ARMADA: A Real-time Middleware Architecture for Distributed Applications

Farnam Jahanian
Real-Time Computing Laboratory
Department of EECS
University of Michigan

http://www.eecs.umich.edu/RTCL/armada

September 1997

Project Overview

Objective:

- Design, develop and demonstrate a software environment for building embedded real-time applications on distributed platforms.
- Transition research results to DoD labs and industry.

Current Thrust Areas:

- Real-Time Channel Communication Services
- Middleware Services
- Dependability Validation & Evaluation Tools
- Application Demonstration

Project Participants

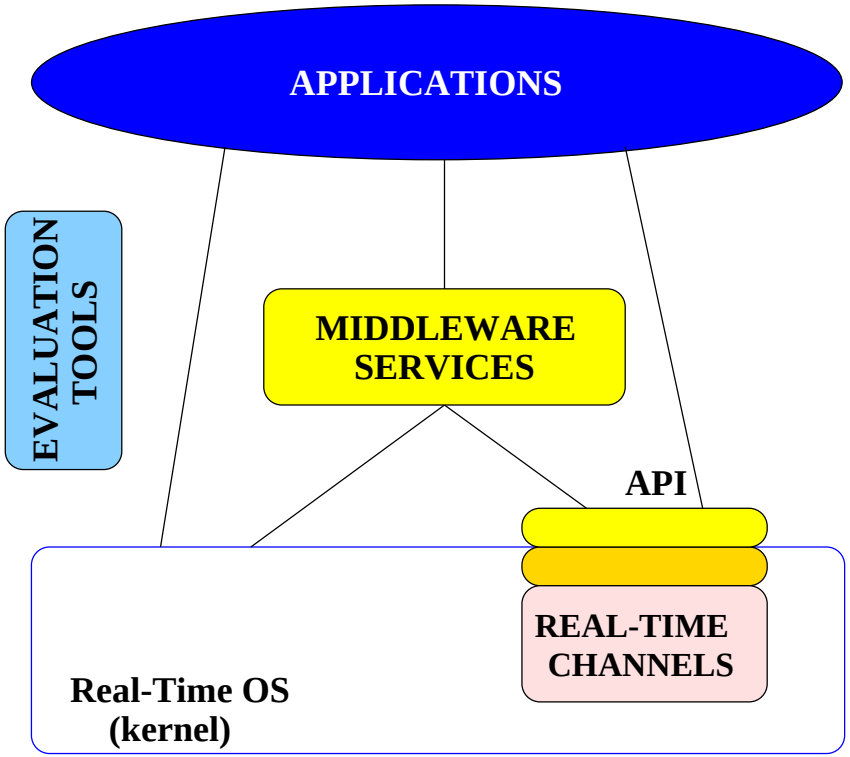
University of Michigan Real-Time Computing Lab:

- Kang Shin
- Farnam Jahanian
- Peter Chen

Honeywell Technology Center:

- Devesh Bhatt
- Jeremy Norton

Project Overview



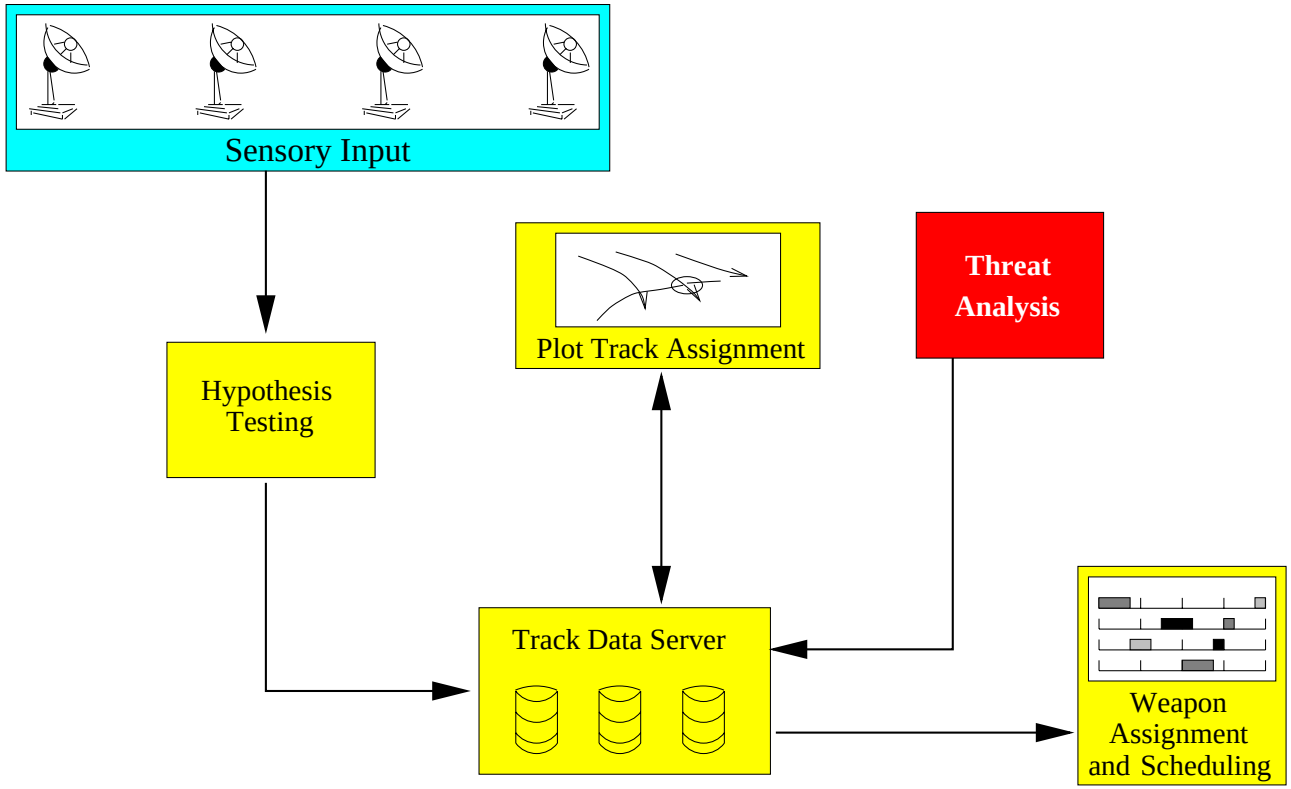
Target Applications

- Embedded fault-tolerant applications
- Industrial and manufacturing systems
- Distributed multimedia

Key Challenges:

- *Timely delivery* of services with end-to-end hard/soft real-time constraints
- *Dependability* of services in the presence of hardware/software failures
- *Scalability* of computation and communication resources
- *Exploitation* of open systems and emerging standards in operating systems and communication services

Tracking, Analysis, and Weapons Assignment Application



Requirements:

- End-to-end timing constraints on computation and communication tasks
- Service and data availability in the presence of faults

ARMADA Real-Time Channel Service

Service Paradigm:

- Simplex, fixed-route, unicast virtual connection with sequenced messages
- End-to-end deterministic delay and bandwidth guarantees
- Admission control, traffic enforcement, and resource scheduling

