



Real-time and Embedded Systems Workshop

Reston, VA USA - July 12-15, 2004

Workshop Program

MONDAY – July 12, 2004

TUTORIAL TRACKS

0830 - 1230 ***The Grand Real-time CORBA Tutorial (Part 1)***

Track 1 Shahzad Aslam-Mir, PrismTech Corporation - Doug Jensen, MITRE Corporation –
Chris Gill, Washington University - Irfan Pyarali, OOMWorks

The first part of the Real-time CORBA Tutorial will present: Real-time CORBA 1.0 specification for static fixed priority systems; The impact of the Extensible Transport Framework CORBA specification as it relates to the use of RT CORBA; The Smart Transducer specification for low-cost, highly-deterministic closed-loop real-time control; Detailed code examples and illustrations of the use of priority semantics in CORBA applications from several vertical domains including process control and software defined radio; RTCORBA 1.0 specification interfaces, the priority machinery, the execution models and mechanisms pertinent to allowing for the deterministic execution of real-time CORBA servers; A detailed session on the anatomy of the GIOP protocol, and guidance on writing compact, low footprint, yet deterministic, high performance IDL; A breakdown of how IDL features are mapped into on the wire GIOP traffic will be delivered to give attendees an idea of those features of IDL types that give rise to latency, jitter and general non-determinism; And rules and guidelines on managing real-time CORBA IDL interfaces and how to create ORB vendor neutral application frameworks such as SCA core frameworks.

0830 - 1230 ***MILS Tutorial: High Assurance Security and Safety for Deeply Embedded, Real-time Systems***

Track 2 Mark van Fleet, National Security Agency,
Bill Beckwith, Objective Interface Systems, Inc.

MILS security represents an evolutionary approach to building secure systems with many concepts that depart from and some concepts derived from the older Bell-LaPadula theories on secure systems. The Bell-LaPadula theories represent the foundational theories of the DoD Orange Book. This tutorial will introduce the MILS Security architecture, discuss the efforts by most major RTOS vendors to implement the MILS Security architecture, and provide several examples of applying MILS Security to real-time, embedded systems. MILS Middleware is an architecture for high assurance secure communications. We will provide examples of applying the MILS Middleware architecture to securing real-time communications.

0830 - 1230 ***MDA Distilled: Principles of Model-Driven Architecture***

Track 3 Stephen J. Mellor, Mentor Graphics

Model-driven architecture is an OMG initiative that promises to deliver standards to enable and encourage model-driven development. These standards will make models assets, in contrast to code-driven development where code is an ongoing expense. The keys to this technology are the ability to model systems at a high enough level of abstraction that decisions about implementation technologies can be deferred, and the ability to weave together such models—including models of implementation technologies and code—into a system only when it is ready to be deployed. This vision has been well received in the IT world, spawning a swarm of acronyms and a large number of

UML profiles. However, much of the original work in MDA came from the world of embedded and real-time systems, and, once stripped of unnecessary IT cruft, MDA can be effectively deployed in RT. This tutorial will explain the fundamentals of MDA; what the basic technologies are; how they fit together; what standards are already in place; and what still needs to be done to realize the vision. It will also identify two basic styles of MDA and show how one, based on agile executable models, can slim down MDA for our domain.

1030 - 1045 Morning Refreshments

1230 - 1315 Lunch

TUTORIAL TRACKS

1330 – 1730 ***The Grand Real-time CORBA Tutorial (Part 2)***

Track 1 Shahzad Aslam-Mir, PrismTech Corporation - Doug Jensen, MITRE Corporation - Chris Gill, Washington University - Irfan Pyarali, OOMWorks

The second part of the Real-time CORBA Tutorial will present: Dynamic Scheduling CORBA – often termed Real-time CORBA 2.0; and Discussion of some of the experiences of current generation implementations and possible application uses in the commercial domain of the use of some of Real-time CORBA 2.0.

1330 – 1730 ***Using the Lightweight CORBA Component Model to Develop***

Track 2 ***Distributed Real-time and Embedded Applications (Part 1)***

Douglas C. Schmidt, Vanderbilt University & Frank Pilhofer, Mercury Computer Systems, Inc.

This tutorial will explain the key features and mechanisms in the Lightweight CCM specification. Several examples will be used to demonstrate how these features and mechanisms can simplify production distributed real-time and embedded application development and integration. These examples will show how to develop CORBA components, how to assemble these components into applications, and how to deploy these applications in the Lightweight CCM run-time environment. Other examples will show how real-time extensions to Lightweight CCM can enable the development of robust DRE applications.

(Part 2 of this tutorial will be presented Tuesday morning from 0830 to 1230.)

