ORiN: Standard Network Interface for Robot/Factory Automation

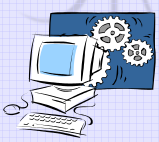
Prof. Makoto Mizukawa, Ph. D
Co-Chair, ORiN Forum
Graduate School of Engineering Management/
Faculty of Engineering, Dept. Electrical Eng.
Shibaura Institute of Technology

Background -in Factory-

◆Various consumer needs
→From mass to customized production

◆Serious competition (speed & cost)
→No loss, No stock, Short TAT

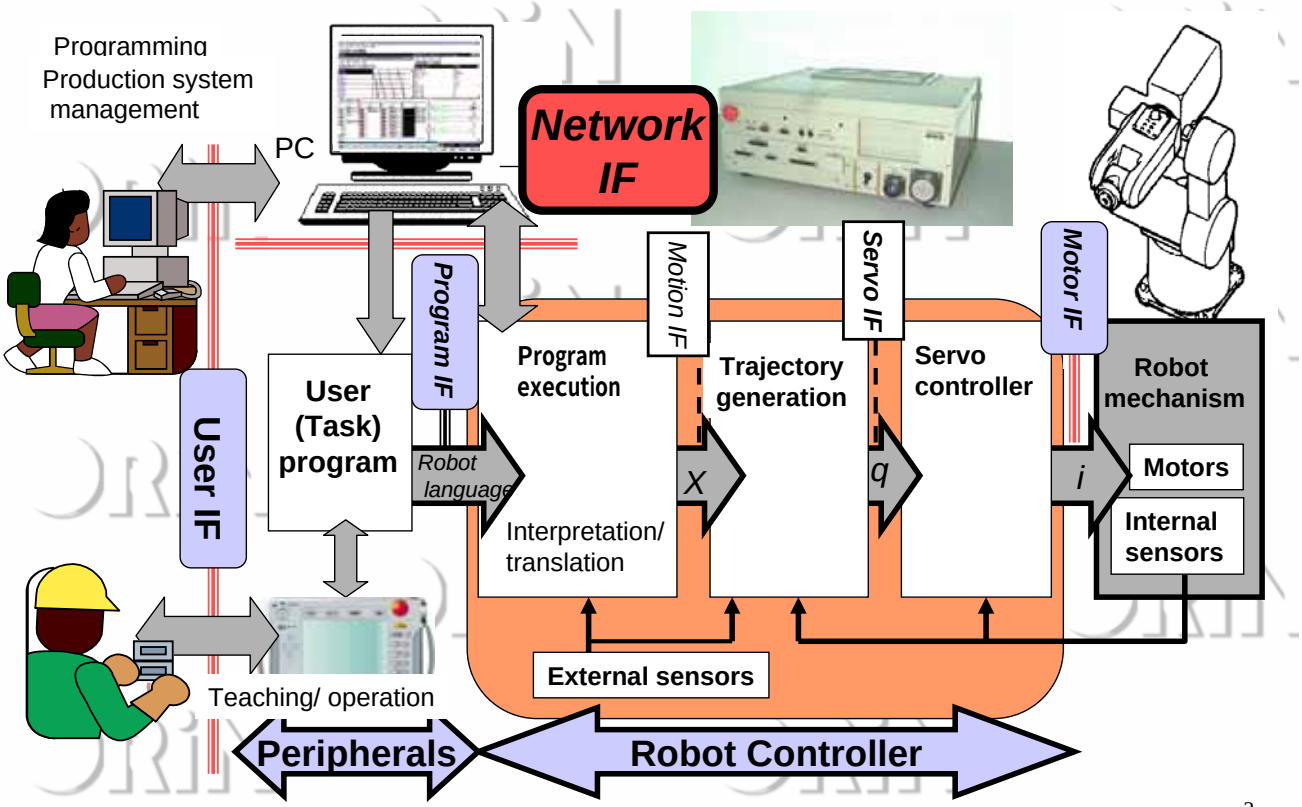
◆IT into Factory (Networking & PC)
→Vertical integration (SCM/ERP)



Continuous monitoring and improvement

Quick start-up / No Loss Operation

Candidates for Open Interface



the standardization of the network interface

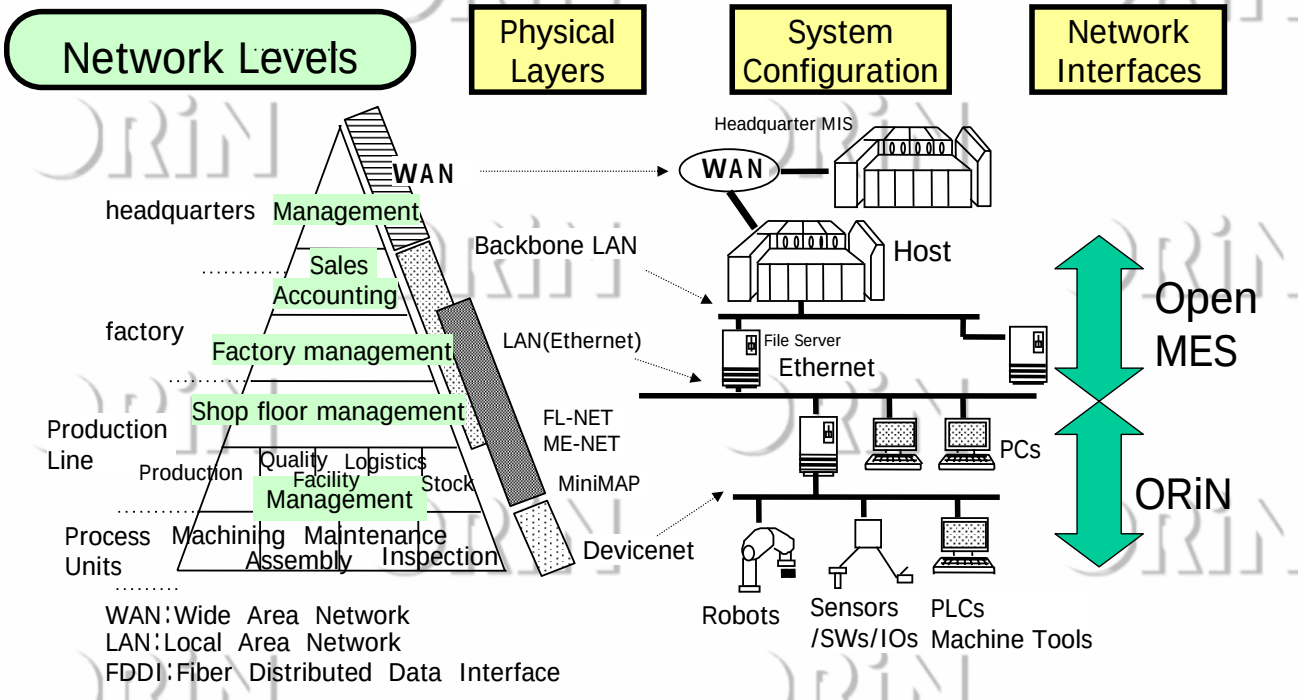
category of IF	examples	user needs	technical problems
mechanical	motor/ encoder IF	△	△
system software	servo IF, motion IF,etc	X	X
robot language	programming language	◎	X
user	operation IF, teaching box IF	○	△
peripheral	network IF (PC IF)	◎	○

Classification of communications related to machine controllers

communication level	usage	releted technology
b/w computers	PCs and machine controllers, controller monitoring, task management	Ethernet (FL-net)
b/w controllers	data exchange b/w machine controllers, real-time I/O data acquisition	FL-net ControlNet ME-NET
b/w devices	data exchange b/w machine controllers and peripheral devices, SWs and displays control	DeviceNet, Interbus-S, Profibus-DP
b/w motor drives	to control motor drives	SERCOS

•Open interface (API) for PCs
•to provide standard robot-data-access method to shop-floor controllers
•to ensure that shop-floor systems have high cost-performance and easy maintainability.