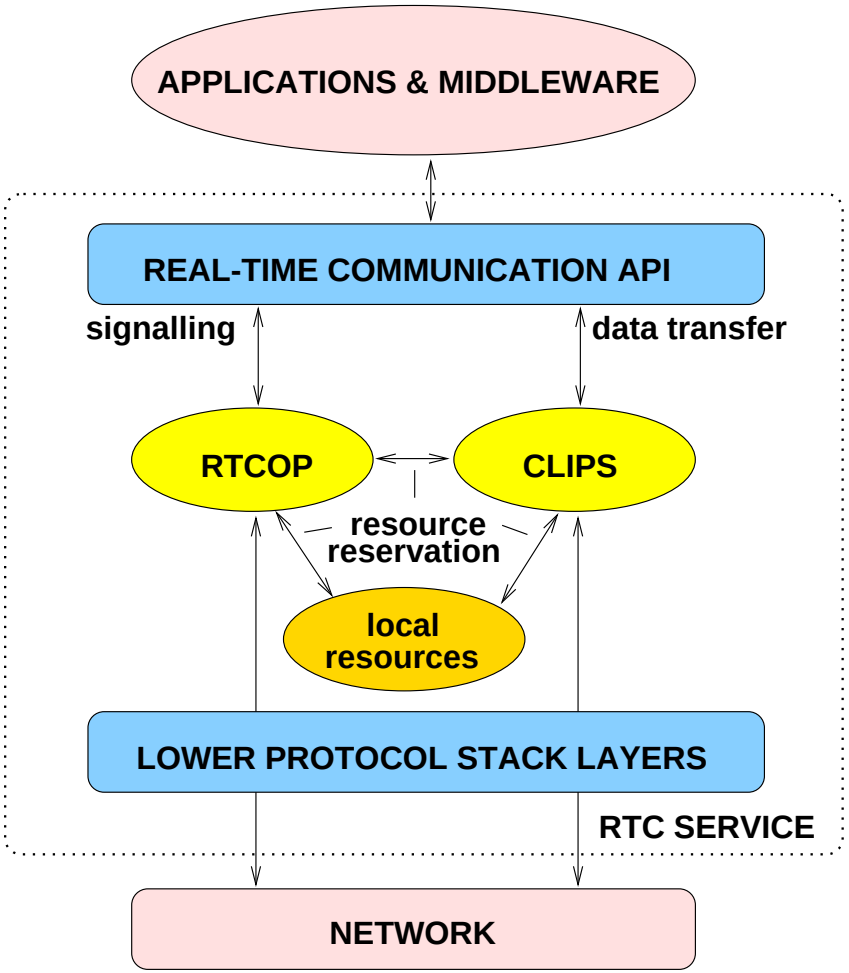
Service Usage:

- Three phases: channel setup, data transfer, channel teardown
- Traffic specification: message-based linear bounded arrival process
  - maximum message size
  - maximum message rate
  - maximum burst size

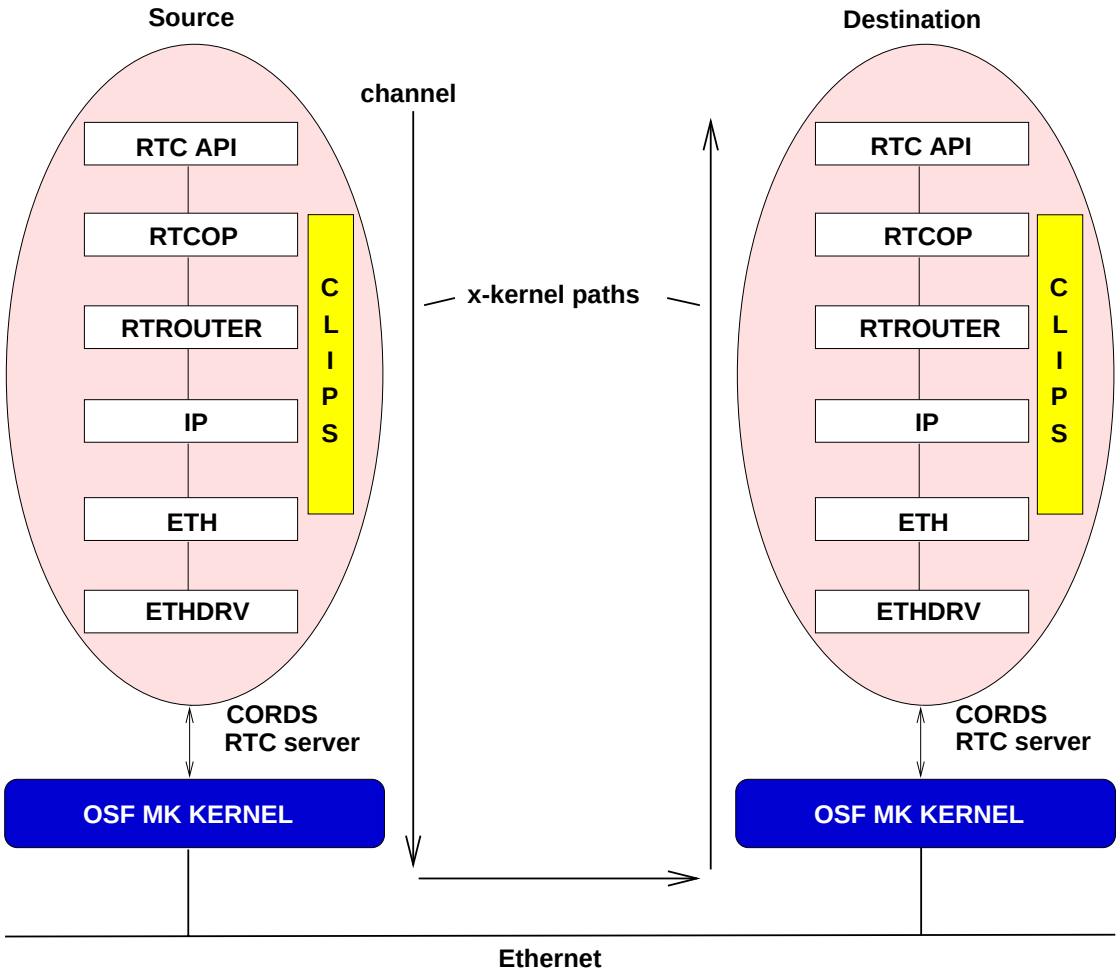
Service Implementation:

- Mach-based API for signaling and data transfer
- core functionality provided by *x*-kernel-based real-time channel server

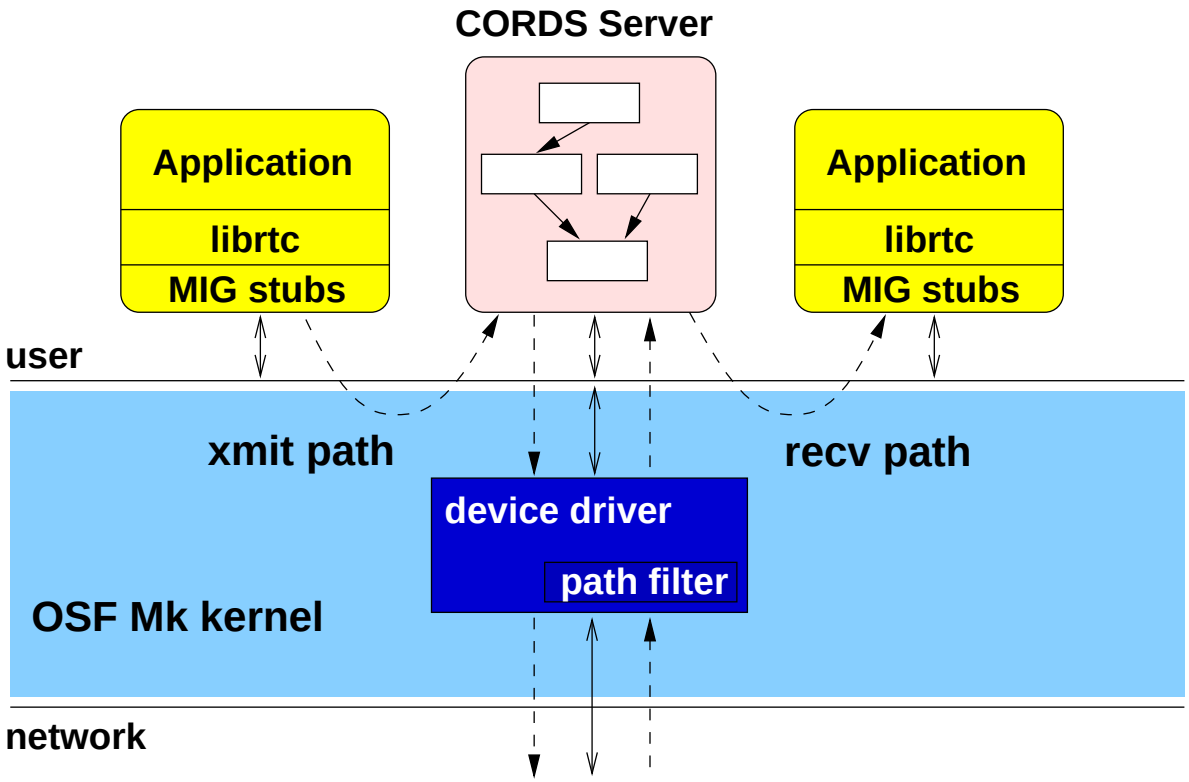
Software Architecture for Real-Time Communication



