



Multi-Data Center MySQL with Continuent Tungsten

Peter Boros, Consultant, Percona
Robert Hodges, CEO, Continuent
Percona Technical Webinars
March 2, 2013

About This Presentation

- Why is multi-data center MySQL important?
- How does Continuent Tungsten fill the gaps?
- Demonstration
- Practical steps to multi-data center MySQL



Why is multi-data center MySQL important?

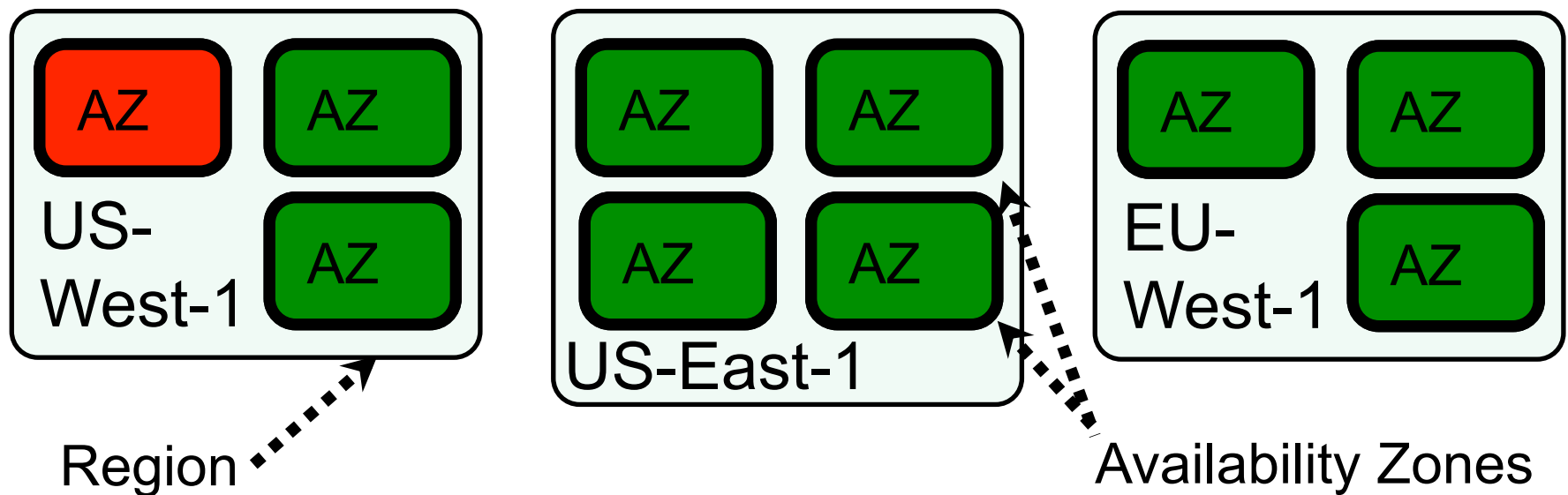
Robert Hodges

Key reasons for multi-data center MySQL

- High availability – Moving data away from problems
- Performance – Moving data closer to users
- Legal – Keeping data within some locations but not others

