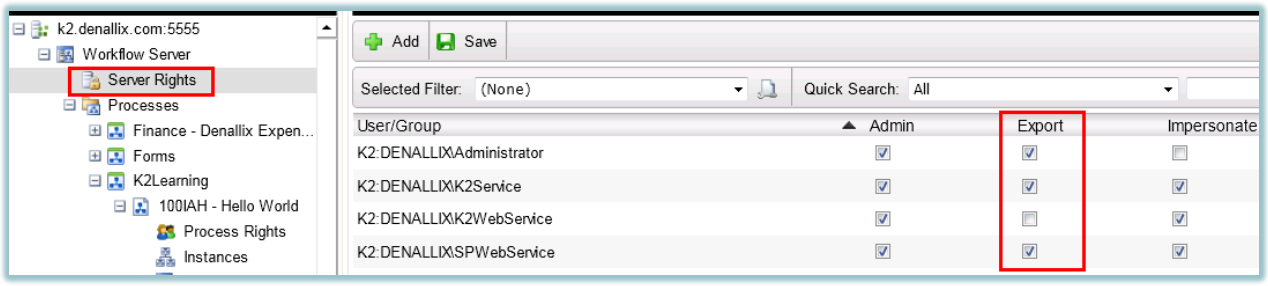


The list of available environments is defined in the K2 environment library, and is normally maintained by the K2 administrator. As part of this maintenance, the administrator will include the various server names and configuration settings specific to the target environment. The K2 deployment task will use the environment library to obtain the connection strings and server names required to connect to the target environment, and then deploy the solution components to the target environment.

K2 security is frequently utilized to restrict deployment to certain people for certain environments, For example, K2 security can be used to “lock down” the Production environment so that only K2 administrators can deploy K2 project to Production, while the Development environment may be left “wide open” so that any user can deploy K2 workflows to Development.

The module **200.DUB K2 Workspace – Administration** describes environment settings and K2 environment security in more detail.

Restricting Export rights for a K2 server



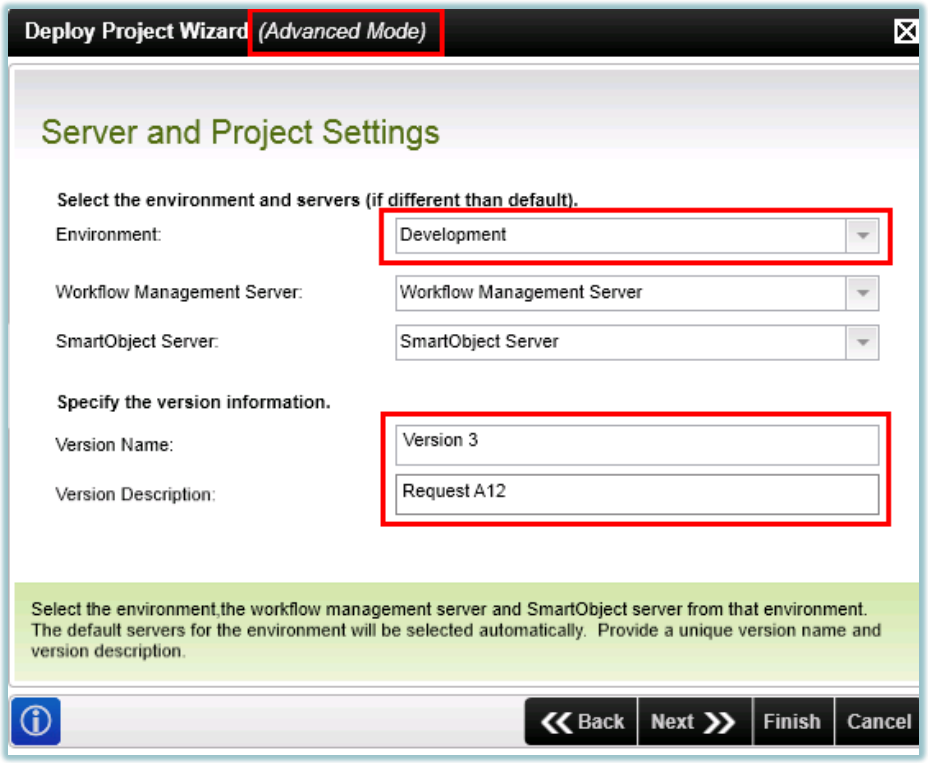
Deployment mechanisms

K2 supports a few deployment approaches you can use to publish workflow-centric applications to a K2 server:

Deploy directly from K2 Studio

This approach is used when the workflow developer has rights to publish the K2 artifacts directly to the K2 server. This deployment mechanism will published the workflows and SmartObjects for the solution and may deploy additional items, depending on the solution.

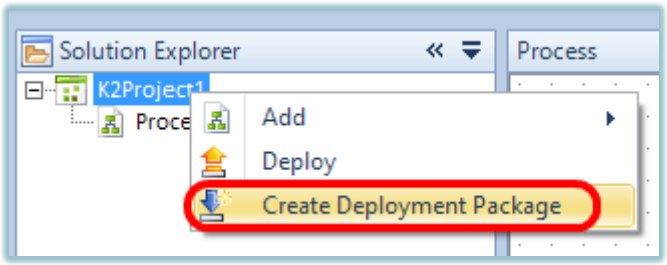
Deploying directly from K2 Studio (advanced mode)



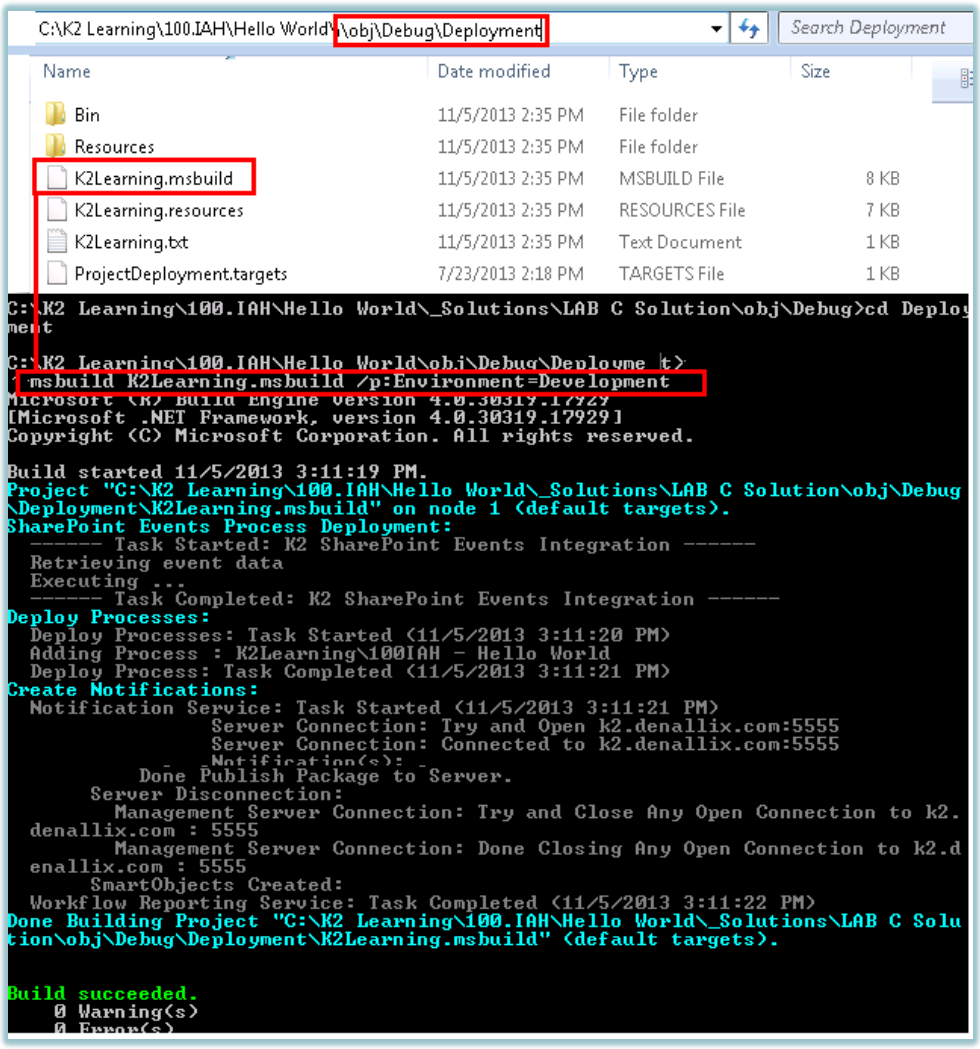
Msbuild deployment package

Developers can generate a .msbuild deployment package from a K2 project as well. This approach is most often used when the developer must hand over a package of items to a system administrator to publish to a K2 server. Note that the deployment package only includes K2 project items. We will discuss this approach further in other learning modules, or you can also refer to the KB article How to use the Deploy Package for more information.

Creating a deployment package in K2 Studio



Deploying a package with the command-line

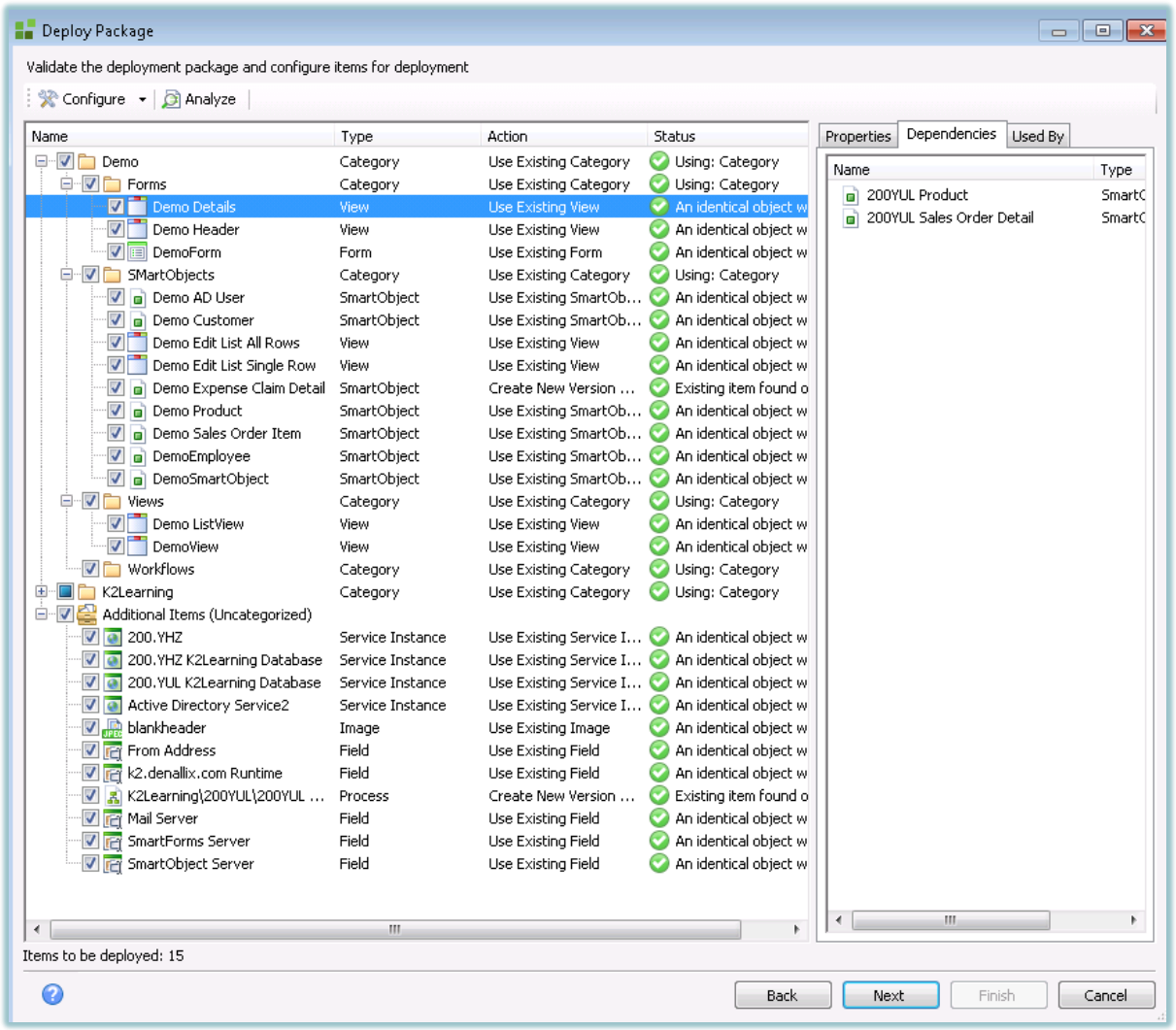


K2 package and deployment utility

For solutions that include K2 smartforms, you will need to use the K2 package and deployment utility to package and publish K2 artifacts.

Using the package and deployment utility to deploy artifacts.

