

# Fundamental SOA Concepts

Jason Bloomberg and Ronald Schmelzer  
Senior Analysts  
ZapThink, LLC

Take Credit Code: EUFUND

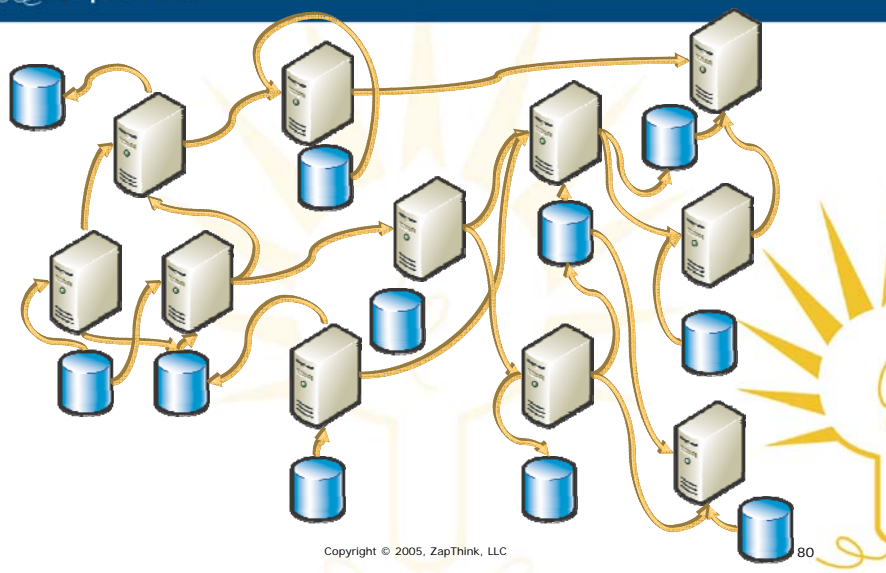
Copyright © 2005, ZapThink, LLC

zapthink



We've had IT challenges for years ...

zapthink

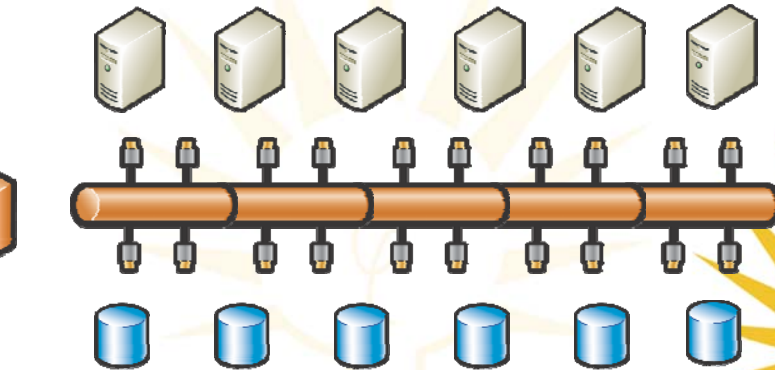


Copyright © 2005, ZapThink, LLC

80



... but even after yesterday's promises...

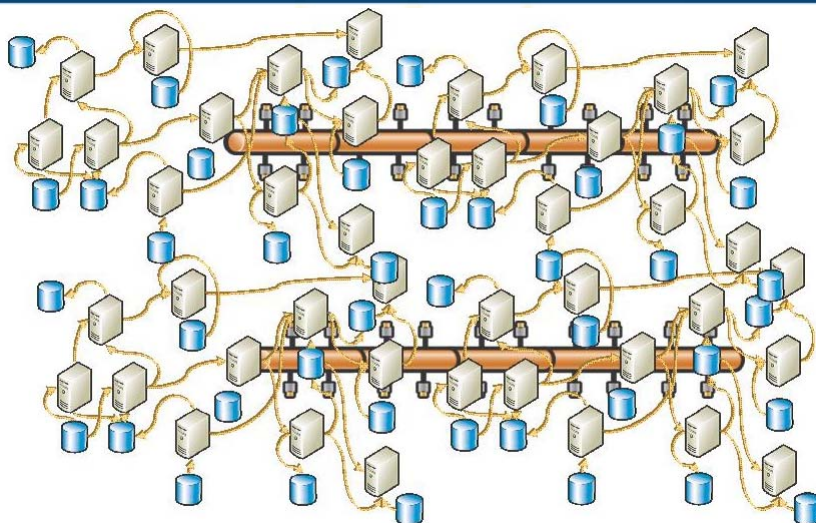


Copyright © 2005, ZapThink, LLC

3 / 80



... we still have the same IT mess, only worse.



