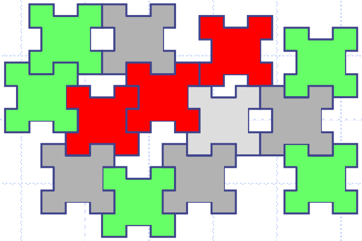
May 2006



RMIX: A Dynamic, Heterogeneous, Reconfigurable Communication Framework

Christian Engelmann[1,2] and Al Geist[1]

[1]Computer Science and Mathematics Division
Oak Ridge National Laboratory, Oak Ridge, TN, USA

[2]Department of Computer Science
The University of Reading, Reading, UK

International Conference on Computational Science (ICCS) 2006

Office of Science
U.S. DEPARTMENT OF ENERGY

Motivation

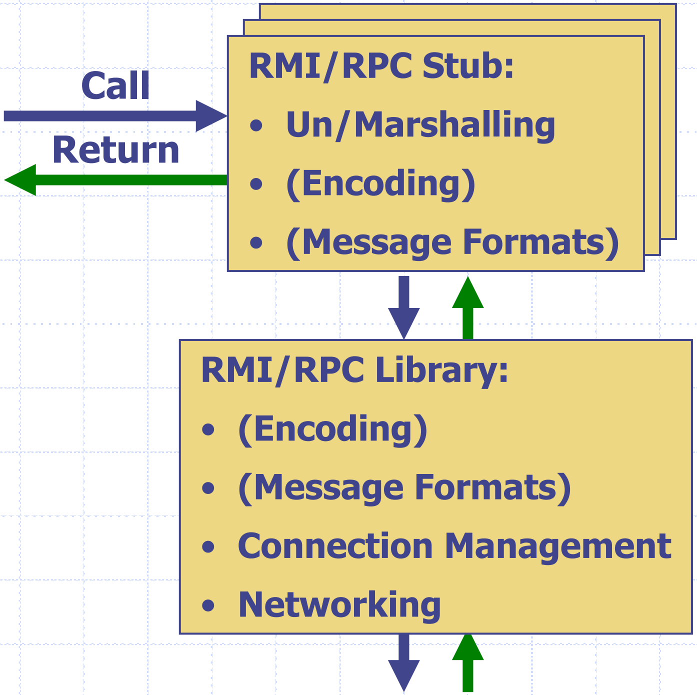
- Remote Method Invocation (RMI) is the most important communication paradigm for heterogeneous, distributed, collaborative environments.
- RMI is an object oriented analogy to the Remote Procedure Call (RPC) concept.
- Traditionally, RMI/RPC communication systems implement one specific protocol stack only.
- However, heterogeneous, distributed, collaborative environments may use multiple protocols.
- For example, two sites may communicate with each other using SOAP, while they use RPC internally.

RMIX Concept

- RMIX is a dynamic, heterogeneous, reconfigurable, multi-protocol communication framework.
- It allows software components to communicate using various RMI/RPC protocols, such as SOAP and RPC.
- The RMIX framework core offers functions common to all protocols: networking and thread management.
- RMIX utilizes plug-ins for protocol specific connection management, message formats, and data encoding.
- Object stubs only perform an adaptation to RMIX.
- Support for protocol switching, advanced RMI/RPC semantics, and customized/optimized protocols.

RMIX Framework Approach

- Traditional RMI/RPC systems:



Call

Return

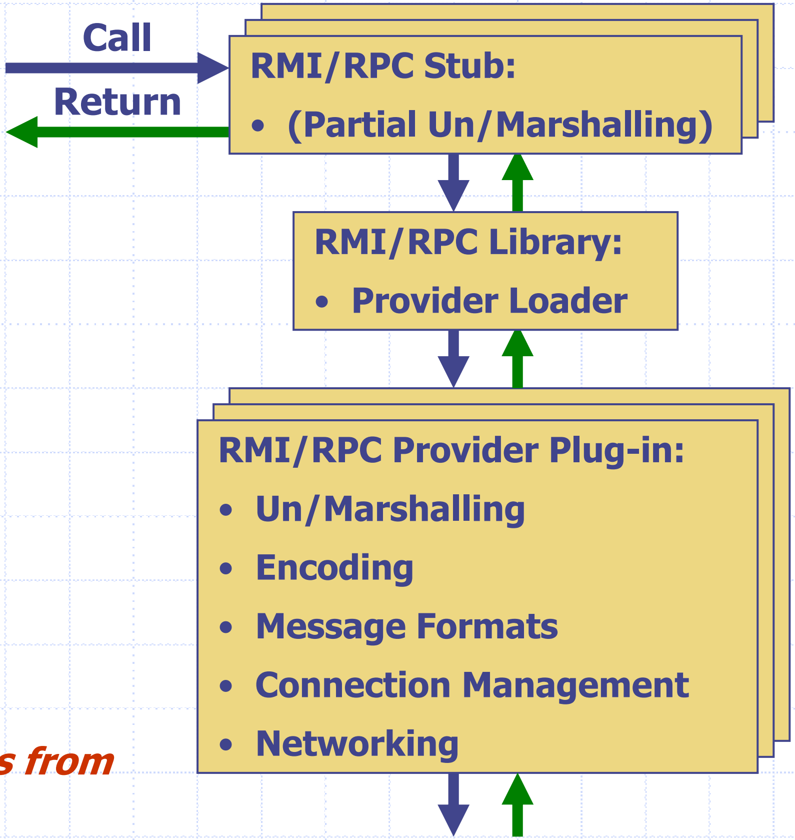
RMI/RPC Stub:

- Un/Marshalling
- (Encoding)
- (Message Formats)

RMI/RPC Library:

- (Encoding)
- (Message Formats)
- Connection Management
- Networking

- RMIX Framework:



Call

Return

RMI/RPC Stub:

- (Partial Un/Marshalling)

RMI/RPC Library:

- Provider Loader

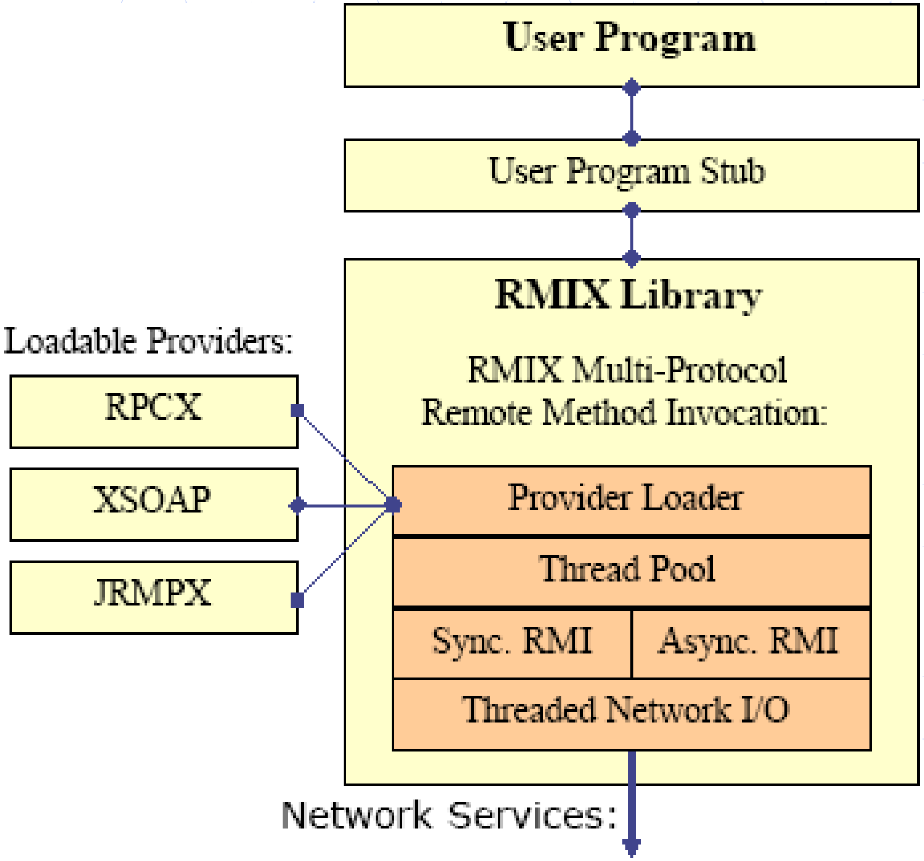
RMI/RPC Provider Plug-in:

- Un/Marshalling
- Encoding
- Message Formats
- Connection Management
- Networking

Moving protocol specific functions from stubs into pluggable providers.

