



RAD: Really Awful Design - Really?

Rob Day & Eoin Woods

Agile Conference, September 2005



Workshop Organisation

- Session Objectives & Introductions
- RAD Origins – Some Architectural Musings
- Software Architecture - Overview
- Software Architecture in DSDM
- Concept of an Architectural Filter
- Working Groups
- Consolidation & Review



Session Objectives

Purpose of the workshop:

- To consider the role of architectural design in software development projects,
- Building on the experience of an earlier workshop at DSDM Netherlands in 2004.

The prime deliverables will be:-

1. A list of factors to consider in the form of an "Architecture Suitability Filter".
2. A recommendation as to whether or not the Architectural Suitability Filter is a concept worth pursuing as a DSDM White Paper.



Introductions

Rob Day

- Chief Technology Officer
- **mwr**, a publisher of digital interactive learning resources
- DSDM practitioner & trainer, member since 1996

Eoin Woods

- Principal Consultant
- **Zuhlke Engineering**, a technology solutions vendor
- Co-author of “Software Systems Architecture, working with Stakeholders using Viewpoints and Perspectives”

... and yourselves

- name, role, organisation
- Workshop objectives



RAD Origins – Some Architectural Musings

- Introduction of GUI tools & DB tools
- Focus on prototyping new front-end onto existing back-end
- Naturally lead to adoption of 2-tier architecture
- Generally implicit rather than crafted decision-making
- Focus on KISS and document-lite approaches
- DSDM conceived to re-dress the drift away from formality
- DSDM focused on single project



Software Architecture – Overview

- Software Architecture – Definition
- Software Architecture in Context
- Architecture and Requirements
- Quality Properties
- Architecturally Significant
- Context of the Role