Copyright © 2005, ZapThink, LLC

4 / 80



zapthink

## There's Got to Be a Better Way!

***Rather than trying to simply throw more software and iron at the problem, we need a better way of organizing our IT resources***

- Service-Oriented Architecture (SOA) represents a fundamental evolution in the IT industry
  - The core business motivation is *business agility*.
  - Rather than “rip and replace” old systems – *make them work better together*
  - We don't want more middleware for our middleware
  - As fundamental a change as mainframe to client/server or client/server to the Internet

Copyright © 2005, ZapThink, LLC

5 / 80



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 © 2005, ZapThink, LLC

6 / 80



zapthink

## Service-Oriented Architecture

- The Definition:
  - An approach to building and managing distributed computing infrastructures that considers **IT resources as Services available and discoverable on a network.**
- The Implication:
  - Rather than dealing with isolated systems that must be *integrated after the fact*, Service Orientation provides business users with understandable Services they can call upon and compose into business processes as needed – building **systems that can adapt as the business changes.**
- The Benefit:
  - The Service Orientation vision is therefore one of providing the **business values of agility and flexibility** for users of technology, coupled with an *abstraction layer* that simplifies the complexity of today's heterogeneous IT environments from those users.

Copyright © 2005, ZapThink, LLC

7 / 80



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	Process Oriented
Long development cycle	Interactive and iterative development
Cost centered	Business centered
Middleware makes it work	Architecture makes it work
Favors Homogeneous Technology	Favors Heterogeneous Technology

Copyright © 2005, ZapThink, LLC

8 / 80



## SOA Hotspots

- Four key areas of SOA investment
  - Reduction in integration expense
  - Increase in Service / asset reuse
  - Increase in business agility
  - Enablement of governance & compliance
- Key Problem areas
  - EAI replacement
  - Legacy enablement/migration
  - Shared Service development
  - Governance
  - Embedding processes in the extended enterprise



Copyright © 2005, ZapThink



## Don't Get Lost in the Terminology

*Abstraction?* **ESB?** **SOA Infrastructure?** **EAI?**  
**Service Network?** **EDA?** *Service Grid?* *SOA?*  
**Fabric?**

- There are many styles for SOA implementation
- Focus on your goals: Reuse? Governance? Reduced integration cost? Agility?

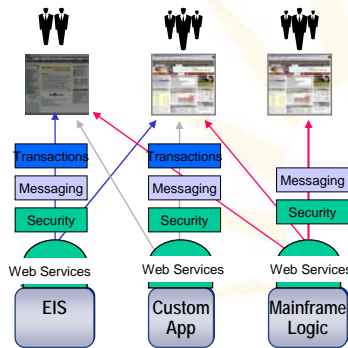
Copyright © 2005, ZapThink, LLC

10 / 80

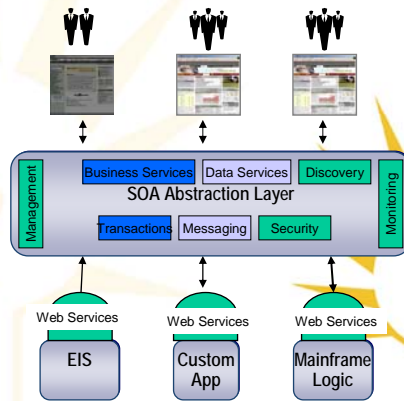


# Web Services make implementing SOA easier, but they aren't the same

Standard architecture with Web Services



SOA leveraging Web Services



Copyright © 2005, ZapThink, LLC

Source: HP  
11 / 80



# Web Services are the Trees...



## Service Orientation is the Forest

Copyright © 2005, ZapThink, LLC

12 / 80





## Service orientation...the next big thing?

zapthink

Approach	Timeframe	Programming Model	Business Motivations
Mainframe timesharing	1960s -1980s	Procedural (COBOL)	Automated business
Client/server	1980s-1990s	Database (SQL) and fat client (PowerBuilder, Visual Basic)	Computing power on the desktop
n-Tier/Web	1990s-2000s	Object-oriented (Java, COM)	Internet/eBusiness
Service orientation	2000s	Service-oriented (SOAP, WSDL, UDDI)	Business agility

Copyright © 2005, ZapThink, LLC

13 / 80



## SOA is a discipline...

zapthink

- Implementing a Service-Oriented Architecture is a *journey*
- Moving from proprietary interfaces to standards-based ones is just the first step...
- SOAs require a combination of security, management, governance, integration, process, and architecture tools
- What are the right steps to guarantee overall success?
- How can you guarantee an ROI while reducing risk?

