Applying CORBA to embedded time-triggered real-time systems

S. Aslam-Mir (Sam)
Principal CORBA Architect – Vertel USA
sam@vertel.com
Synopsis

- Motivation
  Time Triggered vs Event Triggered
- Real-time CORBA present applicability
- Real-time CORBA for the masses MEMs, and the smart transducer interface.
- *Fault-tolerance with real-time capability.
- First order penetration of CORBA into the control plane of mission-critical systems
- Conclusions
Motivation

• For integrated safety-critical real-time systems the time-triggered approach is preferred.
• Safety critical real-time systems prefer functions (in avionics terms) to be kept apart – avoids failure propagation – **Partitioning**
  
  • **Composability desire** – lowers cost to develop a more complex integrated sub-system.
  
  • Partitioning and composability in safety-critical real-time systems determine its predictability (value-time tuple)
  
  • Temporal (time) predictability in the face of failures is **difficult** to achieve in event triggered systems – this is because FDI and Reconfiguration has to occur in a fixed/bounded well known time – thus time-triggered is preferred over event triggered.
Motivation

Achievable at LOW-LOW-COST
What is the new approach

- Chose a generic TTP/C like protocol over several others like Honeywell SAFEbus (777bus), ARINC 629, FlexRay, NASA Spider
- Created a canonical transducer cluster to model the device(s) called **smart transducer** for OMG specification.
- **ORBOS-01-10-02/04**
- ... comprises the integration of one or more MEMS sensor/actuator elements with a microcontroller that provides the following services across standard interfaces:
  - signal conditioning
  - calibration and conversion to standard units
  - diagnostic and maintenance
  - real-time network interface
- The idiosyncrasies of the smart sensor/actuator can be hidden behind a standard CORBA interface neatly.
- In the future, smart transducers will be manufactured in **mixed signal** technology on a single die.
Smart Transducers

- Transducer cluster
- Transducer
- Transducer interface
- Cluster interface
Limitations of present RTCORBA applicability into RT control-loops.

- Application to Time-triggered systems is difficult as it stands for the present
- Support for integrated fault-tolerance and load balancing in one shot does not exist
- Difficult/expensive to achieve well bounded deterministic latency and jitter because of probabilistic models for things like contention resolution on buses like VME and Ethernet
- The above are costly
- *Proving partitioned composable properties of present RTCORBA based systems is costly and difficult if at all.*
Smart Transducer technology - low cost RTCORBA use for the masses

- No noise pickup from long external signal transmission lines.
- Better Diagnostics--Simple external sensor failure modes (e.g., fail-silent, i.e., the sensor operates correctly or does not operate at all).
- "Plug-and-play" capability if the sensor contains its own documentation on silicon.
- Reduction of the complexity at the system hardware and software and the internal sensor failure modes can be hidden from the user by a well-designed fully specified smart sensor interface.
- Cost reduction in installation and maintenance.
The Smart Transducer interface

• An RT-interface is a common boundary between subsystems that allows the timely exchange of *observations* between these subsystems.

• An observation is a *n-tuple* <Name of an RT-entity, time of observation, value of observation>.

• Communication across an RT interface is only possible, if the participating subsystems share a *common set of concepts* concerning
  ♦ Common notion of time and its representation
  ♦ Meaning of the names of RT entities
  ♦ Shared code-space for the representation of values
  ♦ Access protocol to the information.

• A universal smart transducer interfaces must specify this *common knowledge*. 
Conceptual model

All nodes have access to a global time of known precision.

• **Temporal Accuracy**

<table>
<thead>
<tr>
<th>Sensor Data (SD)</th>
<th>Real Time Transport Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Transducer</td>
<td>- Known Delay</td>
</tr>
<tr>
<td></td>
<td>- Minimal Jitter</td>
</tr>
</tbody>
</table>

Different RT Transport Protocols supported

<table>
<thead>
<tr>
<th>RT Image of SD</th>
<th>CORBA Gateway</th>
</tr>
</thead>
</table>
Runtime model

Traditional flow control in uni-directional data transfer

data push

Sender \[\rightarrow\] Control \[\rightarrow\] Receiver

data pull

Sender \[\leftarrow\] Control \[\leftarrow\] Receiver

time-triggered

Sender \[\leftrightarrow\] Receiver
Runtime model

Sensor → IFS → Gateway (CORBA) → ORB

Information Push
Ideal for Sender

Time-Triggered Communication System

Information Pull
Ideal for Receiver
The ST interfaces

• Real-Time (RS) Service Interface-TT:
  – Contains RT observations
  – Time sensitive
  – In control applications periodic

• Diagnostic and Maintenance (DM) Interface-ET
  – Sporadic Access
  – Requires knowledge about internals of a node
  – Not time sensitive

• Configuration Planning (CP) Interface-ET:
  – Used to install a COTS node into a new configuration
  – Not time sensitive

• The ST Submission supports all three of these interfaces.
Smart Transducers

Functional view

MANDATORY

Diagnostic/maintenance interface

Configuration interface

Service interface

End User
Higher Level
Application

Service data
(real-time)
May or may
not be GIOP
based

Encapsulating interface

Transducer Cluster
Structure -

- Cluster A
- Cluster B
- Cluster C
Coupling fault-tolerance with real-time capability . .

- Real-time and Fault-tolerant CORBA cannot be applied to the other side of the Gateway together – why?
- Temporal (time) predictability in the face of failures is difficult to achieve in event triggered systems as such (mathematical literature supports this in addition to empirical research)
- This is because FDI and Reconfiguration has to occur in a fixed/bounded well known time – thus time-triggered is preferred over event triggered.
- TTP/C model chosen for the other side to allow for bounded response in the face of failures that has been proven.
- Literature source at provided at end for comparison and support.
Permeation of RTCORBA into the control plane of mission-critical systems

- Traditional certification hurdles have been a problem with RTCORBA based control loops, thus limited its applicability to life-critical control loops.

- Where low latency, and absolutely minimal jitter are required for hard/brittle real-time control something else is used

- TTP/C married to new forms of GIOP offer a new solution.

- It fuses the substantial fault-tolerant, highly deterministic, overload protection, discovery reconfiguration and isolation advantages of using TTP/C with the interoperability and standards based approach of RTCORBA together.
Conclusions

- TTP/C like protocols plus RTCORBA based GIOP will stimulate the evolution of dynamic closed-loop CORBA for control and automation to much greater levels.
- Smart transducer interface brings real-time time-triggered systems into the CORBA fold without sacrificing any of their strengths.
- TTP/C like protocols offer many advantages in terms of low cost, mass production, and several mathematically provable properties required by mission critical fault-tolerant real-time systems.
- Major adoption in process by standards bodies OMG, TTP Forum.
- Major adoption by industry for TTP/C based systems – **Honeywell in AFCS**, and **Audi in automotive system (in preference to CAN!)**.
- CORBA approach using e*ORB-C/C++ Edition prototype.
- **How much cheaper?** 400$ compared to $15
Useful sites with papers:

- www.tttech.com
- University of Vienna – Professor Kopetz work
- On time-triggered systems UCI Professor Kane Kim’s work in TT systems is a useful second reference point for time-triggered systems.