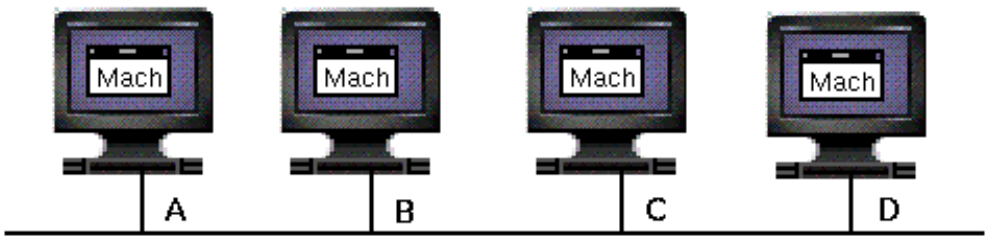
Unit 14: The Mach Operating System

14.4. The Shared Objects Net-interconnected Computer (SONiC)

The Shared Objects Net-interconnected Computer (SONiC)

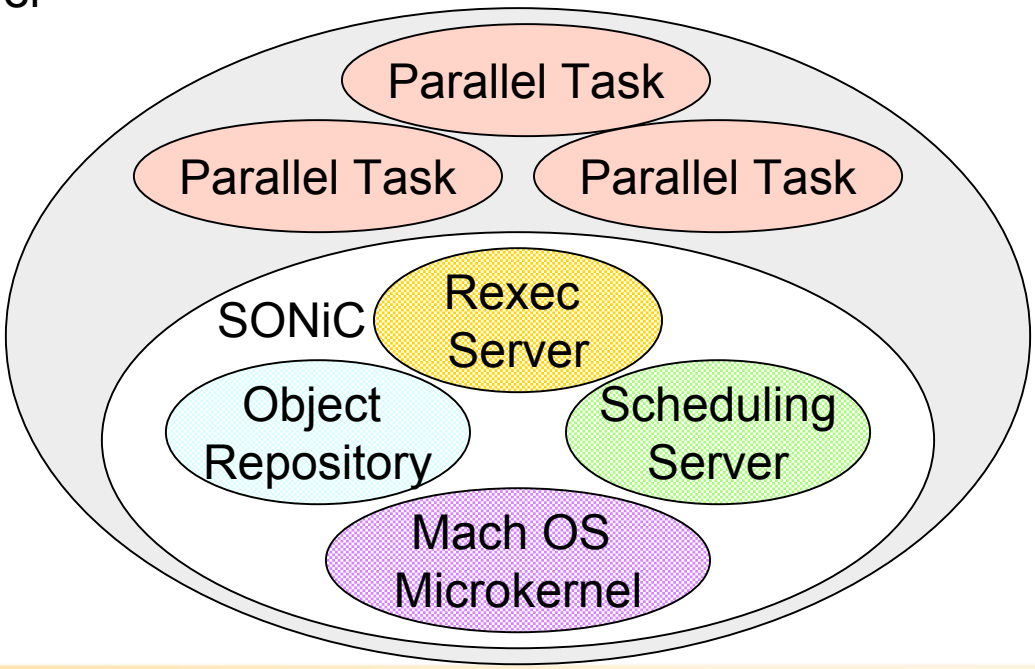
- Parallel Computing in Networks of Workstations (NOW)
  - Spare computing capacity / redundancy
  - Object-based distributed shared memory (DSM) / Ease-of-Use
- Shared Objects – Communication and Synchronization
  - Remote Execution Service - fork/join-Parallelism
  - Programming with replicated C++ objects
- Resource sharing among
  - Interactive users / parallel computations
- Commercial off-the-shelf systems (COTS)
  - Standard system software: Mach, Windows NT/2000



Research project at the Computer Arch. and Comm. Group

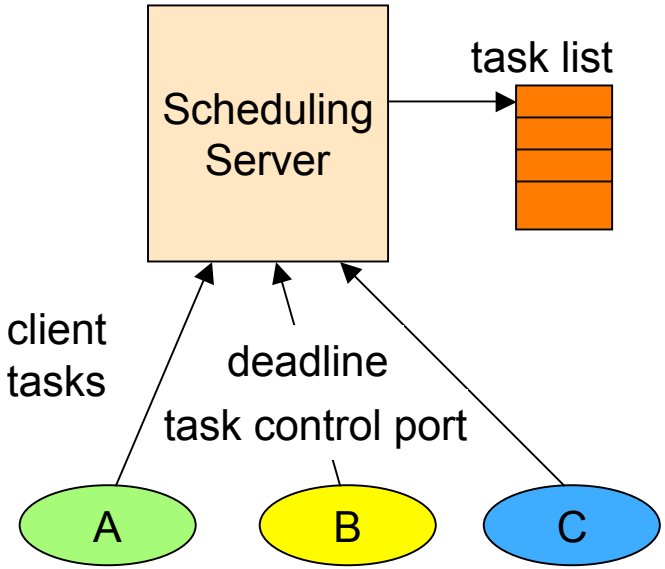
Structure of the SONiC Runtime System

- Mach Microkernel provides a sound basis:
  - Networking implemented by user space-servers
  - Mach supports multiple scheduling policies and provides access to the scheduler
  - Modern OS