FA Network and Layers



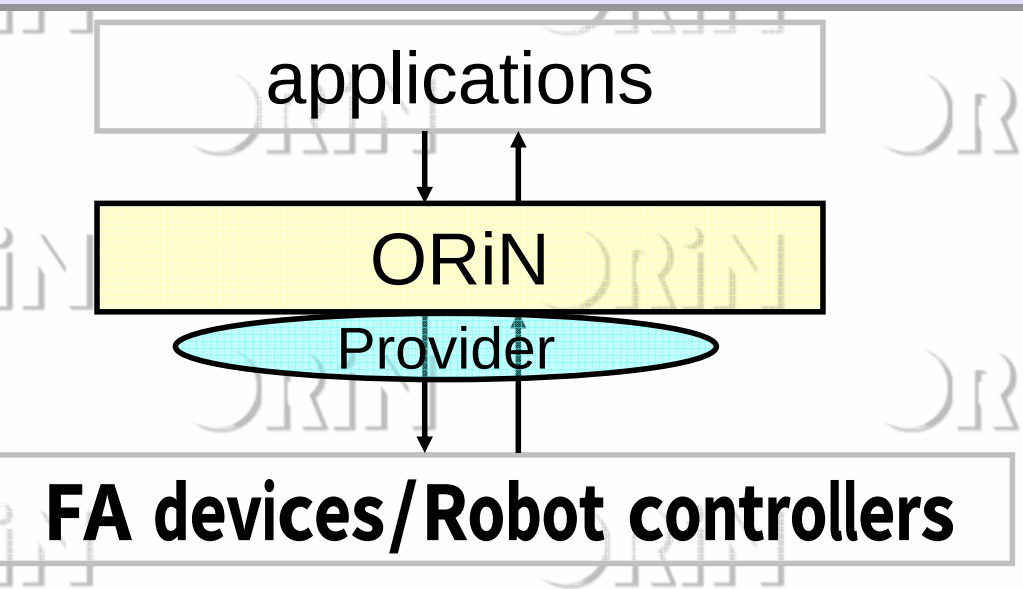
Objective of ORiN

to provide standard methods for accessing FA/robot system information when developing a multi-vendor FA system

- ***a unified access method***
- ***a common application platform [PC and OS]***
- ***a common API***
- ***a unified user interface***

ORiN

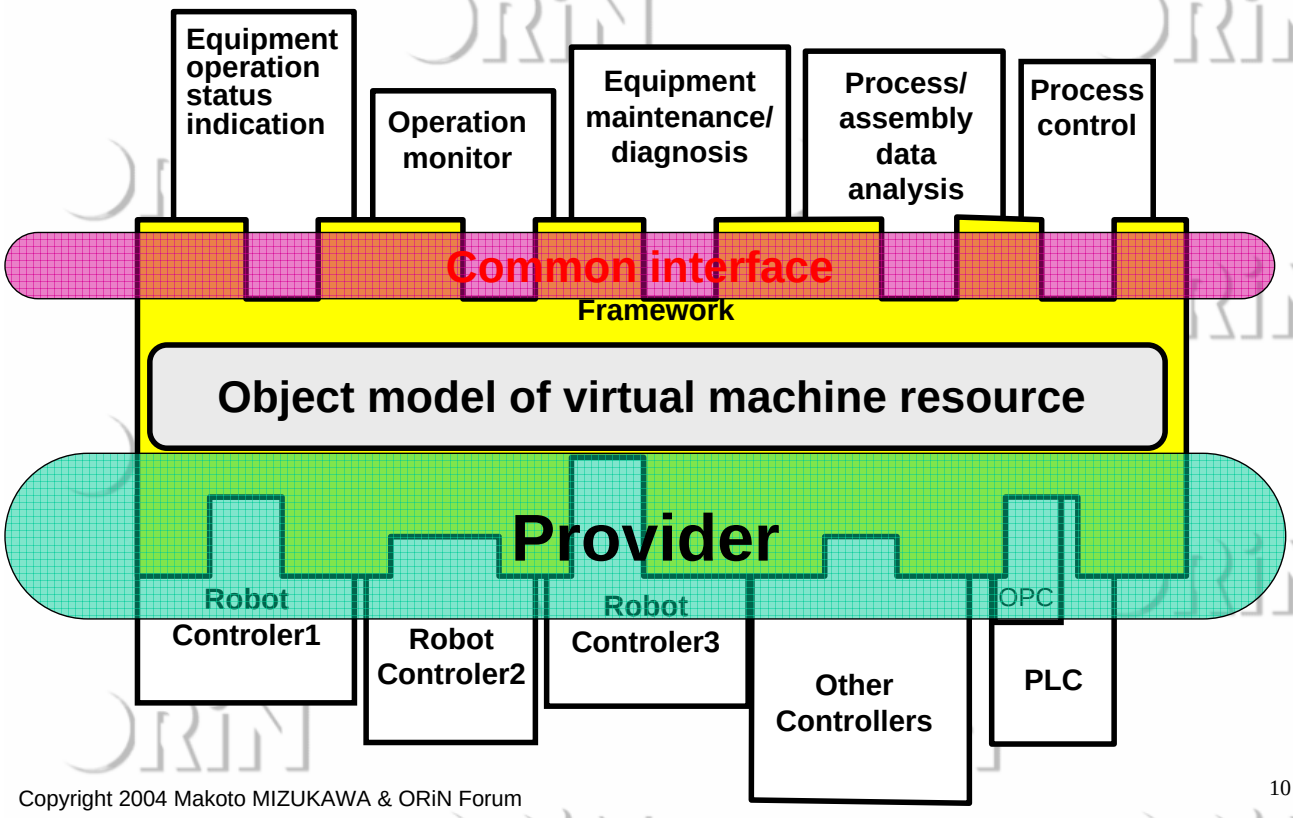
Open **R**esource **I**nterface for the **N**etwork
/**O**pen **R**obot **I**nterface for the **N**etwork



