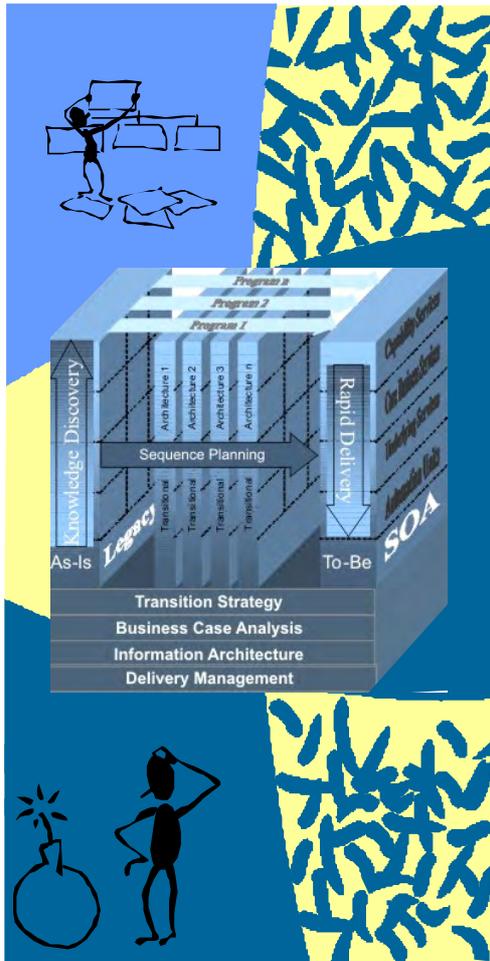


CBDIJournal



Practice Guide

SAE2 Framework for Application Modernization

As the world emerges from recession there is growing interest in applying modernization techniques and technologies to reduce complexity and cost and respond to rapidly changing, post recession business models. In this report we explore how the CBDI SAE framework can be evolved to provide a repeatable, business driven backplane for effective application modernization practice.

By David Sprott

Originally published in the December 2009 edition of the CBDI Journal



Independent Guidance for Service
Architecture and Engineering

Practice Guide: SAE2 Framework for Application Modernization

Application Modernization is a commonly used term covering a wide range of practices and technologies. As the world emerges from recession there is growing interest in applying modernization techniques and technologies to reduce complexity and cost and respond to rapidly changing, post recession business models. In this report we explore how the CBDI SAE framework can be evolved to provide a repeatable, business driven backplane for effective application modernization practice.

By David Sprott

Introduction

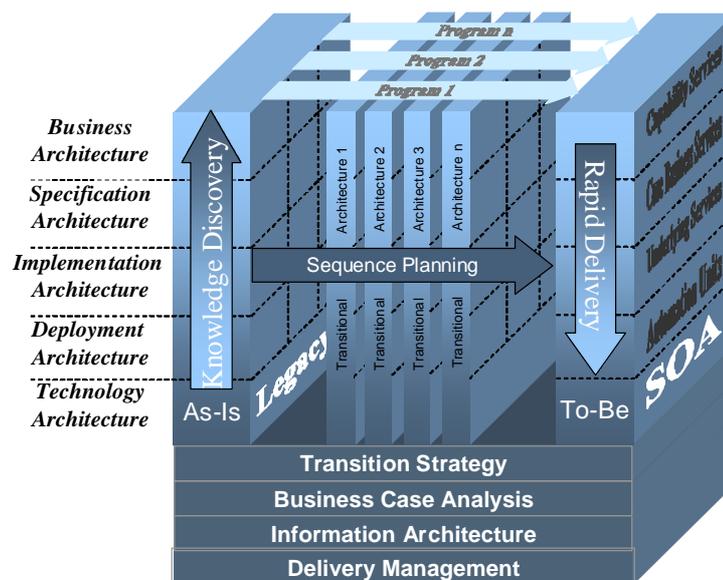


Figure 1 – SAE2 Modernization Framework - the Rich Picture

We are all acutely conscious of the disastrous state of the typical application portfolio and the constraints placed upon business by inherently high cost, complexity and inflexibility. Whilst this has been true for many years, it's clear that post recession business pressures are increasing the priority to address the issue.

No surprise Application Modernization (AM) is rapidly becoming a mainstream strategy.

While AM has been around for many years, there are many interpretations of the concept. Most modernization efforts appear to be focused on the technology, evidenced by the thriving market in tools, technologies and resources for language and platform migrations including virtualization of server and network. The same cannot be said for business focused modernization.

Results from a recent industry survey¹ make clear that a majority of IT leaders are planning AM initiatives. Further, a significant proportion is also planning to make SOA the core target architecture.

