Introduction to ORiN2

DENSO WAVE Inc.

http://www.denso-wave.com/

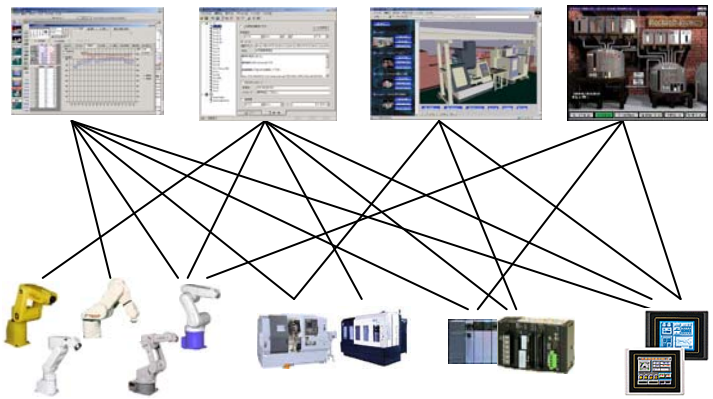
ORiN Forum

http://www.orin.jp/

1. Overview

SKIP

The ORiN Goal



Before

◆Dependent on the devices, networks, protocols, etc.

- ➢"A la carte" applications
- ➢Low reliability and low maintainability

(Background) Reuse of hardware and software becomes more important in order to achieve the vertical start-up of manufacturing system.

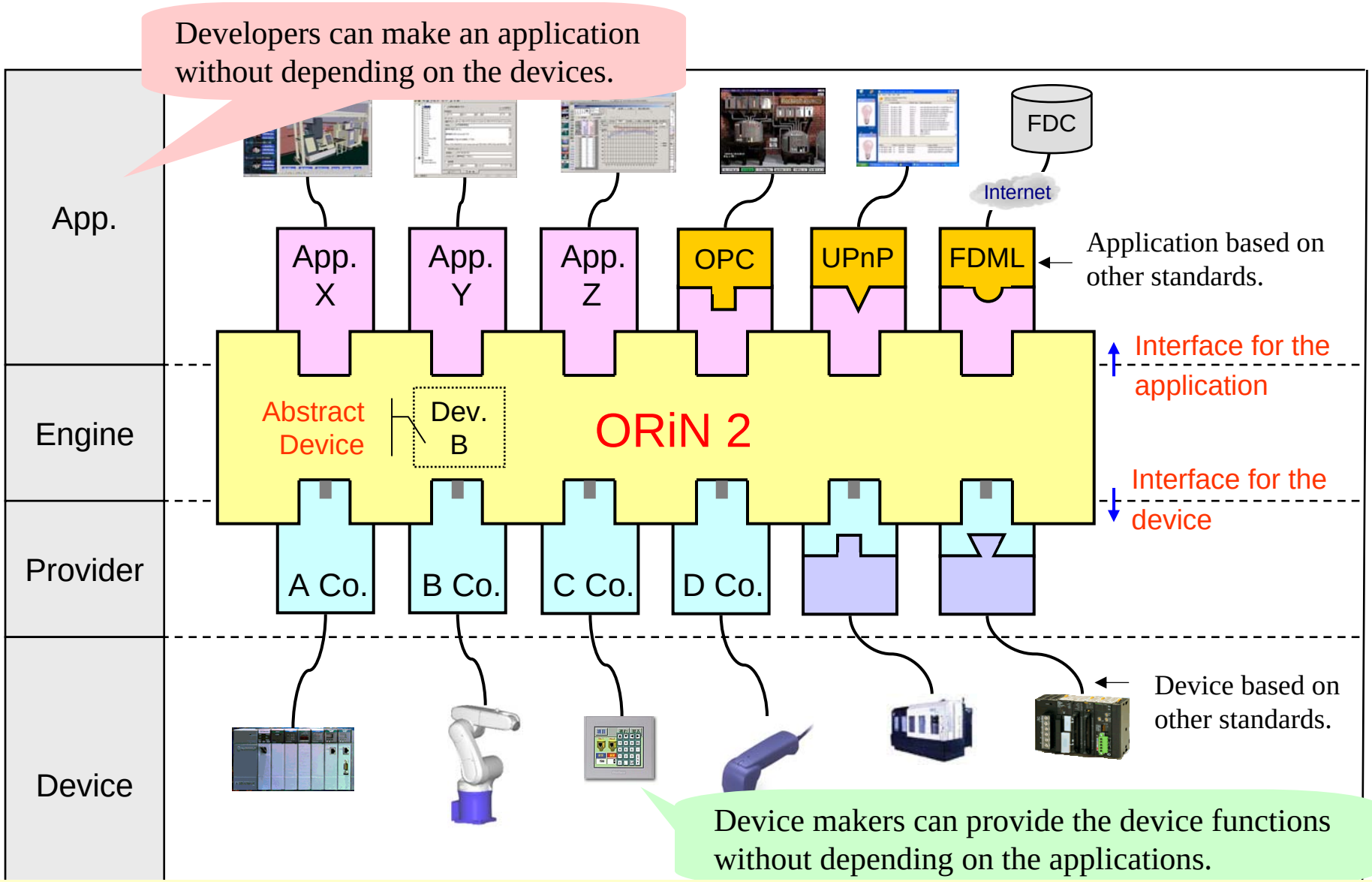
After

◆Independent of the devices, networks, protocols, etc.

- ➢"General" applications
- ➢High reliability and high maintainability

Our goal is to achieve the advanced reusability of FA software.

ORiN Architecture



ORiN provides a standard access means for an application.

Development Policy

Philosophy

Flexible standardization for high-level unification of generality and diversity.

Design Policy

Realizing widely acceptable specification as a standard interface.

(1) Acceptable interface for various kinds of device.
(2) Expandable interface for future extension.
(3) Concurrent development of specifications and implementation.

Implementation Policy

Developing advanced functions by adopting industry standard technologies.

(1) Adoption of de-facto standard such as Windows and Linux.
(2) Adoption of distributed object technology such as DCOM and CORBA.
(3) Adoption of XML technology such as XML-Schema.

High-level unification of standardization and diversity.

History

◆ Started as a standardization project of Japan Robot Association, JARA.

◆ A full-dress development started in 1999 with support of NEDO (Japanese governmental organization).

◆ Several robot makers displayed a prototype application at Int. Robot Exhibition in 1999 and 2001.

◆ At the end of 2001, ORiN version 1 was released.

◆ "ORiN Forum" was established in 2002.

◆ After field test for 3 years, ORiN version 2 was released in 2005.

Leading robot companies developed an advanced software platform.

ORiN Forum

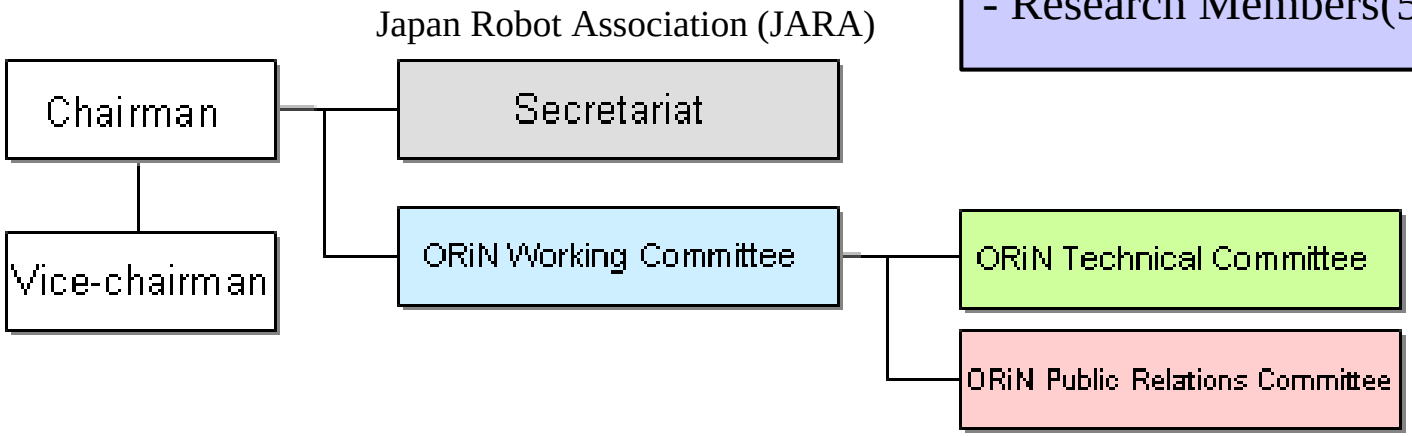
◆ The aim of the Forum

- Promotion of ORiN utilization
  - Release specifications and applications
  - ORiN seminars
- Enhancement and updating of ORiN
  - Timely update of ORiN specifications

◆ Members of ORiN Forum

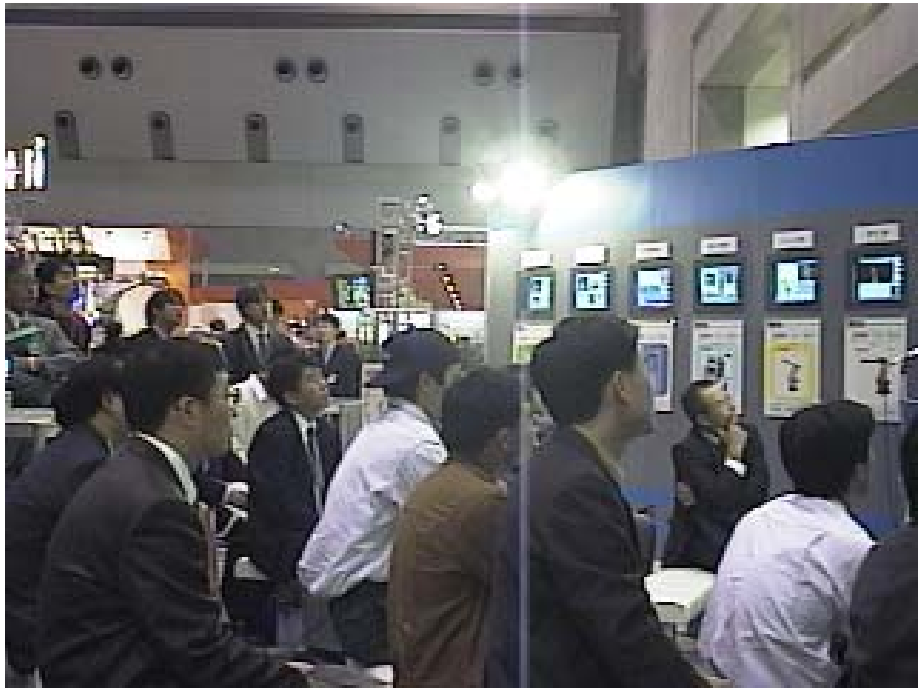
- Executive Members(7)
  - DENSO WAVE
  - Fanuc
  - Kawasaki Heavy Industry
  - Kobe Steel
  - Mitsubishi Electric
  - Nachi-Fujikoshi
  - Yaskawa Electric
- Regular Members (3)
- Associate Members(5)
- Research Members(5)

◆ ORiN Forum Organization



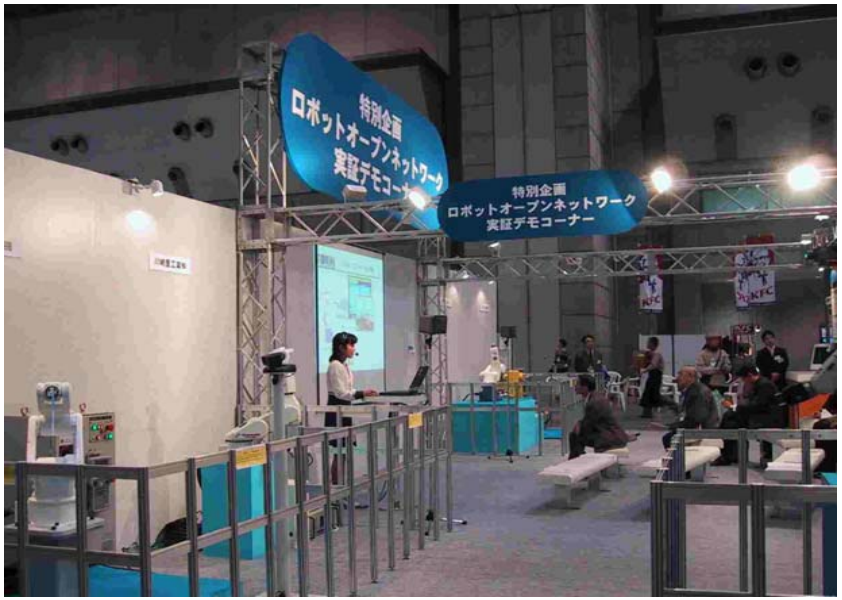
The ORiN technologies are continuously updated by Forum members.

’99 Int. Robot Exhibition in Japan



’99 Int. Robot Exhibition in Japan

’01 Int. Robot Exhibition in Japan



Strategic Standardization

Three major policies in standardization project

1. The new standard is not common subset of existing proprietary specifications, but a strategic standard for a new factory automation field required in the future.

2. The activity is not only defining specifications, but also implementing the standard. The activity includes concurrent development of experimental environment like SDK.

3. This project aims for high-level balance of generality and expandability. The project develops new technologies to be adoptable to various devices.

