

A standardized software platform for
equipment control software:
Does this work for everyone?

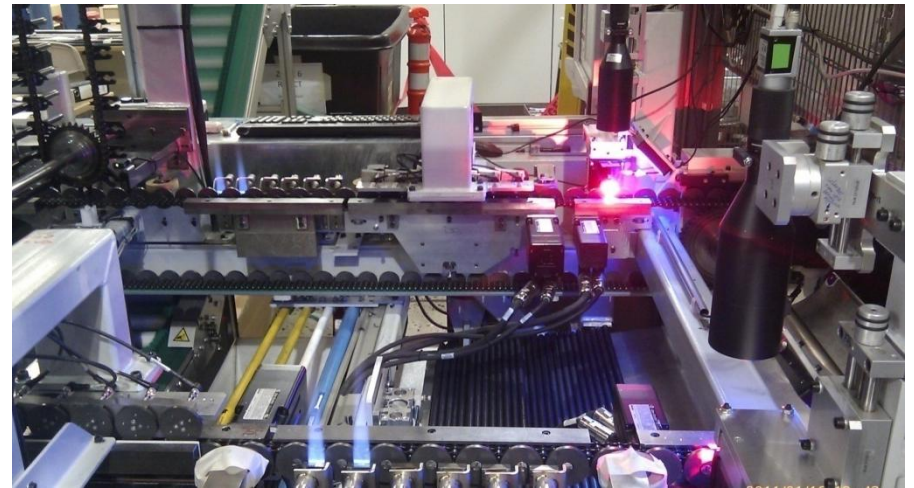
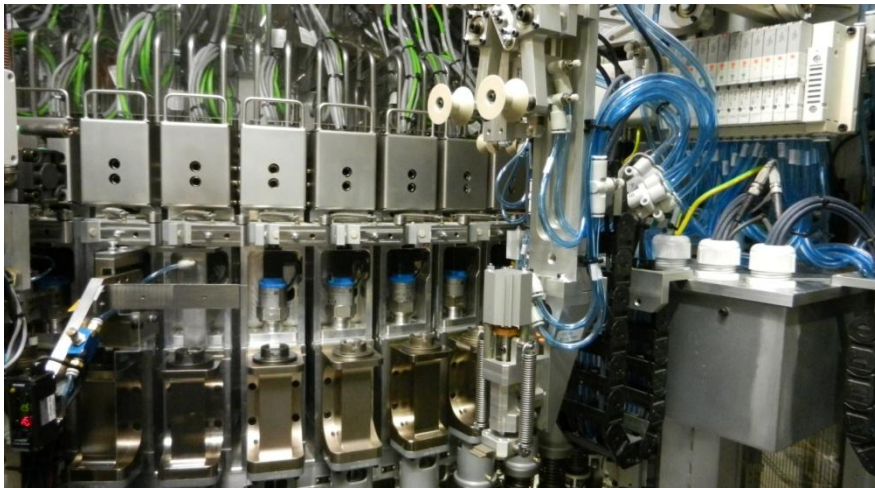
Benno Beuting, Cordis automation

Introduction - Cordis automation

- Founded 2000
- Based at: High Tech Campus, Eindhoven
- Expertise: Equipment control software
 - complex instruments
 - equipment
 - industrial automation projects
- Team of mechatronic engineers



Projects



HET
INSTRUMENT
2012



X-PLORE
AUTOMATION

CORDIS
INNOVATIVE CONTROL SOLUTIONS

Faster – Cheaper - Better

- Shorter development time
- Lower development cost
- Excellent robustness & stability
- Less resource dependent
- Easy debugging / troubleshooting
- Controlled development process!!!

