OMG's Workshop on Distributed Object Computing for Real-time and Embedded Systems

July 14-17, 2003 – Arlington, VA

Program

MONDAY – July 14, 2003

TUTORIALS

0900 - 1230 ***Tutorial 1 - Real-time CORBA***
Victor Giddings, Objective Interface Systems

The OMG Real-time CORBA specification extends CORBA for use in real-time systems. Real-time CORBA provides a clean infrastructure for building distributed applications with time constraints. The recent addition of Real-time CORBA 2.0, Dynamic Scheduling, provides significant extensions that allow application engineers to "plug-in" alternative workload management and scheduling capabilities. In addition, the Minimum CORBA specification offers a feature-optimized version of the CORBA specification that allows application designers to depend on the reduced feature sets of lightweight ORB implementations. This tutorial will highlight the concepts and features of each of the specifications, along with examples of their use.

0900 - 1230 ***Tutorial 2 - Software Communications Architecture (SCA)***
Neli Hayes, The Boeing Company

This half-day presentation covers the SCA, an open architecture framework based on open and evolving commercial standards such as CORBA, CORBAservices, CORBA Component Model, and POSIX, that using UML and XML vocabulary, describes how hardware and software elements are to operate in harmony within an SCA-compliant communications system, in order to promote component interoperability, interchangeability, and portability. An important source of requirements throughout the system design process, the SCA specification is used by communications system design engineers (e.g. radio, network, waveform, software, and others) just as an architect or planner uses a local building code to design and build homes. SCA-compliant communications systems (e.g. Joint Tactical Radio System (JTRS)), when designed in compliance with the SCA, meet standards for interoperability, just as properly designed plumbing or electrical systems meet local codes for construction and safety. Starting with a brief look at the JTRS program mission, leading to the creation of the SCA, this presentation introduces the SCA specification, supplements, and accompanying documents, and covers major aspects of the architecture, including the Core Framework (CF), and the Domain Profile. The CF is the SCA essential "core" set of open software interfaces and profiles that provide for deployment, management, interconnection, and intercommunication of software application components in any distributed embedded system, and through imposed design constraints, provides increased portability of these application components from one SCA-compliant platform to another. The Domain Profile depicts the packaging and deployment of SCA-compliant hardware device and software component implementations into the CF domain, through describing these components, their properties, and interconnections.

1030 – 1045 Morning Refreshments

1230 – 1330 Lunch

1330 - 1700 ***Tutorial 3 - CORBA Component Model***
Nanbor Wang, Washington University
Craig Rodrigues, BBN Technologies

CORBA Components package together related CORBA IDL interfaces into reusable bundles which can easily be assembled into larger applications. The OMG CORBA Component Model (CCM) provides a framework for assembling CORBA components into larger applications. CCM defines: an architecture for defining server-side components and their interactions, a model for packaging binary component executables, and a container framework which provides interfaces to component clients and offers access to CORBA services such for transactions, security, and events. This tutorial will introduce the general concepts of Component-based programming with the CCM to experienced CORBA developers. Through the use of illustrative examples this tutorial will teach developers how to use CCM to compose and assemble powerful distributed applications.

1330 – 1700 ***Tutorial 4 - OMG Smart Transducer Specification***
Hermann Kopetz, Vienna University of Technology

The OMG Smart Transducer Specification introduces a single easy-to-understand standardized interface to access data that are transferred from/to many different types of smart sensors and actuators. The internal details of the sensor/actuator structure are hidden from the user. The standard has been optimized to support cost-effective implementations on small 8 bit microcontrollers that are commonly used in smart transducers. Central to this specification is a simple standardized interface file system (IFS) that contains all data that are exchanged between a smart device and its users. The Smart Transducer Specification is in conformance with the design principles of the time-triggered architecture. The tutorial will start with a presentation of the design principles of the time triggered architecture, which is a generic architecture for safety-critical real time applications. It will then explain the detailed structure of the smart transducer specification and cover existing implementations.