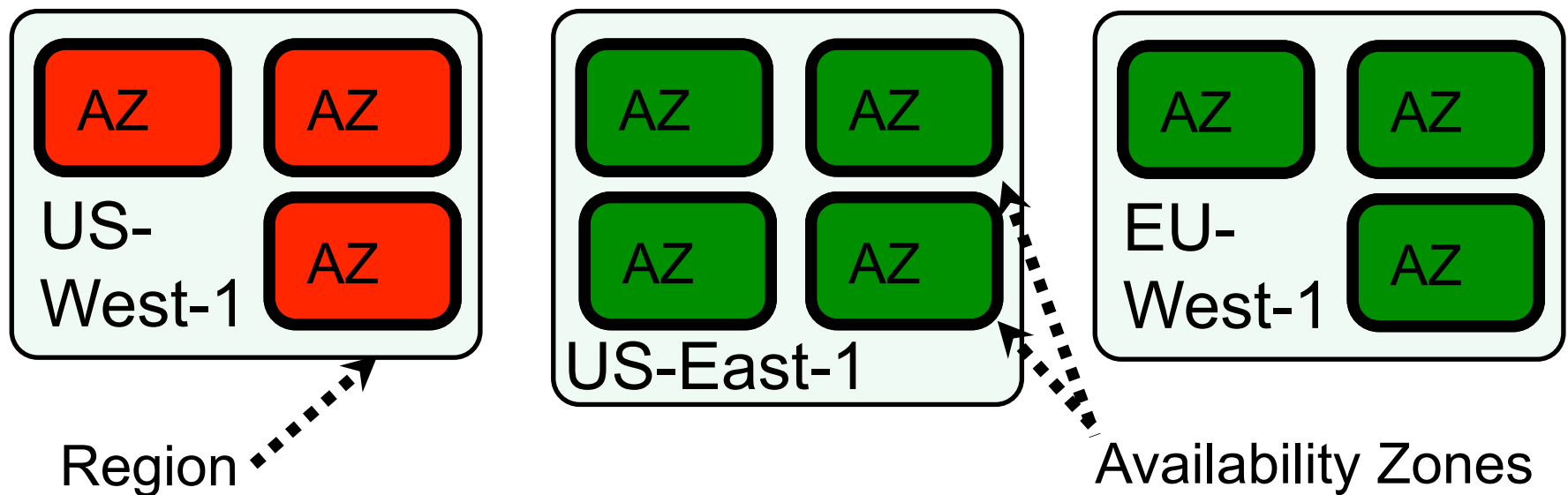
HA Example – Amazon Failure

EBS fails in a single availability zone



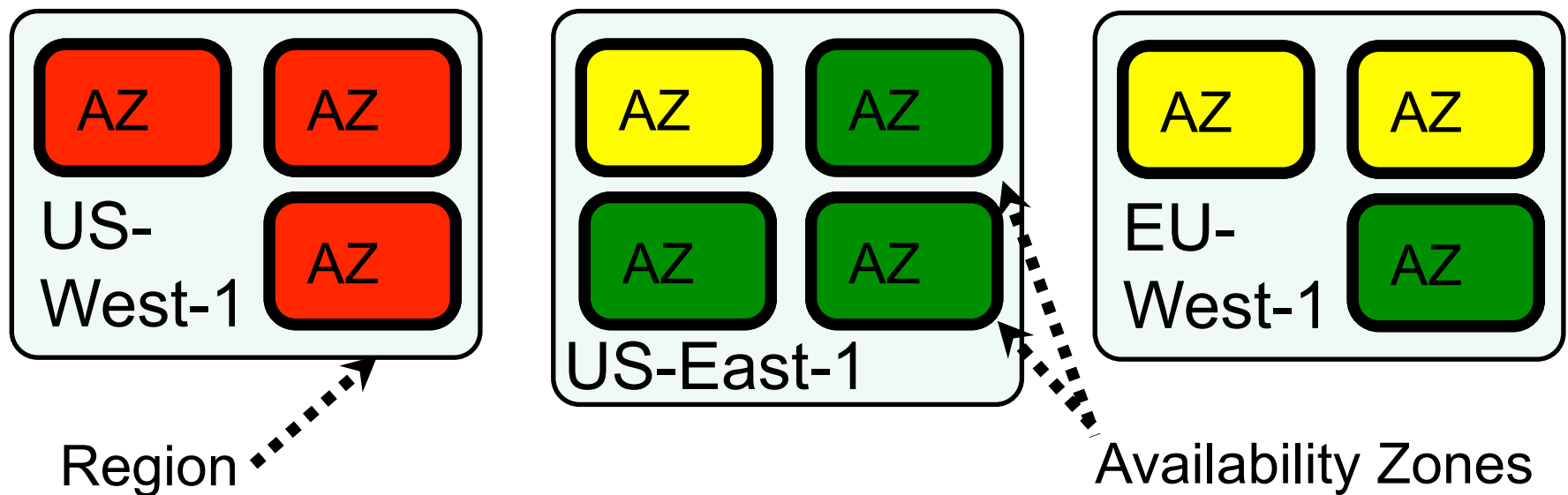
HA Example – Amazon Failure

Leading to failures within the region due to coupled Amazon services



HA Example – Amazon Failure

Leading to failure of tightly coupled applications in other regions



Requirements for successful multi-DC MySQL

- Local HA within geographic regions
 - Failover
 - Planned maintenance and upgrade
- Advanced multi-site replication
 - Flexible topologies
 - Tolerant of faults and WAN latency
 - Transaction filtering
- Minimal changes to apps / migration



How does Continuent Tungsten fill the gaps?

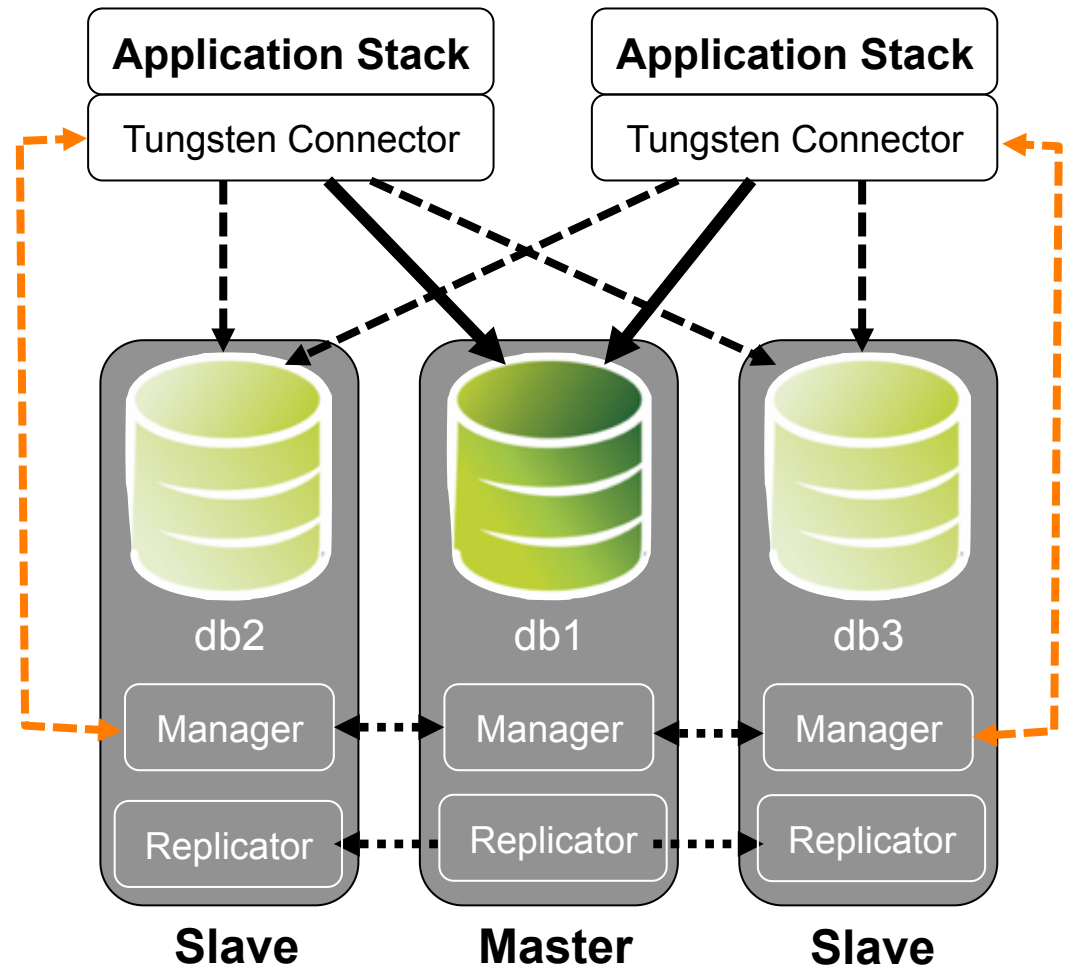
Robert Hodges

Tungsten picks up where MySQL leaves off

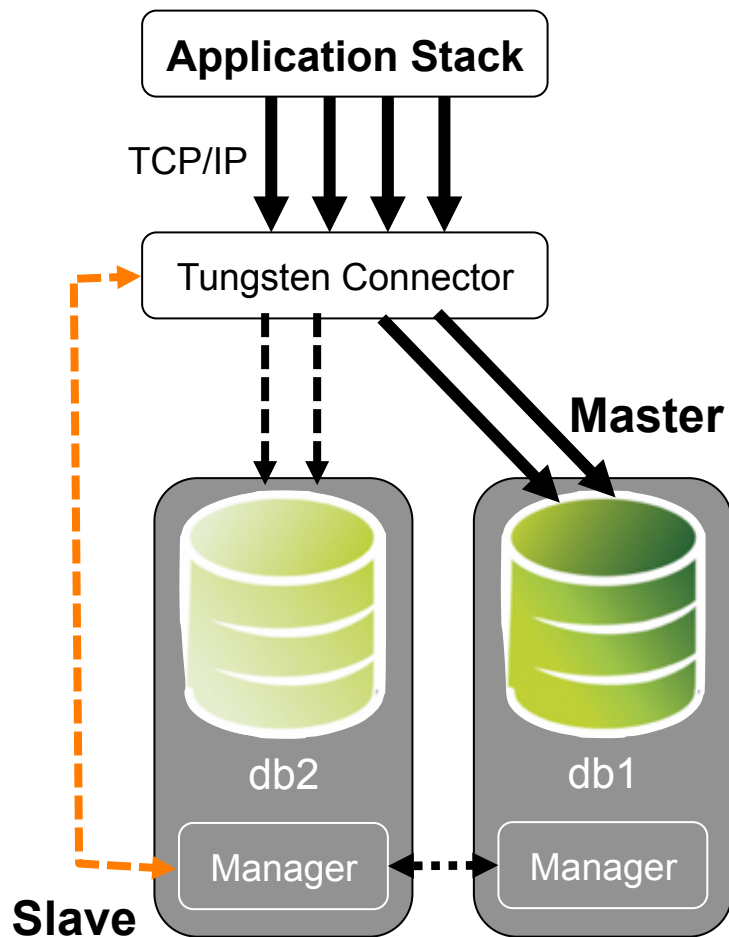
- Handle local DBMS server failures
- Perform maintenance while apps are running
- Load balance SQL across DBMS
- Create disaster recovery sites
- Enable multi-master replication
- Replicate transactions in real-time to other DBMS types