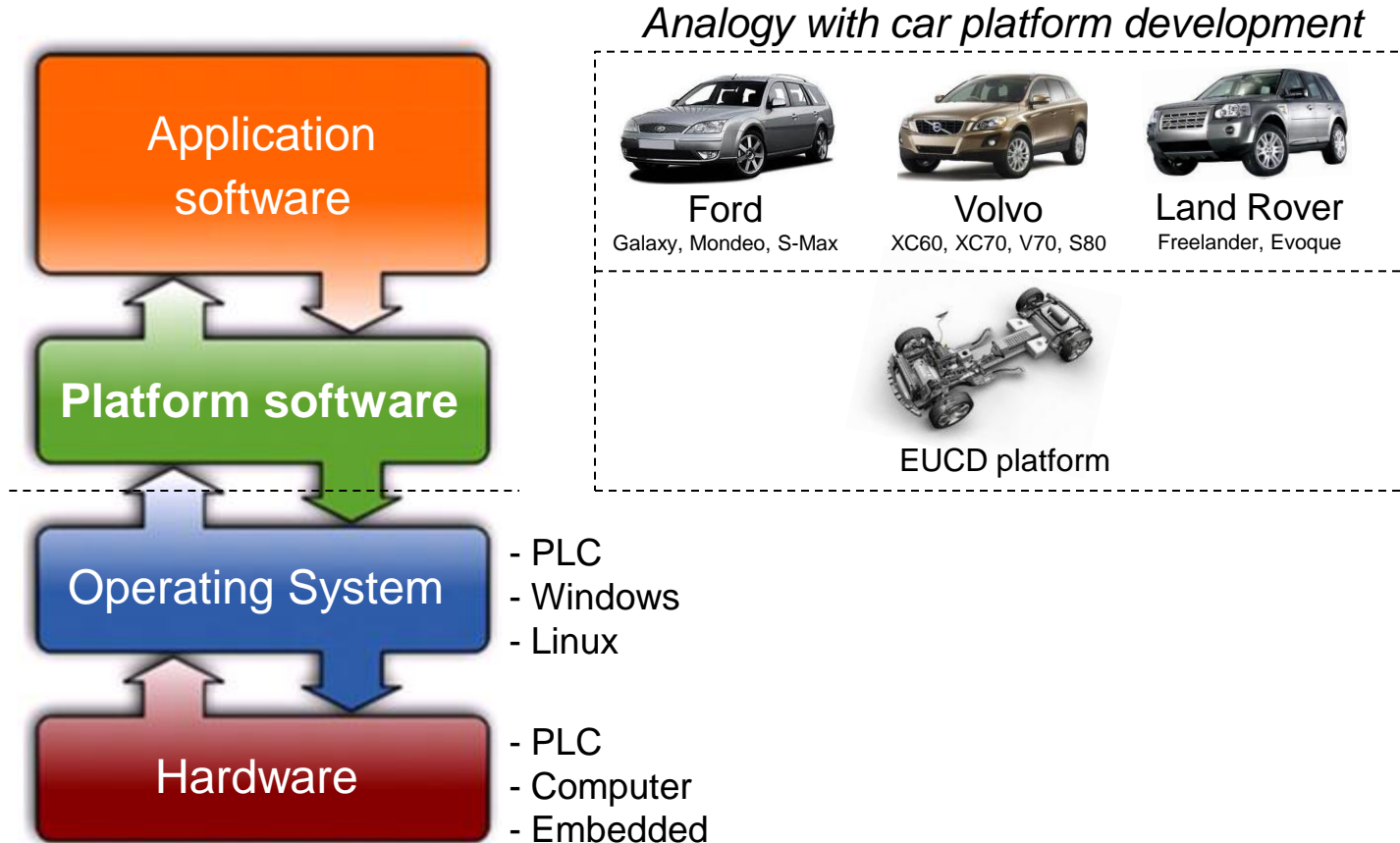


Software creation process

- Design
 - Validation and tuning of the design
- Implementation
 - Fast and fault free construction
- Testing
 - Supporting tools, usable for other disciplines
- Aftercare
 - Diagnostic tools, usable for other disciplines