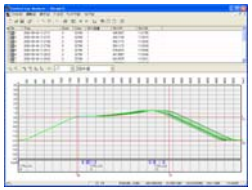
Example:
- Pre-registered Object Collection（CAO model）
- Dynamic binding mechanism (Expandable interface)

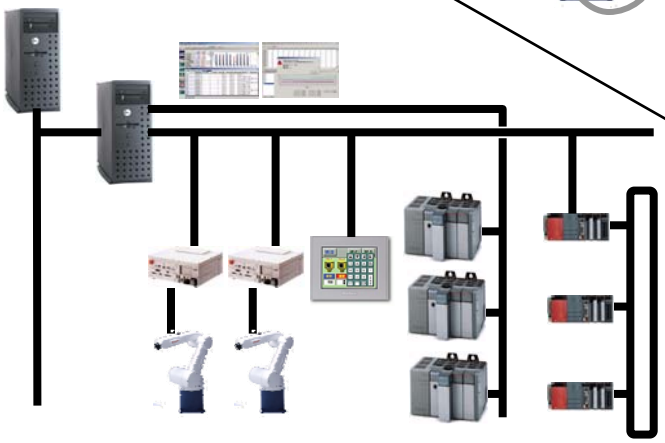
Concurrent propulsion of standardization and development for new technology field

Classification of the ORiN applications

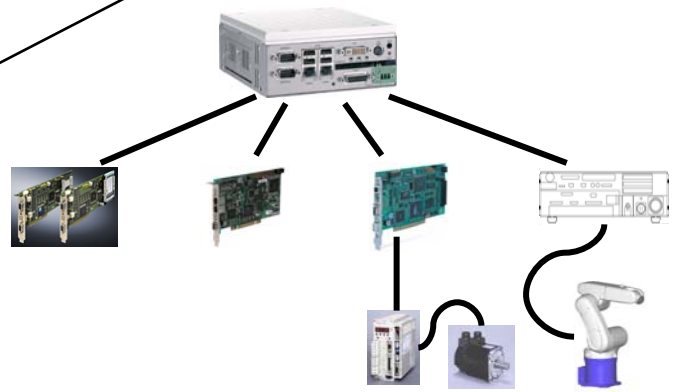
Proprietary Tool

Features:
• Maker proprietary Tools (Off-line programming tool, etc.)





ORiN2



Features:
• Large scale system
• Monitoring Application (SCADA, MES, etc.)

Features:
• PC based Controller
• Custom Application
• Integrated Cell Controller

Open System

Semi-Embedded System

ORiN2 is adoptable to many kinds of application.

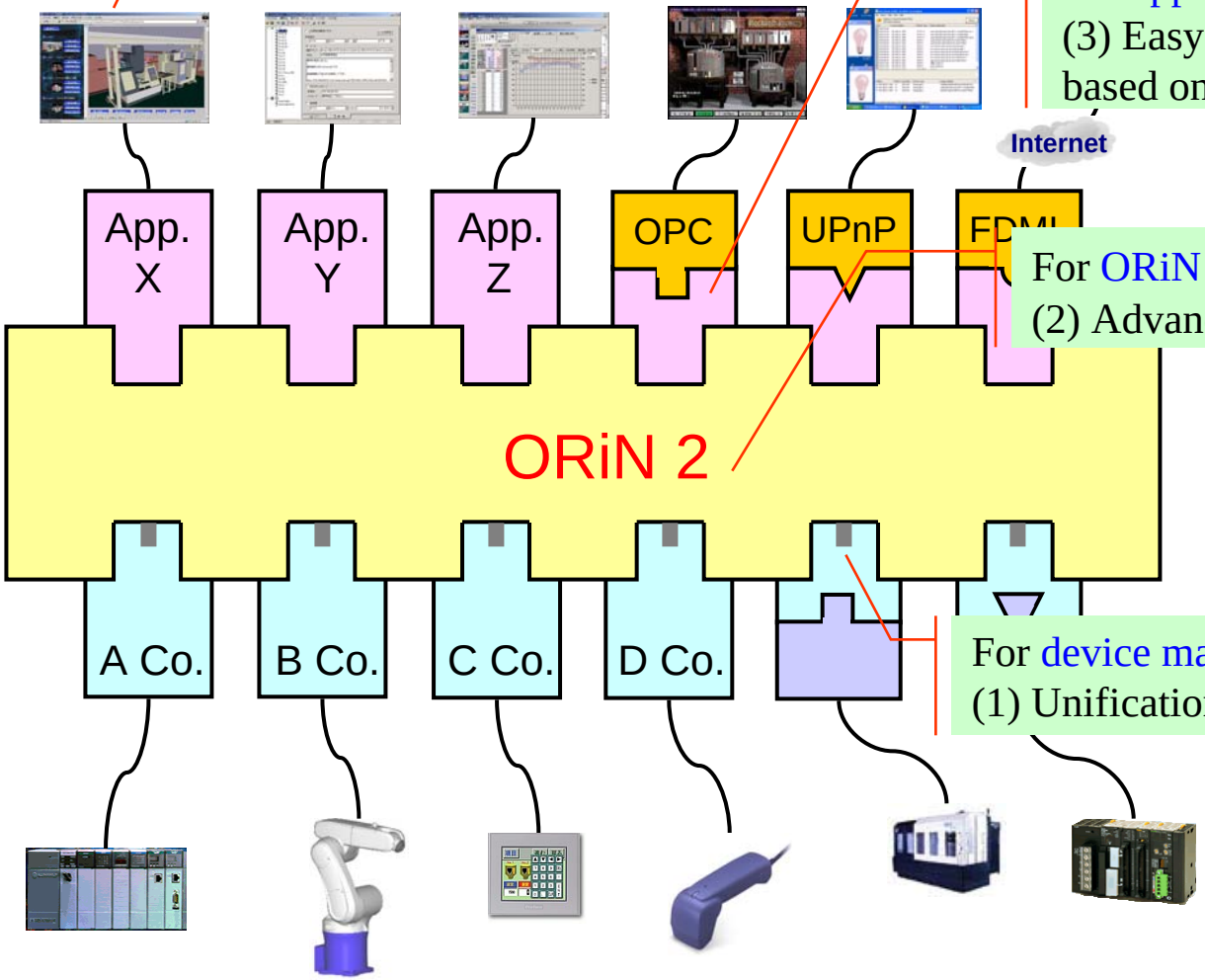
Merits of ORiN

For end-users,
(4) Good reusability of an application.

For application developers,
(3) Easy porting of an application based on the other standards.

For ORiN application developers,
(2) Advanced functions of ORiN.

For device makers,
(1) Unification of a device interface.



The mfg. system development cost can be much reduced.

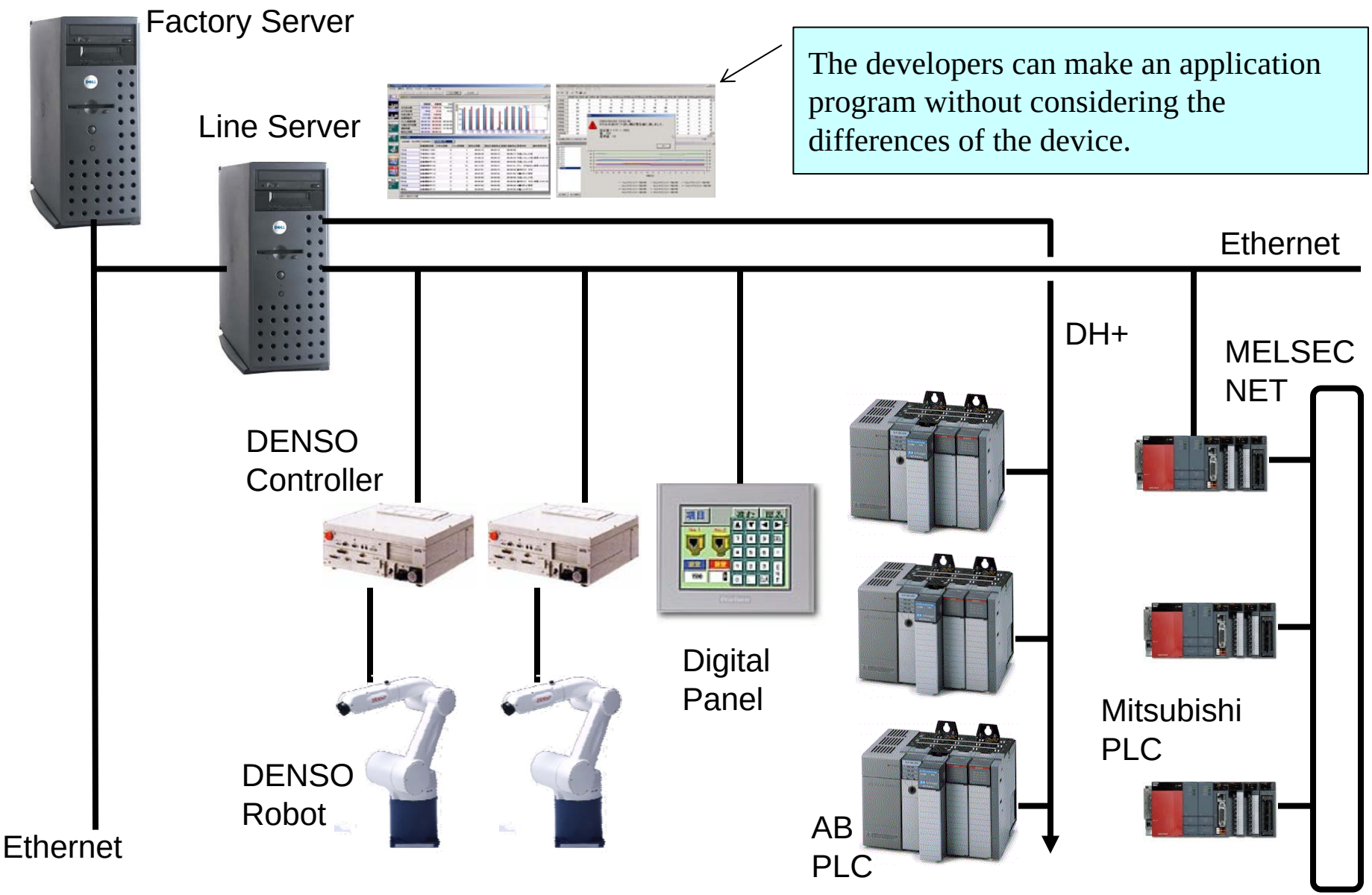
Applications at DENSO

Application 【total：6 kinds】	1. Operation Monitoring System 2. Cylinder Deterioration Monitoring 3. Rotary-Rack Maintenance System 4. Error Notification System 5. Program File Manager 6. Manufacturing Portal Site
Item (Sampling Rate)	12000 information items (8 ms – 30 min)
Database 【total：3 kinds】	SQL Server 7.0 (Windows) PostgreSQL 6.5 (Linux) MySQL 4.1 (Linux)
Network 【total：5 kinds】	Ethernet, IEEE802.11b, DH+ (Allen Bradley), MELSECNET 2 and 10（Mitsubishi）
Device 【total：235 】	ROBOT（DENSO）×84 PLC（Allen Bradley）×54 PLC（Mitsubishi）×45 Operation Panel（Digital）×52

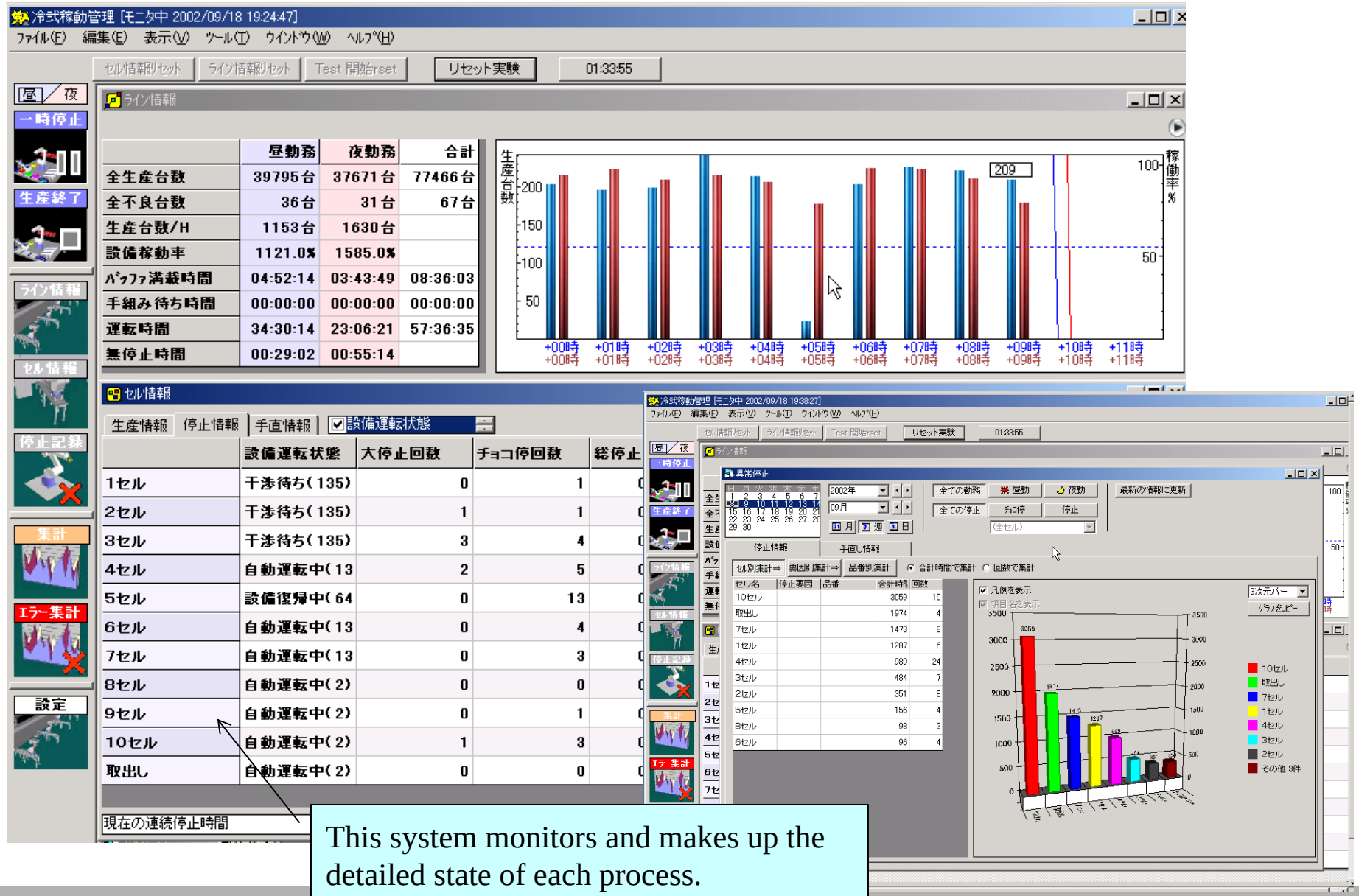
We need much information for "KAIZEN"

