

A stylized sunburst graphic in shades of blue, positioned on the left side of the slide. It features a central circular element with several pointed rays extending outwards, and a decorative flourish at the bottom left.

First Steps to Building a Single View of an SOA

Introducing the SOA Implementation Framework

Ronald Schmelzer
Senior Analyst
ZapThink, LLC



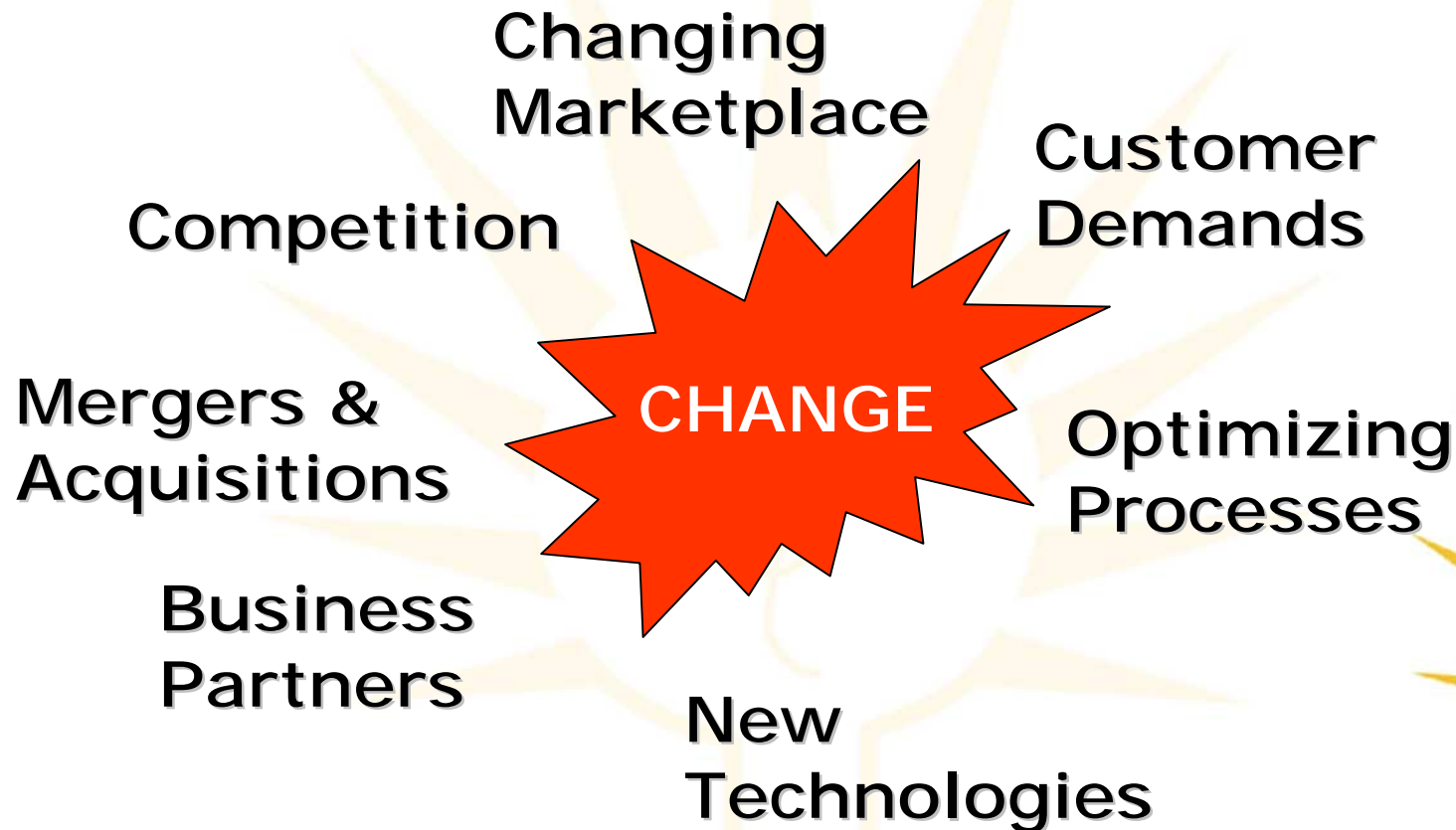
Introduction & Agenda

- 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, integration, process, and architecture tools