SOA won't be the only architecture concept in use, but it will be the dominant influence because it addresses many of the limitations of current AM efforts. It focuses on key outcomes including business agility and reuse, and creates an integration architecture that is equally useful in modernization and rationalization as it is in responding to unforeseen business requirements. And many business and IT leaders are anticipating the next decade will be more turbulent than the last².

Figure 1 illustrates how a structured AM approach could be organized. The challenge is to re-architect existing systems in context with the To-Be business and selectively harvest assets to accelerate business delivery programs that rapidly deliver an SOA based on better structure at all layers of the architecture. This process is supported by cross cutting capabilities for project management, information architecture, business case analysis and transition strategy, management and coordination.

Everware-CBDI has developed the Service Architecture & Engineering (SAETM) Framework over the past five years to create a backplane for structured, repeatable SOA activity. The framework is distinctive being:

- Business driven
- Strongly data oriented
- Meta model based

In this report we will outline how the framework can support the AM process. We will refer to the enhanced framework as SAE2.

The SAE2TM Framework

CBDI drew the original SAE reference model in 2005³ and since then have progressively detailed the concepts in numerous guidance reports and the SAE Knowledgebase methodology⁴. SAE2 (Figure 2) looks remarkably similar.

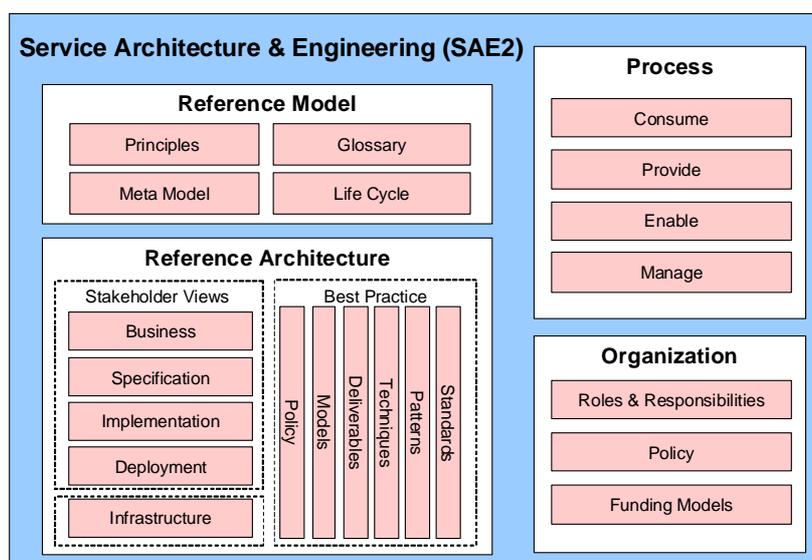


Figure 2 – SAE2 Reference Framework

Readers familiar with SAE will notice firstly that the primary elements of the Reference Framework are essentially unchanged. Second the references to SO and SOA are removed. The framework is intended therefore to be more broadly applicable, not specific to SOA, but straightforward for existing users to migrate to.

We have termed the framework SAE2, indicating that while the impact of the extended scope is significant, the core remains stable. We explore the extensions in the following sections.

Reference Model

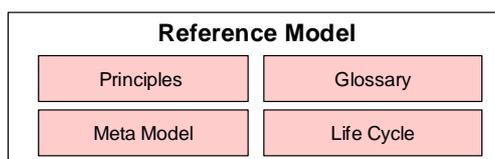


Figure 3 – Reference Model

Principles

The existing architecture principles underlying SAE are entirely appropriate in SAE2. However we recommend additional principles that guide modernization architecture:

- Model driven architecture and design – enables governance and ongoing management of inter and intra application architecture.
- Component based implementation – ensures containment of complexity, defined responsibilities, boundaries and dependencies.
- Application knowledge as a byproduct of delivery platforms and tools – ensuring precise and current documentation of the (business logic and rule) details of the application.
- Defined change management capability with SLA.

Everware-CBDI recommends that AM should be model driven (ideally, but not necessarily using MDA/MDD per se) and service and component based in order to achieve the same level of agility and asset management that can be achieved in the service architecture, at the implementation layer. These two principles can easily be measured by the existence of a) models and b) well formed components and are viewed as important characteristics of modernization and key criteria for assessment of existing systems.

A defining feature of legacy applications is the absence of understanding of what an application does. In contrast a modernized application should not be defined solely by being coded in a modern language, but should also have detailed documentation of structure, business rules and logic compliant with a comprehensive meta model as an integral part of the technology architecture and delivery process. In this way currency and accuracy should be guaranteed.

Finally there has been widespread debate about what agility means, but few answers. The obvious way to tell whether an application is agile is to have explicit agility architecture together with a Change Service Level Agreement that provides estimating guidelines for relevant types of change. Then it would at least be measurable.

