# Automatic Service Relocation Tool for Real Application Cluster in Oracle Database 11*g*

Mihai – Ovidiu Tirsa, Mariusz Piorkowski
mtirsa@cern.ch, mpirkow@cern.ch

Database Services, CERN IT Department

Student Session, CERN Summer Students Program,
August 17th, 2011

# Outline

- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

# Outline

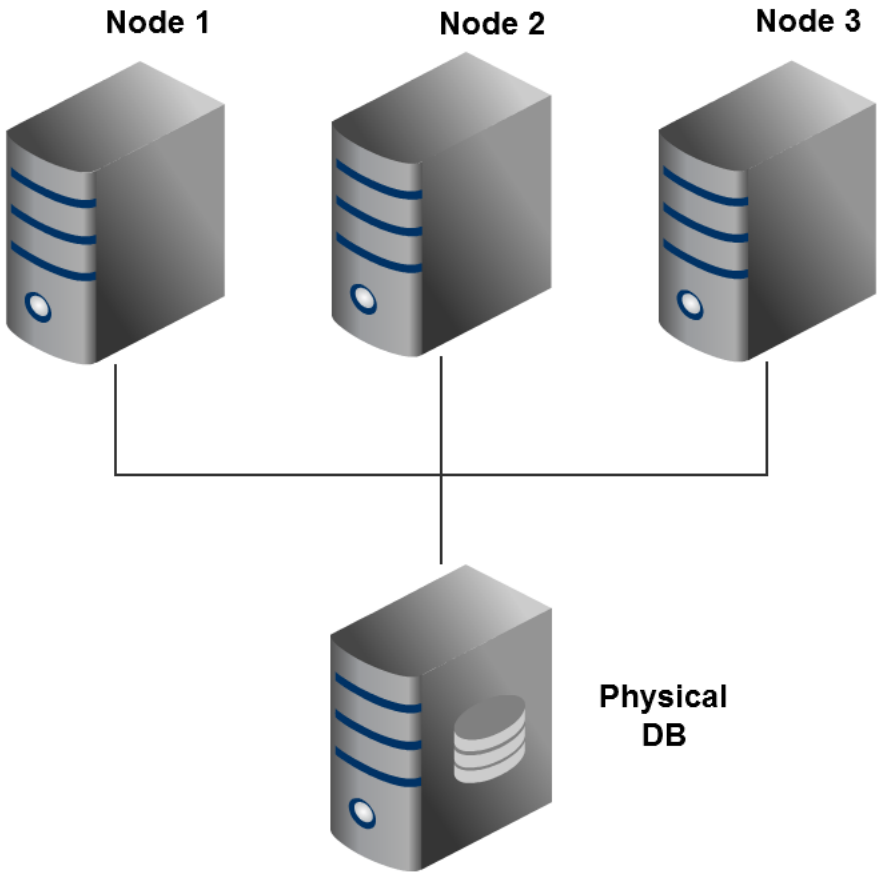
- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

- Real Application Cluster (RAC)
  - feature that allows multiple concurrent instance to share a single physical database
- Service
  - groups of application or a subset of a large application with common attributes, priorities, etc.
- Research Approach
  - before instance restart/update, the services are relocated to other available instances and moved back afterwards
  - some services running on the targeted instance are stopped and then restarted
  - use Server Control (***srvctl***) for service management

