

# AARE – Workflow Management System

## White Paper

netstyle.ch GmbH  
Terrassenweg 18  
3012 Bern  
Switzerland

December 14, 2005

### **Abstract**

AARE is a powerful and extensible workflow management system. It provides a feature-rich workflow engine allowing one to design complex business workflows, run and control them, collect necessary data through sophisticated electronic forms and automatically notify responsible people. AARE has been successfully integrated with a document management system, enabling one to attach metadata annotated documents to workflows and activities. AARE is an entirely web-based application, taking advantage of state-of-the-art Web 2.0 technology.

## Contents

<b>1</b>	<b>Overview</b>	<b>4</b>
1.1	Preface . . . . .	4
1.2	Introduction . . . . .	4
1.3	Workflow Engine Features . . . . .	5
1.3.1	Sequential Routing . . . . .	5
1.3.2	Conditional Routing . . . . .	5
1.3.3	Parallel Routing . . . . .	5
1.3.4	Synchronizing Merge . . . . .	6
1.3.5	Loops and Multiple Instances Without Synchronization	6
1.3.6	Multiple Instances Without a Priori Runtime Knowledge	6
1.3.7	Embedding Sub-Workflow . . . . .	6
1.3.8	Embedding Sub-Workflow Choice . . . . .	6
1.3.9	Implicit Termination . . . . .	7
1.3.10	Other Types of Activities . . . . .	7
1.4	Workflow Systems Compared . . . . .	7
<b>2</b>	<b>Workflow Design</b>	<b>7</b>
<b>3</b>	<b>Workflow Execution</b>	<b>13</b>
<b>4</b>	<b>Workflow Reporting</b>	<b>16</b>
4.1	Interface . . . . .	17
4.1.1	Title . . . . .	18
4.1.2	Selectors . . . . .	18
4.1.3	Conditions . . . . .	19
4.1.4	Fields . . . . .	19
4.2	Views . . . . .	19
4.2.1	Report . . . . .	20
4.2.2	Export . . . . .	20
4.2.3	Pie Chart . . . . .	20
4.2.4	Line Chart . . . . .	20
<b>5</b>	<b>Document Management System</b>	<b>20</b>
5.1	Document Editor . . . . .	20
5.2	Editing Document Types . . . . .	21
5.3	Searching Documents . . . . .	23

## List of Figures

1	Different workflow systems compared I . . . . .	8
2	Different workflow systems compared II . . . . .	9
3	General workflow properties . . . . .	10
4	General activity properties . . . . .	10
5	Define the data collection form of the activity . . . . .	11
6	Preview of possible form-field types . . . . .	11
7	Define activity conditions . . . . .	12
8	Define activity transitions . . . . .	12
9	Workflow activity report . . . . .	13
10	Workflow graph . . . . .	13
11	Workflow diagram . . . . .	14
12	Select workflow to run . . . . .	14
13	Start workflow . . . . .	15
14	Completing the first activation . . . . .	15
15	Next activation . . . . .	16
16	Report about running and completed activities . . . . .	16
17	Log of the changes within the workflow . . . . .	17
18	Running workflow graph . . . . .	17
19	Running workflow diagram . . . . .	18
20	Gantt diagram . . . . .	18
21	Workflow reporting interface . . . . .	19
22	Pie chart – Completion status of workflows . . . . .	20
23	Line chart – Started workflows . . . . .	21
24	Document editor . . . . .	22
25	Document type editor . . . . .	22
26	Searching documents . . . . .	23

# 1 Overview

## 1.1 Preface

This White Paper provides an overview of AARE, a sophisticated, entirely web-based, workflow management system (WfMS).

AARE is composed from a workflow engine, the core of the system, which provides the facility to design new workflows and to execute them. Upon this kernel component of AARE, the user interface and additional, customer-specific behavior is implemented on a per project basis to allow an optimal solution for each customer.

This document is structured as follows. The next section 1.2 introduces the relevant terms used in the context of WfMS. The subsequent section 1.3 exhibits the key features of the AARE workflow engine and presents a comparison with other commercial business WfMS. The next chapters illustrate by means of a concrete project of AARE the definition of workflows (workflow design) in Chapter 2, the workflow execution in chapter 3, the reporting engine in chapter 4. Finally chapter 5 shows the document management system (DMS) that has been integrated with AARE.

This White Paper does not discuss features that are usually very customized for most projects, e.g., the dashboard, a work-list for each user that allows him to quickly identify current tasks, sending notifications of new and overdue activities by email, web-service interfaces etc.

## 1.2 Introduction

**Workflow System.** Workflow systems are defined as “systems that help organizations to specify, execute, monitor, and coordinate the flow of work cases within a distributed office environment”. Workflow diagrams rely on the use of standardized graphical notations to describe workflow structures. AARE is built from the following two base components:

1. The *workflow design* component enables administrations and analysts to design and define processes and activities, analyze and simulate them, and assign them to people or groups.
2. The *workflow execution* is sometimes called the run-time system. It consists of the execution interface seen by end-users and the workflow engine, an execution environment which assists in coordination and performing the processes and activities.

**Workflow.** A workflow is the operational aspect of a work procedure: how activities are structured, who performs them, what their relative order is, how they are synchronized, how information (e.g., electronic forms) flows to support the activities and how they are being tracked.

**Activity.** An activity is part of a workflow definition to accomplish a specific job. In AARE, an activity includes a form definition to collect data, form conditions to validate the entered data, transitions to subsequent activities, transition conditions to decide which following activities should be triggered on completion, and other settings, such as responsible users, expected duration, and documentation.