1330 – 1730 ***Effective Component and Application Development using the Software Communications Architecture***

Track 3 Dominick Paniscotti & Bruce Trask, BAE Systems

The Software Communications Architecture (SCA) and Joint Tactical Radio System (JTRS) technologies have become ubiquitous throughout the US military communications domain. Entire sectors of the US armed forces' communications systems are being redesigned in support of these two technologies. This tutorial will cover basic SCA concepts and terminology as well as discussion of how some of the key terms have evolved during the maturation of this specification. This will clarify important terms that have become overloaded within the SCA community as well as with other technology domains. We will discuss the core responsibilities of the SCA Core Framework - component and waveform development. As part of this, the most effective patterns, principles and practices for developing SCA components will be covered. This section will include:

- How to use CORBA most effectively in SCA systems
- Which architectural, structural and behavioral patterns apply to SCA component development as well as which patterns can and should be used together
- How to leverage generic programming techniques in component development
- What techniques to use to maximize portability and correctness of waveforms

The tutorial will also include a live demonstration of an SCA compliant waveform running on two iPAQs.

1500 - 1515 Afternoon Refreshments

TUESDAY - July 13, 2004

TUTORIAL TRACKS

0830 - 1215 ***Data Distribution Service***

Track 1 Victor Giddings, Objective Interface Systems, Inc.
Gerardo Pardo-Castellote, RTI
Virginie Watine, THALES

The Data Distribution Service (DDS) is a recently adopted OMG specification that adds typed, multi-point data distribution between publishers and subscribers. This data-centric communications decouples publishers of information from subscribers in several dimensions. The DDS specification defines a Platform Independent Model that is independent of implementation technology, as well as a Platform Specific Model for implementation in CORBA technology. DDS consists of two different layers: the Data Centric Publish and Subscribe (DCPS) layer and the Data Local Reconstruction Layer (DLRL). The DCPS presents an easy to use interface that allows anonymous publication and subscription to typed topics. Extensive Quality of Service parameters are defined for the DCPS to allow architects to tailor systems to varied application requirements. The DLRL layer presents an object-oriented view of a data model and methods to tie entities of the data model to the DCPS. These entities will then be distributed and shared. This tutorial will cover the contents of the DDS specification, contrast it with other OMG specifications that offer data distribution capabilities, and discuss its possible application in different system architectures.

0830 - 1215 ***Using the Lightweight CORBA Component Model to Develop Distributed Real-time and Embedded Applications (Part 2)***

Track 2 Douglas C. Schmidt, Vanderbilt University & Frank Pilhofer, Mercury Computer Systems, Inc.
(Continued from Monday)

1015 - 1030 Morning Refreshments

1215 - 1300 Lunch

1315 – 1330 ***Opening Remarks – Program Committee Chair***

Andrew Watson, Object Management Group

1330 – 1530 ***Session 1: Components***

Chair: Andrew Watson, Object Management Group

CORBA and similar middleware standards provide a cost-effective communications infrastructure for building multi-CPU embedded and real-time systems using heterogeneous hardware, including general-purpose processors, digital signal processors (DSPs) and field programmable gate arrays (FPGAs). Component frameworks, including the CORBA Component Model (CCM), can further reduce the cost of building complex embedded systems by allowing applications to be built from reusable software components that are assembled and interconnected into applications, communicating via middleware. This allows rapid prototyping and reconfiguration of applications for ever-changing hardware that is upgraded as application requirements evolve. Presentations in this session describe various aspects of the emerging field of component frameworks for embedded and real-time applications.

Enabling Component-based Applications in Embedded Systems

Frank Pilhofer, Mercury Computer Systems, Inc.

This presentation gives a technical overview of component-based development and the Lightweight CCM as well as the Deployment and Configuration specifications. As a case study, applicability to the JTRS Software Communications Architecture (SCA), a CORBA-based component middleware used in software-defined radio engineering, is explored, demonstrating how SCA implementations can benefit from COTS products based on these recent OMG specifications.

A QoS-aware CORBA Component Model for Distributed Real-time and Embedded System Development

Nanbor Wang, Tech-X Corporation & Christopher D. Gill, Washington University

This presentation will demonstrate how the real-time and QoS-enabled CORBA Component Model (CCM) implementation can be used to manage cross-cutting resources statically for ensuring QoS for DRE systems. Examples and performance results shown in the presentation will illustrate how our research simplifies developing and evolving DRE applications by declaratively composing and configuring components, applications and their real-time policies.

Static Component Configuration Support for Real-Time Platforms

Chris Gill & Venkita Subramonian, Washington University - Nanbor Wang, Tech-X Corporation

Implementations of component models like EJB, COM, and CCM provide standard frameworks which isolate the functional concerns of a software system from para-functional concerns like transactions, security, realtime, etc. Each component model defines details of the component development process - implementation, packaging, deployment and configuration. The configuration of a component is separated from its implementation so that the same component can be run within different contexts. Apart from configuration, connection information between components is also specified so that components can be composed to realize the entire functionality of a system. In this position paper, we describe the component assembly process in the context of an open source implementation of the CCM.