Jan. 15, 2005

DENSO System Overview

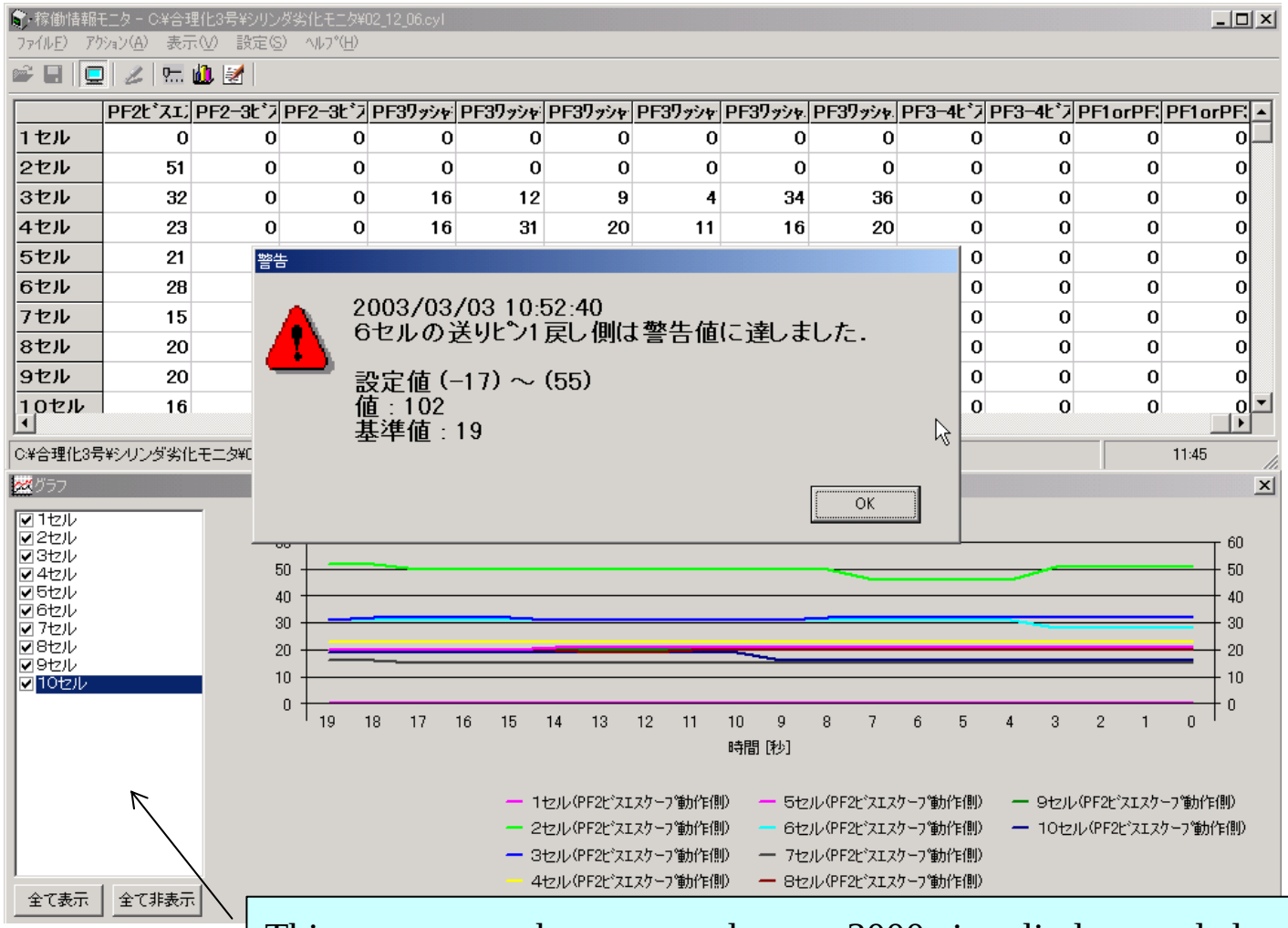


App 1: Operation Monitor



This system monitors and makes up the detailed state of each process.

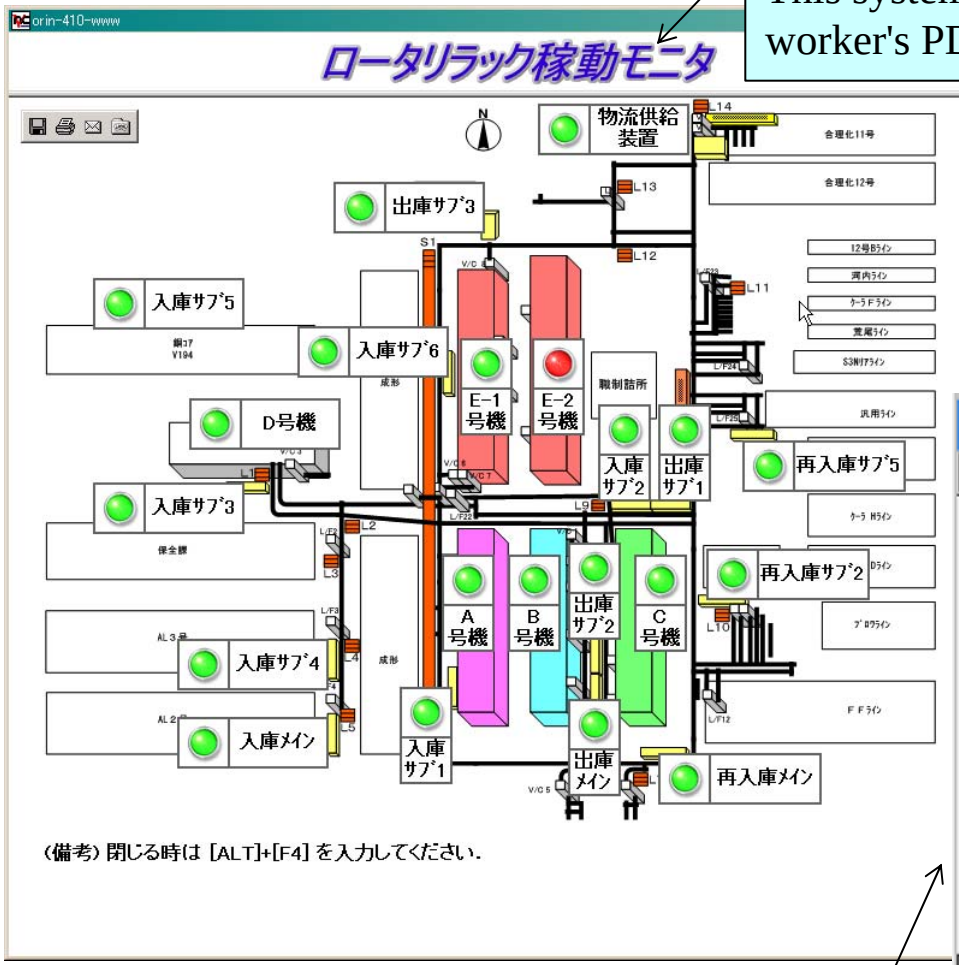
App 2: Air-Cylinder Monitor



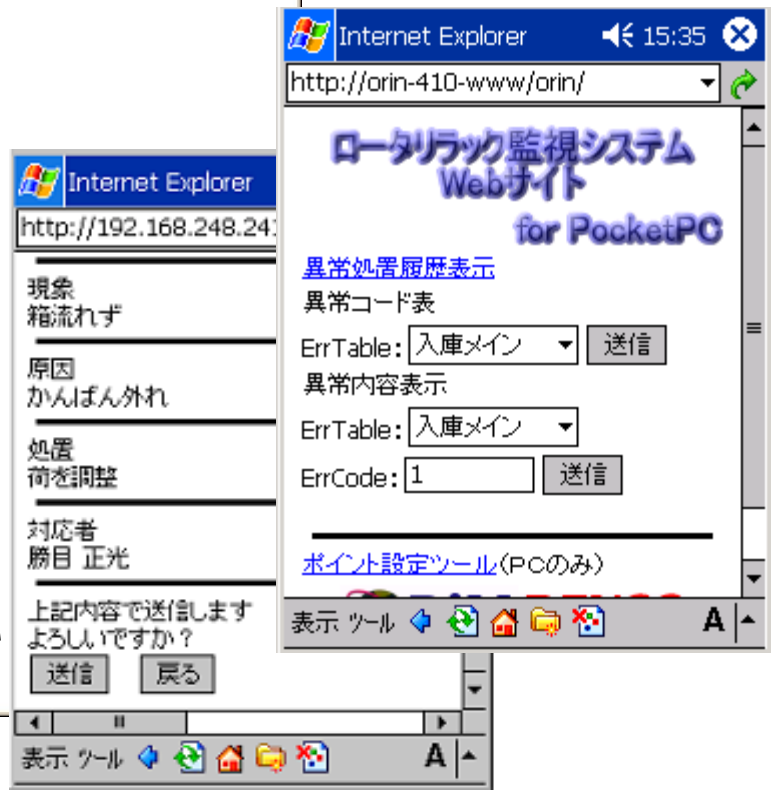
This system watches constantly over 3000 air cylinders, and alerts to a worker before broken.

App 3: Rotary-Rack Maintenance

This system can notify an error and its place through the worker's PDA immediately.