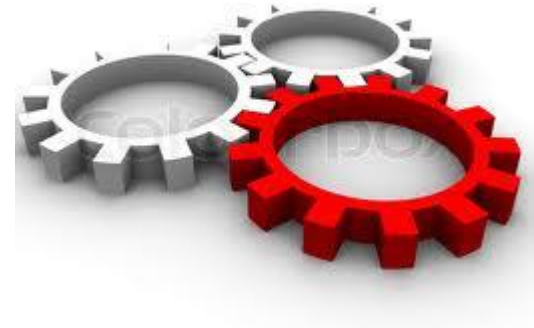


Platform Software



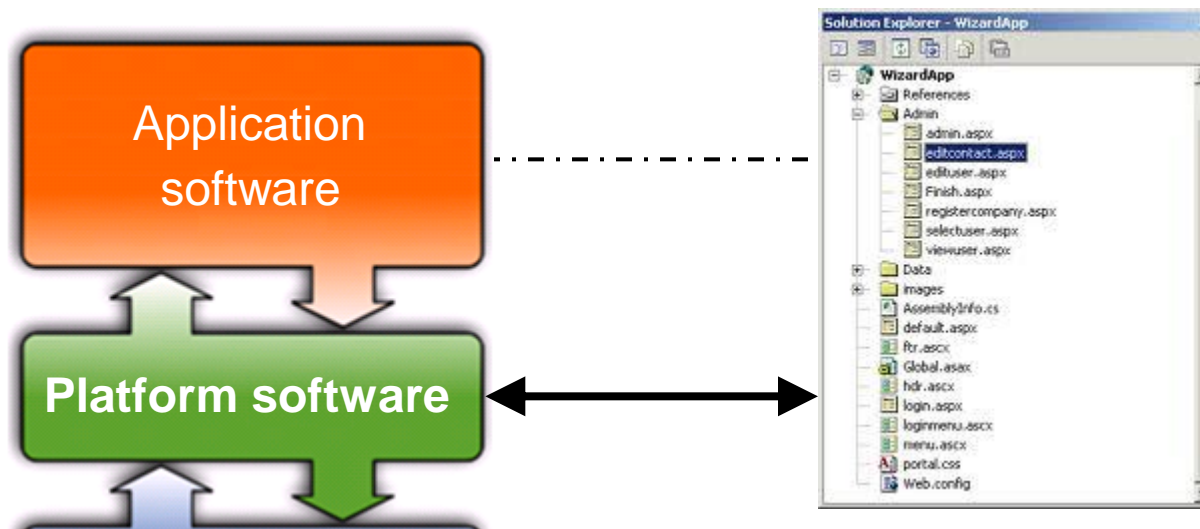
Platform Mechanisms

- Interfacing between architecture components
- Interfacing with User Interfaces (HMI)
- Commands (with arguments)
- State Machine
- Guarding
- Errors, Messages
- Settings, Observers
- Logging history (State Machine, Command, Messages)
- Multi language



Open up the software

- Like Microsoft Explorer




Software Explorer




Platform Explorer

18-9-2012 14:36:55

Platform Explorer - Connected to Application



CORDIS
INNOVATIVE CONTROL SOLUTIONS
www.cordis.nl

 About
 Help
 Disconnect

Machine Parts

- Machine
 - SupplyModule
 - TrayHandling
 - InfeedPosition
 - OutfeedPosition
 - TrayTransfer
 - Servo
 - Gripper
 - ProductTransfer
 - VerticalTransport
 - PickAndPlaceRobot
 - CuttingModule
 - Dummy
 - MessageLogging

Commands

- Reset Error
- Manufacturing
- Maintenance
- Emergency stop
- Enable servo
- Disable servo
- Home
- Move Absolute
- Move Relative
- Stop
- Quick Stop
- Jog Forward
- Jog Backward

Arguments

```


```

Settings

Command Time-out : 0 [ms]
 Acceleration : 12 [mm/s²]
 Deceleration : 100 [mm/s²]
 Jerk : 10 [mm/s³]
 Proportional gain : 25 [%]
 Integral gain : 10 [%]
 Derivative gain : 10 [%]
 Max Velocity : 100 [mm/s]
 Max Position Error : 1 [mm]
 Soft Limit forward : 3610 [mm]
 Soft Limit backward : -10 [mm]




Observers





Position actual: 3605.026 [mm]
 Velocity actual: 32.60007 [mm/s]
 Acceleration actual: 0 [mm/s²]
 Jerk actual: 0 [mm/s³]
 Setpoint Position: 3605 [mm]
 Position Error: 0.026 [mm]
 Enabled: TRUE
 Homed: TRUE
 Controller Error code: 0 []
 Home switch: FALSE
 Limit switch forward: TRUE
 Limit switch backward: TRUE

Statemachines

State	StateName	SubState	RunTime	EntryTime
20	In Position	1	116.530	09-18-2012 14:34:...
13	Move Absolute	1	2.050	09-18-2012 14:34:...
21	In Home Position	1	5.730	09-18-2012 14:34:...
12	Goto Home	1	1.040	09-18-2012 14:34:...
10	Enabled	1	2.960	09-18-2012 14:34:...