The Scheduling Server Approach

- High-priority process manipulates dynamically priority of client processes
  - Based on fixed priority scheduling-policy
  - handoff scheduling - hints to the system scheduler



Scheduling Server implements:

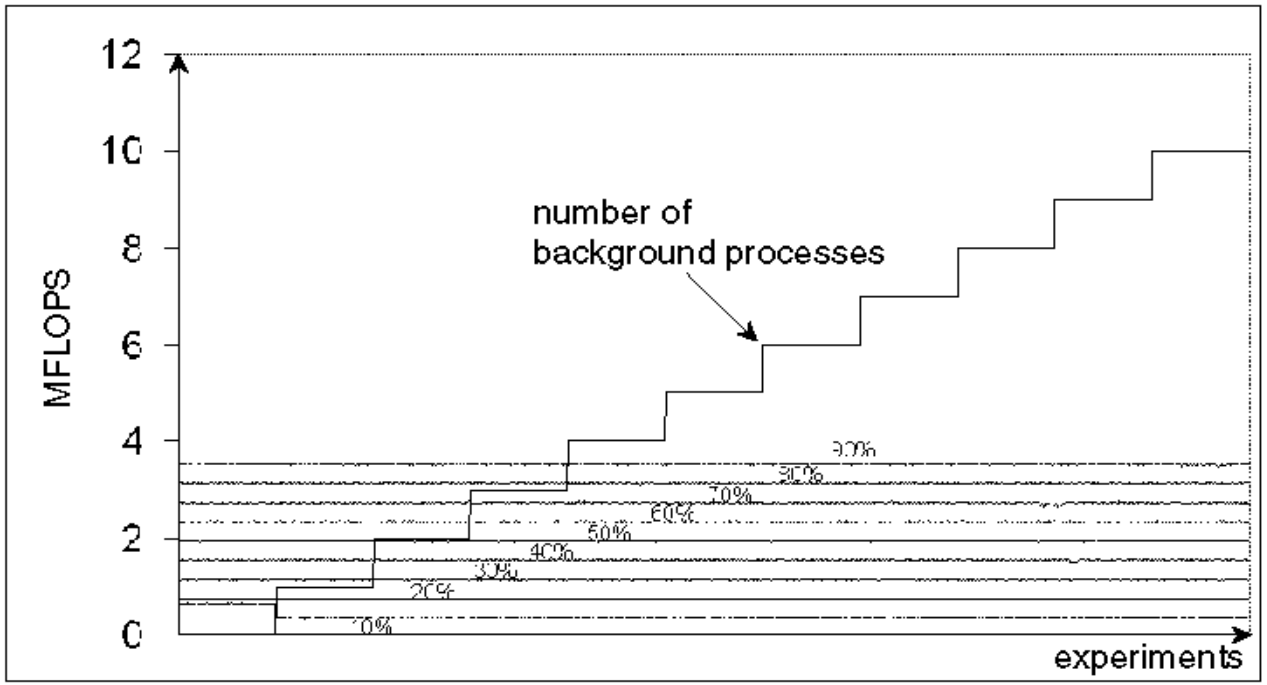
- *Round Robin*
- *Earliest Deadline First* (EDF)
- *Rate Monotonic Scheduling* (RMS)

ensures interactive availability!