**Activation.** An activation is an instance of an activity as part of a running workflow. Since there can be splits and loops within a workflow definition, multiple activations of different or the same activity can exist at the same time in a running workflow.

**Standards.** The Workflow Management Coalition<sup>1</sup> defines several standards, e.g., a Reference Model, a generalized target architecture driving the development of AARE and many other production workflow solutions. The goal of the model is to provide a standard for interoperability among the major workflow subsystems.

### 1.3 Workflow Engine Features

The AARE workflow engine supports a wide variety of well-known business workflow patterns listed below. Other patterns can be easily achieved by combining the existing ones or by extending the workflow engine.

#### 1.3.1 Sequential Routing

This is the most trivial situation: an activation in a workflow process is enabled after the completion of another activation in the same process.

#### 1.3.2 Conditional Routing

A point in the workflow process where, based on a decision or workflow control data (conditions can be based on the data entered by the user in the electronic form), a number of branches are chosen.

#### 1.3.3 Parallel Routing

A point in the workflow process where a single thread of control splits into multiple threads of control which can be executed in parallel, thus allowing activations to be executed simultaneously or in any order. The workflow engine exhibits implicit split semantics, that is, it does not need special routing constructs: each activity can have more than one outgoing transition and each transition has associated conditions. To achieve parallel execution the workflow designer makes sure that multiple conditions associated with

---

<sup>1</sup><http://www.wfmc.org/>

outgoing transitions of the node evaluate to true, what is typically achieved by leaving the conditions blank.

#### **1.3.4 Synchronizing Merge**

A point in the workflow process where multiple paths converge into one single thread. If more than one path is taken, synchronization of the active threads takes place. This means that the activity where several paths merge is only run when all previous active paths were completed. If only one path is taken, the alternative branches automatically re-converge without synchronization. In contrast to many other workflow management systems, AARE achieves this property through sophisticated logic. Hence the designer of the workflow definition does not need to explicitly define any conditions for merging which makes the workflow system very flexible and intuitive to use.

#### **1.3.5 Loops and Multiple Instances Without Synchronization**

A point in a workflow process where one or more activities can be done repeatedly. Within the context of a single case multiple instances of an activity can be created, i.e., there is a facility to spawn off new threads of control. Each of these threads of control is independent of other threads. Moreover, there is no need to synchronize these threads.

#### **1.3.6 Multiple Instances Without a Priori Runtime Knowledge**

For one case an activity is enabled multiple times. The number of instances of a given activity for a given case is not known during design time, nor is it known at any stage during runtime, before the instances of that activity have to be created. While some of the instances are being executed or already completed, new ones can be created.

#### **1.3.7 Embedding Sub-Workflow**

The workflow engine supports explicit embedding of other workflows. The outgoing transition of a sub-flow is activated when the subflow terminates. This allows one to decompose a workflow into sub-flows e.g., for reuse in other workflow definitions.

#### **1.3.8 Embedding Sub-Workflow Choice**

If an embedded sub-workflow is not known at design time a special activity can be created which allows one to choose a workflow at runtime. The chosen workflow is then executed like an explicitly embedded sub-workflow.

### 1.3.9 Implicit Termination

A given workflow or sub-workflow should be terminated when there is nothing else to be done. In other words, there are no active activities in the workflow and no other activity can be made active.

### 1.3.10 Other Types of Activities

The architecture of AARE is designed to be flexible in respect to extensions, e.g., to support new kind of activities. For example, activities that do not need human interaction but when being executed would perform some task, such as interacting with another web-service. Also, some patterns in the subsequent section are not being implemented because they were not needed up until now, although they would be supported by the architecture.

## 1.4 Workflow Systems Compared

In Figure 1 and Figure 2 a comparison of different commercially available workflow systems is shown. Features shown are directly supported by the given engine, without creating spaghetti-diagrams or coding. The feature matrix has been extracted from *W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros: Workflow Patterns. QUT Technical report. FIT-TR-2002-02, Queensland University of Technology, Brisbane, 2002.* See also <http://is.tm.tue.nl/research/patterns/>.

## 2 Workflow Design

This section and the three subsequent ones illustrate the main features and the graphical user interface by examples chosen from a concrete, operational AARE project.

The design of a workflow happens by a simple-to-use web user interface without the need to write scripts. A workflow can be built iteratively by setting workflow properties, adding and configuring activities as well as connecting them. Workflows which are already running can be edited as well. The changes take effect automatically for newly started workflows - already running workflows continue using the old definition to maintain consistency.

The design of a workflow starts with creating a list of typical properties, such as the title and a description to give users an idea what the workflow is to be used for, see Figure 3.

Next, we start adding the first activity. In this example, we create an activity called *Request*, which should collect all the information necessary to start the workflow. On the first screen, see Figure 4, we give a title and description and some other properties, such as its expected duration and the starters responsibility.