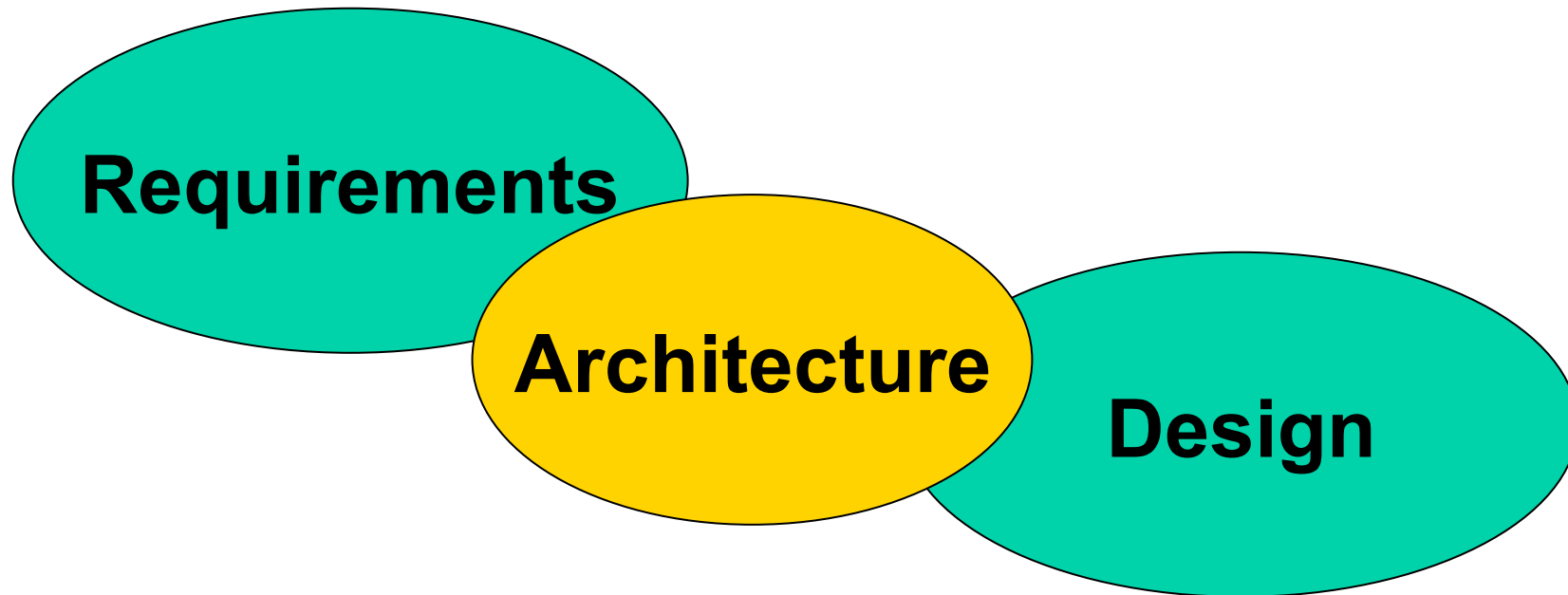
Software Architecture - Definition

*The **software architecture** of a program or computing system is the **structure or structures** of the system, which comprise software **elements**, the externally visible **properties** of those elements, and the **relationships** among them*

