Structure

- Concept of Interoperability
- Methodology: Guiding Principles
- Existing Frameworks
- Theoretical Approaches
  - Information Integration & Dissemination
  - Remoting: Service Orientation
  - Presentation
- Concluding Remarks
Interoperability

is a property that emerges, when

distinctive information systems (subsystems)

cooperatively exchange data

in such a way that

they facilitate the

successful accomplishment

of an overarching task.
Concept of Interoperability

(modified from Kosanke, 2005; IEC, 2005)
Methodology: Guiding Principles

- Layering
- Viewpoints
- Quality Attributes
- Patterns

GP 1: Layering

- A layer: dissect hierarchical system into a stack of dependant layers (specifying services and service interfaces)
- Two types relevant:
  - Responsibility-based layering (functional)
  - Reuse-based layering (system characteristics, organisations)
GP 2: Viewpoints

- Build set of partial models for single participants: isolated perspectives
- Hide complexity, explore hidden relationships
- Various Types:
  - Domains (Physical, Organisational, ...), Classes of Stakeholders (Analyst, Designer, ...), Roles of Users (Learner, Tutor, ...), ...
  - Informal, semi-formal, formal
- Many views make a viewpoint.

GP 3: Quality Attributes

- ... are non-functional requirements
- ... can be conflicting among each other (therefore: NFR framework!)
- System-specific Attributes:
  - Run-time: Performance, security, availability, usability
  - Non run-time: modifiability, portability, reusability, integrateability, testability
- Domain-specific Attributes
  - Take from learning theory, collaboration research (esp. CSCW & CMCW)
  - e.g. awareness, adaptability, seamlessness, collaborative filtering, dynamism
GP 4: Patterns

= recurring problem + solution + forces + …

- Pattern language:
  aligned patterns serving an overarching goal

- Relevant pattern traditions:
  ▪ Pedagogical patterns (e.g. ‘invisible teacher’)
  ▪ Patterns of Cooperative Interaction
    (e.g. ‘overlapping responsibilities’)
  ▪ Architectural Patterns
  ▪ Integration Patterns
    (e.g. remoting patterns like ‘broker’)

- Current shortcoming: learner-centred patterns
Existing Frameworks (I)

<table>
<thead>
<tr>
<th>Layering</th>
<th>Viewpoints</th>
<th>Quality Attributes</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELF</td>
<td>Single (Developer, semi-formal)</td>
<td>Functional / opaque</td>
<td>Security, portability, mobility</td>
</tr>
<tr>
<td>CORDRA</td>
<td>Community based re-usability</td>
<td>Community based re-usability</td>
<td>Security, portability, mobility</td>
</tr>
<tr>
<td>POOL/ECL</td>
<td>Single (Developer, semi-formal)</td>
<td>Functional / opaque</td>
<td>Security, portability, mobility</td>
</tr>
<tr>
<td>IMS TI</td>
<td>+ organisation</td>
<td>Scalability, security, optional multi-linguality</td>
<td>Accessibility, integrateability, evolveability, scalability, high availability, security, optional multi-linguality, investment protection, cost-effectiveness, enterprise readiness</td>
</tr>
</tbody>
</table>

GP 1: Layering
- Widely adopted
- Mainly functional layering
- CORDRA: community based re-usability
- LORI/IMS-TI: integration of legacy systems

GP 2: Viewpoints
- Small number of viewpoints (usually: developers)
- IBM LI/SIS: + organisation
- IMS-TI: + teacher + learner

Existing Frameworks (II)
Existing Frameworks (III)

- **GP 2: Roles**
  - Mainly: application services or information integration
  - IMS-TI / LearnServe: presentation integration

- **GP 3: Quality Attributes**
  - Only generic attributes,
  - Domain-specific attributes are missing

- **GP 4: Patterns**
  - No considerations in the frameworks
  - No patterns for distributed systems
Theoretical Approaches

- Three big streams:
  - Information Integration and Dissemination
  - Remoting: Service Orientation
  - Presentation Integration

Information Integration & Dissemination

- Basic Scenarios
  - Mapping-based approaches
  - Intermediary-based approaches
  - Query-based approaches

- Scope of Integration
  - Persistent Integration (through (partial) replication)
  - Virtual Integration (on demand)

- Composition
  - Data Integration (heterogeneous)
  - Data Exchange (idempotent systems)
I.I. & D.

- Information Retrieval / Querying
  - (Ad hoc) queries are matched against data
  - e.g. query – results
- Information Filtering
  - Data streams matched against standing queries
  - e.g. publish-subscribe
  - Stateful or stateless

Remoting: Service Orientation

- Idea: dissolve functionality from application, provide standard interface
- e.g. web-services: are “a software system designed to support interoperable machine-to-machine interaction over a network” (w3c, 2004)
- Choreography and Orchestration: (BPEL, BPEL4people, SLeD)
Presentation Integration

- Portlets:
  - Encapsulated applications
  - Assembled into pages by a container
  - Return renderable mark-up (e.g. XHTML)
- ... can be made interoperable
  via WSRP
- Java Servlets
- JSP (extend Servlets)
Concluding Remarks

- Guide the design process:
  - Quality attributes
    - (from social-constructivist learning theory)
    - (through usage and interaction analysis)
    - (from documented pedagogical patterns)
  - Patterns
- Draw upon prior experiences
  - Viewpoints: learners, facilitators, organisations, developers
  - Layers: have to be specified
- Integration Architecture: pursue information querying and information filtering, add bridging elements

#eof.

beware, the end is near.