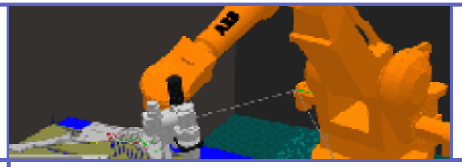
Goran Mustapic
Industrial PhD Student
ABB Robotics, Sweden
Mälardalen University, Sweden
goran.mustapic@se.abb.com

A Dependable Real-Time Platform for Industrial Robotics

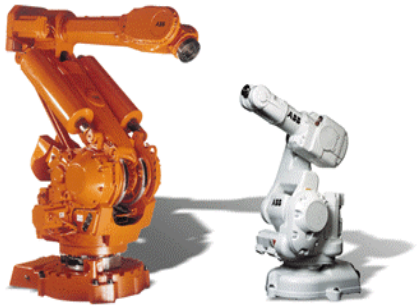
ICSE 2003 WADS



johan.x.andersson@se.abb.com
christer.norstrom@se.abb.com

Outline of the presentation

- A quick look at some industrial robots
- Dependability attributes relevance for Industrial Robots
- System architecture
- ABB Robot Controller architecture



- What do we mean by an open architecture?
- What are challenges in defining an open architecture?
- Initial work on open architecture
- Summary

Dependability attributes relevance for Industrial Robots

- Requirements
  - 60.000 hours MTBF
- Dependability attributes
  - *Availability*
  - *Reliability*
  - *Safety*
  - *Maintainability*
  - *Integrity*
  - *Confidentiality*

