



COMMON KNOWLEDGE WHITE PAPER

Workflow Rules Extender

Making your business rules **your** business™



Business Rules

Rules Extenders

Common Knowledge provides a set of *Rules Extenders* which provide various formats for the representation of business rules. The full set of Rules Extenders includes the following:

1. Decision Tables
2. Decision Trees
3. Decision Grids
4. Workflow
5. Rete Rules
6. Script
7. ActiveScript

Common Knowledge now includes the Workflow Rules Extender which enables business rules designers to architect powerful combinations of declarative rules together with considerations of time, sequence and rule dependencies. The Workflow Rules Extender provides a graphical environment for defining process flows which can incorporate other rules representations as well as external actions.

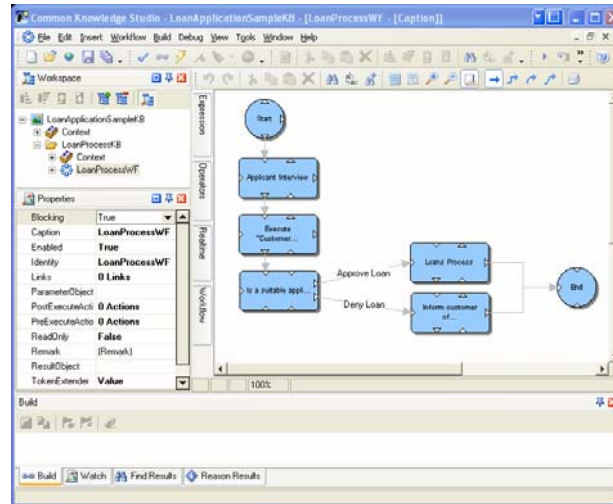


Figure 1 – Common Knowledge Workflow Editor Screenshot

Traditionally, rules management systems have focused primarily on declarative representations of business rules, such as decision tables, decision trees and production system rules. This has been an important strategy in breaking away from the default approach of the IT industry, which has been to implement business rules in procedural language constructs. Whilst the best designs will always extract as many business rules as possible to their declarative form, there are some limitations on declarative rules relating to timing, sequencing and dependencies. These limitations can be addressed by considering the concept of a "flow" through the system, and defining how rules and actions fit into this flow. This "flow" is often termed "workflow" or "rule flow".

Considerations of flow within business rules are now being recognised by various authors, commentators, methodologists and product vendors in the field of business rules system. The responses to this are, however, somewhat varied. Some authors will recommend including flow into the definition of the business rules whilst others will restrict scope to only declarative rule formats. Likewise, some business rules management systems will include facilities for defining process flows, whilst other vendors will recommend that this be co-ordinated by a dedicated workflow management system. There will be certain trade-offs with whichever approach is taken so it is important to make careful consideration of the needs of your particular problem when deciding which approach to take with your particular solution.



Background to Workflow Management Systems

WfMC Reference Model

The Workflow Reference Model, of the Workflow Management Coalition, defines the following five interfaces to a workflow engine:

1. **Process Definition Interface** – between process definition and modelling tools and workflow engine.
2. **Workflow Client Application** – API for client applications to request services from the workflow engine.
3. **Invoked Application Interface** – allows WFE to invoke a variety of applications.
4. **Workflow Interoperability Interface** – for one workflow engine to interact with other workflow engines.
5. **Administration & Monitoring Interface** – for monitoring and control functions of the workflow engine.

It is worth taking a brief look at the background of workflow management systems. It is easy to get confused about the meaning of the term "workflow" when discussing information systems. This confusion stems from the wide variety of workflow-related terminology and concepts used by vendors, consultants and industry analysts. This, in turn, is a consequence of the relative newness of the technology, the lack of standardisation and the variety of ways in which a business process can be described. This paper will not attempt to clarify all terms existing in the landscape of workflow systems. Rather, we will borrow enough terms as is necessary to explain how a business rules system can fit with the workflow requirements of an information system. Let us look at some definitions:

Definition 1: A **workflow** is very general concept describing how to model a process.

Definition 2: A **workflow management system (WFMS)** is a software component that takes as input a formal description of business processes and maintains the state of the processes' executions, thereby delegating activities amongst people and applications.

The first of these definitions is, quite clearly, a very general description of what workflow can be. This is quite a useful definition for us, though, as it encompasses most of the explanations of workflow that you are likely to hear from industry pundits. With this definition, you could define workflow models for something as low-level as the definition of a multi-step mathematical function, or as high-level as an enterprise-wide system for processing customer enquiries. Much of the industry commentary tends to centre around the higher-level enterprise processes, often referring to "Business Process Management". Workflow and Business Process Management tend to have overlapping meanings. "Workflow" is the term preferred by software developer communities, used in a technical context, when denoting a series of related interactions between people and a computer system. Business Process Management has a more broad scope, often covering non-technical issues like analysis organizational impact, from the viewpoint of a manager.