# Outline

- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

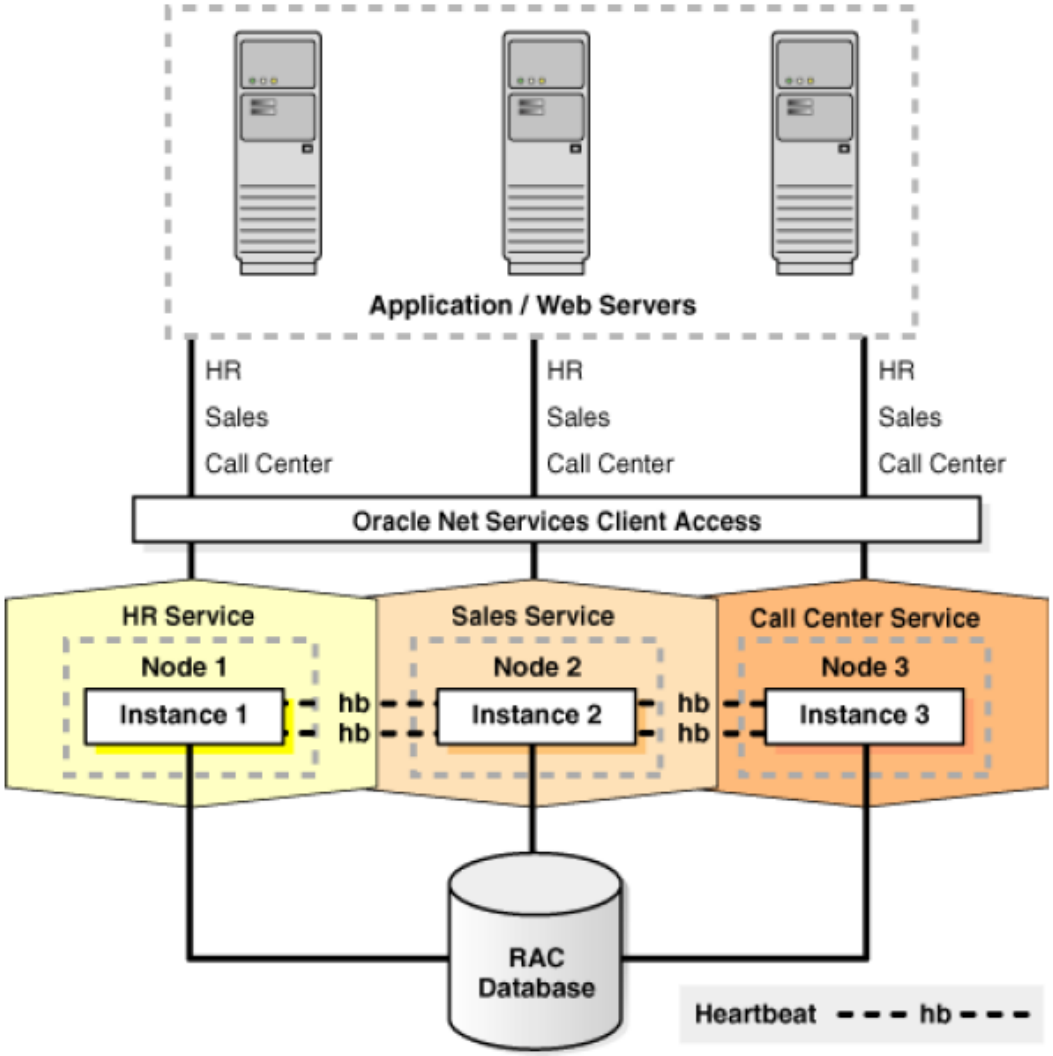
- Oracle RAC




THREE NODE RAC STRUCTURE

- Oracle RAC
  - High Availability
  - Flexible Scalability
  - Automatic Workload Management
  - Oracle Clusterware

## Services


Application / Web Servers
HR
Sales
Call Center
HR
Sales
Call Center
HR
Sales
Call Center
Oracle Net Services Client Access
HR Service
Node 1
Instance 1
Sales Service
Node 2
Instance 2
Call Center Service
Node 3
Instance 3
hb
hb
hb
hb
RAC Database
Heartbeat - - - hb - - -

## *Srvctl*

- Utility used to administer Oracle RAC
- Structure: ***srvctl command object options***
- Command: ***start***, ***stop***, ***add***, ***config***, ***status***, ***enable***…
- Object: ***service***, ***instance***, ***database….***
- Options: ***-s <service_name>, -d <database_name>***…
- Example:
  - ***srvctl status service -d MY_DB -s MY_SRV***

