MooD Transformation Toolset 2006
Version: 2006

NEW

The Salamander Organization, Ltd.
Innovation Close, York Science Park,
York,YO10 5ZF
United Kingdom
Tel: +44 870 161-1700  Fax: +44 870 161-1701
Web: http://www.tsorg.com
Primary Contact: info@tsorg.com

1. Product Overview

The MooD Transformation Toolset is an enterprise-class business modeling environment specialized for helping organizations carry out organizational transformation. It is designed to support the entire range of activities that organizations undergo in any kind of transformation initiative, and allows the user to define their own semantics, whether the focus is based on organization, structure, Service-Oriented Architectures (SOA), roles, skills, personal development, risk, technology infrastructure, process, or information. MooD Transformation Toolset’s underlying technologies observe industry standards, promote business component reuse through frameworks and templates, and integrate with best-of-breed technologies to support Salamander’s own methodology for organizational transformation (the Business Transformation Lifecycle) as well as public domain methods. The software also works at different levels – for example, strategic, technical architecture, or detail web service, and in different modes – free from visualizing, brainstorming, and capture through more structured design.

The MooD Transformation Toolset (“MooD”) features powerful visualization tools for building and deploying intuitive, live business models of an organization, creating a Knowledge Map as a common communications and operational resource. You can model and link all organizational entities to build “real world” enterprise models, then elaborate or drill down to any level, to create a detailed operational or working model. Each item may then be linked to its real-world application.

MooD’s intuitive user interface and the quality of its visual modeling capabilities allow it to support both business users and IT/IS professionals. In addition, the ability to quickly create highly effective models means that MooD can be used interactively in workshop situations (e.g., projected onto a screen or for other presentations). In addition, output can be generated immediately for distribution in either hard copy or web format.

MooD provides support throughout the overall client development/transformation cycle – from initial engagement through business architecture creation to operational delivery. The MooD repository forms the foundation for modeling, analysis, reporting, architecture, and change management. Multi-user access can be concurrent or distributed with synchronization.

MooD’s architecture includes APIs and add-in mechanisms for extending the product's functionality. Integration and transformation capabilities are provided for both client and server side, using XML and XSLT technology. Web outputs are template-driven, and both content and appearance are under user-control. This enables clean integration with other tools and organizational style, with MooD, in effect, serving as the central hub, or “console,” for managing and integrating diverse information and software used for business transformation and development.
Models developed in MooD are “executable” using a unique invocation technique called “Process Activation,” in which model element content can be populated at runtime. This capability facilitates the linkage between the designed business (model) and systems architecture and a visualization of the “activated” model with live process combinations, key performance indicators (KPIs), and knowledge. In this way, for example, live performance dashboards can be automatically generated and maintained in “real time” and SOA components developed within the business context.

The second “secret weapon” in Salamander’s toolkit is the Transformation Blueprint (framework), which allows users to quickly generate a visually rich business model, complete with reports and presentation capabilities. The Transformation Blueprint is a generically developed framework consisting of a collection of elements, processes, notations, dashboards, and linkages, combined with organizational data and implementation capabilities, specific to various technical architecture and business process requirements. Transformation Blueprints provide a foundation for most organizational needs, from repository linkage, performance analysis, presentation, publication, etc., enabling early deployment and quick results from the MooD toolset. Companies can customize Transformation Blueprints to meet their specific organizational requirements. Salamander offers various industry and application-specific Transformation Blueprints for use with MooD designed to jump-start transformation efforts.

MooD allows organizations to create a rich and visual representation of their entire operation, across its multiple dimensions and within its various modes. This capability operates seamlessly across the operational boundary, connecting strategy to execution and all the multiple dimensions of an enterprise into one – for instance, enabling organizations to develop agile service-oriented architectures where all of the different dimensions of the business are connected and aligned; where technology is aligned to the process needs of the organization; where organizational competencies are aligned to strategic goals; and where activity matches purpose.

Table 1. Overview of MooD Transformation Toolset

<table>
<thead>
<tr>
<th>MooD Transformation Toolset 2006</th>
<th>Business process modeling and enterprise architecture toolset emphasizing transformation. MooD offers its own methodology for transformation called the “Business Transformation Lifecycle.” Key components include MooD Business Developer for visual modeling and definition; MooD Web Publisher for generating visual, integrated web sites from the underlying repository, used stand-alone or for integration with corporate intranets; Business Activation for integrating knowledge maps with information and application resources (e.g., content management systems, applications, performance data sources, etc.); Business Simulation for dynamic simulation and analysis.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repository</td>
<td>Multi-user repository that supports multi-perspective process modeling and provides multi-user repository access via locking or mediation. Security includes access history for each repository element.</td>
</tr>
<tr>
<td>Simulation</td>
<td>MooD is tightly integrated with Simul8 Corporation’s Simul8 simulation package, which provides dynamic simulation and animation of MooD models.</td>
</tr>
<tr>
<td>Integration</td>
<td>MooD uses XML for interoperability with external applications and sources of data; MooD supports OLE/COM for development of MooD add-in components, and third-party MooD-driven applications; standard formats for sharing and linking across Microsoft Windows and Office applications such as Excel, VISIO and Word.</td>
</tr>
<tr>
<td>MooD Modeler Add-Ins</td>
<td>Activation of model elements supports dynamic connection of models to other systems and tools in use within an organization, including simulation and web-based services. The MooDXMI Activator extracts repository content for generation into a standard format for input to UML compliant systems design tools such as Rational Rose. The process also works in reverse.</td>
</tr>
<tr>
<td>Pre-built Models &amp; Templates</td>
<td>Salamander’s Transformation Blueprints, Zachman Framework, Six Sigma, Balanced Scorecard, PRINCE, Enterprise Architecture Frameworks, MoDAF.</td>
</tr>
</tbody>
</table>
2. Product Architecture

2.1. Architecture Overview

Figure 1 provides an overview of the MooD Transformation Toolset architecture. MooD is a comprehensive environment built around a client-server architecture designed to support both development and use via the web. This architecture includes Development tools, Activation and Integration tools, Business Modeling, Simulation, and various End-user tools (browsing, query and reporting, etc.), Architecture and Administration tools, Repository, Knowledge/Resource Centers, Activators, and presentation and publishing tools.