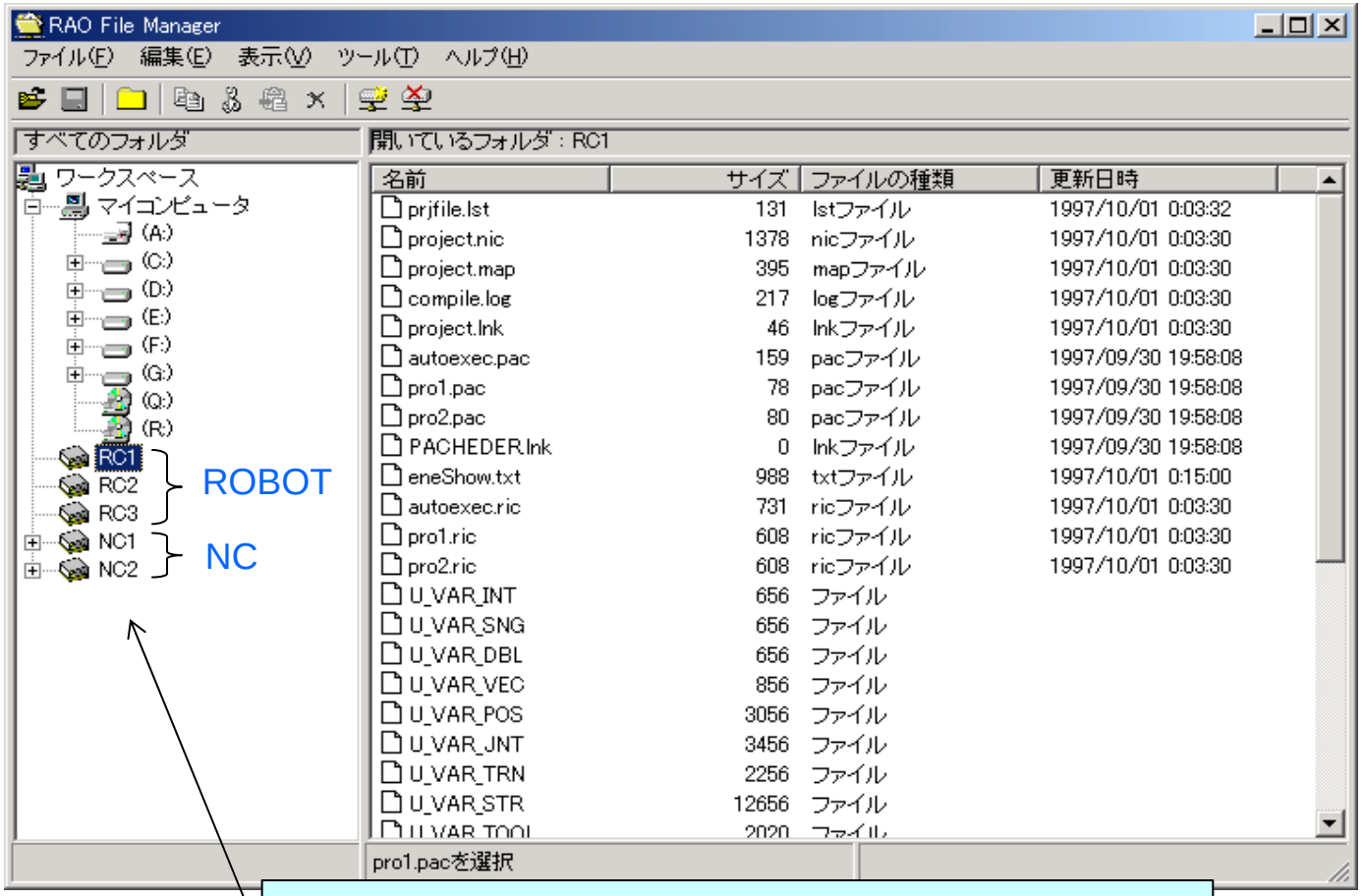


PDA



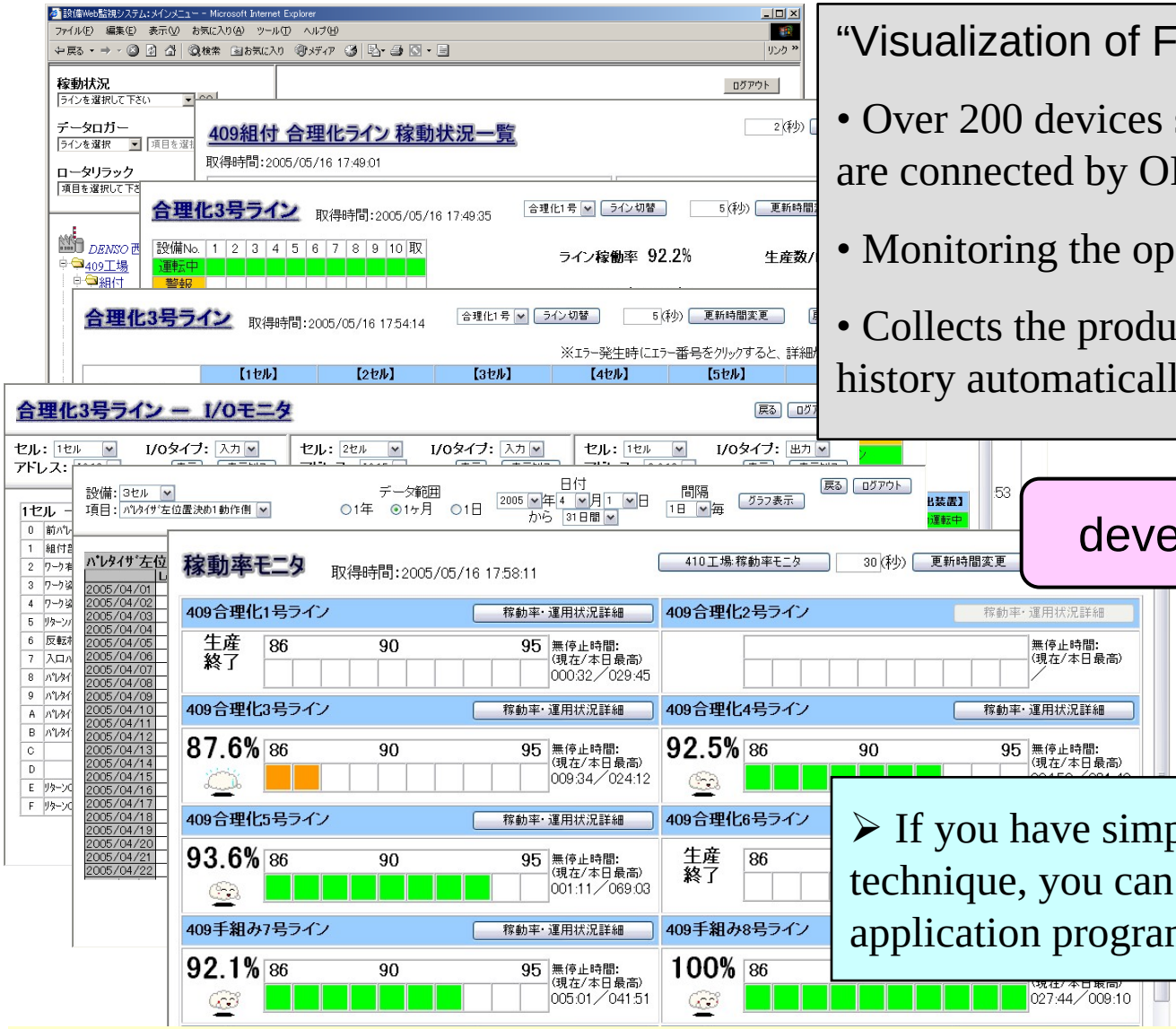
Maintainer can transmit job history to the server on site directly.

App 4: File Manager



Without depending on the devices, this program can manage their files.

App 5: Manufacturing Portal Site



"Visualization of Facility Control"

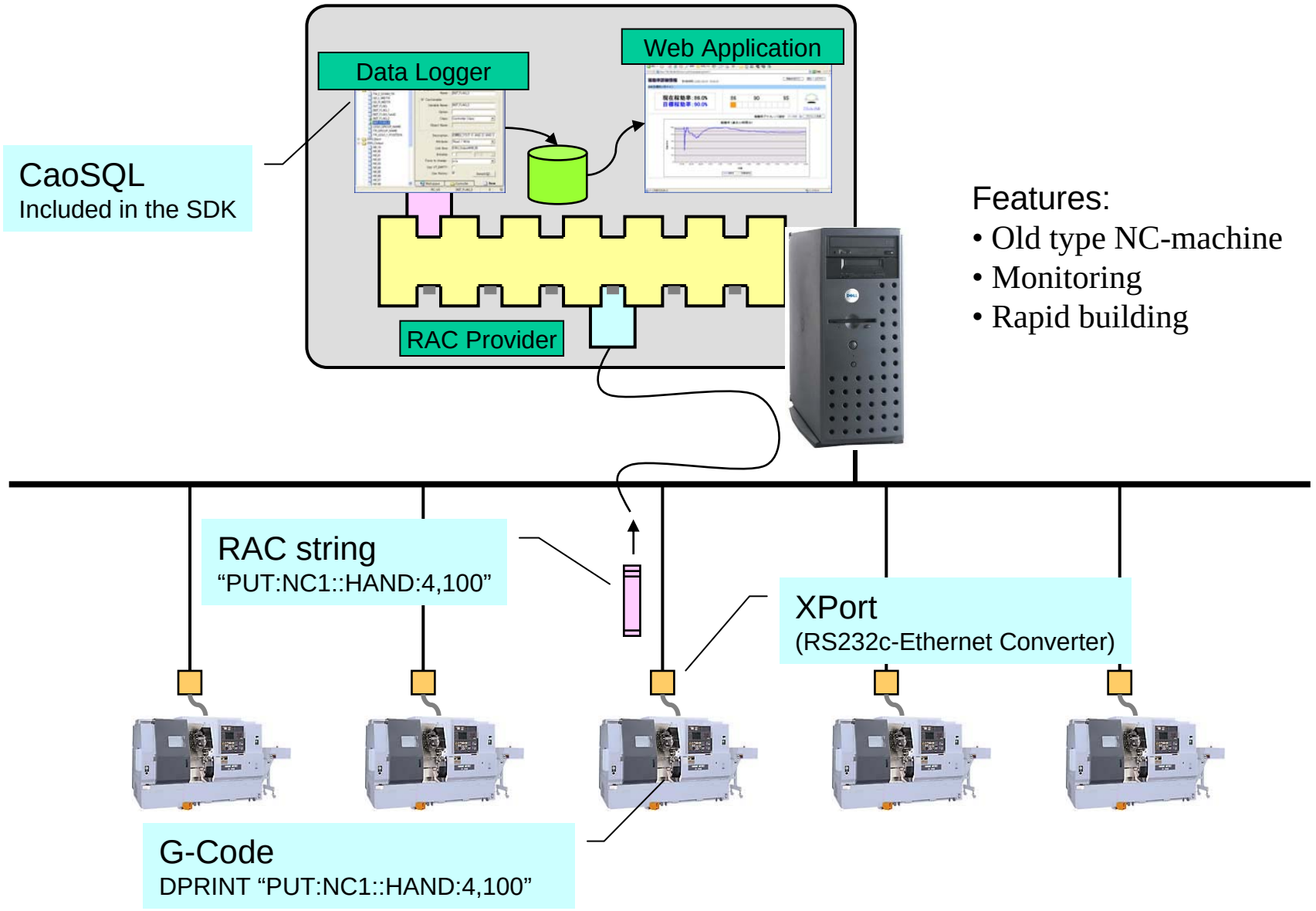
- Over 200 devices such as a Robot and a PLC are connected by ORiN.
- Monitoring the operation in real time in 24H.
- Collects the production data and manages its history automatically.

developed by the USER.

➢ If you have simply web programming technique, you can make these FA application programs easily.

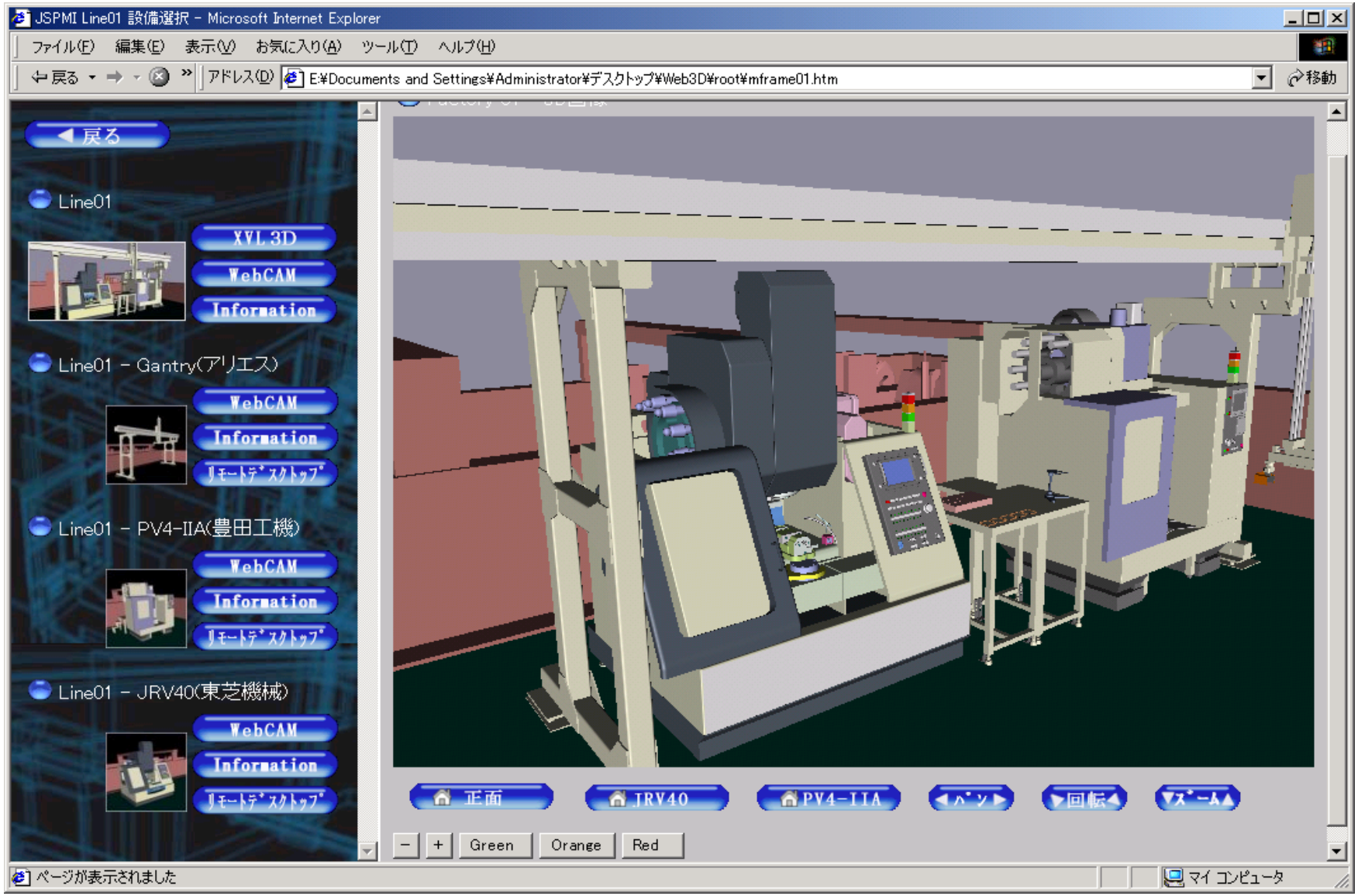
ORiN greatly makes the FA application development easy.

App 6: NC machine monitor



* This application was developed by JSPMI and DENSO WAVE.

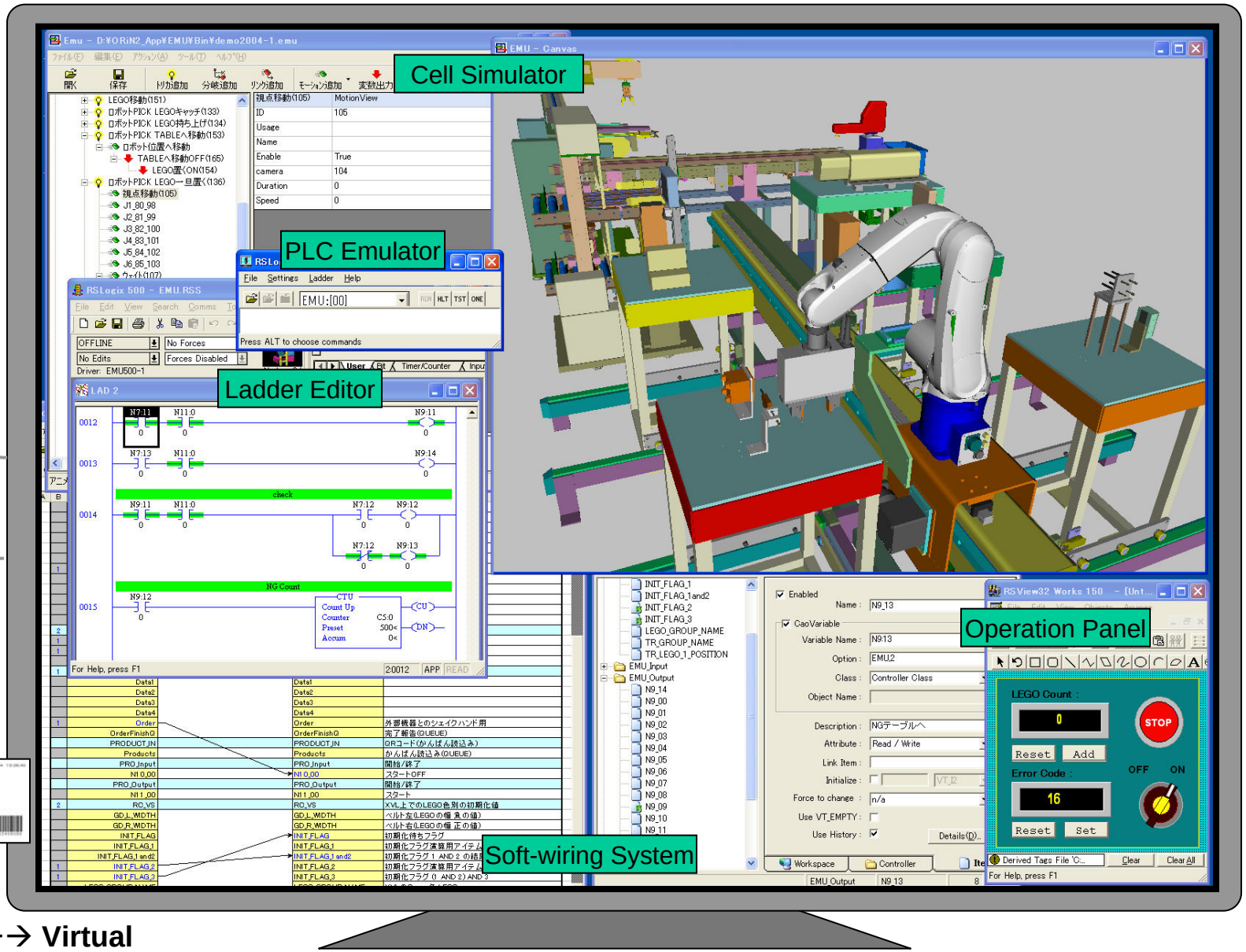
App 7: 3D Remote Maintenance



* This application was developed by JSPMI and DENSO WAVE.