Note Status and dependency checks that are performed as part of the deployment



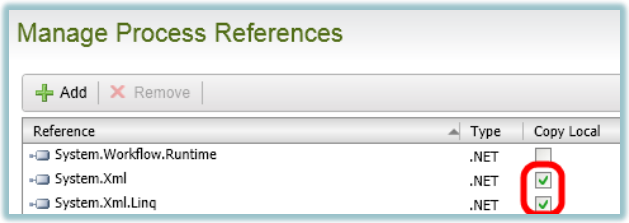
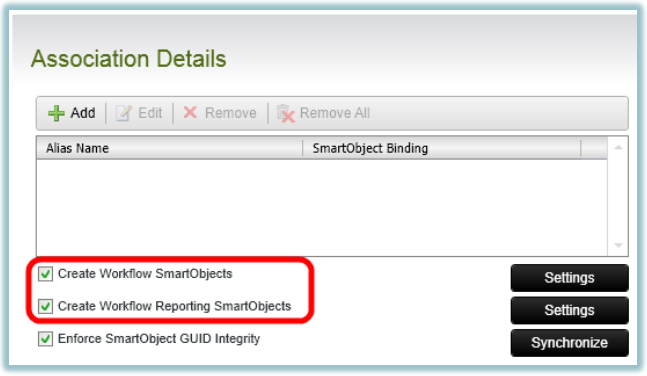
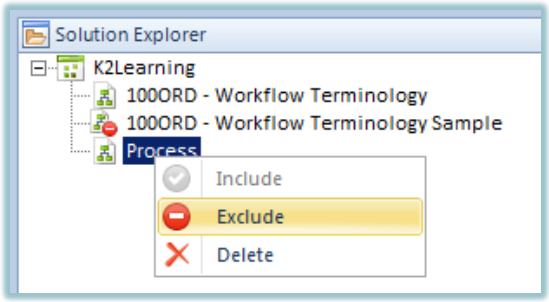
Artifacts deployed by K2

K2 deployment mechanisms only deploy K2 artifacts such as Workflows and SmartObjects. Depending on the type of workflow being published, K2 may also deploy additional items to allow the workflow to work. The table below lists these items. If you are using the Packaging and Deployment utility, you can include K2 SmartForms with the deployment package as well. All other project items (such as custom user interfaces, custom reports, security groups or Roles and so on) will need to be deployed separately.

The table below describes what items a K2 project deployment will publish. Anything NOT mentioned in this table will NOT be deployed by K2 automatically, and the workflow designer or K2 administrator will need to perform additional, manual deployment tasks.

Items that are included in K2 deployment automatically

Process type/Client Event Type/Process item	**Deployment Items included**
InfoPath Integration process	K2 will deploy the InfoPath form template to the specified deployment location (SharePoint form library, content type or network location). If the form template includes SmartObject integration and the form is deployed to SharePoint, K2 will create the necessary data connection files in SharePoint automatically.
SharePoint Workflow Integration process	K2 will deploy the necessary SharePoint workflow entities to expose the K2 workflow as a SharePoint workflow. If Automatic start of the process is enabled, K2 will create the necessary SharePoint event handlers to start the SharePoint workflow.
SharePoint Events Process	K2 will register the necessary event handlers in SharePoint to start the K2

	process when the event occurs in SharePoint.
Forms Generation Client Event	K2 will create a Web Deployment project and publish the generated ASP pages to a web application on the K2 workspace server. Any workflow that includes a Forms Generation Client Event must be deployed locally on the K2 server, it cannot be deployed remotely form the developer's workstation.
InfoPath Client Event	K2 will deploy the form template associated with the client event to the specified deployment location automatically. If the form template includes SmartObject integration, K2 will create the necessary data connection files in SharePoint automatically.
SharePoint Workflow Integration Client Event	K2 will generate and publish the generated pages (.ascx files) to the SharePoint Front-End servers automatically.
Referenced Assemblies	If the "copy local" option is set, K2 will include the assemblies as part of the deployment package and publish them to the K2 server. If this option is not set, the server administrator must copy the referenced assembly to the K2 server manually. 
SmartObject Associations	If workflow SmartObjects or Workflow Reporting SmartObjects are included in the process properties, K2 will generate and deploy the necessary SmartObject automatically. 
Other Project Items	K2 will only deploy SmartObjects and Workflows in the current project if the item is included in the project. To exclude specific items form the deployment process, use the Exclude form Deployment option to exclude those items form the deployment task. **Excluding a specific item form the deployment task** 
SmartForms (Forms and Views)	SmartForms can only be deployed with the K2 Package and Deployment utility.
Service Instances	The Package and Deployment utility can deploy Service Instances as well

Typical solution items that require manual deployment

The table below lists some of the more common items that may be included in a K2 solution that are NOT included in the K2 deployment task. These items will need to be manually configured or deployed. We have provided additional notes where appropriate.

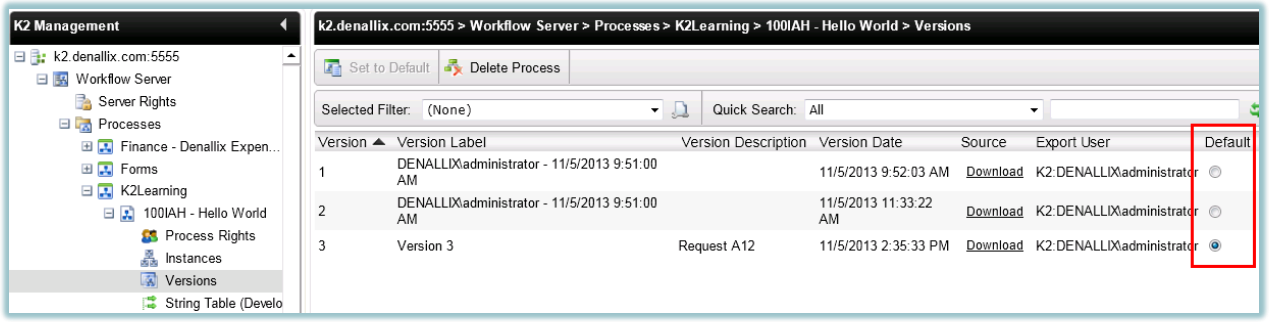
Item/Component	Notes
SmartObject services and service instances	K2 projects only deploy the SmartObjects included in the project. SmartObjects may depend on service types or service instances. Service Types and Instances will need to be manually deployed, and note that the Service Instance Identifiers (GUIDs) must be the same between all K2 environments Note: The package and deployment utility can deploy SmartObject Service Instances
K2 Roles	Roles will need to be manually defined on the target K2 server if those roles are used in the workflow. Note: The package and deployment utility can deploy K2 Roles
Process Security Settings	Process security need only be configured manually at the first deployment. Subsequent deployments will use the same set of process security settings.
Any other security settings	Additional security settings (such as access to databases, SharePoint sites and so on) will need to be manually configured.
Process Portal sites in SharePoint	You will need to create new process portal sites based on the K2 Process Portal site template in the target SharePoint environment
Process Portal settings	You only need to add a process to a Process Portal once. Subsequent deployments do not affect the Process Portal settings.
External data stores	This includes SQL databases, SharePoint lists, other data stores, custom CRM entities and so on. K2 will deploy the SQL tables for SmartBox-based SmartObjects.
SharePoint lists and libraries and list items	Lists and Libraries should have exactly the same definition between the two environments. Also remember to add any list item entries that may be required, such as lookup lists.
SharePoint security and groups	
Custom Reports	Custom reports created in K2 workspace can be saved an imported as SSRS reports.
Custom forms (e.g. ASP.NET projects)	
InfoPath forms with code-behind	See `http://help.k2.com/en/KB001244.aspx`
SharePoint WebParts	

Workflow Versions

Process versioning is supported in K2, and it is possible to run multiple versions of the same process at the same time. For example, let's say Version 1 of the process has been deployed, and a few instances are already running on Version 1. A designer makes changes to the process and deploys the process again. K2 will create Version 2 of the process and set Version 2 as the default version to use when new instances are started. However, the instances that are running on Version 1 will be left alone and will continue running on Version 1 until they complete. Running or Active processes instances are NOT automatically upgraded to newer version of the workflow.

Process versions can be managed by K2 administrators using the administration tools in K2 Process Portals and K2 Workspace. Administrators can download process definition files or can revert to a previous version of a process, in case the current version is broken or was deployed by mistake. The module **200.DUB K2 Workspace – Administration** describes workflow versioning in more detail.

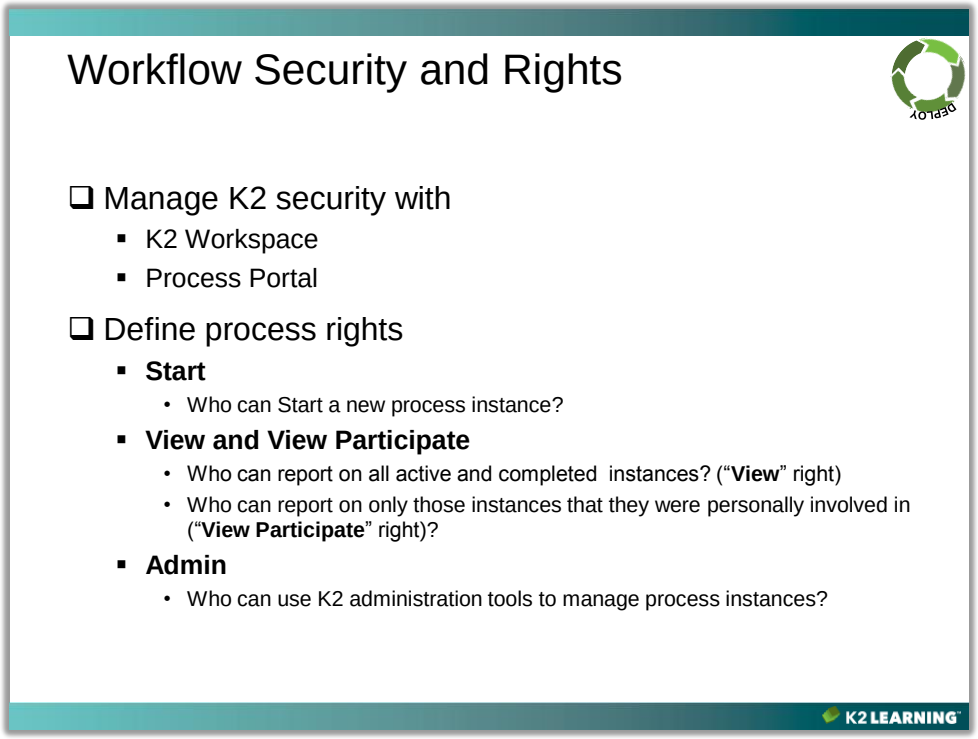
Workflow Versions in K2 Workspace



01001011
00110010

K2 blackpearl provides APIs that allow in-flight processes to be migrated from one version to another. This feature is intended to be used very infrequently and only to address critical issues in processes.
The K2 underground has some sample applications that illustrate the use of this API. See `http://www.k2underground.com/groups/process_version_migration_utility` for an example

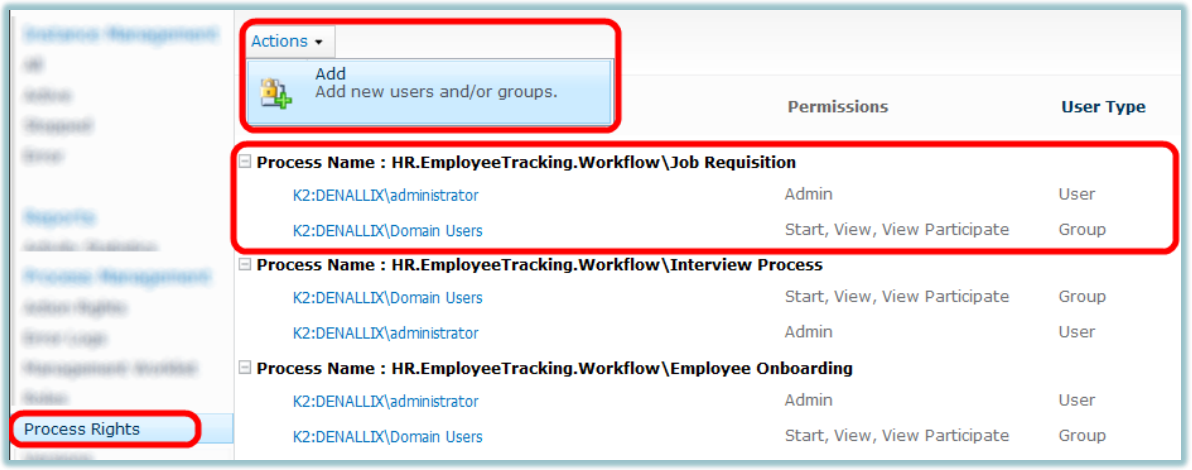
Workflow Security and Rights



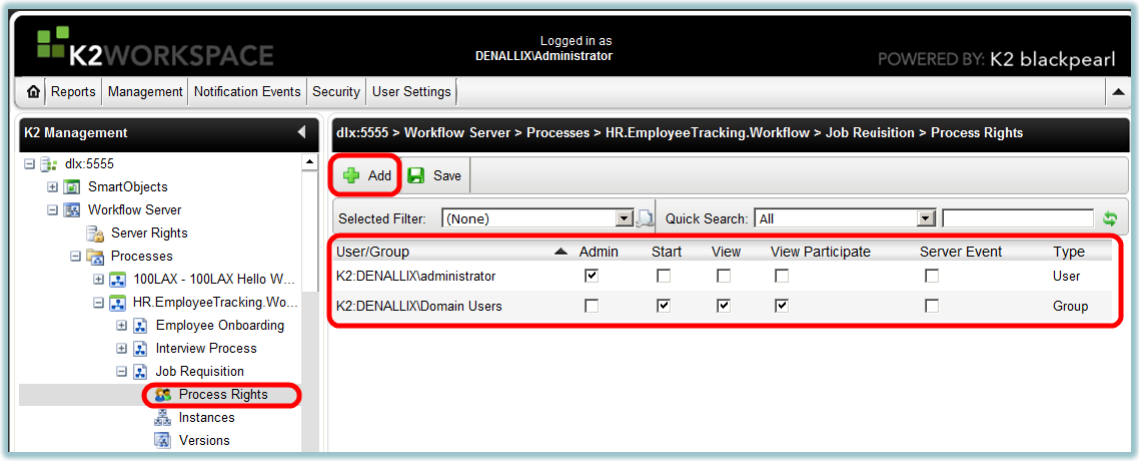
When a workflow is deployed for the first time, only the user who deployed the process will be have access to the process. To make the process available for other users, the process security must be configured.