The primary development tool is Business Developer. Business Developer supports visual modeling and definition, allowing full-function development of themes, elements, references, interdependencies and their logical relationships, and complete definition of the business model. Models and their associated information are stored in the MooD repository. The developer software includes web publishing facilities.

The primary user tool is the browser. Once published, users access models and other project artifacts through a web browser, primarily Microsoft’s Internet Explorer. It is also possible to build a solution enabling users to update information via the web using a Web Update Server option.

The Activation and Business Integration tools center around the key concepts of a Knowledge Map, Activation, and Activators. “Knowledge Maps” (i.e., business modeling projects and all their associated artifacts) offer an intuitive and live model of an organization, and provide an intuitive mechanism for all stakeholders to participate in a project.

“Activation” is Salamander’s term for the integration capabilities used to instantiate MooD’s modeled world with real-world systems and data in the business environment:

- **Knowledge Activation** connects information, documents, and data to the business model
- **Process Activation** discovers and connects application services, orchestrating and monitoring them in the context of the business model
- **Performance Activation** connects performance data from business applications to the business model, displaying dashboard views in the context of the business model, and alerting process owners (and other stakeholders) in the event of important threshold changes

Within the client software, Activation is enabled through batch updates of local repositories, and dynamically, once connected through the server modules. The Business Integration Server forms the underlying technology platform to support either batch or dynamic access to other systems and data via an XML interface to the MooD repository.

The Decision Engine is a web-based system that guides a user through a set of questions and answers to provide a scorecard assessment and a recommended course of action based on content defined in a business model.
The Knowledge/Resource Center allows resources to be stored and managed from a central location, and specifies how the resources it stores will be accessed from the Knowledge Map (once it is published). Activators are pre-built connectors for integrating the business models (maintained in the Knowledge Map) with other systems, databases, and tools used within the organization.
2.2. Usability and User Interface

Although the MooD tool environment is extremely rich and configurable, Salamander’s user interface was designed with user intuition in mind. According to Salamander representatives, users can become comfortable with its features in “less than twenty minutes.”

Upon launching MooD, the user is presented with the Repository Manager. This screen displays actions, commands, and reports appropriate to the privileges associated with an individual user’s defined role and project context. All browsing, merging, editing, and publishing tasks take place within this window. Whenever a Repository action is performed, the user is prompted to sign in to MooD security to ensure that only permitted users with the appropriate privileges can perform the requested action. As shown in the example window in Figure 2, after logging in, users are presented with common commands and various shortcuts to the models and repository items that are available to them.

![Figure 2. Example Repository Manager GUI](image)

MooD functionality is organized around the concept of a Knowledge Map, which can have collections of “themes.” Basically, themes enable and support modeling and design from a particular business perspective – for example, “Customer Service” or “Product.” (See Section 3.1, Managing Project Portfolios for more on themes.) The bottom of the screen shows the icons representing the themes for this user that have been set up for this Repository. This feature offers a quick means for users to generate new models based on existing themes. Upon selecting a theme from the Repository, a Start
Figure 3. Clicking on icon from user's home page leads to a world of user specific activities.

Page offers thumbnail sketches of all the models available within a Knowledge Map, organized by theme, as illustrated in Figure 3.

From this page, a user can easily and quickly build a model with pre-defined theme styles. With the model canvas open, the themed elements that are permitted on the model (as defined by model content rules specified by the Administrator) are presented in gallery form at the bottom of the model page. These only require a drag-and-drop operation to create a new instance, which may be renamed according to the user’s requirements. The user is free to accept the default style or modify it, either from the pre-defined (customizable) style gallery or by selecting from a broad range of formatting options available from a right mouse click. Connections between elements are created in a similar drag-and-drop manner by selecting themed connection links. In this way, the user is able to quickly create visually rich models. Assigning real-world resources to models can be done using similar drag-and-drop techniques, depending upon the source.

Also worthwhile to note is that the MooD 2006 Home page incorporates features that allow each organization to customize their business Repository to reflect their corporate image. The background of the home page can be changed to display a relevant image that shows the Repository's use or the company logo and colors, etc.

This page can also contain Repository-specific welcome text. This text may also be incorporated into the default published web page. The text is entirely configurable and can also be used to convey information to the user that the Administrator, Enterprise Architect, or project manager believes will be useful.
Also found on the Home page are command shortcuts and tabs for common actions such as exploring and searching the repository, and for creating and reusing queries. There are also shortcuts and tabs for creating new models, an editing style gallery, reviewer comments, and so on. Also available is a button for launching the Explorer Browser, which offers alternate access to the contents of a repository once it has been opened.

2.3. Repository Options/Team Development

MooD offers a mix of team-based development options centered on user-centric merge operation control. That is, MooD's security model and element ownership model form the foundation for how teams work. Repository elements are owned exclusively or globally. For exclusively owned elements there’s no contest; only the element owner may make modifications. For global access, MooD employs something similar to record-level protection in that simultaneous update to an element is allowed because the element is a logical and not a physical unit. Elements may contain fields and other attributes, which in relational database terms live in different records or tables that can be updated independently. Users can also be notified if updates to elements or their fields change. Teams working on separate areas of the repository export their portions of the repository, giving them a degree of portability in the process, which can then be recombined at a later stage. A mediation capability allows updates to the repository to be approved before they are committed. Mediators (i.e., those with approval and denial rights) can add comments to support their decision, which, along with the update, is added to the repository’s audit trail.

The MooD repository can support very complex projects. For example, Salamander works with some very large organizations with complex DoDAF/MoDAF architectures that when fully populated may result in repository sizes of 100s MBs.

The underlying database on which the MooD repository is based is relational and uses the ODBC connectivity model. Out-the-box MooD supports Microsoft Access, Microsoft SQL Server (including MSDE), and Oracle. Salamander states that other relational SQL-compliant databases that have ODBC drivers should also be MooD compatible. The MS Access, MSDE, and SQL 2005 Express options enable stand-alone MooD development. While SQL Server and Oracle platforms support enterprise-class team-based MooD development.

2.4. Integration with Other Products

MooD's repository-driven integration capabilities allow the mix-and-match of various development platforms, productivity applications (Microsoft Office, etc.), BPM production environments and other applications. Additionally, MooD has mechanisms in its XML schema for preserving vendor-specific round-trip information at a per-element level. Thus, solutions providers can build services on top of MooD, and MooD will ensure that correct linkage is maintained, even when that content is moved around within the repository.

