GLOBAL CONDITION MONITORING SYSTEM
Implementing MATLAB®-Based Analysis Services

Henri Helanterä
Tampere University of Technology
Institute of Automation and Control
Email: henri.helantera@tut.fi
WHY GLOBAL CONDITION MONITORING?

• Increasing productivity and availability of production equipment in process industry requires **proactive** maintenance.

• Gathering data globally and analyzing it gives a better understanding of the behavior of devices.
  ➔ Better accuracy and reliability of condition monitoring and fault diagnostics.

• Individual factories do not have resources to accomplish this.
  ➔ Automation system manufacturers are becoming service providers.
GLOBAL CONDITION MONITORING SYSTEM
- STRUCTURE AND OPERATIONS

• Distributed agent-architecture with a messaging core (such as JMS).
• Field devices perform low-level self-diagnostics.
• Local analysis agent integrates information of the device database and knowledge base.
• Global supervisory logic further processes the information from different sites.

Data ➔ Information ➔ Knowledge
GLOBAL ANALYSIS SCHEME

Field devices perform low-level (self)diagnostics
JAVA – MATLAB® INTEGRATION

• Condition monitoring system is based on Java framework.

• MATLAB® is de facto standard in numerical computing but difficult to be integrated with a Java-based system.

• Integration techniques available
  – MATLAB® engine functions (C/FORTRAN)
  – Stand-alone COM components
  – Inverted call
ENGINE FUNCTIONS AND COM COMPONENTS

- C-engine functions provide access to MATLAB® computing engine.
  - Can be used from Java as native methods.
    - Platform-independence is lost.
- MATLAB® programs can be converted into stand-alone COM components.
  - Libraries are included in the component.
  - Some bridge is needed between Java and COM (JIntegra is one option).
  - Using COM is restricted to MS Windows.
INVERTED CALL

The analysis agent sends a request.

The server starts a new thread for each request and reply.

The client and the computing server exchange information through an auxiliary class.

MATLAB is started to perform the default command.

The default command listens to analysis requests through the GetCommand class.

= SOCKET SERVER CLASS  = SOCKET CLIENT CLASS  = STARTER CLASS
CONCLUSIONS

• Global system enables reliable, efficient and proactive condition monitoring and fault diagnosis.

• Distributed agent-architecture
  - enables automated operations,
  - avoids unnecessary data transfers,
  - makes the system less error prone.

• Java – MATLAB® integration
  - MATLAB® allows the use of advanced computing methods for analyses.
  - Same tool can be used in both development and implementation of analyses.
  - Inverted call is platform-independent.