	Aare	Staffware	COSA	InConcert	Eastman	FLOWer	Domino	Meteor
Sequence	✓	✓	✓	✓	✓	✓	✓	✓
Parallel Split	✓	✓	✓	✓	✓	✓	✓	✓
Synchronization	✓	✓	✓	✓	✓	✓	✓	✓
Exclusive Choice	✓	✓	✓		✓	✓	✓	✓
Simple Merge	✓	✓	✓		✓	✓	✓	✓
Multi Choice	✓		✓				✓	✓
Synchronizing Merge	✓			✓	✓		✓	
Multi Merge	✓				✓			✓
Discriminator					✓			
Arbitrary Cycles	✓	✓	✓		✓		✓	✓
Implicit Termination	✓	✓		✓	✓		✓	
Multiple Instances without Synchronization					✓			✓
Multiple Instances with a Priori Design Time Knowledge		✓	✓	✓	✓	✓	✓	✓
Multiple Instances with a Priori Runtime Knowledge						✓		
Multiple Instances without a Priori Runtime Knowledge						✓		
Deferred Choice	✓		✓					
Interleaved Parallel Routing			✓					
Milestone			✓					
Cancel Activity		✓	✓					
Cancel Case	✓						✓	

Figure 1: Different workflow systems compared I



	Aare	Mobile	MQSeries	Forté	Vis. WF	Changemg.	I_Flow	SAP/R3
Sequence	✓	✓	✓	✓	✓	✓	✓	✓
Parallel Split	✓	✓	✓	✓	✓	✓	✓	✓
Synchronization	✓	✓	✓	✓	✓	✓	✓	✓
Exclusive Choice	✓	✓	✓	✓	✓	✓	✓	✓
Simple Merge	✓	✓	✓	✓	✓	✓	✓	✓
Multi Choice	✓	✓	✓	✓	✓	✓	✓	✓
Synchronizing Merge	✓		✓					
Multi Merge	✓			✓				
Discriminator		✓		✓		✓		✓
Arbitrary Cycles				✓		✓	✓	
Implicit Termination	✓		✓					
Multiple Instances without Synchronization				✓			✓	
Multiple Instances with a Priori Design Time Knowledge		✓	✓	✓	✓	✓	✓	✓
Multiple Instances with a Priori Runtime Knowledge								
Multiple Instances without a Priori Runtime Knowledge								
Deferred Choice								
Interleaved Parallel Routing	✓	✓						
Milestone								
Cancel Activity				✓		✓		✓
Cancel Case	✓							✓

Figure 2: Different workflow systems compared II

**Edit Workflow**

Close Save Export Roles Run Help

General Graph Diagram Activities Versions

Type:\* Software Change

Description: This workflow describes the process of a software-change request.

Type-Field: ☐ Type-Field

Save

Figure 3: General workflow properties

**Edit Activity: Request**

Close Help

General Documents Form Conditions Transitions Versions

Title:\* Request

Description: Please specify your software change request.

Priority: [dropdown]

Duration: 0 days 0 hours 10 minutes 0 seconds

Responsible:\* (starter) [dropdown]

Authorization:\* (no authorization required) [dropdown]

Simple Progress: ☒ Simple Progress

Save

Figure 4: General activity properties

Then we define a set of form fields that need to be entered before completing the activity, see Figure 5. In the example we add a memo field, where the user should give a detailed description of the change request and a managed-document-field, which enables the user to upload additional documents. The interface allows one to add a variety of different field types as seen in the preview in Figure 6, to configure them and to order them as the user likes.

As a next step we define the conditions that need to be satisfied to be able to complete the activity, see Figure 7. In the given example we tell the

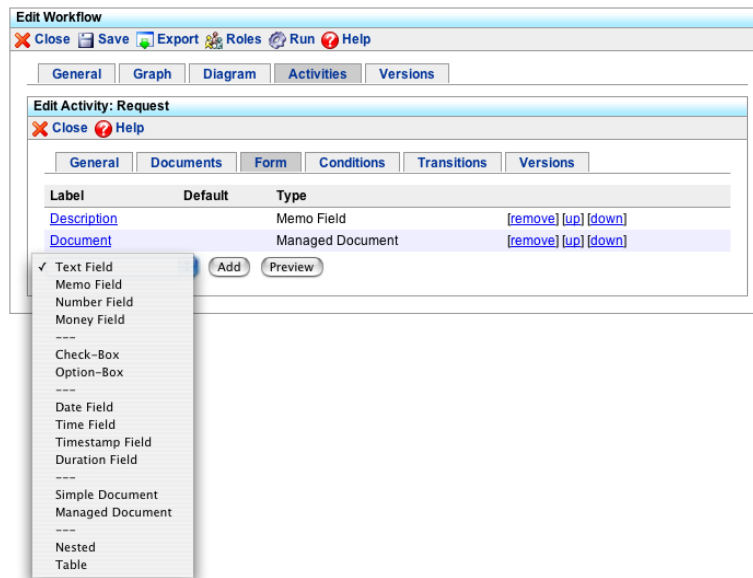


Figure 5: Define the data collection form of the activity

**Form Preview**

**Text Field:**

**Memo Field:**

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

**Number Field:**

**Money Field:**

**Check-Box:** ☒ Check-Box

**Option-Box:**

**Date Field:**

**Time Field:**

**Duration Field:**  days  hours  minutes  seconds

**Document Field:** (no document)

**Nested List:**

**Table:**

	Yesterday	Today	Tomorrow
<b>Morning</b>	<input type="text" value="takimata"/>	<input type="text" value="sea takimata"/>	<input type="text" value="Stet clita kasd"/>
<b>Evening</b>	<input type="text" value="duo dolores"/>	<input type="text"/>	<input type="text" value="voluptua"/>

Figure 6: Preview of possible form-field types

system to check if the memo-field defined in the previous step is not blank.

