Lightweight Remote Procedure Call

Bershad, Anderson, Lazowska, and Levy
LRPC?

- RPC optimized for local communication.
- Why do we want to do local RPC?
Overview

- Why RPC is good.
- Why local RPC is slow.
- How LRPC optimizes RPC.
- Performance of LRPC vs RPC.
RPC: The Good.

- Abstraction
- Encapsulation
- Convenience
- Efficient
RPC Overhead

- Stubs
- Message copying
- Access validation
- Message transfer

- Scheduling
- Context Switching
- Dispatch
LRPC

- Binding
- Calling
- Stub Generation
- Multiprocessors
- Argument Copying

Client Domain

Server Domain

Shared Memory

Kernel

Client thread

Arg0

Arg1

Arg2
RPC Overhead

✓ Stubs
✓ Message copying
✓ Access validation
✓ Message transfer

✓ Scheduling
• Context Switching
✓ Dispatch
Performance

The graph shows the time in microseconds for different procedure calls. There are two main categories: RPC (Optimal) and LRPC (Optimal). The data points for RPC (Optimal) and LRPC (Optimal) are represented by diamonds and squares, respectively. The graph also includes data points for RPC (Measured) and LRPC (Measured), represented by circles and triangles, respectively.
Questions?