1330 – 1700 ***Tutorial 5 - IDL Design Patterns for Building Flexible Embedded Systems with Real-time CORBA***
Bill Beckwith, Objective Interface Systems

Real-time CORBA has been adopted by many projects developing embedded systems. When engineers use Real-time CORBA for the first time the designs they produce show a consistent pattern to other first-time users. These first-time designs differ greatly from the designs those same engineers produce on their second or third generation projects. The purpose of this tutorial is to convey good IDL style, effective name space management, the benefits and use of strong typing, designing for extensibility, designing for reliability, performance impact of language constructs, how to write "fast" IDL, use of CORBA services in embedded applications, and several design patterns especially useful in embedded systems.

1500 – 1515 Afternoon Refreshments

TUESDAY - July 15, 2003

0900 – 0915 ***Workshop Welcome -*** Andrew Watson, Object Management Group

0915 – 1200 **Session 1 –** ***Model Based Development of Real-Time Systems***

Chair: Douglas C. Schmidt, Vanderbilt University

Model-based development of software systems using OMG's Model Driven Architecture (MDA) promises to improve software productivity and reduce error rates while also creating applications that are easier to modify and maintain. Papers in this session describe the application of model-based approaches to real-time and embedded applications, an area where MDA has already delivered impressive results.

1-1 Model-Based Integration of Reusable Component-Based Avionics Systems
Wendy Roll, David Sharp and Dennis Noll, Boeing - Phantom Works

This presentation describes our DARPA Information Technology Office (ITO) Model-Based Integration of Embedded Software (MoBIES) program work towards infusing CORBA Component Model technologies into the Boeing Bold Stroke distributed real-time embedded (DRE) architecture. The Bold Stroke architecture is being applied on several large-scale military embedded systems and has demonstrated the suitability, benefits, and challenges of CORBA-based DRE middleware for these systems. Our MoBIES work has extended this work towards a CCM-inspired real-time component model incorporating support for facets, receptacles, event sources/sinks, homes, and factories, etc, and in a vision for multiple view model-based approaches to real-time component integration, configuration, and analysis.

1-2 Model-based Design & Implementation of Real-time Embedded Systems in Cadena
Gurdip Singh, John Hatcliff, Matthew Dwyer, Venkatesh Ranganath, Xianghua Deng, Prashant S. Kumar and Qiang Zeng, Kansas State University

Distributed real-time embedded systems are often mission-critical and need to satisfy stringent performance goals as well as correctness properties. To address this problem, we are developing Cadena - a modeling, analysis and configuration tool for developing CCM-based systems. We will present the current state of Cadena which takes a model-driven approach to design and implement DRE systems and includes facilities that support (a) specification and synthesis of implementations of systems of components, (b) dependency and state transition-semantics specifications and automated analysis tools, (c) reasoning about correctness properties, and(d) specification of QoS parameters and tools to configure middleware.

1015 – 1030 Morning Refreshments

1-3 CoSMIC: An OMG MDA Toolsuite for Distributed Real-time and Embedded Applications
Aniruddha Gokhale, Balachandran Natarajan, Jeff Parsons, K. Balasubramaniam, Tao Lu and Boris Kolpakov
ISIS Vanderbilt University

CoSMIC is an OMG MDA tool suite tailored to the requirements of DRE applications. CoSMIC addresses the following unresolved challenges in using COTS middleware to build mission-critical DRE systems with time and space constraints: lack of middleware composability to support multidimensional QoS, accidental complexities in integrating software systems, accidental complexities in configuring middleware, lack of principled methodologies to support reflective middleware capabilities

1-4 UML Profile for the Society of Automotive Engineers' Avionics Architecture Description Language
Edward Colbert, Absolute Software Co., Inc.

The Society of Automotive Engineers (SAE) is developing a standard Avionics Architecture Description Language (AADL) using UML and Honeywell's MetaH architecture language as a base. MetaH has been used on 20 demonstration projects. The AADL is designed to support the specification and analysis of hardware and software for real–time, fault–tolerant, safety–critical, securely partitioned, dynamically reconfigurable multi–processor system architectures. The AADL is intended for avionics, space, and other embedded time-critical systems where a highly integrated, rapidly evolvable approach is needed, such as robotics. This presentation will look at the proposed AADL concepts and its UML profile.