Empirical Evaluation of CORBA Component Model Implementations for Distributed Real-time and Embedded Systems

Arvind S. Krishna, Douglas C. Schmidt, Vanderbilt University

COTS middleware is now widely used to develop DRE systems. DRE systems are themselves increasingly combined to form “systems of systems” that have diverse QoS requirements. The new generation of component middleware, such as the CORBA Component Model (CCM) based on the CORBA 3.x standard, addresses the limitations of earlier generation middleware by establishing standards for implementing, packaging, assembling, and deploying component implementations. This presentation will focus on the following topics pertaining to the use of CCM for DRE systems: 1) The challenges involved in benchmarking different CCM implementations; 2) Criteria for comparing different CCM implementations using key black-box and white-box metrics and; 3) The design of CCMPeef, an open-source benchmarking toolkit that can be used to evaluate aspects of CCM implementations to determine their suitability for the DRE domain.

1500 – 1900 ***Demonstration Area Open***

1530 – 1600 Afternoon Refreshments

1600 – 1800 ***Session 2: Resource Management in DRE Systems***

Chair: Shahzad Aslam-Mir, PrismTech Corporation

The designer of deterministically correct and temporally well balanced applications knows well that effective management of principal resources of their systems via effective schedules is key to a successful system. Effective resource management is therefore paramount in real-time embedded systems. This involves specifying and managing the use of the key elements of the system in a symbiotic manner using protocols such that all higher priority activities meet their deadlines or complete execution to a degree of determinism so as to maximize some time-value utility function for which the system has been designed. Therefore, resource management is the primary catalyst in achieving the end-to-end QoS requirements of these DRE systems. This session focuses on the presenters’ experiences in developing and refining tools and frameworks for real-time system resource provisioning, monitoring, and regulation mechanisms. In addition the session deals with the notion of adaptive resource management, and aims to tie it all together with the use of higher level canonical models such as lightweight CORBA components to reduce complexity as the level of sophistication of solution increases for DRE system developers to re-use.

Tools for QoS Aware Allocation and Deployment of CCM Based Applications

Patrick Lardieri & Gautam Thaker, Lockheed Martin Advanced Technology Laboratories

CORBA Component Model's Deployment and Configuration (D&C) specification provides considerable capability for users of component based systems, but it provides virtually no mechanisms to establish QoS aware deployments. We report on our efforts to couple CCM's D&C specification with an allocation and scheduling service to enhance the applicability of this technology to distributed, real-time, embedded (DRE) systems. Our measured results indicate progress towards automated deployment of such systems.

Adaptive Resource Management Services in CORBA

Lonnie R. Welch, David Fleeman, David Juedes & Chang Liu, Ohio University

In dynamic contexts such as space and military environments, a framework for automatically managing the allocation of computing and network resources is needed. The Quality-based Adaptive Resource Management Architecture (QARMA) is a set of CORBA services for resource management. The three major components are the System Repository Service, the Resource Management Service, and the Enactor Service. Experimental results using QARMA show improved latency measurements for applications operating in constrained environments, and indicate the ability of QARMA to provide allocations that allow real-time constraints of application systems to be met and that yield high utility.

Developing a Resource Status Service Using the CORBA Component Model

Balachandran Natarajan, Vanderbilt University, Richard Shapiro, BBN Technologies

In this presentation we present the motivation and design of such a Resource Status Service (RSS) as a collection of CORBA components. We show how the abstract functionality of the RSS maps easily to facets of a generic component. We demonstrate how naive use of CORBA component model to the RSS system can lead to solutions requiring additional programming during the deployment process, and we show how this can be programmed at a meta-level. Finally, we show with an example how RSS components can be deployed and permitted to use some non-standard additions to CCM containers without breaking the portability of the application code.

Provisioning Resources in DRE Systems with Lightweight CCM

Jeff Parsons, Douglas C. Schmidt, Balachandran Natarajan & Aniruddha Gokhale, Vanderbilt University
Patrick Lardieri & Gautam Thaker, Lockheed-Martin - Gary Duzan, BBN Technologies

This presentation will show how we use the Lightweight CORBA Component Model (LWCCM) to provision resources in a distributed real-time embedded (DRE) system by: (1) gathering information about resources and application requirements, and passing it on to other components for analysis and planning; (2) managing application startups, and; (3) launching a distributed resource status service. The presentation will also focus on the interaction between the Resource Provisioner, which operates at the node level, and components described in other RT Workshop presentations, which operate at a higher level.

1800 – 1900 *Demonstration Area Reception* hosted by



Borland®

WEDNESDAY, July 14, 2004

0830 - 1000 **Session 3: *Novel Exploitations of MDA***

Chair: Andrew Watson, Object Management Group

Model Driven Architecture and related concepts and techniques are starting to get used in novel ways in Real-time and Embedded systems in the area of architecture and implementation related to providing Quality of Service and related Configuration and Deployment of systems. Three papers are presented in this session covering this exciting new area.

