

July 18, 2006

Proper Scoping of Enterprise Services

The Open Group IT 2006 Architecture Practitioners Conference

André Alguero, *Vice President/Architect*, Digitas, LLC
Dan Hughes, *Principal Consultant*, Systems Flow, Inc.
Graham Williams, *Vice President*, Integration Architecture, Citizens Bank Financial Group

DIGITAS



In order to move up the SOA value chain, the services that make up an enterprise SOA must be scoped to optimal levels. While SOA pundits continue to focus on the broader challenges of SOA adoption, real decisions must be made daily by architects attempting to create useful and effective services. Using real world examples, this presentation seeks to explore the influencing factors and critical considerations of scoping services appropriately to form a value-based enterprise SOA, as well as offer some decision making guidelines, while recognizing there is no one right answer for a given enterprise.

- I. Definitions and Consequences
- II. Service Scoping Framework
- III. Applying the Framework
- IV. Framework Application Guidelines
- V. Skills
- VI. Questions
- VII. Resources

Definitions and Consequences

- **Webster’s**

- 1 : **Intention, Object**

- 2 : space or opportunity for unhampered motion, activity, or thought

- 3 : **extent of treatment, activity, or influence**

- 4 : **range of operation**: as a : the range of a logical operator

- : a string in predicate calculus that is governed by a quantifier

- b : a grammatical constituent that determines the interpretation of a predicate or quantifier

- synonym see

- **Service Definition**

- The set of concerns encompassed by a service and the bounds in which that service operates to achieve a certain goal.



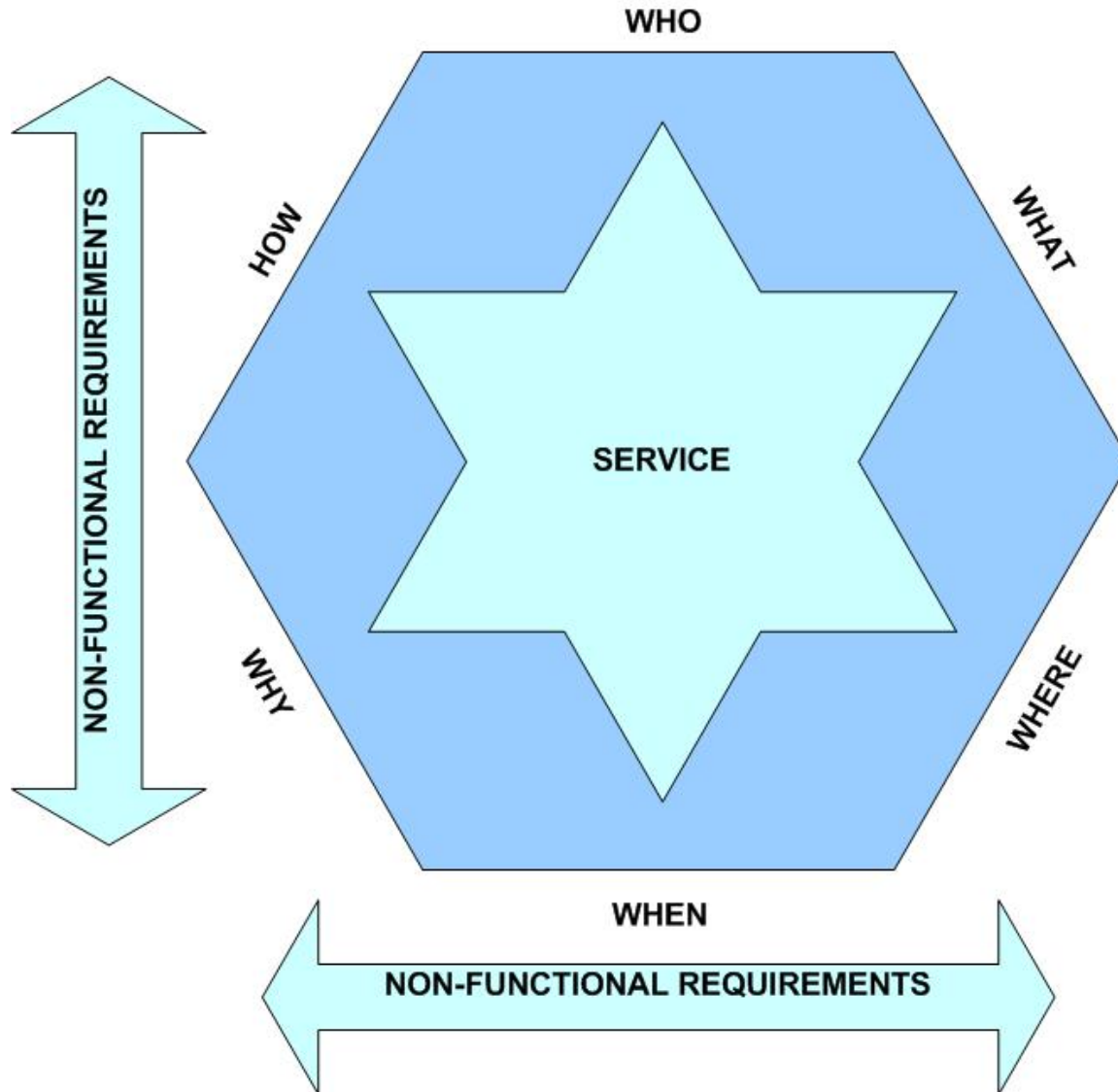
Proper Scoping results in services that are:

