

How To Make Databases on SUSE® Linux Enterprise Server Highly Available

Mike Friesenegger
SUSE Sales Engineer
mikef@suse.com



Agenda

- Clarify the term "Availability"
- What is High Availability
- Minimize a Database Startup Time after a Failure
- Making an IBM DB2 Database Highly Available
- Making an Oracle Database Highly Available
- Demo

Questions to Ask About the Importance of a Database

- How much uptime does the business expect?
 - 24 x 7 x 365
 - Occasional downtime acceptable
 - Not important
- How quickly does the database need to recover from a failure?
 - Recovery time is unacceptable
 - Some recovery time is acceptable
 - Not important

Determine the Level of “Availability”

- Maximum¹ or Continuous² Availability
 - 24 x 7 x 365
 - Recovery time is unacceptable
- High Availability
 - Occasional downtime acceptable
 - Some recovery time is acceptable
- No Availability
 - Not important

¹ “High Availability Overview”,
http://www.oracle.com/pls/db112/portal.portal_db?selected=14&frame=

² “High Availability and Disaster Recovery Options for DB2 for Linux, UNIX, and Windows”,
<http://www.redbooks.ibm.com/abstracts/sg247363.html>

The Cost of “Availability”

- Maximum or Continuous Availability
 - Additional hardware/software costs
 - Most complex configuration
 - Largest amount of time to plan and implement
- High Availability
 - Additional hardware/software costs
 - Complex configuration
 - Time required to plan and implement
- No Availability
 - Normal cost of running a single instance database



The Products That Assist With “Availability”

- Maximum or Continuous Availability
 - Oracle Real Application Cluster (RAC)
 - IBM DB2 pureScale®
- High Availability
 - An OS based HA Clustering solution
 - IBM PowerHA for AIX
 - Microsoft Cluster Server for Windows
 - Veritas Cluster Server
 - SUSE Linux Enterprise High Availability Extension
 - Optional components
 - Database replication
 - Fast Failover capability



A Note About the “Availability” Products

- Maximum or Continuous Availability
 - Availability is only one aspect of these products
 - Flexibility to scale processing capacity is the biggest goal of these products
 - Automatically load balance application connections as a benefit of scaling
 - May address disaster recovery
- High Availability
 - Primary focus on increasing availability
 - May address disaster recovery

Clarify the Term “Availability”

- Depends on your business requirements
 - Maybe you do not need any availability
- If you do
 - Maximum or Continuous Availability provides the most options and features with the most complexity and cost
 - High Availability may be all you need!
- The remainder of this session will focus on providing High Availability
 - Focusing on Oracle and DB2 databases
 - Using the SUSE Linux Enterprise High Availability Extension