Meta Model

The meta model underlying SAE2 will be substantially unchanged from the existing SAE V3 model⁵. However there are a number of areas where the model will evolve to accommodate modernization requirements. Figure 4 illustrates a conceptual view of how the model will change. Actual changes will be the subject of detailed modeling and review.

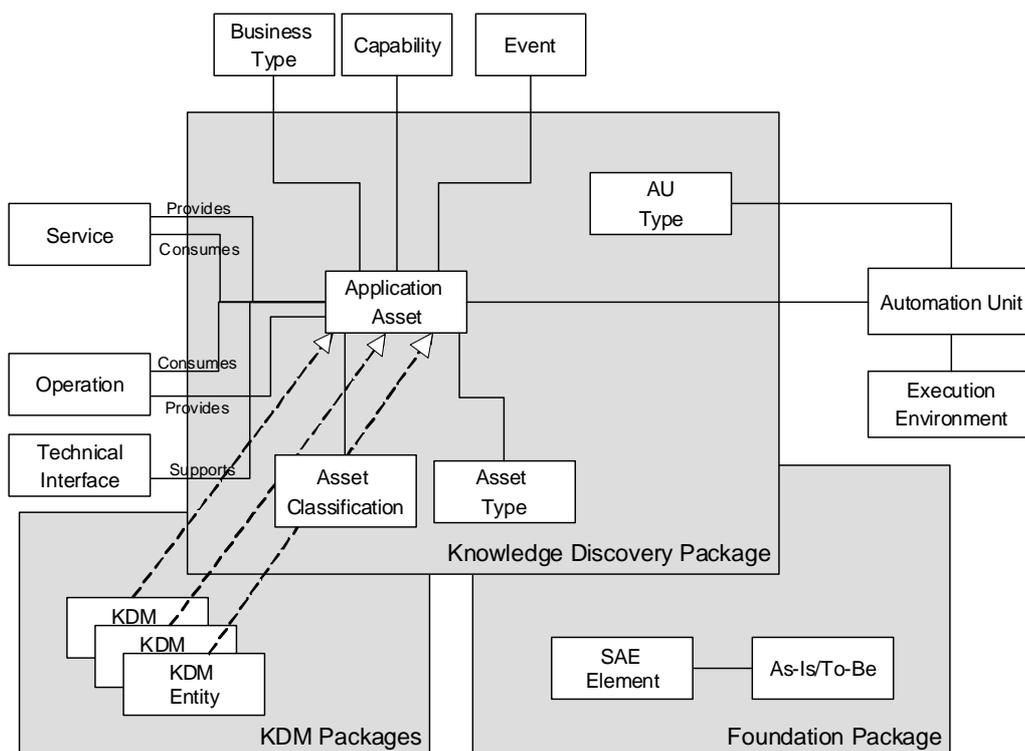


Figure 4 – Meta Model

1. We envisage there will be a Knowledge Discovery package that models details of existing applications. It is probable that this package will comprise mostly relationships with existing meta classes as shown in Figure 4.
2. Continuing long standing CBDI policy, our intent will be to align with relevant industry standards. The OMG Knowledge Discovery Meta-model (KDM)⁶ provides a basis for representing existing software assets and operational environments.
3. The KDM provides a very detailed meta model designed to support asset reengineering and migration between tools. We envisage adopting the meta model in a progressive manner. It is anticipated that KDM alignment will be achieved by subtyping existing SAE meta concepts, initially at a level of abstraction that captures concepts, structure, components and dependencies. This involves elements of the Abstractions Layer (Conceptual, Build and Structure) and the Resource Layer (UI, Data, Platform, Event). Further levels of detail in the Program Elements and Infrastructure layer will be considered on a case by case basis.

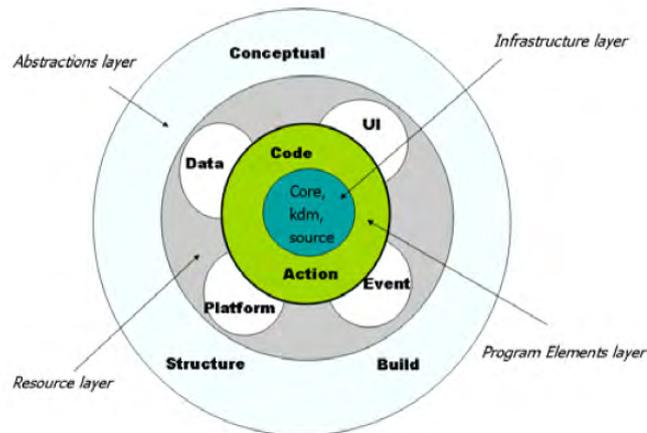


Figure 5 – KDM Layers and Packages (Source OMG)

4. A fundamental change to the model will be required that allows models to record As-Is or To-Be state. This requirement will apply to all meta classes, and it is possible this may be incorporated simply by using Foundation meta classes.