Process security is normally configured using K2 workspace or through a K2 process portal. The screenshots below show where the process rights are set in each environment:

Setting process security in a Process Portal



Setting process security in K2 Workspace



K2 processes have different levels of rights which determine what users will be able to do with the process. Process security can be defined for user accounts as well as groups, and may use security information from Active Directory, SharePoint or a custom security provider.

The table below describes each of the available workflow security settings:

Permission	Description
Start	The user/group is allowed to start (initiate) a new workflow instance. If this permission is not set correctly, users would typically get a “The user/group has insufficient rights to start the process” error when they attempt to start the workflow. This permission is normally given to all users, unless the process is restricted to certain groups/users.
View	The user/group can report on ALL workflow instances for that particular workflow, regardless of whether they actually participated in the workflow or not. This permission is normally given to the process owner, administrator and any users who need to run overview reports on the process.
View Participate	The user/group can only report on those INSTANCES of the workflow that they personally participated in, either as the originator or during a client event somewhere in the workflow. This permission is useful when you want to allow users to report on their own workflow instances, but not on other users’ workflow instances. Normally, this permission is given to all users, so that users can at least report on their instances they started or that they were involved in.
Admin	The user/group is allowed to administer the workflow (for example, delete, stop or start process instances, or redirect process instances to other activities in the workflow). This permission is normally given to the workflow administrator or process owner.
Server Event	The user/group is allowed to finish an asynchronous server event by calling the *FinishServerItem* method in the K2 API. This is a more advanced scenario and normally you do not need to give any user this permission unless you are using asynchronous server events.

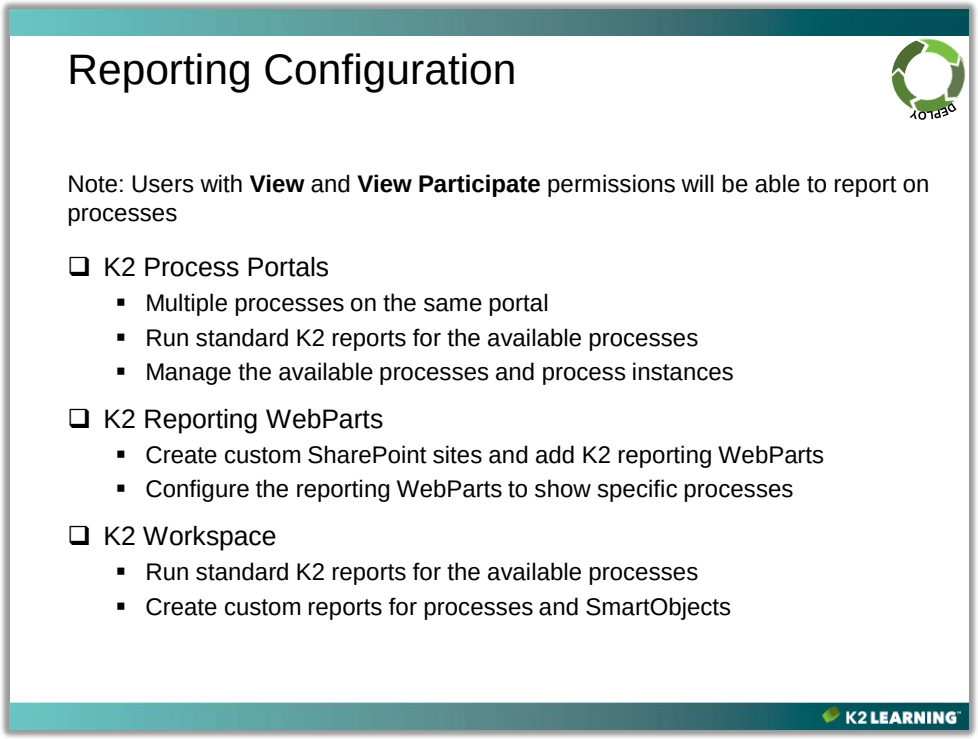
Process security only needs to be configured the first time that the process is deployed. Subsequent deployments of the workflow will use the same set of configured security rights as the previous versions.

```
01001011
00110010
```

K2 security works on a least-restrictive model. So if my user account has both “View” and “View Participate” rights, I will be able to see all instances since “View” is less restrcitive than “View Participate”.

Process security is set on the process definition, not on the process version. Therefore, you cannot use different levels of security on different versions of the process.

Reporting configuration



Once a process has been deployed, the process reporting options for the process can be set up.

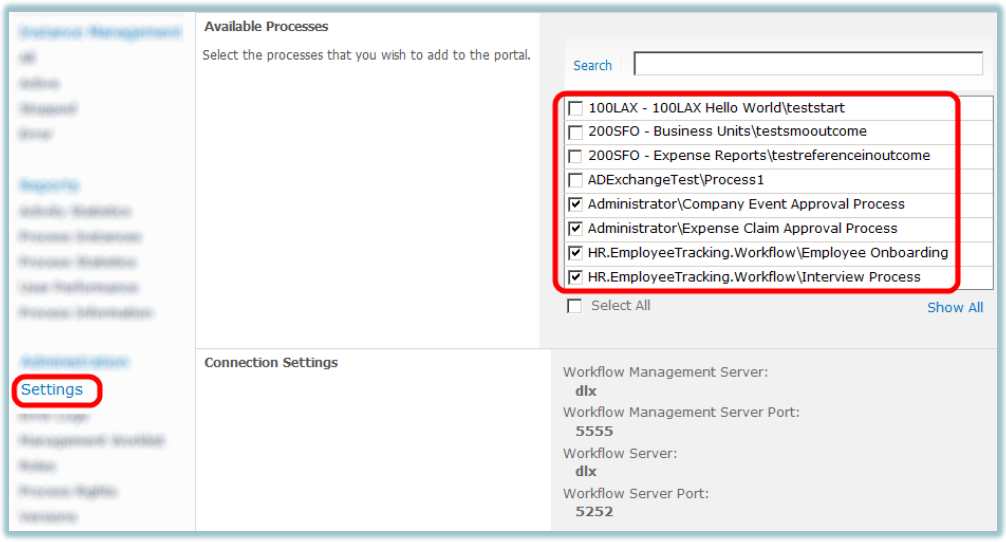
Remember that only users with *View* and *View Participate* permissions will be able to report on K2 processes.

K2 Process Portals

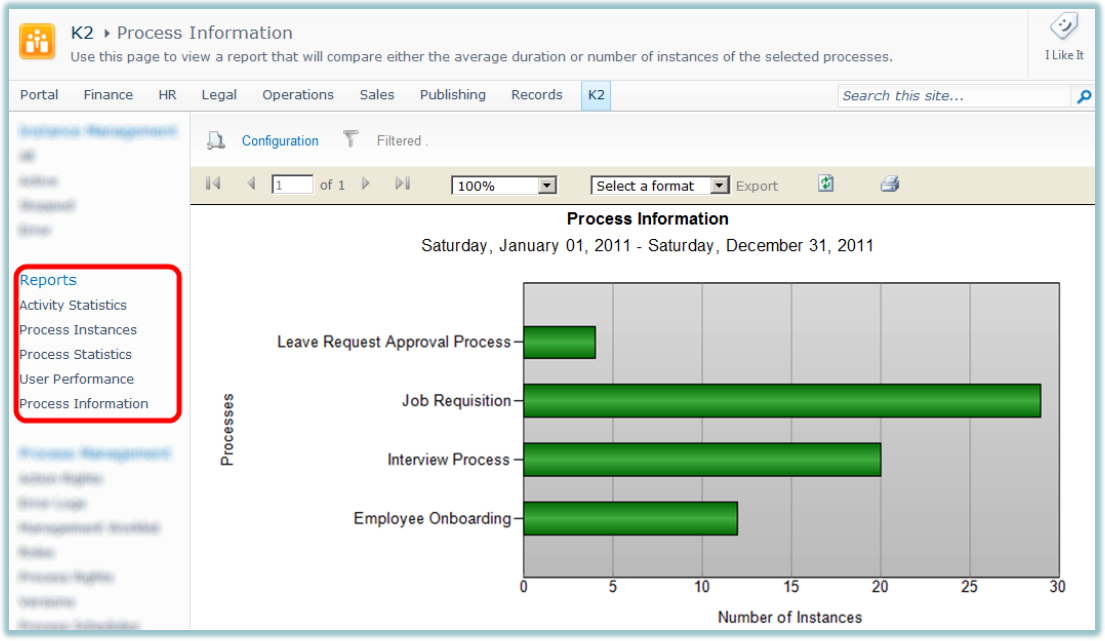
A common approach in organizations that use Microsoft SharePoint is to add the process to a K2 Process Portal site in SharePoint. This will allow users with appropriate permissions to use a SharePoint site to run the standard K2 reports against the process, or manage the process as well as the process instances.

Process portals are SharePoint site templates, and are normally used to separate the reporting and administration of processes into functional areas of the organization. For example, the finance department should only be able to manage and report on their processes, while the HR department should only report on their processes. There may be a global process portal as well, to allow the K2 administrators to have a global view of all processes in the organization. These process portals are also protected through SharePoint security configuration, to ensure that only authorized users are able to access the portal in the first place.

Configuring the available K2 processes for a Process Portal site



Running reports against Processes in a process portal site

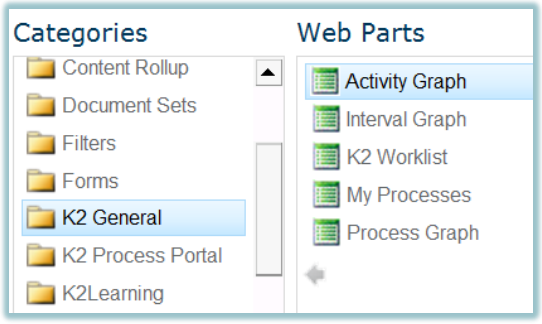


When using K2 Process Portals, it is a good idea to secure the process portal and the K2 process permissions using the same SharePoint groups. This way, the group membership can be maintained in one location and it will be applied automatically to both the K2 process and SharePoint site.

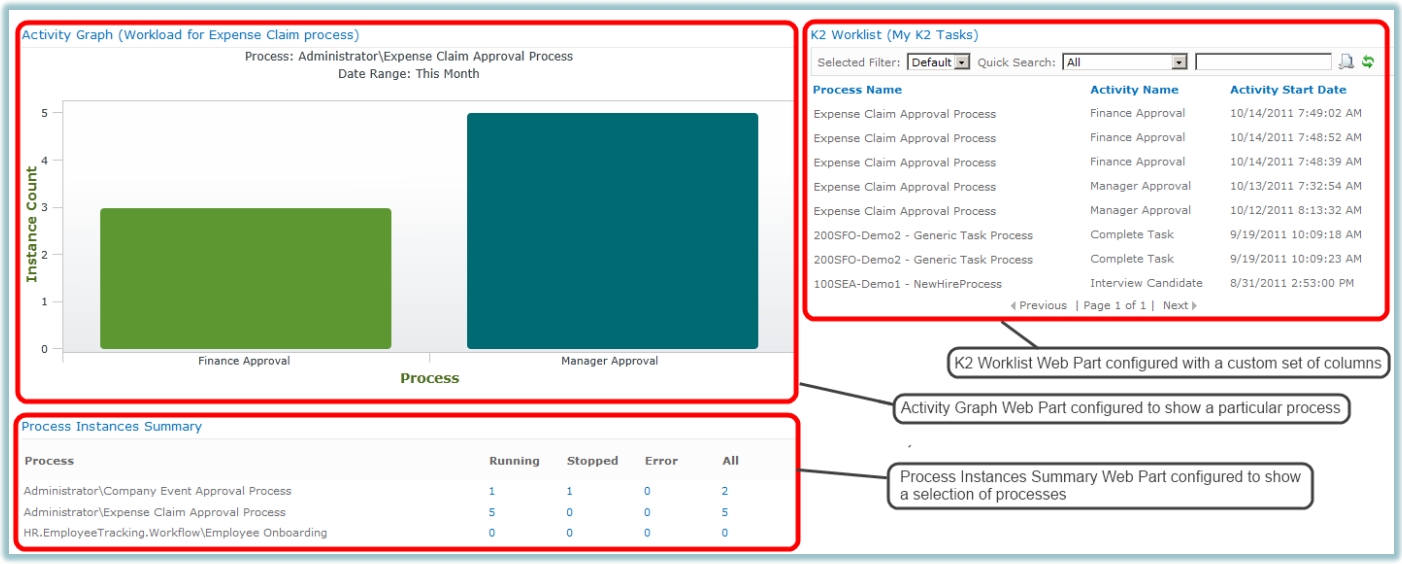
K2 Reporting Web Parts

K2 also provides a set of Reporting Web Parts, which can be used to create customized SharePoint sites that display reports about K2 processes. These Web Parts are particularly useful when configuring dashboard sites that allow process owners to see quick statistics about processes in the K2 environment.

