


Service-Oriented Architecture Trends

Jason Bloomberg
Senior Analyst
ZapThink, LLC

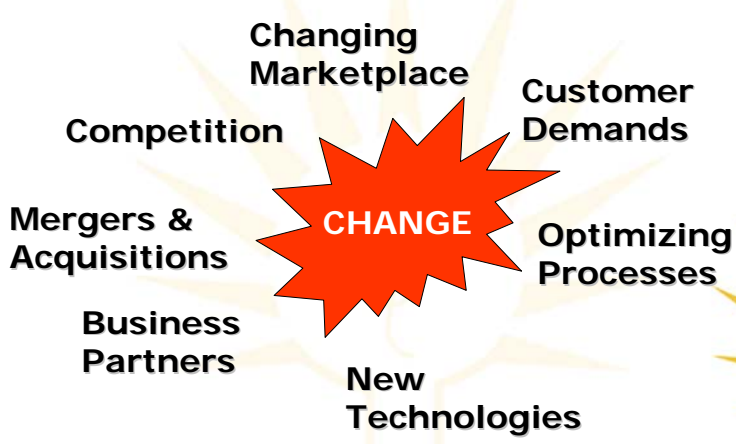
Take Credit Code: SOATIBM

Copyright © 2006, ZapThink, LLC

zapthink



Business Constant: Change



CHANGE

Changing Marketplace

Customer Demands

Optimizing Processes

New Technologies

Business Partners

Mergers & Acquisitions

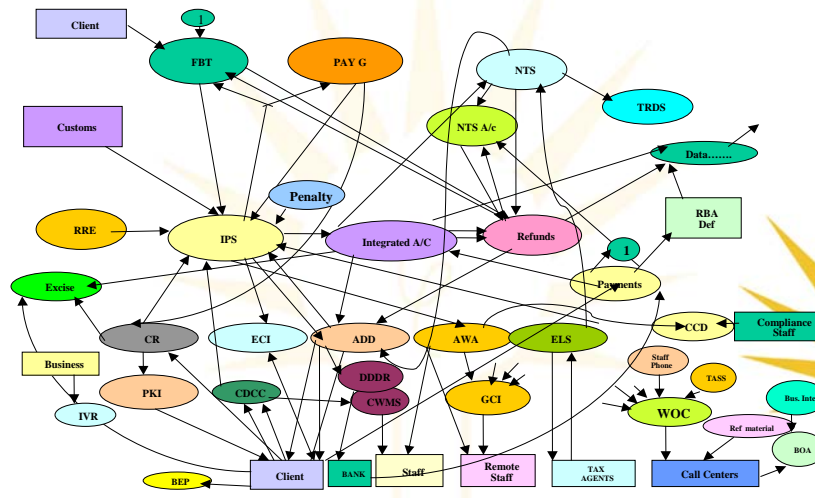
Competition

A Business is Never STATIC

Copyright © 2006, ZapThink, LLC



The Challenge of Complexity...

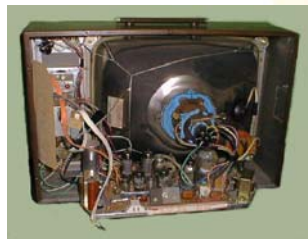


Copyright © 2006, ZapThink, LLC



Yet the Best Technology...

- Is complex on the inside yet simple on the outside



- The secret is the *abstraction layer*

Copyright © 2006, ZapThink, LLC



zapthink

Level Set – What is SOA?

- SOA is *architecture* – a set of best practices for the organization and use of IT
- Abstracts software functionality as loosely-coupled, business-oriented *Services*
- Services can be composed into *business processes* (which are also *Services*) in a declarative manner



Copyright © 2006, ZapThink, LLC



zapthink

The Benefits of SOA...

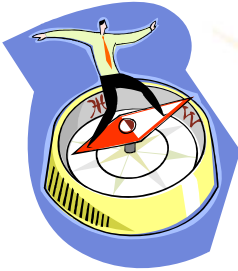
- ✓ Reduced cost of integration
- ✓ Improved value from legacy applications
- ✓ Reuse leading to reduced redundancy
- ✓ Greater visibility for governance & compliance
- ✓ Increased reuse of software assets
- ✓ Business agility...