Glossary

We have commented that there are many interpretations of the concept of Application Modernization. Whilst this might be adequate for purely technology focused efforts, it will cause confusion in architecture led work and particularly in interpretation of governance requirements. We recommend the reference model must provide unambiguous definitions along the following lines:

Application Modernization (AM) is defined as the rationalization of one or more applications or a portfolio to improve business support, technology usage and life cycle and run-time delivery process.

The objectives of AM may include:

- Rationalization - eliminate duplication; making multiple overlapping applications consistent.
- Modernization – upgrade delivery and operational technology and processes including managed service, offshore, outsourced delivery.
- Componentization – reorganize existing arbitrary boundaries to align with business morphology and enable business flexibility.
- Service Enablement – move to service architecture that aligns with business capabilities, services and events.

We recognize there is a wide range of terminology in general use including Integration, Migration, Reengineering, Rewrite, Replacement, Acquire, Buy not build, Elimination, Functional Improvement, Outsourcing and Offshoring. Some of these might be appropriate within the context of AM to describe specific technologies and patterns.

Asset Transformation Life Cycle

There will be considerable variation in how existing software assets are managed both in terms of the scope and granularity of tracked assets as well as the competence of the tools used. It would seem sensible to establish management of assets that are to be modernized in whatever management environment has been selected as strategic for the enterprise.

The modernizing asset life cycle states to be tracked should reflect the modernization state including:

- Assessed
- Transition planned
- Knowledge discovered
- In transition
- Modernized
- Operational

Note these states may vary depending upon the modernization scenario. It will also be important to coordinate the asset modernization state with the concurrent, conventional life cycle of the existing asset from operational to retirement and decommissioned.

Reference Architecture

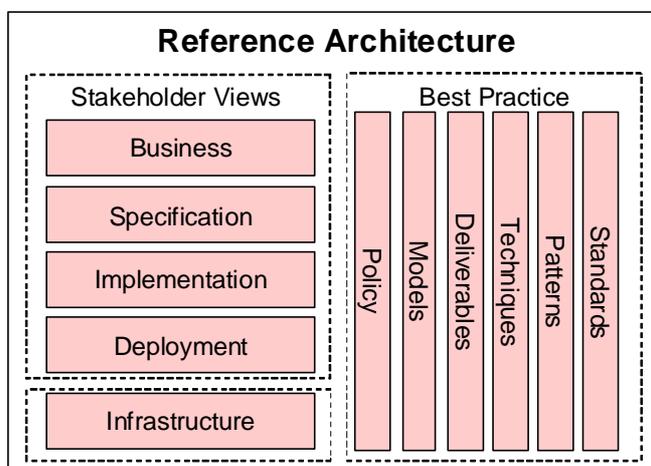


Figure 6 – Reference Architecture

The SAE Reference Architecture⁷ provides a coherent approach to architecture definition which is compatible with other approaches such as TOGAF and Zachman and extends these widely used frameworks specifically for SOA. The challenge for application modernization architecture is to a) integrate existing assets which will probably exhibit various architecture approaches that over time have been compromised by change and b) incorporate an appropriate degree of business context.

Horseshoe Model

In 1999 the SEI published a seminal report titled Options Analysis for Reengineering (OAR): Issues and Conceptual Approach⁸. The report, based on the work of Woods et al⁹, described a conceptual “horseshoe” model that distinguishes levels of reengineering analysis and transformations to the architecture level. The OAR addressed a number of questions:

- What reengineering options/approaches are applicable?
- What do they involve?
- Where and how do architecture considerations come into play?
- Advantages or disadvantages of each approach?
- Interrelationships between the options/approaches?
- Organizational, programmatic, and technical implications?

The Horseshoe model identifies three basic reengineering processes as shown in Figure 7 - analysis of an existing system, logical transformation, and development of new systems.

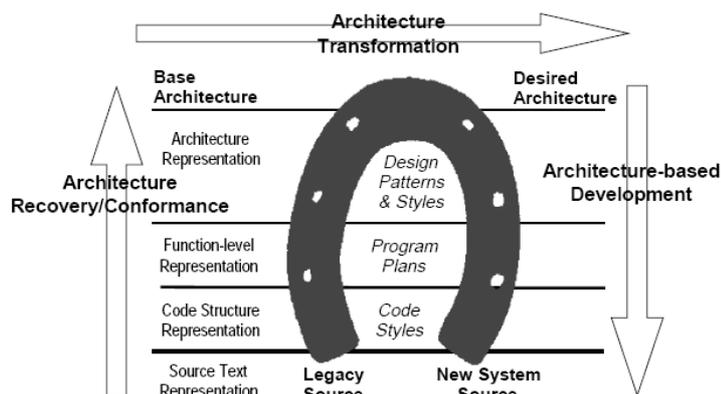


Figure 7 – The Horseshoe Model (Source SEI)

More recently the OMG has developed the Horseshoe model concepts in the Architecture-Driven Modernization Task Force¹⁰ resulting in a defined Architecture Driven Modernization (ADM) approach, summarized in Figure 8, supported by standards for Knowledge Discovery.

