



This Presentation Courtesy of the
International SOA Symposium
 October 7-8, 2008 Amsterdam Arena
www.soasymposium.com
info@soasymposium.com

Founding Sponsors



Platinum Sponsors



Gold Sponsors



Silver Sponsors



Avoiding SOA Pitfalls

Radovan Janecek
 Chief Architect, BTO, HP Software
 June 2008

© 2008 Hewlett-Packard Development Company, L.P.
 The information contained herein is subject to change without notice

Eight Years of SOA Wins and Mistakes

- Co-founded Systinet (2000)
 - Web Services stacks in C++ and Java
 - Service Registry
 - SOA Governance
- Led SOA Center in Mercury/HP (2006)
 - SOA Governance, Quality, Management
- BTO Architecture (2008)
 - Service and Data Models
 - Integration strategy (SOA based)



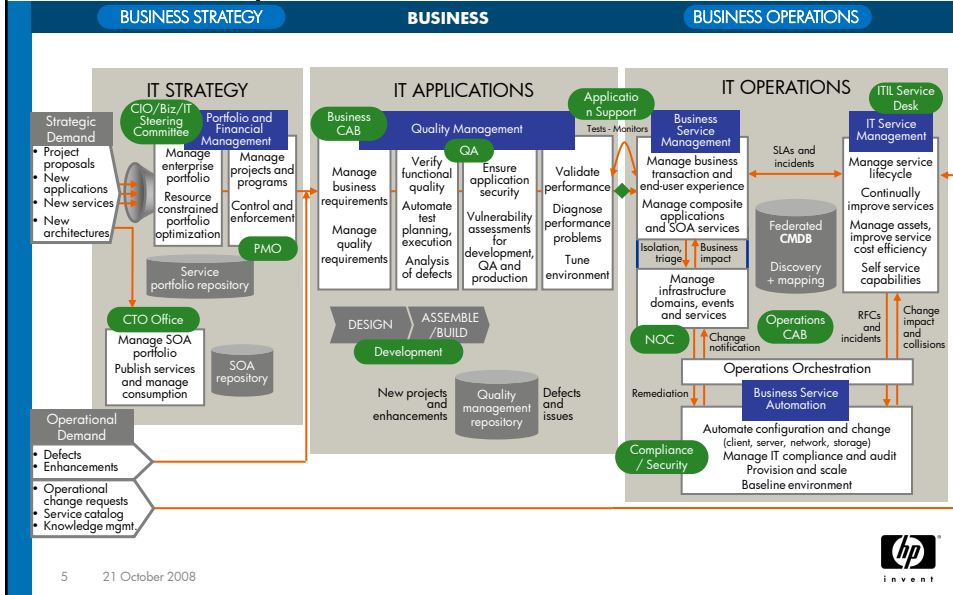
To Remember

- SOA is GOOD as it SIMPLIFIES big initiatives
 - Business Service Management
 - Business Service Automation
 - Service Portfolio Management
- Beware of Snake-Oil Architecture
 - The more EAI the worse SOA
- SOA Governance is a must



4 21 October 2008

BTO Blueprint



LET'S TALK ABOUT PITFALLS



Agenda

In scope

- Organization
- Solutions vs Integrations
- SOA vs EAI
- Point-to-Point vs HUB
- Common Data Model
- API granularity
- Standards

Out of scope

- Performance
- Security
- Language binding
- Testing

7 21 October 2008



Organization

- Project driven SOA
 - Perhaps good validation in small scope
- SOA Governance
 - Lack of
 - Too ambitious
- Only technical view
 - “It’s a software architecture” view

8 21 October 2008



#1: Project-driven SOA

- SOA is implemented within specific project(s)
- Good
 - Validation of the concept
 - Starting point
- Bad
 - Silo reinforcement
 - No proof it will work across silos
- Reasons
 - Alignment with business, Commitment, Experience
 - Financial: funding, incentives
 - Trust!

9 21 October 2008



#1: Suggestion

- Align with business on the importance
 - Cross-portfolio (silo) integrated solutions
 - Identify the most critical solutions (not services!)