Usages of WFMS's can generally be classified as one of the following three types:

- **Enterprise Application Integration (EAI)** - A large enterprise will commonly possess a number of disparate applications each dedicated to a separate functional area. The enterprise may, however, want to combine the functionality of these dedicated applications into a larger organisational business process. This requirement is the focus of many WFMS's that focus heavily on these integration aspects.



Background to Workflow Management Systems (continued)

- **Co-ordinating people-related tasks** - Some WFMSs accompany the process definition tools with easy creation of simple client forms. This approach can deliver high value where the business process involves a lot of people related tasks.
- **Embedded workflow engine** - With this approach, the workflow engine is embedded within another application. The WFMS is not directly visible to the end-user. The main purpose of this approach is to enable better maintainability and reuse within applications that employ complex workflow-related logic.

Aspects of Workflow Implemented in Common Knowledge

Other Rule Extenders

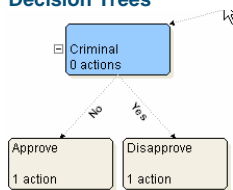
Workflow can be integrated with the following other Rules Extenders (and more):

Decision Tables

	Rule1	Rule2	Rule3
Age	[0:2]	[0:2]	[3:8]
Weight	[0:8]	> 8	[0:20]
Dose	5	8	10

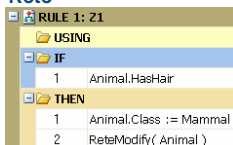
An intuitive, tabular format that provides a highly-maintainable and logical organisation of rules

Decision Trees



Useful for illustrating decision points and outcomes within a decision-making process.

Rete



Rete algorithm for production system rules.

Scripts

```

procedure main;
var
  Index: integer;
begin
  for Index := 1 to 8 do
  begin
    AnimalRules.Reset;
    AnimalRules.Execute();
  end;
end;
  
```

Procedural logic is sometimes needed when rule requirements do not easily match one of the more standard rule formats.

It is important recognise that Common Knowledge Workflow is not intended to act as an enterprise workflow management system. Rather, it is designed to fulfil the role of an embedded workflow engine within a host system that has a complex set of business rules requirements. Common Knowledge incorporates workflow into rule definitions in order to better represent rules that are utilised by a business during its operations. A dedicated WFMS is generally the solution of choice when an organisation needs off-the-shelf workflow capabilities including: administration and monitoring of the workflow; pre-built forms for accessing the workflow; and plug-and-play integration with a large array of other software products. Often, however, an organisation's requirements are not centred around these facilities but are focused on the ability to automate a complex business process that is governed by many and varied constraints. Common Knowledge excels at representing and automating these sorts of requirements.

The Common Knowledge Workflow Rules Extender provides a flexible and feature-rich tool for defining a large variety of types of complex process flows. In addition to this, the Workflow Rules Extender can be combine with any other Common Knowledge Rules Extender (rules representation) to produce powerful hybrid solutions. For example, a set of Workflow rules could be defined to model a business process that incorporates a Decision Tree for deciding the direction of the process flow at a decision point in the flow. However, the tool's capabilities also go beyond the rich expressive-power available in the process definition.. Flexible integration with other systems can be achieved by combining Workflow with Common Knowledge Channels to enable input/output communication with external systems and parties. A range of Channel types are available in Common Knowledge including the SMTP-POP3 E-Mail, Socket, and Microsoft Message Queue.

As previously mentioned, a workflow can be used to describe processes that range from very low-level to very high-level. Common Knowledge Workflow Rules Extender provides a feature-rich graphical environment for creating workflow process definitions, including a large palette of tools that can be readily assembled into models that cover a great range of process types. Four useful categorisations of process types are shown following with a sample of the Workflow rules created to illustrate each.

Aspects of Workflow Implemented in Common Knowledge (continued)

Flowchart: A process-flow model that emphasises the decision-making paths that guide someone through a particular process.