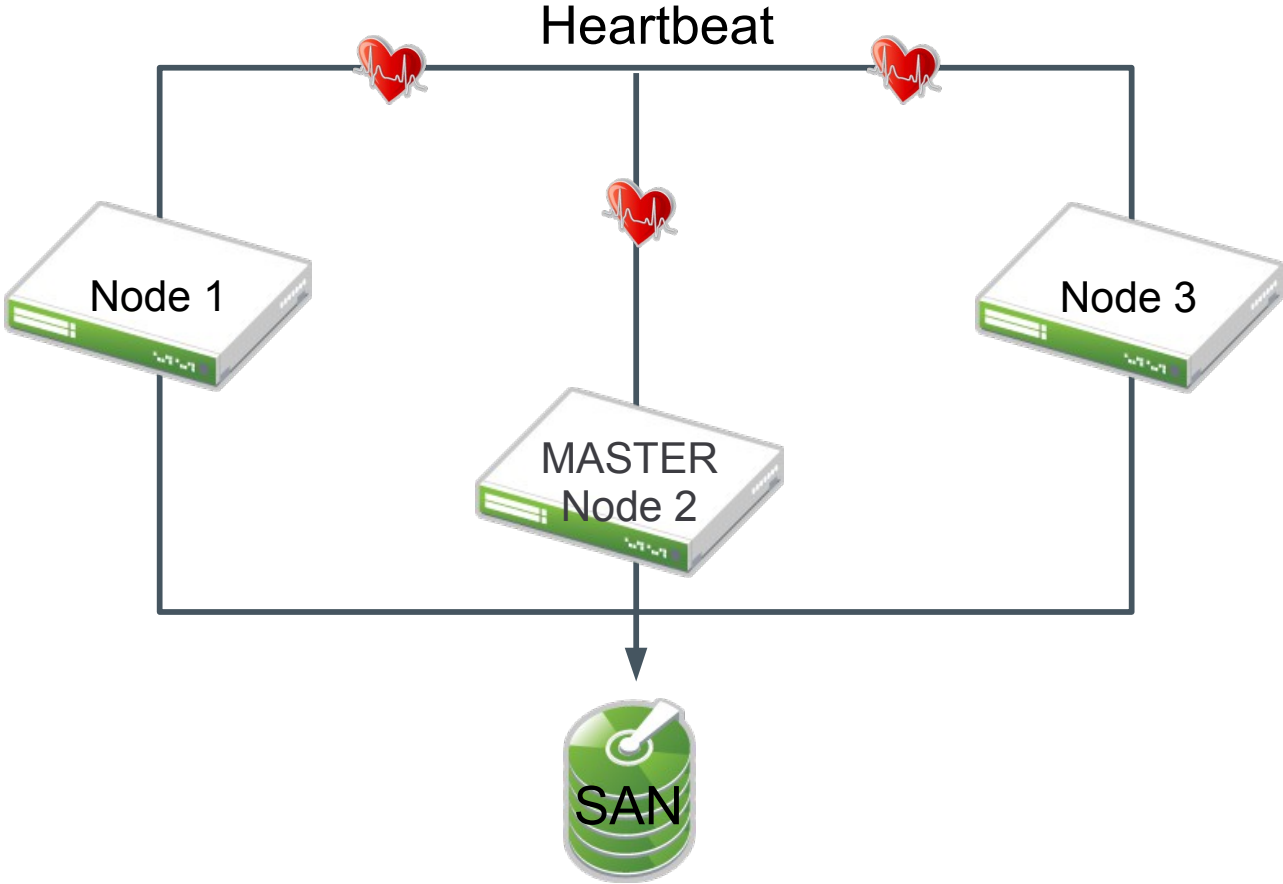
Features

SUSE® Linux Enterprise High Availability Extension

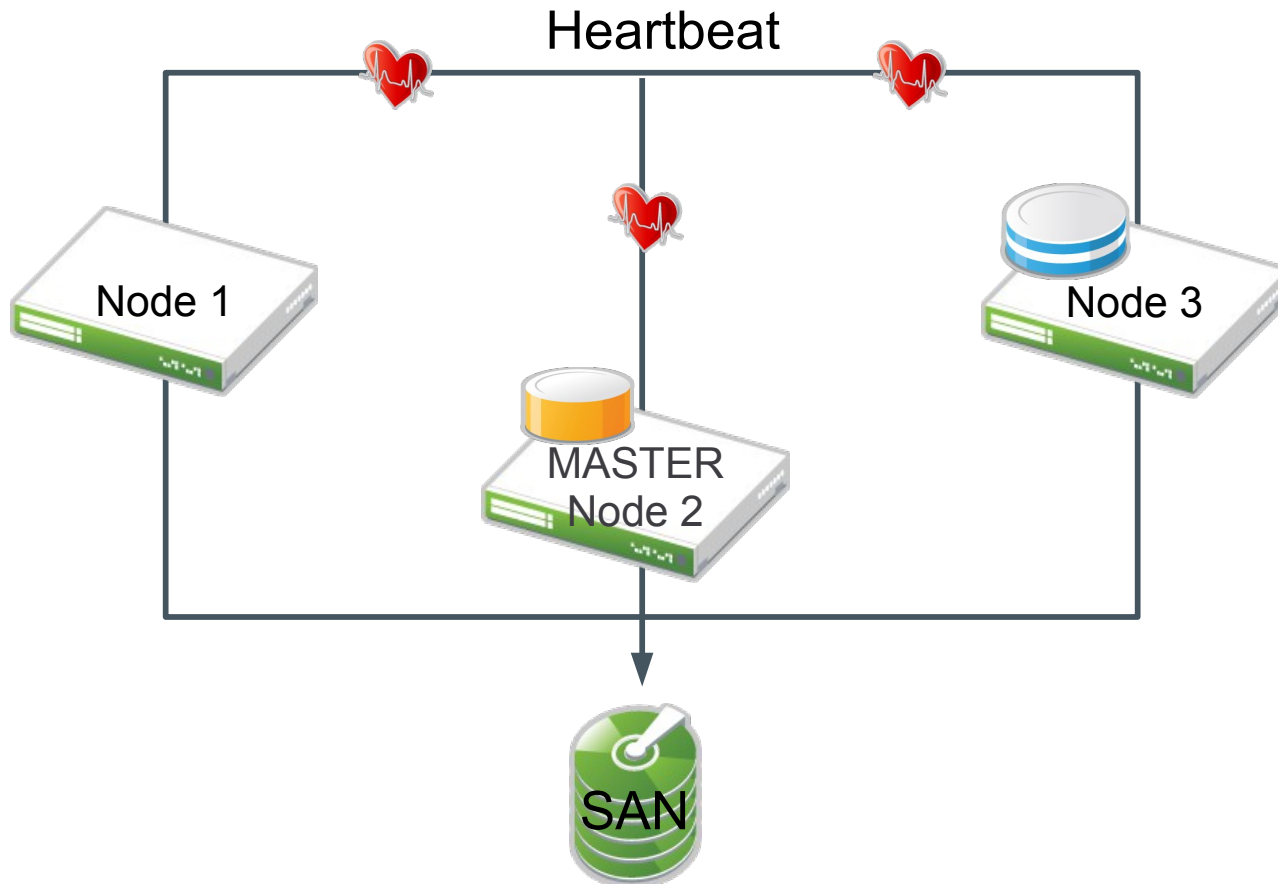
- **Service availability 24/7**
 - Policy driven clustering
- **Shared and Scaled data-access**
 - Cluster file system
 - Clustered Samba
- **Scale Network Services**
 - IP load-balancing
- **Virtualization Agnostic**
 - Platform independent setup
- **Disaster tolerance**
 - Data replication via IP
 - Node recovery
- **User friendly tools**
 - Graphical user interface
 - Unified command line
- **Free Resource Agents**
- **Geo Clustering**
 - Cluster across unlimited distance

What is High Availability?