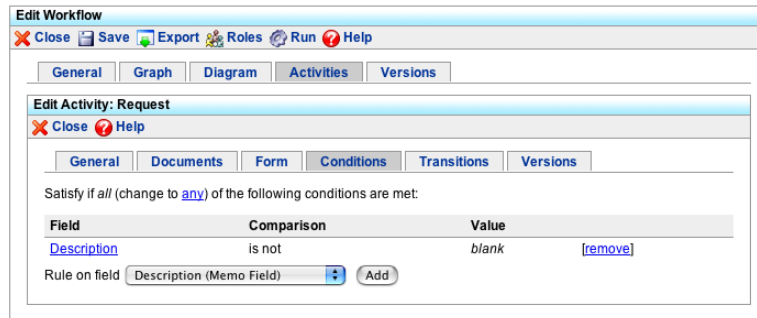


Figure 7: Define activity conditions

Last we define the transitions between this activity and others, see Figure 8. The editor consists of tree parts. On the left there is a list of activities the current one is depending on. In the middle there is a summary of the current activity. On the right there is a list of activities (including conditions) that follow the current one. By selecting an existing activity from the drop-down list and clicking on *add* we are able to actually define the transitions. In the example we connected the activity with the starting-activity of the workflow, that is implicitly available in every workflow.

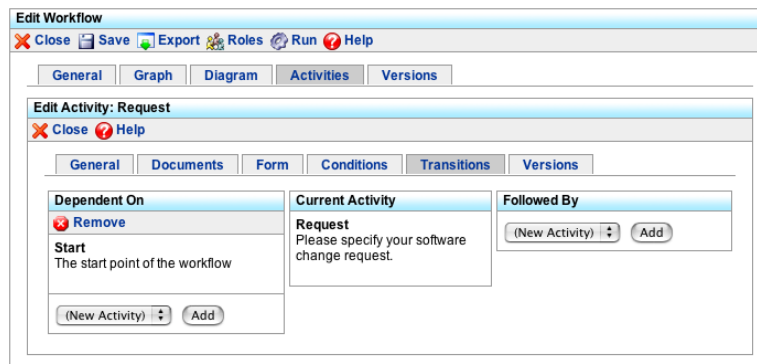


Figure 8: Define activity transitions

Using the screen seen in Figure 8, we are able to create new activities and directly connect them to the current activity by selecting *new activity* and clicking on *add*. This will create a new activity and brings us to an empty form as shown in Figure 4. This makes it very efficient to create a sequence of activities.

In a similar way we create the other activities of the workflow. Different views onto the activities are provided: a textual-report (Figure 9), a clickable graph (Figure 10), and a clickable diagram that is grouped by *responsible*

*user* and that shows all the conditions as well (Figure 11). These views make it easy to build complex workflows with many connected activities without getting lost.

Title	Responsible User	Responsible Role	Authorization User	Authorization Role
<a href="#">Request</a>	Lienhard Adrian	netstyle.ch>netstyle.ch		
<a href="#">Decision</a>	Lienhard Adrian	netstyle.ch>netstyle.ch		
<a href="#">End</a>	Lienhard Adrian	netstyle.ch>netstyle.ch		
<a href="#">Development</a>	Renggli Lukas	netstyle.ch>netstyle.ch		
<a href="#">Resolved</a>				

Figure 9: Workflow activity report

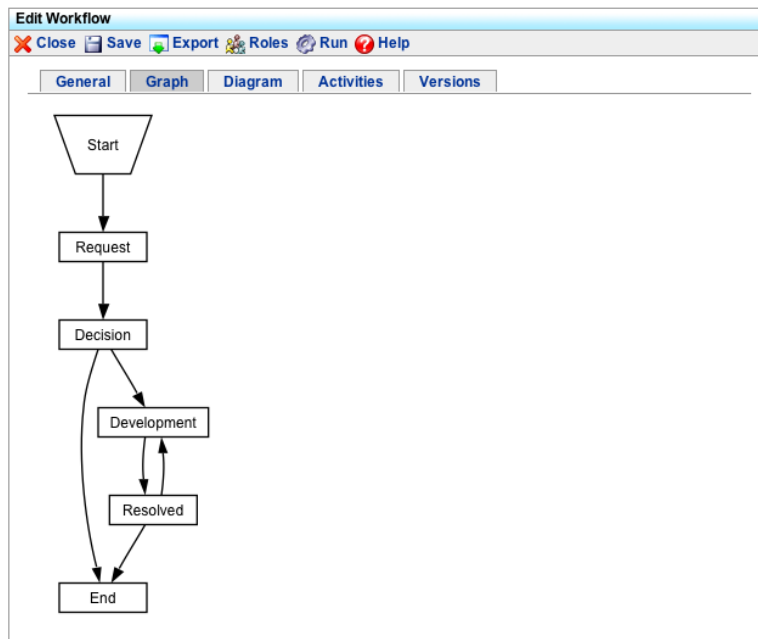


Figure 10: Workflow graph

### 3 Workflow Execution

Before starting a workflow one has to select which one to run. This is possible on the screen presented in Figure 12. In the upper part it is possible to search