Selecting the available K2 reporting WebParts when customizing a SharePoint page



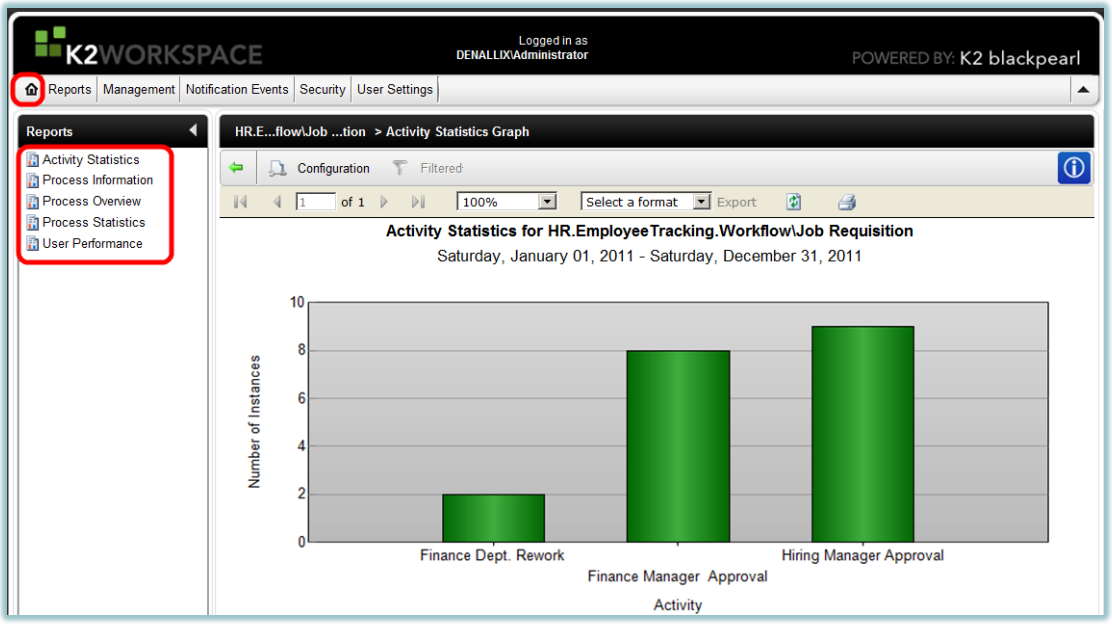
Using K2 WebParts to create a custom site layout with K2 process information



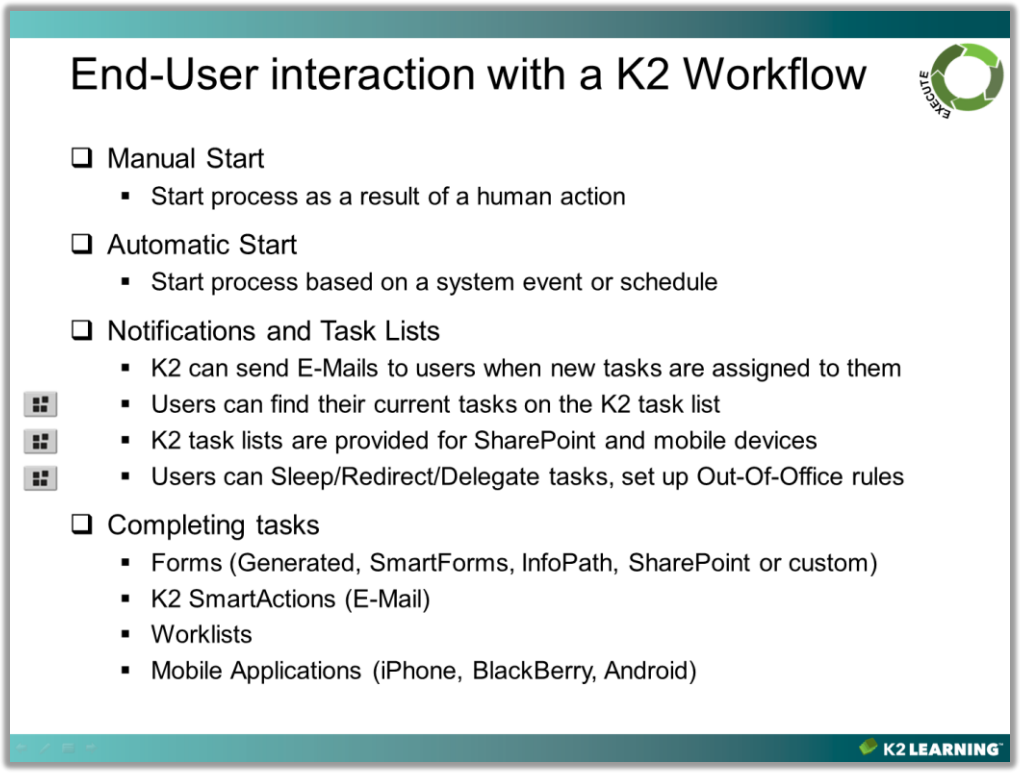
K2 Workspace

K2 Workspace can be used to run reports against K2 processes. This interface does not need to be configured with process settings like a process portal site; it will automatically include all the workflows where the current user has View or View Participate rights.

Running a process report in K2 workspace



End-User interaction with a K2 Workflow



Once a workflow and its supporting artifacts have been deployed the application is available to users with appropriate permissions. Users could interact with the workflow by starting a new process, receiving notifications about new tasks assigned to them, and using forms or interfaces to complete the tasks assigned to them.

Starting new Processes

Users with **Start** permissions on the process will be able to start the process, using the start mechanism selected when the process was created. For example, users could start the process by creating an item in a SharePoint list, by completing an InfoPath form and submitting the form, clicking a button in a K2 smartform or by performing some action in a custom application.

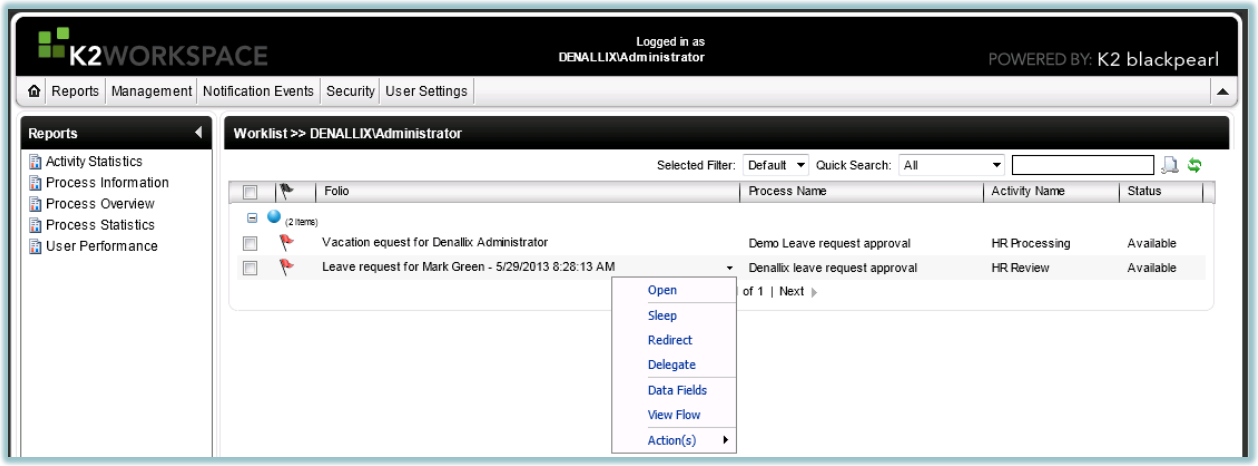
Notifications and Worklists

Once a process has started, the workflow participants can be notified about new tasks that are assigned to them via E-mail (if the process designer selected the option to send an E-mail notification to users for the task)

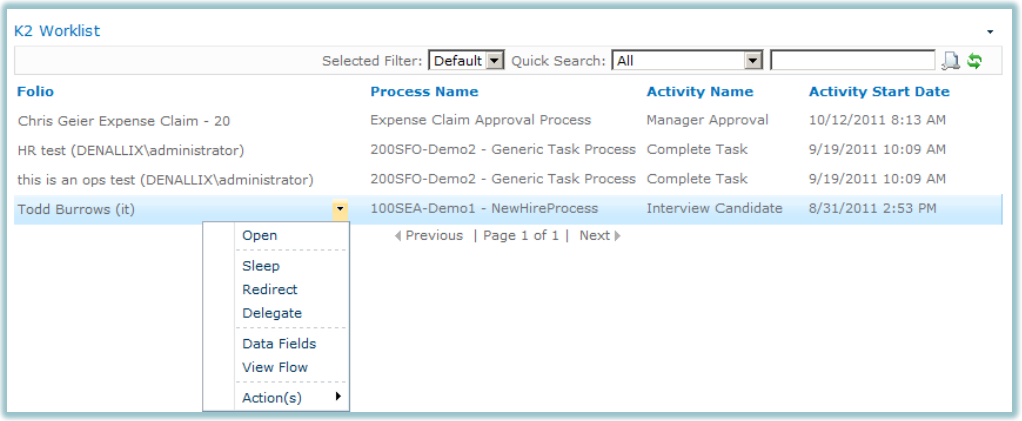
Users will also be able to locate their K2 tasks using K2 worklists. The K2 worklist is a centralized and unified view of all the K2 tasks that have been assigned to the current user. The user can open their K2 task list and see a list of all their tasks across all processes, libraries and sites, and the work list allows users to filter, group and sort tasks and also Open or Action their tasks.

The K2 worklist is located on the default landing page for the K2 Workspace. K2 also provides a SharePoint Web Part which can be added to any SharePoint page so that users can view their task list on, for example, the landing page of the organization's default SharePoint Portal site.

The K2 Worklist in the K2 Workspace



The K2 worklist Web Part in SharePoint



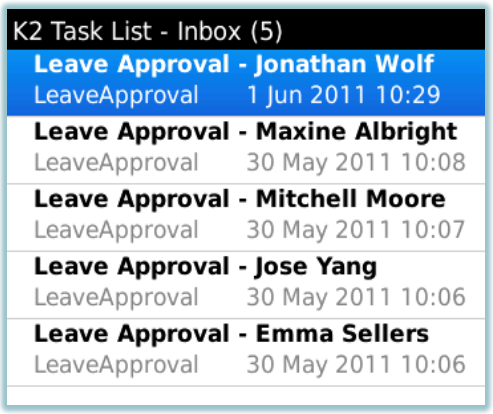
Note that K2 tasks are not the same as SharePoint tasks or Outlook tasks.

Organizations usually add the K2 Worklist Web Part to a landing page in the SharePoint environment, so that users can easily find all the K2 tasks that are currently assigned to them.

K2 also provides applications for iOS, BlackBerry and Android mobile devices to allow users to access their K2 worklists and complete their tasks using these devices.

Task List on iPhone

Task List on BlackBerry



Task List on Android

Task Permissions and Task Management

By default, if a user is assigned a task during a process, only that user has permission to complete the task. Users do not even need any security permissions on a process to complete their tasks, although typically all users are given at least View Participate rights on processes.

Using the task lists, user may perform other actions on their tasks as well, other than opening and completing their tasks. They may:

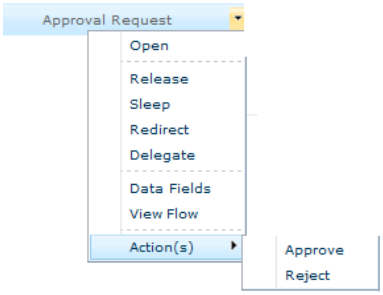
1. **Redirect** the task to another user (i.e., remove the task from my task list, and send the task to someone else)
2. **Delegate** the task to another user (i.e., create a copy of the task and send the copy to the other user. The task will still appear on my task list, and either one of us can complete the task)
3. **Sleep** the task for an interval (i.e., temporarily remove the task from my task list until the interval has passed, then show the task to me again)
4. **Release** the task back into the queue (i.e., if I am a member of a group that the task is assigned to, I can open the task, decide not to complete it, and release the task back to the group so that another group member can open the task)
5. Set up **Out-of-Office** rules (i.e., users can set up rules to forward tasks to other users while they are out of the office, similar to how you might set up an E-mail Out-of-Office rule)

Completing Tasks

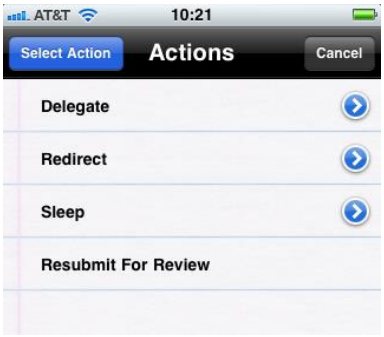
Users will open and complete tasks using the form that was specified by the process designer. Typically, opening a task will direct the user to the associated task form (either a generated page or an InfoPath form), where they will complete the form with the required information, and select an action out of the list of available Actions for the task, and submit the form to complete their task and let the process continue.