Tungsten Cluster Architecture

- Off-the-shelf MySQL build
- Async replicators
- Peer-to-peer management
- Network-based connectors



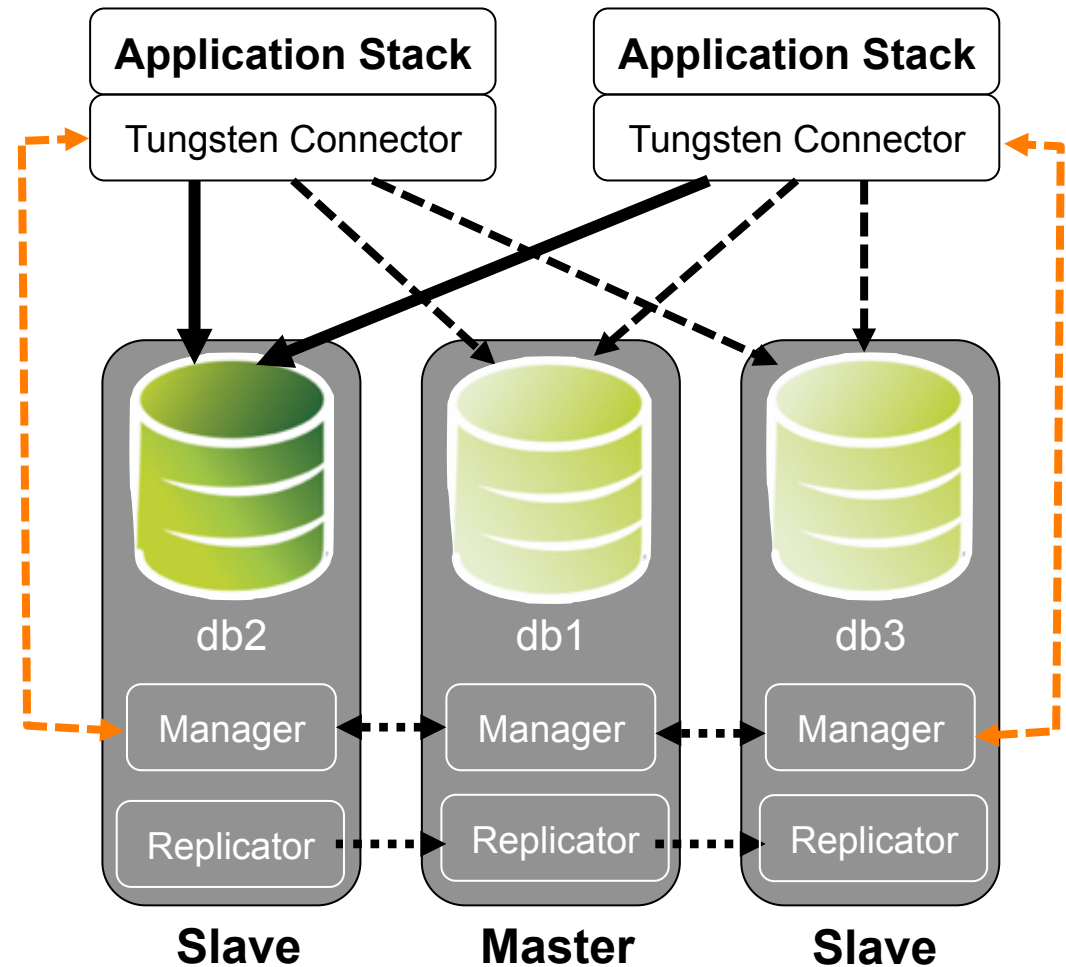
How connectivity works



- Transparent: Implements MySQL wire protocol
- Runs anywhere on network
- Routes transactions to master by default
- Optional read/write splitting
- Reads load-balanced across all slaves

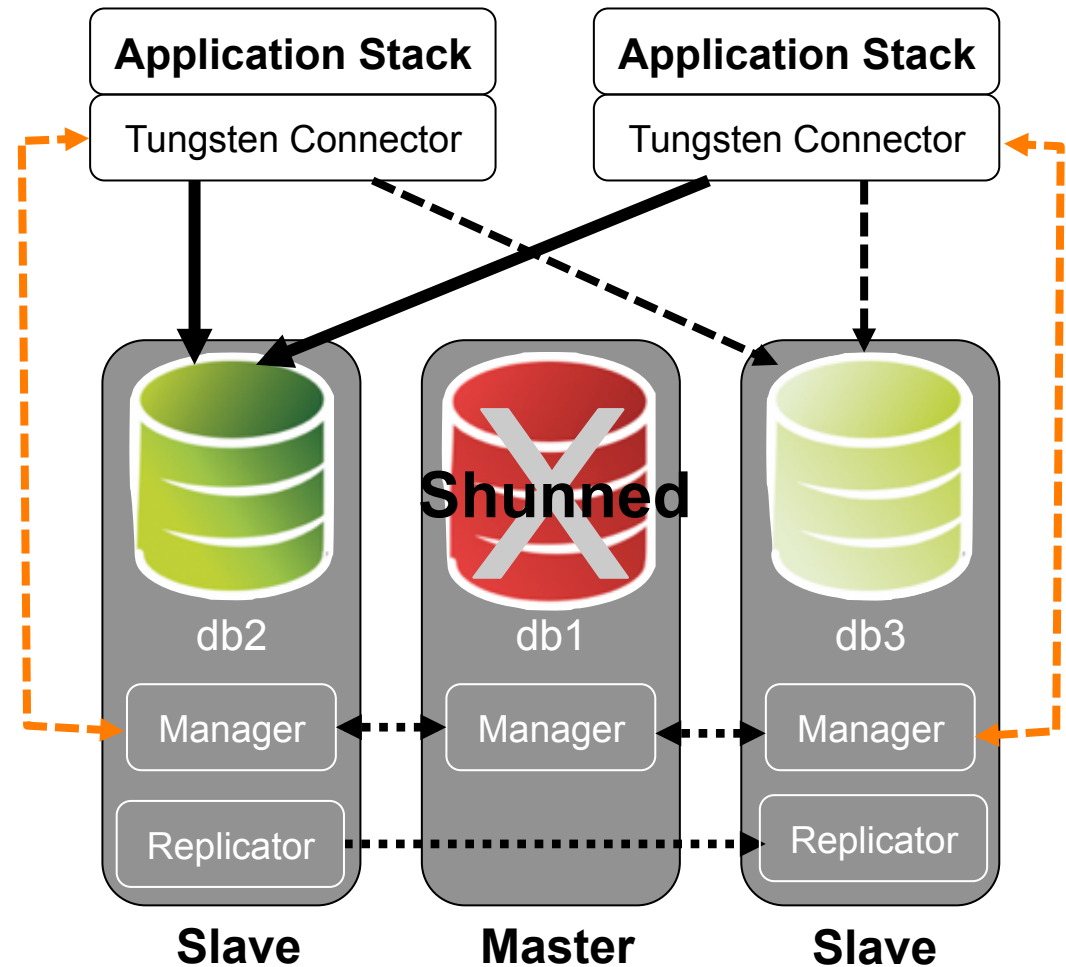
Graceful failover

- Triggered by command
- Zero-data loss
- Transparent to applications
- Automatic reconfiguration