Conventionally, modernization projects have focused on transforming the technology. The ADM model identifies three architecture perspectives B, A and T, within either the business or IT domain [Fig 8-1].

The architecture framework identifies transformations that migrate portions of the existing solution to the target solution. These migrations can be business and or IT driven [Fig 8-2].

The ADM Horseshoe model illustrates that transformation (project) will involve journeys that have varying coverage of the three architecture perspectives (B, A, T) depending on requirement. [Fig 8-3].

These models form the basis for the SAE modernization architecture and also process patterns.

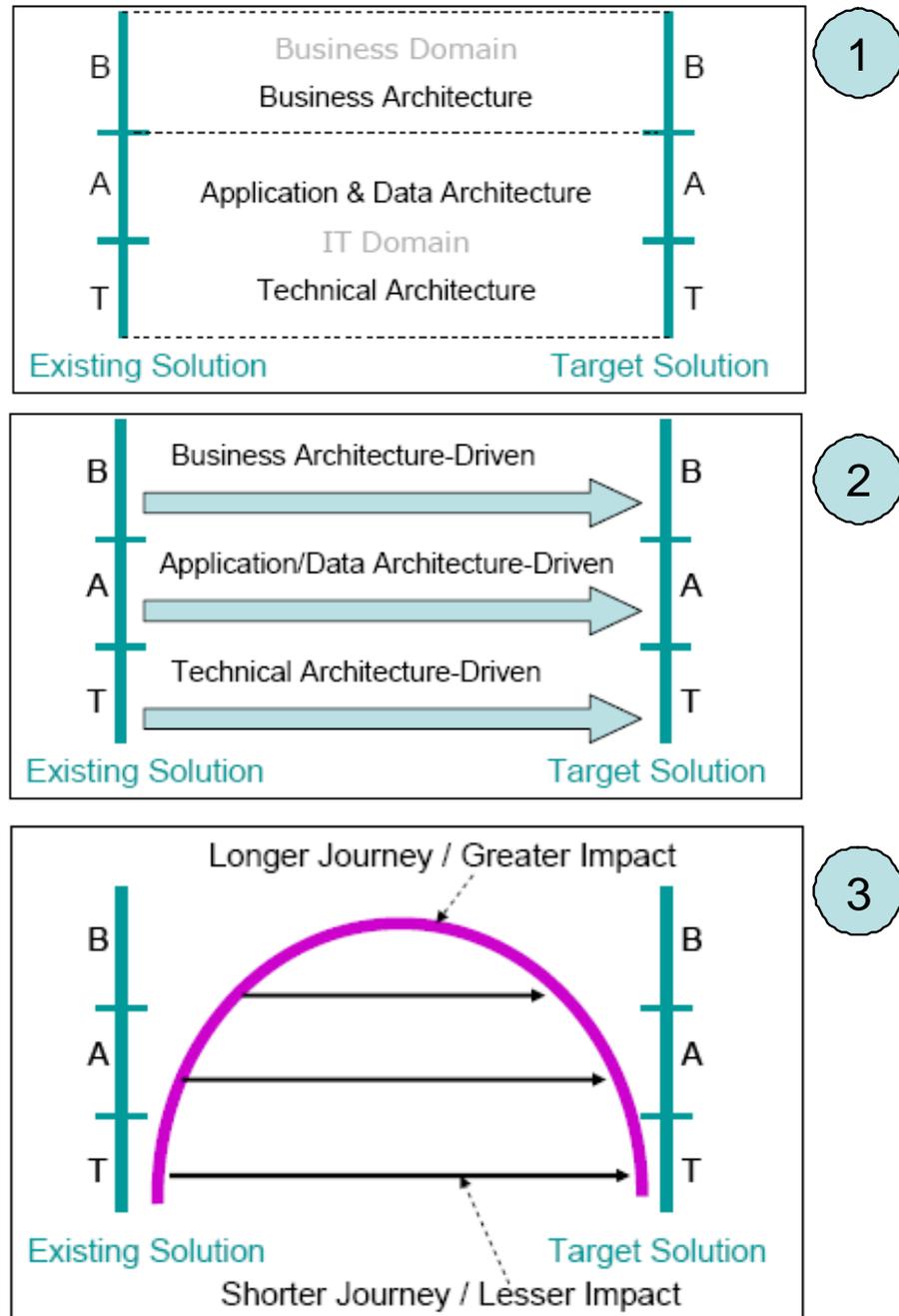


Figure 8 – OMG Architecture-Driven Modernization (ADM)

Detailing Architecture Views

The rich picture of Application Modernization in Figure 1 illustrates the way the As-Is models are used to drive the modernization process to progressively deliver the To-Be architecture. Figure 9 provides an outline of how the As-Is architecture maps to the SAE Views and the SAE To-Be architecture.

