Presentation: A Research Agenda for Service-Oriented Architecture

MESOA Contributions to the Evolution of the Agenda

Grace A. Lewis
Software Engineering Institute, USA
SOA Research Agenda: Development Approach

Assembled an international research group to analyze the current state of the practice and current research initiatives in SOA

Proposed a long-term consensus research agenda

Performed an extensive literature review and looked at case studies of successful SOA adoption

Created a service-oriented systems development life cycle that supports the strategic approach to SOA adoption shown in case studies

Identified areas of SOA research necessary to fill in the gaps

Validated and evolved finding through multiple workshops and interviews
Successful SOA adoption has a strong link between business strategy and SOA strategy.
Expanded View of the SOA Problem and Solution Space

SOA strategy is the way in which SOA is going to address the organization's business drivers for SOA adoption.

The organization's domain area and context enable and/or constrain the SOA strategy.

Feedback loops reflect the dynamic and iterative nature of service-oriented environments.

SOA plans are executed to produce or evolve a service-oriented system.

Measurements are gathered to test the effectiveness of the strategy and the system itself.

SOA plans are executed to produce or evolve a service-oriented system.

Measurements are gathered to test the effectiveness of the strategy and the system itself.
# Mapping Between Phases, Activities and Indicators

<table>
<thead>
<tr>
<th>PHASES</th>
<th>P1: Strategic Analysis</th>
<th>P2: Planning</th>
<th>P3: Construction</th>
<th>P4: Transition</th>
<th>P5: Production</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVITIES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A1: Business Context Understanding</td>
<td>+++</td>
<td></td>
<td></td>
<td></td>
<td>++</td>
</tr>
<tr>
<td>A2: Business Objectives Specification</td>
<td>+++</td>
<td>++</td>
<td>+</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A3: Risk Analysis and Initial Requirements Gathering</td>
<td>+++</td>
<td>+++</td>
<td>++</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>A4: Prototyping and Requirements Tuning</td>
<td>++</td>
<td>+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A5: Implementation</td>
<td>+</td>
<td>+++</td>
<td>++</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A6: Integration and Testing</td>
<td>++</td>
<td>+++</td>
<td>++</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>A7: Deployment</td>
<td>+++</td>
<td>++</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A8: Maintenance</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A9: Management</td>
<td>++</td>
<td>+++</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INDICATORS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I1: Financial Indicator Measurements</td>
<td>++</td>
<td>+++</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I2: Technology Indicator Measurements</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>++</td>
</tr>
<tr>
<td>I3: User Rating Measurements</td>
<td>++</td>
<td>+++</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I4: Compliance Indicators</td>
<td>+</td>
<td>++</td>
<td>+</td>
<td>++</td>
<td>+++</td>
</tr>
</tbody>
</table>
Relationship between Solution Space and Research Topics

The development of a service-oriented system requires business, engineering and operations to be made, as well as other cross-cutting decisions.

Our proposed taxonomy of research topics is divided into these decision areas.

The research topics correspond to areas where new/more/different research is needed to support a strategic approach to service-oriented systems development.
Taxonomy of Research Issues

SOA Research Taxonomy

Business
- SOA Strategy Selection
- Business Case for Service Orientation
- Mapping between Business Processes and Services
- Organizational Structures to Support Service-Oriented Environments
- Business Indicators

Engineering
- Process and Life Cycle Requirements
- Service Selection
- Service Definition and Categorization
- Technology Assessments
- Architecture and Design
- Code
- Tools and Products
- Quality Assurance and Testing
- Deployment
- Maintenance and Evolution
- Engineering Indicators

Operations
- Adoption
- Monitoring
- Support
- Service-Level Agreements in SOA Environments
- Operations Indicators

Cross-Cutting
- Governance
- Security
- Training and Education
- Risk Management in SOA Projects
- Social and Legal Issues
Sample of Engineering Research Topics

- Service-Oriented System Life Cycle Models
- Development Processes and Methodologies for Service-Oriented Systems
- System or End-to-End Testing
- Infrastructure Testing
- Simulation and "What-If" Analysis Testing
- Practices for Service Providers to Support System Testing of their Consumers
- Establishing Test Beds and Benchmarks
- Service Certification

- Process and Life Cycle
- Requirements
- Service Selection
- Service Definition and Categorization
- Technology Assessments
- Architecture and Design
- Code
- Tools and Products
- Quality Assurance and Testing
- Deployment
- Maintenance and Evolution
- Engineering Indicators