Copyright © 2005, ZapThink, LLC

14 / 80



zapthink

## What is Architecture?

The fundamental organization of a system embodied by its components, their relationships to each other and to the environment and the principles guiding its design and evolution. (IEEE P1471/D5.3)

Copyright © 2005, ZapThink, LLC

15 / 80



zapthink

## In Particular, SOA is Enterprise Architecture

Enterprise architecture includes:

- 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 © 2005, ZapThink, LLC

16 / 80





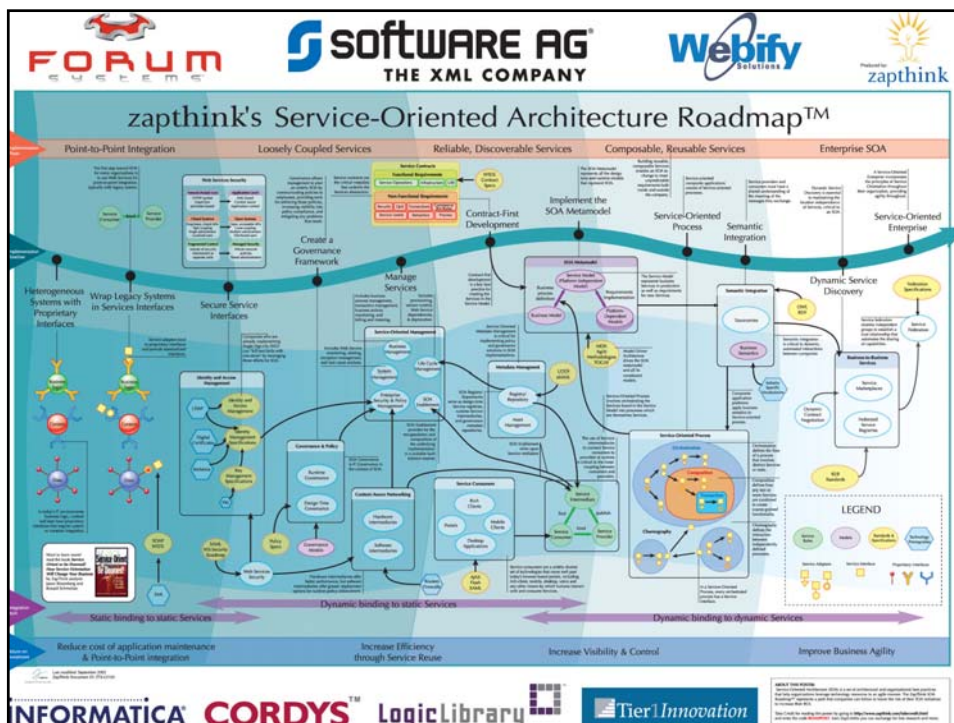
zapthink

# The Challenges with doing SOA right

- Architecture? Sounds difficult!
- Performance issues around XML
- New tool and infrastructure requirements
  - Messaging
  - Contract development
  - Composite application development
  - Metadata management
  - Policy infrastructure
- But... doesn't this infrastructure already exist?

Copyright © 2005, ZapThink, LLC

17 / 80





## ZapThink's SOA Implementation Roadmap



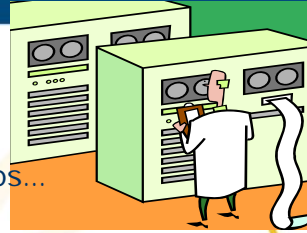
## Grass Roots Web Services

- Web Services are easy to build
- Lower the cost of integration
- Many companies have Web Services springing up in various departments
- Lack management and governance



## Wrap Legacy with Web Services

- What is "legacy"?
  - Host-based systems...
  - SCM, CRM, and other business apps...
  - Anything that's in use...
- Legacy systems enable a significant amount of mission-critical functionality
- Rip-and-Replace vs. Maintain-and-Extend



