MINT Style Based Architectural Migration: Method and Case Study

Simon Giesecke, Johannes Bornhold
<simon.giesecke@acm.org>
Software Engineering Group
Carl von Ossietzky University
Oldenburg, Germany

WMR 2006, Bari, Italy
2006-03-24
Outline

- MIDARCH Method & Context
  - Research Project
  - Vision
  - Activities
- Case Study
  - Setting
  - Preliminary Results
- Conclusion
Part I

MIDARCH Method & Context
Overall Goal

- Improvement of Integration and Migration Processes through Reuse of Design Knowledge
  - **Application Domain**: Distributed and Web-based Business Information Systems
  - **Architectural Level**: Exploration of Candidate Architectures based on Different Middleware Platforms
  - **Means**: Binding Design Knowledge to Architectural Styles
Architectural Styles

- Specify constraints on
  - Component types
  - Connector types
  - Composition rules

- Generic styles
  - pipe-and-filter
  - event-based

- Specialised styles: endorsed by an implementation platform
  - Middleware INTEGRation Styles (MINT Styles)
Middleware to Styles

Software Systems

Middleware Platform

Style Description

Concrete MINT Styles

Style Synthesis

Abstract & Concrete MINT Styles

Middleware Platform

WebSphere MQ : Platform

eGate : Platform

J2EE Platform Specification : Platform

Tomcat : Platform

Sun Application Server : Platform

Cocoon : Platform

Style Description

WebSphere MQ : Style Description

eGate : Style Description

J2EE : Style Description

Tomcat : Style Description

Sun Application Server : Style Description

Cocoon : Style Description

Cocoon Var. A : Style Description

Cocoon Var. B : Style Description

Message-based Style Description

Pipe-and-filter Style Description

Generic : Style Description

J2EE / Sun AS : Style Description

Tomcat : Style Description

Cocoon Var. A : Style Description

Cocoon Var. B : Style Description

Giesecke/Bornhold: Migrating Regional Trade Information System, 2006-03-24
Research Project Overview

- WP1: Analysing Architectural Styles and their Usage
- WP2: Extending ANSI/IEEE Standard 1471
- WP3: Modeling MINT Styles using ADLs
- WP4: Developing a MINT Style Based Architectural Integration Method
- WP5: Providing Tool Support for the MIDARCH Method

**Tools and Case Studies:**
- Tool Support: Pattern Repository with Java & UML Binding
- MINT Style Hierarchy
- MIDARCH (MIDdleware style based ARCHitectural integration) method description
- IEEE 1471 based Reference Model for Style-based Architectural Description
- Case Study 1: German EHC
- Case Study 2: regio
- Case Study 3: EWEtel
- Tool Support: MINT Style Repository
Method Overview

Activity 1: Definition
- Define Scope
- Determine Current and Future Requirements

Activity 2: Preparation
- Develop Project-Specific Quality Model
- Model Current Architecture

Activity 3: Exploration
- Choose/Model MINT Styles
- Model Candidate Architectures
- Evaluate Candidate Architectures

Activity 4: Implementation
- Choose Target Architecture
- Adopt Target Architecture
- Assess Evaluation Results
Part II

Case Study
Case Study 2

- Role in Research Project
  - Apply Preliminary MIDARCH Method
  - Evaluate Feasibility of General Idea
  - Explore Refinements and Issues for MIDARCH Activities

- Current State
  - Ongoing
  - Activities 1 and 2 have been partially completed

- Regional Trade Information System
  - Local authorities <-> Local Companies
Activity 1: Definition

Scope

- Currently independent subsystems
  - Query Interface (web-based)
  - Management Interface (web-based)
  - Legacy Management and Analysis Interface (Java Swing)
- Two separate relational databases with distinct schemas; proprietary file-based database
- Target: Coherent, more integrated system architecture
Activity 1: Definition (2)

- **Customer Goals**
  - **Improve Evolvability** (Architectural Quality)
    - Multiple Customer Support
    - Integration with other Information Systems
  - **Improve Availability** (System Quality)
  - **Improve Maintainability** (System Quality)
    - Enactment of Architectural Changes in the System Implementation
Activity 2: Preparation

Software Systems

Management Interface

Desktop Application

Query Interface

Web Client

Database

Query Database

Export

<<flow>>

<<external>>

<<flow>>

<<flow>>

<<flow>>
Activity 3: Exploration

Archimate

Business Layer
- Business Processes

Application Layer
- Applications

Infrastructure or Technology Layer
- Middleware
  - Operating System, Virtual Machine
  - Hardware Resources

xADL
- Applications
  - MINT Style

Overview Level

Detailed Level
Problem Areas

- Lack of Coherence
  - Violations of Cocoon Style
- Variations between Query and Management Intf.
- Proprietary Page Description Language
- Tight Coupling
  - Access to underlying database hard-coded
  - Within Data Tier of both Subsystems
- Code Clones
  - Parts of Query Interface copied to Management Interface
  - Independent evolution, manual synchronisation
Cocoon Style

Generic Cocoon Style

Component Type Hierarchy

Component
  └─ Filter
    └─ Generator
    └─ XSLT Processor
    └─ Data Source Query
    └─ Serialiser
      └─ HTML Formatter

Request → Generator → Transformer 1 → Filter → ... → Transformer n → Serialiser → Response
Cocoon Style (2)

Generic Cocoon Style

Request -> : Generator -> Transformer 1: Filter -> Transformer n: Filter -> : Serialiser -> Response

Var. A Cocoon Style

HTTP Request -> : XSP Engine -> : Data Source Query -> : XSL Processor -> : HTML Formatter -> HTML Response

: XSP Document

: Data Source

: XSL Stylesheet
MIDARCH Method for Supporting Integration/Migration Projects

Case Study: Web-based Regional Trade Information System

Next Steps
- Define Cocoon Style Descriptions in xADL
- Define Target Architectures Based on these Style Description

Questions?