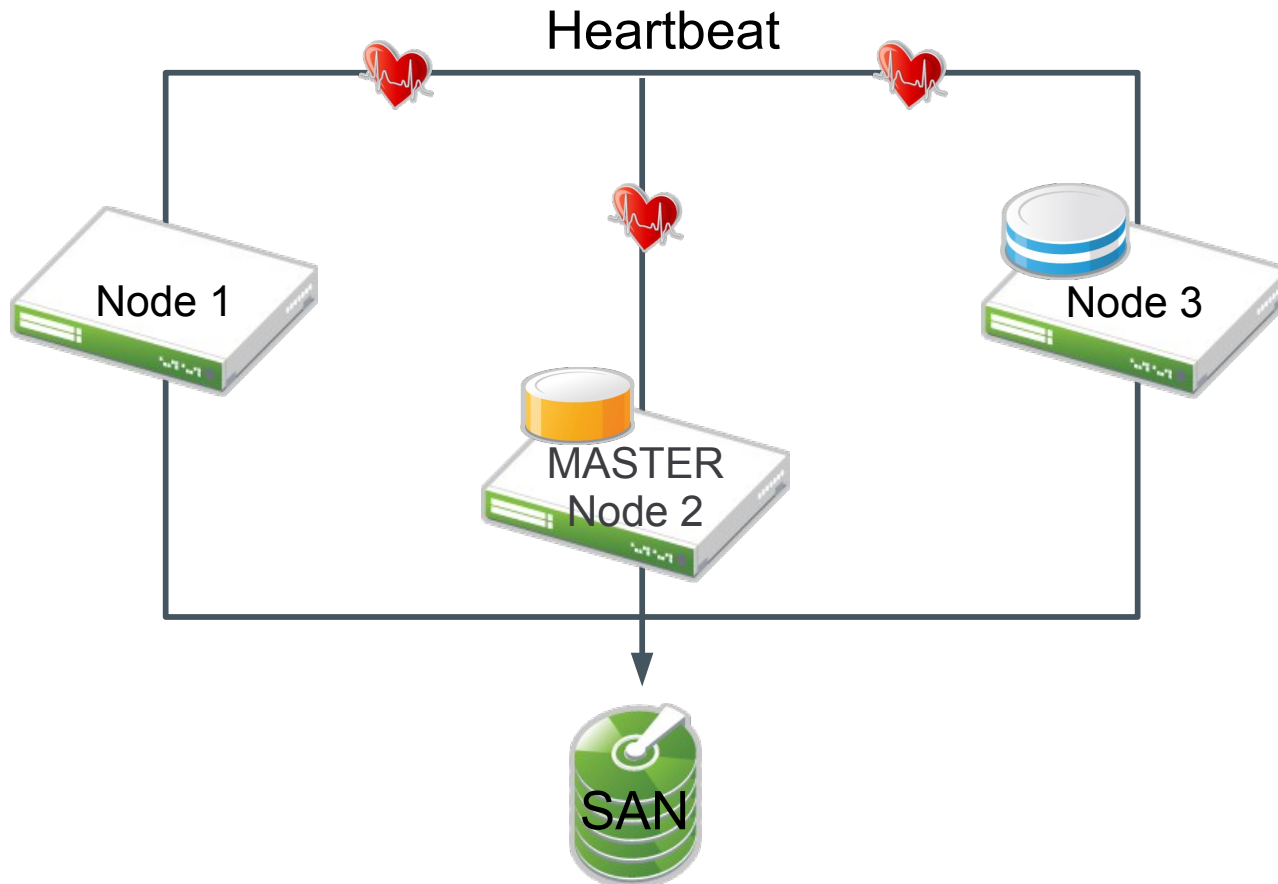
A Simple HA Cluster



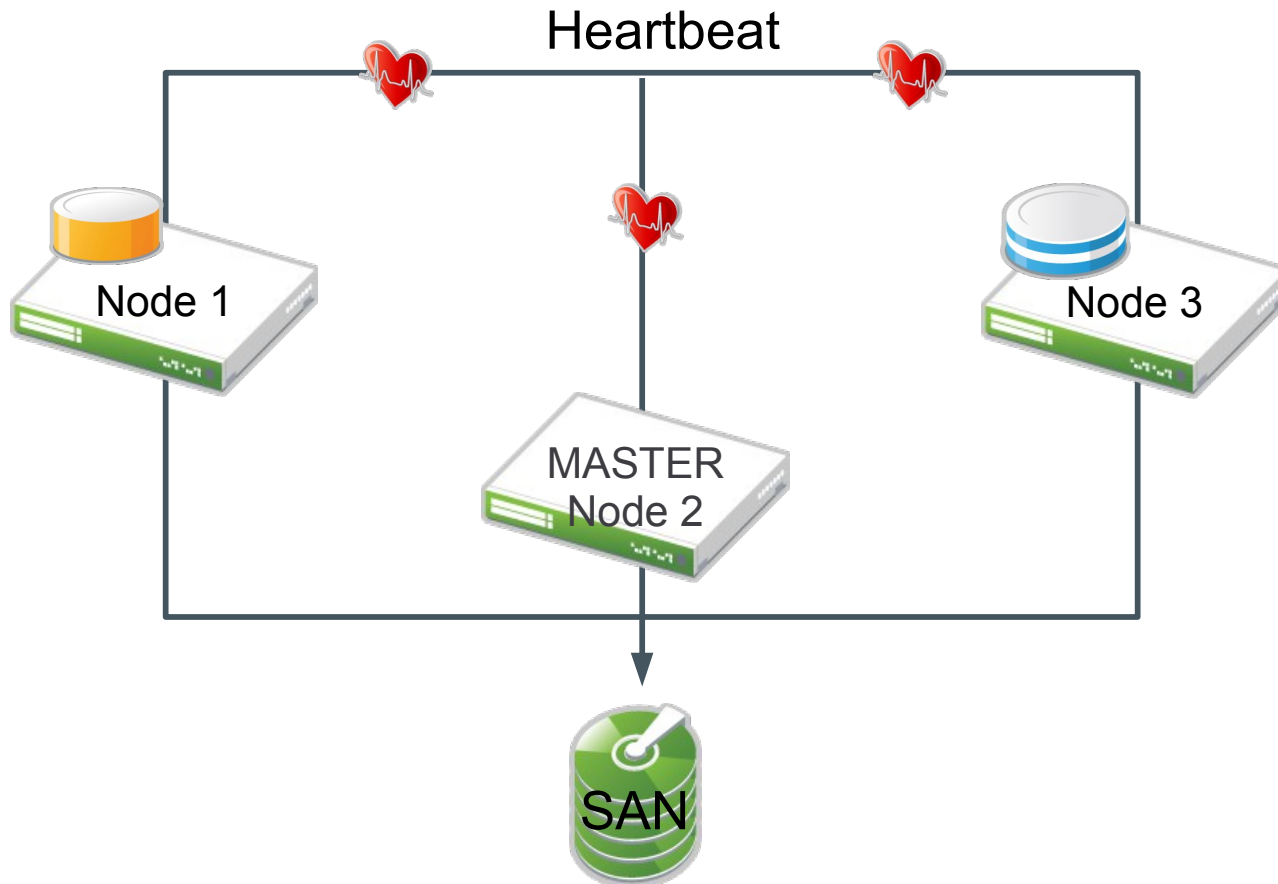
An Active/Passive Database Cluster



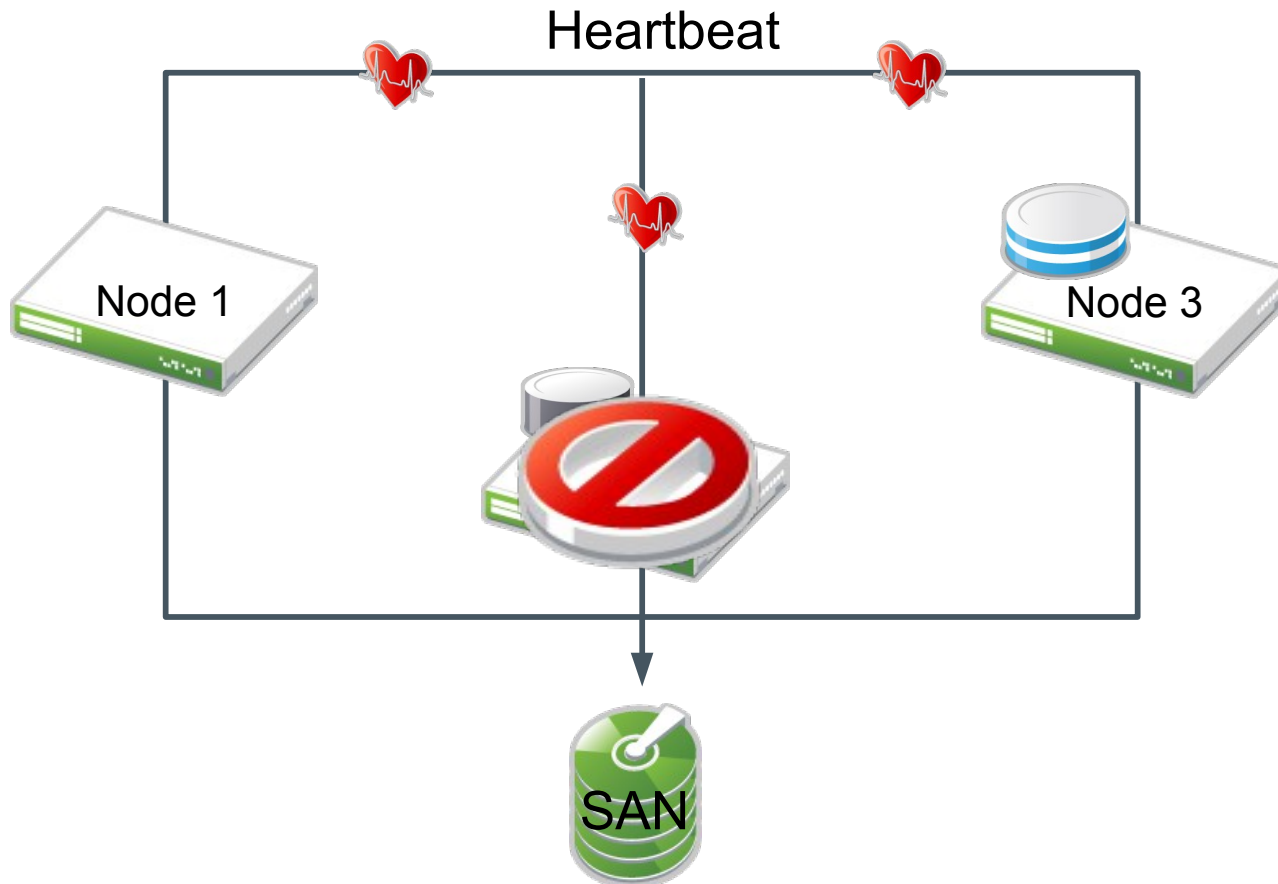