# Outline

- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

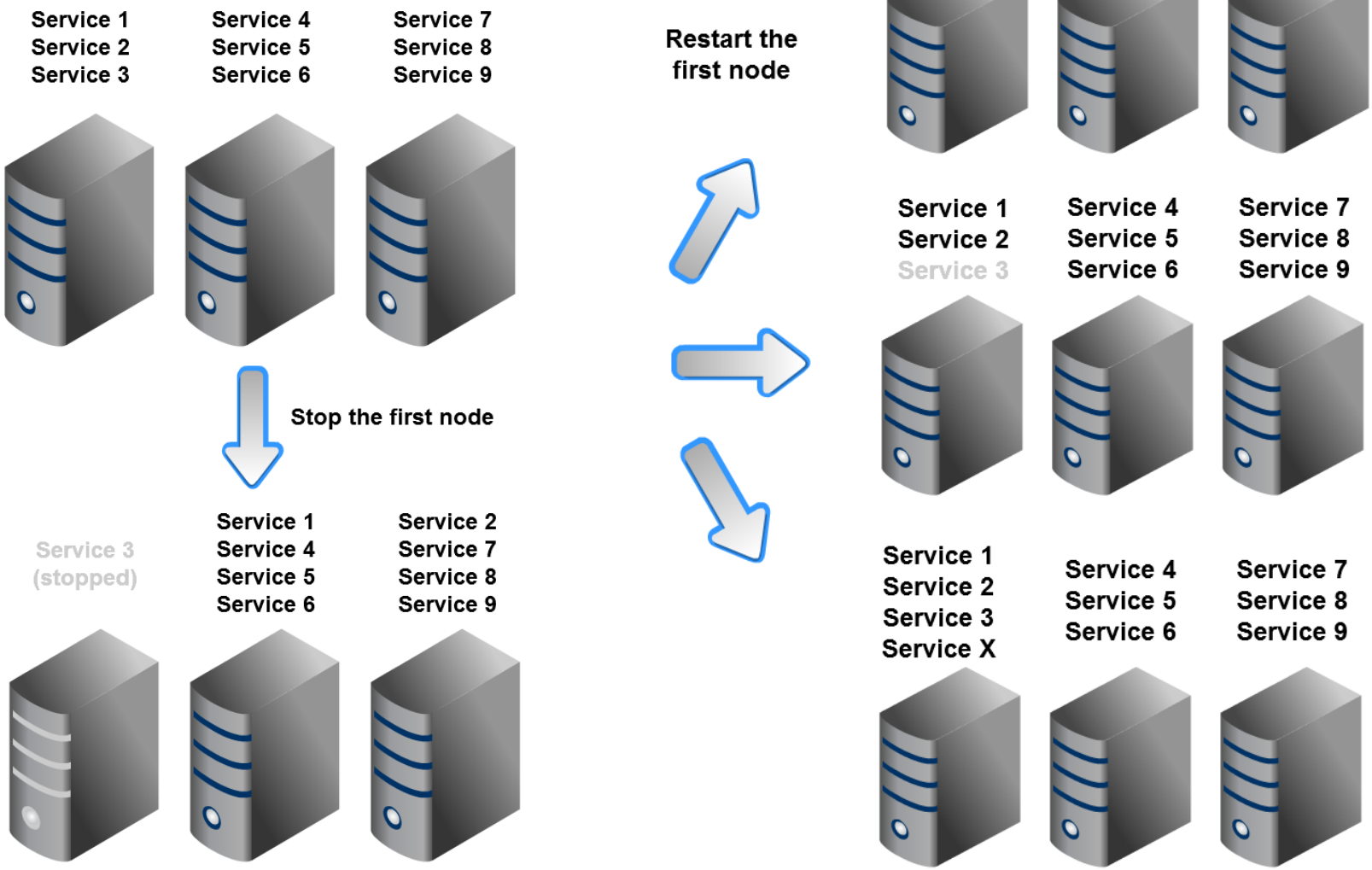
- Restore the exact RAC image after an instance is restarted or updated
- Offer a scalable solutions
- Use PERL as the programming language

# Outline

- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

- Fact
  - Oracle RAC provides automatic service checkpointing, service relocation, load balancing in order to achieve the first two points from slide 7
- Problem – unexpected behavior
  - After preferred instance comes back online, service will not move back from available instance
  - Enabled services restart by default, even if stopped before
  - ***srvctl*** can't start an disabled service, even if it ran before
- Muscle Solution
  - Manually move the services back and forth

## Unexpected behavior



- Brain Solution
  - Write a script that does it for you ☺
    - Get the current snapshot using ***srvctl*** and save it in a file
    - Disable all the processes that are not running
    - Disable all the processes that will be stopped
    - Stop those services
    - Relocate the other services for the other nodes, following a policy
    - Stop the instance
    - …
    - Restart the instance
    - Move services back, according to the initial snapshot
    - Enable the disabled processes
    - Restart the stopped services

- Improvements
  - On 1-node RAC, we can skip the relocation part
  - On 2-node RAC, we can relocate the services only on the other node
    - no policy following algorithm needed
  - Efficiency
    - Log every change of the first snapshot in a change file
    - Log every undo of every change in an undo file
    - After instance restart, just run the commands from the undo file
    - No need to save the first snapshot
      - May still be saved so it can be compared to the result

# Outline

- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

- Conclusions
  - Brain Solution works better than Muscle Solution
  - The tool can be easily integrated with other similar tools
  - The solution scales

- Further Work
  - Add the third instance
  - Stress tests => results
  - Add CPU usage aware load balance policy
  - Allow dynamic relocation policy definition

# Outline

- Introduction
- Technologies
- Contribution
- Implementation
- Conclusions
- Questions

# Questions