Note the Sample As-Is Artifacts are based on the KDM packages, but these will almost certainly be richer in detail and or scope than the existing software assets in order to represent a comprehensive As-Is model.

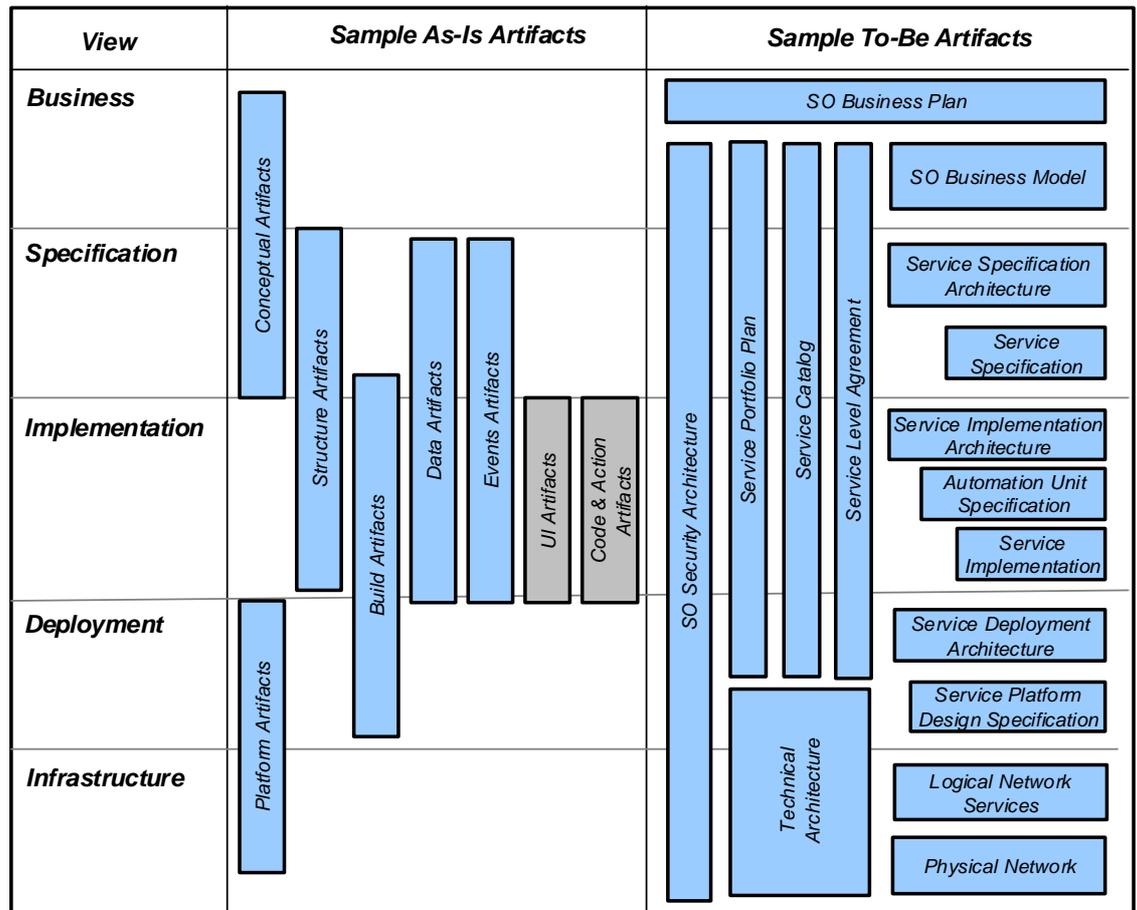


Figure 9 – The Twin Architecture Views of SAE2

The As-Is architecture deliverables and the mapping to the To-Be architecture will be further detailed in subsequent CBDI Journal reports and the Knowledgebase.

A related report¹¹ details how the SAE Process and Patterns guide the transition process between the two views based on the Horseshoe Model.