Copyright © 2006, ZapThink, LLC



Business Agility

- Companies require *Business Agility*...



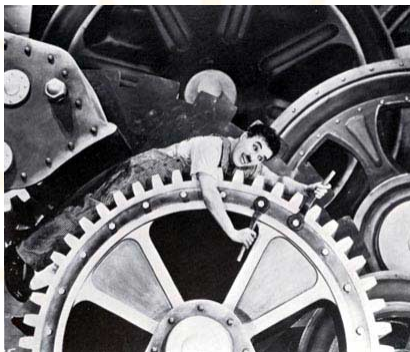
- » Responding quickly to change,
and
- » Leveraging change for
competitive advantage

Agility is the key to innovation

Copyright © 2006, ZapThink, LLC



Business Process the Old Way...



- People plugged into rigid processes
- Inflexible & brittle

Copyright © 2006, ZapThink, LLC



zapthink

Business Process the Service-Oriented Way...



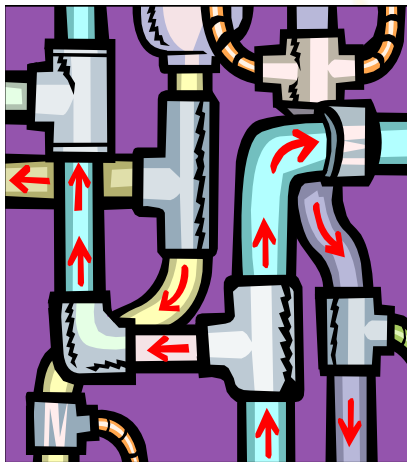
- IT resources (among other resources) available to the business as needed
- Business users create composite applications by composing Services & configuring processes dynamically

Copyright © 2006, ZapThink, LLC



zapthink

SOA Abstracts the Plumbing



- The goal is reusable, composable business Services
- Many different approaches to implementation

Copyright © 2006, ZapThink, LLC



zapthink

SOA shifts the way we think

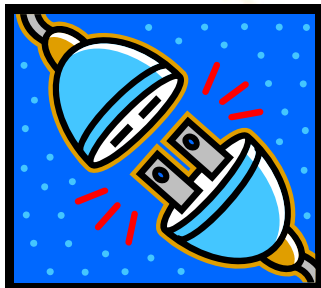
Traditional Distributed Computing	Service Oriented Architecture
Designed to last	Designed to change
Tightly Coupled	Loosely Coupled, Agile and Adaptive
Integrate Silos	Compose Services
Code Oriented	Metadata Oriented
Long development cycle	Interactive and iterative development
Middleware makes it work	Architecture makes it work
Favors Homogeneous Technology	Favors Heterogeneous Technology

Copyright © 2006, ZapThink, LLC



zapthink

The Secret Sauce: Loose Coupling



- Power of Abstraction
- Power of Open Standards
- The Challenge of Loose Coupling
- Loose Coupling enables *Change*

Copyright © 2006, ZapThink, LLC



Characteristics of SOA Implementations

- Services are *loosely coupled* – making a change to a Service provider does not mandate changing any Service consumers.
- Business processes are composed of Services, and are in turn exposed as Services.
- Services are *policy-driven* – business users can change how a Service behaves.
- Systems are *inherently integrated* by virtue of composable Services – not through layers of middleware.

Copyright © 2006, ZapThink, LLC



Characteristics of SOA Implementations (Cont.)

- Services *leverage* legacy systems – SOA does not mandate replacement of runtime infrastructure.
- In SOA, *metadata* control how the system behaves instead of code – business logic trumps application logic.
- In SOA, it's the *contracted interface* that matters, more so than the underlying runtime environment.

Copyright © 2006, ZapThink, LLC



zapthink

How to Think Service-Oriented



- Service Orientation is about **change**
- IT must respond to change and enable innovation

Copyright © 2006, ZapThink, LLC



zapthink

Building Agility with Agility



Traditional Distributed Software Development

- “Waterfall” – Gather requirements, design, develop, test, deploy as separate steps
- Works great when things don’t change
- Typically fails!

