

# *Towards a Re-engineering Method for Web Service Architectures*

Claus Pahl and Ronan Barrett

{cpahl,rbarrett}@computing.dcu.ie.

School of Computing,  
Dublin City University,  
Ireland

# Agenda

- Introduction
- Software Re-engineering
- The Case Study
- Relevant Standards
- Service Description
- Architecture
- Conclusion
- Future Work
- Questions

# Introduction

---

- Re-engineering of software systems
- Apply various techniques
- Web Service Framework (WSF)
- Service Oriented Architectures (SOA)
  - Standardised interaction architecture
  - Service integration layer
  - Flexible integration
  - Share and reuse components & resources
  - Enhanced maintainability
  - WSF is an SOA
- Domain specific approach
  - E-Learning system as case study

# Software Re-engineering - Theory

---

“Software re-engineering is concerned with the reimplementation of legacy software in order to improve it’s maintainability” \*

- Motivated by
  - Requirement change
  - Environment change
- Advantages
  - Reuse
  - Interoperability
  - Maintainability

---

\*Sommerville

# Software Re-engineering - Techniques

---

- Some techniques...
  - Source code translation
  - Structure improvement
  - Data re-engineering
  - Modularisation

# Modularisation

---

“Modularisation is the process of (re)organising a software system such that related parts form coherent components” \*

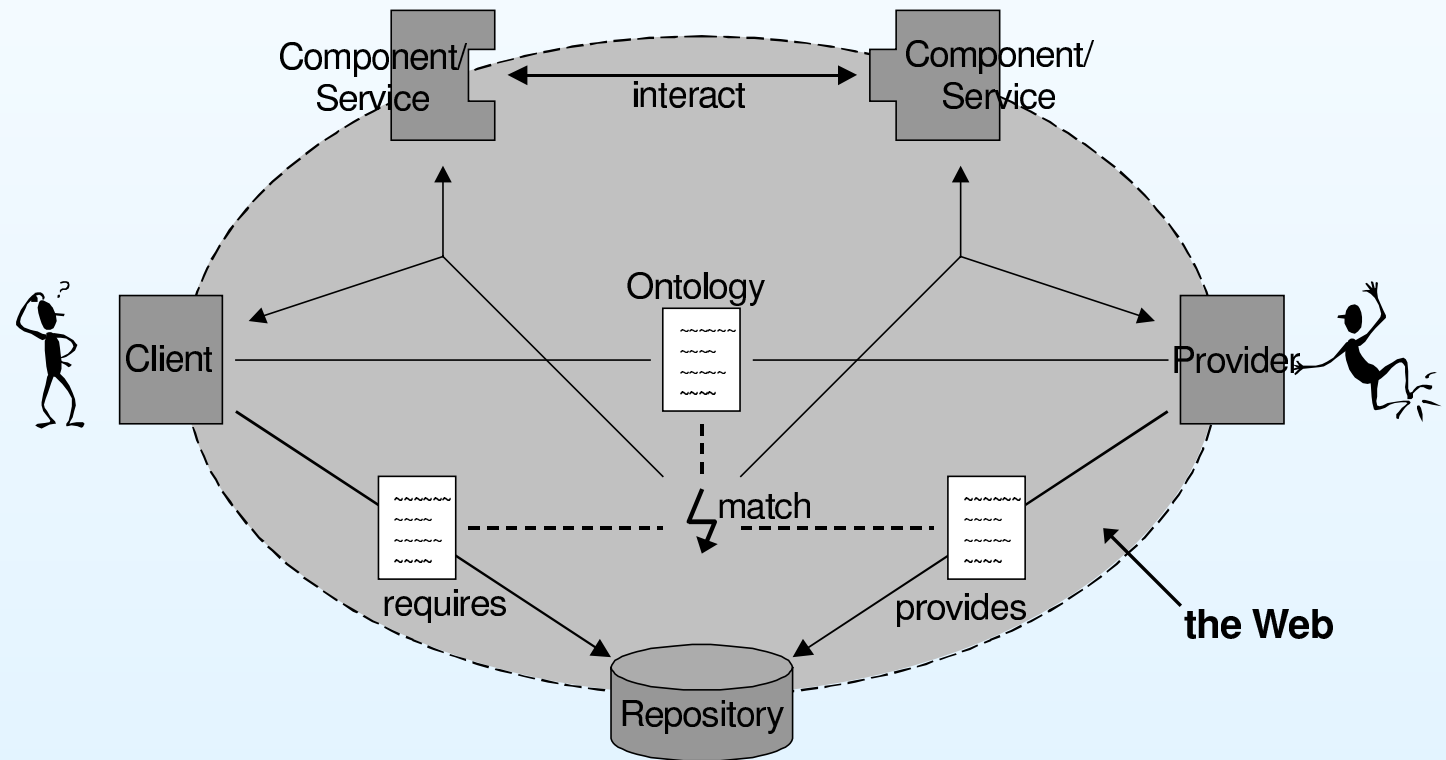
- Development concern
- Deployment concern
- Coherent components
- Architectures
  - Reference architectures - Whole system
  - Architectural patterns - System aspects
- Target is an SOA

