Components and Aspects for Embedded Middleware

Sean Walton (swalton@cs.utah.edu)
Venkat Chakravarthy (vchakra@cs.utah.edu)
Eric Eide (eeide@cs.utah.edu)
John Regehr (regehr@cs.utah.edu)
In collaboration with Ray Klefstad, UC Irvine

Problem
Need a resource-efficient, configurable middleware product line targeted for rapid embedded systems development.

Approach
Use components (Jiazz) & aspects (AspectJ) to modularize features within the product line architecture. Extend tools to address domain concerns including memory and real-time behaviors. Use RT Zen middleware to guide the design of a new, integrated components & aspects language.

RT Zen – CORBA 2.3-compliant open-source middleware
- Characterize and classify existing RT Zen code as components and aspects
- Refactor RT Zen with Jiazz & AspectJ
- Integrate new component and aspect model with RT Java support into RT Zen

RT Java
- “Componentize” or “aspectize” RTSJ idioms within RT Zen
- Support RTSJ concerns in components & aspects language

Jiazz – Java extension providing component definition and linking
- Design and implement model for combining components and aspects
- Support aspects that modify component assemblies
- Provide support for RT Java in model and integrate into Jiazz
- Create multidimensional component model for Jiazz