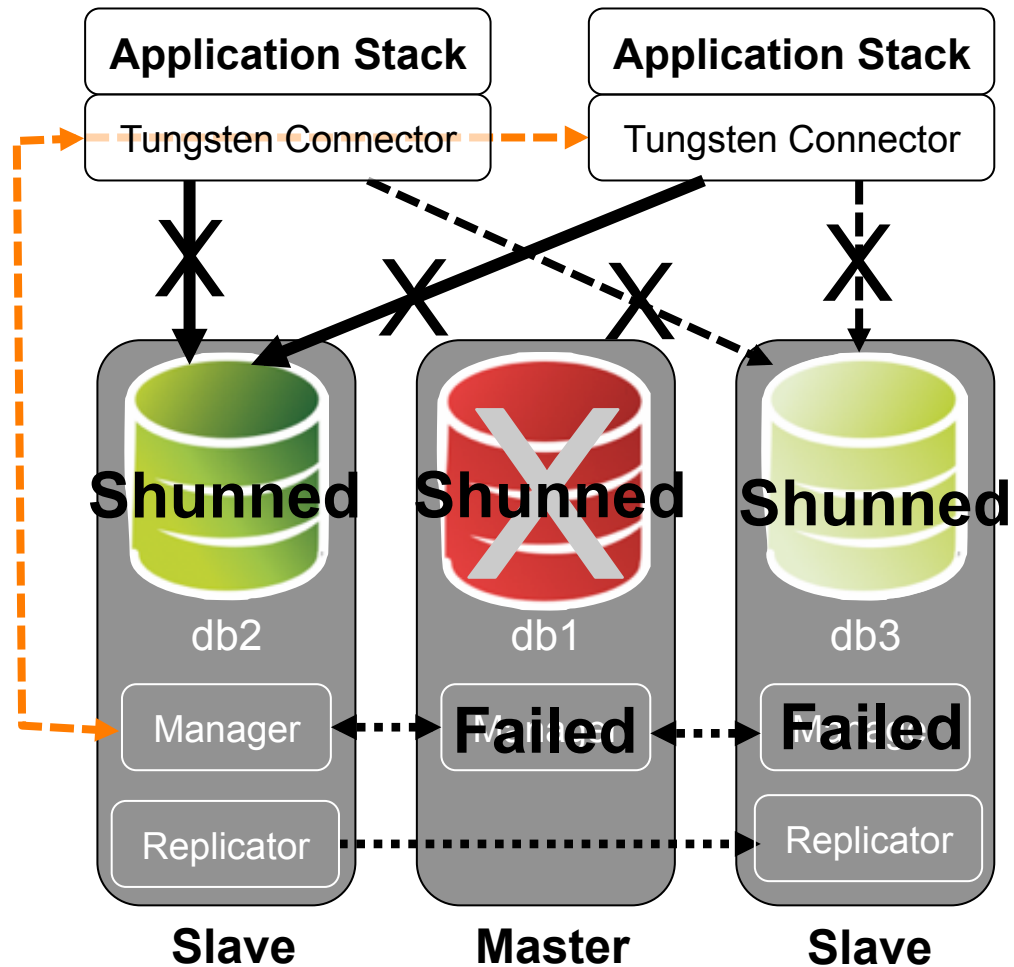
Hard failover – MySQL crash

- Triggered by automatic rules
- Within seconds for MySQL crash
- Up to 1 minute after host crash
- Admin required for recovery



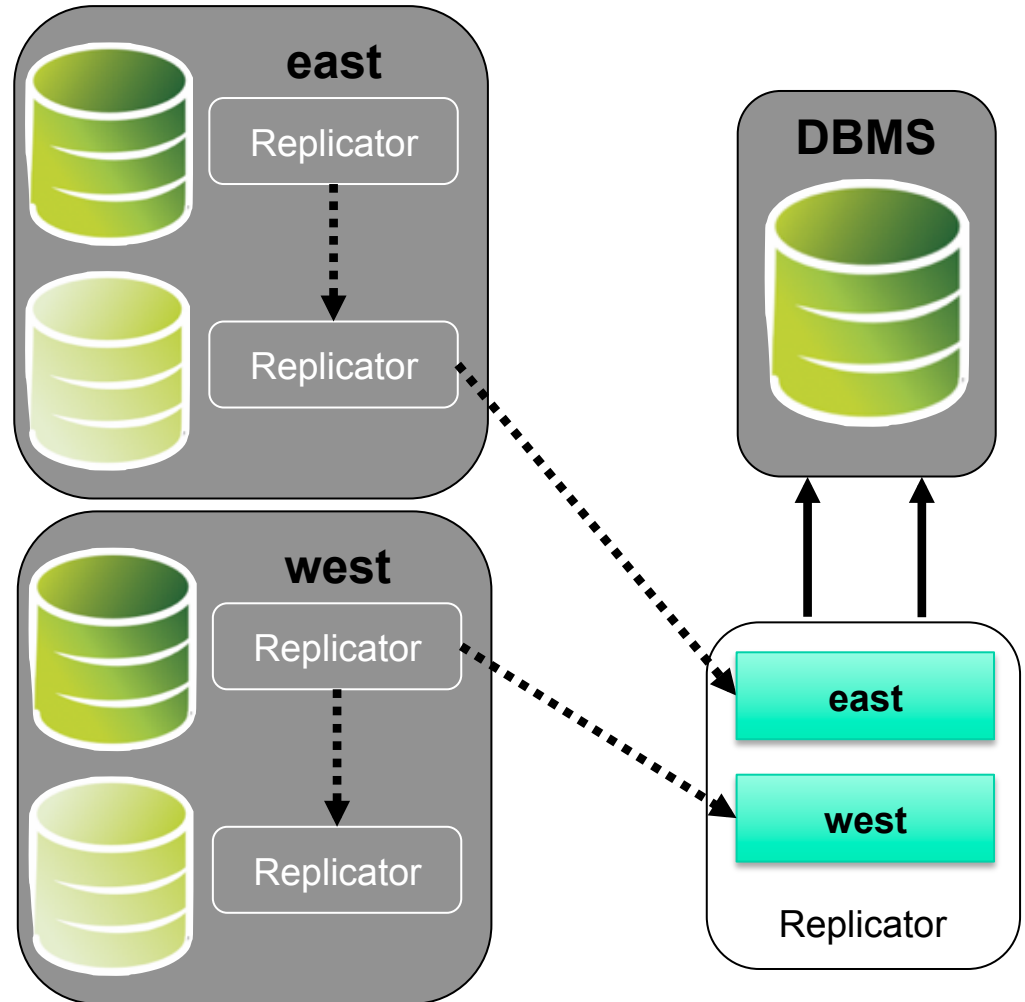
Hard failover – quorum loss

- Managers maintain quorum
- Connectivity stops when lost
- Works across Amazon AZs
- ...But not regions