- Service Modeling Languages
- Modeling Dynamic Runtime Architecture of Service-Oriented Systems
- Features and Properties for Next-Generation Frameworks for Development of Service-Oriented Systems
- Architectural Styles for Service-Oriented Systems
- Communication and Connectors
- Architectures for Service Types
- Design for Personalization, Context Awareness and Adaptation
- Design for Service Composition
- Design for Runtime Semantic-Based Discovery and Composition
- Design for Recovery-Oriented Computing
- SOA and Product Lines
- Design for Mobility
- Real-Time SOA
Sample of Business Research Topics

- Techniques for Service Identification
- Techniques and Processes to Support the Strategic Reuse of Legacy Components and Services
- Analytic Methods for Service Evaluation against Business Needs
- Techniques and Processes for Establishing Relations between Business and Service Models
- Processes to Support the Adaptation of Services to Meet Changes in Business Processes
- Process Refactoring and Reengineering and its Impact on Services

Business

- SOA Strategy Selection
- Business Case for Service Orientation
- Mapping between Business Processes and Services
- Organizational Structures to Support Service-Oriented Environments
- Business Indicators

Techniques to Establish and Document the Business Case for SOA Adoption
- Cost Models for SOA
- Funding Models for SOA
- SOA Business Value Framework

Models for Organizational Structures that Enable Service-Oriented Systems Development
- Skills Required to Develop, Use and Maintain Service-Oriented Systems
- Models for Workforce Allocation in Service-Oriented Systems Projects
- Organizational and Funding Models for Shared Services
Sample of Operations Research Topics

- Monitoring of Business Processes in an SOA Environment
- Operations Monitoring for Auditing Purposes
- Self-Healing Service-Oriented Systems
- Resource Allocation and Configuration Management in SOA Environments
- Verification and Validation of Policies at Runtime

Operations

- Adoption
- Monitoring
- Support
- Service-Level Agreements in SOA Environments
- Operations Indicators

- Service Usability
- End-User Service Composition Tools
- Models of Service Consumer Adoption
- Pricing Models for Service Providers

- Processes for Support of Service-Oriented Systems
- Front-end and Back-end Problem Management in Service-Oriented Environments
Sample of Cross-Cutting Research Topics

- Identity Management in Multi-Organizational SOA Environments
- Secure Dynamic Service Composition
- Security Management in Distributed SOA Environments
- Trust Establishment and Trust Brokering

- Governance
- Security
- Training and Education
- Risk Management in SOA Environments
- Social and Legal Issues

- Techniques and Guidelines to Develop SOA Governance
- Enterprise-Wide vs. Local SOA Governance
- Modeling of Policy, Risk and Trust
- Compliance Monitoring

- Establishing the Area of Services Science
- Investigating and Developing Appropriate University Curricula

Software Engineering Institute
Carnegie Mellon

MESOA 2010
September 17, 2010
© 2010 Carnegie Mellon University
Research Topics in Maintenance and Evolution of Service-Oriented Systems

- Process and Life Cycle
- Requirements
- Service Selection
- Service Definition and Categorization
- Technology Assessments
- Architecture and Design
- Code
- Tools and Products
- Quality Assurance and Testing
- Deployment
- Maintenance and Evolution
- Engineering Indicators

- Tools, Techniques and Environments to Support Maintenance Activities
- Multilanguage System Analysis and Maintenance
- Reengineering Processes for Migration to SOA Environments
- Transition Patterns for Service-Oriented Systems
- Runtime Monitoring of Service-Oriented System Evolution

NEW
Tools, Techniques and Environments to Support Maintenance Activities — Challenges and Gaps

Areas: maintenance processes, change impact analysis, change management and version control, organizational structures and roles

Development of specialized methods and tools to support the maintenance and evolution of large service-oriented systems is in the early stages

- Current efforts seem to indicate that maintenance activities for service-oriented systems are not that different than in traditional systems
- However, we are still in the stage where most service-oriented systems are deployed for internal integration, where there is still some control over deployed services

Emergence of market for third-party services and the deployment of more service-oriented systems that cross organizational boundaries will have to change current maintenance practices
<table>
<thead>
<tr>
<th>Year</th>
<th>Title</th>
<th>Authors</th>
<th>Post-MESOA Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Testing a Service-Oriented Architecture: Installing Order in a Chaotic World</td>
<td>Harry Sneed (ANECON, GmbH, Austria)</td>
<td>• A pilot project for migrating COBOL code to web services</td>
</tr>
<tr>
<td>2007</td>
<td>Challenges of SOA Testing</td>
<td>Scott Tilley (Florida Institute of Technology, USA)</td>
<td>• Documenting service-oriented systems</td>
</tr>
</tbody>
</table>
Multilanguage System Analysis and Maintenance — Challenges and Gaps

Most research in this area is limited to small size projects and a small number of languages, which is a problem for an environment that promotes platform independence.