18-9-2012 12:58:10 | WARNING | CuttingModule | Waste Bin almost full

 Stop
 History
 Reset

 Explorer
 IO
 Start
 Stop

Design - common situation

- Mechanical discipline
 - System architect
 - Working out the mechanization solution
 - Late transfer to Software discipline
- Software discipline
 - Working out the detailed operation of system
 - Determine system exceptions

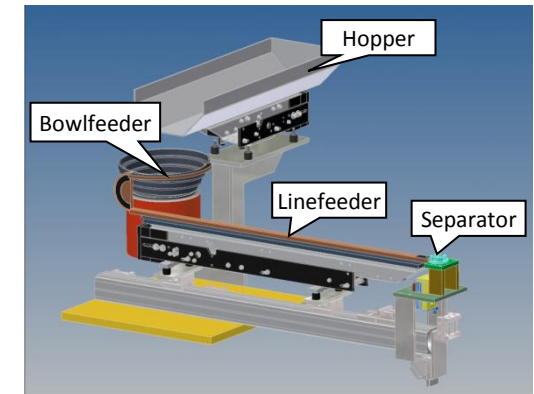
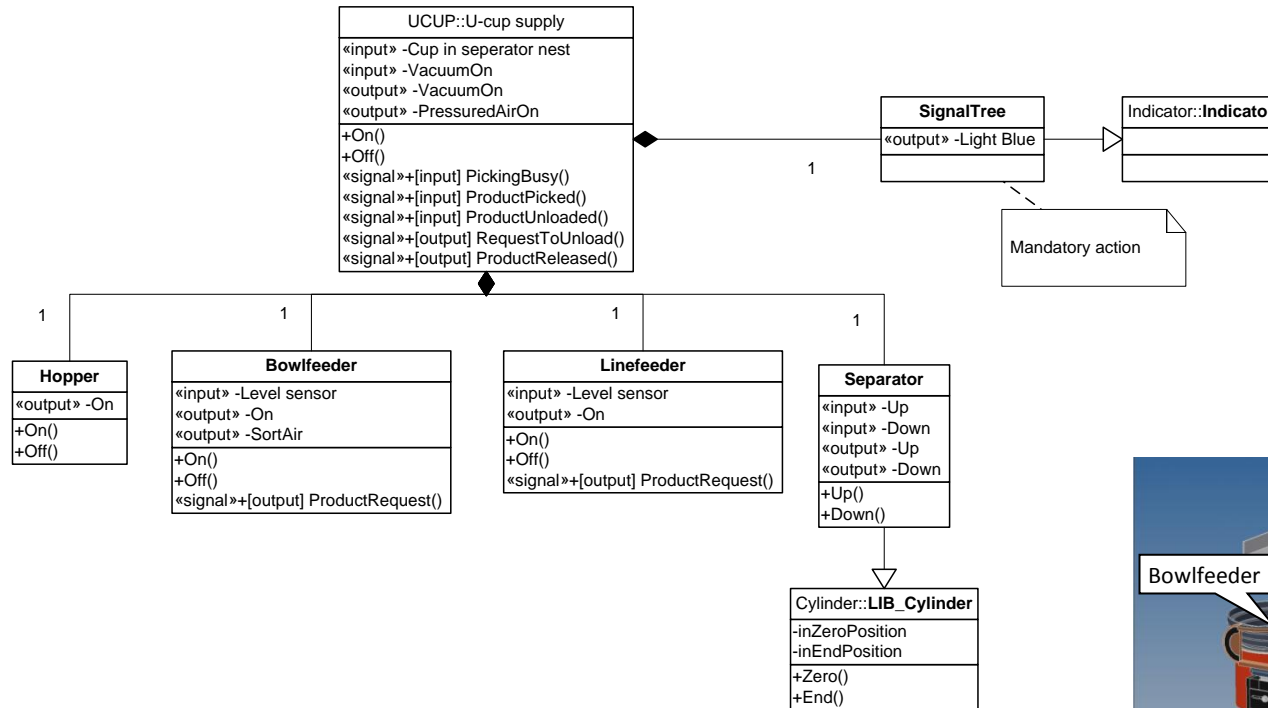