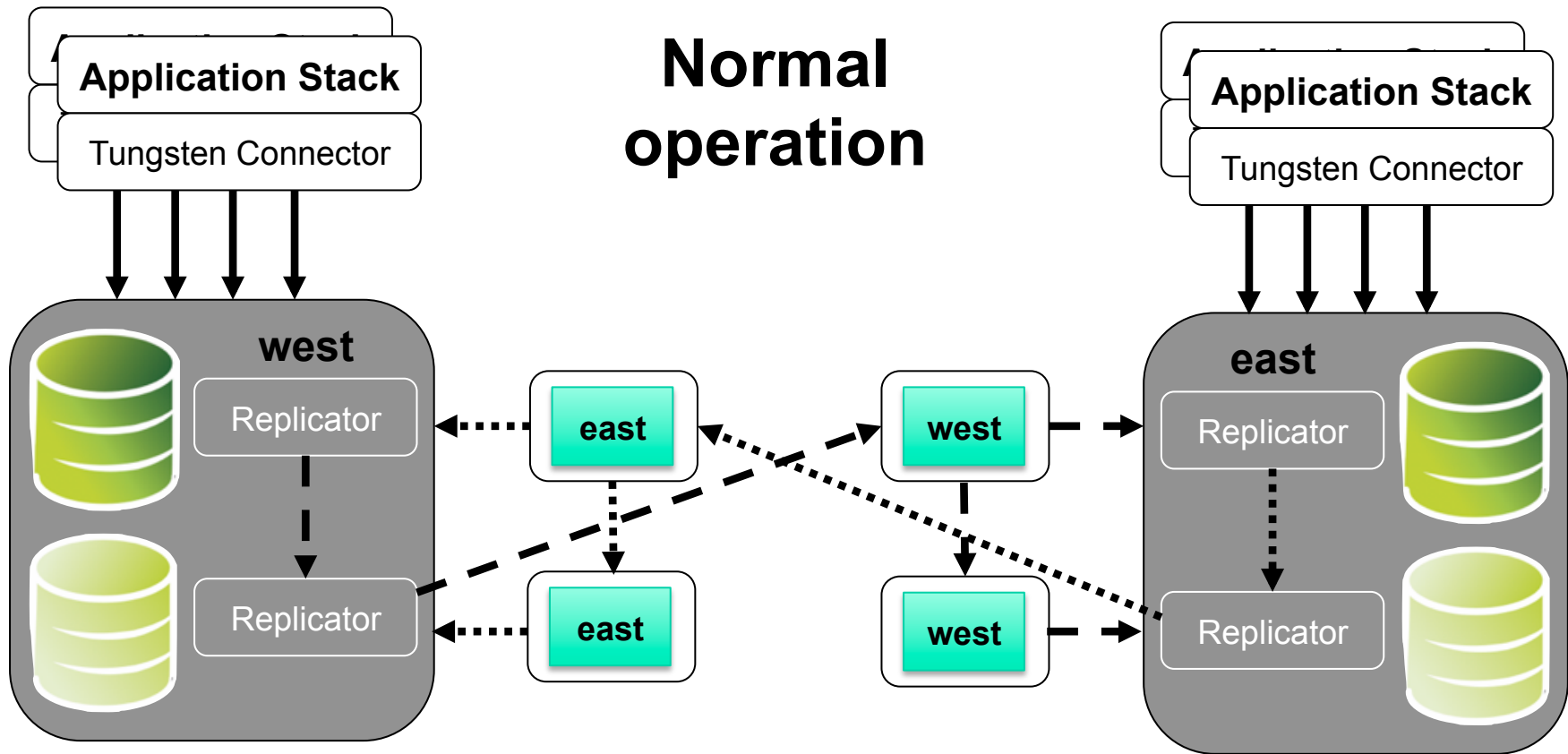


How replication works

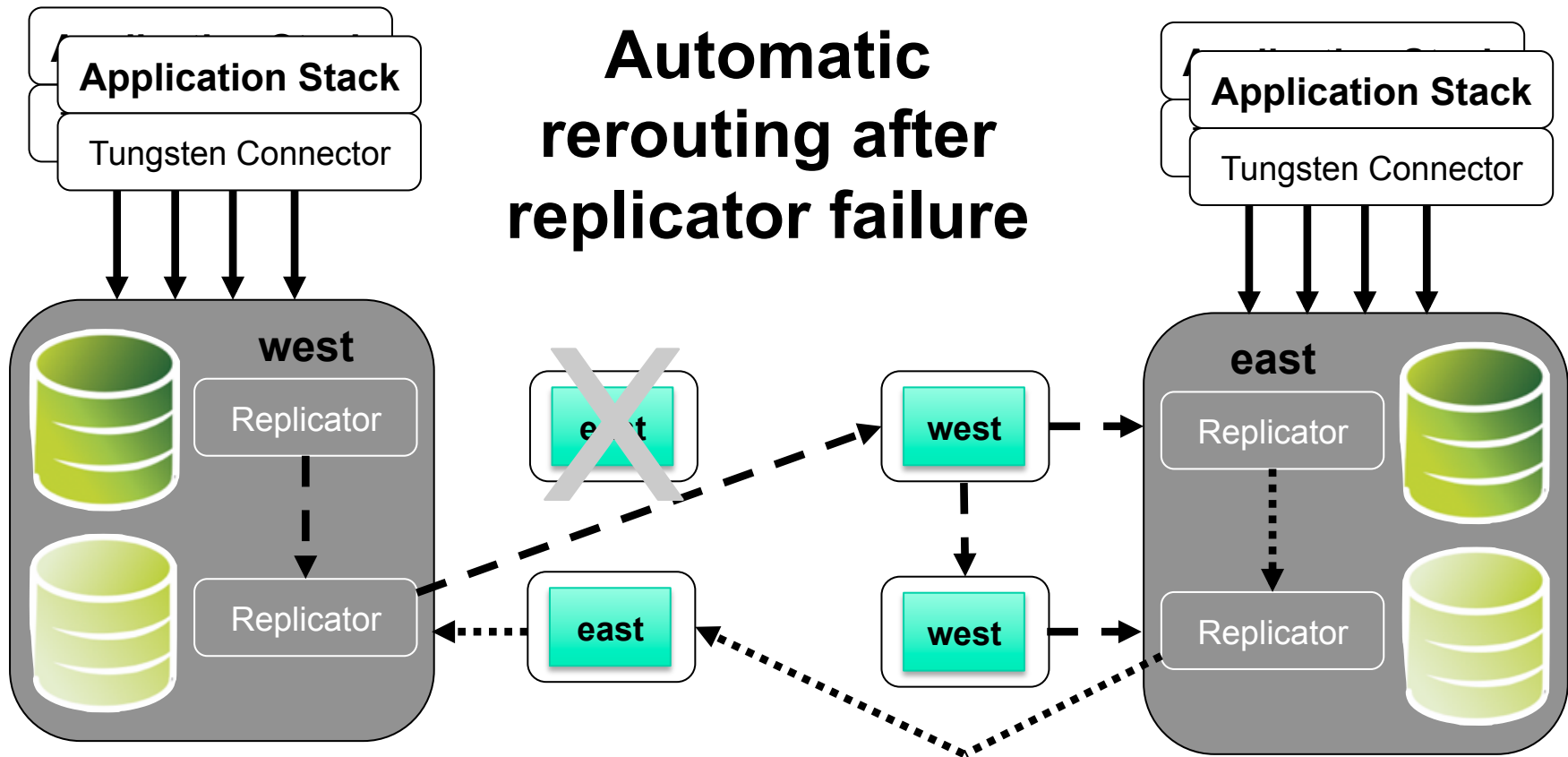
- Moves transactions between points
- Can replicate between clusters
- Can run multiple services
- Global transaction IDs
- Very flexible topologies
- Transaction filtering



