

Transactional Processes in OSIRIS

Can Türker, Christoph Schuler, Klaus Haller

ETH Zurich
Institute of Information Systems
Database Research Group

tuerker@inf.ethz.ch

Global Grid Forum "Grid Transactions BoF", 11.03.2004

Exploiting Different Computing Paradigms

- **Database Technology**
 - ACID Transactions
 - Query Optimization
 - Indexing
- **Peer-to-Peer Computing**
 - Direct communication
 - Scalability
 - Large scale communities
- **Process Management**
 - Programming in the Large
 - Visual Programming
 - Reuse of existing services
 - Composite Services
- **GRID Computing**
 - Resource Management
 - On demand computing
 - Self adaptation
 - Load Balancing

Exploiting Different Computing Paradigms

Transactional guarantees for
composite processes

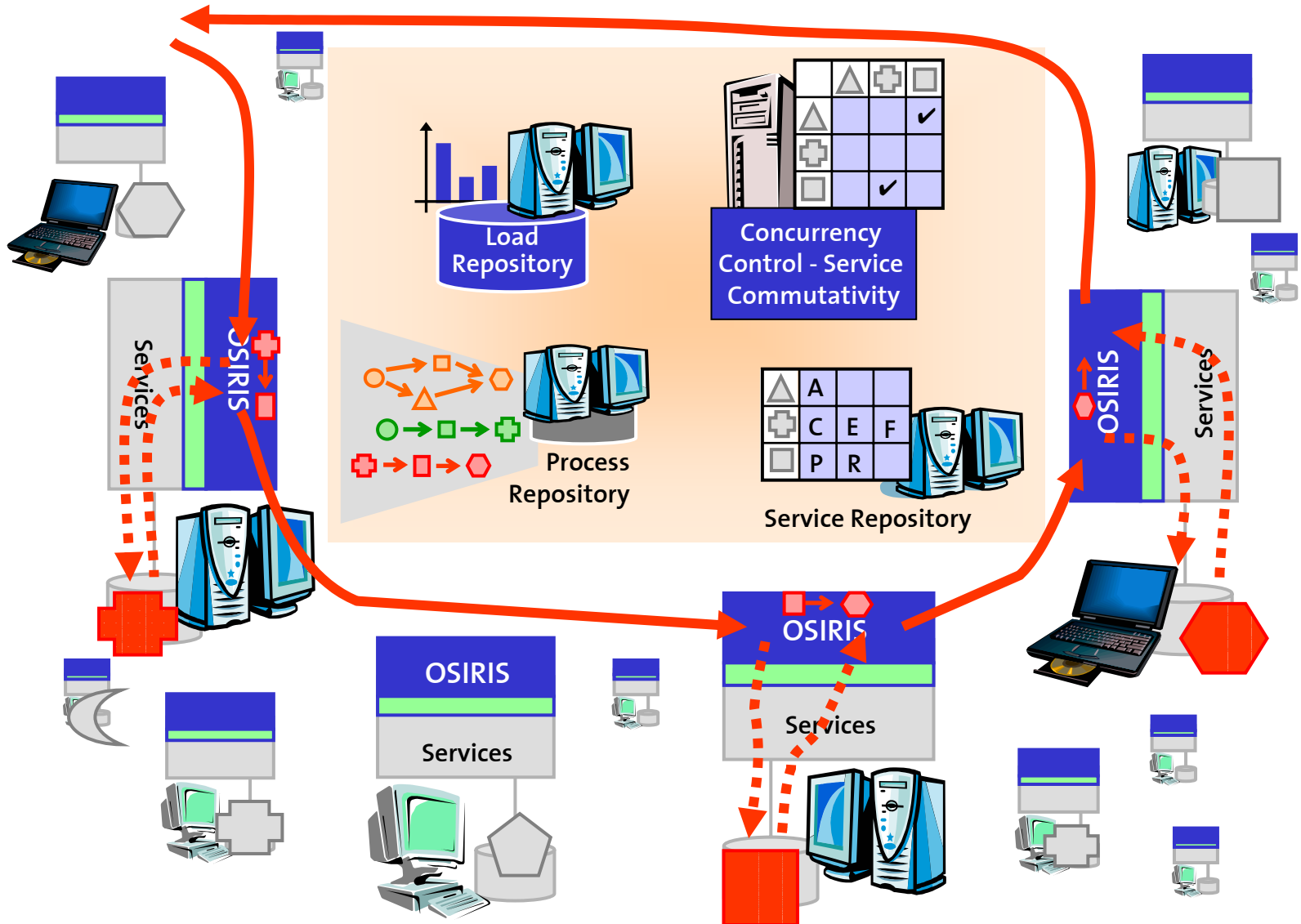
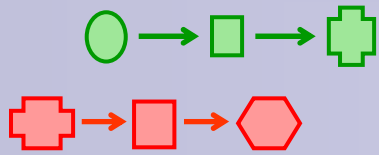
Peer-to-peer execution of
processes