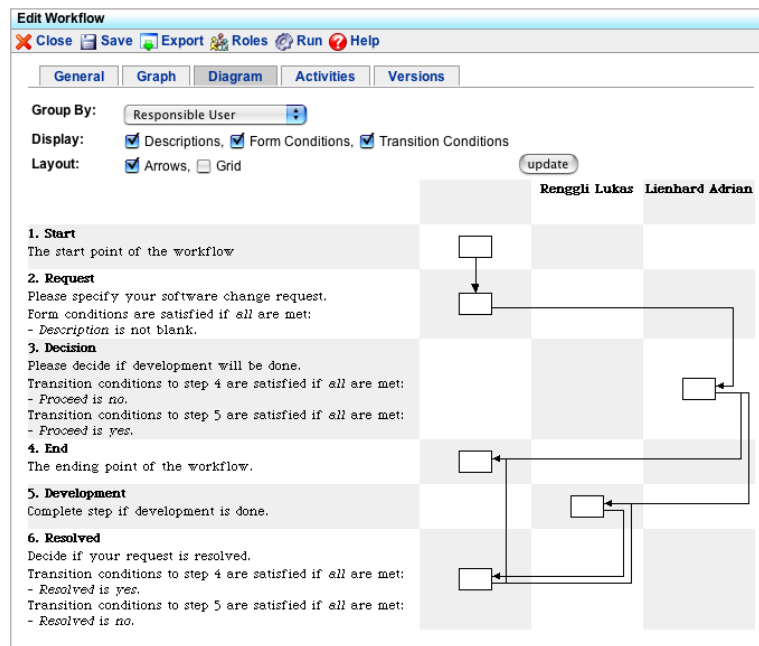


Figure 11: Workflow diagram

for matching workflows according to title and description. Below there is a list of all workflows sorted in categories.

The 'Select Workflow' dialog box has a 'Search:' field with the text 'Software'. Below the search field, there are 'Matches:' listed as 'Request IT [Software Change](#)' and 'Request IT [Software Enhancement Request](#)'. To the right of these matches is the text 'This workflow describes the process of a softwa...'. Below the matches, there are three dropdown menus for 'Workflow:' with the selected values 'Request', 'IT', and 'Software Change'. Below the dropdowns, there is a 'Selection:' field with the text 'Request — IT — Software Change' and a 'Description:' field with the text 'This workflow describes the process of a software-change request.' At the bottom, there are 'Ok' and 'Cancel' buttons.

**Select Workflow**

**Search:**  
Software

**Matches:**  
Request IT [Software Change](#)  
Request IT [Software Enhancement Request](#)

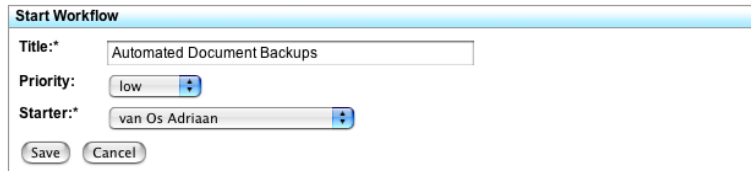
**Workflow:**  
Request  
IT  
Software Change

**Selection:** Request — IT — Software Change  
**Description:** This workflow describes the process of a software-change request.

Ok Cancel

Figure 12: Select workflow to run

To start the workflow, as seen in Figure 13, a title, a priority and a starting person is given to the new running instance. Per default the starter is the person currently logged in, but it is also possible to start a workflow for somebody else.

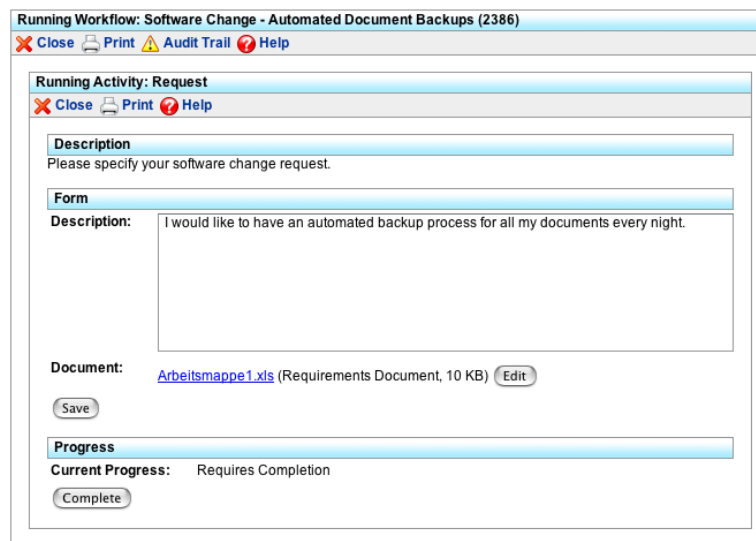


The 'Start Workflow' dialog box contains the following fields and controls:

- Title:** A text input field containing 'Automated Document Backups'.
- Priority:** A dropdown menu currently set to 'low'.
- Starter:** A dropdown menu currently set to 'van Os Adriaan'.
- Buttons:** 'Save' and 'Cancel' buttons at the bottom.

Figure 13: Start workflow

After starting the workflow the first activation is displayed, since the responsible user of this activation is the starter of the workflow, as defined during the workflow design. As seen in Figure 14, the user enters the description of his change request and uploads a document into the form defined in Figure 5. After filling the form the system detects that all the conditions are satisfied and tells the user that it requires completion. This is done by clicking on the button at the bottom of the form and will activate the following activations.



The 'Running Workflow' window displays the following sections and controls:

- Header:** 'Running Workflow: Software Change - Automated Document Backups (2386)' with 'Close', 'Print', 'Audit Trail', and 'Help' buttons.
- Running Activity: Request:** A sub-header with 'Close', 'Print', and 'Help' buttons.
- Description:** A text area with the prompt 'Please specify your software change request.'
- Form:** A section containing a 'Description' label and a text area with the input: 'I would like to have an automated backup process for all my documents every night.'
- Document:** A section showing a file named 'Arbeidsmappe1.xls (Requirements Document, 10 KB)' with an 'Edit' button.
- Buttons:** A 'Save' button below the document section.
- Progress:** A section with 'Current Progress: Requires Completion' and a 'Complete' button.