Migrating a Database to a New Node



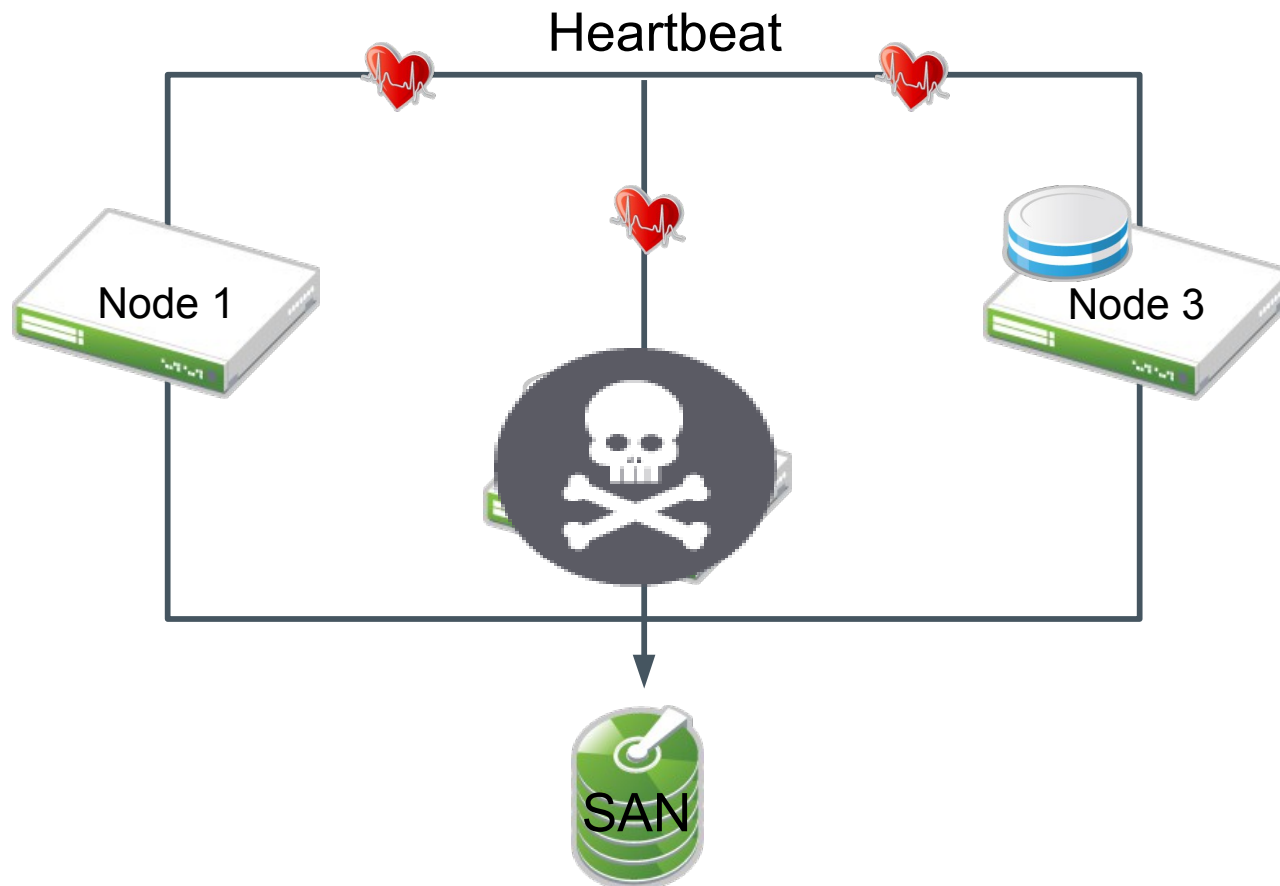
Migrated Database on a New Node



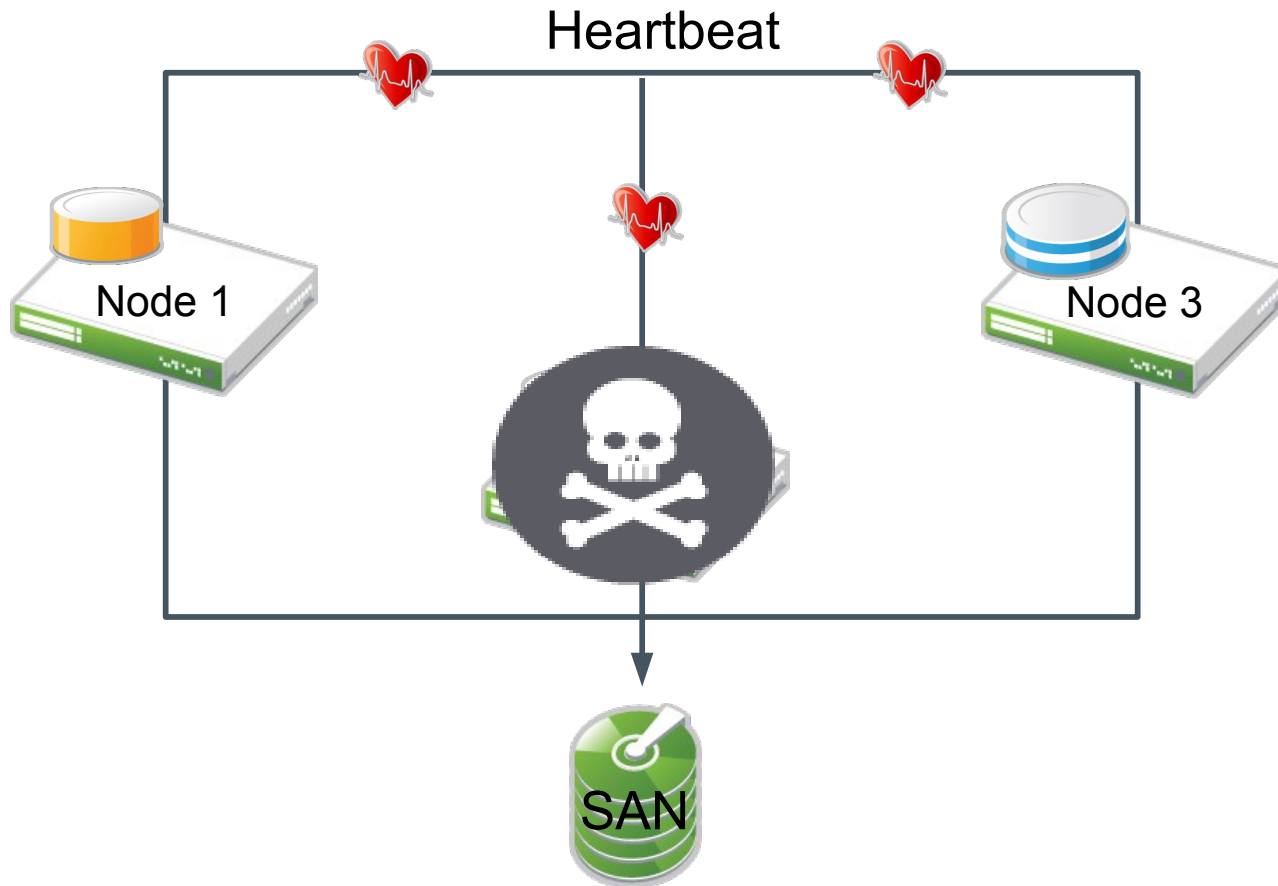
Node Failure in the Cluster



