Getting started with IMS and SOA?

TAKE BACK CONTROL

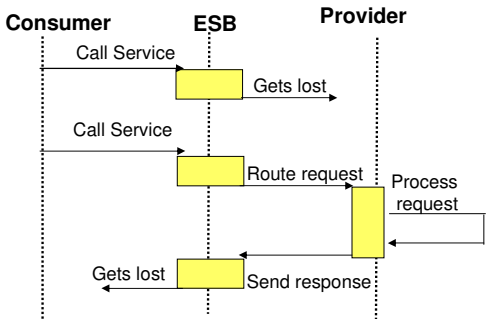
Today

- We started with an overview of SOA
- We talked about ways to service enable your IMS Assets
- We reviewed concepts, tools and technologies
- What next? How do you get started?

Step 1: Education – Understand SOA

- A necessary first step
- Lack of skills is #1 inhibitor to SOA implementations
- SOA seems simple but its not a concrete architecture
- Lot of things to think about and take decisions on

Is your Service Idempotent?



- The ability to redo an operation if you are not sure whether it was completed
- What if the above service deposits money into a customer's account?
- Reading services are idempotent. How do you make writing services idempotent?

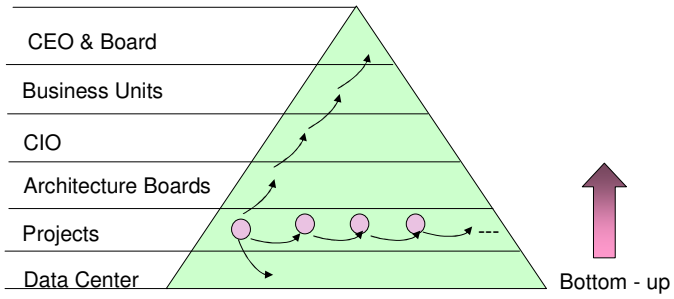
Is it a Stateless or Stateful service?

- Stateless – does not maintain any state between service calls
- Stateless – all local variables & objects thrown away after call
- Load balancing and failover for the stateless service is simple
- ESB simply passes the request to the next best available service thread
- Stateful – State is maintained across multiple service calls
- Stateful service resources are bound to one specific consumer session until that consumer session ends

Message Exchange Patterns (MEPs)

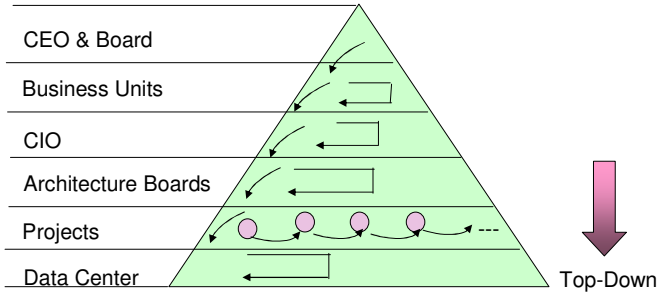
- Sending of messages between a consumer & a provider
- Request/Response
- One-Way (send of requests)
- Request/Callback
- Publish/Subscribe (Event driven Architecture – SOA 2.0)

Bottom-Up (from existing transactions) Implementation



- Services are defined by understanding the existing application transactions and determining how they can be combined into a business process.

Top-Down (from business Processes) Implementation



A top-down approach defines the necessary business services and then evaluates existing transactions as a source of satisfying this need.

SOA & Security

- Services in a Business Process may come from different systems
- Applications are loosely coupled and offered up as services through the registry
- ESB
- Security is maintained thru:
  - Identity Management
  - Software and data authentication
  - Audit Trails

Step 1: Education – Understand SOA

- Websites:
  - http://www.oasis-open.org/home/index.php for Standards
  - http://www.soamodeling.org/ SOA project in action
  - http://www.looselycoupled.com/ articles, newsfeeds on SOA
  - and many more
- Books at your local bookstore
- Vendor Seminars
- IBM will gladly come on site and assist

Step 2: The SOA Pilot

- Try out SOA
- Start with your existing technologies - these are your core business services.
- Start small (Don't try to boil the ocean)
- Pick a business process that spans two systems at most
- Use only basic services – leave service composition for later
- Use a simple message exchange pattern (Request/Response)
- Decide about the amount of loose coupling
- Test out your code to request the basic service.

Step 3: The Second and Third projects

- For the second project you should use bottom-up Implementation
- Considering reusability of services, have different processes using the same service
- Have different teams for the different processes and make sure that their process is able to call your service
- Change your service to a stateful service and now the other process might fail (depending on the code change in your service)
- For the third project try top-down Implementation

Final Thoughts

- With SOA you are looking to build a reuse culture
- Everybody gets involved in SOA – Need to work across organizations
- SOA is a journey, not a one-time project that a single department implements to get a quick success
- It's a corporate - wide process to leverage technology in a way that reflects the business's key business processes, enabling business to change when needed without being constrained by IT.
- IBM is here to support you every step of the way

Soa Long