Model-Driven Quality Assurance Techniques for Distributed Real-time and Embedded Systems

Emre Turkay, Arvind S. Krishna, Aniruddha Gokhale, Douglas Schmidt & Bala Natarajan, Vanderbilt University
Adam Porter, Cemal Yilmaz & Atif Memon, University of Maryland

This presentation focuses on addressing the shortcomings of conventional Quality Assurance techniques by describing a Model Driven Architecture (MDA) approach to distributed continuous QA, which helps improve software quality and performance iteratively, opportunistically, and efficiently around-the-clock in multiple, geographically distributed locations. We describe two MDA tools: 1) the Options Configuration Modeling Language (OCML), which helps in automating the tedious and error-prone QA activities of handcrafting configuration files and, 2) the Benchmark Generation Modeling Language (BGML), which synthesizes appropriate benchmark scaffolding code used for testing the selected configurations.

A QoS-aware Integrated Model Checking Environment for Developing and Validating Distributed Real-time and Embedded Applications

Gabriele A. Trombetti, Aniruddha Gokhale & Douglas C. Schmidt, Vanderbilt University
John Hatcliff, Matt Dwyer & Gurdip Singh, Kansas State University

Significant development efforts and costs are incurred developing and deploying distributed real-time and embedded (DRE) systems due to their inherent complexity and the need to validate and verify (V&V) their correct functioning and QoS. There exist tools that address some aspects of the V&V process but not all. V&V for DRE systems requires an end-to-end thorough V&V that is possible only by leveraging the features of all these tools collectively. This presentation describes an integrated environment that integrates (1) a model-driven middleware tool suite named CoSMIC that configures and deploys QoS-enabled component middleware-based DRE systems with (2) powerful model checking and analysis tools, such as Cadena and Bogor. This integration helps cut down development times and costs while simultaneously increasing the reliability of DRE applications.

Model-driven Deployment and Configuration of Component-based Systems

Krishnakumar Balasubramanian, Boris Kolpackov, Tao Lu, Aniruddha Gokhale & Douglas C. Schmidt, Vanderbilt University

While much attention has been devoted to developing middleware with desirable QoS properties for use in DRE systems, the problem of configuring and deploying such middleware has often been overlooked. This presentation describes techniques and tools for configuring and deploying Lightweight CCM middleware for DRE systems that leverages and enhances the recently adopted OMG Deployment and Configuration (D&C) specification via the following three-pronged MDA approach: 1. A modeling paradigm called Platform-Independent Component Modeling Language (PICML), developed using the MDA paradigm, 2. A generic data-binding approach to processing information contained in XML files conforming to an underlying XML Schema, 3. A run-time infrastructure that performs the actual deployment and configuration.

1000 - 1600 ***Demonstration Area Open***

1000 - 1030 Morning Refreshments

1030 – 1200 ***Panel - MDA in Real-Time: Use and Misuse***

Moderator: Andrew Watson, Object Management Group

MDA encourages developers to propagate design decisions across a system, yet most real-time design methods encourage making each design decision separately, or -at best -smearing them blurrily. Much of the use of MDA in IT systems relies on myriad profiles used for generating "blueprints" into which code can be embedded. The result is silos of related standards that cannot stand alone and architectural structures, though generated, that are each special case. Each real-time and embedded system demands a single application-independent software architecture that applies uniformly across the related portions of a system. Rate-monotonic analysis (RMA), for example, constitutes an unspecialized 'architecture' for those portions of the system that are periodic. MDA is perfectly suited for embedding application detail into this structure. Yet few do so. This panel explores why this is so, and what can be done about it.

Panelists: Stephen J. Mellor, Mentor Graphics
Ben Watson, Lockheed Martin
Gan Deng, Vanderbilt University

1200 - 1245 Lunch

1300 – 1400 ***Sponsor Presentations – DARPA & Borland Software***

1400 – 1530 ***Session 4: Fault Tolerance***

Chair: Victor Giddings, Objective Interface Systems

Fault tolerance, the ability of a system to detect, analyze, and recover from failures that affect the system, is a critical attribute of most distributed real-time and embedded systems. It is an attribute that can be provided by certain types of middleware, such as those conforming to the Fault Tolerant CORBA specifications. But Fault Tolerance mechanisms in middleware do not operate independently of other mechanisms that provide equally important attributes of a system. Indeed, the needs of fault tolerance are often in conflict with other attributes, such as real-time predictability or constrained resources. The interactions of Fault Tolerance with real-time asynchronous events, with CORBA components, and with resource management are explored in this session.

Implementation of a Fault-Tolerant Real-Time CORBA Event Channel

Christopher Gill & Huang-Ming Huang, Washington University

We will describe the design, implementation and preliminary performance characteristics of an open-source fault-tolerant real-time CORBA event channel. We will also compare its design, implementation, and performance to the open-source real-time CORBA event channel upon which it was based. In this research we have focused on the robustness of event channel subscriptions, so that if an event channel crashes the event delivery paths between event suppliers and event consumers are still preserved, and after a crash events can still be delivered. Furthermore, our solution offers configuration options for trading-off the latency of supplier/consumer subscriptions for the number of channel crashes that are assured to be tolerated. We will also discuss promising directions for future work on assuring delivery of events, while again supporting trade-offs between event latency and assurance of delivery.