Kick the Failed Node Out of the Cluster



Database Automatically Restarted On Another Node



Minimize Startup Time of a Database

Migration and Failover Can Be Time Consuming

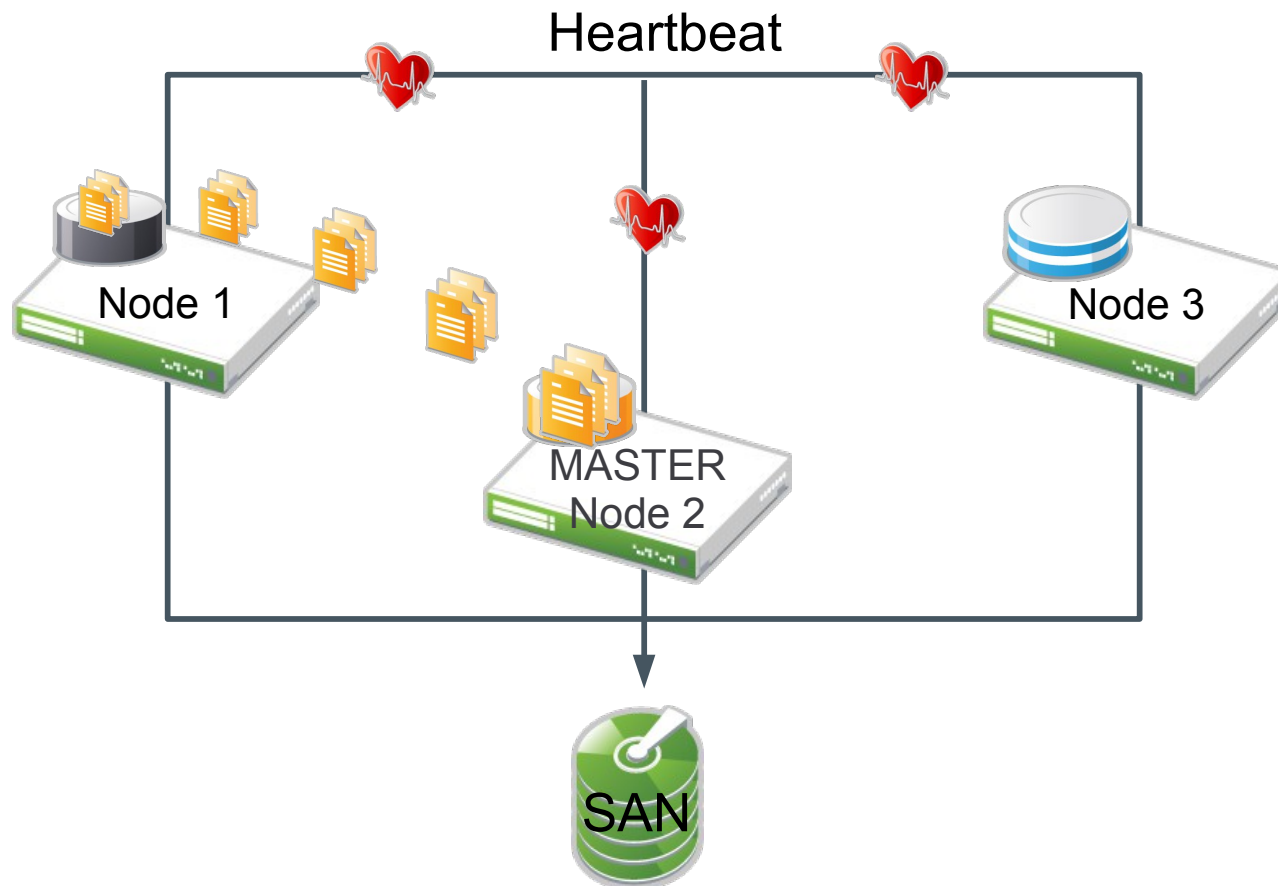
Steps in migration

- Clean database shutdown
- Release shared storage from current node
- Attach shared storage to new node
- Start database and listener