1200 – 1800 Demonstrations Open

1200 1300 Lunch

1300 – 1400 **Sponsor Presentations**

DARPA ***"Advances in Model Driven Middleware for***
Distributed Real-time and Embedded Systems"
Dr. Douglas C. Schmidt, Vanderbilt University

PrismTech ***"Evolution of Real-time CORBA:***
A Perspective on the Good, the Bad, and the Ugly"
Shahzad Aslam-Mir Ph.D., Chief Scientist Prism Technologies USA
and Guest Speaker - Doug Jensen, Ph.D., MITRE

1400 – 1415 Afternoon Refreshments

1415 – 1615 **Session 2 - *Domain Frameworks***
Chair: Dock Allen, MITRE

There is a current trend in embedded and real-time systems to use domain frameworks; these frameworks support functions that are unique to the domain, and domain-specific versions of generic functions such as configuration and deployment. In this session, presentations will consist of a few of these frameworks, discussion of their approach to real-time and embedding support, as well as their relationship to open middle-ware standards.

2-1 Experience Report from Developing and Fielding the First SCA 2.0 Compliant JTRS Radio
Dominick Paniscotti, BAE Systems and Bruce Trask, Object Computing Inc.

BAE Systems recently completed a 27 month engineering development program which culminated in the first successful field testing of a Software Communications Architecture (SCA) 2.0 compliant JTRS (Joint Tactical Radio System) radio. The U.S. Army's Electronic Proving Grounds recently operated and tested a network of these BAE Systems JTRS Engineering Development Model prototype radios, during a three-week exercise at Fort Huachuca, Arizona. This presentation will cover experiences from the development of a CORBA based component framework and CORBA capable software components that run on top of this framework. These experiences will include steps we took to leverage optimizations available from CORBA-compliant ORBs to meet the strict performance requirements of the system, the benefits from CORBA that allowed us to field this system within a very compressed schedule, and views from the trenches using CORBA on a daily basis to produce a distributed, real-time, embedded, multi-threaded, object-oriented, heterogeneous, multi-implemenation language, platform-independent (whew) product that was delivered and successfully field tested by a real customer.

2-2 The TENA Middleware
J. Russell Noseworthy, Object Sciences Corp.

Reliability is frequently the most critical aspect of a real-time application, even more important than raw speed. Ultimately, reliability is determined through the course of time, after considerable testing. However, the time required to achieve the desired reliability can be greatly reduced with the help of middleware and tools to reduce the possibility of human error. 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.

2-3 Performance Testing of the TENA Middleware
Gregory Schultz, Tybrin Corporation

This presentation will discuss the performance testing of the Test and Training Enabling Architecture (TENA) middleware. Foundation Initiative 2010, the sponsor of TENA, is developing the TENA middleware to provide a common software framework for test and training ranges to use that will improve interoperability and reuse. This presentation discusses the TENA object model, test cases, the performance benchmark applications and results. The test-cases involve publishing and subscribing at the various inheritance levels of the object model. The number of published objects is also varied for each test case. The performance results are presented as end-to-end application latencies between the publisher and subscriber.

1615 –1630 Afternoon Refreshments

1630 – 1800 ***Vendors' Panel***
Moderator: Andrew Watson, Object Management Group

In this panel, vendors of Real-time implementations will briefly describe their future plans, ideas on standards conformance, suggestions for future standardization, or whatever else is on their minds. This will be followed by questions from the floor and (if past experience is any guide) lively discussion by all workshop participants.

Panelists: Shahzad Aslam-Mir, Prism Technologies
Bill Beckwith, Objective Interface Systems
Tom Culpepper, 2AB
Doug Schmidt, Vanderbilt University
Malcolm D. Spence, Object Computing, Inc.

Additional Panelists TBA

WEDNESDAY - July 16, 2003

0900 – 1200 **Session 3 - *Real-Time Fault Tolerance***
Chair: Victor Giddings, Objective Interface Systems

Recent advances have separately added end-to-end predictability and fault-tolerance to middleware technologies. The results, such as the Real-Time CORBA and Fault-Tolerant CORBA standards, are not necessarily compatible or sufficient to ensure both of these characteristics. This session is an exploration of attempts to resolve them.

