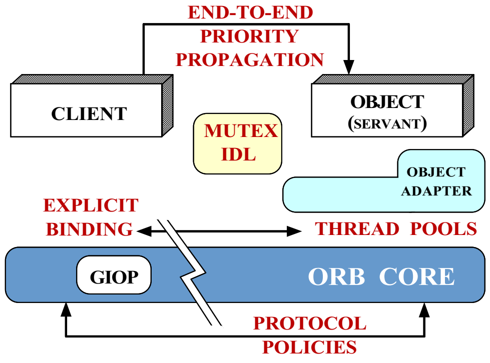
An Overview of the Joint Real-time CORBA Submission

Douglas C. Schmidt
schmidt@cs.wustl.edu

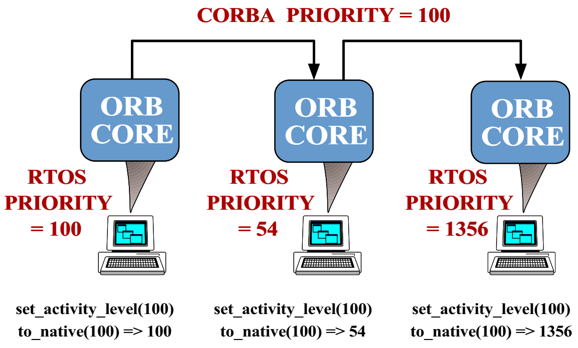
Washington University, St. Louis
www.cs.wustl.edu/∼schmidt/

Overview of Key Real-time CORBA Features



- **Features**
  - End-to-end priority propagation
  - Protocol policies
  - Thread pools
  - Explicit binding
  - Mutex IDL

End-to-End Priority Propagation



- **Features**
  - Priorities can propogate end-to-end
    - ∗ Supports heterogeneous RTOS priority mappings
    - ∗ Supports priority inheritance
  - Servers can also dictate priority

Protocol Policies

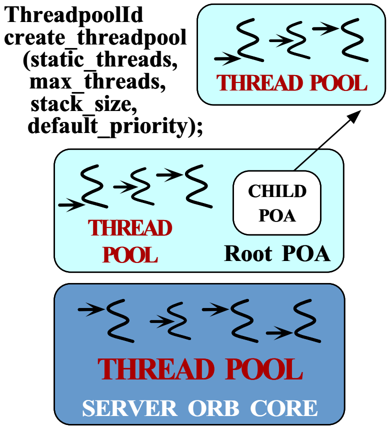
```
interface ProtocolProperties {};

typedef struct {
  IOP::ProfileId protocol_type;
  ProtocolProperties
    orb_protocol_properties;
  ProtocolProperties
    transport_protocol_properties;
} Protocol;
typedef sequence <Protocol> ProtocolList;

interface TCPProtocolProperties
  : ProtocolProperties
{
  attribute long send_buffer_size;
  attribute long recv_buffer_size;
  attribute boolean keep_alive;
  attribute boolean dont_route;
  attribute boolean no_delay;
};
```

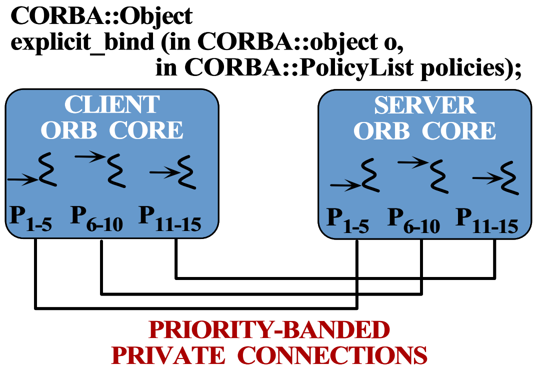
- **Features**
  - Select and configure communication protocols
    - ∗ *e.g.*, TCP socket options
  - Supports *ORB protocol* and *transport protocol* configuration
  - Ordering in `ProtocolList` indicates preferences

Thread Pools



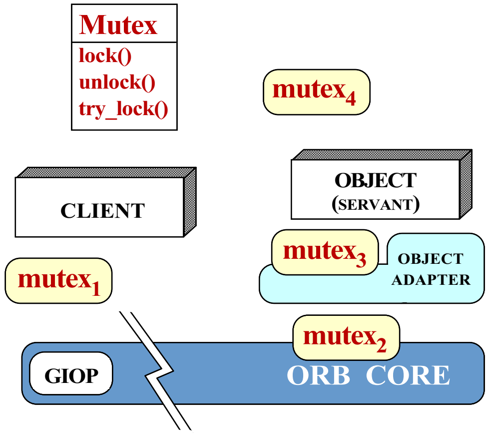
- Features
  - Pre-allocate threads and thread attributes
    - ∗ Stacksize
    - ∗ Static threads
    - ∗ Maximum threads
    - ∗ Default priority
  - Applicable at both the ORB and POA level

Explicit Binding



- Features
  - Enables pre-establishment of connections
    - ∗ Priority-banded connections
    - ∗ Private connections
    - ∗ Protocol policies

Mutex IDL



- Features
  - A portable Mutex API
    - ∗ *e.g.*, `lock`, `unlock`, `try_lock`
  - Necessary to ensure consistency between ORB and application synchronizers
  - Locality constrained