Bass, Clements and Kazman (SEI)
Software Architecture in Practice



Software Architecture in Context



The crucial bridge between requirements and design

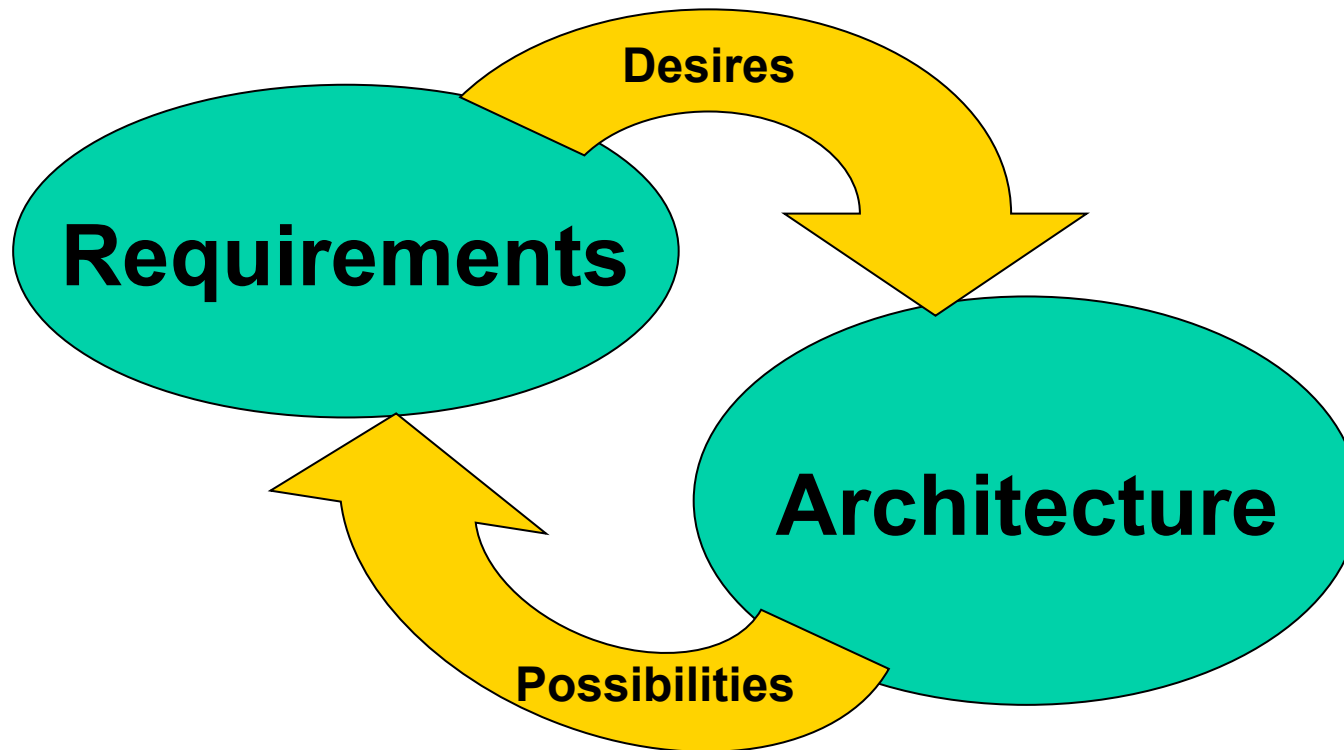


Architecture and Requirements

- Requirements frame the architectural problem
 - Stakeholder needs and desires
- Yet, architecture influences requirements
 - “The art of the possible”
 - Helps stakeholder understanding of risk/cost
 - Helps stakeholder understanding of possibilities



Architecture and Requirements



This interplay is core to the architectural process



Quality Properties

- The non-functional characteristics of the system (“-ilities”)
 - Performance, Security, Maintainability, ...
- Quality properties are crucial to stakeholders
 - Slow functions don’t get used
 - Unavailable systems cause business interruption
 - Security problems cause headlines
 - Unmaintainable systems become irrelevant
 - Yet often overlooked during requirements and design



Quality Properties

- Quality properties are often an afterthought
 - Often expensive to “retro-fit”
 - Disruption to existing operations
 - May conflict with existing qualities
- Achieving quality properties is a key architectural task
 - Understanding the real stakeholder needs
 - Making tradeoffs (e.g. usability vs. security)



Architecturally Significant

- Not all decisions are architectural
 - Make decisions at the right point / level
- Significance generally depends on context
- Architecturally significant decisions
 - Visible by those in other stakeholder groups
 - Have a system wide impact