As touched on previously, “Activation” is Salamander’s term for the integration capabilities that can be used to instantiate MooD models with actual systems and data used by an organization. Basically, MooD's Activation functionality acts like a foreign function call for any element in the MooD repository. For example, using Process Activation business process and other models can be activated with operational web services, third-party software, or proprietary business applications. This enables the operational business models held within MooD's repository to function as a hub — bringing together the business’ systems with its business activities and their flow. An activated model can then be “executed” giving access to the systems directly from the operational model. From here an orchestration can be previewed and evaluated via MooD tools that allow multiple views from results stored in the repository.

Accordingly, Salamander offers a number of pre-built “Activators” for use with MooD. For example, the entire Microsoft Office productivity suite can be activated within MooD. Other document formats, such as PDFs and rich media, are also supported. There are also additional generic
Activators for SQL, ODBC, and web service data sources. All Activators define their own properties (in XML) which are fully configurable for access at design-time and run-time (static and dynamic publishing modes).

Process Activation also supports the export of orchestrations for deployment within a production environment. This is accomplished using the Business Process Execution Language (BPEL), and Salamander has built BPEL import and export Activators for some of the major BPEL platforms.

MooD also integrates with other modeling and development tools. Salamander offers specific links via XMI to IBM Rational Rose, Sparx Enterprise Architect, and Telelogic (Popkin) System Architect for supporting the exchange of UML models to/from those tools. MooD also supports model/diagram exchange with Visio, data exchange with Microsoft Excel and Project, and reporting into Microsoft Word.

Model content held by content management systems and other applications can be modeled and accessed from MooD via the Knowledge/Resource Center. As well as folder and network sharing, Knowledge Activation supports WebDAV protocols for access to document storage. This enables the MooD toolset to access documents held in Microsoft Exchange and SharePoint Portal Server via industry-standard protocols. Any WebDAV-compliant server should be compatible.

Finally, it should be noted that the range of user-based Microsoft integration techniques supported by MooD is quite extensive. For example, the key software tool in MooD is an OLE client: This means that any OLE application (e.g., Word, PowerPoint, Excel, etc.) can be linked into a MooD model as an icon. The user can then invoke this application with a click of the mouse. In this way, presentations, documents, audio messages, numerical models, video clips, and many other forms can be directly associated with MooD elements or models. In addition, information can be pasted from a Windows application, via the clipboard, into any MooD model or definition. And any section of a Knowledge Map can be exported to Microsoft Excel, etc.

3. Analysis and Process Modeling

3.1. Enterprise and Organization Models

The MooD toolset supports rapid development of Organization and Enterprise models. Users create models two ways. The first is to use a pre-built model framework called a Transformation Blueprint. The second way is to create your own by creating a Knowledge Map. Both approaches result in an interactive visual model that can browsed, edited, presented, and published.

A Knowledge Map refers to a collection of terms, structures and notation, perspectives, models, and components that reside together in the MooD Repository. Linkages and integrity within and across Knowledge Maps are automatically maintained by the repository via Activation processes.

Transformation Blueprints are modifiable, pre-built Knowledge Maps that specify a collection of terms, structures, notations, and so on. Although blueprints are typically used as a starting point for model creation, they can also be applied retrospectively to existing Knowledge Maps, giving a fresh look or providing industry-standard compliance to current or proprietary work. Blueprints provide a set of established and tested business components along with knowledge resources and applications pertinent to a choice of perspectives and activities. They also include sector/industry best practices, knowledge management or quality initiatives, and intellectual property and proprietary processes.

The following example illustrates how a Transformation Blueprint that provides a Zachman Framework may be applied in practice as well as how users interact with the system. As shown in Figure 4, the blueprint is structured principally around the artifacts of the Zachman Framework (columns), and secondarily around the views (rows). The main orientation given to the user is the Zachman Framework with the familiar 6-by-6 matrix.
The matrix presents the “front window” into the models that have been defined, providing access to an artifact defined from a particular viewpoint. The columns of the framework are used as the principle structuring for the Knowledge Maps which, along with some essential concepts such as data cataloging and scenario modeling, provide a logical route through the Zachman Framework. The structuring can be viewed via an Explorer window as well as visually.

In order to provide a helpful basis for developing enterprise architectures, the Zachman Transformation Blueprint elaborates on the Zachman Framework in three significant ways. First, it provides a start point for each of the artifacts, offering industry best practice models that draw on business modeling methodology as well as technical modeling notations such as UML. Second, each cell can possess a complete drill-down through different modeling techniques to offer a rich and deep enterprise-scale modeling experience. Finally, it introduces cross-artifact linkage according to an underlying Zachman Framework relational model, thus enabling analysis as well as user browsing capabilities beyond the 6-by-6 matrix.

![Figure 4. MooD Transformation Blueprint for 6x6 Zachman framework with multiple views](image)

As illustrated in Figure 5, each cell is populated with an artifact – a model created from a set of elements. The models used in the blueprint make an assumption based on commercial, public sector, and military usage of architectures to provide an effective start point for enterprise architecture.

So, for example, the “people” artifact (column 4) from the viewpoint of “scope” (row 1) takes us into a template Business Ecosystem, capturing an external organization or business. This artifact gives a suggested start point for modeling the organizations of interest to the business, along with their principle relationships.

As illustrated in Figure 6, the Zachman Transformation Blueprint treats each element in an artifact as something that can be drilled into, to be modeled with an artifact of an appropriate type. As a result, each artifact can be a deep set of integrated model elements, basically a Knowledge Map in its own right, developed, for example, by using other Business Transformation Blueprints. Hierarchy views of elements highlight the logical positioning of any element within a model, while the 6-by-6 matrix provides a constant, top-level point of entry into the Enterprise Architecture.

So, for example, in Figure 6, a “Mortgage Services” process within the “Business Process Model” artifact (column 2, row 2) can itself be modeled so as to provide a set of use cases that implement...
this process. In general, the most appropriate set of models can be used and drilled down into behind each cell. The models may themselves be defined by other Business Transformation Blueprints, which can be merged into the repository, and linked to the appropriate cell.