In the case of third-party service providers, access to source code is most probably not possible.

- If that is the case, an important area of research is the identification of the type of information that service providers would need to expose to service consumers that wish to do code analysis, as well as tools to support the process.
## Multilanguage System Analysis and Maintenance — MESOA Contributions

<table>
<thead>
<tr>
<th>Year</th>
<th>Title</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>Static Analysis and Measurement of Web Service Interfaces</td>
<td>Harry Sneed <em>(ANECON, GmbH, Austria)</em></td>
</tr>
</tbody>
</table>
Reengineering Processes for Migration to SOA Environments — Challenges and Gaps

The ideal reengineering process would be one that implements the SOA-Migration Horseshoe

- Currently techniques and tools that implement portions of the horseshoe but not the full horseshoe
- An important area of research would be the development of concrete processes that implement the horseshoe and tools (or suites of tools) to support the process

Real challenge is mining legacy code for services that have business value

- Tools and techniques for analyzing large source code bases to discover code that is of business value
- Metrics for "wrapability" and business value to determine reusability
- Application of feature extraction techniques to service identification
## Reengineering Processes for Migration to SOA Environments — MESOA Contributions

<table>
<thead>
<tr>
<th>Year</th>
<th>Title</th>
<th>Authors</th>
<th>Post-MESOA Work</th>
</tr>
</thead>
</table>
| 2007 | Towards Service Identification from Legacy Code: An Integrated Approach | Lerina Aversano, Massimiliano Di Penta, Ciro Palumbo (University of Sannio, Italy) | • Use of BPR and workflow analysis to support maintenance and evolution of service-oriented systems  
• METAMORPHOS: MEthods and Tools for migrAting software systeMs towards web and service Oriented aRchitectures: exPerimental evaluation, usability, and tecHnOlogy tranSfer  
• An Empirical Comparison of Methods to support QoS-Aware Service Selection |
| 2007 | Factors Impacting Concept Location in Service-Oriented Enterprise Software | Oleksandr Panchenko (Hasso Platner Institute, Germany) | • Generic Web Services - Extending Service Scope While Preserving Backwards Compatibility |
| 2007 | On The Business Value of SOA | Scott Tilley (Florida Institute of Technology, USA) |  |
| 2007 | Value-Focused Research Directions on SOA Applications | Ned Chapin (InfoSci, USA) | • Research on Maintenance Characteristics of SOA Systems (MESOA 2009) |
## Reengineering Processes for Migration to SOA Environments — MESOA Contributions

<table>
<thead>
<tr>
<th>Year</th>
<th>Title</th>
<th>Authors</th>
<th>Post-MESOA Work</th>
</tr>
</thead>
</table>
| 2008 | Migration of Legacy Systems to the Web: Challenges for Migration to SOA | Andrea de Lucia *(University of Salerno, Italy)* | • METAMORPHOS: MEthods and Tools for migrAting software systeMs towards web and service Oriented aRchitectures: exPerimental evaluation, usability, and tecHnOlogy tranSfer  
• Evaluating legacy system migration technologies through empirical studies (Case study to evaluate the use of MELIS (Migration Environment for Legacy Information Systems) for the migration of legacy COBOL programs to the web)  
• Software migration projects in Italian industry: Preliminary results from a state of the practice survey |
| 2009 | A Method for Automating Model Evolution of Service-Oriented Systems | Sedigheh Khoshnevis, Pooyan Jamshidi, Reza Teimourzadegan, Ali Nikravesh, Alireza Khoshkbarforushha, and Fereidoun Shams *(Shahid Beheshti University, Iran)* | |

---

**Software Engineering Institute**

**Carnegie Mellon**

**MESOA 2010**  
**September 17, 2010**

© 2010 Carnegie Mellon University
Transition Patterns for Service-Oriented Systems — Challenges and Gaps

If a new system is built to replace the old system in \( n \) steps then the Total Modernization Cost (TMC) is

\[
TMC = \sum_{i=1}^{n} (DC_i + IC_i + DMC_i + TC_i)
\]

where

- DC = Development Cost
- IC = Implementation Cost (Deployment + Infrastructure)
- DMC = Data Migration Cost
- TC = Transition Cost