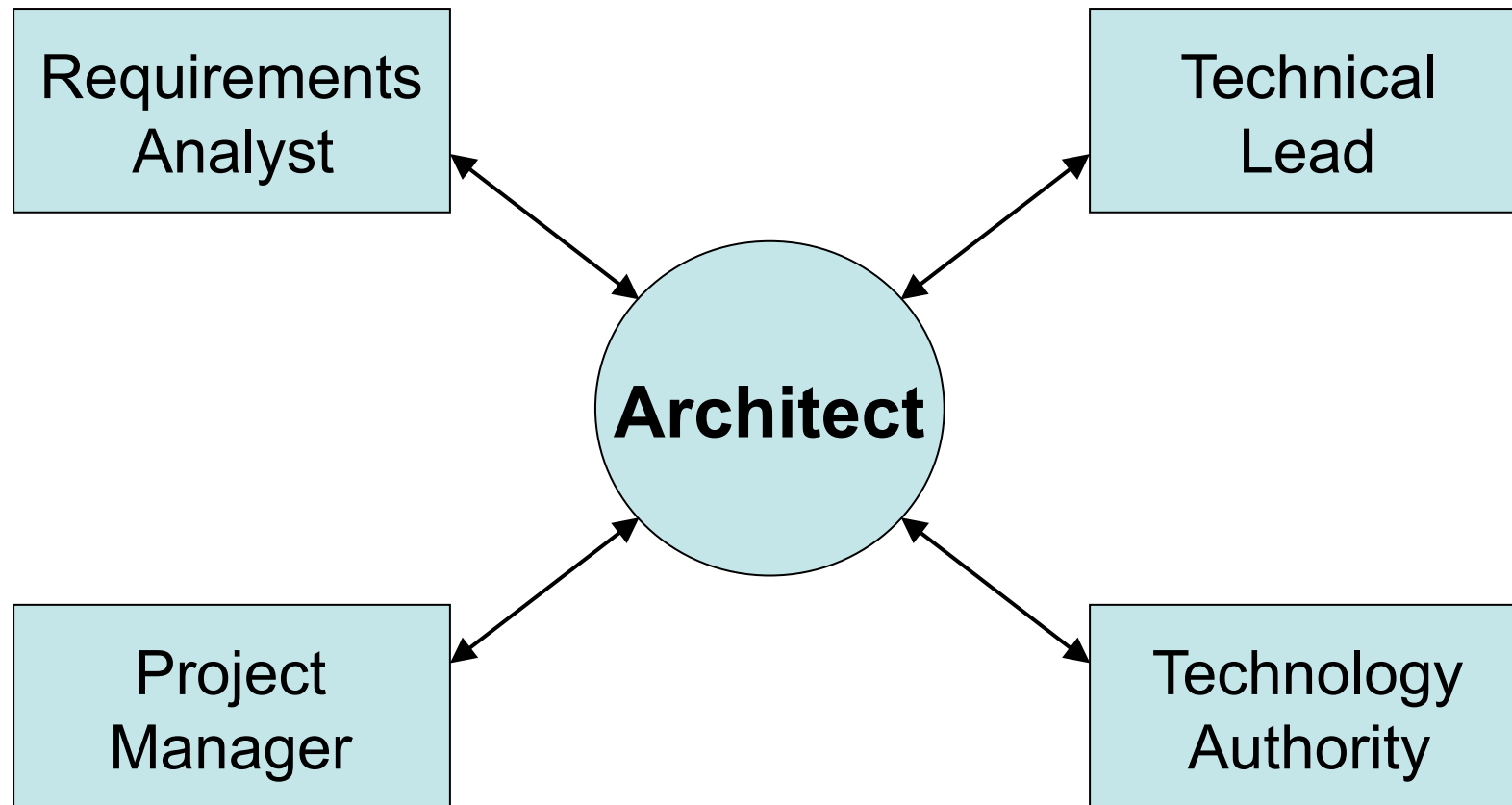


Context of the Role

- Requirements engineer / analyst
 - Provides the architect with requirements and acts as a proxy user
- Project Manager
 - Manages the overall project (including the architect)
- Design Authority / Technical Lead
 - Responsible for internal technical integrity & build
 - Role may well be undertaken by the architect
- Technology Authority
 - Provides specialist knowledge to the architect