zapthink

Business Constant: Change



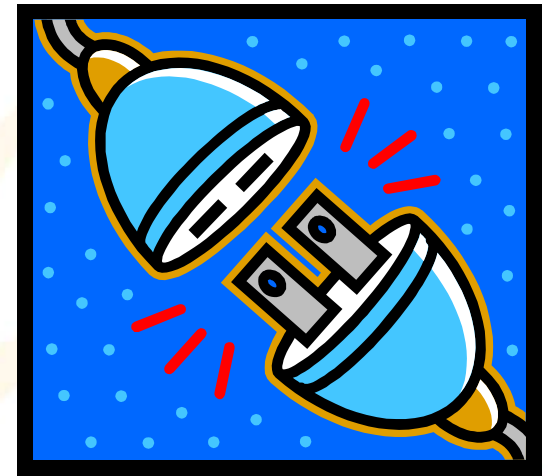
A Business is Never STATIC



zapthink

Making Agility Work: Loose Coupling

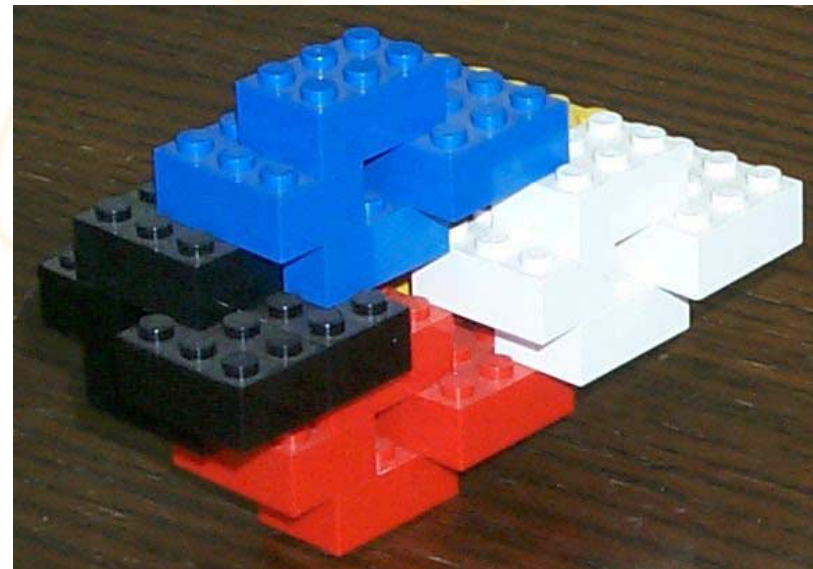
- *Coupling* – the level of common knowledge necessary in a distributed computing exchange
- *Tight coupling* – one participant must have detailed knowledge about the other participant
- *Full decoupling* – the two participants need have no knowledge about each other in order to interact
- *Loose coupling* – the two participants may have specific, limited knowledge about each other





What is Service-Oriented Architecture?

- Access software via discoverable, *loosely coupled* Services
- Users can compose Services into business processes that are also Services
- SOA != Web Services!
- SOA is a *practice* not a *product!*





Tools: Development vs. Architecture

- Architecture is –
 - 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)
- Developer tools different from architect tools
- No real SOA tools market (yet)



zapthink

Introducing the SOAIF...

- The SOA implementation framework is a product or set of products that offer everything companies need to build, run & manage their SOA
- No vendor offers a complete SOAIF (yet)
- Leverages Service orientation for modularity
- Today's markets are converging on the SOAIF



How Technology Segments Fit In...

- Legacy integration
 - Key: wrap legacy systems with Standards-based interfaces to enable them to participate in an SOI
 - Products: EntireX Communicator, XML Gateways, XML Adapters
- Enterprise Service Bus
 - Key: provide a reliable, messaging infrastructure to enable Services to communicate and be composed into business applications
 - Products: EntireX Mediator
- Business process Management (BPM)
 - Key: Allow businesses to achieve agility by defining processes that can be translated into executable Services
 - Products: EntireX Mediator
- Enterprise Information integration (EII)
 - Key: Aggregate information from disparate sources and present a Service interface
 - Products
 - Tamino





Why Should Service-Oriented Architectures be Asynchronous?