SOA – applications are never complete, Services are always in flux

- *Traditional SDLC wholly inadequate*

- Reuse: The Holy Grail of IT

Copyright © 2006, ZapThink, LLC



Telco Example

Verizon

Large regional telecommunications provider formed thru the merger of Bell Atlantic and GTE



Challenge

- Eliminated redundant systems inherited from the merger of Bell Atlantic and GTE
- On average, each transaction had been developed five to 25 times; one was deployed 45 different times

Solution

- IT Workbench SOA project, operational in 2004
- Thousands of developers, .NET and Java
- Focused on 250 business transactions – incl. verifying customer credit histories & looking up customer info
- 57 Service-oriented applications with 200 transactions

Results/Benefits

- 2.5 million to 3 million Web Services transactions a day
- Helped Verizon slash its IT budget by 50 percent
- Included managing and securing the Services, charging for reuse and monitoring the performance of Service-enabled transactions

Copyright © 2006, ZapThink, LLC



Building SOA the Right Way: Take an Iterative Approach

- Top-down only: have the plan, may not be able to execute
- Bottom-up only: build Services, may not be reusable
- SOA planning *must* be both
 - Develop the vision (but not the details) ahead of time
 - Service development should be iterative



Copyright © 2006, ZapThink, LLC



Government Example

The Defense Finance & Accounting Service (DFAS)

US Federal Government agency responsible for all Department of Defense accounting

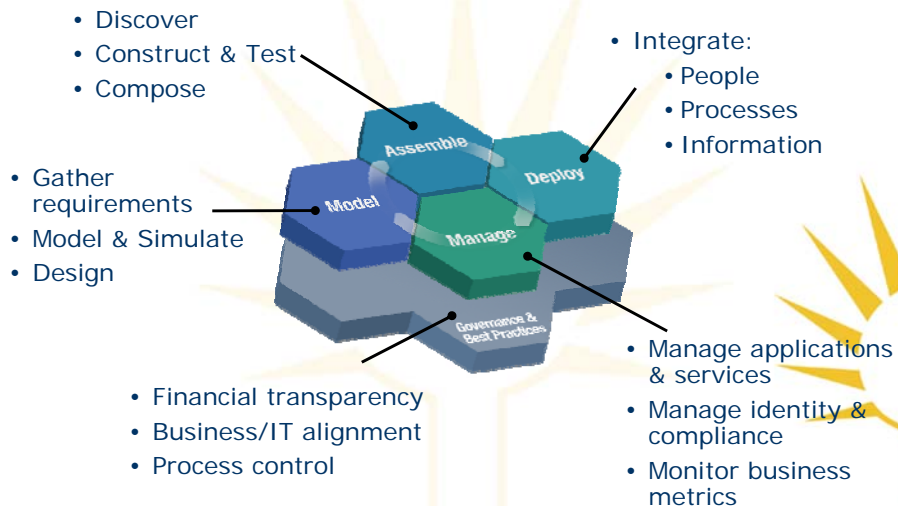


Challenge	Solution	Results/Benefits
<ul style="list-style-type: none"> • Very large, complex organization that handles accounting for Army, Navy, Air Force & Marines • Process \$1 Billion dollars of payments per day • Challenge of how to bring together DoD finances • "Analysis Paralysis" – too much time devoted to design 	<ul style="list-style-type: none"> • Reduced hardware costs by streamlining operations • Achieved "semantic alignment" across organizations • Collated and rolled up information in the face of enormous complexity • Focused on information architecture in the context of SOA 	<ul style="list-style-type: none"> • SOA effective in environments of extreme complexity • Information architecture essential to resolve semantic issues in complex environments • SOA appropriate in environments where there are many stakeholders with many needs

Copyright © 2006, ZapThink, LLC



The SOA Lifecycle: Beyond the SDLC



Copyright © 2006, ZapThink, LLC

Source: IBM



zapthink

The Challenges with doing SOA right

- **Architectural issues**
 - Enterprise Architecture capability maturity
 - Building SOA the *right* way
 - Enabling effective governance
- **Tool & infrastructure requirements**
 - XML performance issues
 - Security & management challenges
 - Contract & policy development and implementation
 - Composite application development
 - Metadata management
- **Organizational issues**
 - Pulling together the SOA team
 - Who's in charge of Services?
 - How will you pay for Services?
 - How will SOA affect your organization moving forward?