Design – get it right

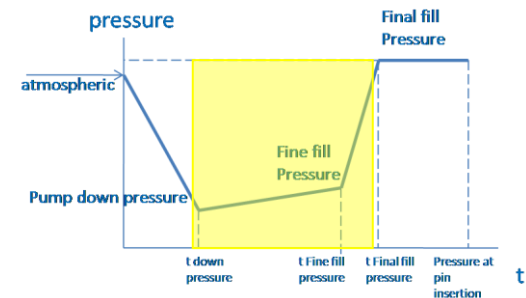
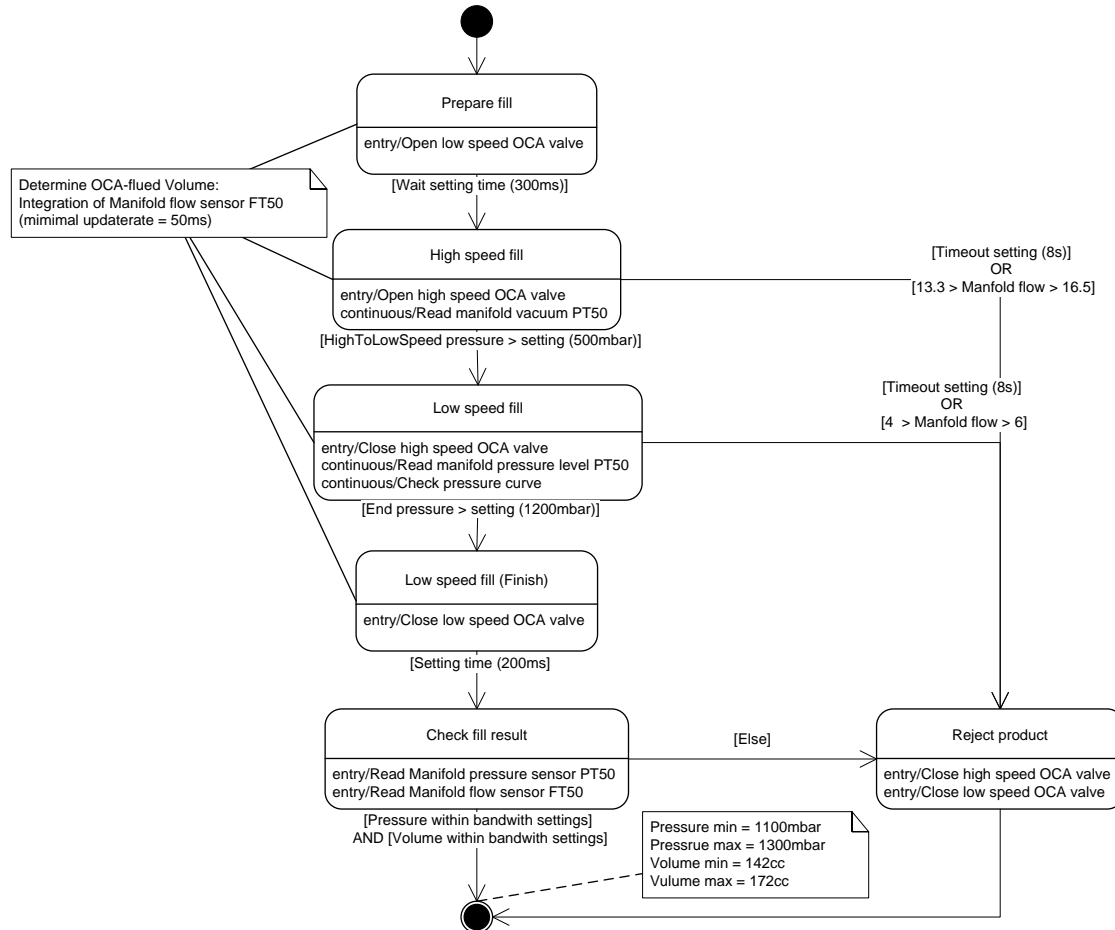
- Faults in design have major impact
- Involve all disciplines
- Make Multidisciplinary validation possible
 - So, no ‘pure software’ description
- Graphical description (UML)
 - State diagrams (Flow charts)
- Use together the same system description



Design – static view



Design – dynamic view



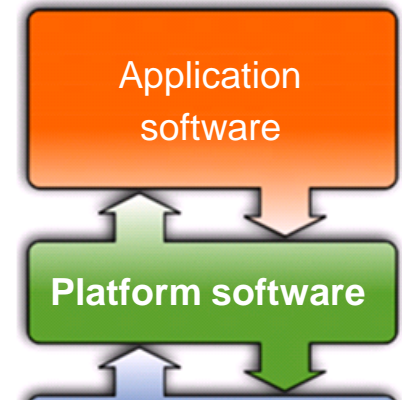
Implementation common situation

- Poor standardization
- Resource dependent / critical
- Poor reuse
- Stripping of previous project
- Standard (library) components



Implementation Fast and Fault-free

- Reuse of a generic software base
 - Platform software (*off-the-shelf*)
- Code template
 - Standardization application components
- Code generation tool
 - Rapid development
 - Reduce code typing mistakes