In some instances, users may also be able to complete tasks without needing to open the task form – depending on the Action and task type the user could select the action and complete the task directly from the K2 task list, as shown below:

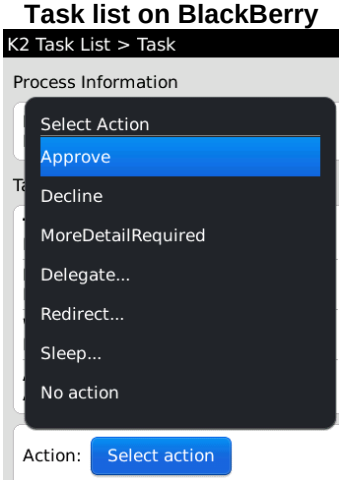
K2 Worklist



Task list on iPhone

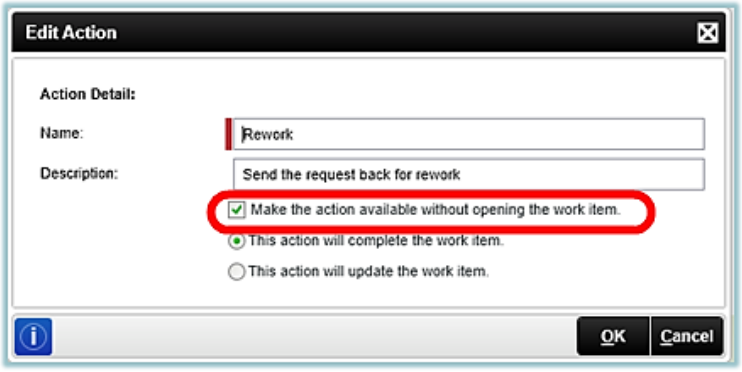


Task list on BlackBerry



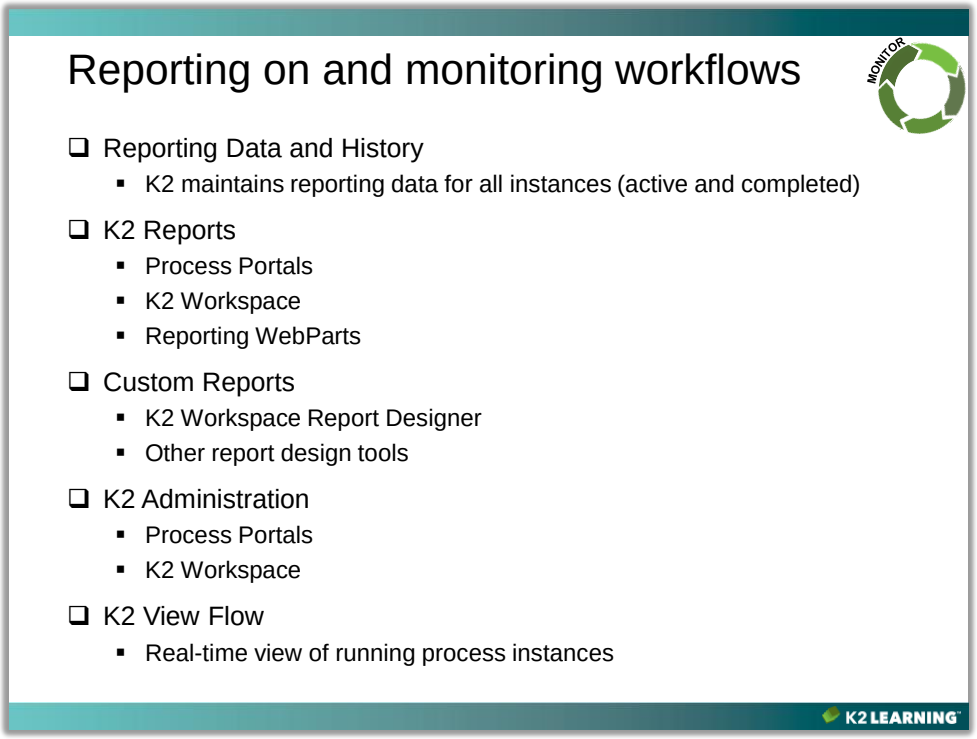
If K2 SmartActions are used, users can reply to the task notification email with their decision to complete their task item.

The ability to complete the task without opening the form requires that the workflow designer selected the **Make the action available without opening the work item** option when defining the action, as shown in the screenshot below.



Once a user has completed the task, K2 will evaluate the user's selected action, and then continue with the process if one or more of the available Outcomes' conditions has been met.

Reporting on and monitoring workflows



The final process lifecycle component is that of monitoring the application. Monitoring typically involves running reports to check the status and performance of processes, using real-time monitoring tools such as the K2 View Flow component to monitor the execution of processes and using administration tools to repair processes that have encountered errors or performing manual overrides in active processes.

Over time, monitoring could highlight bottlenecks or inefficiencies in processes, so the results gained from monitoring may result in application optimization changes that feed back into the Design > Assemble > Deploy > Execute > Monitor cycle. Remember that K2 process designs may go through several iterations during the solution life cycle. This is expected and typical of workflow solutions, since the process evolves with the business and may need to be adapted to address changing circumstances.

Reporting Data and History

The K2 workflow engine will maintain the status of active process instances and also keep reporting data for all instances of the process (active and completed) that have ever executed. K2 will automatically store data for process instances and the elements that make up the process instances, including steps, dates, users, decisions taken, audit history for specified datafields in a process and more. This extensive reporting data collection makes it possible to generate valuable reports that will help to monitor and manage process instances.

While K2 does not automatically clear or truncate this reporting data, it is possible to archive the reporting data to an external database. This is especially useful in large-scale environments where many hundreds of thousands of instances could be saved in the reporting data store.

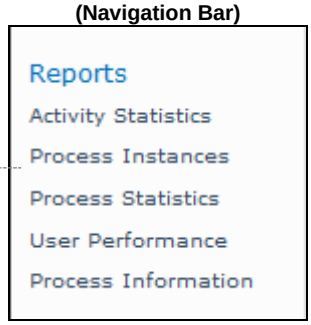
K2 process administrators can also delete the reporting data for a process version or process definition to recover space in the database or delete previous process versions that are no longer relevant.

K2 Reports

K2 provides several standard K2 reports. These reports are available both on the K2 Process Portal and the K2 Workspace.

Accessing the standard reports

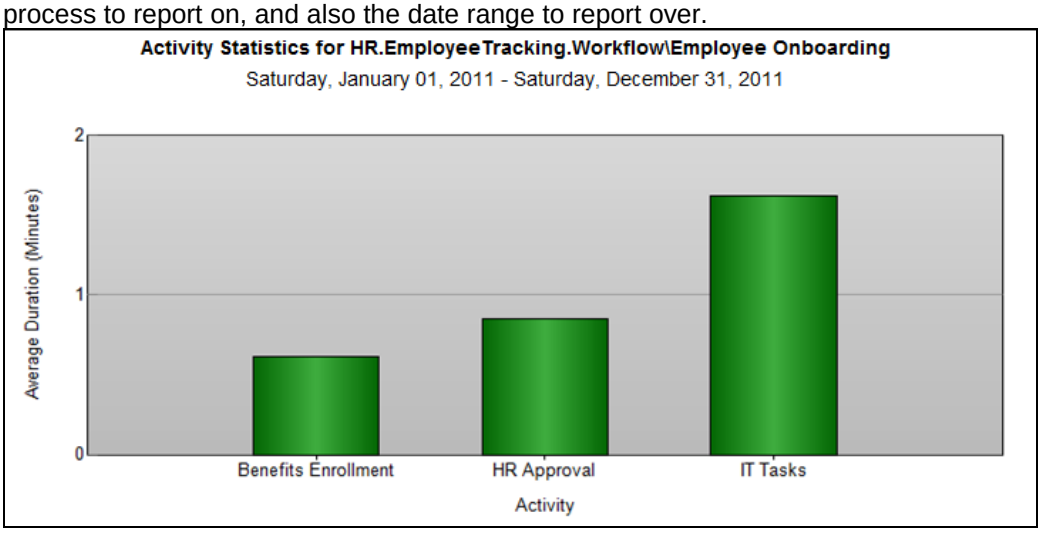
K2 Process Portal (Navigation Bar)

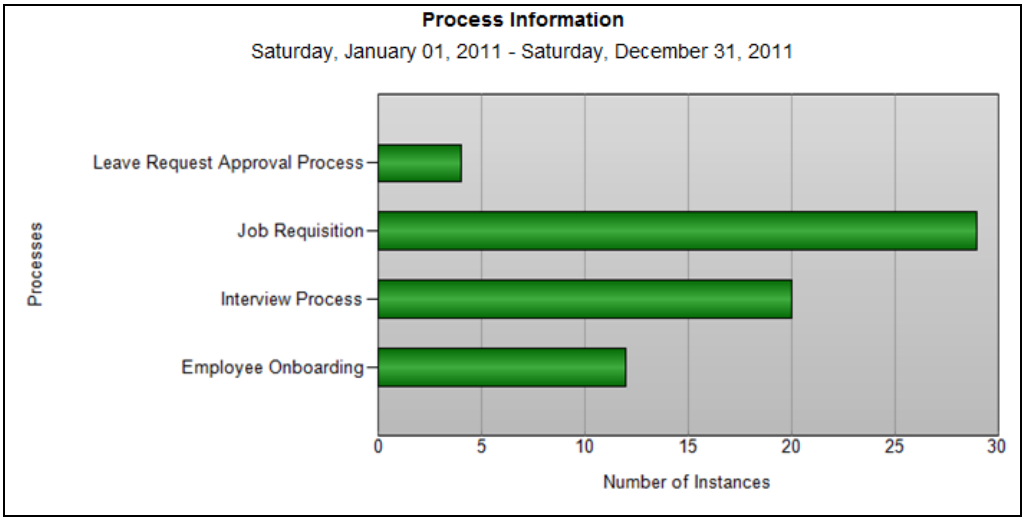


K2 Workspace

The standard reports that are provided with K2 include:

Report	Use Case and Example
Activity Statistics	Typically used to discover bottlenecks in a specific process. Users can select the process to report on, and also the date range to report over.
Process Information	Can be used to compare the use and duration of processes. This report is useful to compare the number of instances or duration of different processes in the organization over a given timeframe – perhaps the user is trying to determine which processes get used most often.



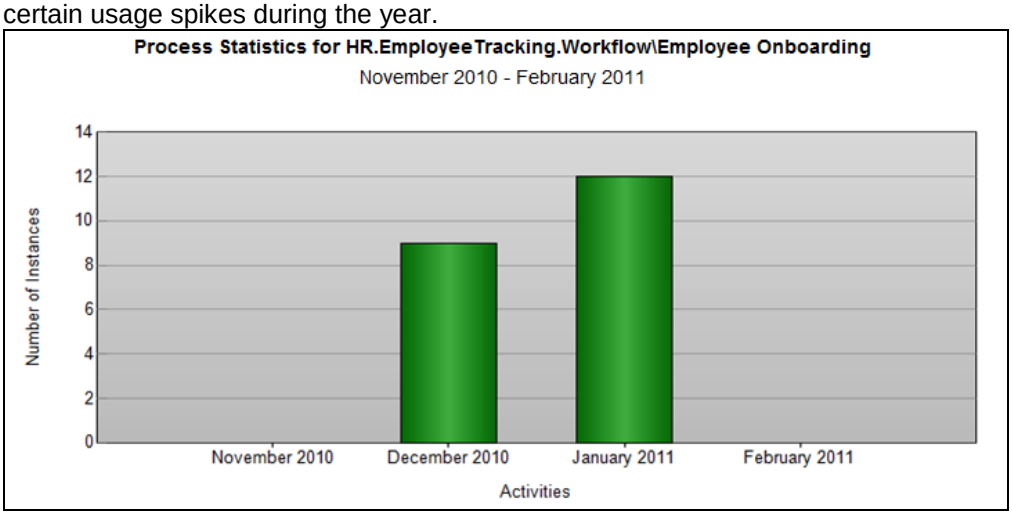


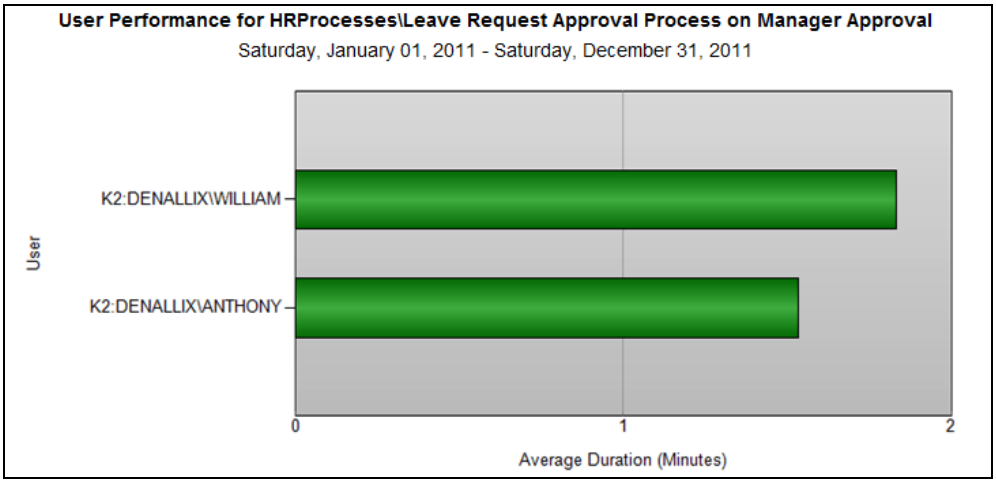
Report	Use Case and Example
Process Overview / Process Instances	Used to list all instances of a workflow and drill down into the workflow instance data and statistics. These reports are very useful for development, testing, troubleshooting and investigation of process instances
Process Statistics	Display a count or duration graph of a specific workflow grouped into selected intervals. For example, this report can be used to understand whether a particular process has certain usage spikes during the year.
User Performance	Display an “average duration” or “number of tasks competed” report for all users that performed a specific step of a particular process. This is useful in team-management scenarios (for example, identifying which users in a team do the most work) or in user performance management (for example, understanding how long a particular user takes on average to complete a specific task)

Process Instances

Process: K2Learning\100LAX–LAB2 – General Announcement Request

Process Folio	Originator	Status	Priority	Start Date	Finish Date	Duration
Second Test	K2:DENALLIX\ADMINISTRATOR	Active	Medium	8/11/2011 10:10:03 AM		00:02:06:57
First Test	K2:DENALLIX\ADMINISTRATOR	Completed	Medium	8/11/2011 8:39:29 AM	8/11/2011 9:14:19 AM	00:00:34:50
First Demonstration	K2:DENALLIX\ADMINISTRATOR	Active	Medium	8/11/2011 8:07:51 AM		00:04:09:09