3-1 A QoS Extension for Fault-Tolerant Real-Time CORBA
Sylvester Fernandez and Joseph K. Cross, Lockheed Martin Tactical Systems

This presentation describes our experience with implementing end-to-end Quality of Service (QoS), specified in terms of desired latency and reliability, to configure one or more fault-tolerant real-time CORBA Event Channels. It demonstrates how distributed, real-time embedded applications can be effectively isolated from the specific technologies that make up the computing infrastructure. Our approach is currently being formulated in terms of patterns that can be used to provide QoS extensions to middleware services in a Model Driven Architecture representation of the solution.

3-2 Real-Time Fault Tolerant CORBA
P. M. Melliar-Smith and Louise E. Moser, Eternal Systems, Inc.

An implementation of Fault-Tolerant CORBA must ensure that the replicas of an object execute with strong replica consistency, so that the states of the replicas are identical at logically equivalent times. Strong replica consistency allows the application program to be programmed substantially without the insertion of additional code for fault tolerance, greatly simplifying the application programming, speeding its development and reducing its cost. An implementation of Real-Time Fault-Tolerant CORBA must, in addition, respect real-time priorities and scheduling, and must execute with predictable low latency. Eternal Systems has developed an infrastructure that satisfies these requirements, in addition to maintaining strong replica consistency.

1000 – 1830 Demonstrations Open

1000 – 1030 Morning Refreshments

3-3 Replication Strategies for Fault-Tolerant Real-Time CORBA Services
Huang- Ming Huang and Christopher Gill, Washington University
Bala Natarajan and Aniruddha Gokhale, Vanderbilt University

Our presentation will make two contributions to the state of the art in assuring properties for multiple system dimensions in CORBA middleware. First it will examine the characteristics of a well-known representative application in the DRE systems domain of avionics mission computing, and describe how two different abstractions for replication apply most appropriately to the event and data management layers of the system. Second, it will examine the relationships between real-time and fault tolerance within each of these abstractions, and between the abstractions.

3-4 MEAD: Support for Real-Time Fault-Tolerant Middleware
Priya Narasimhan, Carnegie Mellon University

Through our scalable, transparent, tunable, real-time, fault-tolerant Middleware for Embedded Adaptive Dependability (MEAD) infrastructure, we aim to reconcile the fundamental conflicts between real-time and fault tolerance in a resource-aware manner. Although MEAD is described in the context of CORBA (with CORBA being one of our first target platforms), its transparency allows it to support other middleware platforms, including Real-Time Java. The real-time fault-tolerant resource-aware MEAD infrastructure combines the strengths of various novel ideas: easy-to-use configuration of reliability, proactive dependability, and hierarchical resource/fault management. In addition, MEAD employs an extended fault model, in order to provide tolerance to a wider variety of faults, including processor/process/object crash faults, communication faults such as message losses, and timing faults such as missed real-time deadlines. MEAD extends its capabilities and protection to its own components, making it a truly dependable system.

1200 – 1300 Lunch

1300 – 1500 Session 4 - *Deeply Embedded Systems: Architecture and Experience*
Chair: Steve Grimaldi, Objective Interface Systems

Since the standardization of Real-Time CORBA, many projects have experienced success implementing the technology. This session presents both new application experience as well as different architecture concepts that expand these existing achievements into even newer realms.

4-1 Launching Real-Time CORBA into Space: Motivation and Implementation
Stuart D. Fowell, Simon Fowler and Roger Ward, SciSys (Space and Defence) Ltd.

SciSys is currently performing a number of projects on the use of software agents, built using CORBA middleware, to improve operations within Space missions, in particular in autonomous satellite/ground systems. We will present the motivation for using a minimum profile of Real-Time CORBA on-board a spacecraft, the underlying technologies that we are using to support this, and then an example of how we are applying these technologies.

