

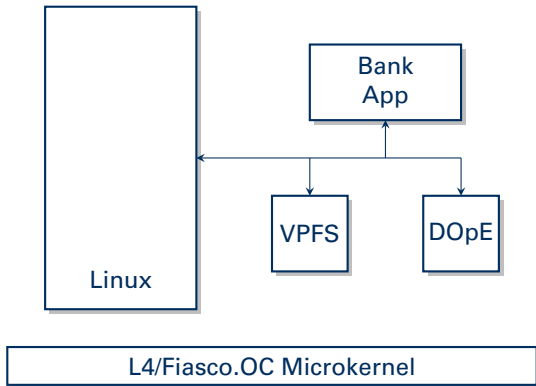
# WHO WATCHES THE WATCHMEN?

## Protecting Operating System Reliability Mechanisms

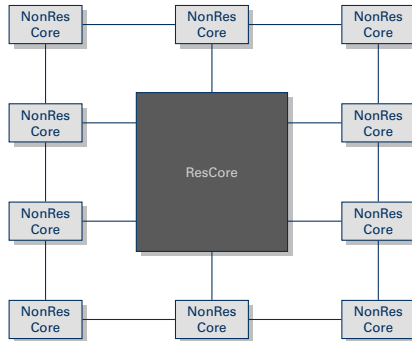
Björn Döbel, Hermann Härtig

Hollywood, 10/07/2012

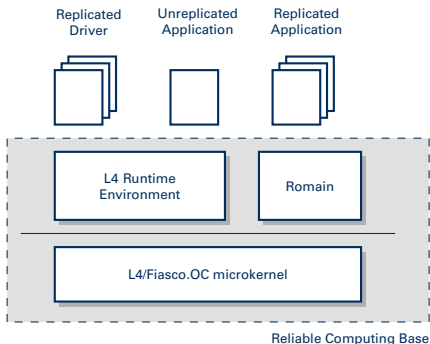
## Splitting Systems



## Assumption: Res & NonRes Cores

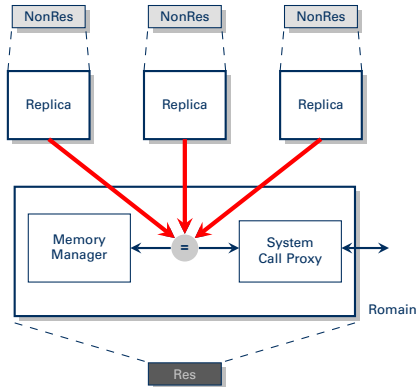


# Transparent Replication as OS Service



[DHE12] B. Döbel, H. Härtig, M. Engel:  
"Operating System Support for Redundant Multithreading", EMSOFT 2012

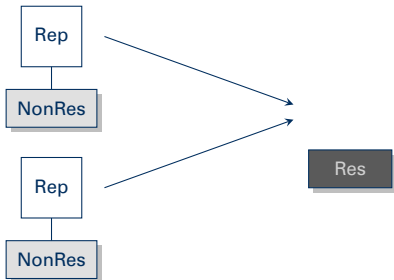
## Romain: Structure



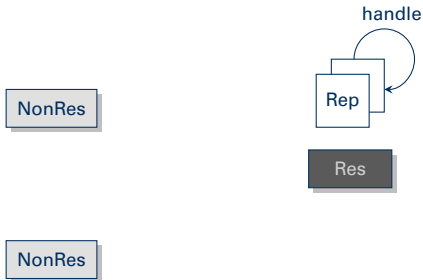
# Three Alternatives for Signalling

1. Thread Migration
2. Synchronous notifications
3. Shared-memory polling

## Alternative #1: Thread Migration

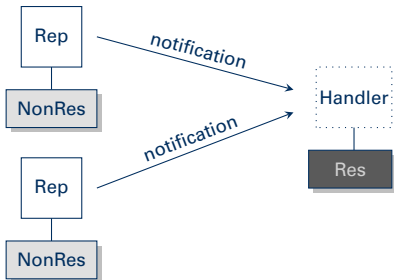


## Alternative #1: Thread Migration

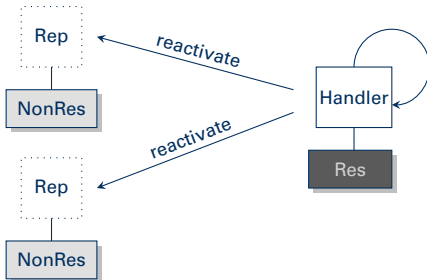




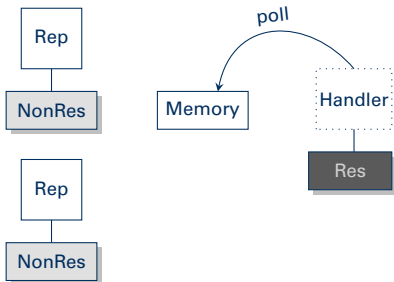
## Alternative #2: Notifications



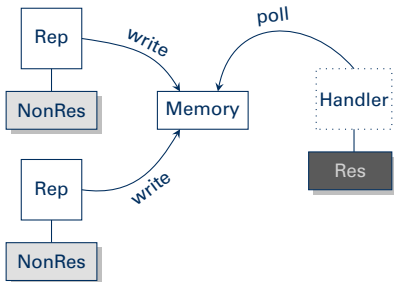
## Alternative #2: Notifications



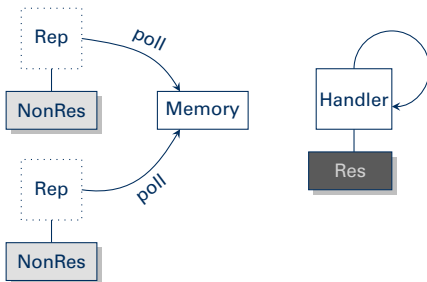
## Alternative #3: Shared-Memory Polling



## Alternative #3: Shared-Memory Polling



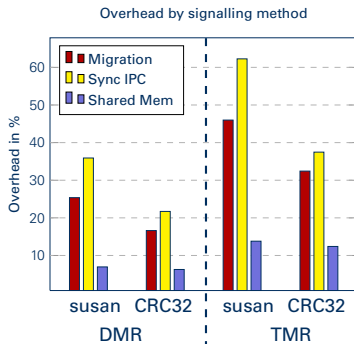
## Alternative #3: Shared-Memory Polling



## Evaluation

- MiBench, single-threaded
  - susan: image filter
  - CRC32: checksumming a file
- Benchmarks with highest overhead in [DHE12]
- Test machine:
  - 12 Intel Core i7 CPUs @ 2.6 GHz
  - Replicas pinned to dedicated physical cores
  - Hyperthreading off
- Double (DMR) and triple (TMR) modular redundancy

# Overhead to Unreplicated Execution



## Transparent Replication as OS Service

- This paper:
  - Protection of RCB components
  - Efficient signalling
- [DHE12]:
  - Application replication
  - Transmission errors
- To be done:
  - Multithreading (determinism)
  - Device drivers, I/O
  - Scalability Analysis



## Key Points

- Reliable Computing Base
- Assumption: Hardware with varying resilience levels
- Replication as OS Service
- Efficient signalling between Res and NonResCores
- Hardware wishlist:
  - Memory isolation between NonResCores
  - Fast inter-core notifications (e.g., Intel SCC)