Component Management in OPRoS

Seungwoog Jung (swjung@etri.re.kr)
Electronics and Communications Research Institute
KOREA
Contents

- Introduction of OPRoS
- OPRoS Component Model
- OPRoS Component Execution Engine
- Conclusion
OPRoS: Open Platform for Robotic Services

Robot software platform that provides component-based development and execution environments for robot software.

providing the robot software component model, component execution engine, various middleware services, development tools, and simulation environment.
Introduction of OPRoS Component

Role of Component Execution Engine

Node 1

Component A

Component B

Component C

classical control loop

Component Execution Engine (Lifecycle, QoS, Fault Recovery, Coordination, State Management ...)

Communication Middleware

OS

Driver

Node 2

Component D

Component E

Component F

Component New

High level control

Component Execution Engine (Lifecycle, QoS, Fault Recovery, Coordination, State Management ...)

Communication Middleware

RTOS

Driver

RTOS
Component Development Process

Component Editor

Robot Simulator

Simulation & debugging

Component Composer

Composition

Component Repository

Editing

atomic component store

Deploying

Component Execution Engine

Simulating & debugging

Target robots

Operation

robot application

Application Package

Simulation & debugging

Composing

packaging

Exploring

ETRI
OPRoS Component Model

**Component Class Diagram**

- **Property Interface**
- **Lifecycle Interface**
- **Port Interface**

**Component**
- onInitialize/onReset
- onStart/onStop
- onError/onRecover
- onExecute/onUpdated
- onEvent
- onPeriodChanged
- onDestroy

**User Component**
- user defined methods

**Provided Ports**
- Input Ports
- Output Ports

**Required Ports**
- Input Ports
- Output Ports

**Component Profile (XML)**

**Port Library**
- Data Port
- Service Port
- Event Port
- Monitoring Port
- Control Port

User defined Component
OPRoS Component Model

Service Port

- Interface for method invocation of component

Component A

- Service Port A1
- Service Port A2

Component B

- Service Port B1

Component C

- Service Port C1
- Service Port C2
- Service Port C3

Component D

- Service Port D1

- Required Ports: A1, A2, B1, C3
- Provided Ports: C1, C2, D1
OPRoS Component Model

Data Port

- Port for transferring data to components

Component A
- DataPort A1
- DataPort A2

Component B
- DataPort B1

Component C
- DataPort C1
- DataPort C2

Data Profile (XML)

Output Ports: A1, A2, B1
Input Ports: C1, C2
OPRoS Component Model

**Data Port**
- asynchronous processing
- data processing after queuing

<execution_semantics>
  <priority>5</priority>
  <type>periodic</type>
  <period>100</period>
</execution_semantics>

<Component Profile>

Execution

Component Execution Engine
OPRoS Component Model

**Event Port**

- Port for transferring event to components
- Immediate processing of data without queuing
Various OS support
- Windows, Linux
- Android, iOS, Mac

OPRoS

Windows
- Kernel 2.6
- RTAI/Xenomai

Linux

Android

iOS

Mac
- Linux on ARM (HBE-SM II-P320)
Open source (www.ropros.org)
source code, documents, components
Thank you!