Each artifact, such as a Business Process Model, contains its own native elements — for example, processes. However, an artifact can contain elements that have been defined within other artifacts — for example, business roles or data objects. Automated support for the reuse and referencing of these elements provides the key to integration — both horizontal and vertical — across the framework.

Figure 5. Clicking on a Cell (Row 1, Column 4) of the Zachman Framework reveals a model

Each element is defined once but reused many times in appropriate artifacts to provide a rich integration of components, all maintained at component level in MooD’s repository.

This reuse of common components continues on down through columns, and across rows, giving an intuitive browsing capability across the entire framework, “behind the matrix,” following links between elements. As a result, each user (according to their defined role), whether technical architect, business analyst, or sponsor, can begin their exploration of the architecture from their chosen perspective, and can easily access and determine how their contributions are being further developed by other roles involved in the development of the architecture.

Resource and Cost Modeling

Resource modeling can be accomplished using a Resource Association Model. For each component activity within an architecture, a model can be created to connect that activity with the resources that it uses or affects. In this way, all resources necessary for the enactment of an activity can be specified, including systems, information, policies, competencies, etc. Resource Association Models can also be used to identify resource usage by processes (i.e., which activities make use of a particular resource and for what purpose).

In raw MooD terms a resource is a reusable themed element. The method itself is soft and permits an arbitrary number of fields to be defined within the theme. So, criteria such as cost, schedules, and usage patterns can be seen as fields against the resource. There is no restriction in scope or number
of fields. Further, these can be Activated to get real values from operational systems. A common approach is to define the kind of criteria cited as measures and link them to the resource. This offers the added advantage that the criteria can be consumed, manipulated, and reported on by Performance Activation and Business Simulation – enabling us to ask relevant questions such as “What effect does an adjustment in usage pattern have on cost?”

**Mapping Organization Strategies to Performance Measures**
MooD Transformation Toolset 2006 provides methods and tools that enable performance-related initiatives to be aligned with the organizations that they seek to influence.

**Figure 6. A hierarchy of Activatable models and elements within a Transformation Blueprint for the Zachman Framework**

Organizational strategies are expressed via Transformation Blueprints or Knowledge Maps, which in turn contain models comprised of elements (objects) in the repository. Elements may contain one or more performance measures that can be statically or dynamically determined.

Any element in a model can have measures associated with it. Multiple measures are allowed, and may be associated with other elements, combined, and presented on dashboards (which are created with models in MooD). MooD supports measure aggregation in the classic sense through relationships to whatever level is of interest. Measures can be incorporated into dashboard views at any level. In MooD 2006 measures are associated with a single theme. The next major release will see this restriction lifted, and measures will live anywhere within the repository with an arbitrary number of user-definable attributes and states.

Because of MooD’s Process Activation capabilities, if a process is activated at runtime as a web service, it could in itself be activated to provide real data back into MooD via Performance Activation. This really is a great end-to-end story: Design a process, describe how that process delivers against KPIs or other measurements, and have the process monitor and report against itself at runtime.

**Managing Process Portfolios**
In MooD, projects are orchestrated and managed around the concept of Knowledge Maps, which basically comprise the complete collection of models and their associated definitions and information
(reports, documents, etc.), including any external, linked resources for a given process or project scope. In addition, Knowledge Map has collections of themes that enable and support modeling and design from a particular business perspective – for example, “Customer Service” or “Product.” Themes provide a way to create enterprise architectures and process models that comprise multiple perspectives to address the interests of different user communities.

Knowledge Maps are stored in MooD’s underlying repository, providing traceability and dependency analysis across the Knowledge Map. (See Section 2.3 for more on MooD’s repository features and capabilities.)

3.2. Defining Processes

Defining Processes

Together, Process Capability and Process Activity Models constitute the process architecture of a business context. Process Activity Models describe activity dependencies and collaborations in the form of a workflow. Process Capability Models elaborate process business activities, describing specific transactional requirements. As depicted in Figure 7, the process architecture is comprised of a layered network of business capabilities and relationships that may have assigned owners (roles). Layering of models is crucial to ensure that a design is understandable to the relevant business area and its enactors. This coherent visualization can be used for communication within the business to support implementation, induction, and the operation of processes.

Figure 7. Process Architecture achieved through elaboration Process Capabilities and Activities

A process is represented by an element or collection of elements and activities (representing subprocesses). Processes defined by capability and activity models are consolidated through a definition of key elements. These elements include purpose statements, responsibility definitions, and
performance measures. These elements can be supplemented with information on object inputs and outputs with additional descriptive fields specific to the business' requirements.

**Process Information Storage and Integrity**

MooD provides a number of features designed to support and enforce model/process integrity. These include:

- The ability to track development and model iterations by comparing variations of an element.
- Propagation of name changes and model consistency (both within and between levels) occurs automatically according to built-in integrity rules.
- Dependency analysis functionality shows immediately which processes may be affected by a change to a policy, or which systems support a particular process, or the processes for which a role is responsible.
- Sophisticated searching and reporting capabilities in the repository, including the ability to search for elements that satisfy specified criteria.

In addition, MooD's multi-user support includes individual user login, ownership at the model and component level, multi-user access control, shared ownership, and repository synchronization via locking and mediation.

**Graphical Notations**

MooD supports a range of techniques and corresponding notations, ranging from visioning techniques such as stakeholder analysis and current-state / future state analysis, to more formal UML or IDEF notations. By means of themes, it is possible to establish and allocate the kinds of techniques and models needed, in terms of components and visual properties.

### 3.3. Subprocesses and Activities

**Handling Subprocesses and Activities**

As described in Section 3.2, a process is represented by an element or collection of elements and activities (representing subprocesses). In MooD, Process Activity Models cannot be nested. If nested workflow-style models are required, it is necessary to use Process Capability Models to construct models with workflow styles and structures, serving as higher-level activity models.

A Process Capability Model describes the elaboration of a business context or process. It shows the constituent processes – direct child processes, re-used components from other models, and inter-relationships. Any child process can be further elaborated as required. This can be done to any number of levels of subprocesses. The root-level processes within each theme describe the business contexts under elaboration and the agreed strategic position defining purpose, relationships, and performance. Relationships drive an understanding of the business’s opportunities and constraints. Processes defined by capability and activity models are consolidated through a definition of key elements. These elements include purpose statements, responsibility definitions, and performance measures. MooD provides a facility to convert an Activity Model to a Capability Model and vice versa.