ORiN : Features

- Distributed Object
- Device Profiling using
- Web Service

Production line application using ORiN



Models

Virtual robot interface : ORiN

- RAO
  - RAO engine
    middleware with RAO Interface, which provides common functions to RAO providers as well as client applications
  - RAO provider
    software module with portions dependent on respective manufacturers
- RRD
  - RRD schema
    the data schema that defines names and data types to robot resources of various robot makers according to the XSD (XML data schema) specified by the XML schema
  - RRD data file
    actual data is described based on the RRD schema and its schema structure

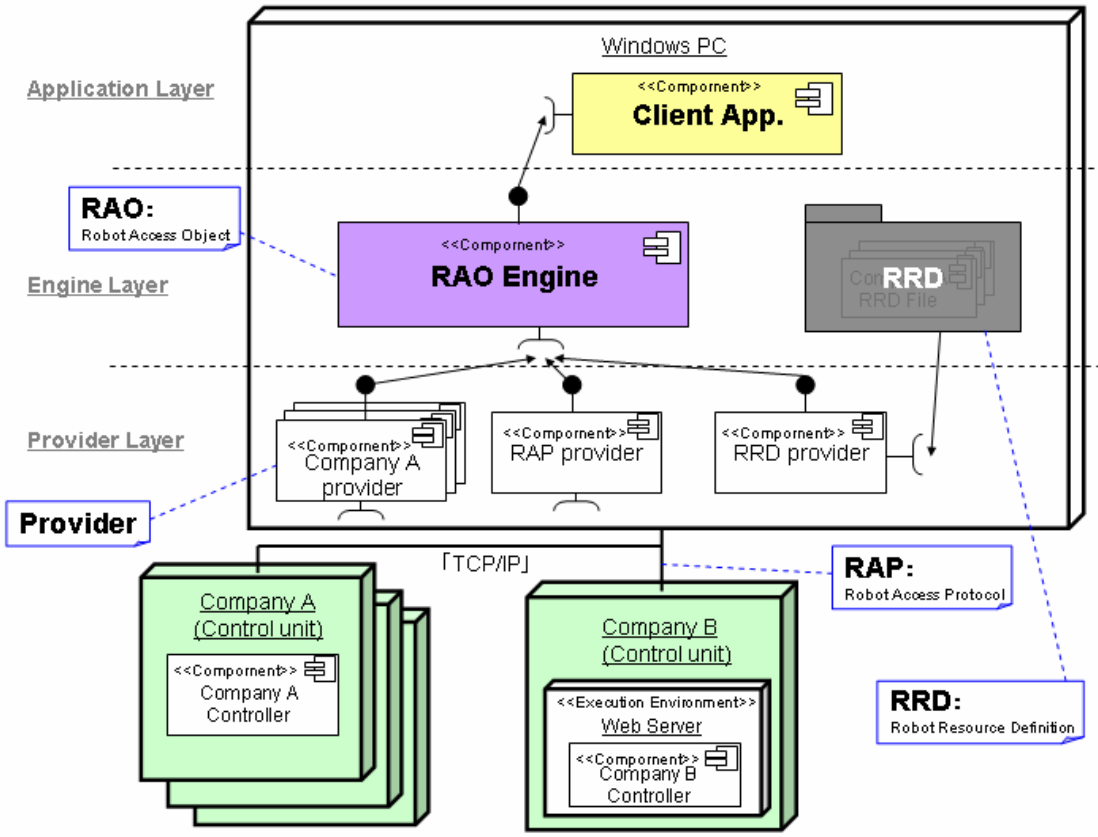
Key Technologies for Open Robot Model

- **Robot Access Object (RAO)**
  a middleware that provides standard program interface and services to robot controller based on the distributed object model

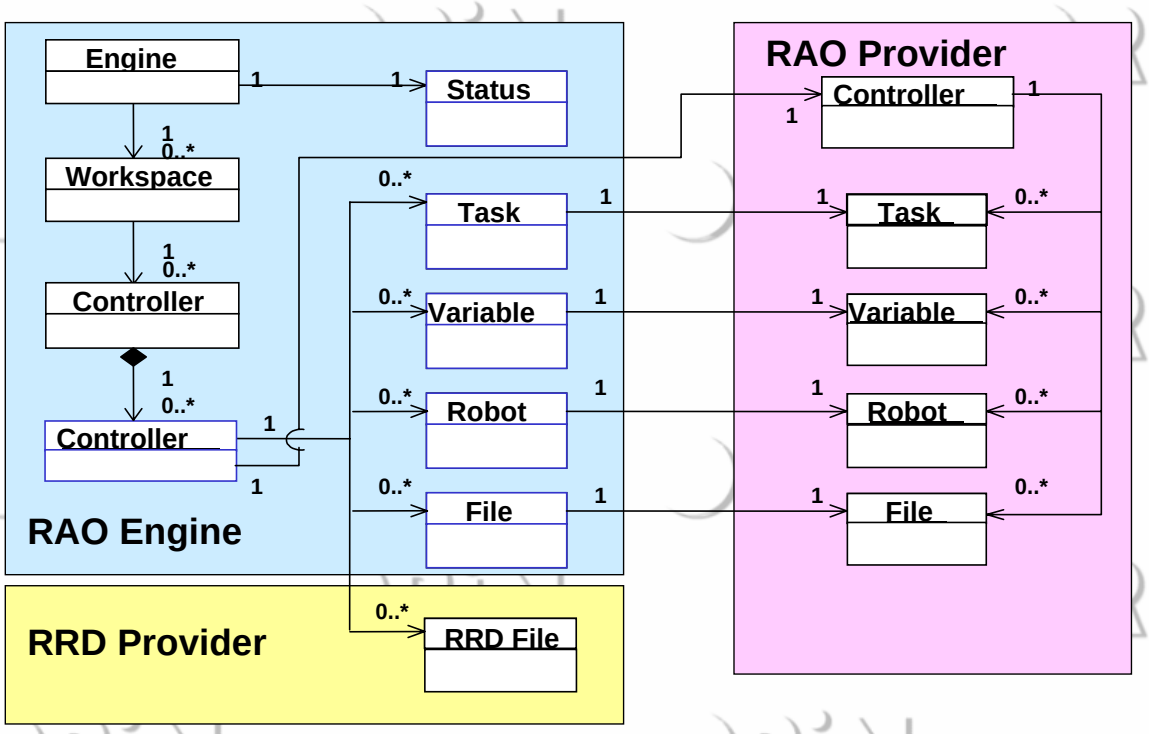
- **Robot Resource Definition Format(RRD)**
  a data schema that provides standard format for data from/to robot controller based on the eXtensible Markup Language (XML)

- **Robot Access Protocol(RAP)**
  standard protocol in the Internet using http and XML to allow data-exchange over firewalls