Funding Model, Commitments

- Define SOA Governance model



Trust, Experience, Alignment

10 21 October 2008



#2: SOA Governance

- No or wrong governance practices
- Good
 - You can move faster short-term
- Bad
 - JBOWS, poor execution
- Reasons
 - Project scope (hard to find ROI)
 - Technical view (we already have technical governance!)
 - Too ambitious model inherited from project experience

11 21 October 2008



#2: Suggestion (part 1)

- Create centralized R&D counterpart to business for strategic decisions
- Create SOA Center that
 - Defines processes, best practices, compliance guidelines
 - Selects appropriate standards
 - Executes the governance processes
 - Centralizes Service and Data models creation efforts



Expertise, Communication

12 21 October 2008



#2: Suggestion (part 2)

- May centralize Solution Testing and Certification
- Keep development decentralized
 - Creation of centralized “integration team” reinforces “somebody-else’s-problem” behavior
- VISIBILITY
 - Everything online: plans, compliance reports, experience sharing, service rating, catalogs, blueprints



Pragmatic Execution Model

13 21 October 2008



#3: Technical View

- SOA seen as software development detail
- Good
 - Focus on technical excellence
- Bad
 - #1, #2
 - Over-engineered architecture
 - Focus on HOW instead of WHAT
- Reasons
 - SOA is driven mainly by architects
 - Software creation doesn’t matter anyway

14 21 October 2008



#3: Suggestion

Start with #1!

15 21 October 2008



#4: Solutions vs Integrations

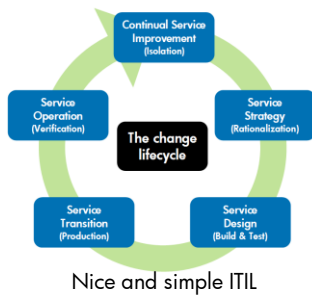
- Building integrations without higher-level view
 - Let's move customer entry from here over there
- Good
 - Integration is done fast
- Bad
 - Too many integrations are not reusable
 - Hard to identify and remove functional overlaps
 - Service and Data model cannot be reasonably created
- Reasons
 - EAI habits, #1 (project-driven soa)

16 21 October 2008



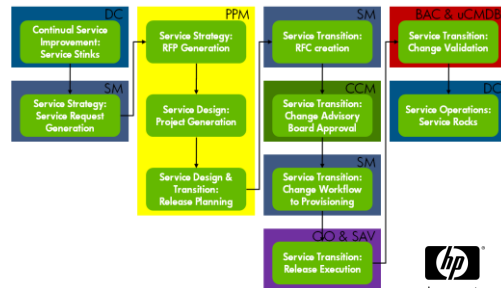
Example: Change Management Solution

- End-to-end
 - From discovering a reason for change
 - Thru planning, approvals, and execution
 - To verifying the effect of the change
 - Multiple reasons for change, multiple workflows/processes



17 21 October 2008

One of multiple scenarios by BTO



#4: Suggestion

Start with #1!

18 21 October 2008



#5: SOA vs EAI

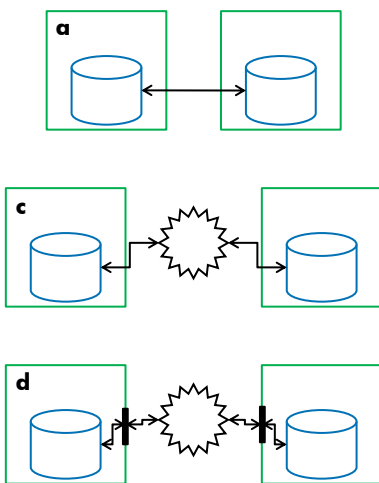
- EAI in angle brackets
- One of the top SOA failure reasons
- Good
 - Leveraging EAI tools and skills
- Bad
 - Everything
- Reasons
 - #1, #2, #3, #4