- Easily understood and therefore easier to use
- Reusable in isolation and in coordinated business processes
- Not inappropriately sensitive to changes in the enterprise architecture
- Easier to operate, support, and maintain

Bad Scoping results in services that are:

- Brittle and sensitive to internal and external changes
- Seldom reused
- Vulnerable to prolonged maintenance cycles and rework
- Apt to force service consumers to be inappropriately scoped as a result of what the service can or cannot do for them (“Domino Effect”)

Service Scoping Framework



- WHO:**
The people and entities impacted by the service
- WHAT:**
The data provided to, acted on, and produced by a service
- WHERE:**
The locations related to the service
- WHEN:**
The impact of time on the service
- WHY:**
The motivations of the service
- HOW:**
The process and algorithms that drive the service

- Who are the stakeholders impacted by the service?
 - Users (e.g. Customers, Sales Representatives, Tellers)
 - Beneficiaries (Customers, Employees)
 - Indirect Users (e.g. Auditors, Managers)
 - Operational (Call Center, Developers, Network Engineering)

- What is the impact of each stakeholder on the scope of the service?
 - Will the stakeholders' needs be addressed by a single or multiple services?
 - What does the service need to know about persons identity?
 - What authority should each stakeholder be granted?
 - Who is responsible for the business rules that the service must implement or obey?

- Where will the service be used from?
- Where will the service be operated?
- Where are the components on which the service is dependent? (data repositories, sub-services, rules engines, etc.)
- What is the impact of the processing location vs. the location of the consumer or user?

- What is the impact of location on service scope?
 - Regional business rules and regulations
 - May drive need for insulation from changes in location
 - Network Bandwidth
 - Internationalization

- When will the service be used?
- How frequently will the service be used?

- What is the impact of time on the service scope?
 - Execution time restrictions
 - ❖ Business process restrictions
 - ❖ Operational dependency driven restrictions
 - Response Time
 - Time of Request vs. Time of Execution (e.g. real time, queued, batch)
 - Data Freshness

- What are the business goals that need to be met?
- What rules support the business goals?
- Why are services being considered to address the business process?

- What is impact of the business motivation on the scope of a service?
 - What rules are in the scope of the service vs. surrounding business activities?
 - What rules should be shared between components?
 - What rules should be encapsulated by the service?
 - Is the service replacing, replicating, or augmenting existing rules?

- **What data and entities are required or acted upon by the service?**
- **What data is produced by the service? (e.g. transactional data, calculations, logging, operational attributes)**
- **What is the classification of the data?**
 - Enterprise vs. Localized
 - Secure vs. Non-Secure

- **What is the impact of data on the service scope?**
 - What data does the consumer require from the service?
 - What data does the enterprise need from the service?
 - What knowledge does the service have about the data?
 - What is "provided to" vs. "found by" the service?
 - ❖ Provided by Consumer - good if consumer has the data as a natural consequence, bad if consumer has inappropriate knowledge to get it
 - ❖ Found by the Service - saves consumer work but can couple the service to the data provider

- How does the service align with the overall business processes and the individual actions taken within that process?
- How should the service align with existing services?
- How should the service perform its function?