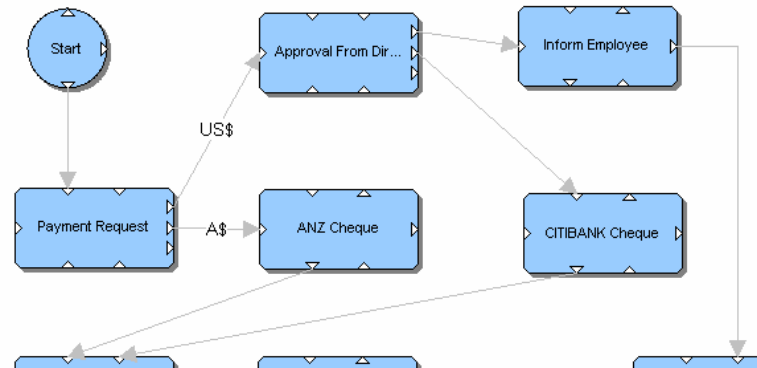


Figure 2 – Portion of a Payment Request Flowchart

Workflow: A set of activities and events that occur during the execution of a business process.

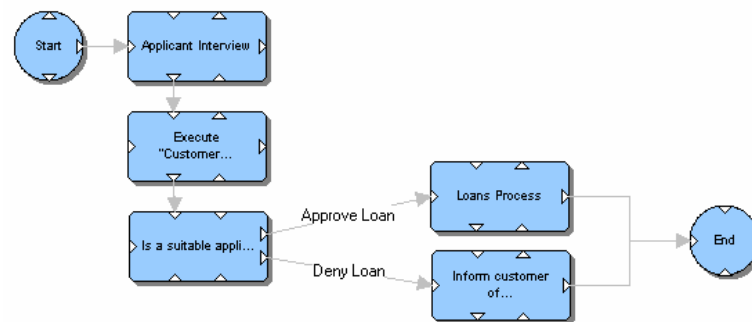


Figure 3 – Example of a Loan Approval workflow process

Statemap: A model describing a system that exist in only one particular state at any given time, and moves from one state to another in response to certain external events.

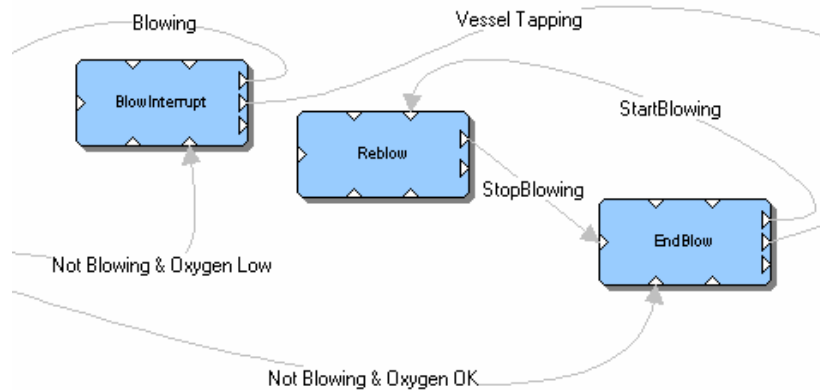


Figure 4 –Statemap of basic oxygen steelmaking process

Aspects of Workflow Implemented in Common Knowledge (continued)

Ruleflow: Used to define control over the sequence of rule execution and chaining.

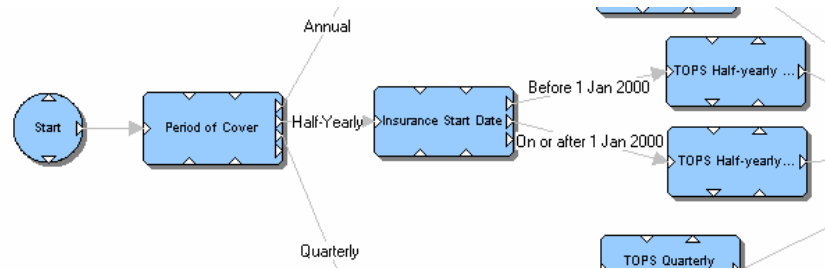


Figure 5 – Portion of an Insurance Quotation Ruleflow

The same Workflow Rules Extender can be used to create each of the above types of Workflow rules. The Workflow Rules Extender provides tools to model workflows using a variety of control-flow patterns. By adopting this control-flow perspective on workflow, Common Knowledge provides a base set of tools that can be built up into powerful combinations to model complex flows.

Another point worth noting is that workflow can provide a neat alternative to using scripting with your business rules. There has long been much debate about whether or not there is a place for scripting languages with business rules definitions. The opponents of such use of scripting will argue that scripting is inappropriate because: a) it is not a declarative representation of business rules, and b) it is not an intuitive medium for use by business-oriented users of rule systems. That said, there have always been business rule requirements that are difficult to solve using declarative rules alone that can be easily worked around with a little scripting. Often these problems centred on the control of execution of business logic. This is where workflow can represent an attractive alternative. The business rules can be incorporated into a workflow process definition that is graphical and, therefore, more intuitive to the more business-minded users of the rules.