Copyright © 2006, ZapThink, LLC



zapthink

SOA is Architecture!

- Remember...SOA is architecture – in particular, *Enterprise Architecture*, including:
 - An aggregated architecture of all the individual IT systems within an organization
 - The human element within the enterprise
 - Systems, people, and organizational constructs at other companies that have relationships with the enterprise
 - Individual consumers who are that enterprise's customers
 - Corporate governance



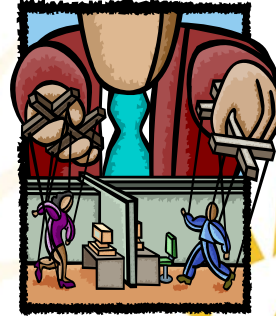
Copyright © 2006, ZapThink, LLC



zapthink

The Role of Governance...

- Establishing and communicating the policies that employees must follow
- Giving employees the tools they need to be compliant with those policies
- Providing visibility into the levels of compliance in the organization
- Mitigating any deviations from established policy



Copyright © 2006, ZapThink, LLC



zapthink

SOA Governance

- Policy management
 - SOA configured & controlled via metadata, including policy
- Visibility
 - Services abstract heterogeneous data sources, providing necessary business intelligence
- Flexibility
 - Ability to build Services that address compliance issues and adjust them as regulations or business needs change

***Not just governance of SOA...
governance in the context of SOA***

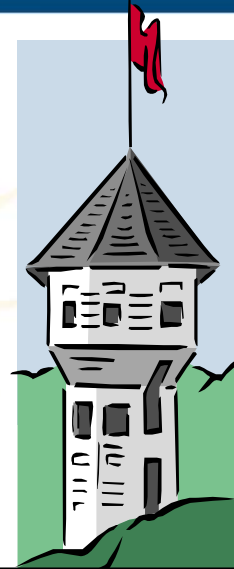
Copyright © 2006, ZapThink, LLC



zapthink

Organizational Issues -- The "Ivory Tower" Problem

- Architects create design and other artifacts, but don't have the authority or mandate to require their use
- Development team considers them optional
- Business likes idea of architecture in principle, but short-term needs trump best practices
- When architects are external consultants, the "not invented here" syndrome makes the Ivory Tower worse



Copyright © 2006, ZapThink, LLC



zapthink

Energy Company Example

Large Global Petroleum Firm

Challenge	Solution	Results/Benefits
<ul style="list-style-type: none">• Different divisions had different concepts of SOA• No unified approach to Enterprise Architecture (EA)• Lines of business often had very little in common	<ul style="list-style-type: none">• Brought together EAs from across company for first time in 6 years• Worked to develop a common vocabulary• Identified areas of redundancy suitable for shared Services• Developed a SOA maturity model	<ul style="list-style-type: none">• Moved toward establishing enterprise-wide EA committee• Undertaking SOA pilot projects

Copyright © 2006, ZapThink, LLC



One Approach: Service Domains

- A *Service Domain* is a logical grouping of shared Services with a common *business context*
- Manage Services by managing the Domains
- Move away from traditional IT silos

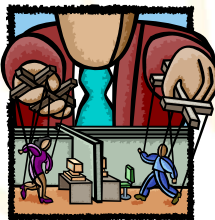


Copyright © 2006, ZapThink, LLC

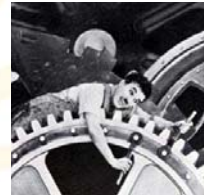


Resolving SOA Challenges...

- People...



- Process...



- Information...



Copyright © 2006, ZapThink, LLC

Thank You!



zapthink

ZapThink is an industry analysis firm focused exclusively on XML, Web Services, and Service-Oriented Architectures.

Read our new book,
Service Orient or Be Doomed! How Service Orientation Will Change Your Business.



Jason Bloomberg
jbloomberg@zapthink.com

Copyright © 2006, ZapThink, LLC

zapthink