**Defining Activities**

Creating and defining an Activity model in MooD takes place in a visual manner and is similar to creating a Process Capability Model. Activities can be opened and defined with purpose, responsibilities, and measures in the same way as processes. And relationships (requirement links) between activities are drawn to show the flow of objects and information through the activity model.

**Documenting Decision Rules**

A logical process flow is defined in MooD via a series of drag-and-drop activities, reusing existing capability, creating new, adding decision points, etc. (Figure 8). As the visual construction activities
advance, the real-world web service operation enactments are selected and dropped onto the relevant execution points that serve the orchestration (these are the operations that have already been discovered and mapped from UDDI, for instance); stub services and parameters can also be created (with default values). Once operations and parameters are available, decision rules can be created. The user might, for example, select a “Choose” decision point [symbol] on the model then edit the rule. The rule will be scoped according to the operations and parameters visible to the decision point. In the current version of MooD, decision rules have a 1-to-1 association with their decision point. Salamander plans to lift this restriction to enable users to reuse decision rules throughout their orchestrations (subject to the same scoping considerations).

Figure 8. Documenting decision rules in MooD

Rules Entry
Rules are entered using a dialog that ensures that correct scope and integrity are preserved. Even within the dialog the user has the option to drag-and-drop the available parameters onto the rule for accuracy, gaining a productivity boost at the same time. Once entered, rules can be made visible on the model with the same flexibility in style as other model information.

Activity Costs, Resources, and Time Data
Any process may have a role accountable for its successful performance. This role is usually referred to as the “process owner.” In addition, there may be many other roles with delivery responsibilities for the process. Process owners will normally be assigned to processes at a high level. Subordinate process owners might be assigned to specific process responsibilities within a high level process. Assignment of process owners consequently defines a hierarchy of accountability. If no subordinate process ownership is specified, the higher level accountability for the parent process is assumed to apply – i.e., accountability is inherited. At the level of activity, the role assigned refers to the role responsible for its enactment. These are sometimes represented as swim lanes that can offer a useful visual indication of role interactions during the course of a process. It is possible that the role assigned as being accountable at a high level is also responsible for performing the required activities.

A Process Insight Model can be created for any level of process (excluding activities). The Insight Model is constructed automatically by MooD as follows: For the selected process, all descendent
activities are identified; then for each activity, the objects and services used in the activity model are accumulated, and added to the Insight Model. Each time an Insight Model is opened, it is automatically refreshed. In this manner, the Insight Model gives indication of the behavior and scope of the process by revealing the resources and services used in its implementation.

3.4. Simulation

Simulation Capabilities
Salamander uses Simul8 Corporation’s Simul8 simulation package for conducting simulations and analysis on business processes within the MooD toolset. It should be noted that the Simul8 technology is tightly integrated with the MooD Transformation Toolset via a software add-in (the Business Simulation add-in), and is launched from the MooD process Explorer tool. To assist end users to build and conduct simulations, Simul8/MooD also offers various wizard-driven templates.

Integration of MooD and Simul8 allows users to generate dynamic models for selected processes and components within a Knowledge Map. These models can then be simulated and analyzed using Simul8 in order to assess performance and other model design considerations. Basically, the interaction between the two tools consists of using MooD for visualizing and defining the business model, and Simul8 to animate and quantify the operational implications for deployment.

MooD and Simul8 co-operate within a cycle of business transformation with development of MooD and Simul8 models taking place in parallel. The cycle begins by generating a model for selected components of the business design within the Knowledge Map. The next steps involve generating and refining the Simul8 model, using animation and analysis within Simul8. The model can then be run to explore the performance implications of the design. Results from the Simul8 run are available within MooD Performance Activation for reporting, analysis, and viewing within the context of a Knowledge Map. Simul8 also allows a presentation version of the model to be generated.

A Simul8 model appears identical to the MooD model it represents, reflecting the MooD activities, the links that connect them, and the roles that carry them out, etc. Thus, basically, when the Simul8 model is run, it is like the MooD model brought to “life”: Work items are shown (i.e., animated) flowing through the model; resources are assigned to the tasks as they are performed; and Simul8 queues fill and empty to show current work loads.

Analytic Capabilities
MooD/Simul8 simulations offer animated analysis capabilities that are useful for a number of tasks, including

- Analyzing and optimizing performance characteristics and constraints of activities
- Analyzing the use/assignment of additional resources of various kinds (e.g., people, systems, etc.)
- Analyzing and predicting the way in which work items flow through a process

Real-time Data Utilization
Simul8 can be coded to draw data from external sources using the embedded Visual Logic engine. This can read data from sources such as Excel or SQL. In addition, MooD Performance Activation will collate and compare expected data from simulations with actual data from other real time data feeds.

Model Distribution and Simulation on Enterprise Networks
Simul8 also allows a presentation version of the model to be generated. Presentation models are self-contained animation models. They can be distributed independently of MooD (i.e., to users who do not have access to the MooD toolset as stand-alone Simul8 simulations) to different process stakeholders.
Statistical Fit/Data Analysis
Results data within the Simul8 engine can be captured, and then analyzed. Specific results on resource utilization, process timings, and many other data are selected as required. Results may be exported to Excel for further analysis using the Visual Logic tool. Trials may also be run, providing statistic spreads of performance over several individual runs.

Capture and Reporting of Simulated Metrics
Results from the Simul8 run are available within the MooD toolset for reporting, analysis, and viewing (e.g., on a performance dashboard as a prediction indicator). Results from simulations are saved and “remembered” by MooD. As noted previously, MooD/Simul8 allows a presentation version of the model to be generated for running and viewing by various process stakeholders.

4. Business Process Methodologies

4.1. Business Process Methodologies
In addition to Salamander's own methodology for organizational transformation, the Business Transformation Lifecycle, MooD supports a variety of methodologies, including Balanced Scorecard, Six Sigma, as well as enterprise architecture methods (whether proprietary or public domain). Other supported methodologies include PRINCE for project management or health care pathway development.

4.2. Six Sigma Support
See Section 4.2.

5. Report Generation and Document Management
MooD Transformation Toolset offers a number of report generation options. Reports can be generated for Microsoft Word and Excel by default. In addition, the toolset’s publishing and integration capabilities support custom reporting to almost any environment and format.

