

# What's the Right Run-Time Infrastructure for RoSES?

Meredith Beveridge



Robust Self-Configuring Embedded Systems

## Build our own

- Strength: Meets needs exactly
- Weakness: Time spent on development, when off-the-shelf technology may be sufficient

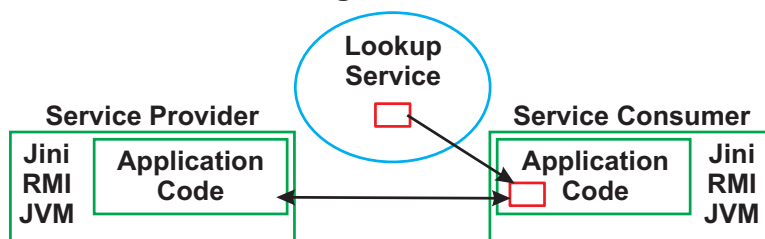
## CORBA

- Definition: Well-defined, platform-independent interfaces to connect heterogeneous systems
- Strength: Remote objects accessed as naturally as local objects
- Weaknesses:
  - Requires too much memory
  - Immature real-time capabilities
  - Optimized for TCP/IP
  - Doesn't support data-push; only does data-pull



## Jini

- Definition: Platform-independent, spontaneous federated networking built on Java RMI

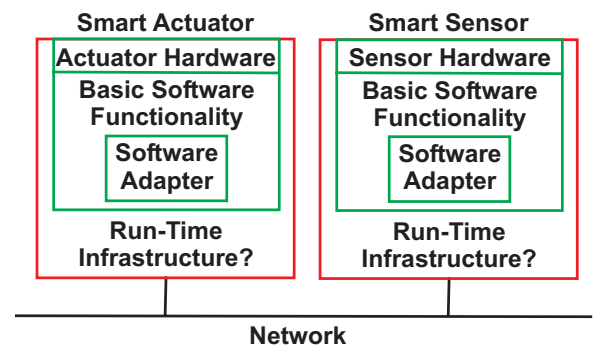


- Strengths:
  - Proxies allow protocol flexibility
  - Designed for embedded systems
  - Event-triggered, multicast model matches CAN
- Weakness:
  - Java on an 8-bit microprocessor??



## What RoSES Wants

- Multiple “smart” sensors/actuators connected to embedded real-time network
- Every sensor is a “server”
- Automatic software component allocation
- Real-time operation



## Plan of Attack

- Map Jini onto CAN, see what breaks
- Experiment: navigation software on distributed testbed
- Anticipated hurdles:
  - TCP/IP??
  - Fault tolerance?
  - Will JVM really get small enough, and will it stay small?
  - How do you do real-time scheduling with Java?
  - Can end-to-end timing be determined?
  - How do Jini messages fit into CAN messages?

## Jini on CAN?



Carnegie Mellon



Electrical & Computer ENGINEERING