4-2 CORBA Interfaces for Power Management
David Haverkamp, David Jensen and Steven Koenck, Rockwell Collins

Designers creating handheld and other computing systems with low power requirements face an intimidating set of requirements. Size and weight often limits the battery capacity and available power. The power limitations directly affect the microprocessor selection, memory size, and software implementation. In contrast, designers want the advantages of portability, maintainability, and design cost savings that CORBA designs offer. However, it is recognized that the current generation of commercial CORBA software and their supporting commercial operating systems assert a significant (often unattainable) power demand on these low power embedded devices. Rockwell Collins is using CORBA software and commercial operating systems in its implementation of the JTRS Cluster 1 communication system. We are developing approaches to address the power demand being introduced by the CORBA software and the operation system. Future JTRS Clusters will support handheld devices and Unmanned Air Vehicles (UAVs) and introduce even greater power constraints. In this talk, we will review our experiences with using CORBA in JTRS platforms and outline several research avenues we are exploring to reduce the power overhead introduced today with CORBA.

4-3 Fine Grained CORBA Services to Build Scalable DRE Architectures
Victor Giddings, Objective Interface Systems, Inc.

As well as the functional services directly provided, the CORBA Naming and Events Services provide important "architectural glue" to systems built with CORBA. Unfortunately, these services are traditionally implemented as monolithic standalone servers. This presentation will discuss experiences with the use of a set of fine-grained CORBA services that are more suitable for architecting scalable DRT&E systems. These library-based implementations are relatively "light-weight" in both CPU and memory usage, and are hosted by a RTCORBA-compliant ORB. These characteristics combine synergistically with CORBA's location transparency to provide an "architectural transparency".

1500 – 1600 Afternoon Refreshments

1500 – 1600 ***Poster Session***
Coordinator: Andrew Watson, Object Management Group

These papers, covering a variety of topics, will be presented as posters available for viewing any time during Wednesday and Thursday. The posters' authors will be available to discuss their work during this hour-long session on Wednesday afternoon.

1 Publish/Subscribe Data Service for Real-Time Safety-Critical Systems
Ray Richards and David Haverkamp, Rockwell-Collins

We have developed a middleware based data distribution service for safety critical, fault tolerant applications, suitable for avionics use. The programmatic abstraction of ‘location transparency’ is extended to provide ‘data location transparency’ to provide an ‘aircraft abstraction layer’. The Avionics Data Service relies on the underlying CORBA ORB to provide platform, operation system and language independence. It enhances this interoperability by providing transparent data type conversions and engineering units conversion, where possible. To date, we have developed implementations of the Avionics Data Service that provides data location transparency and interoperability between a Windows-based simulation and Java-based simulations and avionics functionality executing on an embedded PowerPC platform.

2 Integrating Real-Time Engineering into Development of a Power Distribution SCADA System
Huáscar D. Espinoza and Jorge A. Nava, CPGISC Universidad Mayor de San Andrés

This work addresses the problem of integrating real-time requirements into SCADA design, related to the electric power industry. The proposed focus covers the timing understanding and resource scheduling that would have to consider any SCADA system for a predictable response when different situations of computer loading are presented, and to maintain a suitable performance during peak activity states. The used techniques have been: an adapted version of RUP as development process, MAST as real-time scheduling tool, UML for architecture modeling, RMA theory for real-time mathematical analysis, and specific standards for power utilities to obtain an open, extensible, and reusable system.

3 Performance of Java Communications With and Without CORBA
Victor Giddings, Objective Interface Systems, Inc.

This poster will present a comparative analysis of Java socket I/O and direct socket I/O when used directly and via several Java ORBs. The market's desire for "100% Java" solutions mandates the use of the current Java library implementation for sockets when developing infrastructures for Java-based distributed systems. This inserts another layer that must be accounted for in the performance analysis of distributed systems. This presentation will discuss this cost relative to the overhead introduced by Java ORBs.

4 The IST HRTC Project
Ricardo Sanz, Autonomous Systems Laboratory Universidad Politécnica de Madrid

The IST HRTC Project is a research project investigating enhancements to the CORBA specifications for distributed control systems that have real-time requirements with hard timing constraints. It is focused on the analysis of requirements in CORBA-based distributed control systems and the elaboration of theory and methodology for the development of hard-real time CORBA applications. The requirements will be developed based on available experience and on two project-specific experiments in robot control and distributed process control.