A Fault Tolerant CORBA Component Model

Tom Bracewell, Maureen Mayer, Robert Kukura, Raytheon - Marc J Balcer, Model Compilers
Priya Narasimhan, Carnegie Mellon University

Model-based development of fault-tolerant systems requires CORBA and CCM standards and MIC tools that will let us design, deploy (and redeploy) such systems in a model-driven manner. We show how these standards and tools need to evolve. As replication is key to fault tolerance, component-based designs need a suitable entity of replication: the container. Containers need fault tolerant services to support components and container-state. XML schemas need to separate logical and physical deployment of replicas, and specify fault tolerance policies to middleware. MIC tools need to support replica configuration, deployment and run-time redeployment. This work was performed for DARPA PCES.

Proactive, Tunable, Resource-Aware Fault-Tolerance for Distributed CORBA Applications

Priya Narasimhan, Tudor Dumitras, Soila Pertet, Charlie Reverte, Joe Slember & Deepti Srivastava,
Electrical and Computer Engineering Dept, Carnegie Mellon University

Our research on the MEAD (Middleware for Embedded Adaptive Dependability) system attempts to identify and to reconcile the conflicts between real-time and fault-tolerance in a resource-aware manner. MEAD was born out of the lessons learned from implementing fault-tolerant CORBA, from recognizing its limitations, and from the emerging need to support: 1) informed decision-making to assign fault-tolerance properties, 2) trading off real-time and fault-tolerance to suit the application's needs, and 3) the transparent tuning of fault-tolerance, on the fly, in response to dynamic system/resource conditions. This presentation focuses on the key features of the MEAD's tunable, resource-aware fault-tolerance architecture, along with our promising preliminary results that indicate advances in our understanding of the development of real-time, fault-tolerant middleware systems.

1530 – 1600 Afternoon Refreshments

1600 – 1800 **Session 5: *Performance Considerations for DRE Systems***

Chair: Irfan Pyarali, OOMWorks

To simplify the development of complex distributed system with stringent QoS requirements, software developers have adopted several standards and practices that have provided higher level abstractions making it easier to build and model complex systems. These adaptations include component-based frameworks, distributed object request brokers, system and domain object services, and object-oriented programming languages. Though these tools and practices have undoubtedly raised the level of abstraction, it is imperative that developers understand the impact on system performance and scalability of embracing these technologies. This section will look into the performance implications of choosing and using different implementation languages, ORBs, and object services.

High-Performance Java™ : Comparing the Results of a Common ORB Design Implemented in 3 Languages

Kevin Buesing, Objective Interface Systems, Inc.

In this presentation a single, common high-level, language-independent ORB design targeting real-time and embedded systems implemented in Java™, Ada 95, and C++ is analyzed based on end-to-end latency of a roundtrip message. In addition trade-offs among language tcp transport choices are compared. Finally, the presentation discusses some of the techniques discovered for producing high-performance Java Software.

CORBA Inter-Operability Testing

Charlie Fudge & Traci McDonald, Naval Surface Warfare Center Dahlgren Division

This presentation will discuss the results of a standalone benchmarking effort involving two CORBA products, TAO and ORBexpress. This project provided insight and data on two key Open Architecture implementation issues, product interoperability and product interchangeability. Specifically, the presentation will contain introductory material stating the motivation for the project and illustrating the equipment and software configuration. Data charts will be shown depicting latency and throughput measurements for various data types and structures for TAO and ORBexpress in a number of different configurations. The presentation will conclude with several summary slides stating conclusions related to the goals of the project.

LBPerf: An Open Toolkit to Empirically Evaluate the QoS of Middleware Load Balancing Services

Ossama Othman, Jaiganesh Balasubramanian & Douglas C. Schmidt, Vanderbilt University

LBPerf is an open benchmarking toolkit used to empirically evaluate middleware load balancing strategies in a systematic and controlled manner. This presentation describes the design of LBPerf, focusing on its ability to (1) tune different configurations of middleware load balancing, including choosing different load balancing strategies and run-time parameters associated with those strategies, (2) evaluate application performance when using those run-time parameters under the presence of workloads, mimicking the natural workloads seen in a range of distributed applications, and (3) systematically evaluate the performance of different load balancing strategies.

RMBench: CORBA Benchmarking Services for Resource Management Middleware

Matt Delaney & Lonnie R. Welch, Ohio University

Numerous middleware packages for resource management and communications are emerging, however there is no toolkit available to systematically evaluate and compare them. HDB, Dynbench, and SWSL/SWG are all useful for finding the performance limits of real-time infrastructure components, however they do not support arbitrary workloads or QoS parameters. These deficiencies have been addressed in the RMBench system, a set of CORBA services for benchmarking the performance of distributed real-time middleware. Major features of RMBench that will be discussed are a System Specification File, Experiment Specification File, Sensor Node processes, Path Node processes, and robust Experiment Log Files.