Testing - common situation

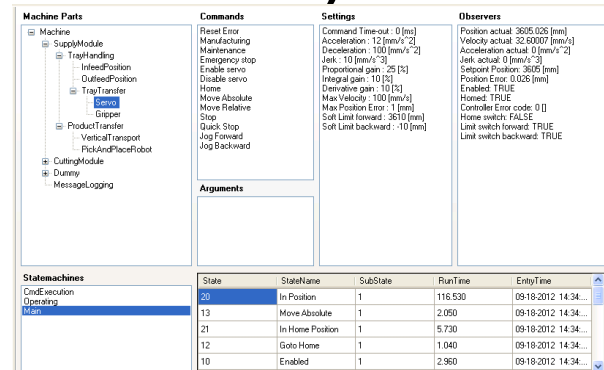
- Heavily involved supporting other disciplines
 - I/O testing
 - Mechanical (module) testing
 - Process testing
 - Diagnose System and Process problems
- Tools
 - Software Development Kit
 - Dedicated HMI service functions



Testing - be prepared

- Unpredictable planning, flexibility required
- Provide other disciplines supporting tools
- Software explorer tool (*off-the-shelf*)

- opening-up the software
- supporting test functions
- diagnostic information
- also usable for other disciplines



The screenshot displays a software interface with several panels. On the left, a tree view under 'Machine Parts' shows a hierarchy including SupplyModule, TrayLanding, InfeedPosition, DottedPosition, TrayTransfer, ProductTransfer, VerticalTransport, Gripper, PickAndPlaceRobot, CuttingModule, Dummy, and MessageLogging. The 'Commands' panel lists actions like Reset Error, Manufacturing Maintenance, Emergency stop, Enable servo, Disable servo, Home, Move Absolute, Move Relative, Stop, Quick Stop, Jog Forward, and Jog Backward. The 'Settings' panel contains numerical values for parameters such as Command Time-out, Acceleration, Deceleration, Jerk, Proportional gain, Integral gain, Derivative gain, Max Velocity, Max Position Error, Soft Limit forward, Soft Limit backward, and Homed. The 'Observers' panel shows real-time data for Position actual, Velocity actual, Acceleration actual, Jerk actual, Setpoint Position, Position Error, Homed, Controller Error code, Home switch, Limit switch forward, and Limit switch backward. At the bottom, a 'Statesmachines' table provides a summary of the current state.

State	StateName	SubState	RunTime	EntryTime
20	In Position	1	116.530	09-18-2012 14:34:...
13	Move Absolute	1	2.050	09-18-2012 14:34:...
21	In Home Position	1	5.730	09-18-2012 14:34:...
12	Goto Home	1	1.040	09-18-2012 14:34:...
10	Enabled	1	2.360	09-18-2012 14:34:...



Aftercare - common situation

- Heavily involved support other disciplines
 - Analyzing system problems
 - Analyzing process problems
- Tools
 - Software Development Kit
 - Dedicated HMI service functions



Aftercare - don't be surprised

- Provide other disciplines supporting tools
- Problem diagnoses performed by end-user
- Software explorer tool (*off-the-shelf*)
 - opening-up the software
 - supporting test functions
 - diagnostic information

The screenshot displays a software explorer tool interface with four main panels: Machine Parts, Commands, Settings, and Observers. The Machine Parts panel shows a tree view of the machine structure. The Commands panel lists various control actions. The Settings panel shows parameters for the selected command. The Observers panel displays real-time data for the selected command. Below these panels is a StateMachines table.

StateMachines	State	StateName	SubState	RunTime	EntryTime
CncExecution	20	In Position	1	116.530	09/18/2012 14:34...
Operating	13	Move Absolute	1	2.060	09/18/2012 14:34...
	21	In Home Position	1	5.730	09/18/2012 14:34...
	12	Goto Home	1	1.040	09/18/2012 14:34...
	10	Enabled	1	2.960	09/18/2012 14:34...



Platform approach is a good way forward

- Shorter development time
- Lower development cost
- Excellent robustness & stability
- Less resource dependent
- Easy debugging / troubleshooting
- Controlled development process!!!



Three types of organizations

- They who *make it happen*
- They who *watch it happen*
- They who ask themselves: *What happened?*



Control your software development

Focus on your core business



CORDIS

INNOVATIVE CONTROL SOLUTIONS

benno.beuting@cordis.nl

HET
INSTRUMENT
2012



X-PLORE
AUTOMATION

CORDIS
INNOVATIVE CONTROL SOLUTIONS