Figure 14: Completing the first activation

As we can see in Figure 11 the activation to follow has another responsible person, so the current user is unable to continue working with this workflow. A mail message is automatically sent to the responsible person implicating that a new activation is waiting for his interaction. A screen with all running and completed activations, see Figure 15, is always visible for all involved people to observe the current status of the workflow.

There is a variety of different views available for running workflows: first






Running Workflow: Software Change - Automated Document Backups (2386)					
 Close  Print  Audit Trail  Help					
Running Activities					
Description	Responsible	Deadline	Progress	Summary	
 <a href="#">Decision</a> Please decide if development will be done.	<a href="#">Lienhard Adrian</a>		0%	<a href="#">[summary]</a>	
Completed Activities					
Description	Completed By	Completed	Summary		
<a href="#">Request</a> Please specify your software change request.	<a href="#">van Os Adriaan</a>	30 November 2005	Description: <i>I would like to have an automated backup process for all my documents every night.</i> Document: <i>a BLDocumentHistory</i> <a href="#">[summary]</a>		

Figure 15: Next activation

the default view showing a list of running and completed activations and a brief summary, see Figure 16. Then there is a log listing all the changes within the running workflow, see Figure 17. Again we have a workflow graph (Figure 18) and a workflow diagram (Figure 19), but this time they are annotated with runtime information (running activations are in blue, completed activations in green). Last but not least there is a Gantt diagram, see Figure 20, showing past and future efforts of within this running workflow.

Running Workflow: Software Change - Automated Document Backups (2386)

Close

Print

Audit Trail

Help

Activities

Report

Graph

Diagram

Gantt

General

Running Activities

Description	Responsible	Deadline	Progress	Summary
<div><div></div><div>Resolved</div><div>Decide if your request is resolved.</div></div>	<a href="#">van Os Adriaan</a>		0%	Resolved: no <a href="#">[summary]</a>

Completed Activities

Description	Completed By	Completed	Summary
<div><div><a href="#">Request</a></div><div>Please specify your software change request.</div></div>	<a href="#">van Os Adriaan</a>	30 November 2005	Description: <i>I would like to have an automated backup process for all my documents every night.</i> Document: <i>a BLDocumentHistory</i> <a href="#">[summary]</a>
<div><div><a href="#">Decision</a></div><div>Please decide if development will be done.</div></div>	<a href="#">Lienhard Adrian</a>	30 November 2005	Proceed: yes <a href="#">[summary]</a>
<div><div><a href="#">Development</a></div><div>Complete step if development is done.</div></div>	<a href="#">Renggli Lukas</a>	30 November 2005	<a href="#">[summary]</a>

Figure 16: Report about running and completed activities

## 4 Workflow Reporting

The workflow reporting tool is a utility to create, edit, run and save user defined textual and graphical reports using all the accessible data of running workflows within the system.



Running Workflow: Software Change - Automated Document Backups (2386)				
Close Print Audit Trail Help				
Activities Report Graph Diagram Gantt General				
Title	Message	Comment	User	Timestamp
Resolved	started		Renggli Lukas	30 November 2005 2:18:54 pm
Development	completed		Renggli Lukas	30 November 2005 2:18:52 pm
Development	progressed 100%	completion with simple progress	Renggli Lukas	30 November 2005 2:18:51 pm
Development	started		Lienhard Adrian	30 November 2005 2:18:29 pm
Decision	completed		Lienhard Adrian	30 November 2005 2:18:27 pm
Decision	progressed 100%	completion with simple progress	Lienhard Adrian	30 November 2005 2:18:26 pm
Decision	started		van Os Adriaan	30 November 2005 2:11:54 pm
Request	completed		van Os Adriaan	30 November 2005 2:11:52 pm
Request	progressed 100%	completion with simple progress	van Os Adriaan	30 November 2005 2:11:51 pm
Request	started		van Os Adriaan	30 November 2005 2:01:20 pm
Start	started		van Os Adriaan	30 November 2005 2:01:19 pm

Figure 17: Log of the changes within the workflow

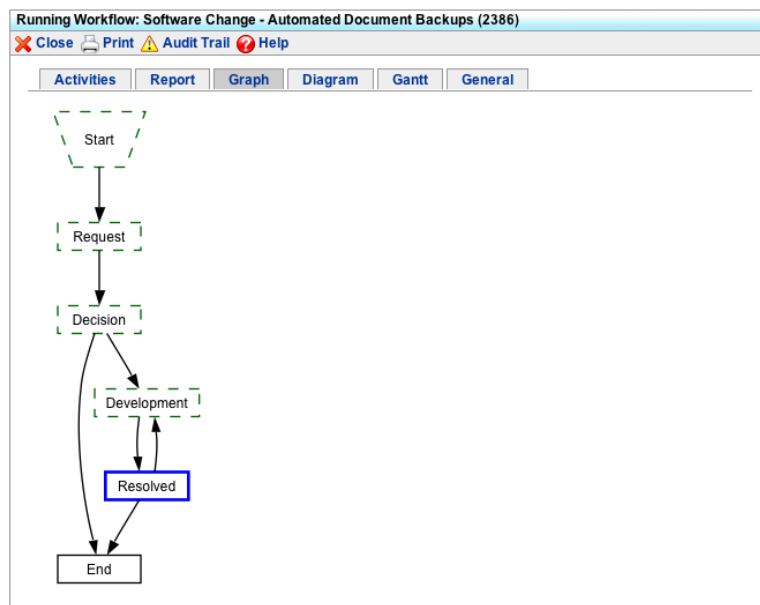


Figure 18: Running workflow graph

## 4.1 Interface

The reporting tool consists of a toolbar to perform different actions and four different parts to customize the reports, as seen in Figure 21. The same interface is used to create textual- and graphical-reports.