1830 - 2030 ***Workshop Reception*** hosted by



Borland®



THURSDAY, July 15, 2004

0830 - 0930 **Session 6: *RT CORBA & Services***

Chair: Doug Jensen, MITRE Corporation

Two major enhancements to Real-Time CORBA are highlighted in this session. First, results from the implementation of Real-Time CORBA 1.2 (ne'e 2.0) with its Distributable Threads and pluggable scheduling framework; second, a real-time notification service for dynamic asynchronous event-based systems.

Experience Implementing and Evaluating Real-Time CORBA 2.0

Yamuna Krishnamurthy & Irfan Pyarali, OOMWorks - Christopher Gill, Louis Mgeta, Yuanfang Zhang & Stephen Torri, Washington University - Douglas C. Schmidt, Vanderbilt University

To support the dynamically changing QoS needs of open DRE systems, it is essential to propagate QoS parameters and to enforce task QoS requirements adaptably across multiple endsystems dynamically in a way that is simultaneously efficient, flexible, and timely. This presentation makes three contributions to R&D activities on QoS-enabled middleware that supports these types of open DRE systems: it describes the design and implementation of a Dynamic Scheduling framework based on the OMG Real-Time CORBA 2.0 specification; it examines the results of empirical studies conducted to validate our RTC2 framework in the context of open DRE systems; it presents a case study of our integration of multiple adaptive middleware QoS management technologies to monitor and control the quality, timeliness, and criticality of imagery adaptively within a representative DRE avionics system.

Design and Performance of a Real-time Notification Service

Pradeep Gore & Irfan Pyarali, OOMWorks LLC - Christopher Gill, Washington University - Douglas Schmidt, Vanderbilt University

The presentation will cover the following topics - Introduction to Real-Time Notification, programming paradigm, discussion of design and implementation challenges and, performance analysis. The Notification Service component structure will be discussed along with the requirements of RT-Notification RFP. We will discuss the integration with RT-CORBA 1.0 and IDL extensions made in our implementation. Thread-pool/lane support in RT-Notification, possible thread-pool configurations and end-to-end priority preservation will also be covered. The performance analysis will present the throughput, latency and jitter numbers for different load and configuration scenarios. The sources of overhead and performance tradeoffs will be discussed.

0930 - 1030 **Session 7: *Securing Embedded Systems***

Chair: Andrew Watson, Object Management Group

Distributed real-time and embedded (DRE) systems are becoming increasingly interconnected and open to external access via wide-area networks such as the Internet. These systems are often mission-critical and interdependent, so that failures in some parts of the system infrastructure can have catastrophic consequences elsewhere. In one well-publicized example, an Australian sewage system was attacked via the radio links between pumping stations, causing the release of millions of gallons of sewage into public waterways. Designers will in future have to pay more attention to the problem of securing embedded systems; the presentations in this session will describe two independent approaches to this problem.

The Partitioning Communication System: A Platform for Secure Communications within the MILS Architecture

Jeff Chilton, Objective Interface Systems, Inc.

In this presentation, we describe the Partitioning Communication System (PCS), a communications sub-system capable of maintaining the separation of security-critical data in transit between the nodes of a distributed, real-time system. The PCS is intended to fill the role of communications middleware in the Multiple Interacting Levels of Security (MILS) architecture and form the basis for real-time MILS CORBA. The PCS will be suitable for building distributed systems that are accredited to be Multi-level Secure (MLS).

Techniques for Managing Access to Secure CORBA Services

Phillip L. Mesnier, Object Computing, Inc.

This presentation addresses new techniques for managing access to secure CORBA services that is less burdensome than the CORBA Firewall specification, but more robust than simple port-forwarding firewalls. The solution presented here is a novel Inter-ORB protocol, PBXIOP. Modeled on a telephone exchange system, PBXIOP provides a single "exchange" process to listen for and mediate connection establishment, eliminating the need to open many ports through a firewall. The PBXIOP further enhances security by only allowing "approved" users to start services that may attach to the exchange.

1030 - 1045 Morning Refreshments

1045 - 1215 **Session 8: *Protocols for Distributed Real-time and Embedded Systems***

Chair: Douglas C. Schmidt, Vanderbilt University

For years, developers of DRE systems have built applications that ran over special-purpose networking protocols to meet their end-to-end QoS requirements. These types of systems are increasingly being built using standard Real-time CORBA middleware. To support the end-to-end QoS requirements of these DRE systems, the middleware must be integrated seamlessly with the underlying protocols. This session focuses on advances in the networking and ORB layers that significantly enhance the support for stringent QoS requirements of DRE systems. In particular, the talks describe the integration of Real-time CORBA with the IETF Stream Control Transmission Protocol (SCTP) and QoS-enabled IP routers via various means, including the new OMG Extensible Transport Framework.