- What is the impact of function on the service scope?
 - Drives primary service goal
 - Illuminates the potential for reuse
 - Services must account for multiple business line specific conditions
 - ❖ Business process context
 - ❖ Application integration

Adding complexity to the analysis, dimensions can intersect. Some examples include:

- Regional Rules (*Where*) may be set by a central authority or a decentralized authority (*Who*)
- The data acted upon (*What*) may need a particular “freshness” date (*When*)
- The people (*Who*) using or benefiting from the service directly influences the motivations of the service (*Why*)
- What level of security (*Why*) is required for users (*Who*) to manipulate the data being acted upon (*What*) by the service?

- Query Services - Information Providers
- Execution Services - Perform Updates
- Infrastructure Services - Provide foundational services not specific to business needs - i.e. Logging, Alerting
- Business Logic and Coordination Services
 - Services that encapsulate logic and lookups such that one or more consumers can share logic with one or more services
 - If available consumers and services could invoke a business rules engine, but even those might be exposed as services

Applying the Framework

1. Analyze service factors in isolation to a Line of Business
2. Identify proposed services rationalized to the catalog of existing enterprise services
3. Refine service factors to Enterprise level
4. Determine in-scope factors, consulting catalog of logged design decisions for similar patterns
5. Refine proposed services, adding any new services to catalog classified as line of business or enterprise level
6. Record decisions and long-term goals

- The Business Process - For organizations that have a need to send correspondence to its Customers, “Change Address” is a critical business process and/or application function that is necessary to service Customers.
- System Complexity - In this particular example, a Financial Institution stores many types of addresses (i.e. primary, secondary, seasonal) for Customers and Accounts. The addresses are stored in many disparate legacy systems, and the address attributes (i.e. types, # of address lines, country codes, no city fields) are inconsistent between the systems.
- Motivation for Technology Solution – Due to the complexity of the system and process necessary to change addresses, the Branch Platform business line explored the opportunity to implement a technology solution to simplify the Change Address process for Bankers.

	Line of Business View	Enterprise View
WHO	<ul style="list-style-type: none"> - Back Office Users (for exception processing) - Branch Bankers 	<ul style="list-style-type: none"> - Call Center Service Agents - Online Banking Customers
WHAT	<ul style="list-style-type: none"> - Retail / Small Business Customer Address - Deposit Account Address 	<ul style="list-style-type: none"> - Commercial Customer Address - Credit Card Account Address - Consumer Loan Account Address
WHY	<ul style="list-style-type: none"> - “Once and Done” (No follow-up actions from Banker) - Simplify system interactions for Banker 	<ul style="list-style-type: none"> - Consistent “Change Address” experience for Customer regardless of delivery channel
WHERE	<ul style="list-style-type: none"> - Primarily Domestic U.S Addresses - Branch Network, Regional Back Offices 	<ul style="list-style-type: none"> - Multiple Call Center Locations - Online Banking (any location) - Consideration of owning company (U.K.)
WHEN	<ul style="list-style-type: none"> - Interpreted “Once and Done” as “The function needs to be completed now and in real-time for the Branch Banker” - During Branch Hours 	<ul style="list-style-type: none"> - Time Zones of Call Center Locations - Online Banking (24 x 7) - The completion of address changes is typically not sensitive to very precise timing (opportunity for batch processing).
HOW	<ul style="list-style-type: none"> - Services focused on updating addresses on the systems of record - In this example, constrained by how the existing Branch application suite already functioned 	<ul style="list-style-type: none"> - Service to get / view all customer addresses in the enterprise - Service to accept change requests in real-time (edits, consistency checks, etc.) - Service to handle changes in batch - Service to handle exceptions in batch

- Considering the Line of Business Only resulted in:
 - Services that work for the Branch and the applications used by Branch Bankers
 - Services that work for specific Account types (i.e. Checking, Savings)
 - Change Address Services that are coupled with the system(s) of record (due to the misinterpretation of “once and done”)
 - Services that are light in business rule implementation and provide little reuse value
 - Application logic that is “locked up” inside the Branch application suite

- Enhancing the scoping exercise to consider the Enterprise would have increased the likelihood of:
 - Services that implement the business rules necessary for multiple delivery channels
 - Services that handle address for all Account types
 - Services that support a consistent experience throughout the enterprise
 - ❖ Services that work in real-time to accept the Change Address request
 - ❖ Services that work in batch to deal with system complexities and to handle exception conditions

Framework Application Guidelines

Today's decision is not necessarily tomorrow's, plan for refactoring:

- It is critical to assess the potential need for change
- Record design decisions and follow up actions with assessment deadlines
- Plan for the anticipated changes and keep budget set aside for the unanticipated

- The relationship between service and service consumer should be loosely coupled through abstraction and proper scoping.
Loose coupling is difficult to achieve in an unstable environments. Focus on stable aspects of the enterprise first
- A service should focus on highly related activities.
Elements of a service should "all work together to provide some well-bounded behavior" (Booch on OOP)
- Error handling can introduce scope creep or force scope externalization
Scope Creep: Handling an error that requires additional knowledge about external business processes
Scope Externalization: Forcing consumers to handle errors that are meaningless to them (E.g. Mainframe "file closed" error)
- Query and update functions should be separated into distinct services.
Both consumers and update services can use the query services, and query services tend to have different performance characteristics
- Update services should achieve a business scoped goal - i.e. "Cash Check" vs. "Post Debit to GL". Determining this scope is a similar activity to correctly scoping Use Cases.
- Model Consumer-Service interactions as a method for fleshing out correct service scoping.
This will sometimes force refactoring of a service based on the business scoped goal.
- Scope services based on functional and not operational attributes
Don't have a "call center" vs. "Branch" "Change Address" service, but determine the functional differences between the two and scope with those.
- A business scoped service can have optional activities. "Send Email" can include or not include an attachment. This doesn't require a "With attachment" service and a "Without" service because the optional activity does not change the business goal of the service.
- A service must be defensive. It should reconfirm business rules even if the UI has already checked them.

Skills

- Enterprise Application Integration
- Object Oriented Design and Patterns
- Experience with Refactoring
- Business and Systems Analysis
- Mentors to transfer experiential knowledge

Questions?

- **André Alguero** (aalquero@digitas.com) is a software architect with 15 years of experience building software solutions and consulting with enterprises to build and execute technology strategies. André has a passion for software engineering best practices with a focus on using models to build robust software solutions. André is a VP/Architect at Digitas, Inc. where he designs enterprise marketing solutions and technology strategies for Fortune 500 companies with a focus on content management. He previously was a Principal Consultant at Systems Flow, Inc.
- **Dan Hughes** (dh@sysflow.com) is a principal consultant with Systems Flow, Inc., www.sysflow.com. Systems Flow helps organizations dramatically improve their competitive advantage through the practical, effective application of best practices in enterprise architecture and software development. He is engagement lead at Citizens Bank where he guided the launch of the enterprise architecture practice and is currently enterprise architect for Citizens Bank's Basel II implementation. Dan has 15 years of software engineering experience spanning a broad range of technologies and techniques. Startup to enterprise, he has launched, managed, and executed all aspects of both product and enterprise life cycle for clients in industries ranging from industrial automation to banking and insurance. He maintains a blog on software engineering at www.xengineering.com.
- **Graham Williams** (graham.williams@citizensbank.com) is a solution architect with 10 years of experience designing, developing, and supporting enterprise technology solutions for the financial services industry. Graham currently works for Citizens Bank, where he leads a team responsible for the integration & services architecture used by multiple customer delivery channels, including the Call Center, Online Banking, and the Branch. He has a B.S. in Chemical Engineering from Oklahoma State University and an Masters in Business Administration from the University of Oklahoma.

DIGITAS



- "Coarse-grained interfaces enable service composition in SOA". Jeff Hanson. (<http://builder.com.com/5100-6386-5064520.html>).
- " SOA service design: Developing for the future". David S. Linthicum. (http://www.infoworld.com/article/06/05/08/77670_19FEsoalife4_1.html)
- ""My" SOA definitions--What I mean when I say "service-"" (http://elementallinks.typepad.com/bmichelson/2005/07/my_soa_definiti.html) .brenda michelson.
- "Applying UML and Patterns", 2nd Edition. Craig Larman 2002. Prentice Hall PTR
- "Enterprise Integration Patterns", Hohpe,Woolf 2004. Addison Wesley
- "Increase flexibility with the Service Integration Maturity Model (SIMM)". Ali Arsanjani, Ph.D. IBM Web Site (<http://www-128.ibm.com/developerworks/webservices/library/ws-soa-simm/>)
- "The Zachman Framework", <http://www.zifa.com/framework.html>

	Line of Business View	Enterprise View
WHO		
WHAT		
WHY		
WHERE		
WHEN		
HOW		