ORiN System Configuration



Classes of Virtual robot



RAO Engine Class

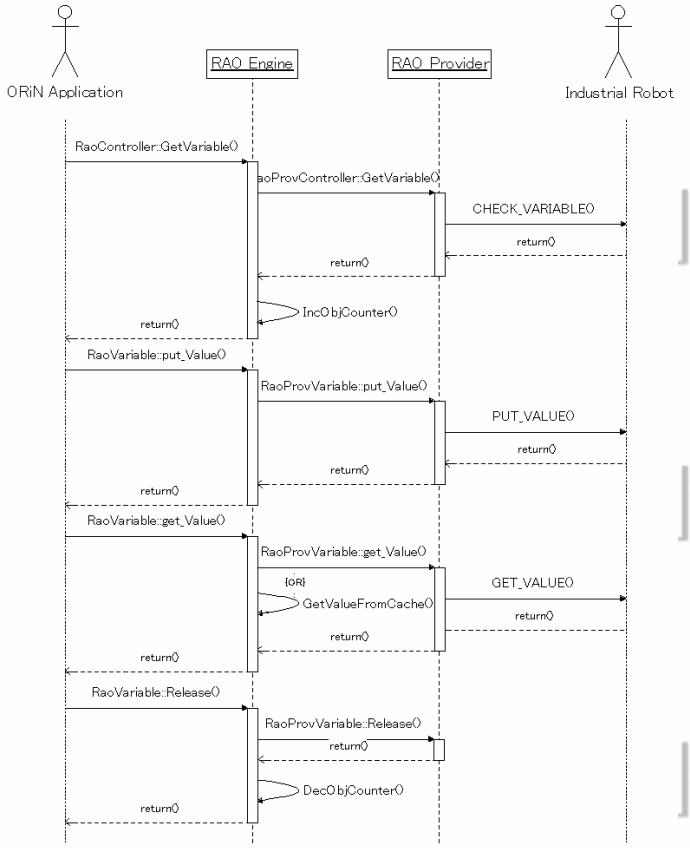
Class name	Resource	Description
RaoEngine	-	The upper-most ranked class of RAO, which is the only class that can directly create instance externally. [Root class, from which respective classes are generated]
RaoWorkspace	-	Class that manages robot controllers and users Note: User class is not supported at present. [Work class]
RaoController	Controller	Class that manages the entire resources of robot controller [Controller class]
RaoVariable	Variable	This handles all variables that are handled by the robot. [Variable class]
RaoRobot	Robot	Class that handles the information dependent upon the robot arm including arm length and link configuration Robot class
RaoFile	File	This includes the user program file that operates file-related information within the robot controller. [File class]
RaoTask	Task	This handles the status and information relating to the execution type user program within the robot controller. [Task class]
RaoStatus	-	Class for debugging, which handles the information relating to the RAO class. [Status class of RAO (for RAO engine supervision)]

RAO provider Class

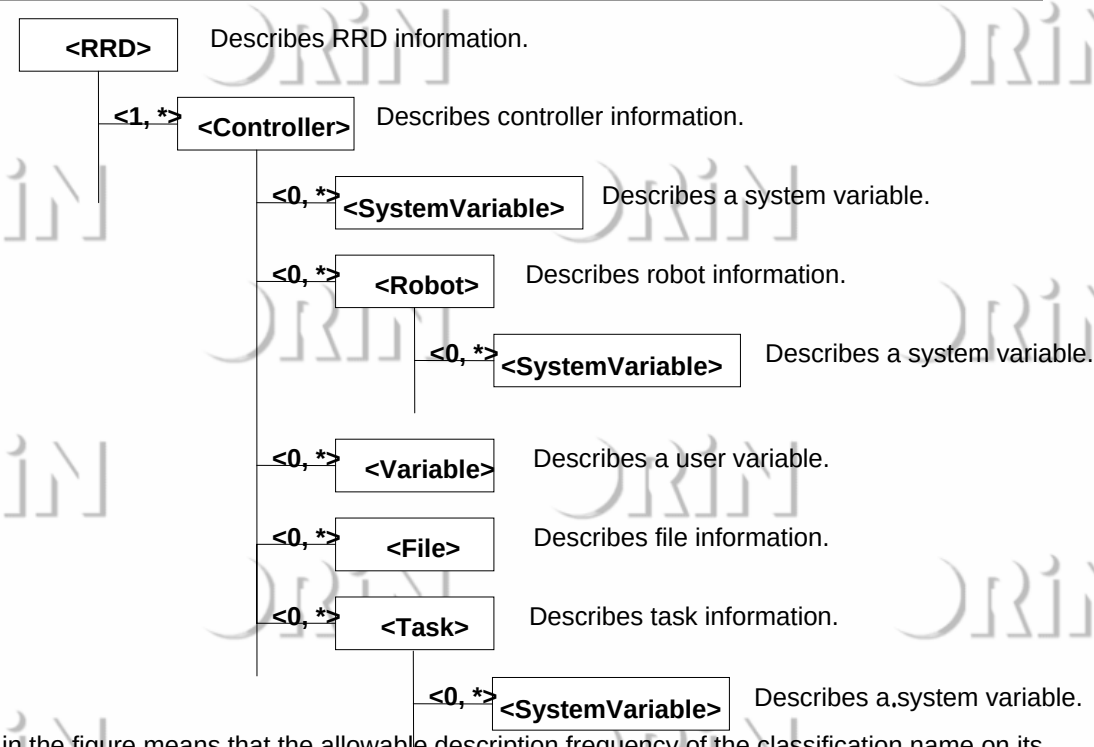
Class name	Resource	Description
RaoProvController	Controller	Controller class dependent upon each robot vendor corresponding to the RaoController [Controller class] It provides the functions relating to controller resources
RaoProvVariable	Variable	Variable class dependent upon each robot vendor corresponding to the RaoVariable [Variable class] It provides the functions relating to variable resources.
RaoProvRobot	Robot	Robot class dependent upon each robot vendor corresponding to the RaoRobot [Robot class] It provides the functions relating to robot resources.
RaoProvFile	File	File class dependent upon each robot vendor corresponding to the RaoFile [File class] It provides the functions relating to file and folder resources.
RaoProvTask	Task	Task class dependent upon each robot vendor corresponding to the RaoTask [Task class] It provides the functions relating to task resources.