Matrix Reports
Of the various reporting options available in MooD, Matrix Reporting is considered the most insightful in terms of the information that it can uncover. Matrices offer a powerful and expressive way to discover and communicate properties and relationships. Matrices can be used as

- Analytical tools to identify patterns, overlaps, and gaps
- Informative communication tools, complementing models and other reporting options, and
- Data entry tools

Matrices are configurable, structurally and visually, and they can be published to the web. A matrix view is automatically kept up to date with changes to relevant information (e.g., references or property values). Some matrix types also allow data entry, providing an efficient means to capture referential or property information. Matrices, like models, belong to an element (such as a capability, business process, system, or measure), and any element might have any number of matrices, meaning that

- A matrix can be created as an alternative “look inside” for an element that also has another model. For example, in addition to an activity model, a business process might also offer a matrix view of its components and their mappings. Alternatively, an element can be created solely to hold a more general matrix that is then re-used on appropriate models.
- In a matrix, the elements in rows, columns, and (depending on matrix type) cells are defined by queries. This is key to understanding their flexibility and power. There are six different
types of matrix, each of which offers distinctive structuring properties. Many of the visualization options are common across all types. In addition, it is possible to create a copy of an existing matrix, which can provide a useful starting point for creating another, similar matrix.

**Document Management**

MooD's Knowledge Center concept allows documents to be stored and managed from a central location. Typically, these resources are held in folders or document management systems. Basically, the Knowledge Center serves as a reference point for everything inside it. The Knowledge Center is given a logical name, a physical location, and a specification of how the resources held in the Knowledge Center will be accessed from the published Knowledge Map – from its current location, from another location, or copied.

**Publishing**

MooD provides two options for publishing information – a Web Publishing Wizard and an “Active Publisher.”

**MooD Web Publisher Wizard.** Publishing repository content is simple with the MooD Web Publisher Wizard, which is accessed from the Repository Manager. Basically, the Wizard takes a snapshot of a MooD repository and creates a set of HTML web pages that can be viewed through a browser. Generated pages represent the contents of a repository at the time of publication. When subsequent changes are made to the repository, the updated web pages can be republished. Published content can also be customized according to a publishing template that can be defined by the MooD administrator, etc.

**Active Publisher.** Active Publisher provides template-driven publishing features similar to those provided by Web Publisher, but includes the addition of several very important capabilities. In addition to HTML, JavaScript, and optional ActiveX client-side controls, Active Publisher adds ASP.NET server-side publishing. As a result, Active Publisher provides an “always-on, demand-driven” publishing capability in which any changes made to the MooD Repository (and its activated resources) are immediately visible in the next page download or refresh. Consequently, any updates to the Repository can be applied from the Web Browser directly, or through mediation (approval process). And reports are generated for Microsoft Word and Excel by default.

Active Publisher also supports performance monitoring and incorporates a Microsoft Outlook style configurable rule-based alerting and escalation capabilities. When business events are triggered – normal flow, external, or exception based – Active Publisher will execute pre-defined responses. This might be an email alert to a key decision maker or the execution of a web service. Sensitivity controls can be configured to ensure that responses are not raised unnecessarily. Similarly, escalation rules can be applied in the event lower-level responses are not having the desired impact.

Finally, decision engine support, where configured, is available to web users.

**6. Development Environment**

**6.1 Language of Tool**

The Toolset is written in a mixture of C++, C#, XML, XSLT, XHTML, and JScript.

**6.2 Product Support, Maintenance, and New Versions**

Salamander is a UK-based organization with offices in York, Reading, and London. A Customer Services Team offers telephone/fax/email based service to support customers in respect to technical or usage difficulties. Support is charged at a percentage of license price, and is renewed annually. In response to an enquiry, the typical outcome is advice to address the customer’s difficulty.
Occasionally this will involve a site visit, but normally is handled over the telephone or by email. Where the difficulty implies a fault in the product, this is transmitted to the Development Team for a rapid hotfix or for scheduled inclusion within a forthcoming release.

Salamander issues two classes of release – major releases and intermediate service packs. Salamander typically carries out two major releases of MooD Transformation Toolset a year, with one or two intermediate service packs. Users with support agreements are entitled to service packs at no extra charge. Major releases can be purchased by existing license holders at an upgrade cost. Alternatively (and more commonly), an annual upgrade service is offered, charged at a percentage of license price, which automatically provides all major upgrades at no additional cost.

Some organizations make use of a custom service whereby Salamander also undertakes additional support responsibilities, including the operating platform for the MooD software (e.g., database server). All such cases are negotiated specifically with the customer.

7. Software Modeling and Code Generation

7.1 UML Model Generation

MooD Repository content can be imported and exported using XML. Equally, external data can be imported using XML into a MooD Repository. The data exchange is facilitated by MooD’s Business Integration Engine Services and Activators. The Activation process creates an XML file; this being a standard data format, that in turn is formatted into the desired output (e.g., Excel). For example, Salamander offers the MooDXMI Activator for extracting content for generation into a standard format suitable for input into UML compliant systems design tools such as Rational Rose. The whole process can work in reverse so external data is converted into XML and then moved into a MooD Repository.

7.2 BPEL Generation

Salamander has built BPEL import and export Activators for some of the major BPEL platforms. (For more on Activators see Section 2.4 – Integration with Other Products.)

MooD provides a series of wizards that allow business analysts and architects to create process flows that will satisfy the requirement to produce a BPEL definition, describing the logic associated with the flow of web services, including such constructs as `choose` and `while`.

Web services are associated with process models by first discovering them and importing them into the modeling environment. Directory services such as UDDI provide a means for Process Activation to view and compare existing web services and build a link to their source for future use.

Once a web service is imported for use within the modeling environment, Process Activation interrogates the service to find the operations that it offers, updating the MooD repository with all the information required to make the service available within any model. Once imported, process activities can then be activated, associating them with the appropriate service by simply dragging onto a process element in the model.

During the design and development of Process Activation, usability has been a key factor. Process Activation for MooD enables the business user to harness the power of web services and enables the user to produce fully functional business process models quickly and easily.