Real-Time Channel Protocol Stack



Server Implementation Using CORDS Framework



Key Middleware Services

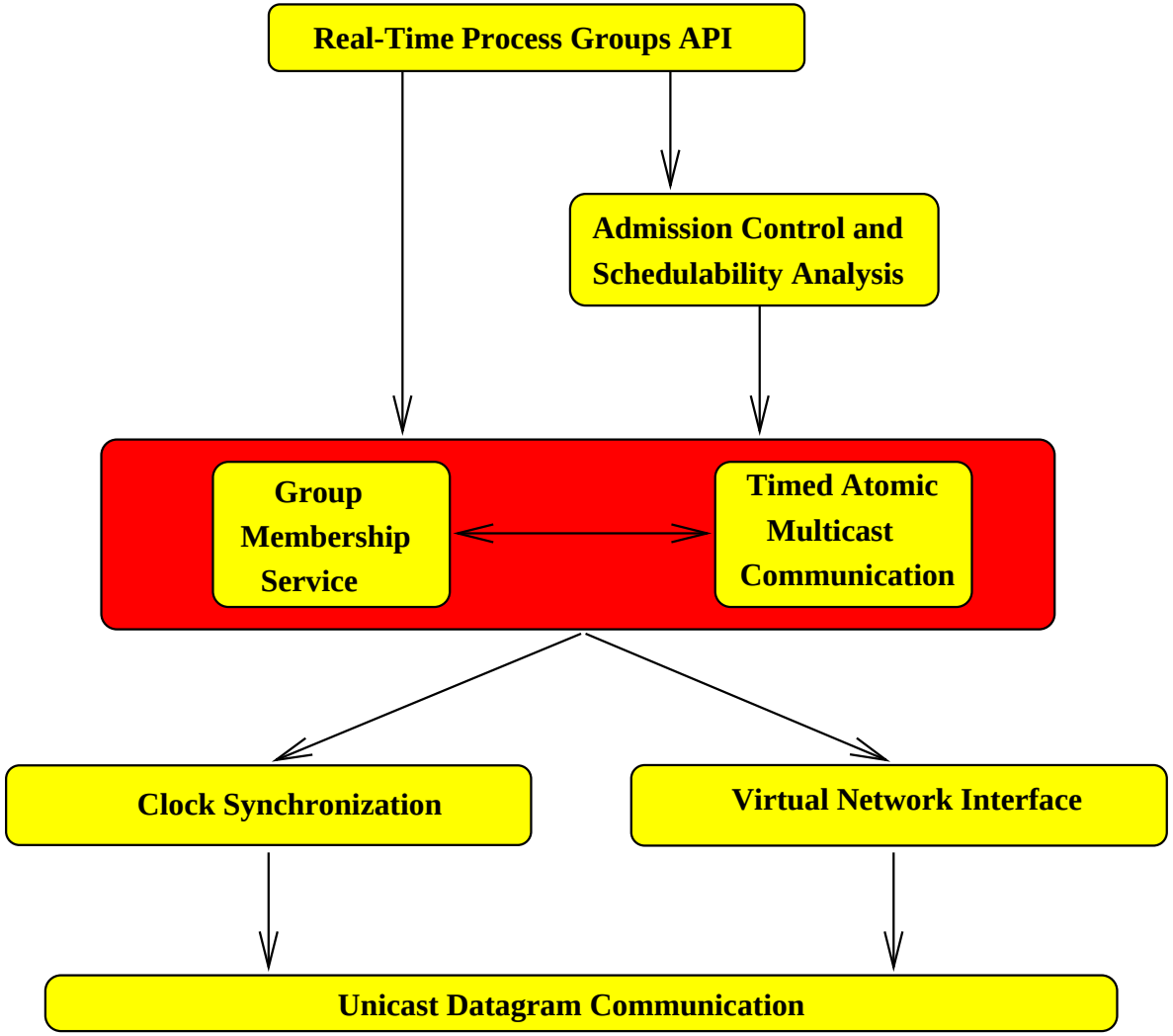
RTCAST - Real-Time Multicast for Group Communication:

- Supports soft or hard real-time multicast communication
- Provides atomic, ordered semantics
- Provides group status and membership service
- Maintains system consistency in the presence of failures
- Separates timing concerns from consistency concerns

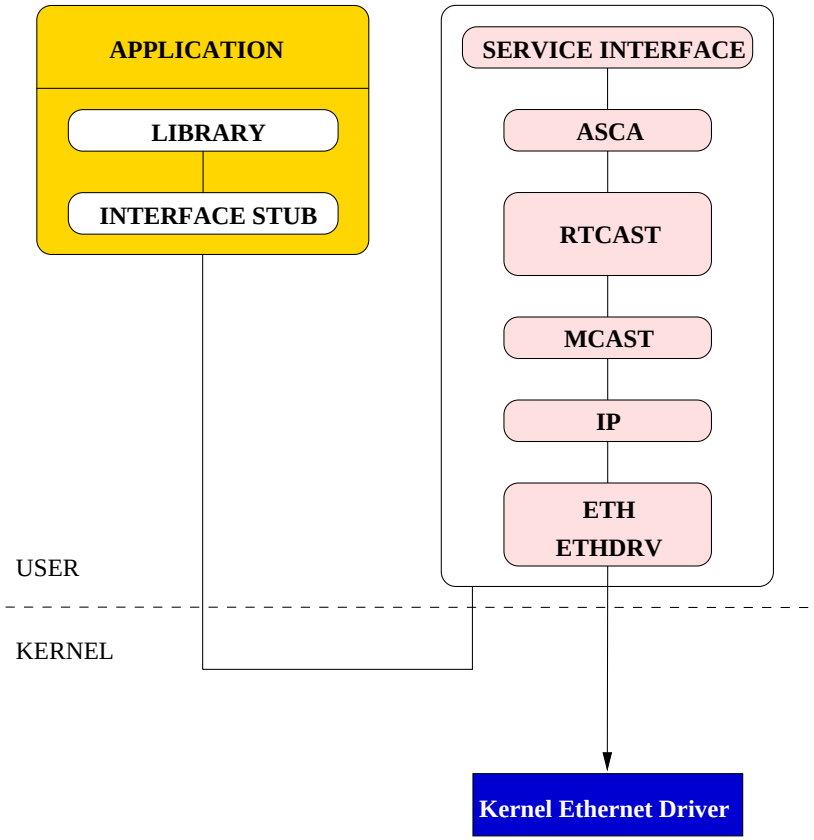
RTPB - Real-Time Primary-Backup Replication Service:

- Provides fault-tolerant, primary-backup replication
- Controlled inconsistency between primary and backup
- External and inter-object window consistency
- Fast response to the client