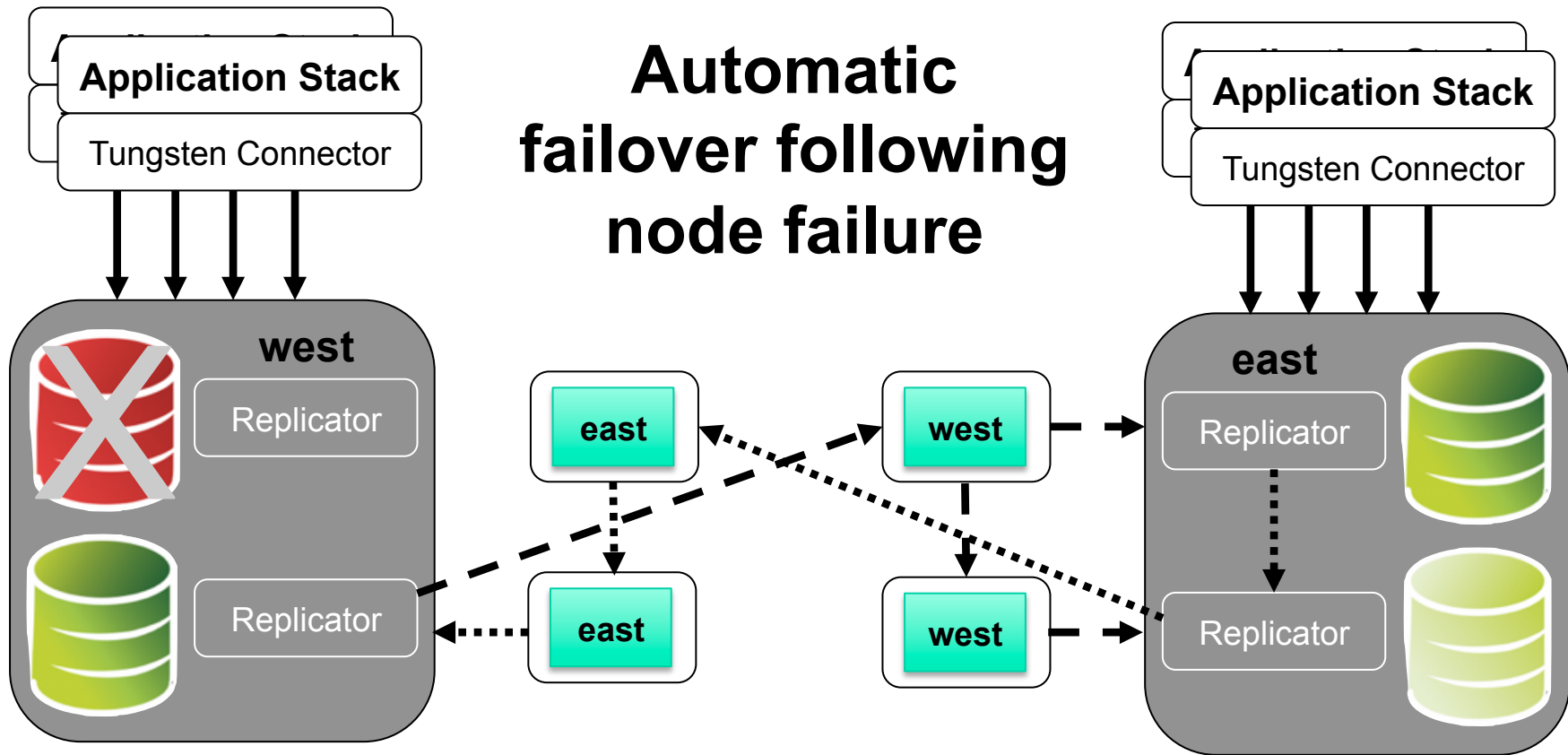
Cross-site multi-master with independent clusters



