Accord: Autonomic Components, Composition and Coordination

• Accord is an autonomic component framework.
  – Autonomic component
  – dynamic composition
  – Deductive rule engine

Presentation Objectives

• How to construct autonomic component
• Prototype implementation
Constructing an Autonomic Component

<table>
<thead>
<tr>
<th>Function</th>
<th>Aspect</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>Aspect</td>
<td>Description</td>
</tr>
<tr>
<td>Operation</td>
<td>Aspect</td>
<td>Description</td>
</tr>
</tbody>
</table>

weave

Agents

Optimizing agent  Security agent  Healing agent  Context agent  Rule agent

Agent data  rule knowledge  agent status

Personality  rule  knowledge  agent

Specified by authorized users or other agents
Generated by AutoMate programming toolkit

Prototype implementation -- DIOS++

Computational node

Autonomic component

Gateway

Rule engine

Computational object

Rule operations

Computational node

Rules

Policies

Sensors

Actuators
Example Scenario

A simple application consists of 2 objects: "List" is to generate a list of random integers; "Sorter" is to provide functions of sequential sorting and quick sorting.

`Example Scenario Contd.

Rule1: IF List.getLength < 100 THEN List.getList ELSE List.getLength
Rule2: IF List.getLength < 50 THEN Sorter.sequentialSort ELSE Sorter.quickSort

Rule Conflict Resolution

Disable the conflicting rules with lower priority.

"List" getLength unlocked
setMinValue unlocked locked unlocked
Experiment Evaluation

The following figures present experimental evaluation of DIOS++ library using the IPARS reservoir simulator framework on a 32 node Beowulf cluster.

Fig 1. Runtime overhead introduced in the DIOS++ minimal rule execution mode.

Fig 2. Comparison of computation and rule deployment time.

Fig 3. Comparison of computation, object rule execution and application rule execution time.

Next: Autonomic Composition

Thank you!