Feasibility to replace Interprocess Communication by the Message Passing Interface in microkernel contexts

René Drolshagen
Lasse Löffler

Overview

- Introduction
- Message Passing Interface (MPI)
- Interprocess Communication (IPC)
- Comparison
- Feasibility
- Conclusions

Introduction

- IPC is the communication standard in the microkernel context
- IPC was introduced 35 years ago
- Since this day IPC was only improved and never replaced
- This presentation will do a comparison of both

History, concept and functionality of the

MESSAGE PASSING INTERFACE

Message Passing Interface (MPI)

- MPI is a specification which tries to solve the problem between portability, efficiency and functionality

- Was developed for distributed shared memory (DSM) architectures

- As trends changed towards NUMA and NoRMA this specification was adapted

Message Passing Interface (MPI)

- Today MPI could handle all architectures seamlessly and transparently

- Reasons for using MPI:
  - Portability
  - Standardization
  - Performance
  - Functionality

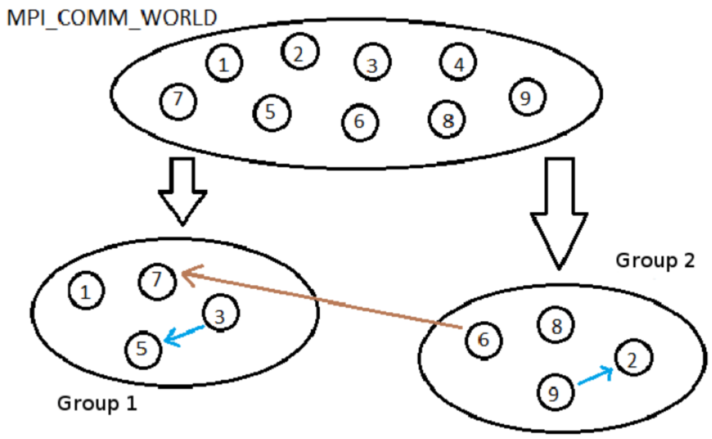
Message Passing Interface (MPI)

- Communication Methods in MPI:
  - Blocking send / blocking receive
  - Non-blocking send / non-blocking receive
  - Combined send and receive

- Messages are received in the order they were sent by guarantee

Message Passing Interface (MPI)

- One defined Communicator per Group
  - Handles the whole communication
- Can be compared with the Clans and Chiefs Model



History, concept and functionality of

INTERPROCESS COMMUNICATION

Micro- vs. Monolithic-Kernel

- Operating monolithic kernels are divided in user- and kernelspace

- Issue
  - all the basic services are running in privileged mode

- Idea of the microkernel
  - Moving all the basic services from kernel- to userspace

Idea and usecase of IPC

- Problem
  - Communication between service inside the user space
- Solution
  - Interprocess Communication (IPC) via the microkernels kernelspace

Functionality of IPC

- Passing unbuffered messages between services

- Two fundamental operations
  - send
  - receive

- Both operations can be executed
  - Blocking
  - Non-blocking

Interprocess Communication (IPC)

- IPC needs a high performance
  - Passing thousands of messages

- Also used inside the kernel to handle
  - Interrupts
  - Memory-Managing

- IPC Permissions / Addresses managed via Capabilities

MPI vs. IPC

COMPARISON

Comparison

- MPI allows n-to-1 and 1-to-n operations
- Asynchron MPI send operations could possibly sleep forever

- IPC has a defined message structure
- MPI messages are defined by the data which is passed inside the message

- MPI has the ability of process groups
- IPC is much faster (!)

Replacing IPC by MPI

FEASIBILITY

Feasibility

- **Problem:** No Capability feature in MPI
  - Every process could communicate with each other

- **Solution:**
  - Group processes together which are allowed to communicate
  - Only communication inside groups is allowed

- **Example:**
  - Group 1: process A and process B
  - Group 2: process A and process C

Feasibility

- **Problem:** No Capability feature in MPI
  - Every process could communicate with each other

- **Solution:**
  - The communicator of a group has the ability to check whether a communication is allowed
  - Implement a kind of rights table

Feasibility

- **Problem:** Denial-of-Service
  - A malicious user-level program could send a unlimited amout of non-blocking messages

- **Solution:**
  - Forbid non-blocking messages

Feasibility

- **Problem:** Starving by blocking messages
  - A performed blocking send operation could starve a process, if the receiver never appears

- **Solution:**
  a. Implement a timer in messages
  b. Give the communicator the ability to break down the send operation
     - Causes dramativ overhead and implementation work (!)

Feasibility

- **Problem:** Performance and amount of Code
  - MPI has got a huge amount of Code
  - The performance of MPI compared to IPC

- **Solution:**
  - Not available

THE

CONCLUSION

Conclusion

- Replacing IPC by MPI is not advisable
- Some problems could easily be solved
- The Denial-of-Service attack is a critical issue

- There is a microkernel with MPI, which is called PARAM9000

Questions?

THANKS FOR YOUR ATTENTION