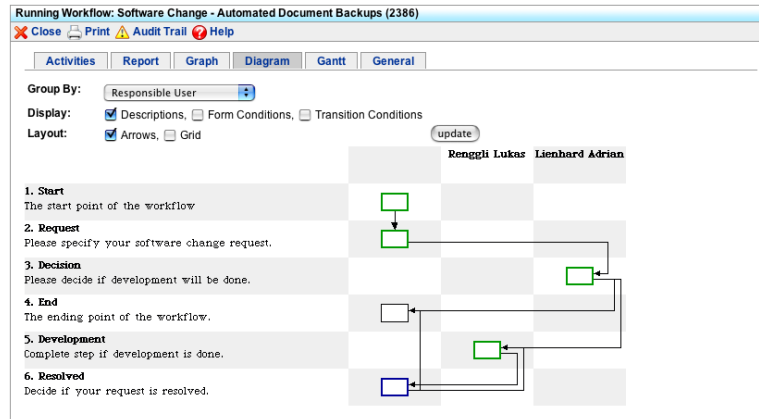


Figure 19: Running workflow diagram

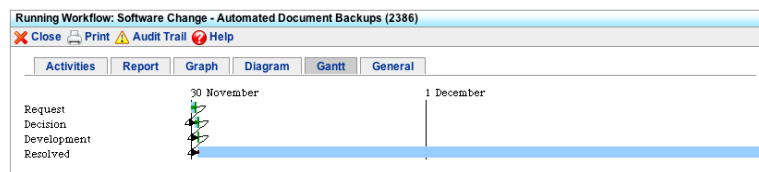


Figure 20: Gantt diagram

#### 4.1.1 Title

The first part of the reporting tool allows one to give a title and a description to the report. It is solely used to distinguish different saved reports and to give users a hint what the report is supposed to display.

#### 4.1.2 Selectors

The second section within the reporting tool allows one to restrict the processed data to certain workflows and their activities. It is important to note that the reporting tool is working on the level of activations, hence keeping the default settings will result in a list of all the activations being currently managed by the system.

The second line of drop-down boxes allows one to restrict the query to a certain period of time: the workflow starting time and the activation creation time. The distinct checkboxes are particularly useful when the report is used to create diagrams: since reporting works on the level of activities, it might be necessary to remove duplicates, e.g. if one is working on a workflow level or activity level.

The screenshot shows a 'Reporting' window with a menu bar (New, Open, Save, Roles, Help). The main area contains several sections:

- Title:** A text box containing 'New Report'.
- Description:** A large empty text area.
- Views:** A section with four checked checkboxes: Report, Export, Pie Chart, and Line Chart. Below them are 'Save' and 'Cancel' buttons.
- Workflow and Activity:** Two sections, each with a dropdown menu set to '(all)' and a 'Select:' dropdown set to 'All'. Below each 'Select:' dropdown is a 'Distinct' checkbox, which is currently unchecked.
- Conditions:** A section titled 'Conditions: List if all (change to any) of the following conditions are met:'. It contains a table with headers 'Field', 'Comparison', and 'Value'. The table is currently empty, with the text 'The report contains no conditions.' below it. There is an 'Add' button next to the table.
- Field Title:** A section titled 'Field Title' with the text 'The report contains no fields.' below it. There is an 'Add' button and a 'Default' button next to it.

Figure 21: Workflow reporting interface

### 4.1.3 Conditions

Adding conditions allows one to further restrict the resulting activities with user defined relations. The fields where conditions might be defined depend on the workflow and activities specified in the *selectors* part, e.g. custom data-collection fields are only available if the query has been restricted to a particular activity.

### 4.1.4 Fields

The fields are the columns that should appear in the final report. Similar to the conditions the available fields depend on the basic selectors specified in the second part of the reporting tool.

## 4.2 Views

There are four different view types available to visualize the result of the report.

### 4.2.1 Report

The report simply emits all the matching entries and fields in a sortable textual report.

### 4.2.2 Export

This is basically the same as the report, but emits it in a format readable by applications such as Microsoft Excel.

### 4.2.3 Pie Chart

The pie chart is showing the percentage of the grouped queried data, such as about the completion status of the workflows in the system. See example Figure 22.

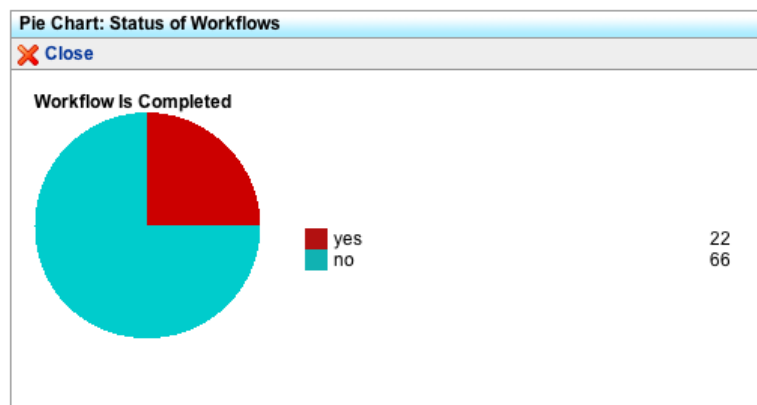


Figure 22: Pie chart – Completion status of workflows

### 4.2.4 Line Chart

The line chart is a graph showing the devolution of activations, such as the number of workflows started at a specific date. See example Figure 23.