---

\*Medvidovic, Rosenblum & Taylor

# WSF - Infrastructure

- 2 infrastructural aspects...
  - Discovery - Development
  - Assembly - Deployment



# The Case Study

---

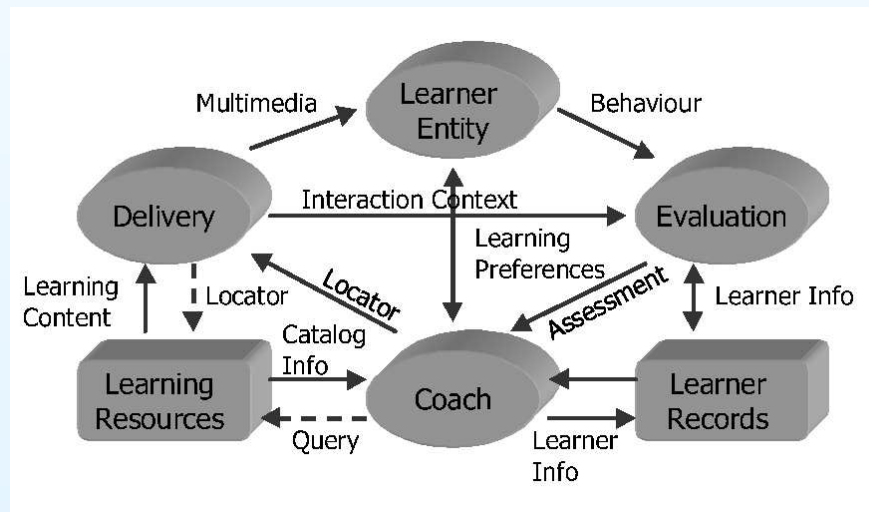
- Teaching & learning Environment (TLE), IDLE \*
- Existing system
  - Features
    - Audio supported lectures
    - Animated tutorials
    - Authentic database development labs
    - Many more.....
  - Components
    - Audio Server
    - Graphical modelling tool
    - Web Server
    - Databases
    - Many more.....

---

\*Pahl, Barrett & Kenny

# Relevant Standards - I

- Adherence to relevant standards
- Reference Architecture
  - Learning Technology Standard Architecture (LTSA) \*
    - Framework for development, evaluation & discovery
    - Models processes, storage & interactions
    - Promotes sharing & reuse



\*IEEE LTSC

## Relevant Standards - II

---

- Domain specific ontology
  - Learning Object Metadata (LOM) \*
    - Metadata framework
    - Basic annotations
    - Learning object as unit of discovery
    - Learning object as unit of assembly

---

\*IEEE LTSC

# Service Description

---

- Expose components as services
- Open market model
- Service identification
- 2 description dimensions
  - Domain-specific descriptions
    - Supports discovery & retrieval
    - Domain specific description
    - LOM
    - Publishable services exposed
  - Infrastructure
    - Supports integration
    - Interface & messaging aspects
    - WSDL
- Semantic Web?

# Architecture - I

---

- Connection of discovered services
- Application of reference architecture
- Architectural Topologies
  - Centralised
  - Peer-to-Peer
- Reference Architectures
  - Influence topologies
  - LTSA exhibits SOA characteristics
  - Components exposed as services
- Some Architectural Transformations
  - Moved IDLE Architecture to LTSA
  - LTSA as SOA
  - Expose LTSA stores & components as services

# Architecture - II

---

- Fine-grained architectural patterns
  - Model View Controller (MVC)
    - Observer
    - Composite
  - 3 tier architecture
  - SOAP toolkits
    - Factory & proxy patterns
  - Navigational contexts \*
- Core patterns in TLE or SOA
  - Factory
  - Proxy
  - Serializer

---

\*Jacyntho, Schwabe & Rossi

# Conclusion

---

- Re-engineering mechanisms identified
  - Modularisation
- Domain specific standards
- Infrastructure
  - Separate Description
    - Domain-specific descriptions
    - Infrastructural
- Architectures
  - Course grained
  - LTSA as an SOA
  - Fine grained
  - Core patterns
  - Secondary patterns

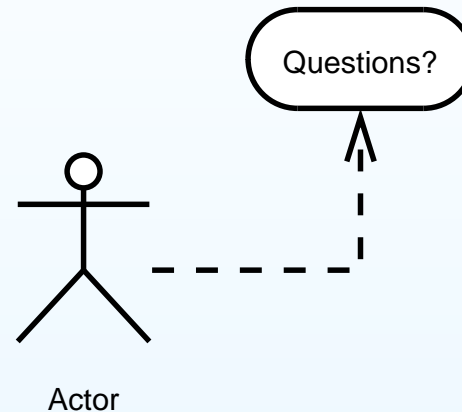
## Future Work

---

- Identify further general patterns/styles
- Identify more domain specific patterns
- Additional separation of concerns mechanisms
- Clear methodology for re-engineering needed

# Questions

Thanks for listening!



Open Source tools used to make this presentation.....



This project is funded by: Irish Research Council for Science, Engineering and  
Technology: funded by the National Development Plan

And

Dublin City University Teaching and Learning Fund, DCU-TLF