http://www.orin.jp/

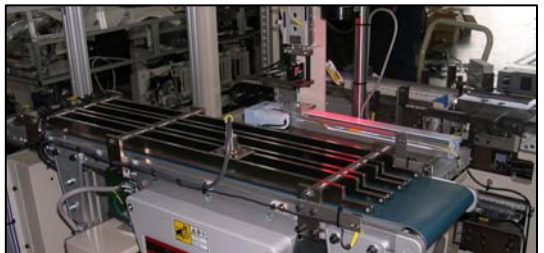
App 8: Distributed Simulation

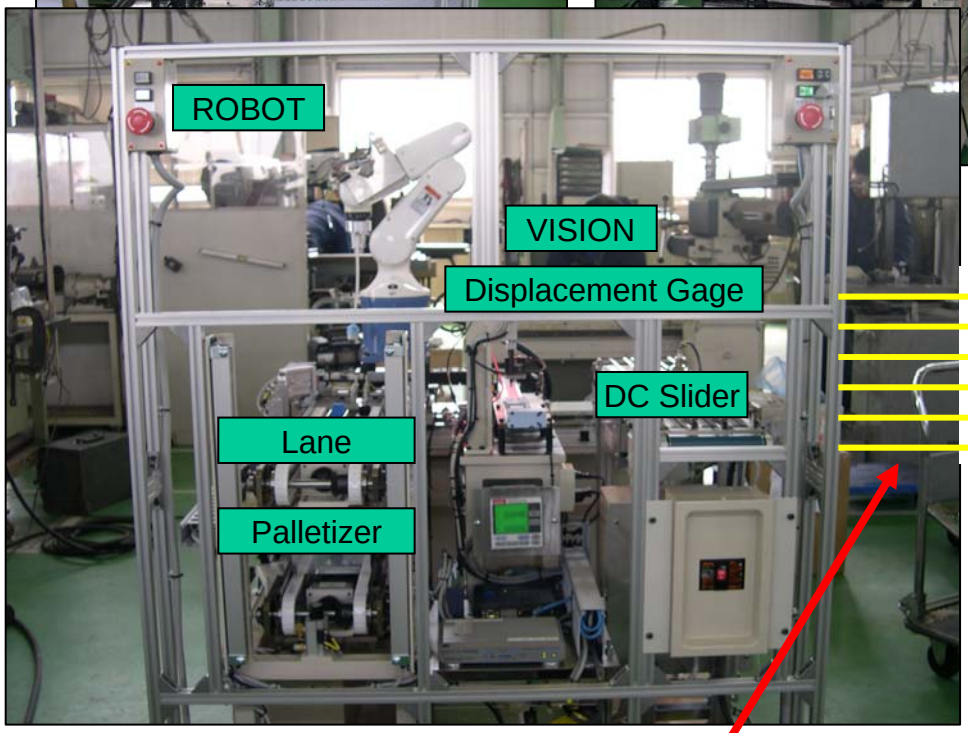


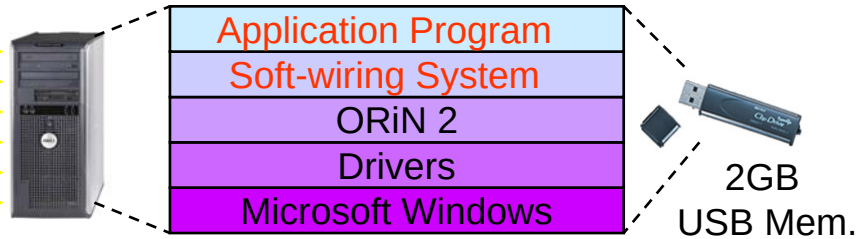
* This application was developed by JSPMI and DENSO WAVE.

http://www.orin.jp/

App 9: PC Integrated Cell Controller







All signals are connected to a PC except for the critical safety signals.
(DeviceNet, S-LINK, Ethernet, DIO, RS232C, etc.)

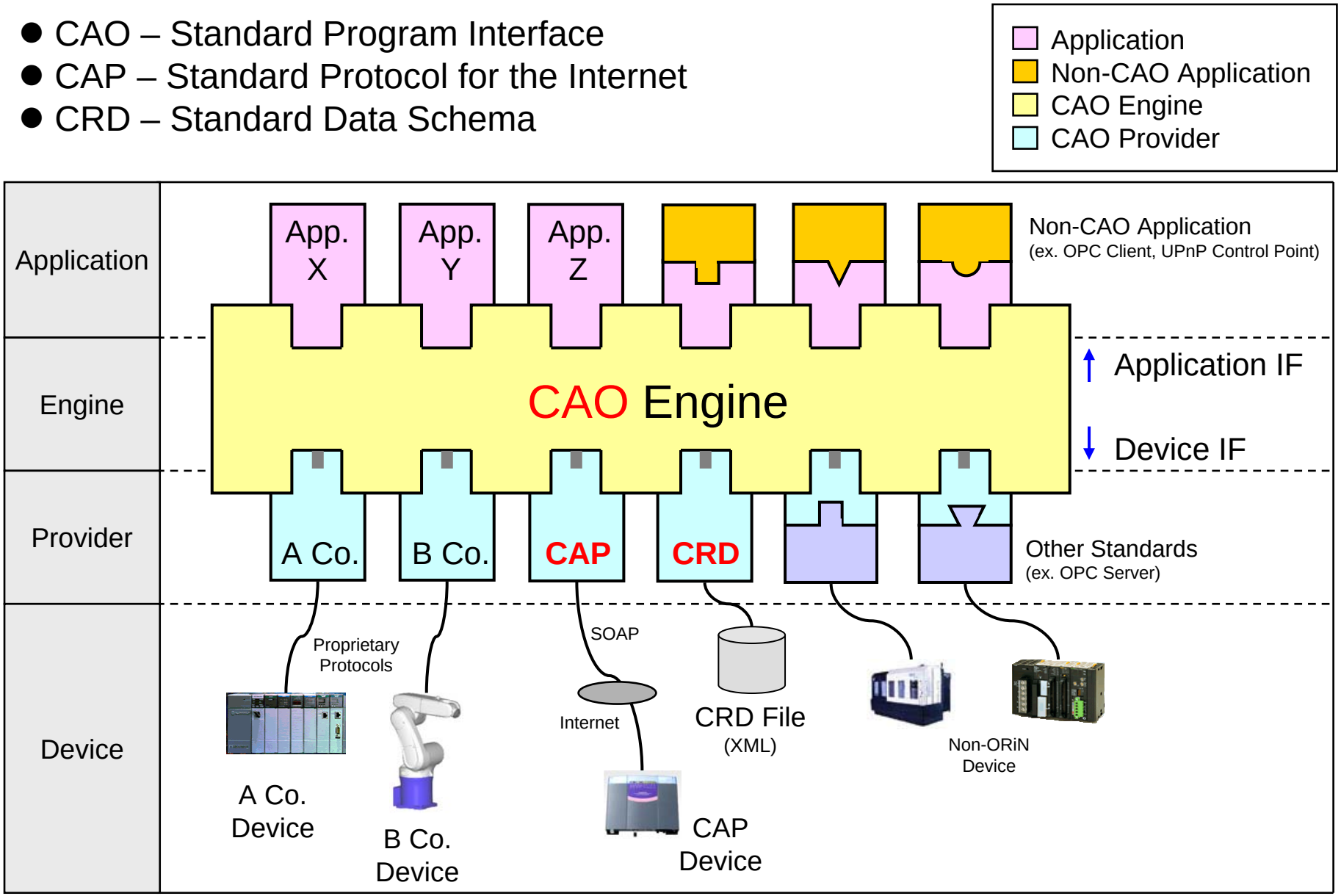
Expectations:
Program Size ⇒ 1/5
Processing Time (Non-motion command) ⇒ 1/10
Backup Time ⇒ 1/20
Maintenance Cost ⇒ 1/3

2. ORiN Architecture

SKIP

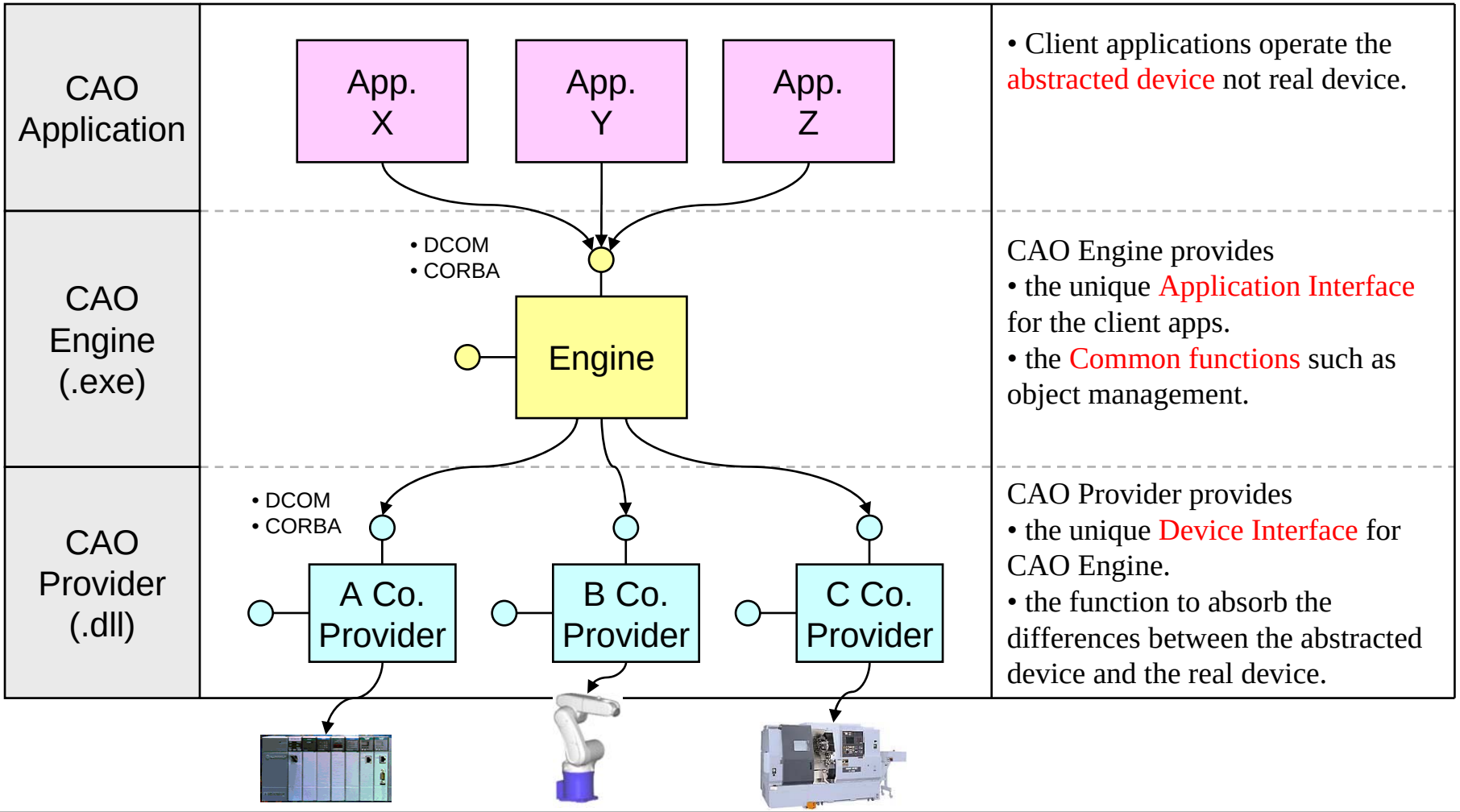
ORiN Architecture

- CAO – Standard Program Interface
- CAP – Standard Protocol for the Internet
- CRD – Standard Data Schema



