# **Tool Mentor - SOMA Modeling Environment (SOMA-ME)**

#### **SOMA-ME - An Introduction**

SOMA Modeling Environment adds Service Oriented Modeling and Architecture (SOMA) support to IBM Rational Software Architect (RSA) and IBM Rational Software Modeler (RSM). It is distributed as an update site feature for RSA and RSM.

#### **Features**

- Extends UML meta-model for SOMA methodology "UML 2.0 Profiles for SOMA"
- Provides Model Template suitable for capturing output of SOMA according to the flow phases and activities.
- Automates repeatedly performed tasks in SOMA with MDD enhancements
  Transformations and Patterns.
- Reduces effort and duration drastically with SOMA Work Product Generation capabilities.
- Ease modeling with User Interfaces to create SOA/SOMA model elements.
- Model Validation for completeness and consistency.

### 1. SOMA Identification

#### 1.1 Domain Decomposition > Functional Area Analysis

SOMA-ME provides stereotypes to model Domain, Functional Areas and Functions (a.k.a. Business Capabilities) in Rational Software Architect (RSA).

It provides a model to model transformation to generate functional area analysis artifacts from CBM meta-model based models.

It can import tab separated files containing functional area analysis artifacts and create corresponding model elements. It also provides user interface to create the model elements manually.

It can create candidate services from fine-grained "Functions". It sets service identification technique used in stereotype attribute and maintains traceability link to the function.

#### 1.2 Domain Decomposition > Process Decomposition

SOMA-ME provides stereotypes to model business process elements in RSA.

When business process models are available in WebSphere Business Modeler (WBM) format, SOMA-ME can leverage "Business Contract Modeling" based WBM - RSA native integration. Please note that this exercise requires Modeling Integration > WebSphere Business Modeler optional component installed in RSA7.

In the absence of business process models in WBM format, user interfaces provided by SOMA-ME can be used to create business process elements.

It can create candidate services from process, sub-process, task or decisionbox. It can capture service identification technique used in stereotype attribute and maintain traceability link to the business process model element.

### 1.3 Goal-Service Modeling

SOMA-ME provides stereotypes to model Goal, Sub-Goal and KPI identified from Goal-Service Modeling in RSA.

When goals are available in Rational RequisitePro, SOMA-ME can leverage the "Requirements Explorer" view based RSA-RequisitePro native integration. Please note that this exercise requires **Team Integrations > Rational RequisitePro Integration** optional component installed in RSA7.

In the absence of goals in Rational RequisitePro, user interfaces provided by SOMA-ME can be used to model the Goal and Sub-Goal in RSA.

User interfaces provided by SOMA-ME can be used to create KPI of the goals.

SOMA-ME can map the Goals / Sub-Goals to Candidate Services in "Service Portfolio". It can create candidate services from Goals / Sub-Goals which can not be mapped to candidate services. In addition, it captures service identification technique used in stereotype attribute and maintains traceability link.

# 1.4 Existing Assets Analysis

SOMA-ME can leverage Subsystem stereotype of Default Profile to model existing systems, applications and information management systems identified from Existing Assets Analysis. It provides stereotypes to model existing services and data models.

It can create Service Messages from Business Items in WBM models and maintain traceability link to the business item.

It provides import facility for assets stored in WebSphere Services Registry and Repository (WSRR). Currently we have to download the wsdl and xsd to local file system and then import using SOMA-ME. Forthcoming version of SOMA-ME will provide integration with WSRR for search and retrieval of the assets.

# 1.5 Service Hierarchy

SOMA-ME uses diagrams to create functional relationship (grouping) of candidate services. The candidate services in Service Portfolio can be dragand-dropped to diagram corresponding to its functional grouping.

# 2. SOMA Specification

### 2.1 Service Exposure

SOMA-ME provides user interfaces to capture Service Litmus Test results. The context menu approach can be used for capturing SLT results per service. The SLT view approach can be used to capture SLT results of multiple services in a tabular fashion. When Exposure Decision is set to true in SLT UI, SOMA-ME creates corresponding exposed services in Service Exposure package. It also creates traceability link between candidate service and its corresponding exposed service.

#### 2.2 Service Dependencies

SOMA-ME uses UML Dependency notation to model dependencies between services. In a typical engagement, service dependencies will be modeled in multiple diagrams.

#### 2.3 Service Composition

SOMA-ME uses UML Composition notation to model composition relationship between services. In a typical engagement, service composition will be modeled in multiple diagrams.

#### 2.4 Service NFRs

SOMA-ME uses a simple text property in "Service" stereotype to capture Non-Functional Requirements (NFR) of the exposed services.

#### 2.5 Service Messages

SOMA-ME leverages Message stereotype of "UML 2.0 Profile for Software Services" to model input and output messages used by the services. When industry data models and enterprise data models are available in xsd (XML Schema Definition) format, it can be imported to WBM as Business Items and SOMA-ME can import Business Items as service messages. SOMA-ME can also import XSD directly and create service messages. In the absence of xsd, user interfaces provided by SOMA-ME can be used to model service messages.

# 2.6 Service Operations

SOMA-ME leverages UML Operation to model service operations of exposed services. Service message consumed by the service operations can be modeled as parameter type of the operation. Service message returned by the service operations can be modeled as return type of the operation.

### 2.7 State Management

SOMA-ME uses simple text property in "Composite Service" stereotype to capture state management decisions of exposed services.

#### 2.8 Subsystem Analysis

SOMA-ME leverages UML Subsystem stereotype of Default Profile to represent subsystems identified from Sub-system Analysis.

# 2.9 Component Identification and Specification

SOMA-ME provides stereotypes to model Service Components, Functional Components and Technical Components identified from Component Identification activity. Technical component has an enumeration attribute for allocating it to Service Components Layer (default) or any of the cross-cutting layers of SOA Solution Stack.

#### 3. SOMA Realization

#### 3.1 Service-Component Allocation

SOMA-ME provides user interface to map the exposed services to service components.

#### 3.2 Realization Decisions

SOMA-ME uses simple text property in "Service" stereotype to capture realization decisions of the exposed services.

# 3.3 Component Layering

SOMA-ME provides "Service Model to SOA Solution Stack" transformation to generate initial version of SOA Solution Stack from Service Model.

# 4. SOMA Implementation

#### 4.1 WPS Artifact Generation

SOMA-ME provides a transformation to generate WPS compatible SCA artifacts (bpel, wsdl, xsd, import, export) from SOMA Service Model. It leverages "UML to SOA" transformation of RSA.

# 5. Work Product Generation

#### 5.1 Generate Work Products

SOMA-ME can generate Service Model and Functional Areas work products from the UML Model. Support for other SOMA work products is under development.