Objective

To create a unified framework (UniFrame) that will allow a seamless integration of heterogeneous and distributed software components

UniFrame Approach

- **Unified Meta-component Model**
  - Components, QoS, Infrastructure
- **Generative Domain Model**
  - Domain model, Composition/Decomposition Rules, Generative Programming
- **Two-Level Grammar**
  - Formal specification of components
- **Process**
  - For integration
Unified Meta-component Model (UMM)

- Component
  - Autonomous and non-uniform
- Service and its guarantees
  - Offered by each component with QoS
- Infrastructure
  - Environment
    - Headhunters
    - Internet Component Broker

Quality of Service Reference Model

- A general categorization of different kinds of QoS; including QoS that are fixed at design time as well as ones that are managed dynamically
- Identification of the basic conceptual elements involved in QoS and their mutual relationships. This involves the ability to associate QoS parameters to model elements (specification)

Quality of Service Parameters

- These are parameters that describe the fundamental aspects of the various specific kinds of QoS based on the QoS categorization identified in the reference model. This includes but is not limited to the following:
  - time-related characteristics (delays, freshness)
  - importance-related characteristics (priority, precedence)
  - capacity-related characteristics (throughput, capacity)
  - integrity related characteristics (accuracy)
  - safety-related characteristics
  - availability and reliability characteristics

Generative Domain Model Generator
**Automation of Code Generation**

- Two-Level Grammar (TLG) is used to formally specify UMM and GDM (feature modeling and generation rules).
- TLG interpreter will execute generation rules to create a programming language program.

**Component Discovery Process**

- Natural Language Query (Functional and Non-functional Requirements)
- Query Processor
- Application GDM (in TLG)
- Component Search Parameters (in UMM)
- Head Hunters
- Interconnected World of Components (with UMM descriptions)
- Set of Selected Components for Application

**System Generation and Validation Process**

- Set of Selected Components for Application
- System Assembly
- GDM for interoperability of heterogeneous components (in TLG)
- A Possible System (and UMM description)
- Instrumented with Event Grammars
- System Execution and Validation
- Test Cases from User
- Final System

**Case Study - Bank Account Management System**

- **Components**
  - CorbaAccountServer (CORBA) - Availability $\geq 90\%$, Response Delay $< 10$ ms
  - JavaAccountServer (RMI) - Availability $\geq 85\%$, Response Delay $< 30$ ms
  - JavaAccountClient (RMI) - Availability $\geq 90\%$, Response Delay $< 50$ ms

- **Query** - Create a bank account management system that has availability $\geq 50\%$ and response delay $< 100$ ms.
Summary of Approach

- Address key issues that need to be resolved to assist organizations to manage their distributed software systems
  - Meta-model allows a seamless integration of heterogeneous components
  - Formal specifications assist in automated construction and verification of parts and the whole of a distributed computing system (DCS)
  - Support a unified approach to iterative as a pragmatic solution for software development of DCS
  - Incorporation and validation of QoS implies the creation of more reliable DCS
  - Interactions with the industry and standards organizations provide practical feedback and enable proliferation of research results in a timely manner

Salient Features

- A meta-model and a unified approach
- QoS-based generative process
- Generation based on distributed resources in the form of components – use of HHs
- Event grammars for dynamic QoS metrics
- Automation (to the extent feasible) for system generation
Relationship to OMG Work

- Infrastructure & Interoperability
  - CORBA, CORBA Services, CCM, IIOP, COM/CORBA, SOAP/CORBA, CSIv2,
  - Head-hunters, Internet Component Broker
  - Validation Metrics / Instrumentation
- Model Driven Architecture
  - Consistent with our Meta-model approach
  - Common concept of using QoS parameterization (requires a QoS catalog)
  - Need to explore QoS considerations for Platform Independent Model (PIM) to Platform Specific Model (PSM) transformations

Contact Author at ECOOP

Barrett Bryant will be available throughout the conference for discussions about this poster. To arrange a specific time for discussion, please place a notice on the message board or send an e-mail to bryant@cis.uab.edu.

Webpage

http://www.cs.iupui.edu/uniFrame

Message Board