5 Enabling High-Performance via Zero-Copy Real-time CORBA
Bill Beckwith and Lance Kibblewhite, Objective Interface Systems, Inc.

Object Management Group’s members are currently developing a specification for "High-Performance Enablers" that extend Real-time CORBA to accommodate its efficient use on very high-performance communication technologies. This poster will discuss the HP Enablers specification effort and the performance implications of a successful specification in this area; specifically, the mathematical effect of both internal and networking stack copies on the performance of ORBs will be detailed.

1600 – 1800 **Panel -** ***Directions In Military and Government Middleware Research and Development***

Moderator: Gary Toth, US Office of Naval Research

This panel session will bring together prominent research and development Program Managers on the panel, with researchers and developers in the audience, to discuss the future of middleware research and development projects in military and government applications. The panel will include presentations on middleware research and development directions from the US research funding agencies, as well as presentations from a new major US Navy development program, and from a major US Air Force development program. The format will be short presentations of the Program Managers' visions followed by an extended question and answer session with the audience.

Panelists:
Joseph Cross, Program Manager, DARPA
Lou DiPalma, Raytheon Corporation
Helen Gill, Program Manager US National Science Foundation
Doug Schmidt, Vanderbilt University and US DARPA

1830 –2030 ***Workshop Reception*** ***hosted by*** DARPA & PRISMTECH

Thursday, July 17, 2003

0900 – 1145 Session 5 - *Implementing Real-Time CORBA Specifications*

Chair: Shahzad Aslam-Mir, Prism Technologies

Presentations in this session investigate aspects of real-time CORBA implementation.

5-1 Design & Implementation Issues in the Dynamic Scheduling Real-Time CORBA 2.0 Specification
Yamuna Krishnamurthy and Irfan Pyarali, OOMworks LLC – Chris Gill, Washington University –
Victor Fay-Wolfe, University of Rhode Island

Quality of Service (QOS) enabled Distributed Object Computing (DOC) middleware has found widespread acceptance in the area of Distributed Real-Time and Embedded (DRE) Systems. DRE systems are of two types: (1) static and (2) dynamic. Static distributed real-time systems are those where the set of applications that will run in the system, their load and lengths of execution are known. Hence, these applications can be scheduled a priori and the feasibility of that schedule can be checked before the system is active. Dynamic real-time distributed systems, however, may not have the knowledge of the applications that will run on their system or the order in which they will be executed. In this case, the system should be able to adapt to real-time QoS requirements in a dynamically changing environment. Hence, there is a need for a means to propagate and preserve the QoS requirements of DRE systems on every node they span. The Dynamic Scheduling Real-Time CORBA 2.0 (DSRTC2) specification addresses the above concerns. The presentation covers our experiences in implementing the DSRTC2 framework.

5-2 Effective Use of Real-time Java features to Implement Real-Time CORBA
Arvind S. Krishna and Douglas C. Schmidt, Vanderbilt University
Krishna Raman and Raymond Klefstad, University of California, Irvine

We present the results of our R&D activities associated with identifying how RTSJ features can be integrated with components in RT-CORBA ORBs. These components are typically associated with RT-CORBA Thread-pool lanes and span the following three layers: (1) I/O Layer (e.g., Acceptor-Connector and Reactor), (2) ORB-Core (e.g., CDR streams and Buffer Allocators), and (3) Object Adapter Layer (e.g., Thread Pools and the POA). To eliminate priority inversions related to invocations of the garbage collector during a request upcall, it is essential that key ORB objects be allocated either within Scoped or Immortal memory to enable use of NoHeapRealTime threads. Proper use of NHRT threads ensures that user requests are not interrupted in an inappropriate time.