Open Service
Infrastructure for Reliable and
Process Support

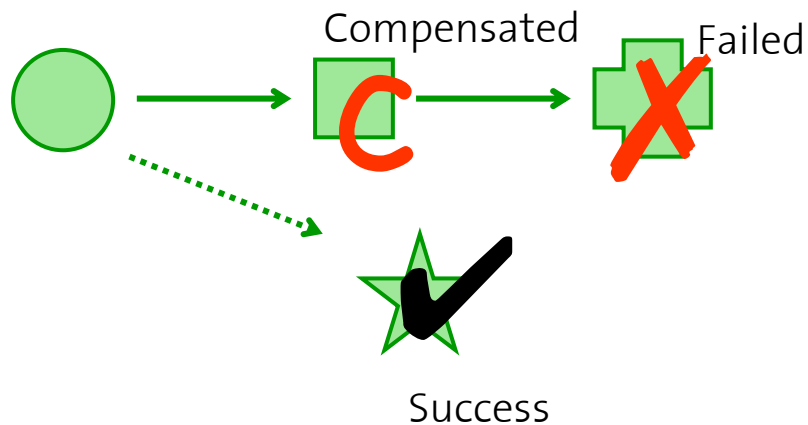
Processes to combine services

Dynamic routing of services

OSIRIS Architecture

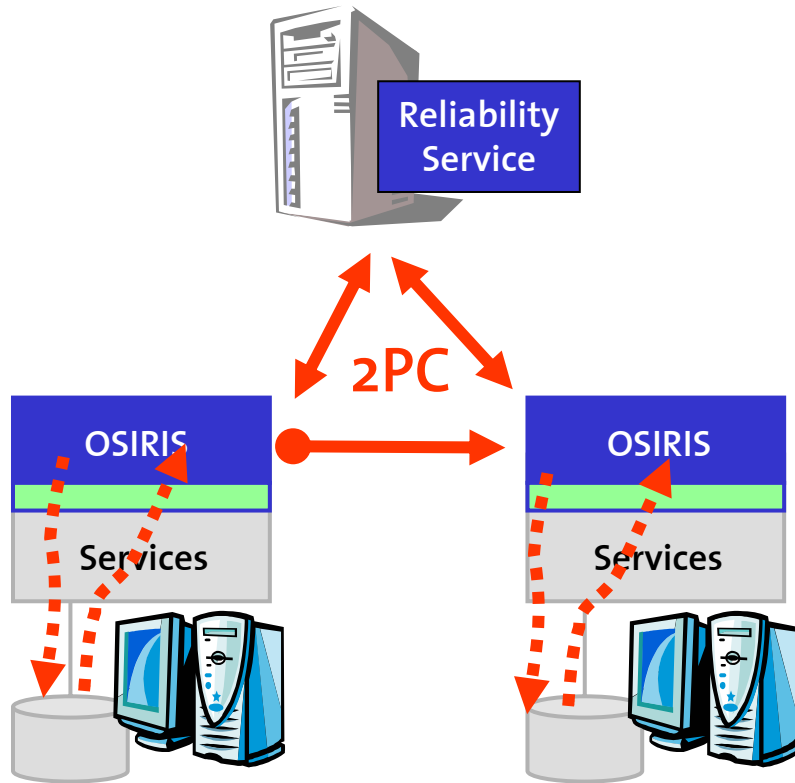


Providing “Database like” Guarantees



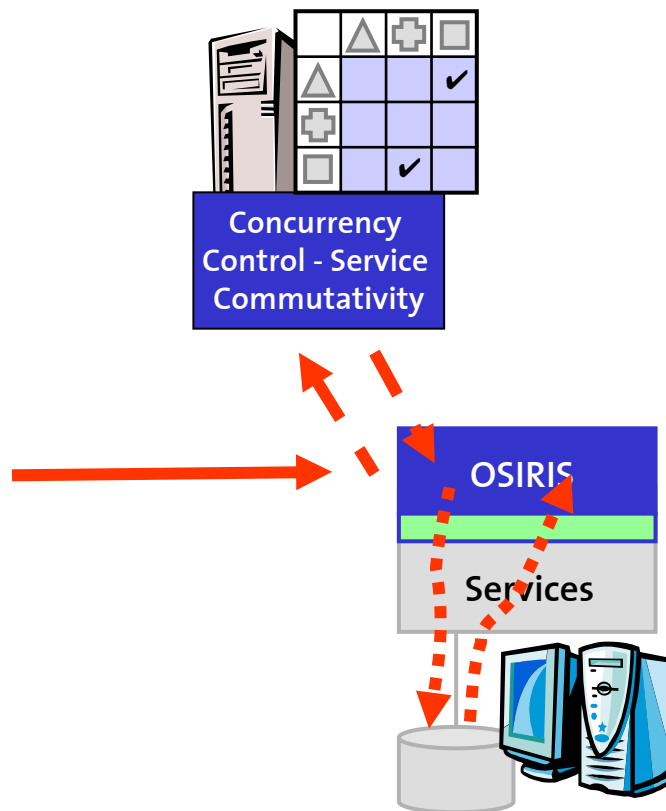
- **Transactional Processes**
 - **Guaranteed Termination**
 - **Compensation and alternative execution paths**
- **Persistent process routing**
 - 2PC Protocol from Peer to Peer
 - Reliability Service
- **Concurrency Control on Service level**
 - Intercepting Service Call
 - Global Concurrency Control Service

Providing “Database like” Guarantees



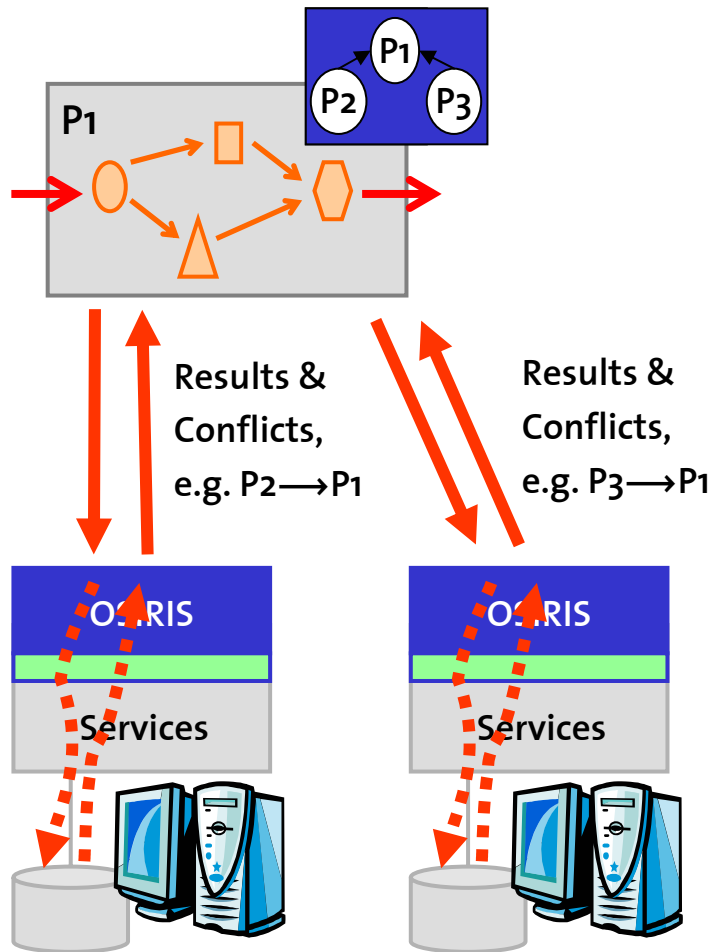
- Transactional Processes
 - Guaranteed Termination
 - Compensation and alternative execution paths
- Persistent process routing
 - 2PC Protocol from Peer to Peer
 - Reliability Service
- Concurrency Control on Service level
 - Intercepting Service Call
 - Global Concurrency Control Service

Providing “Database like” Guarantees



- Transactional Processes
 - Guaranteed Termination
 - Compensation and alternative execution paths
- Persistent process routing
 - 2PC Protocol from Peer to Peer
 - Reliability Service
- **Concurrency Control on Service level**
 - Intercepting Service Call
 - Global Concurrency Control Service

Completely Distributed Concurrency Control



- No global concurrency control service!
- OSIRIS layer extended
 - service commutativity matrix
 - local serialization graph
- Transactional Ad-Hoc Processes
 - local serialization graph
 - P_1 may commit if there is no active process P_2 such that $P_2 \rightarrow P_1$
- Global deadlocks guessed by timeouts

- OSIRIS has a high potential to **scale** w.r.t. increasing **number of service providers** as well as to increasing **number of processes**
- Grid transactions
 - composite processes (higher order transactions)
 - semantic concurrency control (service commutativity)
 - guaranteed termination (service compensation and retrievable alternatives)
 - scaleable, peer-to-peer execution (replication of metadata)
 - dynamic routing / late binding of services