Reference Process

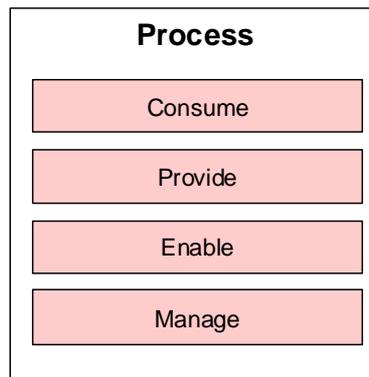


Figure 10 – Reference Process

The SAE Process provides a consistent view of the activities required to plan, architect, enable, deliver and manage SOA based solutions. In considering how the SAE Process will evolve to accommodate modernization we recommend that the basic model that separates consumer and provider as shown in Figure 10 is entirely appropriate because the Target Architecture is going to be intrinsically SOA based.

The top level SAE2 Process Model shown in Figure 11 identifies process disciplines and their primary dependencies.

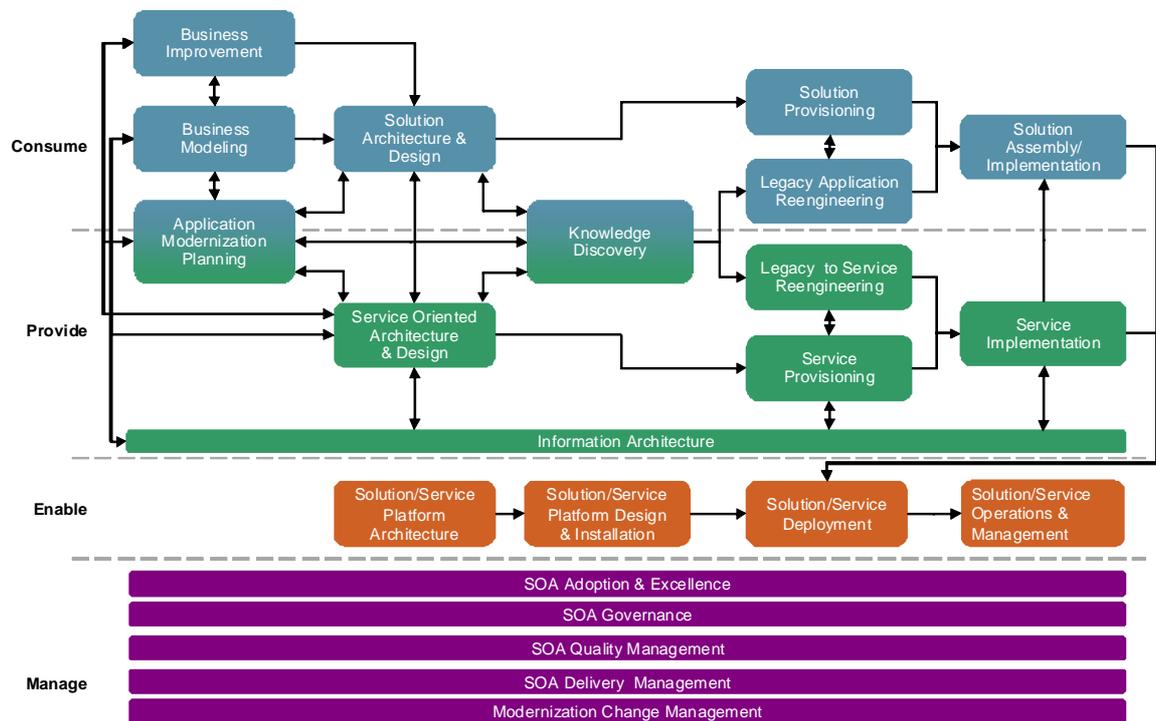


Figure 11 – SAE2 Process Model

Points to note in the model are as follows:

- Several references to SO and or SOA are removed to accommodate the broader scope.

- Application Modernization Planning comprises Assessment and Planning activity in order to give direction to both the Provide and Consume layer processes.
- Knowledge Discovery describes the structural level description of existing assets for use in Application Modernization Planning..
- Legacy Application Reengineering becomes a major part of the Consume track.
- Information Architecture is promoted to a separate discipline because it is a standalone activity separate from SOAD and Solution Modernization Planning.
- A cross cutting management discipline have been added for Modernization Change Management.
- The SOA Governance discipline in particular will require extension to ensure the modernization objectives are achieved.

Further details of the SAE2 Process are included in the complementary CBDI Journal report.

It should also be noted that TOGAF 9 includes some guidance on Migration Planning in the ADM (Architecture Development Method) and SAE2 task structures will be aligned with TOGAF where appropriate.

