Towards a Taxonomy of Software Connectors

Nikunj Mehta

with
Neno Medvidović

Whither Software Architectures?

• The *chronic* software crisis
• Essential difficulties of software
  – Complexity, conformity, changeability and invisibility
• Complex systems
  – Prefabricated components
  – Heterogeneous infrastructures
  – Product families
  – Moving targets
  – Bloated sizes

2/8/2000  USC - CSF Technology Week
### Architectural Elements

<table>
<thead>
<tr>
<th>Components</th>
<th>Connectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory and Computation</td>
<td>Interactions and protocols</td>
</tr>
<tr>
<td>Domain dependent</td>
<td>Domain independent</td>
</tr>
<tr>
<td>Realize functional requirements</td>
<td>Meet extra-functional properties</td>
</tr>
<tr>
<td>Fairly well understood</td>
<td>Mostly not well understood</td>
</tr>
<tr>
<td>Original inhabitants</td>
<td>Second-class citizens</td>
</tr>
<tr>
<td>Bugs</td>
<td>System dynamism</td>
</tr>
<tr>
<td>Logic optimization</td>
<td>Middleware</td>
</tr>
</tbody>
</table>

### Software Connectors

- Role of connectors
  - Mediate interactions among components
  - Provide auxiliary mechanisms for interaction
- Not all connectors are simple
  - Complex software involves complex interactions
- Simple connectors
  - Procedure calls, module dependencies, pipes
- Complex connectors
  - DNS, remote procedure calls, repository access, semaphores
A Disconnected Understanding

- Connector modeling
  - ADL constructs, graphical notations, module dependencies
- Large scale development does not adequately address issues of interaction
- Herding mentality – one size fits all!
- Focus on connectors
  - Component logic is essentially frozen early
  - Connectors evolve to improve levels of service

Understanding Connectors Better

- What constitutes a connector?
- Study and classify interaction mechanisms
  - A classification framework required
  - Atomic building blocks to be identified
- Effects
  - Composing new connectors by combining existing mechanisms
  - Improve coupling properties, better predict system properties
Classification Framework

- Atomic elements
  - Ducts, data transfer and control transfer
- Service categories
  - Communication
  - Coordination
  - Conversion
  - Facilitation
- Connector types, dimensions
  - Finitely many values

2/8/2000 USC - CSE Technology Week
Applications of the Taxonomy

- Architecture is not just a dependency graph
- Linux architecture – Bowman et al
  - A fully connected graph of components
  - Dependencies as connectors
- Rich connectors simplify architecture representation and improve understanding
- Connectors have internal architecture and contain simpler components and connectors internally

Future Work

- The taxonomy is comprehensive – not complete
- Proposed USC DASADA project
  - Evolution of connector dimensions and values
  - Composition of arbitrarily complex connectors
  - Architectural gauges and connector instrumentation
  - Dynamic architecture simulation tools
  - Enhancing the C2 infrastructure for experimentation
  - OTS middleware for realizing rich connectors
References

- Towards a Taxonomy of Software Connectors
  - Mehta, Medvidovic and Phadke
  - CSE Technical Report USC-CSE-99-529