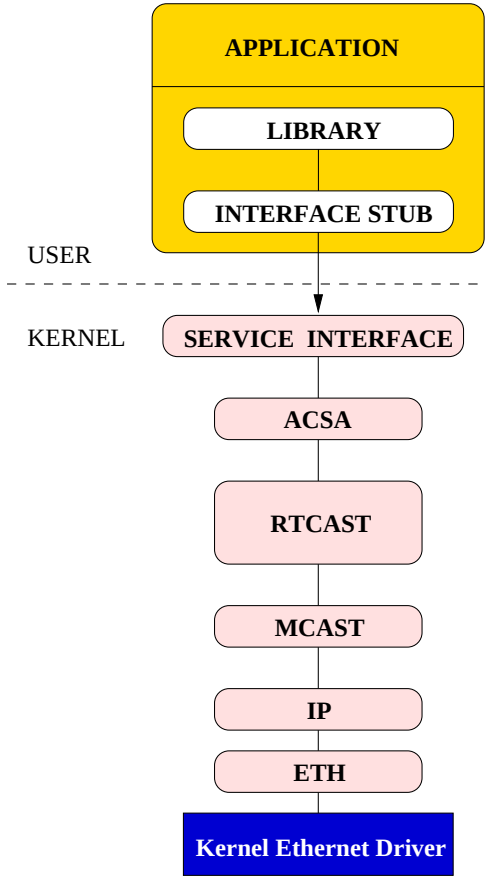
RTCAST Middleware Layering



RTCAST Protocol Stack

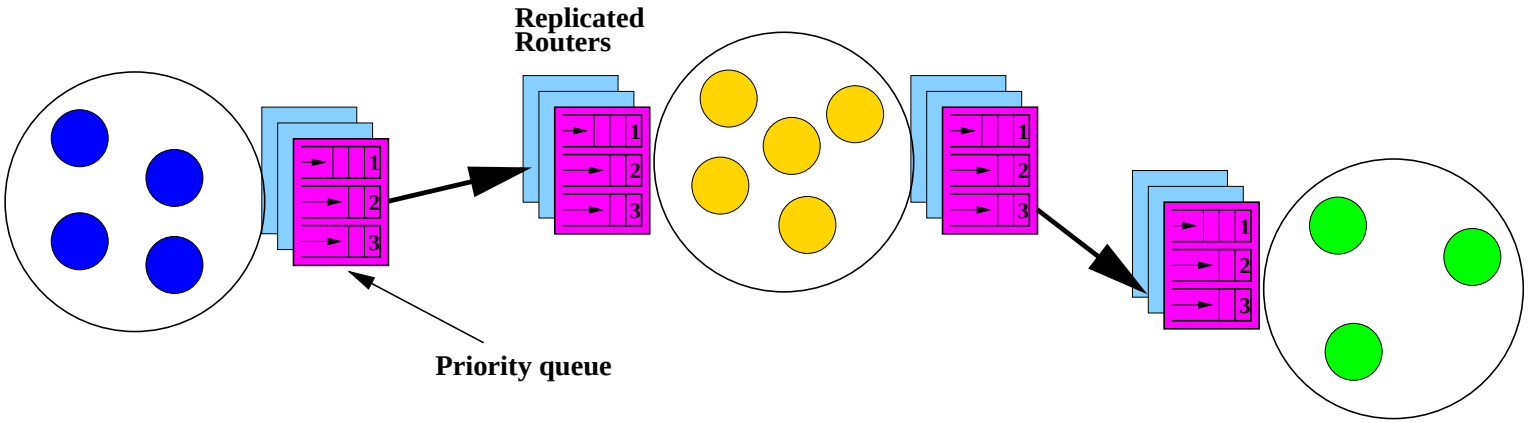


(a) CORDS User-level Server



(b) Split In-kernel CORDS Server

Scalable, Priority-based Inter-group Communication



- Construct scalable distributed systems by composing process groups.
- Address timeliness req. via priority-based intergroup communication.