System architecture

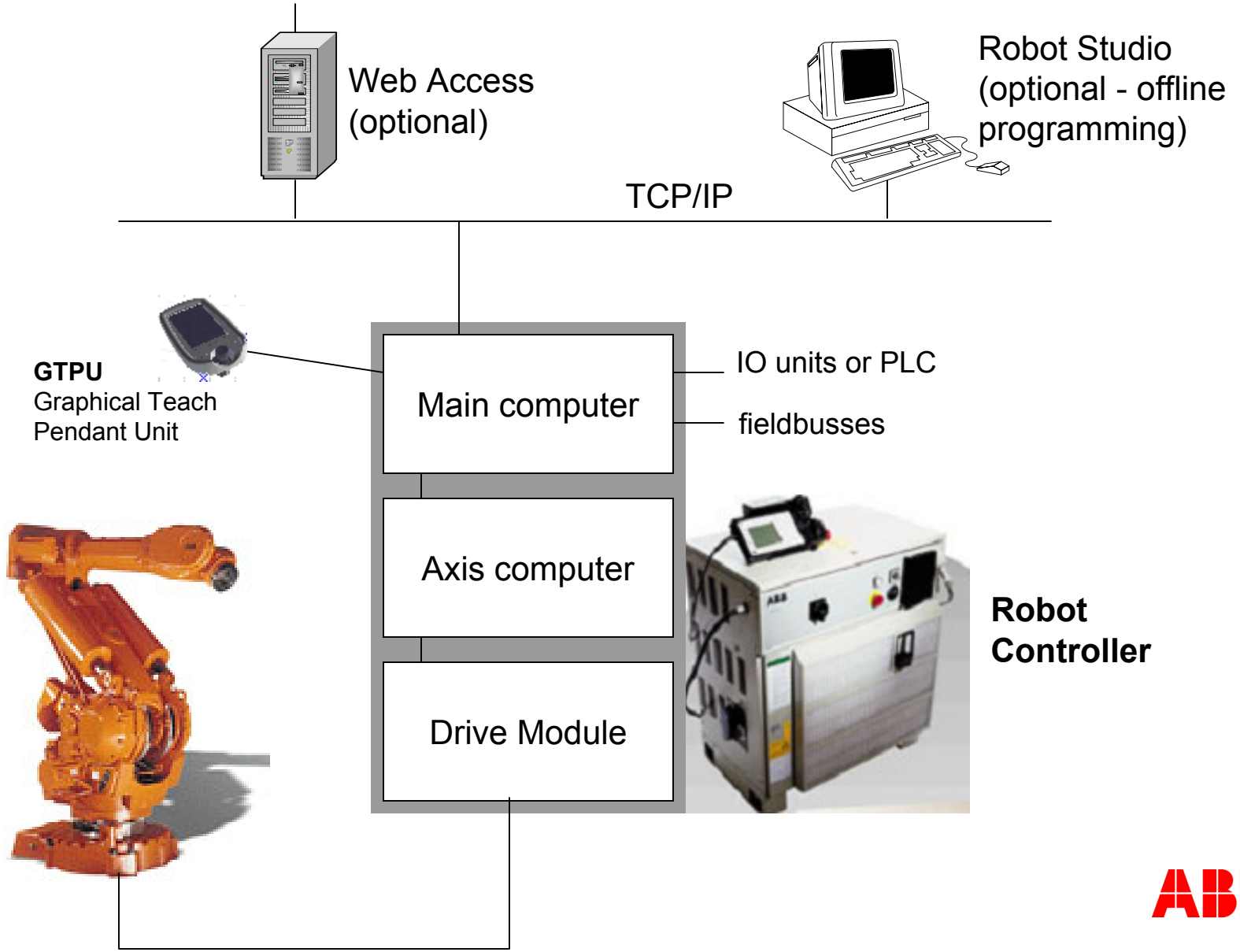


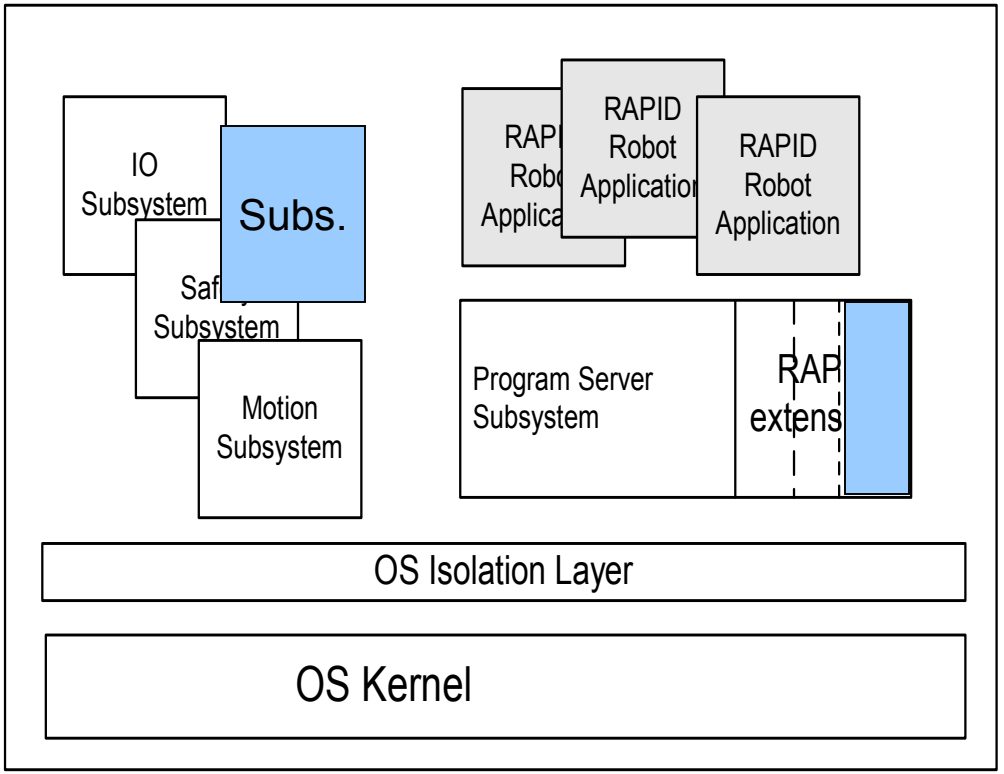
ABB Robot Controller Architecture

- Initial design in the beginning of 1990-ties; first release 1994
- OO design, C language implementation
- Product line architecture
- 2 500 KLOC ~ 500 components in 15 subsystems
- Portable: VxWorks, UNIX, W2K, win9x, WindowsCE
- RAPID – a language for robot programming

- More than 50 000 units on the market
- Users >10000
- Developers >150
- Used in a variety of application fields such as those for car manufacturing, foundry, painting and food packaging.

What do we mean by “open architecture”?

- V. Issarny: “*In an open systems, components do not depend on a single administrative domain and are not known at design time*.”
- Goal of our work is a domain specific open dependable platform
- Example



What are challenges in defining an open architecture?

1. **Business model**
   - How to work with subcontractors, platform, integrators and end customers in an open system?
2. **Organizational**
   - What is an optimal organization to work with 3rd party?
3. **Functionality**
   - What kind of functionality add-ons will the open system allow?
4. **Technical**
   - …

Technical challenges

- *Dependable Platform Architecture*
  - What platform interfaces shall we create?
  - Assuring platform dependability.
  - Define (component) model for adding 3rd party SW.
- *Support for the Development of Extensions*
  - SDK and tools for development of extensions.
- *Certification of platform extensions*
  - Defining criteria for certification.
  - Implementing tools for certification.
- *Predictable Assembly of the platform and extensions*
  - How will properties of extensions be measured and described?
  - Verifying the assembly properties.

Initial work on open dependable architecture

- Related research areas
  - Dependability
  - Software Architecture and CBSE
  - Software Testing
- Open architecture of GTPU (Graphical Teach Pendant Unit)
  - Based on .Net Compact Framework and Windows CE
- Probabilistic simulation-based analysis
  - Enable early reasoning about architectural properties
  - Model the platform
  - Add models of components/extensions to the platform model
- Facilitating fault removal
  - “Black-box” functionality for real-time systems

Summary

- Technical issues to be solved:
  - *Dependable Platform Architecture*
  - *Support for the Development of Extensions*
  - *Certification of platform extensions*
  - *Predictable Assembly of the platform and extensions*
- We need to combine research experiences and results from multiple research areas
- We can still benefit from the technical aspects without ever implementing other aspects

ABB

Probabilistic simulation-based analysis