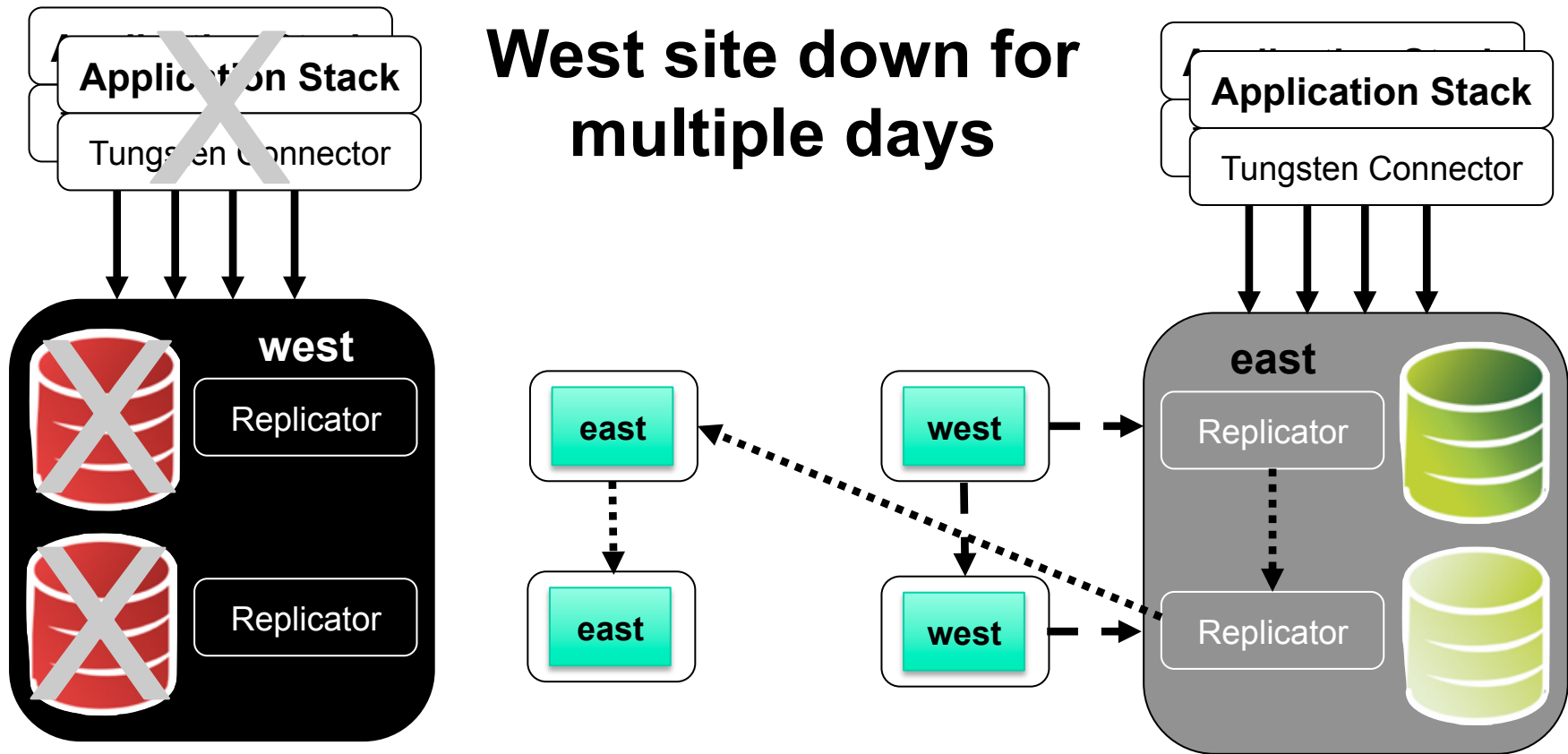
Multi-master replicator failure



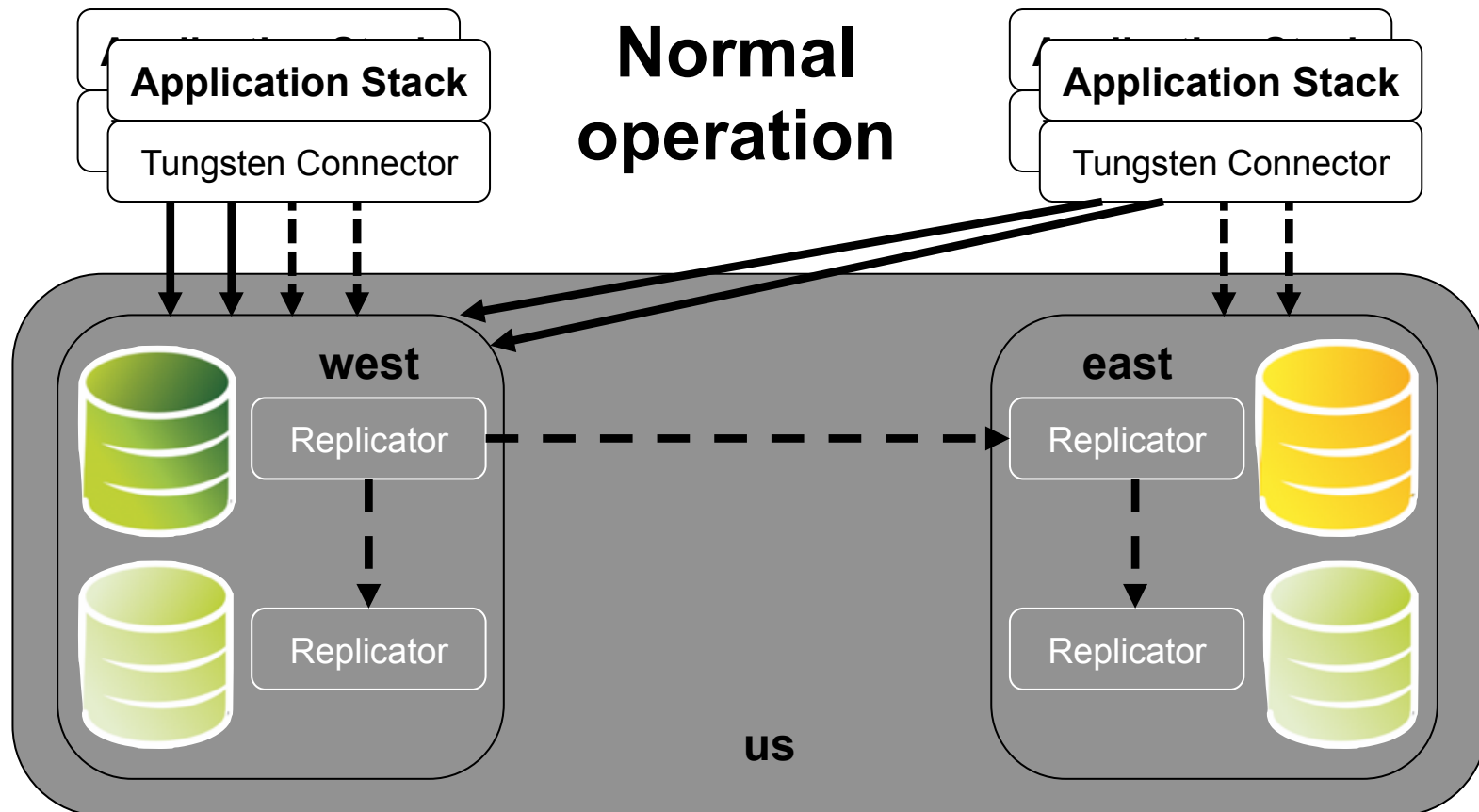
Multi-master node failure



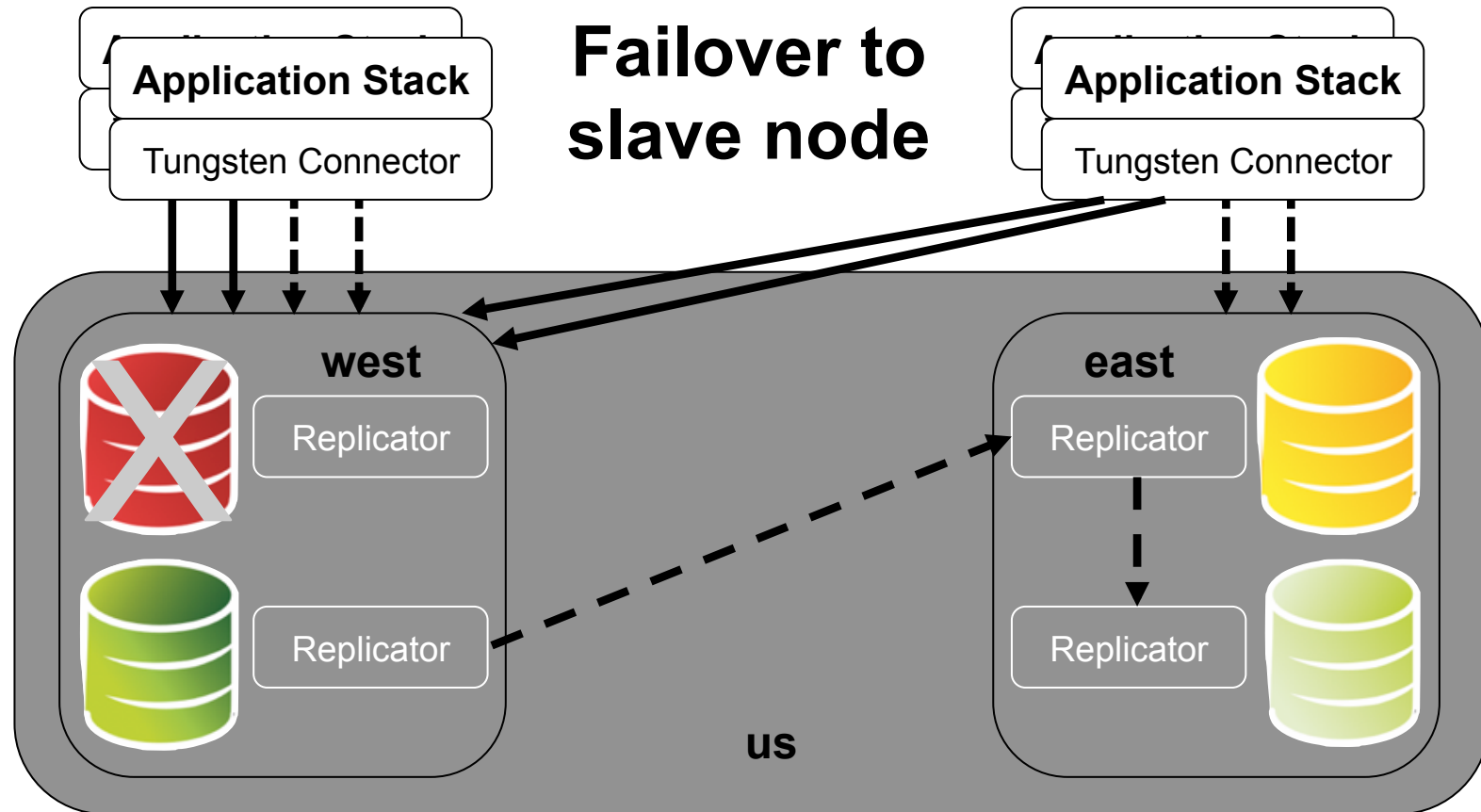