1000 – 1015 Morning Refreshments

5-3 Strategized, Coordinated Services for Real-Time Middleware
Lisa Cingiser DiPippo, Victor Fay-Wolfe, Matthew Murphy, Priyanka Gupta, Jiangyin Zhang, Jianming Ye and Kevin Bryan, The University of Rhode Island

Real-time middleware services should be coordinated in such a way that the algorithms used for schedulability analysis, request binding, priority assignment, and overload management work together in a cohesive way. We have developed a pair of strategized, coordinated services for Real-Time CORBA that can implement various coordinated algorithms for binding and scheduling tasks. These services provide a framework for implementing various scheduling schemes so that coordinated algorithms can be “plugged-in” to the services based upon the kind of system that will use the middleware. We will present the design of the strategized services, the coordination of a set of algorithms that solve a particular problem in the problem space, and the use of a model integrated computing tool to automate the creation of the correct service implementations based upon the definition of the problem to be solved.

5-4 SCIOP Implementation in a Real-time ORB Using an Extensible Transport Framework
Gautam Thaker, Patrick Lardieri, Chuck Winters, Ed Mulholland, Jason Cohen and Keith O'Hara,
Lockheed Martin Advanced Technology Laboratories

Middleware and network infrastructure that support many distributed, real-time, embedded systems must support stringent quality of service (QoS) requirements. These QoS requirements generally span a broad spectrum from timeliness to fault tolerance to security. Here we report on our ongoing effort to significantly enhance network link failure tolerance for DRE systems by exploiting the emerging Stream Control Transmission Protocol (SCTP), a new transport protocol standard from IETF. We describe the design decisions made and show measured performance under stressful conditions where network links are constantly failing and being restored. Our measurements show that maximum latencies suffered even under these conditions do not exceed 50 msec. We also briefly describe about a dozen SCTP protocol parameter settings, emphasizing the subset of these parameters that impact the maximum latency metric.

1145 – 1245 Lunch

1245 – 1445 Session 6 - *High-Assurance and Security*

Chair: Thomas Culpepper, 2AB

High assurance and security are becoming paramount, not just for real-time or embedded systems, but for all computing needs. The possible consequences of mission-critical system failures are severe, and malicious attacks are becoming much more sophisticated. High assurance techniques and technologies defend against such failures and attacks. Several OMG security standards/technologies are in existence and have proven themselves as viable solutions in combating malicious and accidental security breaches within the IT world. In this session, we will be exposed to several enlightening ideas and concepts while exploring ways to continue providing secure and/or high-assurance solutions.

6-1 Real-time MILS CORBA: High Assurance Security for Real-time, Distributed Systems
Bill Beckwith and Jeff Chilton, Objective Interface Systems, Inc.

A frequent requirement in mission-critical systems is the need for high-assurance security. 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 (which form the foundation of the DoD Orange Book). The interaction of CORBA and MILS security is not addressed by the existing CORBA Security specification, since the CORBA specification does not address multilevel security (mandatory access control) and provides a overly wide trust model. This presentation will describe an initiative to integrate and standardize Real-time CORBA with MILS.

6-2 Embedded Real-time CORBA in the Architecture of Control Systems: A Case Study
Shahzad Aslam-Mir, PrismTech Limited and James L. Paunicka, Boeing Phantom Works

For some time CORBA has been thought unsuitable for use in the control-loops of mission critical flight control systems. As a consequence CORBA's use has been limited to mission management and avionics management layer applications. This paper will use a Boeing Phantom Works case study to show that embedded real-time CORBA software mechanisms facilitate deterministic and predictability characteristics, so as to easily accommodate more in mission critical flight control systems for guidance, control and navigation in a truer control loop-driven over a CORBA backplane sense.

6-3 CORBA in the Time-Triggered Architecture
Hermann Kopetz and Thomas Losert, Vienna U. of Technology;
Ricardo Sanz, U. Politécnica de Madrid and Miguel Segarra, SCILabs Ingenieros S.L.

A TT communication system sends periodically at an a priori known instant of the global time a new version of a state message from the sender to the receiver nodes. A predetermined fraction of the bandwidth of a state-message transmission slot can be assigned to a node-local ET message server that implements the exactly-once semantics for the transport of ET messages. The standard IIOP protocol is implemented by using this event-message server. Access to all standard CORBA services is thus provided. Additionally, access to the real-time data contained in the TTA state messages is provided locally at each node by mapping the state message addresses into the CORBA Orb. The workshop presentation reports about the real-time performance, such as delay and jitter, that has been achieved in the prototype implementation.