Example: Sequence Chart

Sample Sequence : Variable Synchronus Call



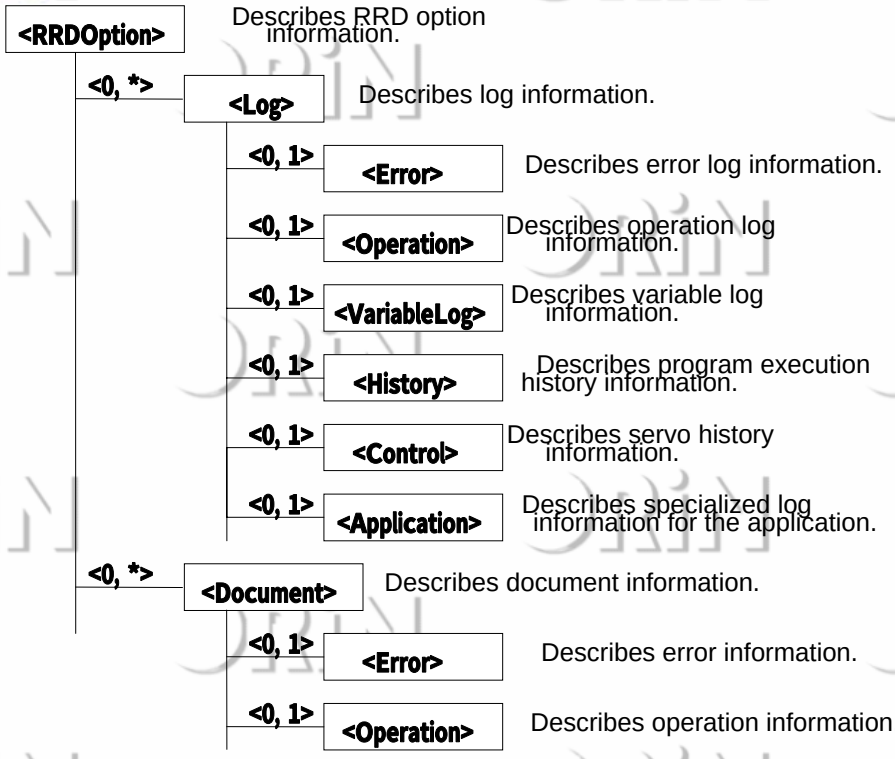
RRD Data Schema Configuration



Note 1) <0, *> in the figure means that the allowable description frequency of the classification name on its right side is at least 0, and unlimited at most.

Note 2) <1, *> in the figure means that the allowable description frequency of the classification name on its right side is at least once, and unlimited at most.

RRD Optional Data Schema Structure



Note 1) <0, *> in the figure means that the allowable description frequency of the classification name on its right side is at least 0, and unlimited at most.

Note 2) <1, *> in the figure means that the allowable description frequency of the classification name on its right side at least 0, and unlimited at most.