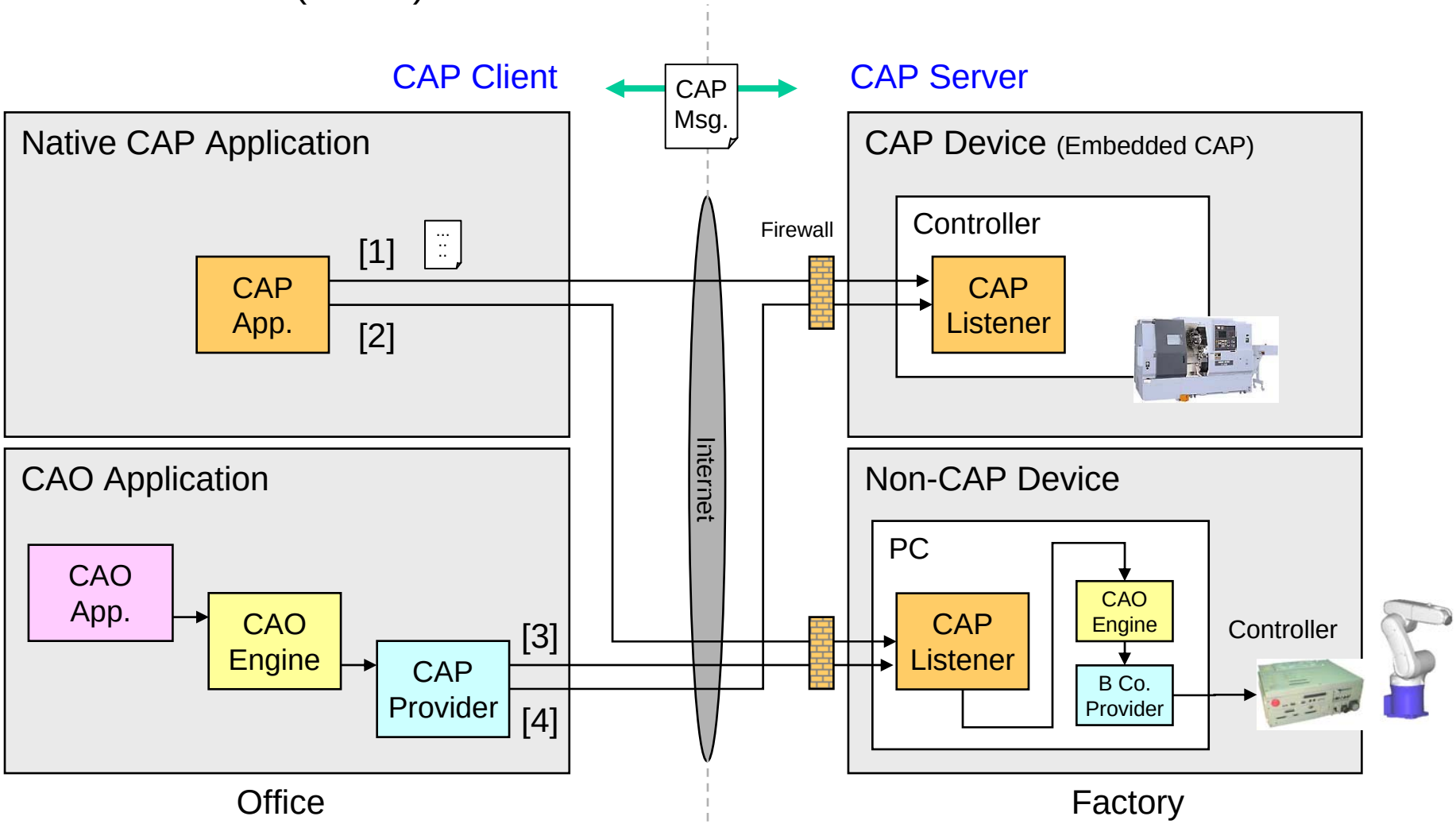
CAO Architecture

- CAO: Controller Access Object
- Standard Program Interface
- Distributed Object Model (DCOM, CORBA)



CAP Architecture

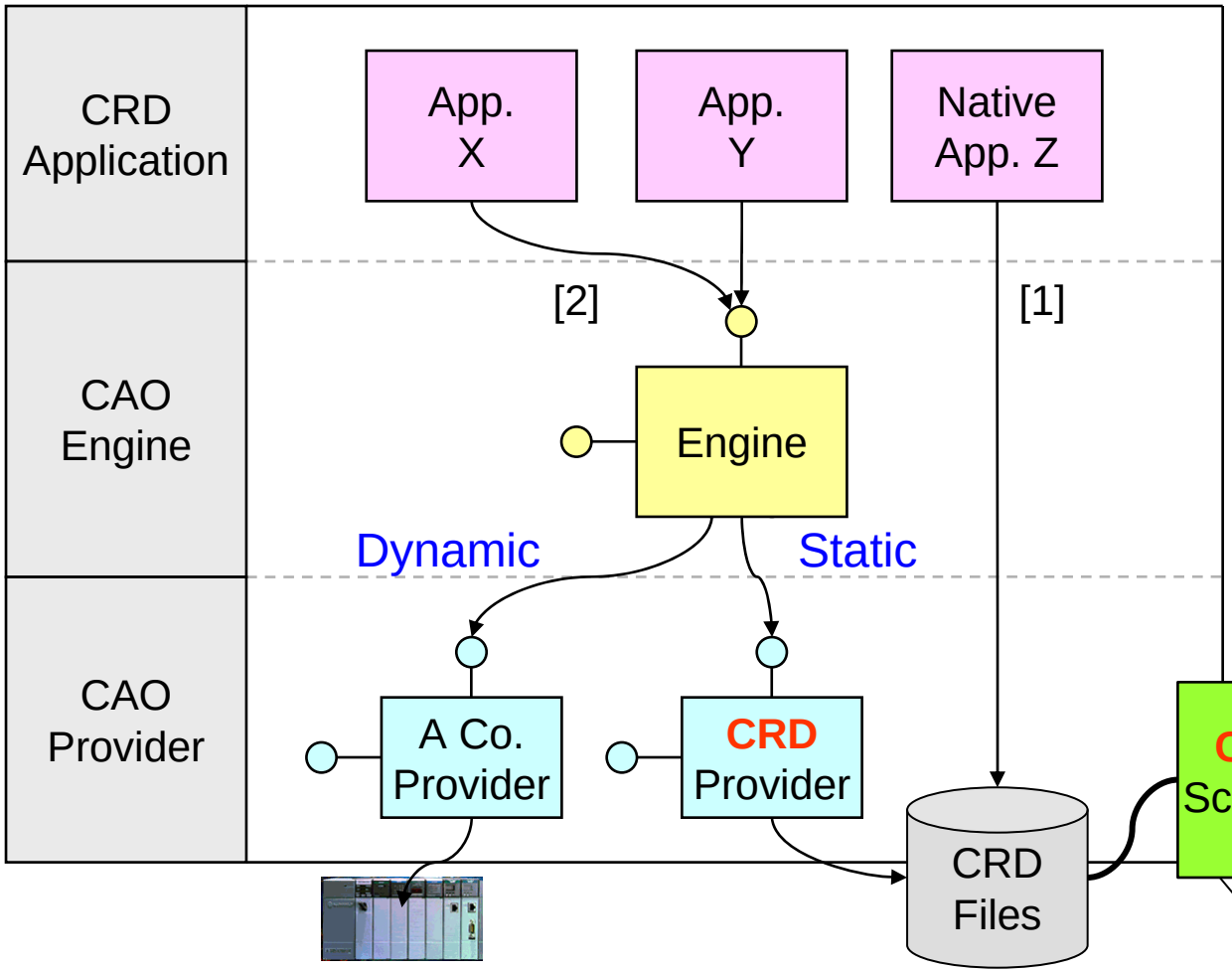
- CAP: Controller Access Protocol
- Standard Protocol for the Internet
- WEB Service (SOAP)



CRD Architecture

- CRD: Controller Resource Definition
- Standard Data Schema
- XML Schema

Example:
- Robot kinematics definition
- CAO model definition
- Data sharing with legacy system



[1] Native Application
App. ➔ CRD

[2] CAO Application
App. ➔ CAO ➔ CRD

Target Resources

Controller (ex. Robot, PLC, NC)

File

Task

File Mgr.

Task Mgr.

• Interpreter

Opt. Board

• Extension Card

Controller

Pendant

I/O

Variable

Robot Mgr.

Path Gen.

Servo

Non-volatile Resource

Message

Command

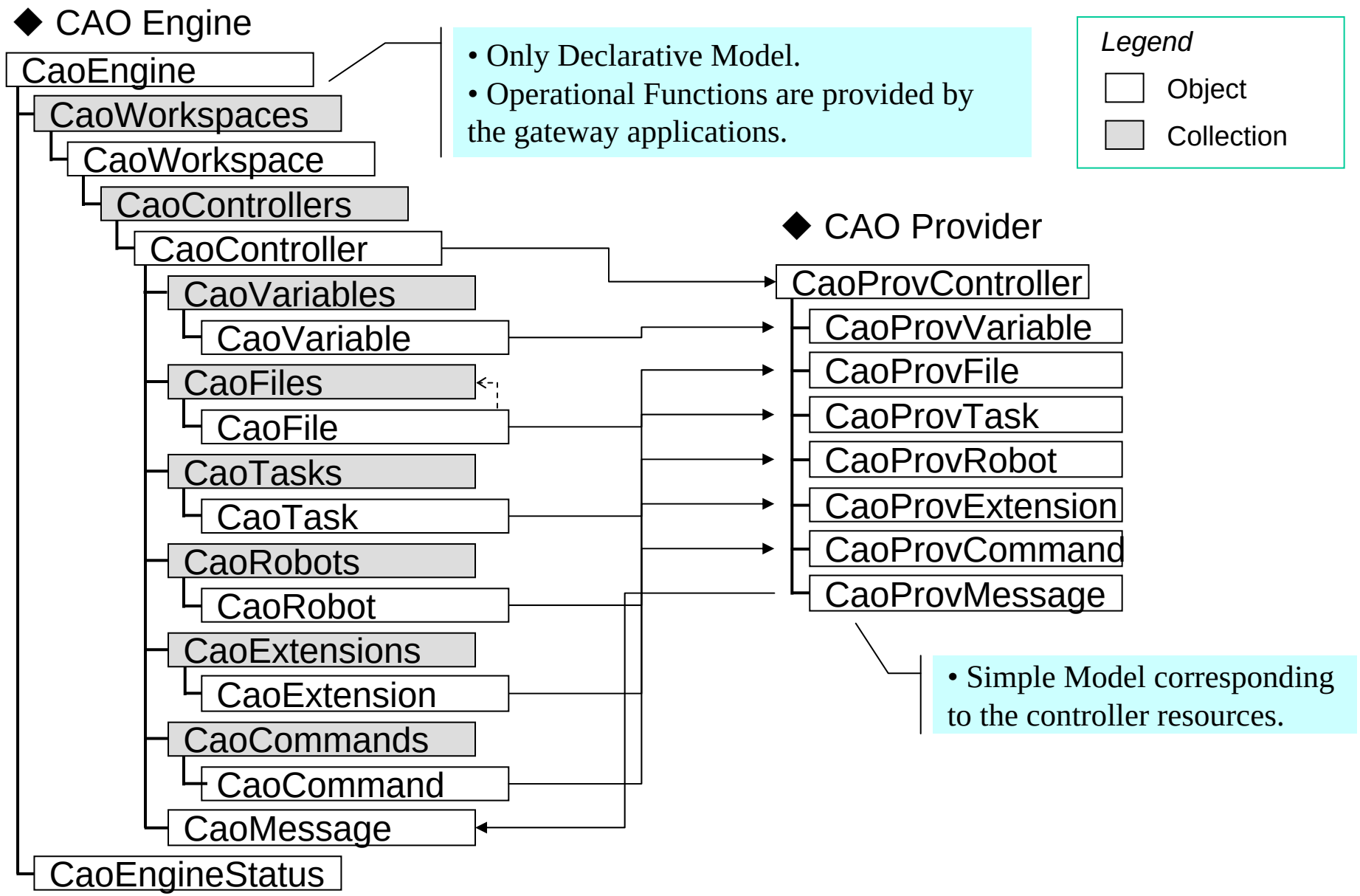
Volatile Resource

Two types of device:
Pull (Store) and Push (Event)

Pull type

Push type

CAO Object Model



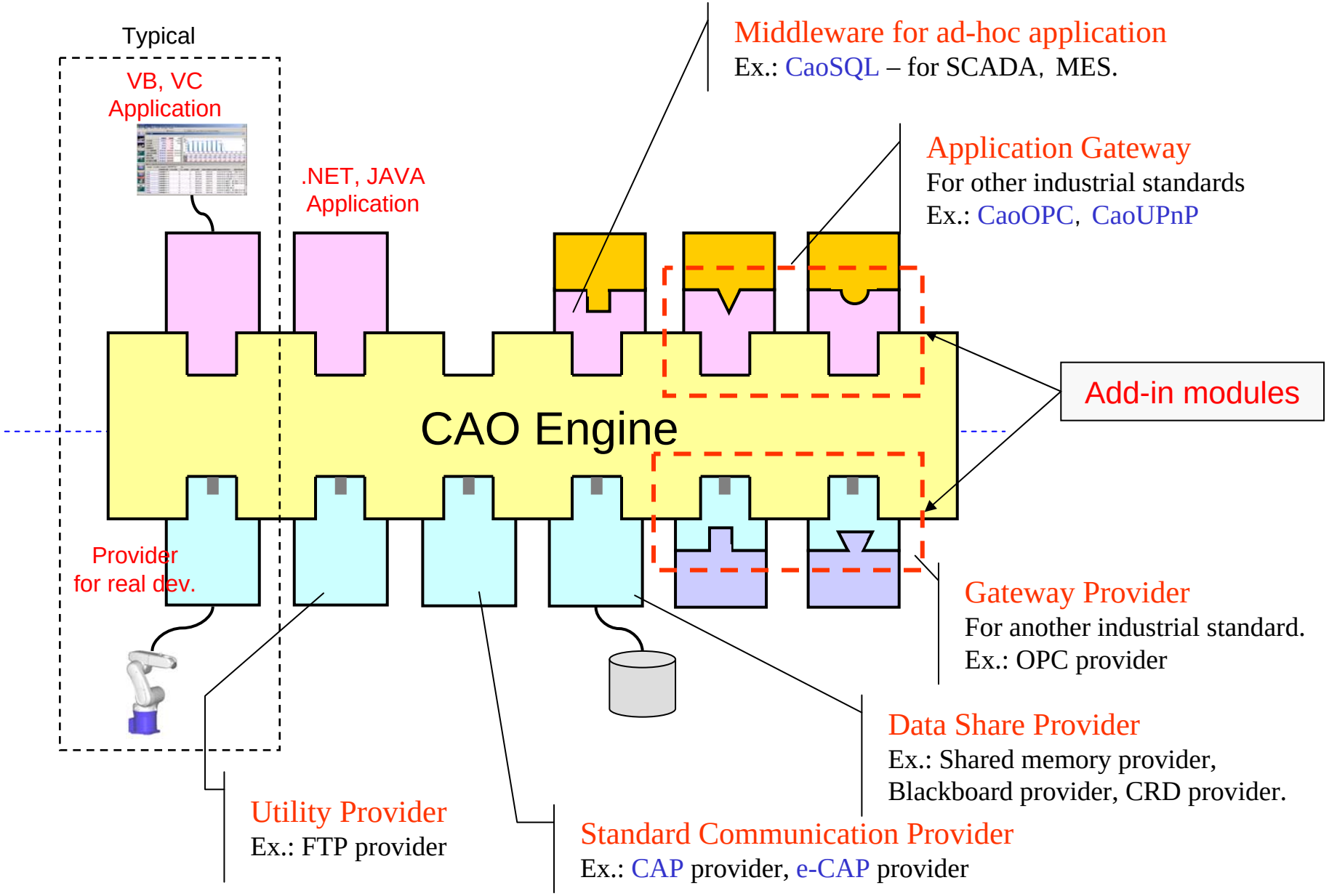
Common Functions of CAO Engine

CAO Engine provides a general common function without depending on the device.