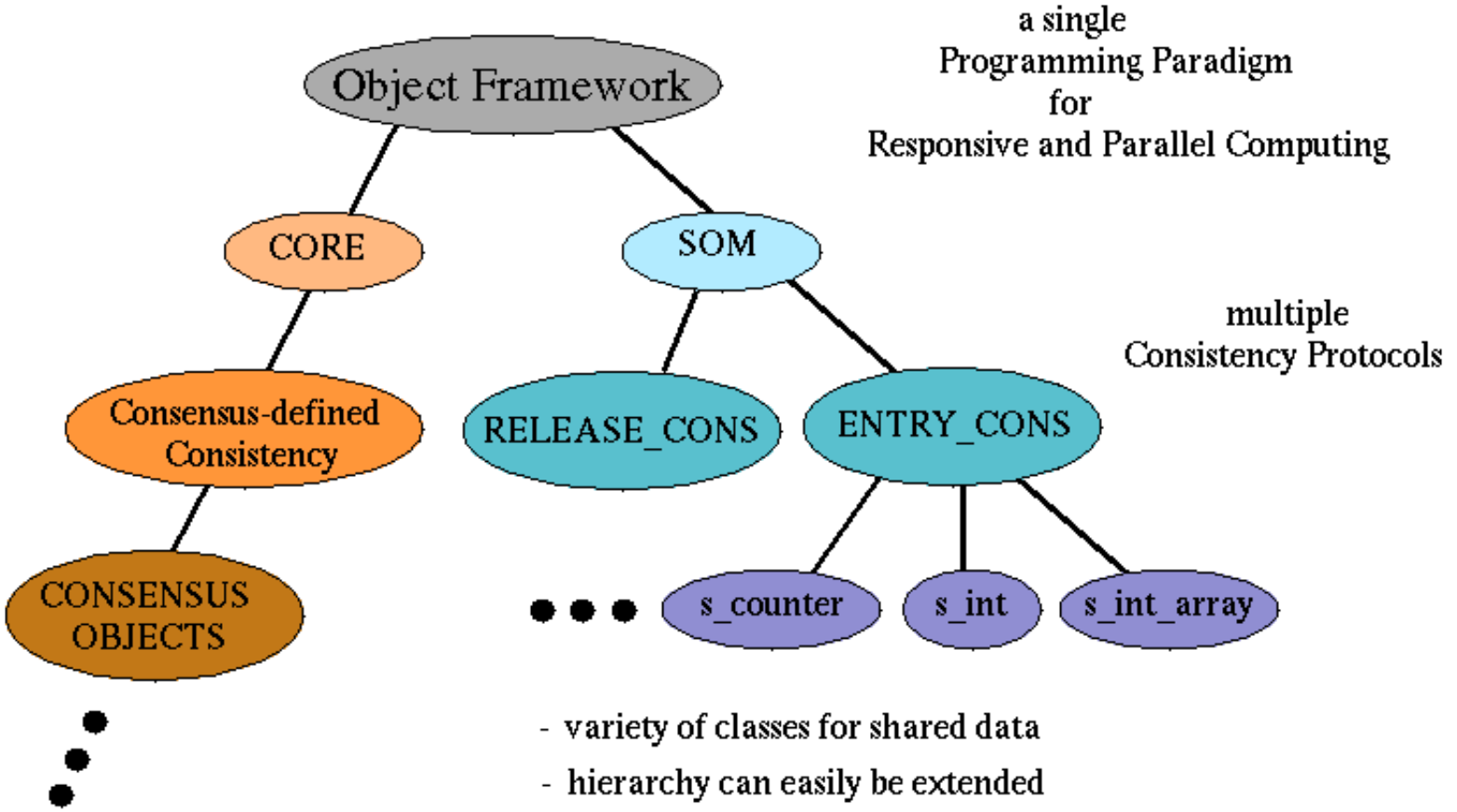
Without changes to operating system kernel