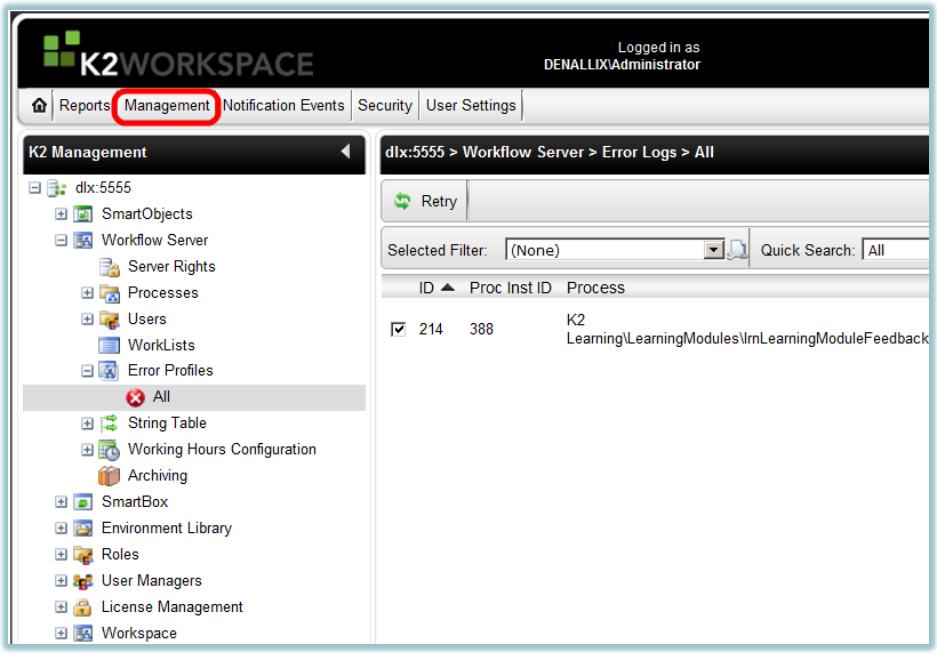
K2 Administration

Monitoring of workflows also involves administration of the processes to ensure that they are running as expected and to perform occasional manual intervention when required. This could include using the Error Retry and Repair

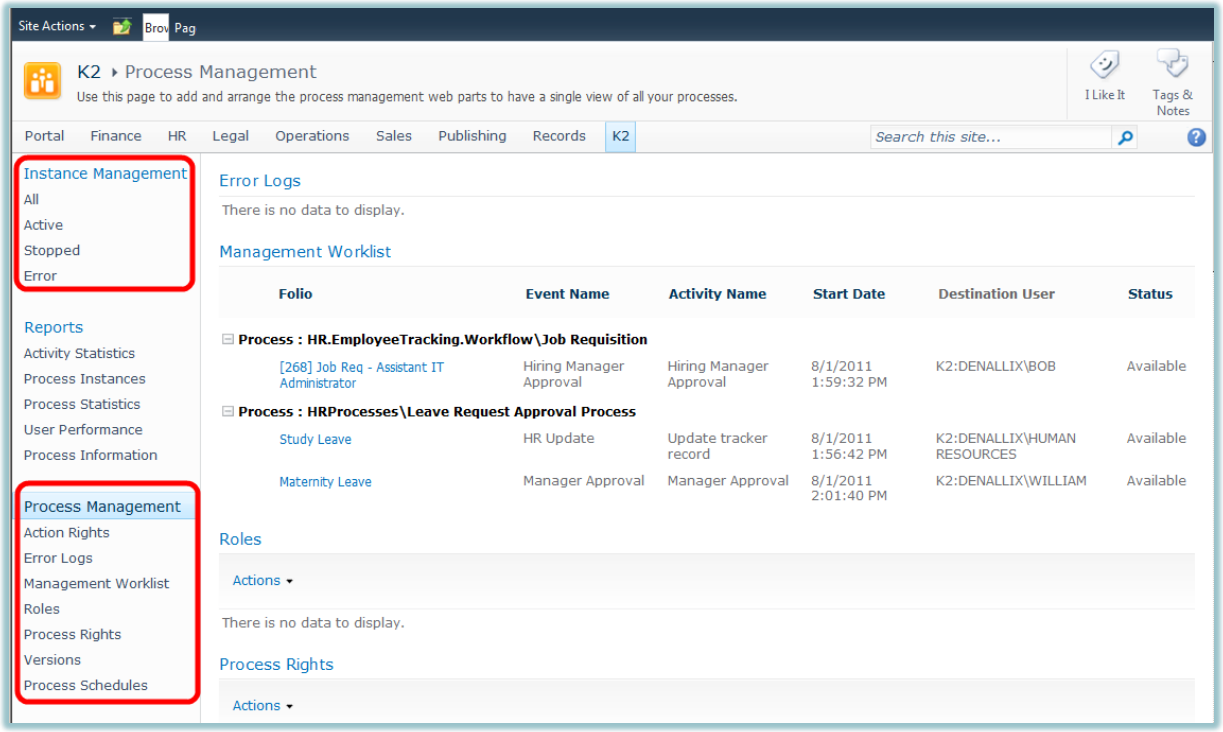
capabilities in K2 to repair processes that have gone into Error status, manually overriding task assignment, or manually forcing a process to go back to a previous step of the workflow.

K2 Administration is discussed in more detail in other learning modules like **200.CPH Process Portals and Web Parts** and **200.DUB K2 Workspace - Administration**

Administration tools in K2 Workspace



Administration tools in a process portal site

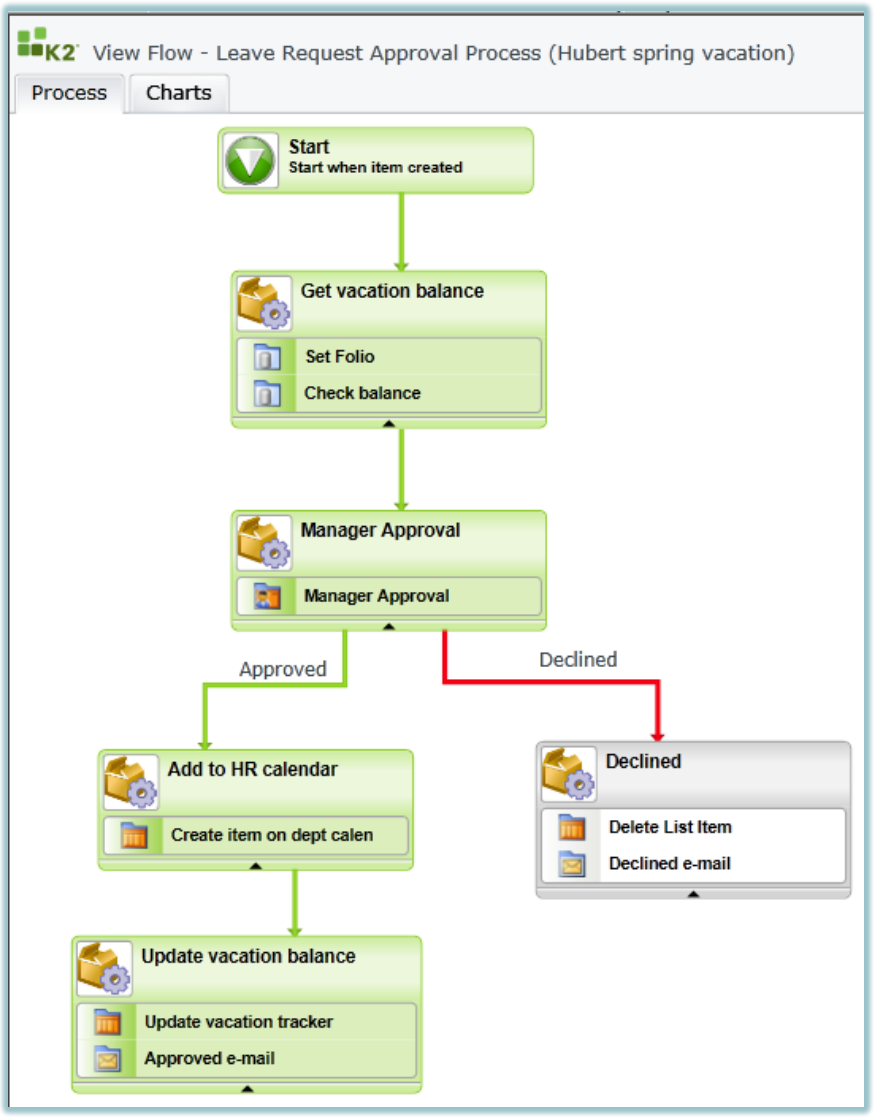


K2 View Flow

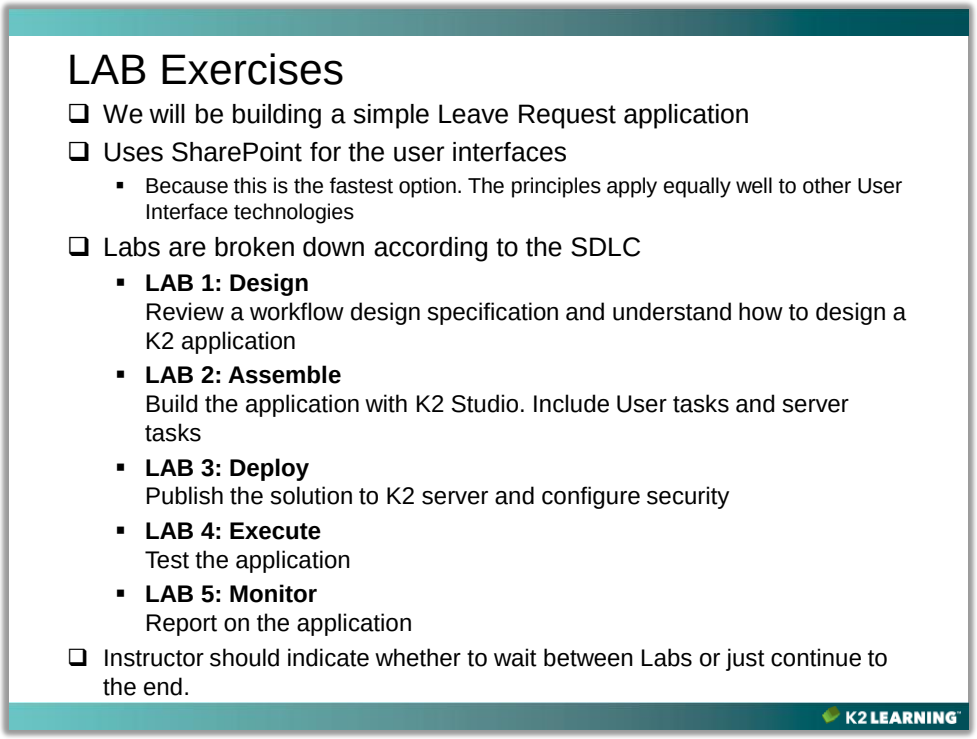
The K2 View Flow Component is an extremely useful tool to monitor a process' execution in real time, or to understand the path that was taken for a specific process instance.

Consider the example of the view flow below. The Red Paths in the View Flow indicate that the path was not followed, while the green paths and steps indicate the path that was followed.
In this particular example, the View Flow report shows that the manager approved the leave request. You can click any of the steps in the workflow to drill down and see more details about the step, for example who approved the Leave request and when they completed the task.

The K2 View Flow Report



LAB Exercises

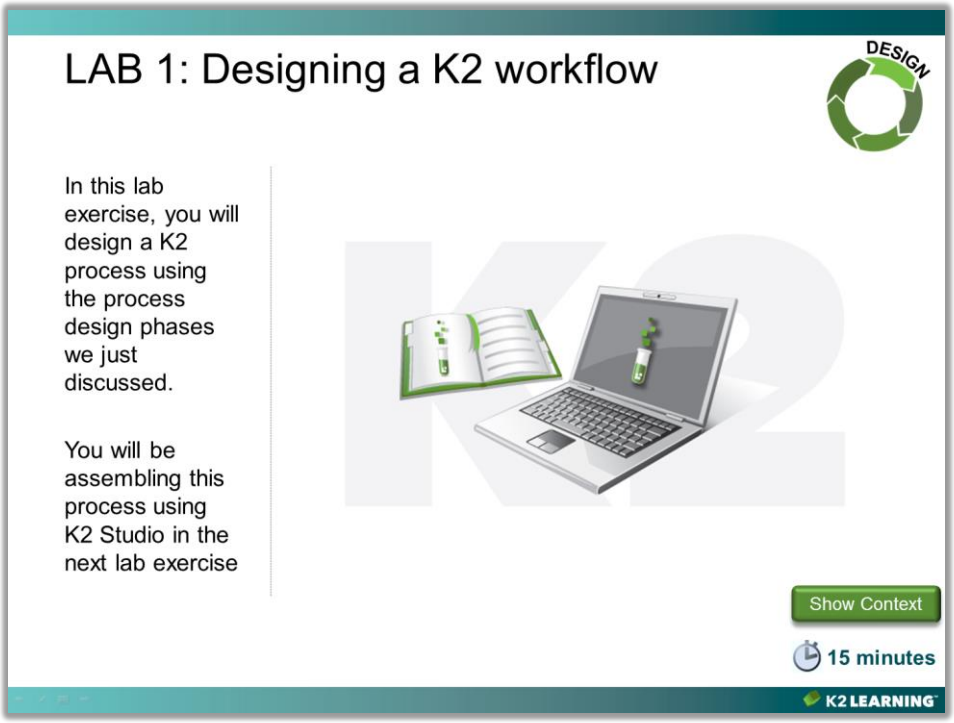


Now that we have covered the basics of designing, assembling., deploying, executing and monitoring workflows, it is time to put that knowledge into action and create a K2 application. In the following series of lab exercises you will create a new K2 leave request and approval application.

Note: To keep things simple, we will base the application on user interfaces in SharePoint because this is a quick way to create K2 applications. We will dive into other user interface technologies later. For now, we want you to become familiar with the basic principles of building K2 applications and K2 workflows rather than focusing on the user interfaces.

The labs are broken down into the 5 main phases of the SDLC: Design, Assemble, Deploy, Execute and Monitor. Your instructor will determine whether you should wait for the other participants to complete the lab exercise or whether you can continue with each lab exercise until the end.

LAB 1: Designing a K2 Process



Objective

This exercise illustrates how to design and gather specifications for a K2 process in preparation of assembling the process using the K2 Studio design tool.