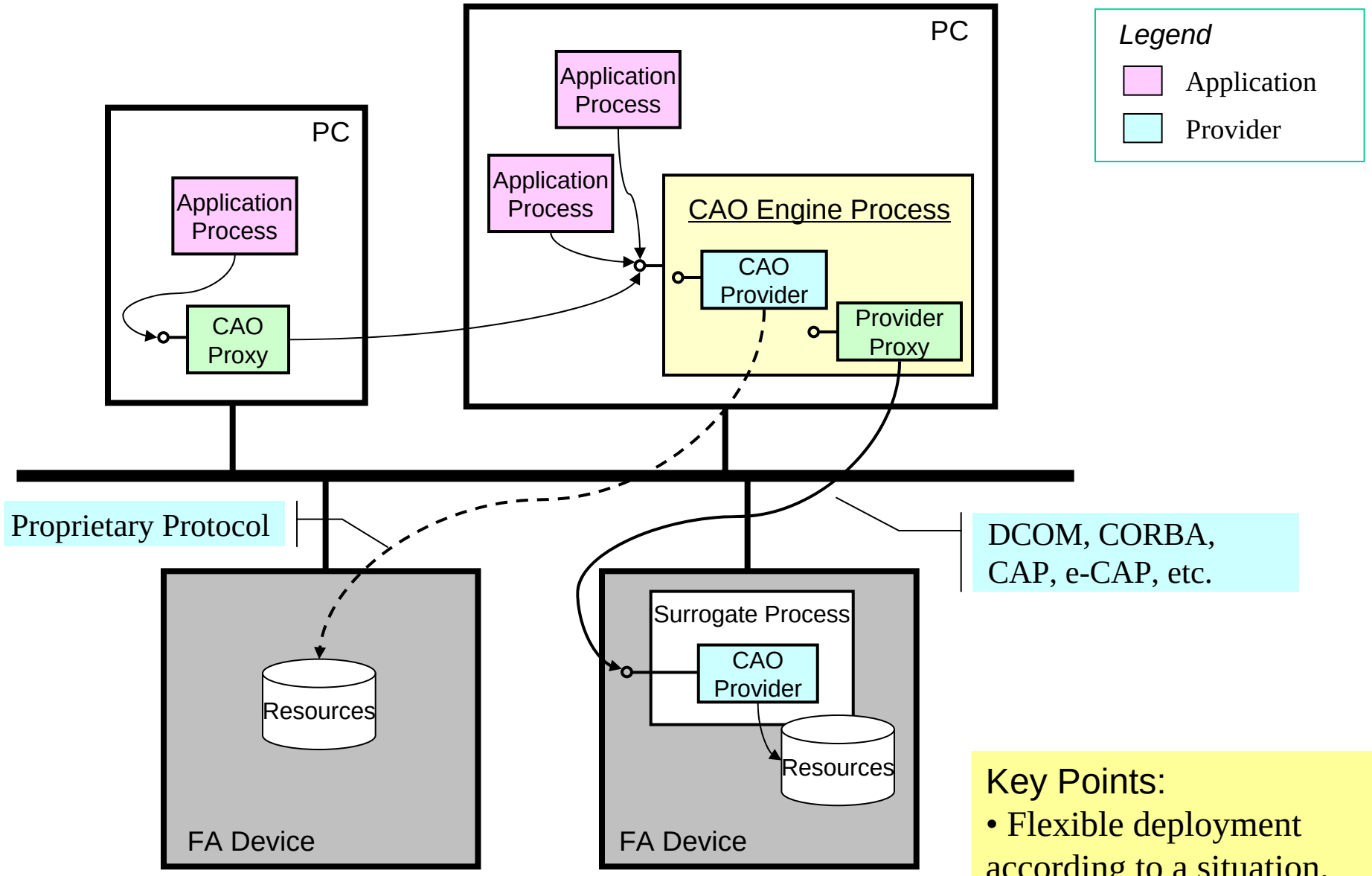
It is unnecessary for makers to develop similar functions.

- Provider Management
- Object Collection Management
- Asynchronous Call (Event) and Message Pool →
- Pre-registered Object Collection（CAO model） →
- Expandable interface by Dynamic Binding mechanism →
- Automatic information merge by a CRD file. etc.

Classification of the ORiN Modules



CAO Module Deployment



Key Points:
- Flexible deployment according to a situation.

Summary

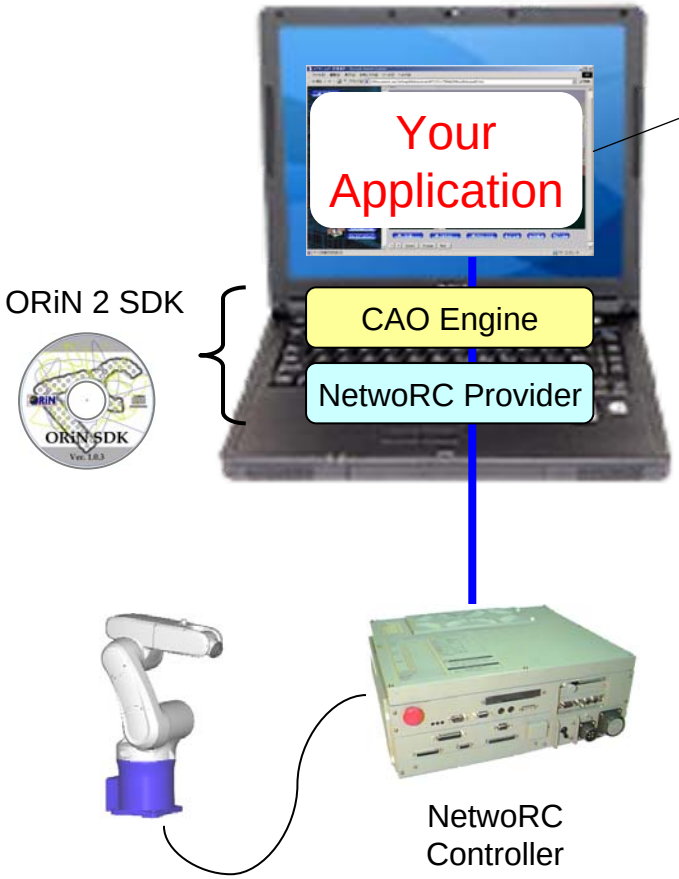
	ORiN 1 (2002 Released)	ORiN 2 (2005 Release)
Operating System	Windows	Windows, Linux - CORBA is independent from an OS.
Programming Language	*Application* : Visual Basic, C++, etc. *Engine, Provider* : C++	*Application* : Visual Basic, C++, VB.NET, C#, JAVA, VBScript, Delphi, etc. *Engine, Provider* : C++
Communication Protocol	DCOM, SOAP (RAP)	DCOM, CORBA, SOAP (CAP), HTTP (e-CAP), UPnP, TCP (RAC)
Target Device	High Function Controller (ex. Robot, PLC, NC)	High Function and Low Function Controller (ex. DA/AD, DIO board)
Target Field	FA	FA, HA, ROBOT
Device Profile	XML Schema (RRD)	XML Schema (CRD) UPnP Device Description
Framework	1. 'Engine' provides the abstracted device. 2. 'Application' operate the abstracted device. 3. 'Provider' absorbs the differences between the abstracted device and real one.	(same as ORiN1)
Interface	Application Interface and Device Interface.	(same as ORiN1)

Summary (cont.)

	ORiN 1 (2002 Released)	ORiN 2 (2005 Release)
Development Support	• ORiN 1 SDK (including Samples, Documents, Simple Tester, etc.) • 'Provider Wizard' automatically generates the C++ provider template.	• ORiN 2 SDK (including Samples, Documents, Integrated Tester, etc.) • 'Provider Wizard' automatically generates the C++ provider template.
Configuration Tool	RaoConfig : Security setting, etc.	CaoConfig : Security and Locale setting, etc.
Development Environment	Available on the market. (ex. Visual Studio, Eclipse, etc.)	(same as ORiN1)
Applications	• Operation Monitoring System • Cylinder Deterioration Monitoring • Rotary-Rack Maintenance System • Error Notification System • Program File Manager • Manufacturing Portal Site, etc.	(In addition to ORiN1) • Distributed Simulation for Manufacturing • ORiN based Cell Controller, etc.
Notes	Over 200 FA devices are connected by ORiN at the real factory, and above-mentioned applications are running for 24 hours.	ORiN 2 specification will be proposed to ISO.

3. DENSO ORiN SDK

Sample Client Program

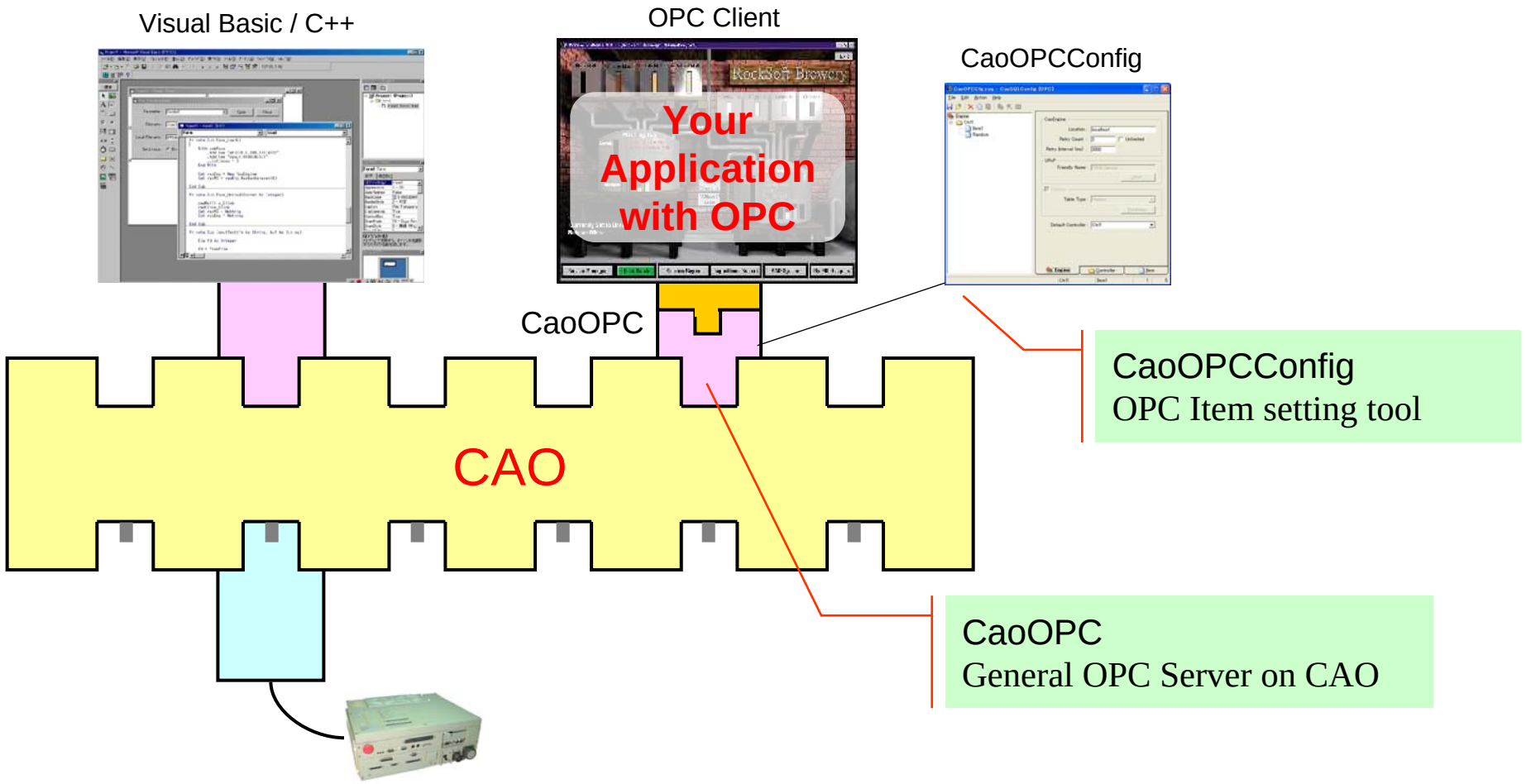



```
Visual Basic Sample:

‘ create CAO objects
Set caoEng = new CaoEngine.
Set caoWS = caoEng.Workspaces(0)
Set caoCtrl =
caoWS.AddController(“RC5”, ”CaoProv.DENSO.NetwoRC”, ””,
”Conn=eth:192.168.0.1”)

‘ set an I/O value
Set caoVar = caoCtrl.AddVariable(“IO128”)
caoVar.Value = 1              ‘ ON

‘ get the I/O value
Msgbox caoVar.Value

‘ get the current robot angle
Set caoRob = caoCtrl.AddRobot(“Arm1”)
Set caoVar2 = caoRob.AddVarialbe(“@CURRENT_JOINT”)
Msgbox caoVar2.Value(0)  ‘ show an angle of J1

‘ move
caoRob.Move 1, “P1”        ‘ PTP, move to P1
```

Program = Algorithm + Data; ORiN targets the algorithm.