Context of the Role



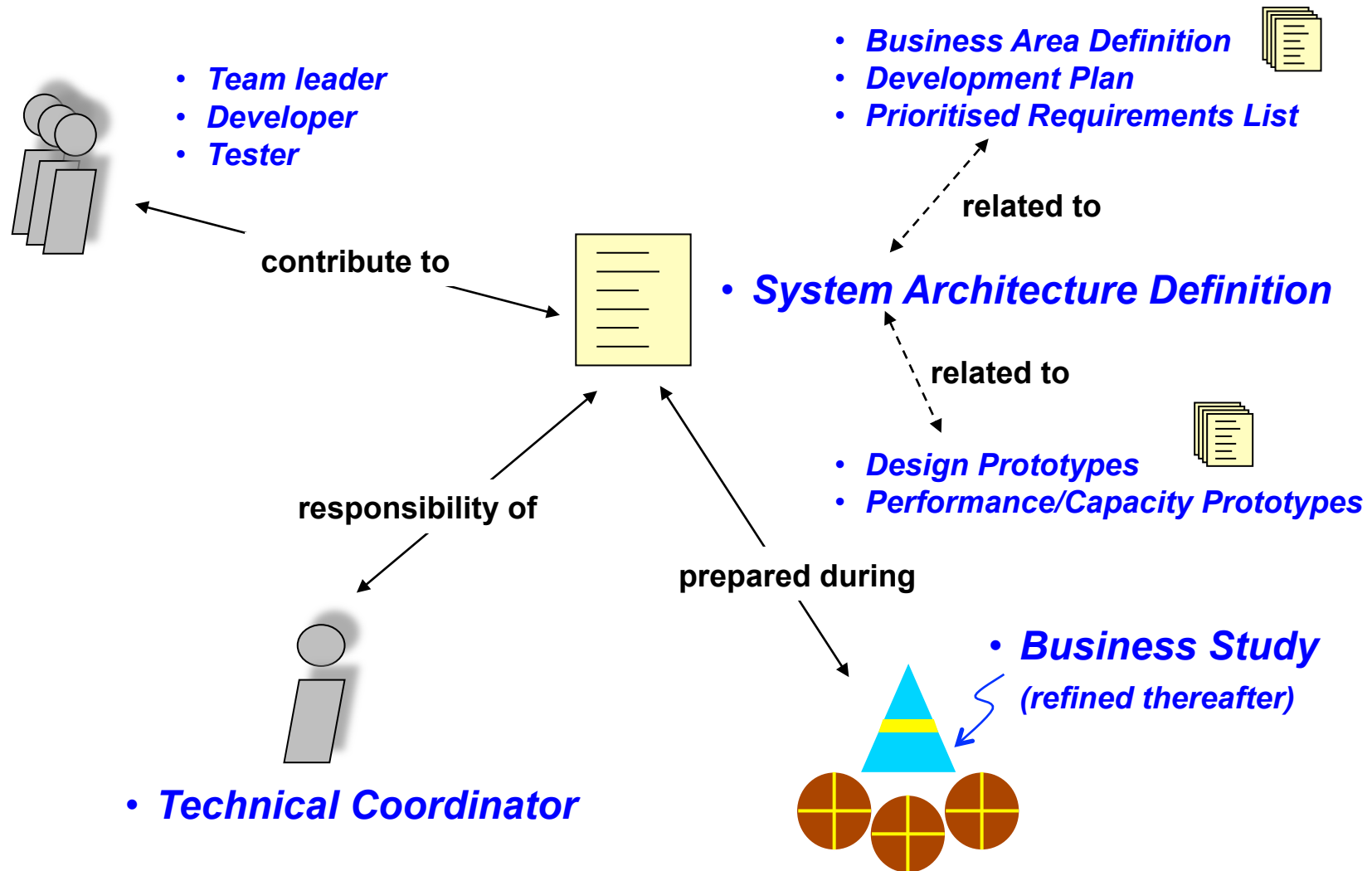


Software Architecture in DSDM

- Products, People and Processes
- SAD Template
- DSDM & TOGAF White Paper
- Architecture in Other Development Approaches