RMIX Architecture

- RMIX base library:
  - Automatically un-/loads protocol providers.
  - Performs multithreaded communication and method execution.
  - Offers synchronous and asynchronous RMI.
- RMIX providers:
  - Supply protocol specific connection management, message formats, and data encoding.
- Stubs adapt to the protocol-independent RMIX interface.

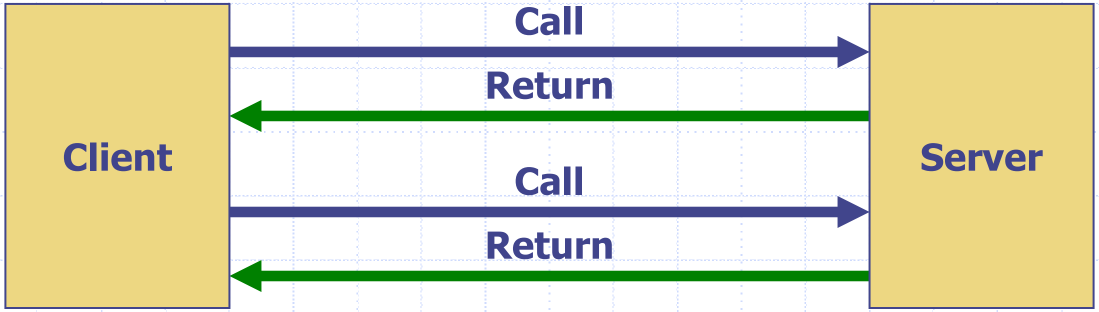


RMIX Prototypes

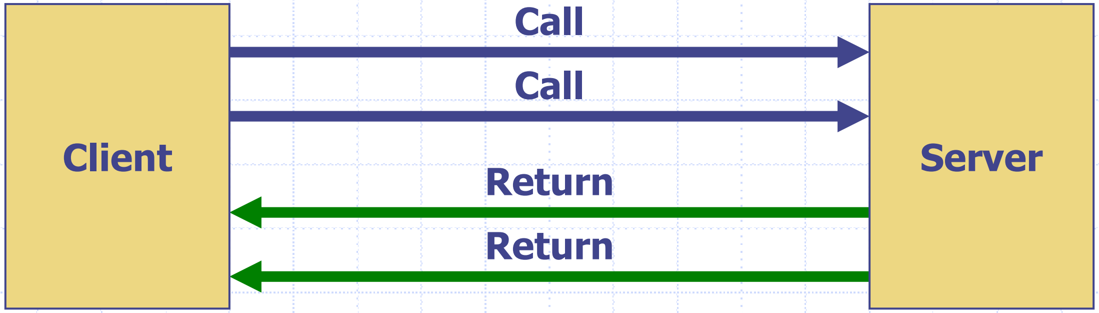
- First RMIX variant was developed earlier in Java by our partner team at Emory University.
  - An object-oriented RMI-based solution that translates RMI to RPC when needed.
- Second RMIX variant has been developed in C at Oak Ridge National Laboratory (*this paper/presentation*).
  - A modular RPC-based solution that translates RPC to RMI when needed.
- Both variants are compatible as they use standard RMI/RPC protocols and advanced RMI/RPC semantics.
- RMIX has been integrated into Harness as a plug-in.

Advanced RMI Semantics

- Standard synchronous RMI calls:



- Advanced asynchronous RMI calls:

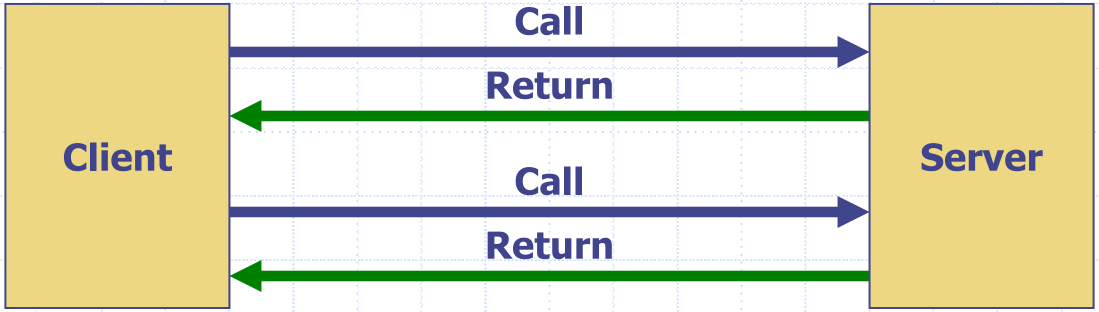


Asynchronous RMI

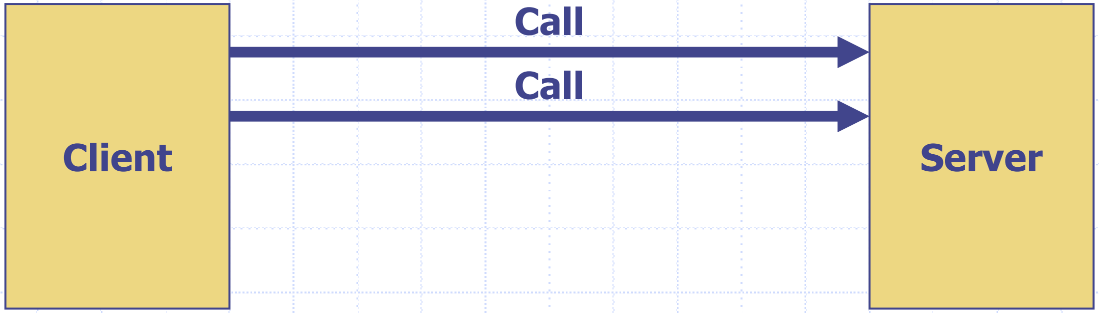
- A separate RMI interface allows the caller to continue after the request has been sent and accepted.
- The caller obtains an invocation reference in order to retrieve the response later.
- The client-side provider uses a separate thread to wait for the response and to store it locally.
- Multiple invocations may be interleaved, i.e., called in succession without retrieving the response in between.
- The server-side protocol guarantees invocation order.
- Asynchronous RMI needs to be supported by the protocol provider.

Advanced RMI Semantics

- Standard synchronous RMI calls:



- Advanced one-way RMI calls:



One-Way RMI

- Another separate RMI interface allows the caller to continue after the request has been sent and accepted.
- Any response is eliminated at the client-side provider plug-in in a separate thread to maintain RMI/RPC protocol compliance.
- One-way RMI needs to be supported by the protocol provider.