Terminology Used In Workflow Rules Extender

Before examining the tools provided with the Workflow Rules Extender in detail, it is worth taking a look at the main terms that are used when describing Common Knowledge Workflow rules.

Nodes: A Workflow has one or more Start Nodes and one or more End Nodes. Individual activities (such as evaluating expressions or performing business tasks) are represented by specialised Nodes which are connected to one or more other nodes by Connections, ultimately originating from a start node and leading to an end node. The specialised Nodes provided with Workflow Rules Extender implement the control-flow patterns needed to build workflow models. Additionally, custom actions can be attached to nodes in the workflow.



Terminology Used in Workflow Rules Extender (continued)

Connections: Connections are unidirectional and an arrow on one end of a connection indicates the direction in which the workflow process flows from one node to the next. State information can flow along these connections and, additionally, custom actions can be defined to execute upon transition along this connection.

Tokens: Tokens represent the data (state information) that flows through a workflow or drives the workflow and can range in type from a simple value such as an integer to a complex object such as an XML document. Tokens flow along connectors between nodes. Data can be read from, or written, to tokens during execution of the workflow.

Commands: Each node supports a set of one or more commands including the Default command which is supported by all nodes. When a Connection is made to a Node then the target command is specified. The Default command instructs a node to perform its normal processing with the given token. More sophisticated nodes may support additional commands such as a Reset or Clear command for nodes that maintain state.

Node Types in Workflow Rules Extender

The Common Knowledge Workflow Rules Extender more than thirty different types of Workflow nodes that can be used to assemble complex workflow models. The nodes are grouped into the following categories: Workflow, Expression, Operators and Realtime. The following section describes a subset of the nodes available in each category in order to illustrate some of the functionality that is available in the Workflow Rules Extender.

Node Icons



Workflow Nodes

These nodes represent traditional aspects of business processes such as parallel flows, decision points and activity synchronisation.

Task: Task node represents a Workflow task for which acknowledgement of completion is required before the Workflow execution continues.

Split: The Split node allows a single Workflow path to be split into multiple paths.

Merge: The Merge node allows multiple Workflow paths to be joined into a single path.

Expression Nodes

These nodes can be combined together to form complex expressions, formulae and calculations.

Expression: The Expression node executes an expression. The result of the expression can be assigned to the Token if the AssignToToken property is set to true.

Average: The Average node calculates the average value of received Workflow tokens and outputs the result.





Node Types in Workflow Rules Extender (continued)

Node Icons



Operator Nodes

The Operator category of nodes consists of four sub-categories: Arithmetic, Logical, Comparison and Bitwise. These nodes represent a set of low-level operators that can be used to build more complex function definitions.

Binary Arithmetic: The Binary Arithmetic node performs an arithmetic operation on two or more Workflow token values and outputs the result.

Binary Comparison: The Binary Logical node performs a logical operation on two or more Workflow token values and outputs the result.

Real-time Nodes

A set of nodes useful typically suited to real-time systems such as those used in manufacturing processes.

Signal Generator: The Signal Generator node outputs a series of Workflow tokens whose values are generated from the selected waveform. Each value is a sample taken at a regular interval in the waveform cycle.

Analogue Alarm: The Analogue Alarm node provides an alarming mechanism for analogue (floating point) values provided by incoming Workflow tokens. The Analogue Alarm node may output a Workflow token indicating a particular alarm as a result.

Capabilities of Workflow Rule Extender

The innate power of Workflow Rules Extender is readily apparent when you examine the expressive-power of its graphical workflow process definition, but this becomes even more impressive when you really understand the possibilities that are generated by combining Workflow with the existing Common Knowledge Rules Extenders. There are a several other aspects that are worth considering in order to gain an understanding of the real potential of the Workflow Rules Extender.

Standalone Execution: Like all other Rules Extenders in Common Knowledge, Workflow can be fully executed, standalone within the Common Knowledge Studio environment. Within Studio's run-time simulator/executor, environment conditions can be set up, external system actions can be simulated, data values can be logged and the run-time execution flow can be observed. This provides an extremely powerful feedback mechanism when analysing, developing or debugging workflow processes. The rules will then be deployed with the host system along with the Common Knowledge Rules Engine.