ARMADA Evaluation Tools

ORCHESTRA: A fault injection tool for protocol implementations

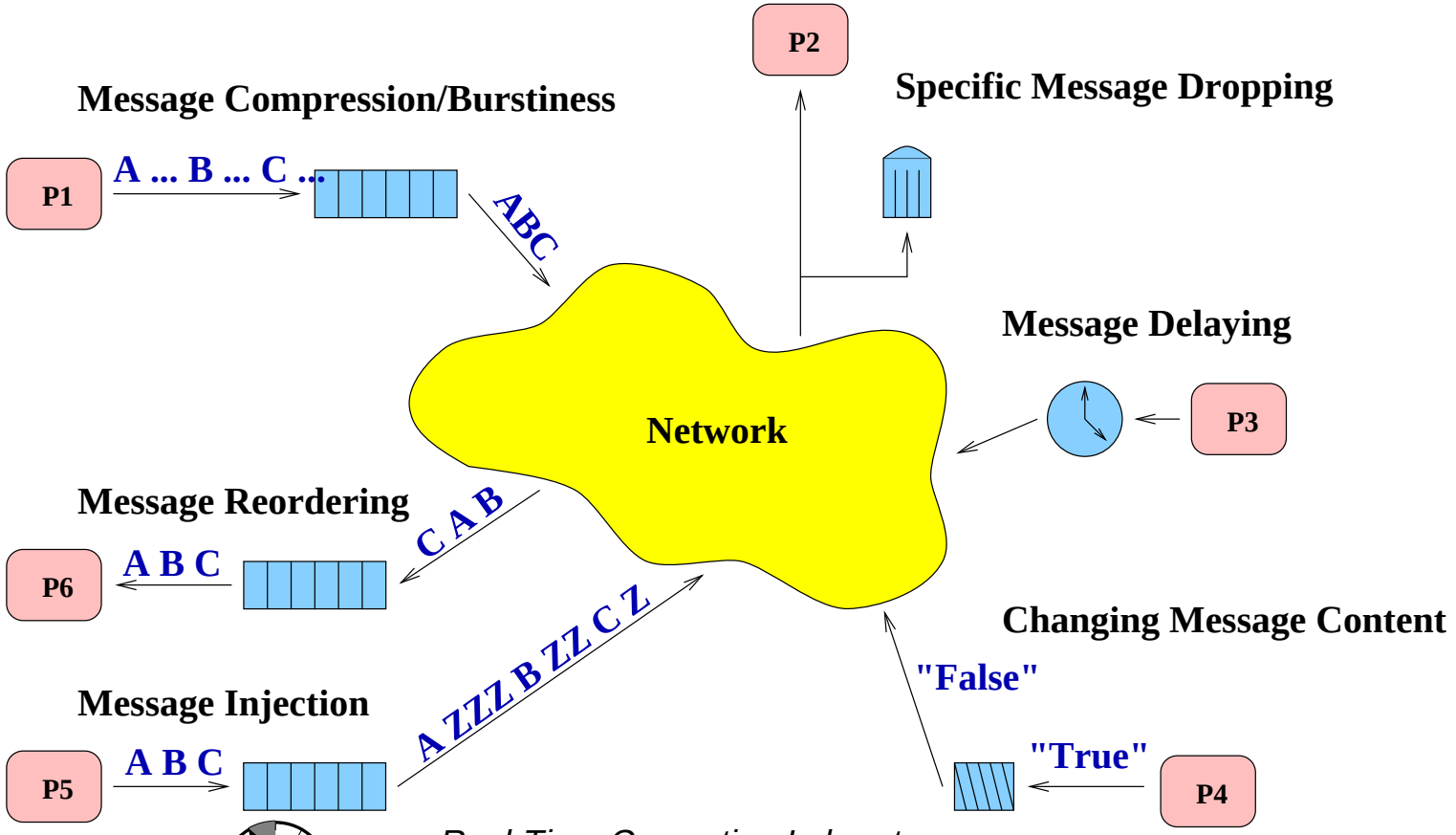
- Avoid instrumenting target protocol code
- Provide flexible/rapid test specification
- Give user deterministic control over messages
- Offer support for various fault models

COGENT: Controlled Generation of Network Traffic

- Generic client-server workload model
- Flexible specification of client and server resource usage

ORCHESTRA Fault Injection Environment

- A probing and fault injection environment for testing protocol implementations
- A framework that allows manipulation of messages exchanged between protocol participants



Current Project Status

- Implementation and demonstration of real-time channels on OSF MK 7.2. Includes support for signaling and data transfer.
- Implementation and evaluation of RTCAST middleware services on OSF MK 7.2 and CMU Mach-RT.
- Detailed parameterization and profiling of communication protocol stack.
- Design and Implementation of API libraries for RT Channel, RTCAST, and RTPB services.
- Migration of Orchestra fault-injection tool to *x*-kernel MK platform. Experimental evaluation of Open Group's GIPC and ARMADA middleware.
- Detailed specification of the tracking, analysis and weapon assignment application for demonstration and evaluation of ARMADA software technology.