Implementation

Open Resource Interface for the Network /
Open Robot Interface for the Network

ORiN forum
The Japan Robot Association

http://www.orin.jp/

Design and Implementation Policy

Design policy

Vendor acceptable and useful standard

(1) Loose standardization
- allow various types of robot specifications

(2) Expandability
- vender unique option allowed

(3) Modularity of network protocols
- applicable to robots in the market

(4) Separation of specifications and implementation
- OOP technology

Implementation policy

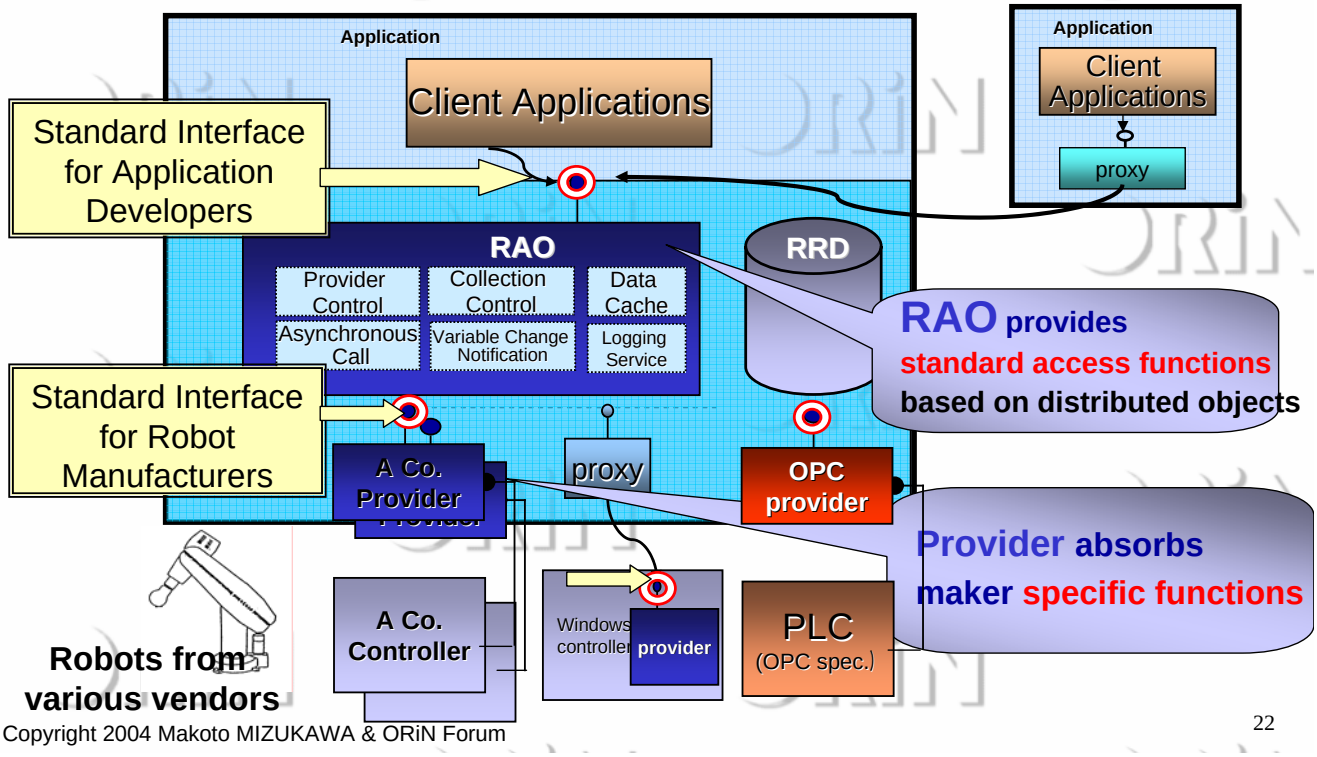
PC technology of the states of art

(1)De-facto standard platforms

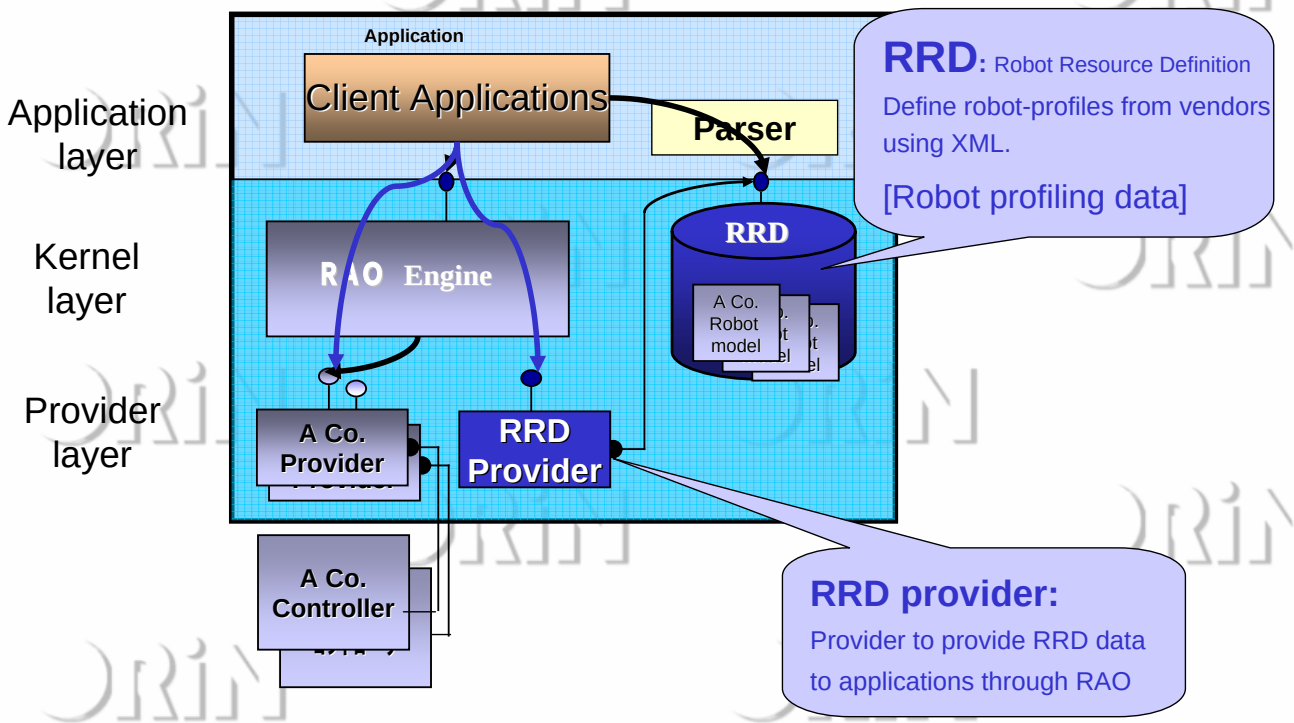
(2) Distributed object model
Network transparency,
Language independence

(3) Extended markup Language (XML)
Standard framework for defining vendor unique specification