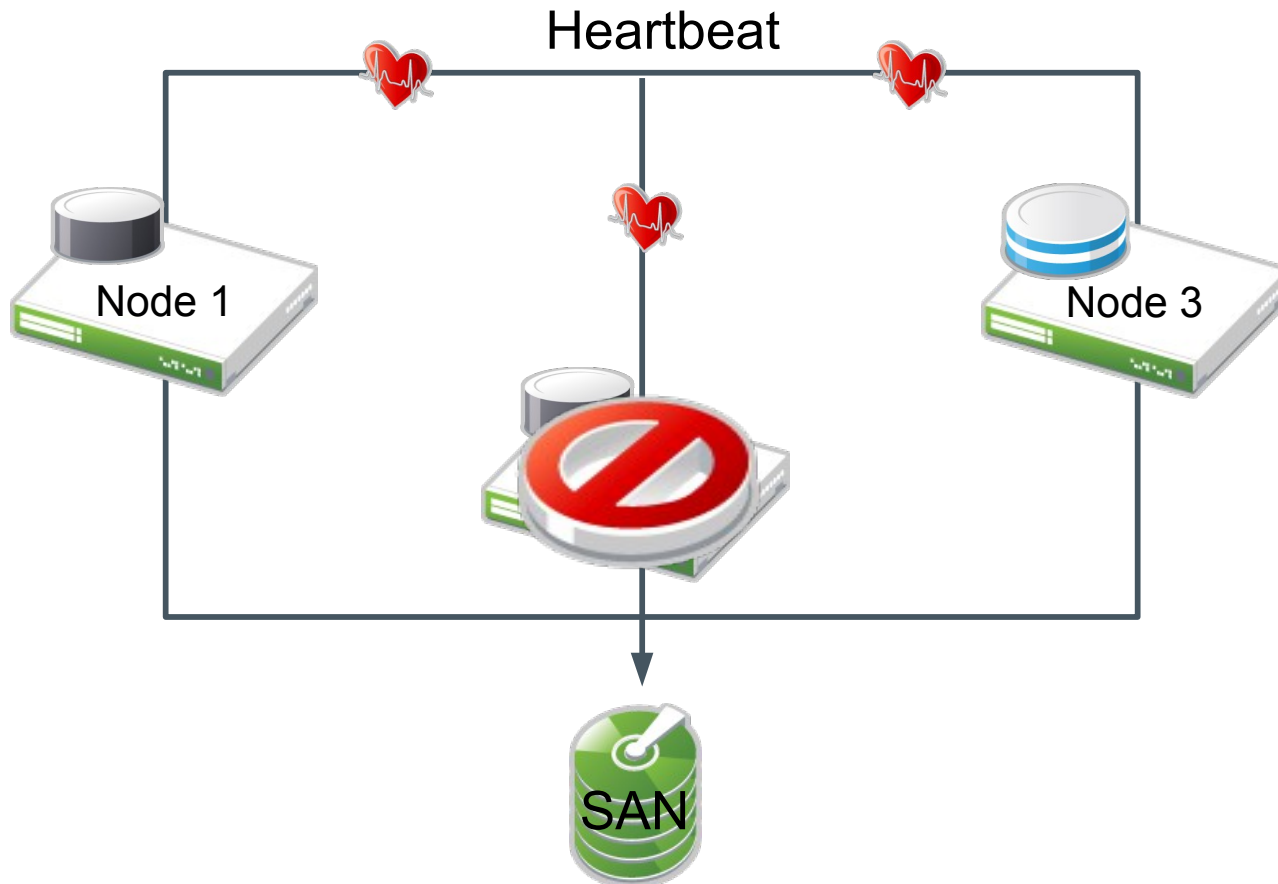
Steps for failover

- Detection of failed node
- Kick the failed node out of cluster
 - Issue node eviction
 - Wait for node eviction
 - Guarantee that node is kicked out
- Attach shared storage to new node
- Start database and listener
 - Verify no database corruption

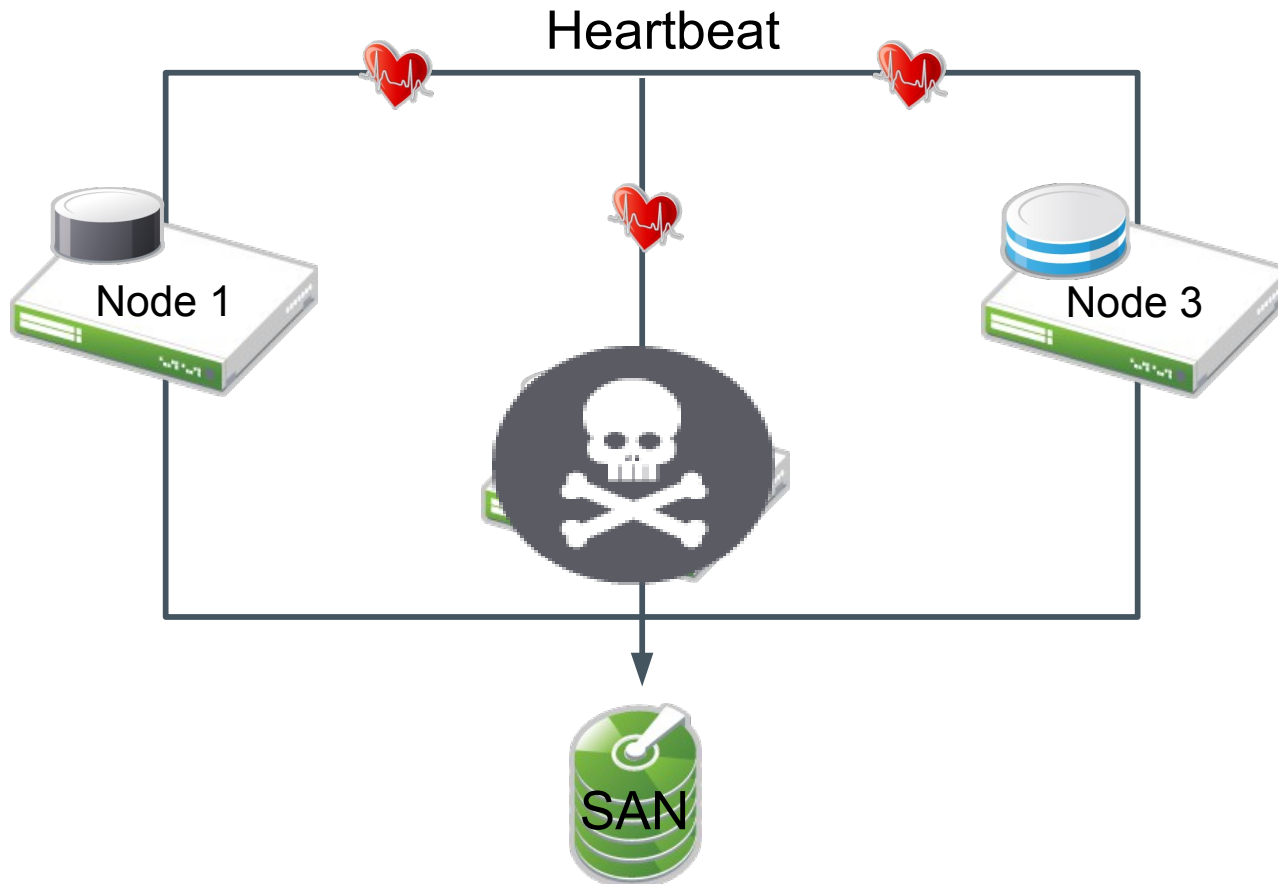
Replication to Standby Database Via Log Shipping



