Service-Oriented Modeling Framework (SOMF) for Business & Technology

An SOA Implementation Framework

Service-Oriented Discovery & Analysis Example
“Service-oriented modeling is a software development practice that employs modeling disciplines and language to provide strategic and tactical solutions to enterprise problems. This anthropomorphic modeling paradigm advocates a holistic view of the analysis, design, and architecture of all organizational software entities, conceiving them as service-oriented assets, namely services.”
What is Service-Oriented Modeling?

Is About
- Behavior, Structure, and Relationship Inspection
- Construction of a Virtual Computing World
- Hypothesis Verification
- Feasibility and Capability Analysis
- Motivation Assessment
- Simulation

Furnishes Modeling Disciplines that Provide
- Modeling Process
- Modeling Standards
- Modeling Best-Practices

Service-Oriented Modeling

Offers a Language that Provides
- Simple Vocabulary and Taxonomy
- Intuitive Syntax
- Universal Terminology

Contributes to
- Loosely-Coupled Computing Environment
- Software Asset Reusability Enhancement
- Reduction of Time-to-Market
- Alignment of Business and IT Organizations
- Loosening Structure of Silo Organizations
- Strategic and Tactical Organizational Solutions
- Software Assets Consolidation
- Expenditure Reduction
Service-Oriented Modeling Framework

Conceptual Environment
- Service Conceptualization Discipline
- Conceptual Architecture Discipline

Analysis Environment
- Service Discovery & Analysis Discipline
- Business Integration Discipline
- Analysis Service

Logical Environment
- Service Design Discipline
- Logical Architecture Discipline
- Design Service
- Logical Architecture

Solution Service
Physical Architecture

Service-Oriented Architecture Modeling Framework
Version 1.0
Service Metamorphosis

Service Lifecycle

Design Time
Run Time

Conceptual Service
Analysis Service
Design Service
Solution Service
Management of Service Life Cycle States

**Promotion** – increasing business functionality, funding, and reuse

**Decommission** – retiring and terminating execution

**Demotion** – reducing functionality and reuse, and limiting budget allocation

**No Change** – Service continues to operate in its current state
Service Structures
Service-Oriented Discovery & Analysis
Service-Oriented Analysis Modeling
Service Discovery & Analysis Toolbox

Service Typing  Granularity Analysis  Aggregation Analysis  Decomposition Analysis  Unification Analysis  Intersection Analysis  Subtraction Analysis
Modeling Analysis Services

It’s Time to Play!

Revealing a Service Ecosystem...

- Understand Service Evolution & Metamorphosis
- Understand Service-Oriented Development
- Understand Service Life Cycle
- Understand Service-Oriented Asset Management
- Understand Service-Oriented Governance
- Understand Business & Technological Traceability
- Record Analysis Decisions & Train of Thought
I have always wanted to own a bank. So my initial step was to provide Checking Account offerings to my first clients…
But my clients demanded more than a Checking Account service. I added a Savings Account service to my line of business!
To further generalize my business and expand it to other territories, I grouped these services under the name Banking Account Service, so in the future I can add more banking services...
Here I even suggested to accentuate Customer as the most important aspect of my business. Thus, I added Customer Accounts offerings to provide customer support, internet access, and more.
And finally I was able to provide a small Equity Trading Account Service to augment my business offerings!
Subtraction Operations Are all about Elimination of Assets!

But when a recession hit the street, people diversified their investments, and moved their attention to Fixed-Income investment opportunities. My business suffered a great loss! Thus I instructed to retire the Equity Trading Account service.
Software Asset Consolidation is on the Horizon! We Start with Service Decomposition...

Just as every good business, it was the time to reorganize! It seemed logically that the Customer Accounts service should be consolidated with the Banking Account service. So what do we do next?
I ordered to demote the Customer Accounts service before merging it with the Banking Account service.

Use the “Transformed” symbol to signify the transition from a composite to an atomic service.
Continue with Decomposition of Assets!

And I also ordered the demotion of the Banking Account service!

The "Decomposed" symbol indicates service break down.

- C: Banking Account Composite Service
- D: Checking Account Atomic Service
- E: Savings Account Atomic Service
- A-F: Customer Accounts Atomic Service

Service-Oriented Discovery & Analysis Notation:
- Aggregated
- Subtracted
- Intersected
- Overlapped
- Decomposed
- Unified
- Comment
Continue with Transformation of Assets!