Scheduling Server: Stability with little Overhead

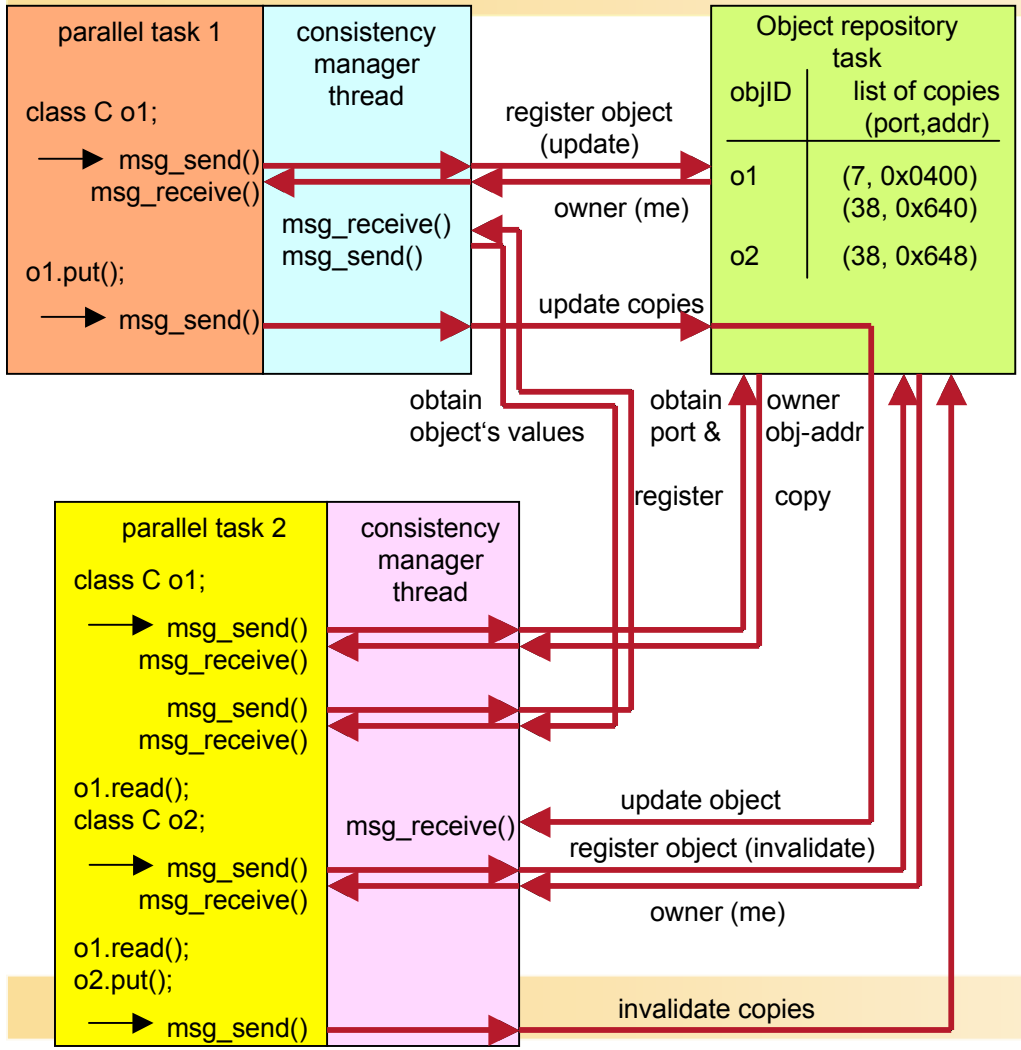
- Implementation based on Mach OS (NeXTSTEP), HP PA-RISC
- Little impact of varying background/disk-I/O loads
- Overhead less than 10%, typically 5%



The Programmers View

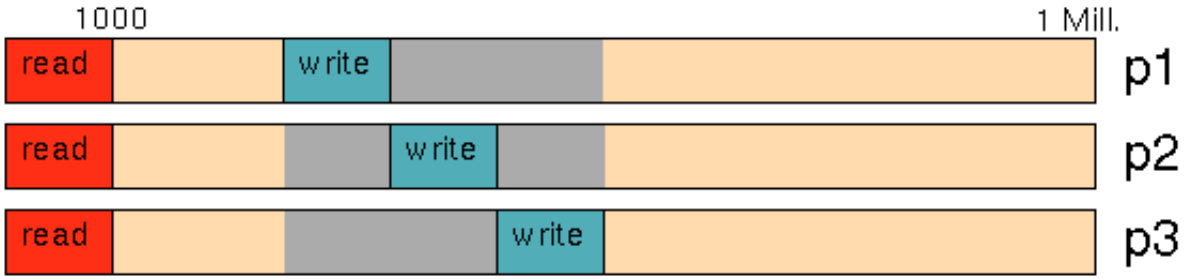


SONiC Communication Structure



- Write-invalidate and write-update protocols supported
- Programmer deals with replicated C++ data structures (objects)
- „invisible“ consistency management

Memory Representation of Replicated Data



- Example: Processes write disjunctive portions of an array
- Multicomputer (Sequent Symmetry):
  - Hardware defines layout of a data structure
  - Exclusive write accesses to memory pages
- Shared Objects:
  - Programmer (Algorithm-Designer) defines layout of data structures
  - Data are represented as replicated Sub-Arrays t, Read-replication
  - Partially allocated structures
  - Simultaneous write-accesses to disjunctive sub-arrays are possible (!)

Observations

- Software-DSM systems are easy to use (sequential programming model)
- Well suited for coarse-grained control parallel programming
- Variety of weakly consistent memory management protocols;
- many experimental systems:
  - Munin, TreadMarks (Rice Univ.) (release consistency),
  - MIDWAY (CMU) (entry consistency),
  - PANDA (U.Kaiserslautern) (page differentiation, migration),
  - Linda (Yale) (Tuple Space)
- No single standard system
- Reliability? – predictable system behavior?

Motivation for research on middleware-based systems