This lab will cover the following aspects of process design.

- How to gather requirements for a K2 process before starting the design phase
- What questions to ask when gathering requirements and designing the process
- Capturing the design specifications

Duration

This lab has 8 steps and should take around 15 minutes to complete.

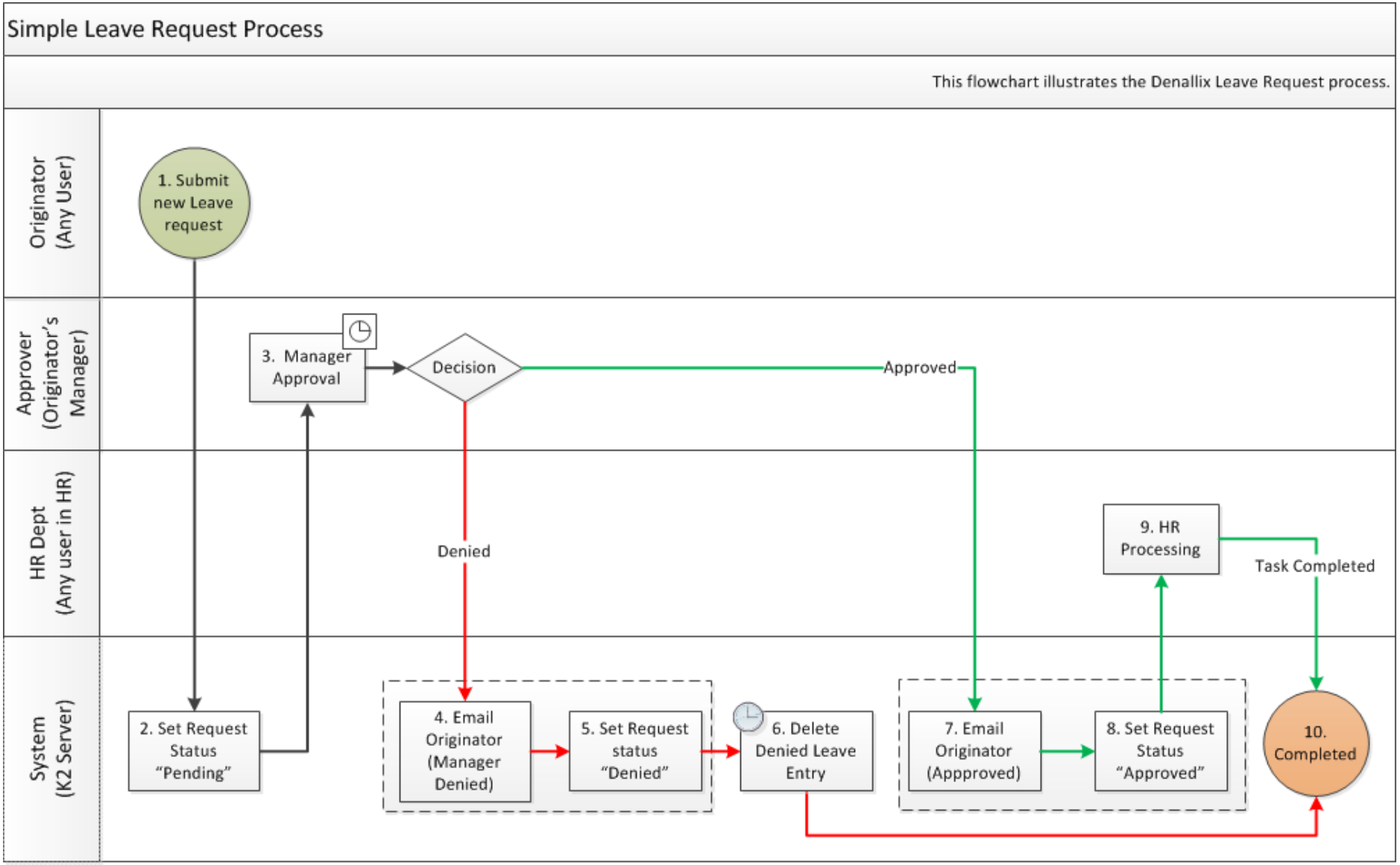
Context

Denallix wants to implement a very basic Leave Request approval process. The primary purpose of this process is to replace the inefficient paper-based process currently in place. One of the major requirements managers have asked for is a way to see all leave requests on a calendar, so that they can understand the impact of approving a particular leave request. It would also be good to have a global leave calendar list so that anyone can see who is on leave, when,

As the first step to implementing this process, we should design the process using the design phases discussed in the learning topics we just covered.

This lab is a group discussion activity: discuss and answer the questions as a group. The answers to each question are provided in the training material at **C:\K2 Learning\100.IAH\LAB 1\ 100.IAH LAB 1 - Answers to questions.docx**

For the purposes of this exercise, assume the following flowchart describes the process:



Additional notes and practical application

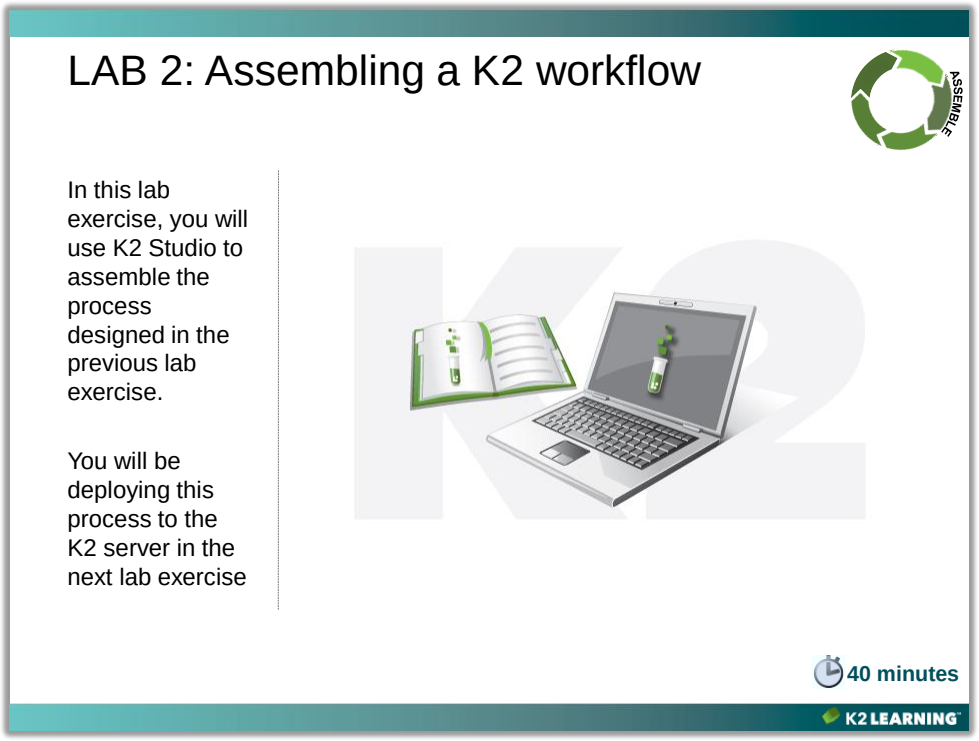
You may not always know which of the K2 Design tools to use when gathering the process specifications.

Some workflows may work well when implemented in the K2 Designer for SharePoint, and some can only be implemented in K2 Studio or K2 for Visual Studio. Another aspect you should bear in mind is what user interfaces will be used to during the workflow.

The best starting point is to use a process specification template like the example we have used to gather the process requirements. Once the process requirements are understood, you can make the appropriate selection of design tool and forms technology.

Gathering requirements and specifications for K2 solutions is discussed in much more detail in the learning module
200.IDO Gathering specifications and requirements for K2 projects
The document template used to capture the sample process' specifications is provided along with the 200.IDO learning module. This template can be freely re-used, modified or extended for your own K2 projects.

LAB 2: Assembling a K2 Workflow



Objective

In this lab exercise, you will be assembling a basic K2 process using the K2 Studio design environment. The lab will cover the following concepts:

- Creating workflows in K2 Studio
- Associating a process with a SharePoint workflow
- Updating and deleting SharePoint list items during a process
- Configuring User Tasks
- Sending E-mails during a process
- Using Start Rules to delay steps in a process
- Configuring simple escalations

Duration

This lab has three parts and should take around 40 minutes to complete.

1. Review the prerequisite items required to implement the process.
2. Create the new process using K2 Studio
3. Assemble the process by adding and configuring Activities, Events and Lines to the process

Context

In this lab, you will assemble the simple Leave Request Approval process that we discussed in the previous Lab exercise. For reference, the following document contains the design specification for the workflow:

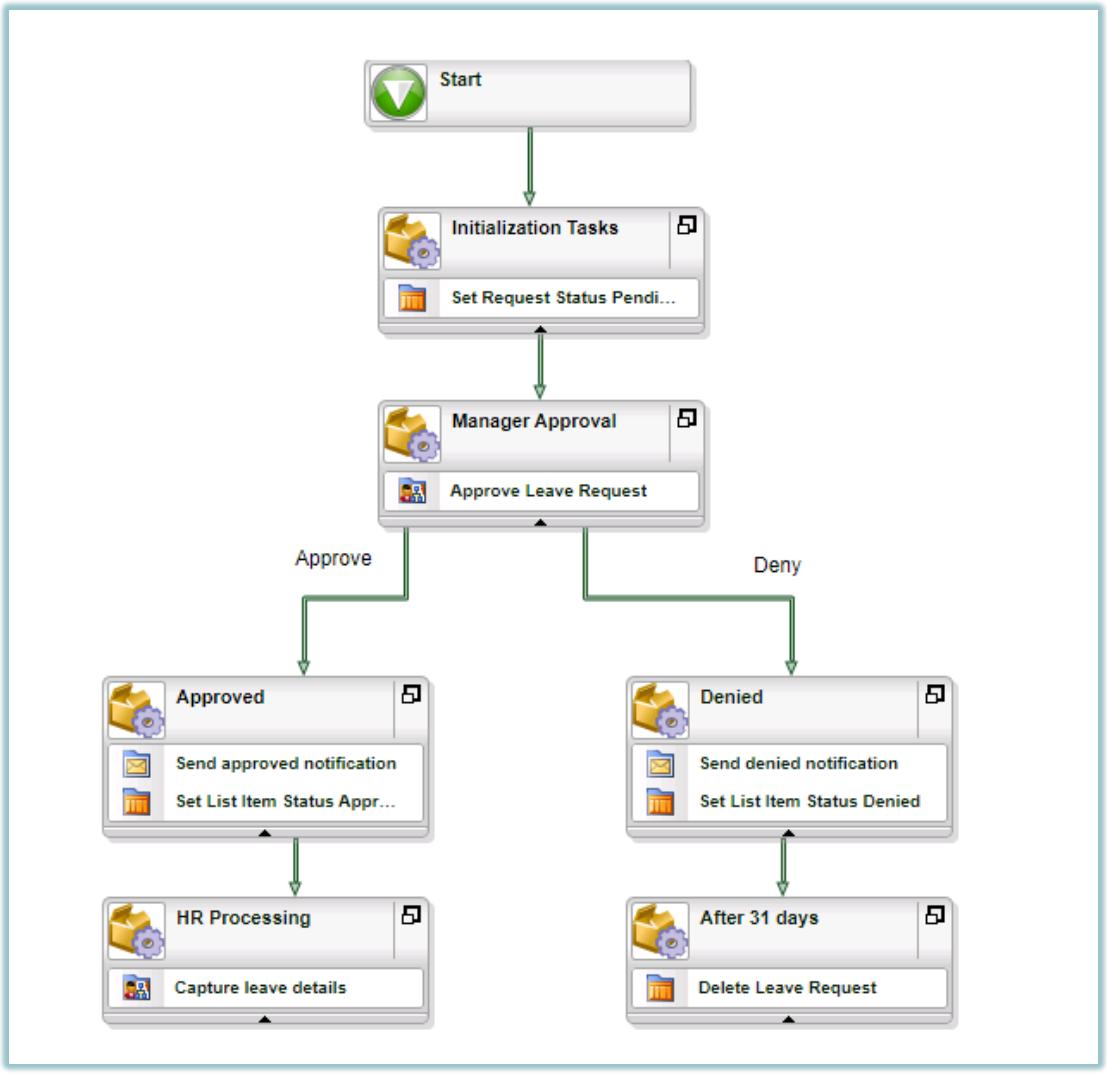
C:\K2 Learning\100.IAH\LAB 1\100.IAH LAB 1 - Process Design Document.docx

You should be familiar with the contents of this document so that you understand why certain steps are performed in the lab exercise.