- Fundamental tenet of loose coupling: not being aware of end point requirements
- Compositing (virtualized) Web Services may require greater time for processing, requiring asynchrony
- B2B processes are often asynchronous
- Distributed systems can be more reliable when they are asynchronous
- Heterogeneous systems, especially those with limited bandwidth devices, function better asynchronously.
- Support human involvement in processes



zapthink

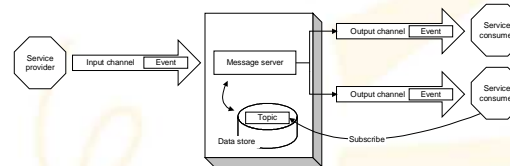
Invocation Mechanisms in SOA

- SOA is more flexible than client/server – supports multiple invocation mechanisms

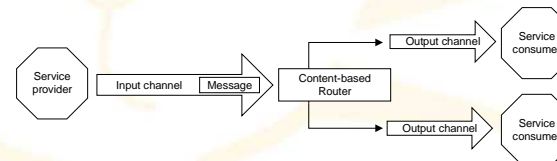
– Request/Reply



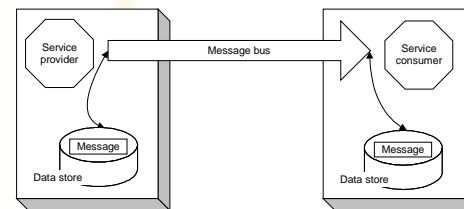
– Publish/Subscribe



– Routed Events



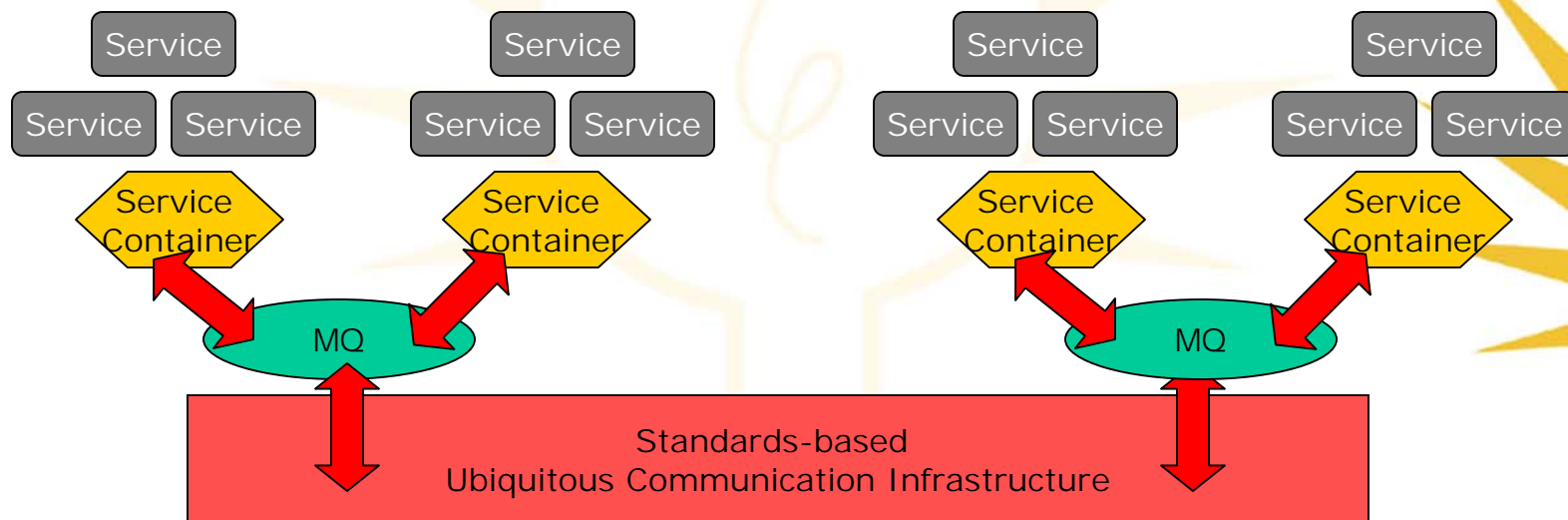
– Reliable Messaging





An ESB makes a SOA Possible

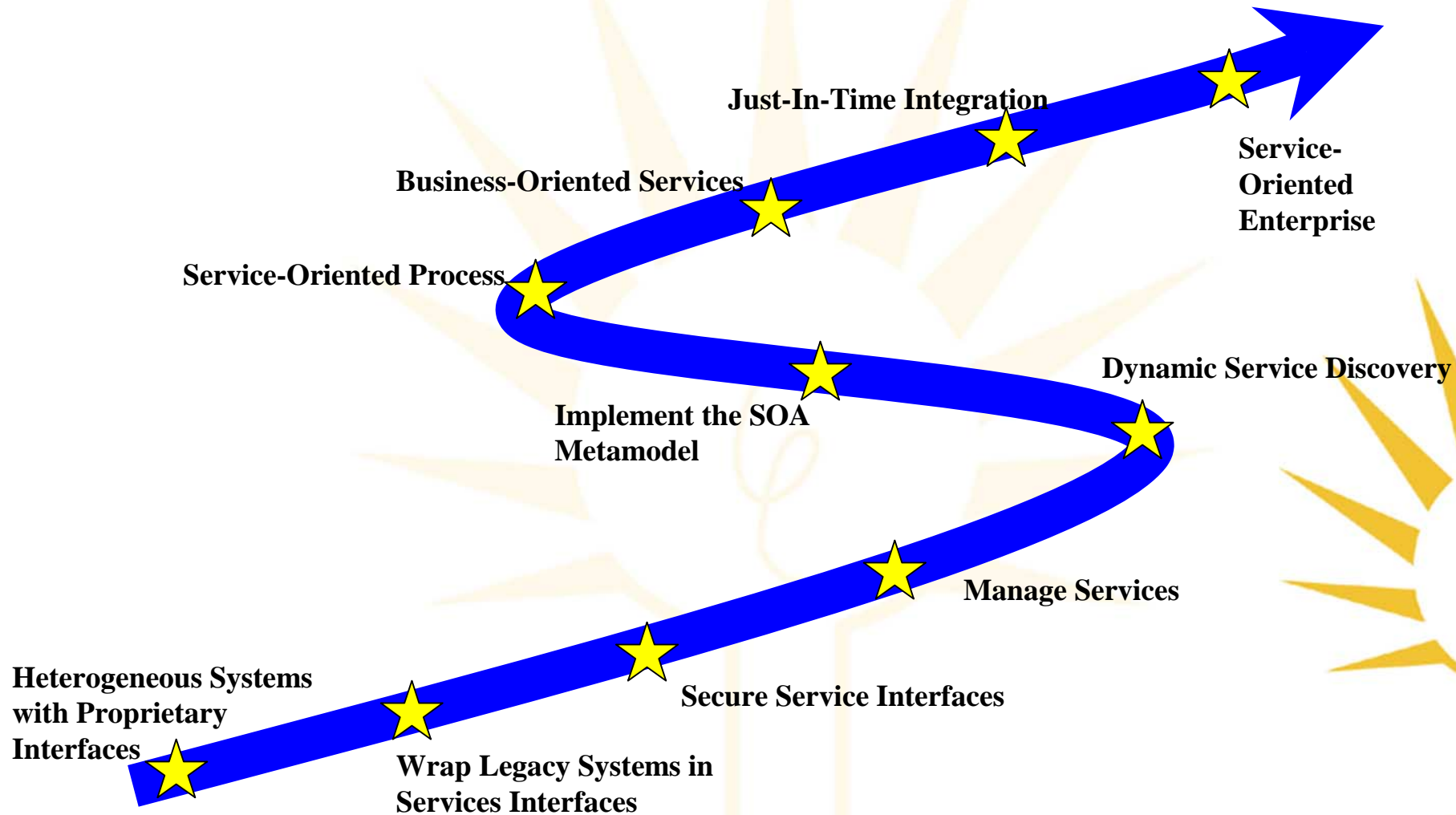
- THREE fundamental tenets of SOA:
Loose Coupling, Coarse Granularity, and ***ASYNCHRONY***
- Making SOAs work - the Enterprise Service Bus





zapthink

The SOA Implementation Roadmap





The State of the Market

- All balls must be in the air at once
- Web Services do not create a permanent, distinct market
- New entrants jockeying for position while incumbents wait/build/acquire





SOA Culture Shock

- SOA crosses product, project, and business responsibility boundaries
- Who is qualified to be an architect anyways?
- What happens to the packaged app. Vendors?
- Just who pays for SOA anyways?
- Incremental vs. Enterprise-Wide





Next Steps?

- Take iterative approach to reduce risk
- Security & management usually come first
- *Conceive* of the SOA top-down (architectural plan)
- *BUILD* the SOA bottom-up (build Services from existing resources)
 - SOA leverages existing resources, so use them!
 - Moving to standards-based interfaces first step
 - Adopting standards-based infrastructure helpful



Thanks!



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

Take Credit for attending this presentation!



- Go to www.zapthink.com/credit and enter the code **SWAGSOA**.
- Download a digital copy of the presentation
- Sign up for our ZapFlash newsletter
- Get a ZapCredit good toward free research and ZapGear!



Ronald Schmelzer
rschmelzer@zapthink.com