The Banking Account service was demoted. At this point, we were ready to consolidate....
Unification of Atomic Structures

We finally consolidated these two service offerings in anticipation to even merge more assets.
Comments and Sequence of Events are always Welcome!
Aggregation of Services Combines Related Software Entities

A-C, A-F atomic service transformed to CF (composite service) because it aggregates now services D and E.
OK, It’s Time to Play Again!

Revealing a Service Ecosystem...

- Understand Distributed Formation
- Understand Interoperability
- Understand Reusability
- Understand Service Relationship
Creating Cluster Formations

Cluster CL-A aggregates services D and G.

Service-Oriented Discovery & Analysis Modeling Dashboard

My World

Asset Portfolio
- CL-A. Small Business Service Cluster
- D. Small Business Loan Service
- G. Small Business Profile Service

My Virtual Modeling World

Software Assets
- Atomic Service
- Composite Service
- Service Cluster

Service-Oriented Discovery & Analysis Notation
- Aggregated
- Subtracted
- Intersected
- Transformed
- Decomposed
- Unified
- Overlapped
- Comment
Aggregating Services by Employing Clusters

Cluster CL-A also aggregates services E and F.
Exploring Service Commonalities

Cluster CL-B aggregates services H, J, and I.
Increasing Software Asset Reusability

Service-Oriented Discovery & Analysis Modeling Dashboard

My World

Asset Portfolio

CL-B: Home Loan Service Cluster
H: Home Loan Service
I: Customer Name and Address Service
J: Customer Profile Service

CL-A: Small Business Service Cluster
D: Small Business Loan Service
E: Risk Assessment Service
F: Credit Verification Service
G: Small Business Profile Service

My Virtual Modeling World

Software Assets

- Atomic Service
- Composite Service
- Service Cluster

Clusters CL-A and CL-B intersect. Services E and F reside in the overlapping clusters’ area

Service-Oriented Discovery & Analysis Notation

- Aggregated
- Subtracted
- Intersected
- Transformed
- Decomposed
- Unified
- Overlapped
- Comment
Another Example: Application Level Service-Oriented Analysis
Imagine a Java Program named BankingAccount.Java that has 16 business activities that provide Banking Account services.
Wouldn’t it be Easier to logically partition these methods into four distinguished groups?
1. Login method group
2. Savings Account method group
3. Checking Account method group
4. Banking Account method group
...And Wouldn't it be practical to regard each of these method groups as a service?
1. Login Service
2. Savings Account Service
3. Checking Account Service
4. Banking Account Service
If this makes sense, we can visualize this formation as a composite service that contains smaller and finer grained services. The most generalized service, the Banking Account service that consists of general banking account activities (methods), is regarded as a composite service that aggregates smaller services (fined-grained), each of which is an atomic service (indivisible entity).
During your service-oriented analysis phase, you may want to decompose your Baking Account Composite Service, reduce its size and increase the reusability rate of the Login Atomic Service. The Login.Java is the program that executes the Login Atomic Service.
While analyzing your services, the Login Atomic Service may be a candidate for aggregation. You may want to include it in the already existing BankingUtility.Java program to join the other utility services that are aggregated into the Banking Utility Composite Service.
Analysis Decisions are Your Company's Intellectual Property. SOMF Can Assist You to Document this Process and Your Train of Thought, and Preserve Analysis Considerations
The Service-Oriented Modeling Framework (SOMF) introduces a formal language that can be used to describe an analysis process. This approach advocates that you preserve and document the train of thought that influenced your analysis decisions. Note the sequence of events in the below service analysis diagram.

Partial Service-Oriented Discovery & Analysis Notation

- Aggregated
- Decomposed

**Analysis Decision:** We better aggregate the Login service into the Banking Utility service, because of their technical commonalities.

**Analysis Decision:** Login does not belong to the Banking Account composite service.

Banking Utility Composite Service

Login Atomic Service

Banking Account Composite Service

Download Reports Atomic Service

Schedule Payments Atomic Service

Checking Account Atomic Service

Savings Account Atomic Service
SOMF Enables Business & Service Life Cycle Traceability
If you are describing a service life cycle you can even add next to each step the affiliated costs and execution dates. This can improve your future business traceability and enhance your future budget projections.
Governance Traceability Perspective

SOA Governance & Architecture Best Practices Traceability
Governance/Best Practices Traceability Perspective

During your service-oriented analysis process identify SOA best practices that are advocated by your SOA government organization.
Another Example: Enterprise Level Service-Oriented Analysis
Enterprise Architecture Best Practice Traceability

Diagram showing various components and service flows in an enterprise architecture model.
END