Capabilities of Workflow Rule Extender (continued)

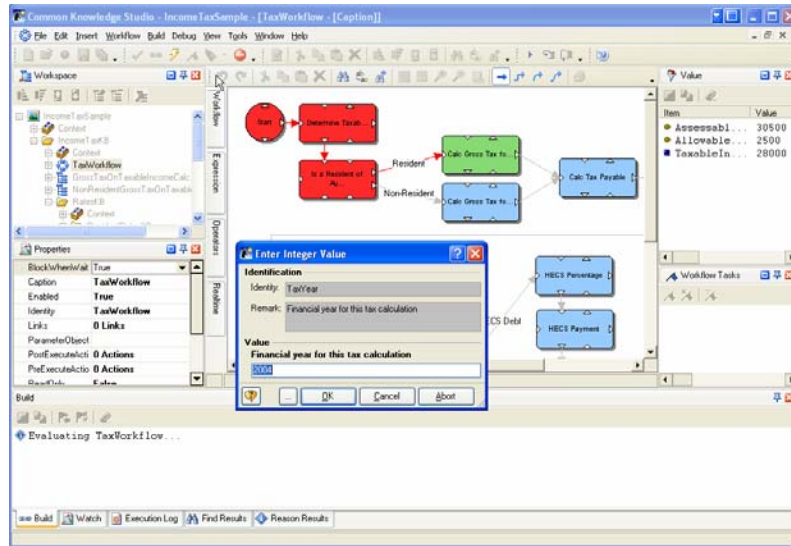


Figure 6 – Screenshot of Common Knowledge Studio mid-way through executing a Workflow

Process Definition Model: Workflow Rules Extender is designed around generic "token flow" model for describing the workflow processes. This means that it can be used to represent all manner of "flow type" diagrams. This way we are not limited to using only Statemaps, or only Process Flow Diagrams, but we can build the type of model that suits our needs...even if this means building some sort of hybrid model!!

Large Range of Nodes: As we have already seen, Workflow Rules Extender provides a large range (more than thirty!) of nodes with which to build workflows. Because these nodes implement a wide array of basic control-flow mechanism, they can be combined to represent most process flow scenarios that you can imagine. These nodes are built upon an extensible architecture within Common Knowledge which means that new node types can be easily created. It is likely that new node types will be continually added as more commonly-required scenarios emerge with continued use of Workflow. Additionally, it would be relatively straightforward for an OEM Partner of Object Connections to implement their own Workflow nodes.

Persistence: The Workflow Rules Extender also includes capabilities for persistence of Workflow state so that Workflow can be re-executed at a later time using the restored state. By implementing the specified interface to provide the Rules Engine with a "persistor", you can persist Workflow state to a database, a file, or whatever data store mechanism you would like to implement. Additionally, Workflow Rules Extender provides file-storage persistor which can be used "out of the box".



Capabilities of Workflow Rule Extender (continued)

Channels: In Common Knowledge a Channel is an element that can be used to define the attributes of a connection to an external point of communication or interaction. Channels can be likened to a "pipe" through which information can be sent from Common Knowledge to external systems or received by Common Knowledge from external systems. Channels can be used from within any of the Rules Extenders in Common Knowledge, but they are particularly useful with Workflow because of the likely need to interact with other systems or people. The following Channel types are provided with Common Knowledge:

- MSMQ Channel - for integration with Microsoft Message Queue.
- SMTP-POP3 E-mail Channel - for sending/receiving for e-mail.
- TCP-IP Socket Channel - for communicating with a remote process using TCP.

These Channels provide the base mechanisms that would be required to build integration capabilities for most likely required integration scenarios.

Summary

..... The Workflow Rules Extender of Common Knowledge provides a format for representing the timing, sequencing and dependency aspects business rules that are not easily represented using declarative formats. Workflow Rules Extender not designed to be an enterprise workflow management system, rather, it is intended to act in the role of an embedded workflow engine within a system that needs the ability to model complex business rule requirements. Workflow Rules Extender is designed around generic "token flow" model that enables the construction almost any "flow-type" model including: flowcharts, workflows, statemaps and ruleflows. It includes a range of node types for building such models, as well as Channels for integration with external systems and parties. The primary power of the Workflow Rules Extender, however, is its ability to combine with the other Common Knowledge Rules Extenders in order to provide powerful hybrid business rule representations. These rules can then be executed standalone within Common Knowledge Studio as an aid to development and testing. Workflow Rules Extender provides an intuitive and configurable mechanism for representing and automating the "process" aspects of the business rules of an organisation.

Sydney
Suite 4, Ground Floor, 13a Narabang Way, Belrose, NSW, 2085, AUSTRALIA
T. +61 2 9450 2999 F. +61 2 9450 2744

Newcastle
Mezzanine Level, 3 Bradford Close, Kotara, NSW AUSTRALIA
T. +61 2 49573146

W. www.objectconnections.com E. info@objectconnections.com