RMIX Remote Object Registry

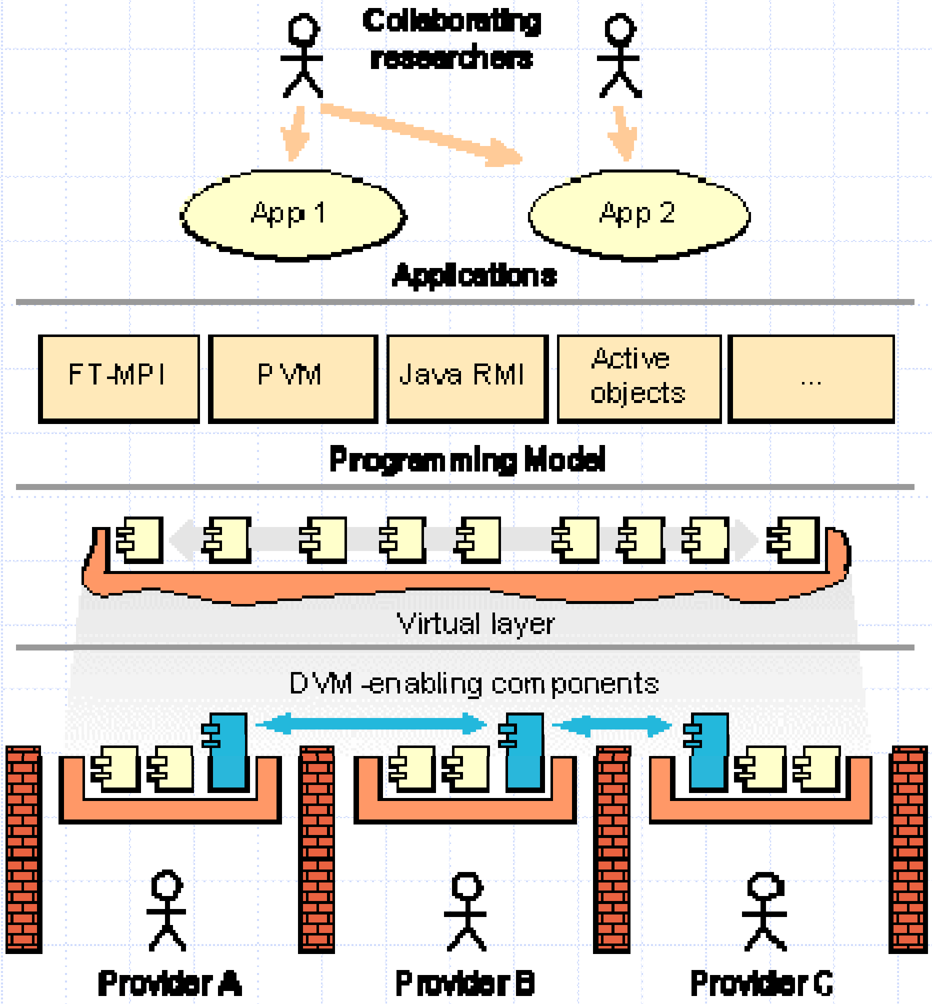
- A name server style registry to dynamically associate remote object references with names.
- Like Java RMI registry, but with multi-protocol support.
- The RMIX registry is itself just a name server object that is exported using RMIX.
- Textual representation of remote object references for command line tools, such as the RMIX registry client.
  - PROTOCOL=RPC OBJECTID=42
  - PROTOCOL=RPC OBJECTID=42 HOST=192.168.1.1
  - PROTOCOL=RPC OBJECTID=42 HOST=192.168.1.1 PORT=5555
  - PROTOCOL=RPC OBJECTID=42 BINDING=X

What is Harness

- A pluggable, reconfigurable, adaptive framework for heterogeneous parallel and distributed computing.
- Allows aggregation of resources into high-capacity distributed virtual machines.
- Provides runtime customization of computing environment to suit applications needs.
- Enables dynamic assembly of scientific applications from (third party) plug-ins.
- Offers highly available distributed virtual machines through distributed control.
- Implementations in C and Java.