## 5 Document Management System

As well as from within workflow definitions and running workflows, documents can be searched, edited and uploaded independently.

### 5.1 Document Editor

The document editor provides all the functionality to view, edit and modify a managed document depending on the permissions of the user, see Fig-

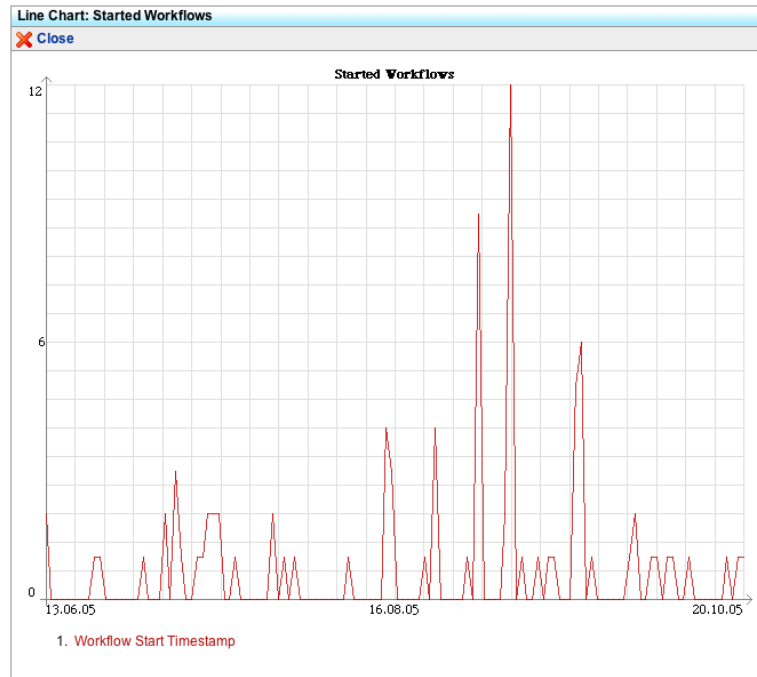


Figure 23: Line chart – Started workflows

ure 24. The toolbar allows one to quickly upload a new version or download the current one, however depending on the rights of the current user these capabilities might be restricted. The document management system detects conflicts, i.e., when two users have modified the same document.

**General.** The General tab consists of both, the default and custom fields of a document. The default fields are the id, the title, a comment and a “valid-until-date” telling up to when this document is valid. The custom fields depend on the type of the document. Users that have the write permissions are able to edit the fields, others just get a read-only view of the data.

**History.** The History tab lists all the document versions that have been committed in the past. The current version is displayed as the topmost entry.

## 5.2 Editing Document Types

The document type editor, see Figure 25, illustrates the definition of document types.

Document: Lukas Renggli

Close Download Upload Help

General History

Id: 45

Title:\* Lukas Renggli

Comment:

Valid Until: Choose

First Name:\* Lukas

Last Name:\* Renggli

Birthday: 11 June 1980 Choose

Save Cancel

Figure 24: Document editor

Document Types

New Type Help

Job Application Edit

Definition of Job Application

Field Label	Default	Required	Type	
First Name		yes	Text Field	<a href="#">[remove]</a> <a href="#">[up]</a> <a href="#">[down]</a>
Last Name		yes	Text Field	<a href="#">[remove]</a> <a href="#">[up]</a> <a href="#">[down]</a>
Birthday		no	Date Field	<a href="#">[remove]</a> <a href="#">[up]</a> <a href="#">[down]</a>

Text Field Add Field

Read Access Restricted To	Department	Default User	
netstyle.ch	netstyle.ch (NET)	Renggli Lukas	<a href="#">[remove]</a>

netstyle.ch > Anybody Add Role

Write Access Restricted To	Department	Default User	
netstyle.ch	netstyle.ch (NET)	Renggli Lukas	<a href="#">[remove]</a>

netstyle.ch > Anybody Add Role

Figure 25: Document type editor

Selecting an existing type from the drop Document Types drop-down list, displays the definition of that particular type below.

**Metadata Fields.** Lists the custom fields of that particular type. New fields can be added by specifying the settings of the field, such as the label, a comment, a default value and whether it is required or not. Fields can be reordered as well, which does not affect existing data since it just changes the order to display the fields.

**Access Rights.** Specifies the roles that are allowed to read/write the documents of the selected type.

### 5.3 Searching Documents

The document search allows one to find documents in the system, see Figure 26. The search is structured in three parts that can be combined: document type, validity (on time, overdue) and a text. Latter searches the document metadata fields: id, document title, document comment and field text values.

The screenshot shows a web interface for document search. At the top is a 'Document Search' header with a 'Help' link. Below it are three search criteria: 'Types' (a dropdown menu set to 'Job Application'), 'Valid:' (a dropdown menu set to '(all)'), and 'Text:' (an empty text input field). A 'search' button is located to the right of the text input. Below the search filters is a 'Documents' section with an 'export' link. It contains a table with the following data:

Id	Title	Type	Version	Uploaded By	Uploaded At	Valid Until	First Name	Last Name	Birthday
46	<a href="#">Ursula Freitag</a>	Job Application	1	<a href="#">Renggli Lukas</a>	30 November 2005 3:27:43 pm		Ursula	Freitag	19 September 1980
45	<a href="#">Lukas Renggli</a>	Job Application	2	<a href="#">Renggli Lukas</a>	30 November 2005 3:35:26 pm		Lukas	Renggli	11 June 1980

Figure 26: Searching documents