Using SCTP to Improve QoS and Network Fault-Tolerance of DRE Systems

Yamuna Krishnamurthy & Irfan Pyarali, OOMworks LLC - Craig Rodrigues & Prakash Manghwani, BBN Technologies - Gautam H. Thaker, Lockheed Martin ATL

In our presentation, we will (a) describe some of the implementation details of how we added SCTP support to our DRE application, and (b) present experimental results investigating the following properties of SCTP: throughput performance, network-level fault tolerance performance, and performance with DiffServ based network prioritization services. Our results will be of interest to DRE developers who are interested in high performance, fault tolerant applications, and those considering using SCTP in their applications as an alternative to TCP/IP.

Network QoS Assurance through Admission Control

B. Coan, B. Dasarathy, S. Gadgil, K. Parmeswaran, I. Sebuktekin and R. Vaidyanathan, Telcordia Technologies & M. Conarty, PrismTech

We present adaptive network QoS (quality of service) technology that provides ongoing, end-to-end assurance that critical traffic belonging to admitted flows has bounded queuing loss, delay, and jitter. Our technology uses a Bandwidth Broker to provide admission control, and leverages differentiated services and related policing capabilities of commercial high-end routers and switches for enforcement. The technology adapts to dynamic configuration changes and uses device-independent layer-3 and layer-2 discovery algorithms to maintain a current view of resource availability. Under the DARPA ARMS (Adaptive and Reflexive Middleware) program, our technology is being integrated into a CORBA-based multi-layer resource management framework.

The Implications of the Use of QoS Sensitive Custom Transports under a Real-time CORBA ORB in Light of the Recent Extensible Transport Framework Specifications from the OMG

Shahzad Aslam-Mir, PrismTech Corporation

The recent OMG specification for extensible transports aims to simplify and unify the thinking behind the idea of plugging in custom transports under an ORB. Whilst this specification offers major benefits, some aspects that emerge from it require further clarification. This presentation will use experience from the trenches to highlight both shortcomings and recommendations of workarounds for some aspects of the specification and detail the constraints it imposes on the end-user developing software defined radio applications. It will also provide experiences of using custom transports for secure and narrow-bandwidth ORB communications for classes of application requiring a high degree of QoS control. Finally it will attempt to de-mystify some of the details behind what is common (and possibly incorrectly) termed zero-copy semantics as introduced into the engineering community.

1215 - 1300 Lunch

1315 - 1445 **Session 9: *Software Defined Radio: Experiences of Implementing the Software Communications Architecture (SCA)***

Chair: Andrew Foster, PrismTech Corporation

Software defined radio (SDR) has been a key goal of the signal and communications as well as the intelligence and defense community for some time now as it offers significant advantages over traditional hardware radios. However, interoperability has been a major issue in its adoption. Papers in this session will address SDR interoperability initiatives such as the Joint Tactical Radio System's (JTRS) Software Communications Architecture (SCA). The SCA defines a standard set of CORBA interfaces that allow waveform applications to run on multiple hardware sets. The SCA also defines a Core Framework (providing a standard operating environment) that must be implemented on every hardware set. Interoperability among radio sets is enhanced because the same waveform software can be easily ported to all radio sets.

Experience Report on Developing a Software Communications Architecture (SCA) Core Framework (CF) and Developer Kit (DK)

Bruce Trask & Dominick Paniscotti, BAE Systems

BAE SYSTEMS has designed, implemented, deployed and fielded a fully functional SCA Core Framework and developed a corresponding SCA Component Developer's kit. This presentation will cover the experiences developing both of these, including: performance characteristics of the Core Framework; the role Core Framework actually plays in an SCA system; techniques employed to ensure minimal change when moving from one ORB to another; principles of framework design applicable to the SCA CF; aspects of CORBA that can be abstracted away from component developers; what is entailed in undergoing JTEL testing; some of the myths of SCA CF and component development; and the key patterns for SCA component development.

Optimised CORBA to Meet SDR Requirements

Dave Dohse & Eric Christensen, General Dynamics C4 Systems

CORBA is part of the JTRS SCA standard Operating Environment (OE) for Software Defined Radios (SDRs); however, it is sometimes blamed for imposing processing requirements as much as five times that of legacy radio applications. Reliable CORBA-based operation is enabled by alternate data transports to TCP/IP, minimization of marshalling, persistent connections and RT CORBA policies that assure predictable performance. This presentation quantifies the potential penalties incurred through misuse of CORBA in an SDR. Techniques are presented for assuring that CORBA and related transport mechanisms contribute to Real-time stability without introducing onerous communication overhead.

Waveform Portability – The Impact of CORBA on a Critical Concern for the SCA/JTRS Community

Andrew Scheurer, Northrop Grumman Space Technology
Shahzad Aslam-Mir & Mark J. Glenn, PrismTech Corporation

The JTRS initiative for SCA compliant waveforms that are portable between various radios is an important milestone for CORBA fulfilling its promise of platform neutral portability. In addition the SCA specification, being laid out in terms of IDL, illustrates that CORBA continues to fulfill its promise. However, the methods, and mechanisms used in achieving this are still to some degree circumspect. This presentation will describe how CORBA and various language mappings help to achieve both SCA compliance but also, more importantly, waveform portability. Elements of concern in creating portable, small, compact SCA framework code will be discussed and guidance provided.