Where web services do not exist to deliver the required functionality the specification can be produced within the toolset. This specification, which complies with the industry standard description language, WSDL, is then used to produce a stub of the service, which can be exported into a development environment for ongoing design and build by the development team.
Process Activation also supports the export of orchestrations for deployment within a production environment via BPEL. Process Activation allows BPEL to be generated for the following “best-of-breed” server workflow engines (production environment):

- IBM WebSphere Business Integration Server Foundation
- BEA WebLogic
- Oracle BPEL Process Manager

8. Templates and Frameworks

Salamander offers for use with MooD various pre-built frameworks that are built into its collection of Transformation Blueprints. Blueprints can be methodological, providing a framework within which a particular methodology process is to be followed (e.g., Balanced Scorecard, Six Sigma, enterprise architecture methodologies, etc.). In addition, Transformation Blueprints include standard collections of perspectives aligned with the kinds of models and corresponding components that these require.

9. Systems Administration and Security

Access to and manipulation of a MooD Repository is controlled through user identification and ownership. Only authorized users may open a Repository and view its contents. Only the owner of an element (or Administrator) is able to modify it. The element owner is also able to control the publication rights associated with the element. Shared ownership is available for situations in which multiple users are to be allowed update access to common elements.

Repositories have two general categories of users. Those who work with the Repository and are the owners of its components, known as “Users,” and a distinguished user, known as the “Administrator,” who is responsible for setting up, administering, and deleting the Repository and its users. The Administrator has full access rights to all aspects of the Repository. Mediators are a subcategory of user that have approval and denial rights relating to updates and audit histories. Every repository element has an audit history that records information pertaining to the event, the user id of the person who has been working on an element, the date and time, and the workstation.

10. Scalability

Horizontal scaling may be achieved by distributing toolset components across different physical servers and clients. Neither the client or server components within MooD Transformation Toolset impose any set limits to scaling. Scaling is limited by the underlying platform and database licensing and resource limit.

11. Platforms

MooD client software runs on Windows XP Professional and Windows 2000 Professional.


Apache Web HTTP Server is supported for the publication of static Knowledge Map content. Active Publisher requires Microsoft ASP.NET and is therefore not supported on the Apache platform.
MooD’s browser software of choice is Microsoft Internet Explorer 5.01 or later. There is general compatibility with some of the more popular open-source browser programs, but Salamander does not endorse their application across the full MooD Transformation Toolset.

The MooD repository is created in Microsoft SQL Server (versions 6.5, 7.0, 2000, and 2005), Microsoft Access (recommended for single-user client-side repository development), and Oracle 7.3, 8i, 9i, and 10g.

Salamander is fully committed to supporting industry best practice and standards compliance, such as XHTML and CSS.

12. Pricing

The MooD Transformation Toolset has no per-unit pricing; pricing is derived based on the total solution, which includes consulting, Transformation Blueprints (i.e., solution frameworks), the required software components needed to support individual solutions, and support.


13.1 Company Background Information

The Salamander Organization is a privately held company based in York, England. Founded in 1996, Salamander is considered a pioneer in the development and integration of enterprise architecture and business transformation technologies.

Salamander’s strategy combines the key solution areas of corporate governance, transformation, and best practices, as well as enterprise architecture and business process outsourcing, with industry focus and expertise. In addition, Salamander supports consulting organizations across these and other markets by providing instruction, methods, and tools in consulting and transformation engagements.

Salamander’s efforts are focused primarily in the United Kingdom and the European Union; however, the company’s MooD transformation Toolset is used and supported worldwide.

Salamander has approximately 55 employees. The company’s annual turnover is about £6.2M (approximately US $10 million). The MooD Transformation Toolset is used by more than 200 organizations worldwide, representing some 25,000 installations.

In 2004, Salamander was awarded the UK’s most prestigious business innovation award, the Queen’s Award for Enterprise. Salamander has also achieved industry recognition from the IEE innovation award (2002), DTI Smart Award (2003), and British Computer Society Technology Finalist award (2005). In 2006, Salamander achieved a place in the Deloitte Fast 50 Technology Companies list.

13.2 Positioning

As described in this report, the MooD Transformation Toolset is a feature rich environment whose capabilities are extensive. As a result, it can support a range of EA modeling and BP change activities, including:

- Enterprise architecture modeling and analysis
- Process modeling and analysis, redesign, and improvement
- IT/software development/automation activities
- Human performance improvement initiatives
- Development of management and measurement systems
- Detailed process modeling and analysis (simulation)
One of MooD’s many key strengths is its ability to model organizations in their entirety. In short, MooD provides capabilities for defining and modeling all of an organization’s components and their associated relationships, including organization, strategies and goals, roles, skills, personal development, technology infrastructure, business processes, and information.

Salamander has also paid considerable attention to making MooD a highly intuitive environment whose functionality can be configured to the appropriate business perspective or view. As a result, MooD supports the entire user community with the appropriate feature sets and functionality, thereby ensuring that the tool is easily accessible to general business users and more technically skilled analysts and architects.

Salamander’s belief that communication is an essential component to the success of any transformation effort is reflected in MooD’s ability to function not just as a modeling tool, but also as knowledge and information management hub that provides a centralized mechanism for integrating, sharing, and communicating architecture and design information. Consequently, in addition to being able to publish modeling information to the internet or intranet, MooD can serve as an interactive presentation tool. Thus, it can be used to drive workshops for business process improvement and other transformation initiatives. This is achieved through strong repository capabilities designed to maintain the integrity of modeling information while at the same time supporting multi users with the appropriate perspectives and the ability to carry out team development.

13.3 Product Training

Salamander specializes in helping organizations transform, and works with partners who are supporting organizational transformation. It also offers product training for software development, integration, and use of the MooD Transformation Toolset.

13.4 Business Process Consulting

Salamander provides a broad range of services to assist organizations build and deploy Knowledge Maps, these services include installations, template development and version upgrades, integration development and customization, and hosting of published Knowledge Maps.

Salamander also works closely with its various distribution partners, many of whom also provide regional or global support for the MooD Transformation Toolset.

14. Case Study: The Program Architecture and Coherence Toolset Project

The Programme Architecture and Coherence Toolset (PACT) project is a constituent project of the UK Ministry of Defence (MoD) Information Exploitation (IX) program. Its aim is to test the applicability and usefulness of information management techniques and modeling using the MooD Transformation Toolset, to a variety of change initiatives across Defence. The project has the twin objectives of proving or otherwise the value of such an approach and of initiating its use to deliver value in as wide a selection of scenarios as possible.