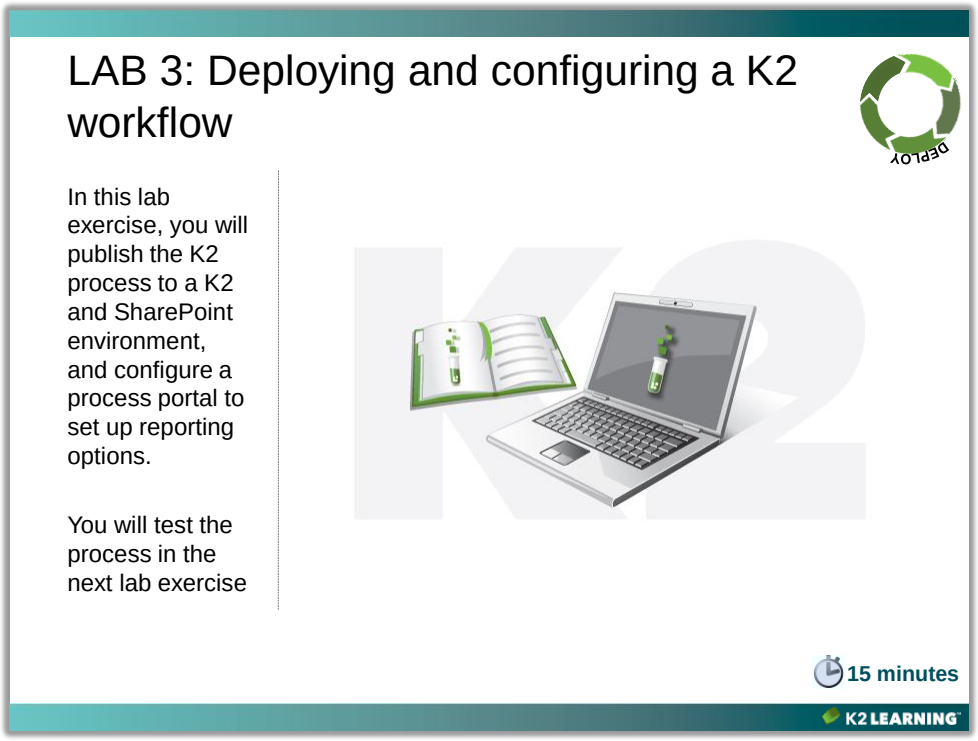
From an implementation perspective, the following decisions were made after evaluating the design document:

- We will use a calendar list in SharePoint to capture the leave request, and modify the status of the item in SharePoint as the workflow executes.
- The workflow will start when a new item is added to this calendar.
- The calendar item in SharePoint item will contain all the data necessary for the process and user interfaces
- Since the user interfaces are simple, we will use the SharePoint Workflow Integration Client Event to generate user interfaces for the workflow.

The eventual workflow solution should look something like this:



LAB 3: Deploying and configuring a K2 workflow



Objective

In this lab exercise, you will be deploying the K2 process created in the previous Lab exercise, configuring an existing K2 Process Portal to include the newly deployed process and setting process security rights for a deployed workflow. The lab will cover the following concepts:

- Deploying a K2 process to a K2 server
- Adding processes to a K2 Process Portal for reporting and administration purposes
- Setting process security using a Process Portal site

Duration

This lab has 3 parts and should take around 15 minutes to complete.

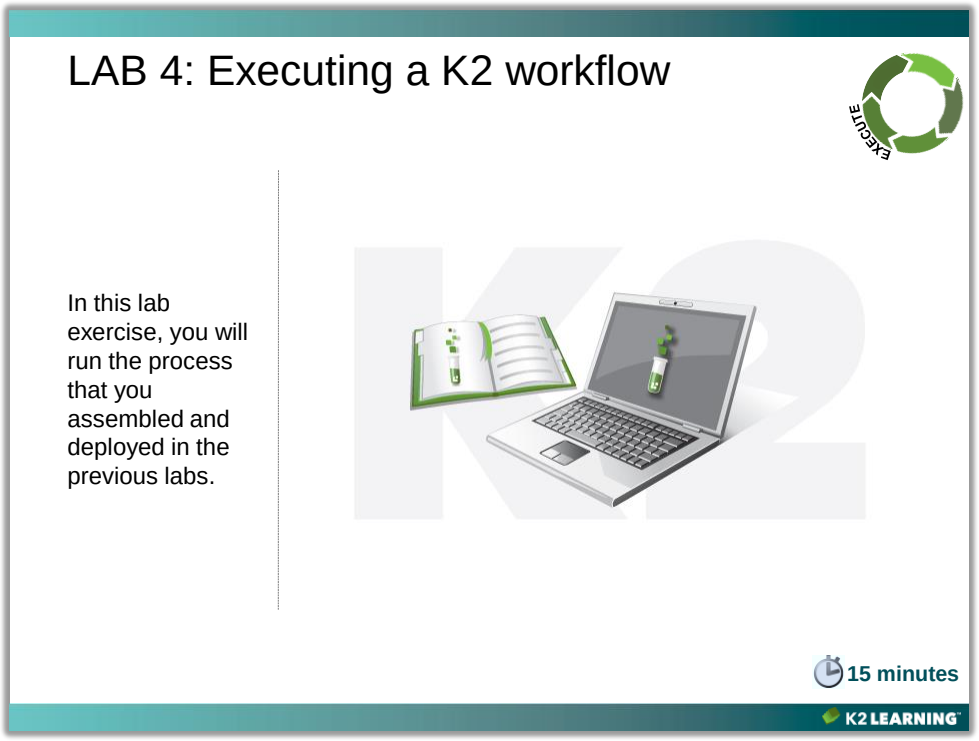
1. Deploy the process to a K2 server
2. Configure a K2 Process Portal to include the deployed process
3. Set the process security

Context

Now that the simple Leave Request process is assembled, we need to publish the process to a K2 server. As part of the deployment process, K2 may also create additional integration points. In this particular case we used a SharePoint Workflow Integrated process, which means that K2 will perform additional tasks that will associate the K2 process with various SharePoint items at deployment time.

We will use a SharePoint site based on the K2 Process Portal template to report on the workflow, so we need to add the process to a Process Portal site. Finally, we need to give users permissions on the workflow according to the process security specified in the process design document.

LAB 4: Executing a K2 workflow



Objective

In this lab exercise, you will be running the simple Leave Request process that you created and deployed in the previous labs. The lab will cover the following concepts:

- Starting a new K2 process by creating a SharePoint item
- Locating tasks using the K2 task list
- Completing tasks with E-mail and with generated forms

Duration

This lab has 5 steps and should take around 15 minutes to complete.

Context

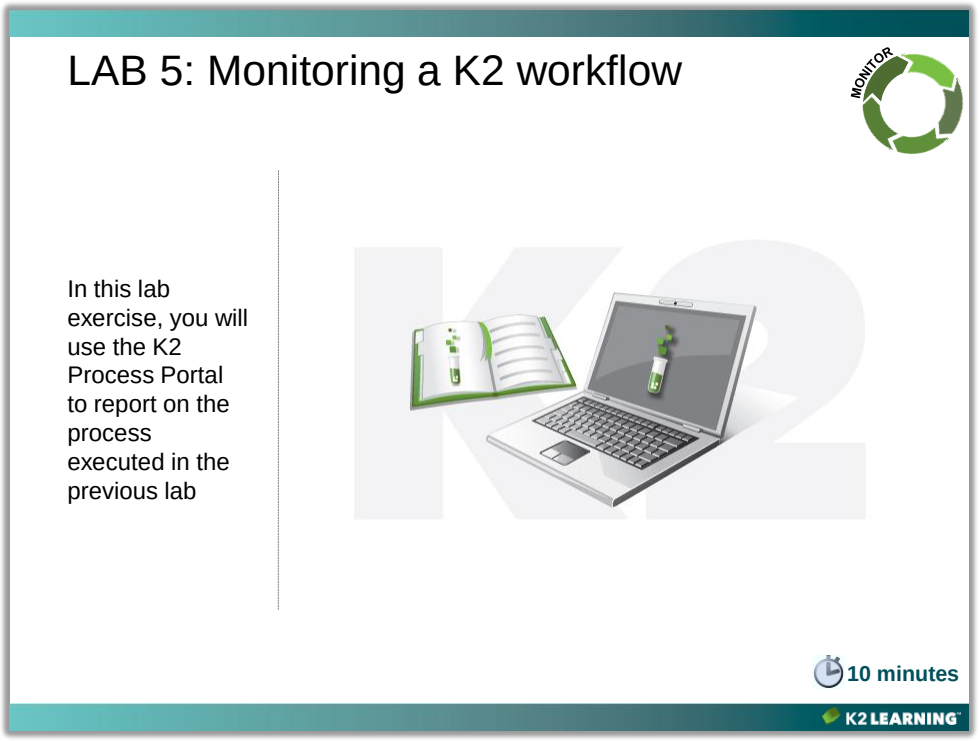
The simple Leave Request process has been deployed to the K2 and SharePoint environment, and is now ready for use. Remember that this process starts when a user creates a new item in the **100.IAH Leave Requests** list. As the process executed, K2 will perform server tasks or generate tasks for users, depending on the workflow design.

Practical Application and Additional Notes

This was a very simple process. In the real world, it would be useful to format dates and times properly in the notification e-mails, to allow the manager to enter comments and send the request back for rework, to include the manager comments in the rejection E-mail and so on. You will learn how to do all this and more in lab exercises from other Learning Modules; for now, the main purpose was to introduce the main parts of the workflow development cycle (**Design**, **Assemble**, **Deploy** and **Execute**. **Monitor** will be discussed in the next learning module) without going into too much detail or complexity.

It is certainly possible to create very complex workflows, user interfaces and reports in K2 solutions, and after completing more of the K2 learning modules you will be capable of building more advanced workflow solutions. For now, it's more important that you understand the basics on how to approach workflow design and assembly.

LAB 5: Monitoring a K2 workflow



Objective

In this lab exercise, you will report on the Leave Request process instance(s) started in the previous lab exercise. The lab will cover the following concepts:

- Using the process instances report to drill down into a process instance's history
- Using the K2 View Flow report to view the process instance's flow

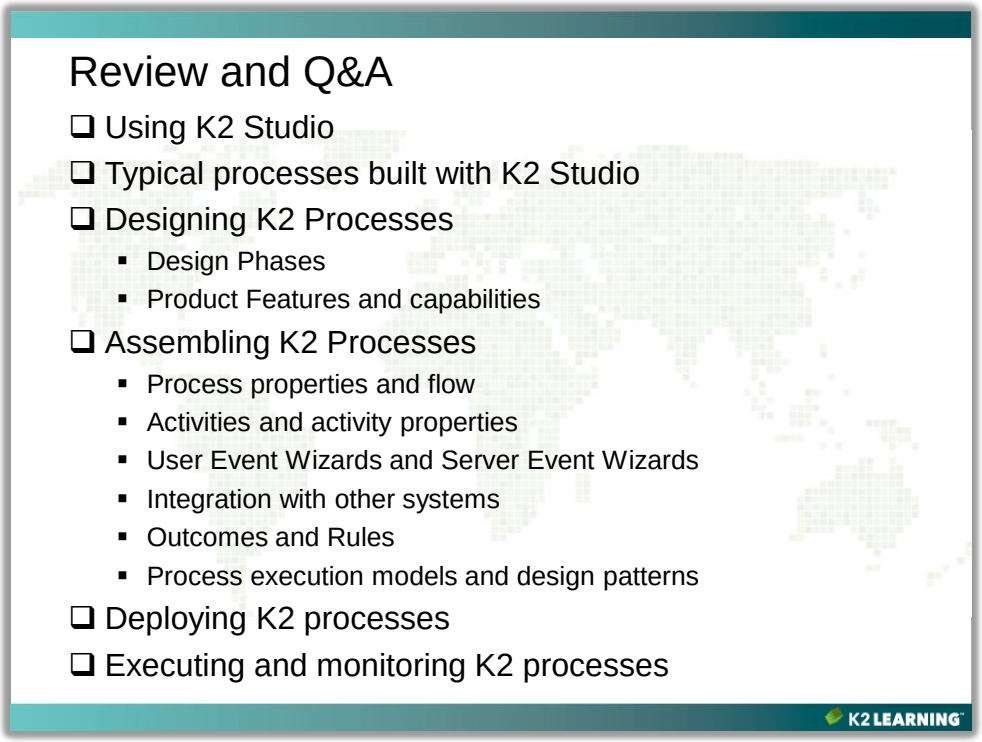
Duration

This should take around 10 minutes to complete.

Context

Now that we have some instances of the Leave Request approval workflow, you can use the a K2 process portal site to report on the workflow instances. (You can also use the K2 Workspace to report on workflows, if you prefer)

Review and Q&A



This was a rather long and very theoretical module, but it was important to establish a sound understanding of the fundamental aspects and approaches used when designing, assembling, deploying, executing and monitoring K2 workflows.

Let's review the main topics covered in this module:

- Using K2 Studio
- Typical processes built with K2 Studio
- Designing K2 Processes
    - Design Phases
    - Product Features and capabilities
- Assembling K2 Processes
    - Process properties and flow
    - Activities and activity properties
    - User Event Wizards and Server Event Wizards
    - Integration with other systems
    - Outcomes and Rules
    - Process execution models and design patterns
- Deploying K2 processes
- Executing and monitoring K2 processes

Consider how these topics apply to your requirements; how might you apply an existing process requirement using the knowledge you gained in this module? How would users initiate and interact with workflows? If you are using SharePoint in your environment, how could you use K2 to simplify the SharePoint administration and maintenance, and what kinds of business processes could you implement using SharePoint lists and Libraries?

This is also your opportunity for asking questions to clarify any of the topics covered during this module. Following modules will build on the knowledge you gained in this module, so if you are unclear about a topic, please do ask the instructor to explain or elaborate.

Additional Resources

The following table lists additional resources that support or supplement the information provided in this module:

Resource	Location and Notes
Video: K2 SmartActions	http://www.k2.com/videos/4097 *Duration: 20 minutes* *This video discusses and demonstrates K2 SmartActions*
Video: Standard K2 Reports	http://www.k2.com/videos/587efac5 *Duration: 3 ½ minutes.* *This video demonstrates the standard reports included in the K2 Process Portal*
Mobile Applications: How to work with Mobile Task Lists	http://help.k2.com/en/KB001252.aspx *Describes setting up access for mobile task list applications for BlackBerry and iPhone, and how to use the applications on mobile devices.*
K2 blackpearl Best Practices	http://help.k2.com/en/KB000352.aspx *Discusses best practices and approaches to follow when assembling K2 solutions*
How to use K2 deployment packages	http://help.k2.com/en/kb000188.aspx *Describes how to use .msbuild deployment packages in K2 to publish project artifacts*

The first word spoken from the moon on July 20, 1969 was “Houston”. Houston is of course home of the NASA Johnson Space Center, focal point for U.S. manned space flight program