Harness Architecture

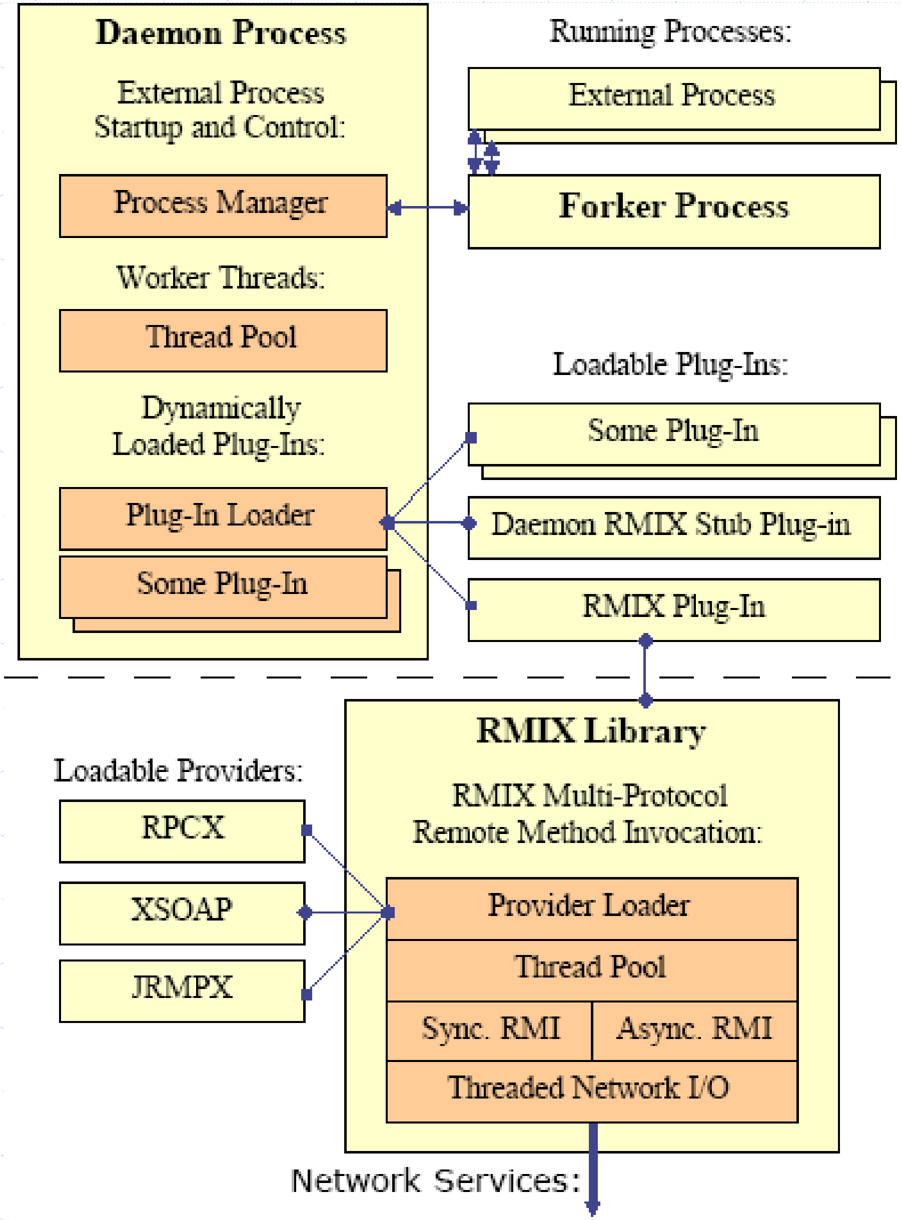
- Light-weight runtime environments (RTEs) share their resources.
- Plug-ins offer services.
- Support for diverse programming models.
- Distributed Virtual Machine (DVM) layer.
- Highly available DVM using distributed control.
- Highly available plug-in services via DVM.



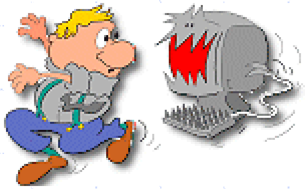
Harness-RMIX

- Replacement for HCom plug-in.
- Harness-RMIX plug-in wraps around RMIX base library.
- Harness plug-ins provide client and server stubs.
- Harness RTE stub plug-in for remote RTE communication.
- RMIX reuses Harness plug-in and thread management.

- Harness plug-ins are able to communicate via RMIX.
- Harness-RMIX further improves flexibility and heterogeneity.



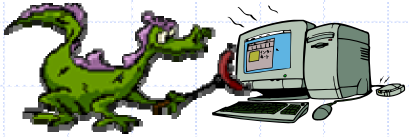
Current Status



- Stand-alone C-based RMIX variant is fully functional.
- TCP/IP ONC RPC protocol provider is fully functional.
  - ~500 microseconds for local RPC call on eth device.
- Other C-based providers, like SOAP, are in development.
- Harness-RMIX plug-in is fully functional.
- Harness uses RMIX for coordination and communication.
- Recent research in parallel plug-in programming paradigms using Harness-RMIX (see HPCC 2006 paper).

- http://www.csm.ornl.gov/~engelman/software

Summary



- RMIX is a dynamic, heterogeneous, reconfigurable, multi-protocol communication framework.
- The RMIX framework core provides functions that are common to all protocol stacks.
- Plug-ins provide protocol stack specific functions.
- Object stubs only perform an adaptation to RMIX.
- Support for advanced RMI/RPC semantics, such as asynchronous and one-way invocation.
- Implementations are available in C and Java.
- RMIX has been integrated into Harness as a plug-in.

May 2006

Questions or comments?

Christian Engelmann[1,2] and Al Geist[1]

[1]Computer Science and Mathematics Division
Oak Ridge National Laboratory, Oak Ridge, TN, USA

[2]Department of Computer Science
The University of Reading, Reading, UK

International Conference on Computational Science (ICCS) 2006