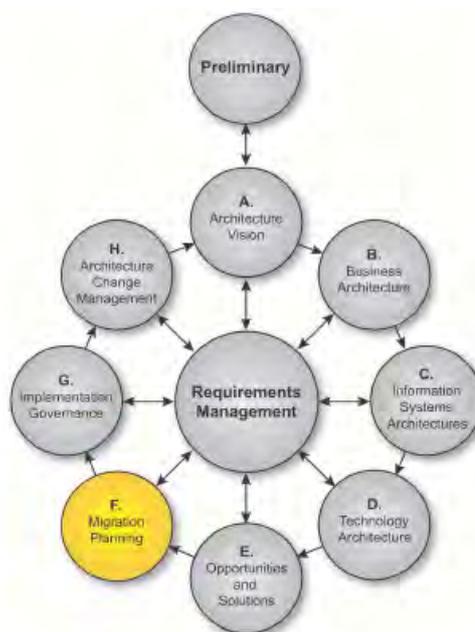


Figure 12 – TOGAF Architecture Development Method (ADM)

Organization

In principle there is little change in the SAE Reference Model for Organization. However there will be some important questions to answer regarding how the organization will need to change in context with modernization? For example:



- What modernization responsibilities are required for each role and general awareness
- In what areas should modernization continue as a technology issue, and where should it be business driven?
- What are the needs for business involvement and education?
- In view of the high priority to maintain the integrity of knowledge during and following knowledge discovery and modernization, what are the roles and responsibilities for knowledge management?
- How to manage and govern outsourced, offshored and SaaS activity to exert much tighter governance over 3rd parties and contracts to ensure modernization objectives are achieved?

Final Remarks

This report has outlined how SAE will evolve to support modernization in SAE2. The Application Modernization framework will be primarily but not exclusively focused on delivering an SOA target architecture. There will always be requirements for conventional architecture coexisting with SOA.

There are good opportunities to align with key standards in this area, although alignment should be initially restricted to conceptual approaches and patterns plus structural meta data, creating options to use more detailed discovery meta data in particular when it is appropriate and there is ROI.

¹ Application Modernization and Migration Trends, Forrester Consulting, November 2009 http://www.ithound.com/v3/view_abstract?id=3318

² In the past, huge market setbacks have been followed, not by periods of calm, but by long stretches of volatility.
Economist, http://www.economist.com/theworldin/displaystory.cfm?story_id=14742674

³ Enterprise Framework for SOA, CBI Journal March 2005

⁴ CBI Service Architecture & Engineering (SAE™)

⁵ SAE Meta Model V3

⁶ OMG Knowledge Discovery Metamodel
<http://www.omg.org/technology/kdm/index.htm>

⁷ The Architecture Component of the SAE™ Reference Framework for SOA

⁸ Options Analysis for Reengineering (OAR): Issues and Conceptual Approach
<http://www.sei.cmu.edu/library/abstracts/reports/99tn014.cfm>

⁹ Woods, S.; Carriere, S.J.; & Kazman, R. "A Semantic Foundation for Architectural Reengineering and Interchange," 391-398.

¹⁰ OMG Architecture-Driven Modernization Task Force
<http://adm.omg.org/>

¹¹ CBI Journal Report – SAE2 Process and Patterns



About CBDI

CBDI Forum is the Everware-CBDI research capability and portal providing independent guidance on best practice in service oriented architecture and application modernization.

Working with F5000 enterprises and governments the CBDI Research Team is progressively developing structured methodology and reference architecture for all aspects of SOA.

CBDI Products

The CBDI Journal is freely available to registered members. Published quarterly, it provides in-depth treatment of key practice issues for all roles and disciplines involved in planning, architecting, managing and delivering business solutions.

Visit www.cbdiforum.com to register.

Platinum subscription – A CBDI Forum subscription provides an enterprise or individual with access to the CBDI-SAE Knowledgebase for SOA delivering ongoing practice research, guidance materials, eLearning, tools, templates and other resources.

Everware-CBDI Services

At Everware-CBDI we enable large enterprises and governments to become more agile by modernizing their business systems. We have repeatable processes, resources, tools and knowledge-based products that enable enterprises to transform their current applications in an efficient, low risk manner, into an optimized service-based solutions portfolio that supports continuous, rapid and low cost evolution. Our consulting services range from providing practices and independent governance to architecture development, solution delivery and service engineering.

Contact

To find out more, and to discuss your requirements visit www.everware-cbdi.com or call

USA and Americas: 703-246-0000 or 888-383-7927 (USA)

Europe, Middle East, Africa, Asia, and Australasia: Telephone: +353 (0)28 38073 (Ireland)

www.everware-cbdi.com

IMPORTANT NOTICE: The information available in CBDI publications and services, irrespective of delivery channel or media is given in good faith and is believed to be reliable. Everware-CBDI Inc. expressly excludes any representation or warranty (express or implied) about the suitability of materials for any particular purpose and excludes to the fullest extent possible any liability in contract, tort or howsoever for implementation of, or reliance upon, the information provided. All trademarks and copyrights are recognized and acknowledged. The CBDI Journal may be distributed internally within customer enterprises that have current corporate subscriptions. Otherwise CBDI Journals may not be copied or distributed without written permission from Everware-CBDI.