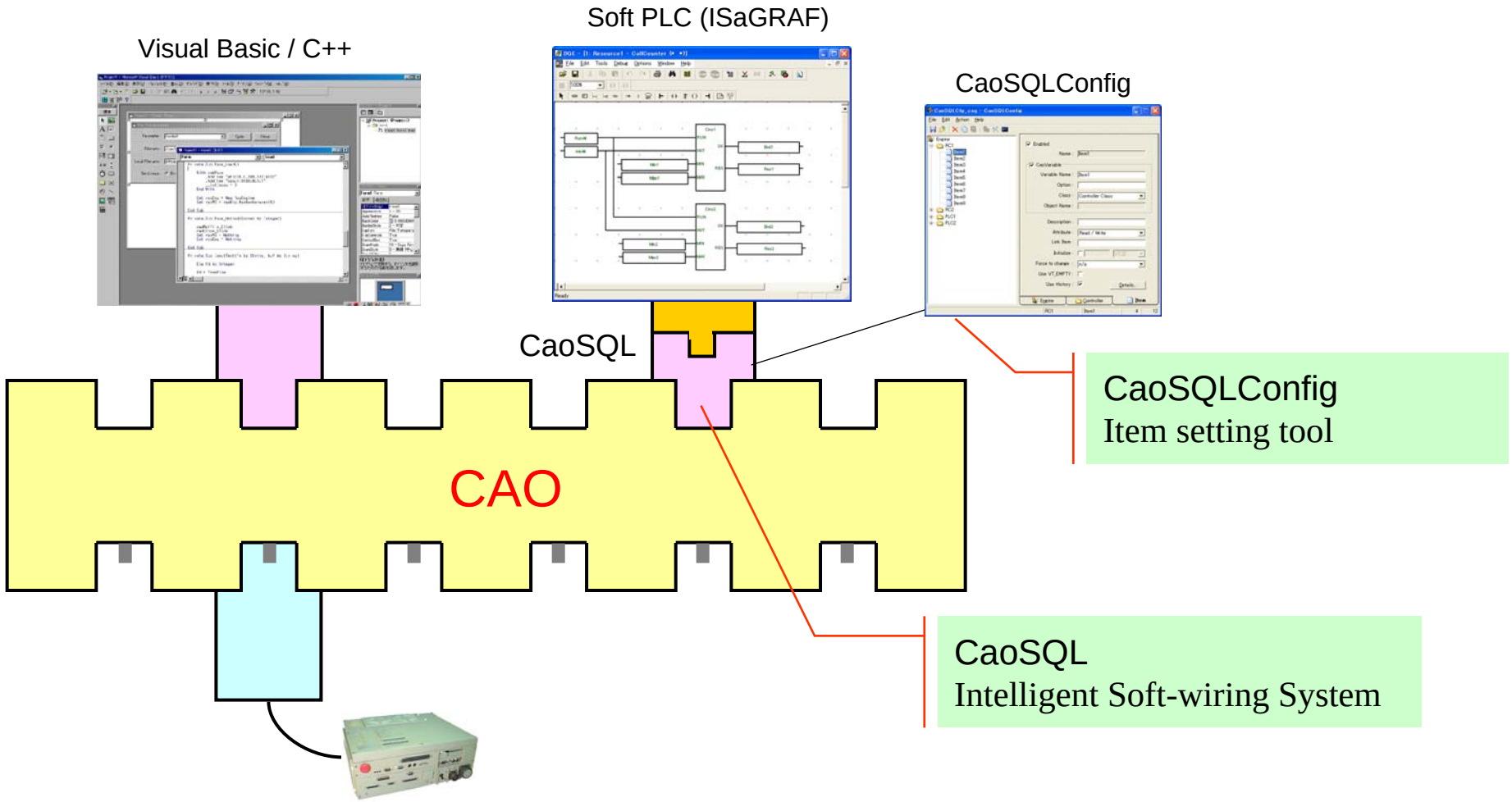
OPC Gateway

◆ Many OPC client applications which are available on the market can be used with ORiN2.
Ex. : Rockwell RSView32.



Soft PLC Gateway

◆ Software PLC also can be used with ORiN.
Ex. : ICS Triplex ISaGRAF



(NOTE) This is one of sample applications of CaoSQL, and now Soft-PLC is in the experimental stage.

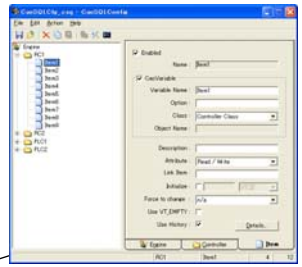
Simple ASCII Text Communication

◆ Commercial Simulation Software
Ex. : ROBOCAD, I-GRIP

Background:
Many traditional simulation software which is ported from UNIX have NOT rich external communication function. However, most simulator has a TCP socket communication function. By using Robot Access Command, RAC, a communication between various devices and simulators is achieved easily.

RAC String
(TCP Socket)

CaoSQLConfig



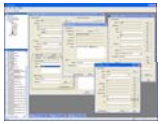
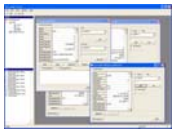
RAC Example:
[request] “GET:RC5::IO128”
[reply] “0, 11, -1”
‘ 0:OK, 11:BOOL, -1:TRUE

CaoSQL

CAO

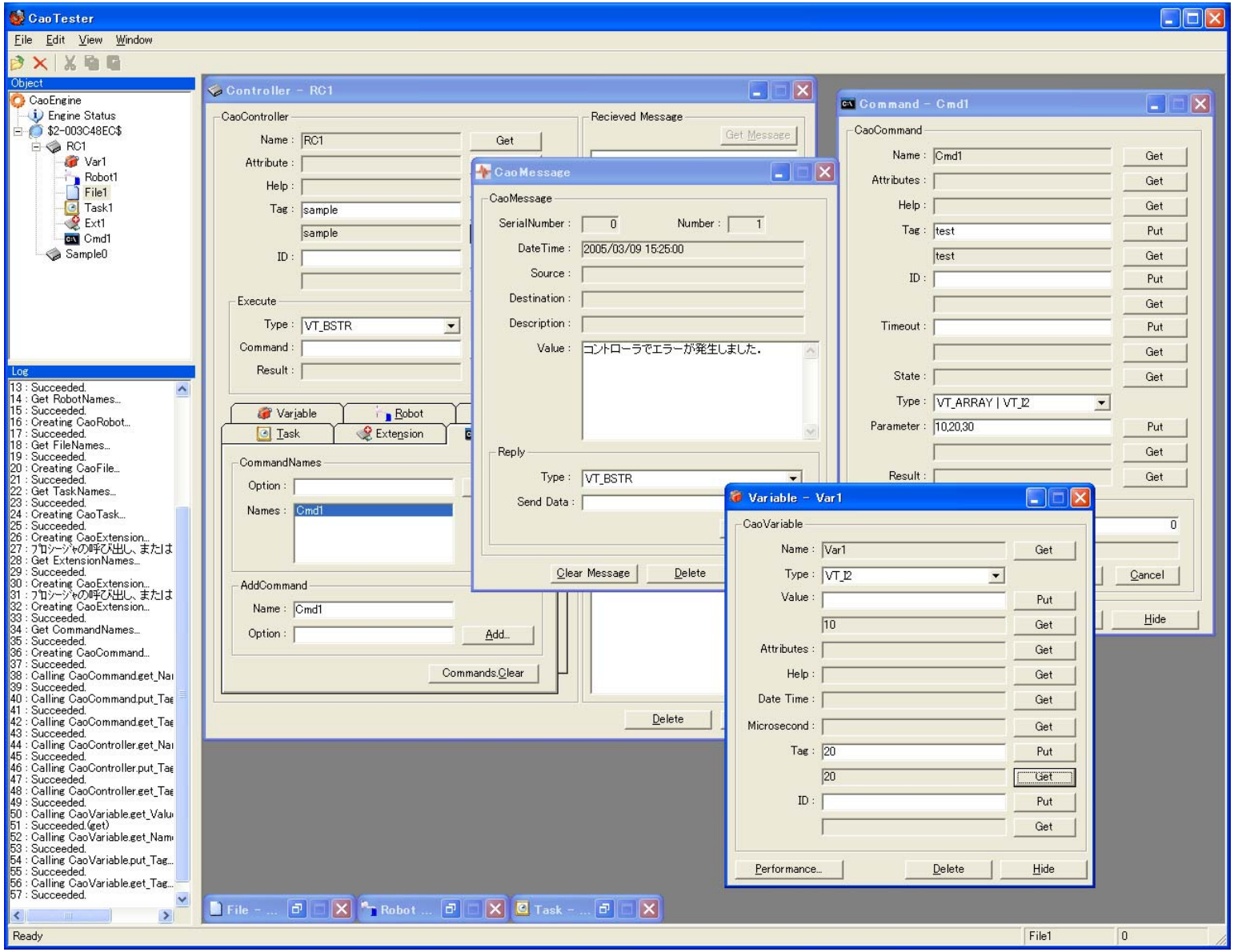
CaoSQL has a RAC service function.
RAC is a simple ASCII string.

Tool List

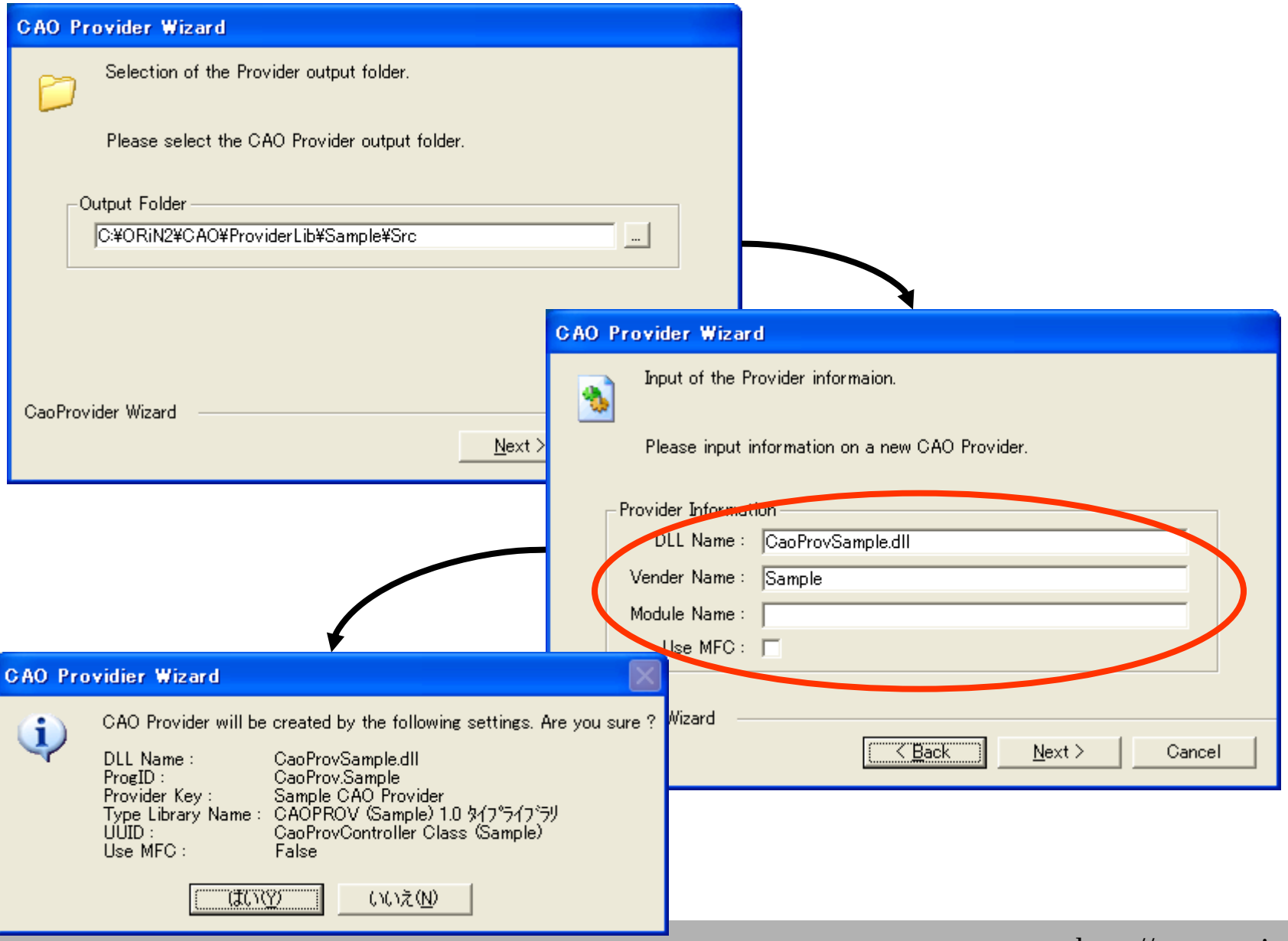
Filename	Snapshot	Function	Directory
CaoConfig.exe		Configuration tool for CAO Engine and Provider.	ORiN2¥CAO¥Tool¥CaoConfig¥Bin
CaoTester.exe		CAO Provider test tool.	ORiN2¥CAO¥Tool¥CaoTester¥Bin
CaoProvWiz.exe		Provider Template Wizard.	ORiN2¥CAO¥Provider¥Bin
CaoSQLConfig.exe		Configuration tool for CaoSQL, CaoOPC and CaoUPnP.	ORiN2¥CaoSQL¥Bin
CaoSQLTester.exe		CaoSQL test tool.	ORiN2¥CaoSQL¥Bin

And, many tiny tools and samples are included in each provider directory.

CAO Test Tool



CAO Provider Wizard



Thank you for your attention.