DSDM – Products, People and Processes





SAD Template

DEVELOPMENT ENVIRONMENT

Components

Non functional requirements

Reuse

Hardware

Testing

Database or object model

Developer's day to day

TARGET ENVIRONMENT

Components

Licensing

Network

Hardware

Migration Strategy

Backup and recovery

Operations

SYSTEM ARCHITECTURE

Software architecture

Interfaces

Context Diagram

Reuse

Security

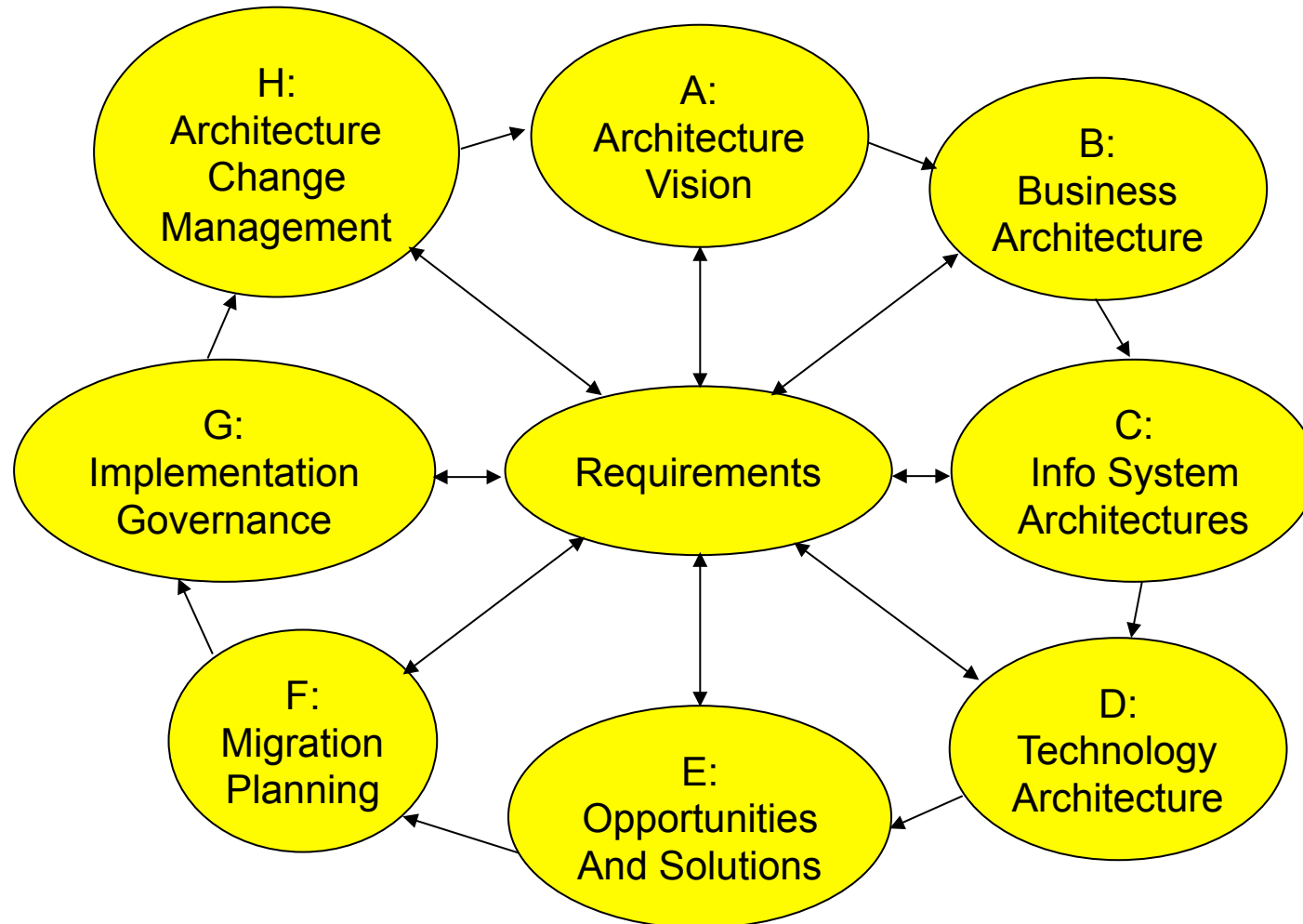


DSDM & TOGAF White Paper

- TOGAF = The Open Group Architecture Framework
- White Paper identifies similarities, differences and how facets of each could be used with the other
- Key points of synergy:
 - TOGAF principles, especially “architectural”
 - TOGAF emphasises “Enterprise Architecture”
 - TOGAF “Business scenarios” complement DSDM facilitated workshops
 - TOGAF has comprehensive material on modelling “Architectural Views”
 - TOGAF details an Architectural Change Management process to establish and support the Enterprise Architecture
 - TOGAF can contribute to completion of DSDM products



TOGAF 8-Phase Lifecycle





Architecture in Other Development Approaches

- Rational Unified Process (RUP)
 - Architectural views (4+1 model)
 - Architectural prototypes
 - Elaboration phase (to de-risk project)
- eXtreme Programming (XP)
 - Design as an ongoing process (with code-test-listen)
 - Write tests first
 - Simplest design that runs the test suite
 - Avoid predicting tomorrow's problems
 - Architecture captured in a “system metaphor”



Concept of an Architecture Filter

- Similar in concept to the DSDM Suitability/Risk List
 - 10 “critical success factors”
 - 7 project situational factors
 - 18 “additional” questions
- How do the DSDM principles relate to architecture?
- What impact do architectural aspects have on a DSDM project?
 - From an organisational perspective
 - From a system perspective
 - From a project perspective
 - Any other perspectives?



Workshop

- Team work:
 - to identify list of factors
- Group work
 - to consolidate, categorise and refine list of factors
- Group work
 - to consider potential for a white paper.