**The first step to extending functionality: abstracting the implementation – aka "Service Wrappers"**

Copyright © 2005, ZapThink, LLC

21 / 80



## SOA Pilots

- A few high ROI Services
- Build acceptance for SOA
- Get team up to speed
- Work out the kinks
- Pilot the governance model
- Part of an iterative approach to SOA



**DANGER! Avoid the SOA Pilot Pitfall**

- Piloting only the *Services* instead of the *architecture*
- Remember, the pilot is one step on the roadmap

Copyright © 2005, ZapThink, LLC

22 / 80



zapthink

## Secure Service Interfaces

- XML is text-based, human readable – give hackers the keys *and* the instructions
- Existing security inadequate to address content security issues
- Authorization, threat prevention, confidentiality key issues
- What security issues are pertinent to you?

Copyright © 2005, ZapThink, LLC

23 / 80



zapthink

## Build a Governance Framework

- How will you communicate corporate policies to your team?
- How will you enforce those policies?
- What is the role of SOA in your governance plan?

Copyright © 2005, ZapThink, LLC

24 / 80



## Manage Services

- Loose coupling requires Services to behave as expected
- Must handle infrastructure management issues “behind the scenes”
- If a business user knows how Services work, then something is wrong

Copyright © 2005, ZapThink, LLC

25 / 80



## Mission-Critical SOA

- Services incorporated into core business processes
- Issues of security, management, granularity resolved
- Provides ongoing agility – built to change

Copyright © 2005, ZapThink, LLC

26 / 80



zapthink

## Contract-First Development



- Service contracts specify required functionality to IT and provide functionality to the business
- Service contract thus acts as a requirements document for all Service implementation activities

Copyright © 2005, ZapThink, LLC

27 / 80



zapthink

## Build the SOA Metamodel

- Architecture is design – requires a formal approach.
- A metamodel is a model of models.
- Models provide ongoing architectural guidance
- The core model of the SOA Metamodel is the *Service Model*, which represents Services in production *and* ongoing requirements for Services

Copyright © 2005, ZapThink, LLC

28 / 80

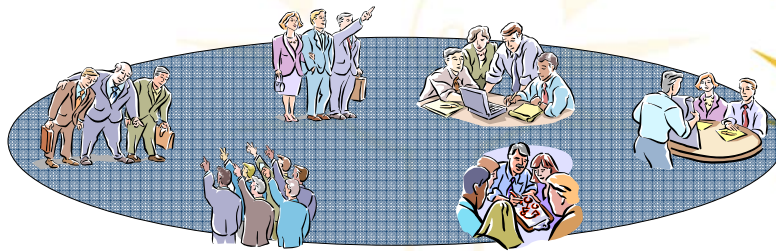




zapthink

## Cross-Departmental SOA

- Organizational issues of governance and control become paramount
- Long-term architectural plan critical
- Increased focus on semantic issues



Copyright © 2005, ZapThink, LLC

29 / 80



zapthink

## Service-Oriented Process

- Compose Services into process that are themselves Services
- Put responsibility for such agile composite applications into the hands of the business
- Business logic increasingly represented by configuration of composite applications

Copyright © 2005, ZapThink, LLC

30 / 80



## Semantic Integration

- Balance loose coupling with semantic consistency
- Leverage industry standard vocabularies
- Tools still immature – still requires substantial manual work



Copyright © 2005, ZapThink, LLC

31 / 80



## Enterprise SOA Buildout

- The corporate enterprise architecture becomes Service-Oriented
- Service lifecycle becomes dominant framework for IT change
- Service abstraction frees company to retire legacy as appropriate

Copyright © 2005, ZapThink, LLC

32 / 80



## Dynamic Discovery

- Service consumers identify and select appropriate Services dynamically at runtime
- Most applicable in business-to-business scenarios
- Advanced capability – many years away from a reality

Copyright © 2005, ZapThink, LLC

33 / 80



## The Service-Oriented Enterprise

- IT resources are available on demand to businesses as Services
- The Service-oriented abstraction layer enables companies to run their operations and conduct business with each other in a dynamic and automated fashion
- Business drives IT, and agile IT enables agile businesses

Copyright © 2005, ZapThink, LLC

34 / 80

# Thank You!



Jason Bloomberg  
jbloomberg@zapthink.com



Ronald Schmelzer  
rschmelzer@zapthink.com

Copyright © 2005, ZapThink, LLC

zapthink  
Photos © Lisa Polucci

35