The PACT project initially identified a group of 12 change initiatives that became, from the point of view of the IX Programme, pilots for PACT. One such initiative is intended to develop a single model of support chain processes showing the interactions between leading industry suppliers and MoD logistics and purchasing operations. To date, each time new capability was sourced, a new supply chain design was created, almost from scratch. At base, a supply chain design consists of a process for information management and flow, and the definition of the information that will be involved. While some re-use of previous designs occurred through informal process (relying on individual and, sometimes, on supplier corporate memory), no reliable, formal capability was in place.
to capture and define designs for re-use. This also had the added effect that there was no opportunity to recognize, develop, or re-use best practice.

While formal benefits’ modeling for the project has not yet taken place, the rationale for the work is based on a strong conviction that efficiency and effectiveness improvements are available. This is reflected in the exceptionally strong level of sponsorship for the work amongst senior MoD personnel and all leading industry partners of the MoD. The scale of backing for the initiative is best illustrated by consideration of the participation of industry partners who have contributed in the order of 2000 person days to the work without charge. Clearly, one result will be a reduction in the costs experienced by industry partners in supply chain delivery; however, this is widely seen as a vote of confidence in the project.

Given the volume of sourcing that takes place annually through process subject to this work (approximately £760 million), small percentage efficiency savings will deliver significant absolute cost avoidance and acceleration in FOC. Improvements in quality will result in more operationally efficient support chain processes to add to the benefits related to designing and implanting the processes.

The project is divided into 3 work streams. PACT is being used on the Business Process and Information Requirements (BPIR) work stream. The BPIR work is due to complete in mid April, with the web-publishing of the BPIR model using MooD technology. This resource will provide a comprehensive reference for MoD and industry partners describing a unified, re-usable supply chain design (Figure 9). It will also provide a governance mechanism, by applying the latest MooD capability to capture update information and manage its adoption.

Benefits Model
In developing a benefits model for PACT on DLO Coherence BPIR, two top level benefits types were identified – benefits that are realized during the project through use of PACT rather than core...
tools, and benefits that can be expected from the coherence project as a whole, which can, at least in part, be attributed to the use of PACT either with PACT as either as a direct or as an indirect. In discussions with stakeholders, it was widely agreed that the BPIR work has required significantly greater time to deliver a useful model than is being achieved using PACT. One estimate is that the work would take 4 times as long to reach this point. This leads to an estimate of cost avoided for MoD staff equaling 9 months full time salaries for the MoD team.

The use of PACT in BPIR has reduced the duration of the work by 9 months when compared to undertaking the project using core tools. This provides cost avoidance equal to at least 9 months capitation for the MoD team members and 9 months cost of project management, amounting to £308k.

Reduction in BPIR duration means that the model is available for use in April 2006, compared to January 2007 which would have been the case with core tools only. As benefit is being evaluated for IX up to a fixed date, this acceleration has the effect of providing additional benefits equal to ¾ of the aggregate annual operational benefit of the BPIR work, and this additional benefit is directly attributable to PACT. At present, benefits modeling for the BPIR work as a whole is not mature. However, it has been suggested by team members that a saving of 1% in overall sourcing costs for in scope activity is a realistic and, in fact, conservative target. This view was sanctioned by the lead stakeholder as sponsor of the BPIR work stream. At present, it is not considered justified to include this assessment in the quantified financial benefits model for the work. However, it is important that progress on development of a benefits model for BPIR continues and that IX monitors this development. This is required because the potential benefits attributable to PACT are large. Calculating on the above basis, they amount to 0.75 x 1% of approximately £750 million annually, that is £5.6million.

The use of PACT in BPIR has reduced the duration of the work by 9 months when compared to undertaking the project using core tools. This means that the model is available for use 9 months earlier than would have been the case with core tools, bringing in stream the BPIR benefits as a whole earlier, as well as providing additional benefit equal to ¾ of the average aggregate benefit for BPIR. Current calculations suggest that this amounts to approximately £5.6million. This figure has not yet been included in the benefits model.

A more conservative estimate of BPIR’s overall benefit is currently available and is robust, in that it was provided by and verified by the lead stakeholder as sponsor of the BPIR team. This states that the cost avoidance available annually from use of the model in reduction of costs per project will be at least £20k. This leads to an annual cost saving of a minimum of £400k resulting from 20 projects. In turn, this leads to a benefit attributable to PACT of 0.75*£400k = £300k. This figure is currently included in the benefits model.

The quality and usability of the BPIR model, produced, published, and maintained using PACT, differs from a model produced, published, and maintained using core tools. Previous modeling work suggested that if MS PowerPoint were used in the order of 100 slides would be required to present the information and a mechanism would need to be developed to capture and manage dependencies between information elements. Accuracy and currency of the model would be significantly compromised, and this would impact on its being fit to deliver all of the benefits expected from the BPIR work. Again, at this point, not only is the BPIR benefits model not available but also the level of compromise involved has not been agreed. An estimate that the efficiency impact of the work could be reduced by a quarter to a half of the modeled value is yet to be verified. The use of PACT in BPIR is necessary in production of a support chain model that can deliver all of the efficiency and effectiveness benefits expected. Use of core tools only would reduce the efficiency benefit by a factor, currently estimated at 25% minimum. This value of benefit has not yet been included in the benefits model but, if calculated using current estimates, would equal 1.875 million annually. Re
applying the more conservative estimate of BPIR benefit used above (£400k annually), this benefit can be shown to be £100k. This value has been included in the benefits model.

MooD technology provides a cost effective method for model publishing and maintenance when compared to use of core tools. Additional provision would be required to publish the BPIR model in a non-PACT environment, attracting technology and personnel costs. At present, this benefit has not been quantified. MooD provides a more cost-effective method for model publication and maintenance than would be available with a PowerPoint model, for which additional technology and effort provision would be required. No estimate is yet available for this amount, and no value has yet been included in the benefits model.

15. **Company Offices**

The Salamander Organization Ltd. home office is located at

Innovation Close, York Science Park,
York, YO10 5ZF United Kingdom.
Tel: +44 870 161-1700.
Fax: +44 870 161-1701.
Web: [http://www.tsorg.com](http://www.tsorg.com)
Primary Contact: [info@tsorg.com](mailto:info@tsorg.com)