1445 - 1500 Afternoon Refreshments

1500 - 1600 ***Panel – Building Better Software-Defined Radios***

Moderator: Joseph M. Jacob, Objective Interface Systems

Building software-defined radios is hard. This panel will compare various Software Defined Radio (SDR) specifications, including the semantics of the deployment models and component models (including their associated descriptor files) of these specifications. The panel will also examine different ideas which might ease development and deployment of SDRs, including applying the Lightweight CORBA Component Model to the Software Communications Architecture as well as hosting CORBA on a DSP. The session will also include participants' experience in resolving roadblocks encountered in designing and deploying SDRs.

Panelists: Jerry Bickle, Raytheon Corporation
Steve Johnson, PrismTech Limited
Steve Osselton, PrismTech Limited
Dominick Paniscotti, BAE Systems

1600 - 1730 ***Session 10: Case Studies & Experience Reports***

Chair: Bill Beckwith, Objective Interface Systems

This session has presentations with case studies of systems and reports of experiences building, running, and/or using those systems.

Model-Based Integration Of Component-Based Embedded Systems—A Case Study

Mark Schulte, Boeing

For large-scale systems, the integration of common and project specific software components into systems that respect cross-cutting embedded system requirements such as hard and soft real-time deadlines, fault tolerance, and distribution is a challenge. Significant advances result from an integrated approach to explicit modeling component configurations, analysis of these models to ensure that they meet requirements prior to coding, and automated component configuration code generation. These have been implemented in an integrated suite of tools from a number of academic and industrial researchers, and an extensive set of tests and metrics have been captured to quantitatively and qualitatively assess their performance. This presentation describes these tools, the resulting integration process and performance, and the resulting potential for dramatically improving our ability to integrate large-scale embedded distributed real-time embedded systems.

Applying Model Driven Architecture Tools and QoS-enabled Component Middleware to Large-scale DRE Systems: A Case Study

Andrey Nechypurenko, Siemens Corporate Technology - Tao Lu, Gan Deng, Emre Turkay, Douglas C. Schmidt & Aniruddha Gokhale, Vanderbilt University

To evaluate the extent to which MDA and QoS-enabled component middleware technologies actually improve DRE system development productivity, quality, and understanding by working at a higher abstraction level than components and classes, we have created an Inventory Tracking System (ITS) that monitors and controls the flow of goods and assets in real-time. The ITS case study is novel since it applies a set of MDA tools and CCM components that help to automate the following two aspects of ITS development: Warehouse Modeling & Modeling and synthesizing deployment and configuration aspects. This presentation describes these two modeling aspects, focusing on the domain models and the model interpreters.

Model-Based Distributed Application Development for High-Reliability DoD Range Systems using the TENA Middleware

Russell Noseworthy, Object Sciences Corporation

The Test and Training Enabling Architecture (TENA) Middleware uses UML-based model-driven automated code generation to reduce the amount of software that must be written (and tested) by humans. Furthermore, the TENA Middleware provides the application developer with a powerful distributed shared memory programming abstraction. This programming abstraction is easy for the application developer to understand, resulting in applications with fewer mistakes. The TENA Middleware API relies heavily on compile-time type-safety to help ensure reliable behavior at run-time. Careful API design allows a great number of potential errors to be detected at compile-time that might otherwise go unnoticed until run-time, where the cost of an error could be extremely high.

PROGRAM COMMITTEE

Chair: Andrew Watson, Object Management Group

Dock Allen, MITRE
Shahzad Aslam-Mir, PrismTech
Bill Beckwith, Objective Interface Systems
Carol Burt, 2AB
Ben Calloni, Lockheed Martin Aeronautics
Joseph Cross, Lockheed Martin
Bruce Douglass, I-Logix
Andrew Foster, PrismTech
Mark Gerhardt, TimeSys
Victor Giddings, Objective Interface Systems
Andy Gokhale, Vanderbilt University
Doug Jensen, MITRE
Hermann Kopetz, TTTech
Kevin Loughry, Object Management Group
David McKinnon, Washington State University
Steve Mellor, Mentor Graphics
Jishnu Mukerji, Hewlett Packard
Priya Narasimhan, CMU
Steve Osselton, PrismTech
Irfan Pyarali, OOMWorks
Craig Rodrigues, BBN Technologies
Jorge Rodriguez, Borland
Doug Schmidt, Vanderbilt University
Bran Selic, IBM Rational Software
Jon Siegel, Object Management Group
Richard Soley, Object Management Group
Dave Stringer, Borland
Venkita Subramonian, Washington University
Gautam Thaker, Lockheed Martin Advanced Technology Labs
Fred Waskiewicz, Object Management Group
Lothar Werzinger, Xcerla