19 21 October 2008

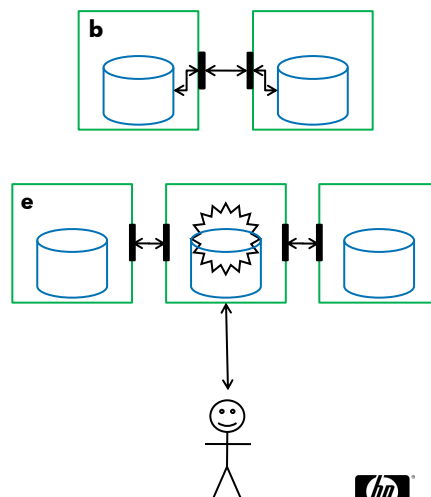


More on SOA vs EAI

EAI



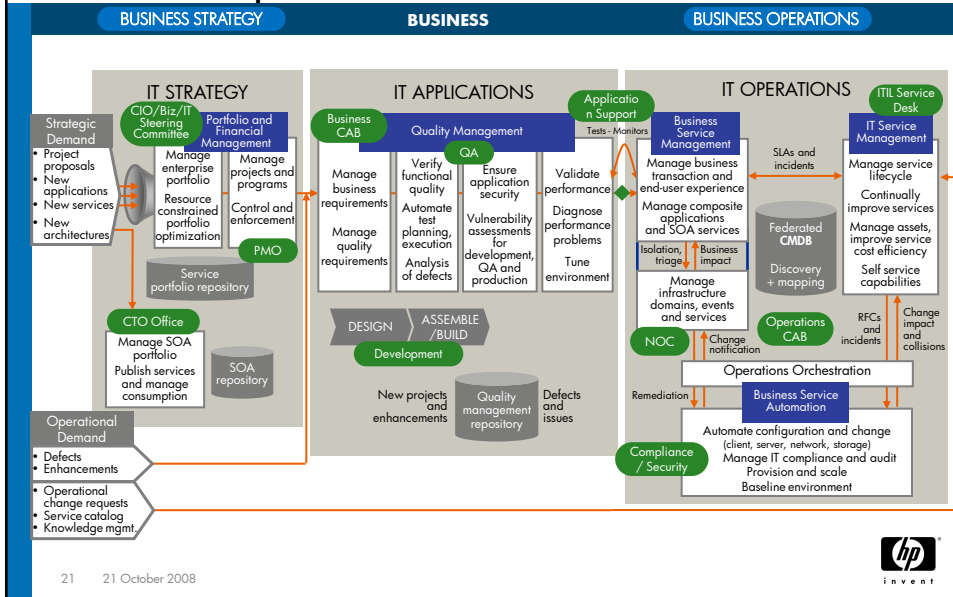
SOA



20 21 October 2008



BTO Blueprint



#5: Suggestions

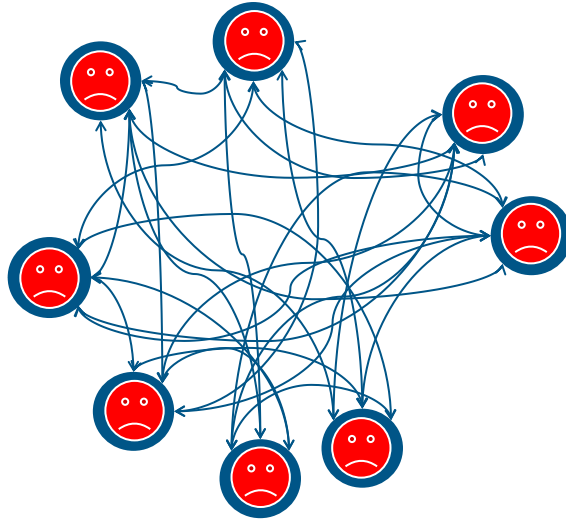
- Observe warning signs
 - “Let’s put these two onto the same database”
 - “We need distributed transactions here”
 - ...
- Be SOA fundamentalist until tightly coupled scenario is needed



Understanding of SOA vs EAI



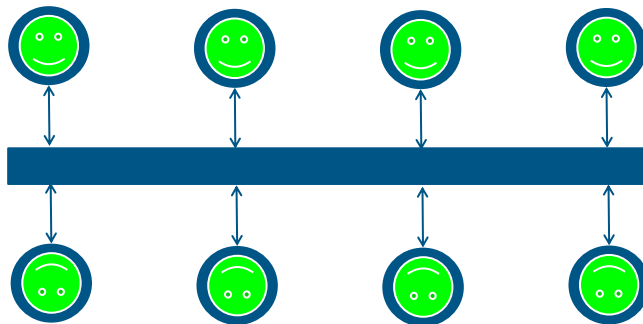
#6: HUB Better Than Point-to-Point



23 21 October 2008



#6: HUB Better Than Point-to-Point



24 21 October 2008



#6: HUB Better Than Point-to-Point

- Nothing wrong on P2P if Governance is in place
- HUB will not help if Governance is missing
- Advantages hypothetical
 - Real dependencies are not that complex
- Disadvantages are real
 - Deployment cost, integration cost (multiple HUBs), evolution issues (multiple places to change)
- HUB de-facto implements additional business logic
 - E.g. content based routing, orchestration, etc.
 - Who owns it? What about contracts?
 - Why is this logic not provided by a service?



25 21 October 2008

#6: Suggestion

- SOA: Service, Consumer, Contract – no HUB
- Use Service Registry for late binding
- Strictly use middleware-type HUBs **behind** service's façade
- Do contract management (even very simple one helps)



Time saving, Right focus, Success



26 21 October 2008

#7: Common Data Model

- False: Strict CDM is a must for SOA success
- Good
 - Common vocabulary and shared data structures help
- Bad
 - Slows down too much
 - Questionable ROI
- Reasons
 - EAI thinking not realizing SOA has bigger scope

27 21 October 2008



#7: Suggestion

- Align on key business taxonomies
- Define data model guidelines
 - Standards, metadata, evolution, customizations
- Identify key use cases (solutions) and key services
- Allow for relaxed semantics across them
- Again: model is driven by **contract**