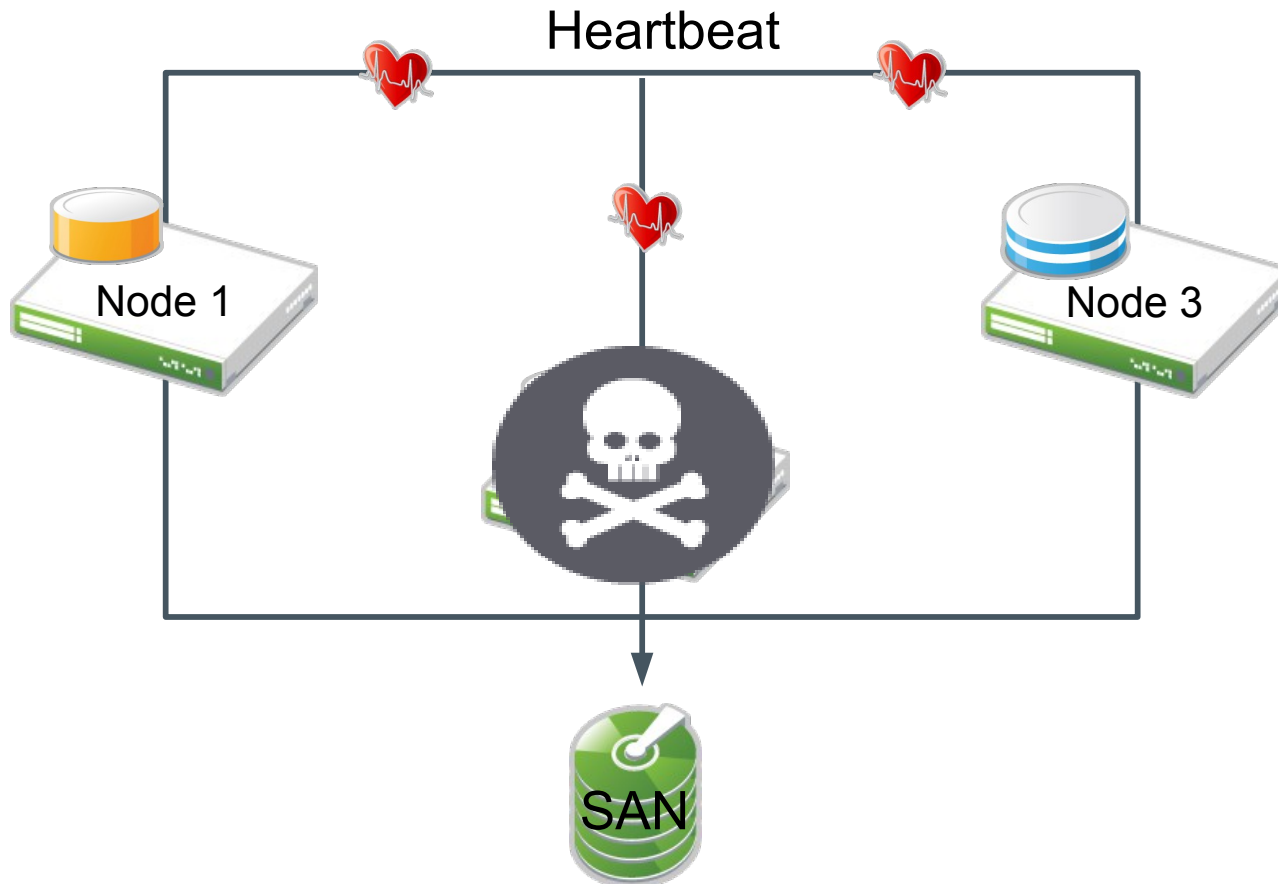
Node Failure in the Cluster



Kick the Failed Node Out of the Cluster



Standby Database Automatically Started on Another Node



Making an IBM Database Highly Available

Components in a Highly Available IBM DB2 Database

- IBM DB2 10.1 for Linux
- IBM DB2 High Availability and Disaster Recovery (HADR)
 - Provides log shipping feature
 - Recommended to use a third-party clustering solution
- SUSE Linux Enterprise High Availability Extension 11 SP3
 - Manages start/stop/failover of resources
 - Shared storage
 - Service IP Address
 - IBM DB2 Database and HADR
 - Provides node monitoring
 - IP address failover



Making an Oracle Database Highly Available

Components in a Highly Available Oracle Database

- Oracle Enterprise 11gR2 for Linux
- Oracle Data Guard
 - Provides log shipping feature
 - Oracle recommends using Oracle Clusterware
 - Other third-party HA clustering solutions will work
- SUSE Linux Enterprise High Availability Extension 11 SP3
 - Manages start/stop/failover of resources
 - Shared storage
 - Service IP Address
 - Oracle Database and Data Guard
 - Provides node monitoring
 - IP address failover



Demo

More About High Availability with SUSE Linux Enterprise

- CAS1417** A Xen cluster success story using the SLES HA Extension
- CAS1589** A carrier grade cloud phone system based on SUSE Linux Enterprise Server
- TT1395** How to Build an HA environment with Linux on IBM System z
- TT1422** Linux Clusters Made Easy with SLES HA Extension

Thank you.





Corporate Headquarters
Maxfeldstrasse 5
90409 Nuremberg
Germany

+49 911 740 53 0 (Worldwide)
www.suse.com

Join us on:
www.opensuse.org

Unpublished Work of SUSE. All Rights Reserved.

This work is an unpublished work and contains confidential, proprietary and trade secret information of SUSE. Access to this work is restricted to SUSE employees who have a need to know to perform tasks within the scope of their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated, abridged, condensed, expanded, collected, or adapted without the prior written consent of SUSE. Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

General Disclaimer

This document is not to be construed as a promise by any participating company to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. SUSE makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for SUSE products remains at the sole discretion of SUSE. Further, SUSE reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All SUSE marks referenced in this presentation are trademarks or registered trademarks of Novell, Inc. in the United States and other countries. All third-party trademarks are the property of their respective owners.