1445 – 1500 Afternoon Refreshments

1500 – 1700 Session 7 – *Applying Components and Composition Techniques to Distributed Real-time Systems*

Chair: Craig Rodrigues, BBN Technologies

OMG CORBA middleware technologies have greatly simplified the development of distributed applications by abstracting away low-level network programming concerns. However, traditional CORBA has some shortcomings, namely: (1) conveying complex interdepencies between interfaces in IDL is hard, and (2) IDL does not express how interfaces should be assembled and deployed in applications. The CORBA Component Model addresses these concerns by (1) introducing the Component as a software artifact which encapsulates a set of related interfaces, and (2) defines facilities for assembly and deployment of components in applications. This session will cover how Component-based techniques can improve the development of distributed real-time systems.

7-1 A QoS-aware CORBA Component Model for Distributed Real-time and Embedded System Development

Nanbor Wang and Christopher Gill, Washington University

In this presentation, we will show 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 can simplify developing and modifying DRE applications by declaratively composing and configuring components, applications and their real-time policies. Finally, we will explain how CCM implementations can be extended to facilitate the injection of various cut-points to manage aspects that cross-cut multiple layers in an applications and how they can be used to support adaptive and dynamic QoS provisioning.

7-2 A Generative Programming Approach to Middleware Development

Venkita Subramonian and Christopher Gill, Washington University

The OMG's Model Driven Architecture is emerging as a way to enable seamless interaction between applications developed using various middleware platforms like CORBA, Java/EJB, .NET and XML/SOAP. The Generative Programming paradigm offers techniques which can be used by the component implementor to configure the component implementation based on the configuration settings of the component and configure the infrastructure more flexibly. In this presentation, we investigate the applicability of C++ Template Meta Programming - a specific Generative Programming technique - to the development of configurable middleware infrastructure mechanisms.

7-3 Data Distribution Strategies with Fault-Tolerant Components

Richard T. McClain, Joseph N. Licameli and Victor R. Rienzo
Lockheed Martin Naval Electronics & Surveillance Systems-Syracuse

As developers, you must consider many factors when designing data distribution mechanisms in a distributed system. Fault-tolerant and real-time requirements add to the complexity of the design. Using a component architecture based on CORBA has proven an effective strategy in solving these issues in a deployed integrated combat system (ICS). The system used the client-server model as its communication paradigm and used a publish-subscribe approach for data distribution. In this presentation, the speaker discusses strategies for resolving fault-tolerant issues when using redundant servers and provides important lessons-learned when using publish-subscribe interfaces based on CORBA.

1700 – 1715 *Workshop Wrap-up and Closing Remarks*

Program Committee:

Chair: Andrew Watson, Object Management Group

Dock	Allen,	MITRE
Shahzad	Aslam-Mir,	PrismTech
John	Bay,	DARPA
Bill	Beckwith,	Objective Interface Systems
Ken	Black,	Borland
Carol	Burt,	2AB
Ben	Calloni,	Lockheed Martin Aeronautics
Joseph	Cross,	Lockheed Martin
Bruce	Douglass,	I-Logix
Mark	Gerhardt,	TimeSys
Victor	Giddings,	Objective Interface Systems
Chris	Gill,	Washington University
Andy	Gokhale,	Vanderbilt University
Steve	Grimaldi,	Objective Interface Systems
Doug	Jensen,	MITRE
David	Jeong,	ROCOZEN
Hermann	Kopetz,	TTTech
Kevin	Loughry,	Object Management Group
David	McKinnon,	Washington State University
Trudy	Morgan,	SPAWAR Systems Center
Jishnu	Mukerji,	Hewlett Packard
Priya	Narasimhan,	CMU
Steve	Osselton,	PrismTech
Irfan	Pyarali,	OOMWorks
Craig	Rodrigues,	BBN Technologies
Doug	Schmidt,	DARPA
Bran	Selic,	IBM-Rational
Jon	Siegel,	Object Management Group
Richard	Soley,	Object Management Group
Gutam	Thaker,	Lockheed Martin Advanced Technology Labs
Fred	Waskiewicz,	Object Management Group
Lothar	Werzinger,	KRONES AG