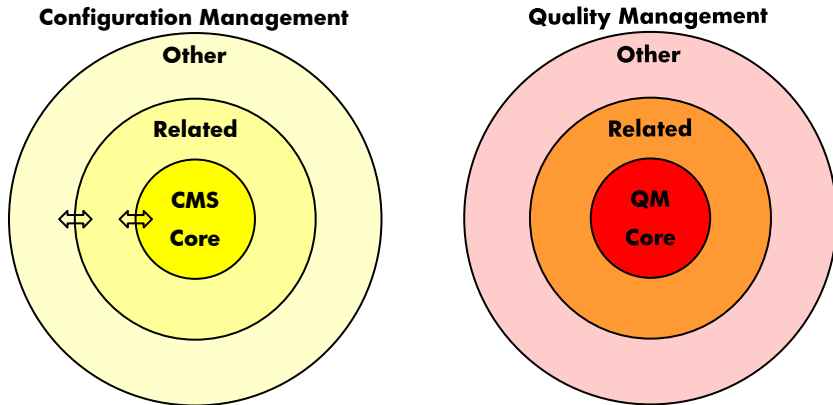


Data Model will grow with your SOA

28 21 October 2008



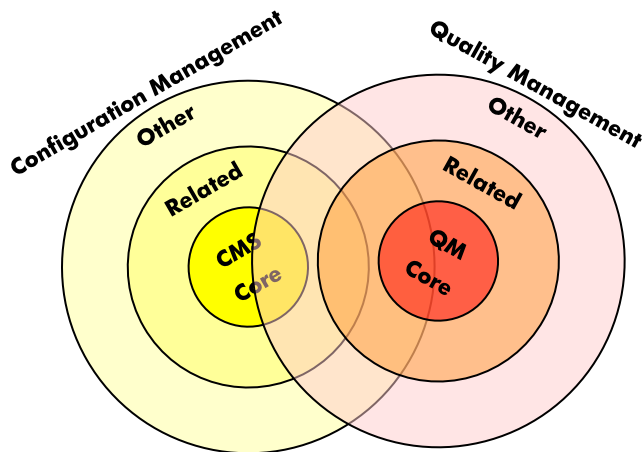
#7: Suggestion Visual



29 21 October 2008



#7: Suggestion Visual



30 21 October 2008



#8: API Granularity

- Services provide rich 'chatty' interfaces
- Good
 - Fast legacy API re-use
- Bad
 - Tight coupling
 - Exploding complexity
- Reasons
 - Services treated as components
 - Low control over 3rd party software

31 21 October 2008



#8: Suggestion

- Refactor existing API
 - Consider REST
- Move as much business logic to the endpoints as possible

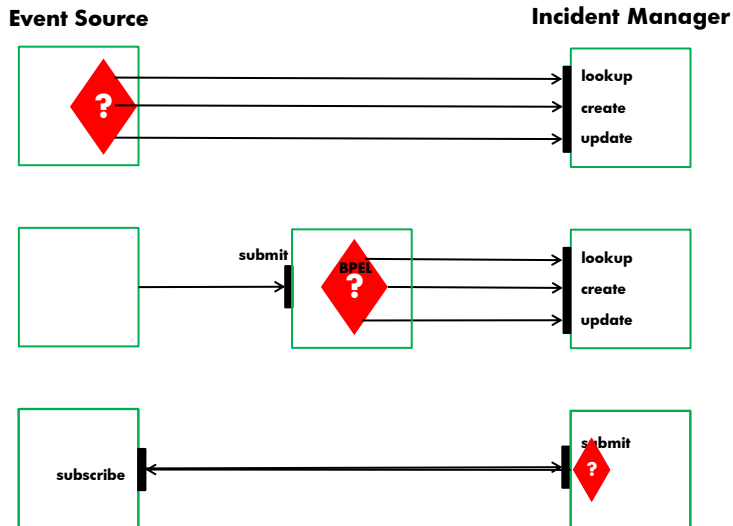


Less features, More reliability

32 21 October 2008



#8: Suggestion Visual: Create Incident



33 21 October 2008



#9: Standards

- Standards are not enough!
 - Generic envelopes
 - Industry standards often 'tailored' when used
- Data externalization rules
 - Mapping to standards
 - Dates, Versions, References, MIME types, etc.
 - Identification
 - Cross references (hyperlinks?)
- Business vocabulary and taxonomies
- Look carefully at adoption outside of your company

34 21 October 2008



Summarizing...

- SOA is more about good methodology and process rather than technology
 - More guidelines than middlewares
 - More communication than features
- Beware of pitfalls
 - Most of them come from 'legacy thinking'
- Governance is key as we are working on 'global' level

35 21 October 2008



Q&A

THANK YOU

36 21 October 2008