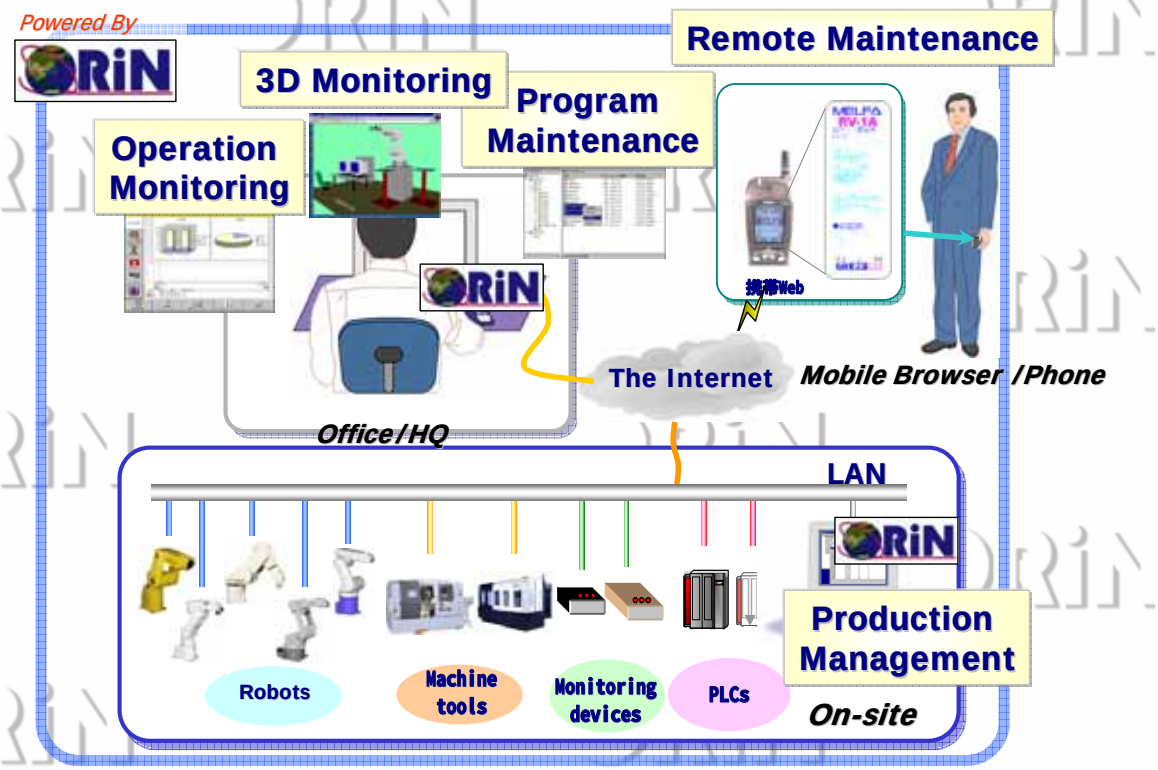
RAO Engine and Provider



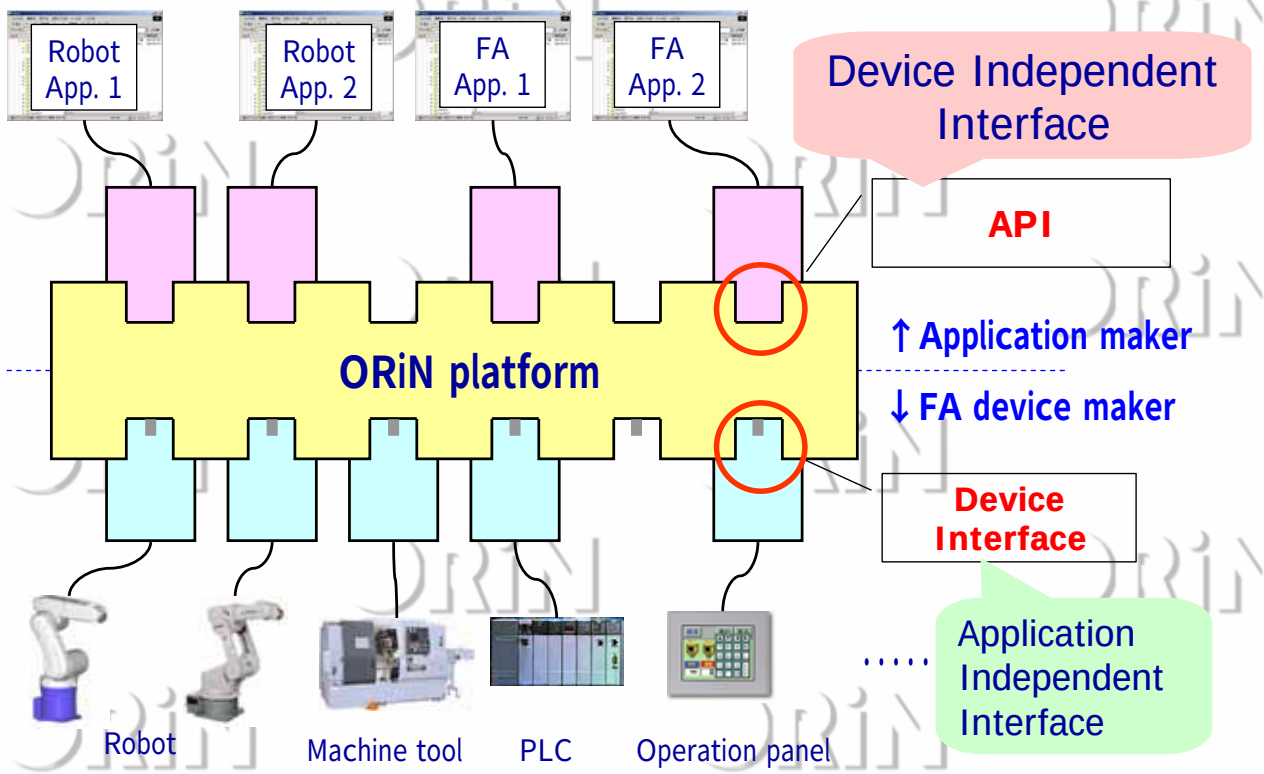
RRD: Robot Resource Definition



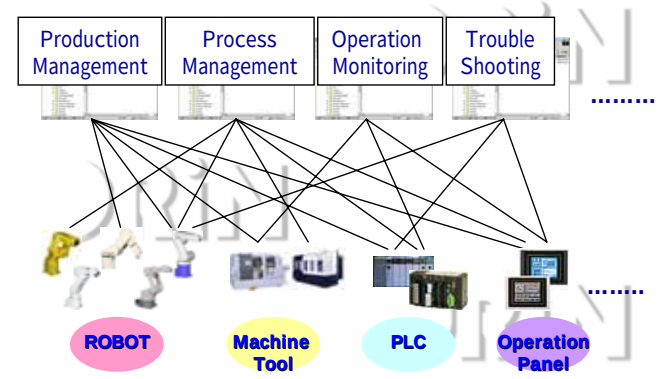
ORiN Applications



ORiN: Summary

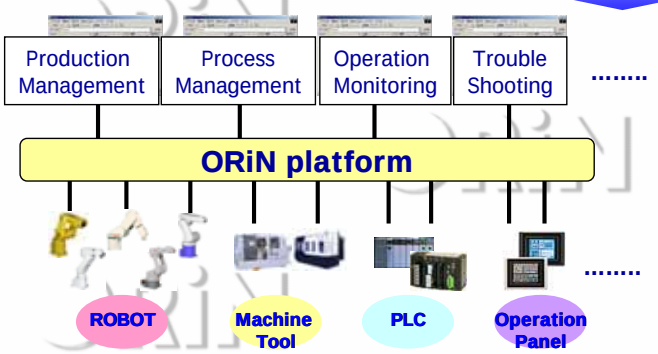


ORiN: Summary



Before

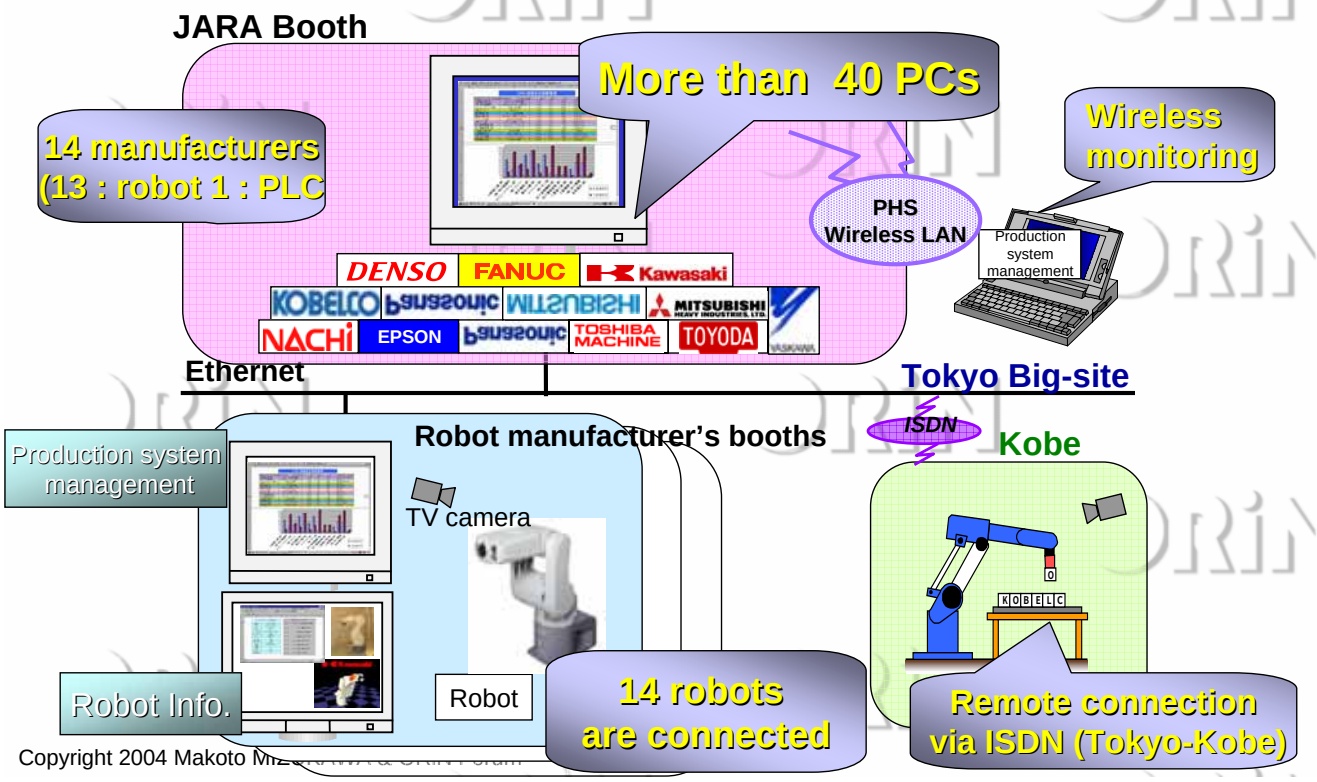
◆ Close dependency on Devices/ Networks/ Protocols
-> One-off/ Order-made
-> Low Reliability
-> Poor Maintenance