And, if we define Transition Cost as \( TC = ECC + NTC + TI \), where

- ECC = Existing Code Changes + Maintenance Cost
- NTC = New Throwaway Code + Maintenance Cost
- TI = Temporary Infrastructure and Operation Cost

Source: Ongoing work between Grace A. Lewis from Software Engineering Institute and Parviz Dousti from Carnegie Mellon University
Transition Patterns for Service-Oriented Systems — Challenges and Gaps

What would be the process and supporting techniques and tools that could determine

- The **right number of increments**
- that would **minimize the throwaway costs** due to
  - Temporary infrastructure such as gateways and ETL tools
  - Temporary code to deal with mismatches
  - Changes to legacy code waiting to be modernized, e.g. adding code to invoke a service knowing that it will be modernized in a future increment
- **in a repeatable fashion** such that it could be recalculated when business changes its mind about priorities

Source: Ongoing work between Grace A. Lewis from Software Engineering Institute and Parviz Dousti from Carnegie Mellon University
Runtime Monitoring of Service-Oriented System Evolution — Rationale

“Service-oriented systems projects are maintenance-intensive projects.” — Ned Chapin

One of the benefits of SOA adoption is business agility

- Ability to create and change systems as the business changes
- Based on the identification of reusable services with high business value

Service-oriented systems are implemented as a collection of collaborating services and systems

- Collaboration usually takes the form of workflows that enact specific business processes

However, it is difficult to monitor the satisfaction of requirements, reference architectures, SLAs and business goals as business processes and services change

- Service-oriented systems need to incorporate self-adaptation and self-management mechanisms
Runtime Monitoring of Service-Oriented System Evolution — Current Efforts

Autonomic computing elements as a way to validate SOA governance policies at runtime

Given the strong links between business strategy and SOA, runtime monitoring is a best practice to verify if business goals are being met

- Current SOA infrastructures provide capabilities to define metrics and collect data—real challenge is not how to measure what to measure
- There is always a tradeoff between data collection and performance
- Unprecedented levels of monitoring and instrumentation technology

With the emergence of the third-party service market, SLAs have to be monitored continuously during software usage and evolution

If the thresholds of defined metrics are reached, how does the service-oriented system adapt at runtime?

- What are adaptation strategies given the type of metric?
- What is the effect on service consumers—applications, business processes, other systems, …?
## Runtime Monitoring of Service-Oriented System Evolution — MESOA Contributions

<table>
<thead>
<tr>
<th>Year</th>
<th>Title</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>Implications of Autonomic Computing for SOA Maintenance and Evolution</td>
<td>Hausi Müller (<em>University of Victoria, Canada</em>)</td>
</tr>
<tr>
<td>2008</td>
<td>Runtime Monitoring of Service-Oriented Systems: Implications for Maintenance and Evolution</td>
<td>Hausi Müller (<em>University of Victoria, Canada</em>)</td>
</tr>
<tr>
<td>2009</td>
<td>SOA Governance Optimizes the Business and Evolution of Service-Oriented Systems</td>
<td>Hausi A. Müller, Priyanka Gupta, Ron Desmarais, Alexey Rudkovskiy, Norha Milena Villegas, Qin Zhu (<em>University of Victoria, Canada</em>) and Leho Nigul (<em>IBM Center for Advanced Studies, Canada</em>)</td>
</tr>
</tbody>
</table>
SOA Research Agenda: Conclusions on Key Challenges

Engineering challenges are significant if SOA is to be used in “advanced ways”, e.g.

- Multi-organizational
- Dynamic discovery and composition
- Real-time applications

Main challenges for enterprise applications are related to business and operations, and not engineering. As third-party services become the new business model, there needs to be support for

- Service-level agreements
- Runtime monitoring
- End-to-end testing involving third parties
- Pricing models for third-party services
- Service usability—from a design and an adoption perspective
SOA Research Agenda: Conclusions on Key Challenges

There are some areas where what is needed is not more basic research, but rather non-vendor surveys, studies and experiments to produce more concrete guidance

- SOA governance
- Business case for SOA adoption
- ROI for SOA adoption
- Development processes and practices for service-oriented systems development

There needs to be more collaborative research between industry and academia to create real practices
Future of the SOA Research Agenda

The SOA Research Agenda will not be published as a “full volume”

- Huge amount of work
- Difficult to keep updated

We will publish one or two “leaves” of the agenda per year

- Rationale
- Current efforts
- Challenges and gaps

We encourage all of you to work on “leaves” of the agenda

- We can participate as co-researchers, reviewers or just “cheerleaders”
- We can publish the work as an SEI report — high visibility in the software engineering community
NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.