After

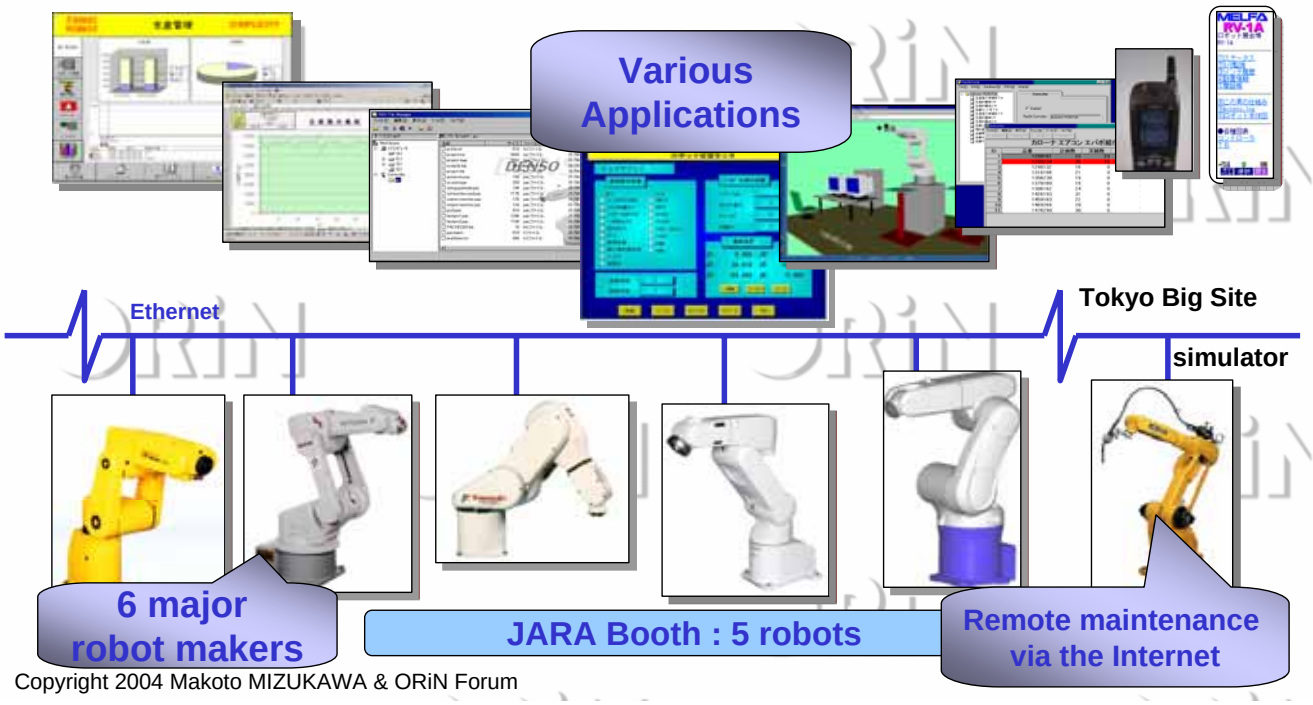
◆ Independency on Devices/ Networks/ Protocols
-> Standard Products
-> High Reliability
-> Good Maintenance

1999 International Robot Exhibition

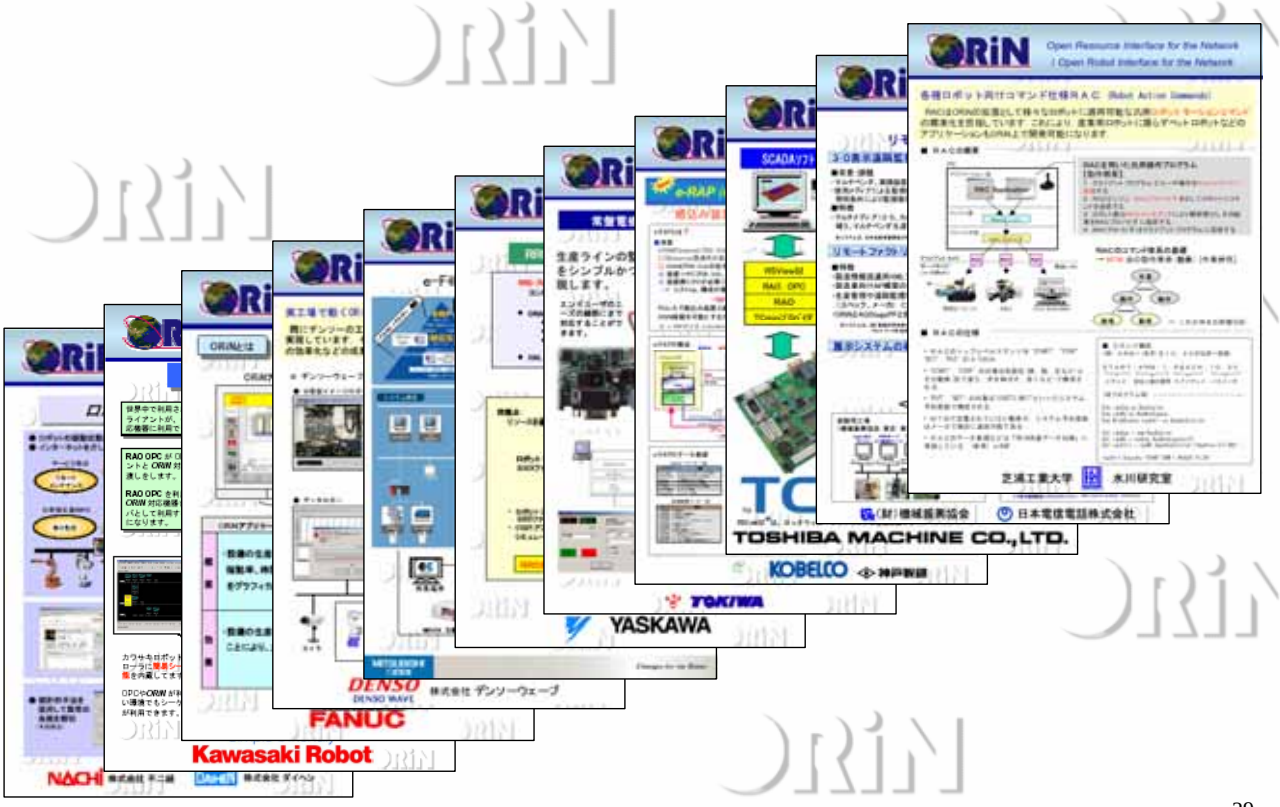


2001 International Robot Exhibition

FANUC YASKAWA Kawasaki MITSUBISHI DENSO KOBELCO



Int'l Robot Exhibition in Tokyo 2003



Merits of RAPI/ORiN

End Users

- Multi-vendor system
- Mfg. monitor system
- Remote maintenance

Cost Effective Operation

Software Vendors

- Market for various application software
- Cost reduction by standard package

Packages for Industries

- ***unified access method for robot data exchange***
- ***common application platform and API***

Independent from robot/FA-controller architectures

System Integrators

- High quality Mfg. System with less-cost
- WEB service solutions

Customizing Service in FA

Manufacturers

- Reduce development cost for comm. interface and testing
- Interoperability bw. manufacturers

Expand FA/ Robot Application Fields

ORiN forum Web-site

Open Resource Interface for the Network /
Open Robot Interface for the Network

Japan Robot Association

ORiN forum

http://www.orin.jp/