Full site failure



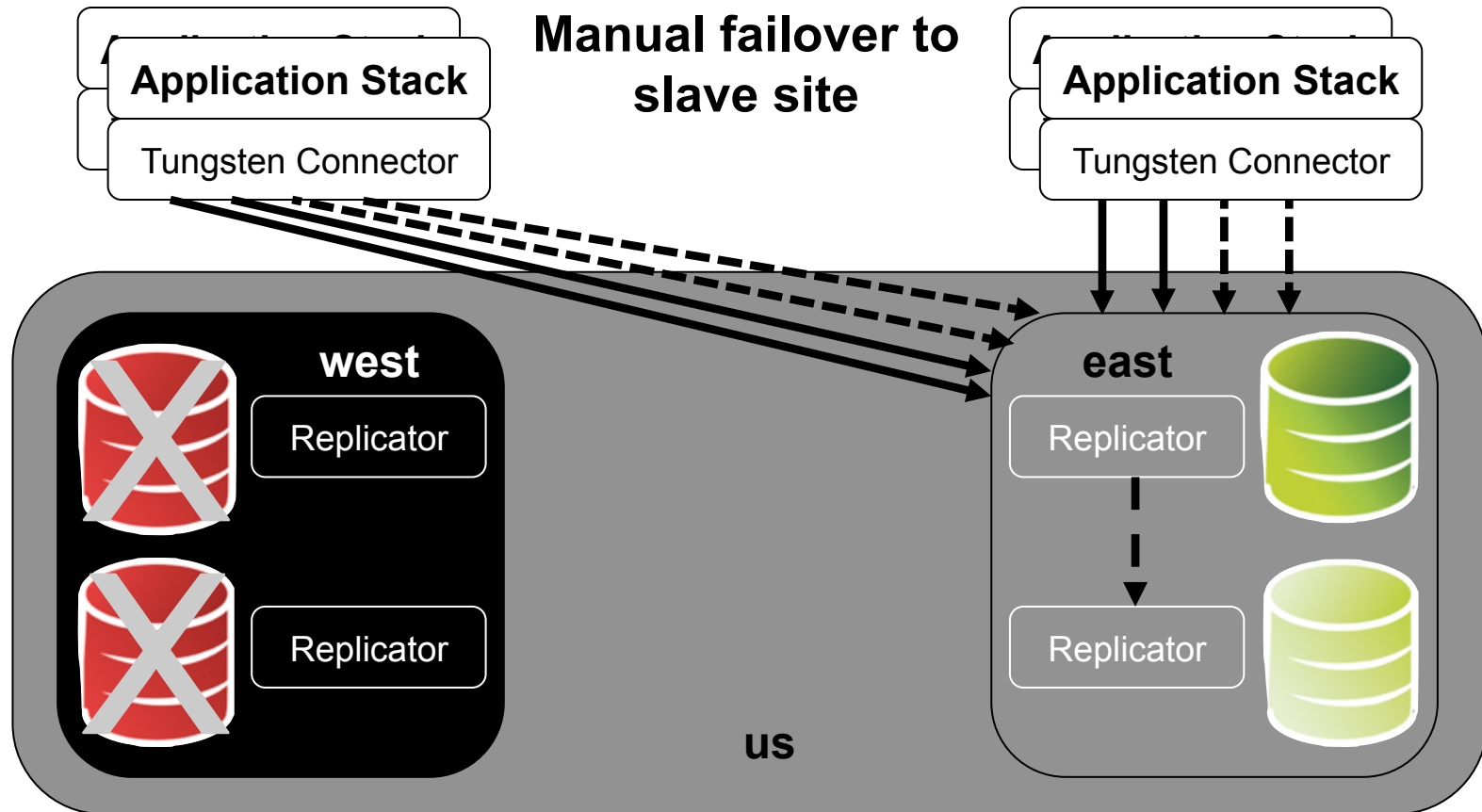
Cross-site primary/DR with composite data service



Composite service node failure



Composite service site failure

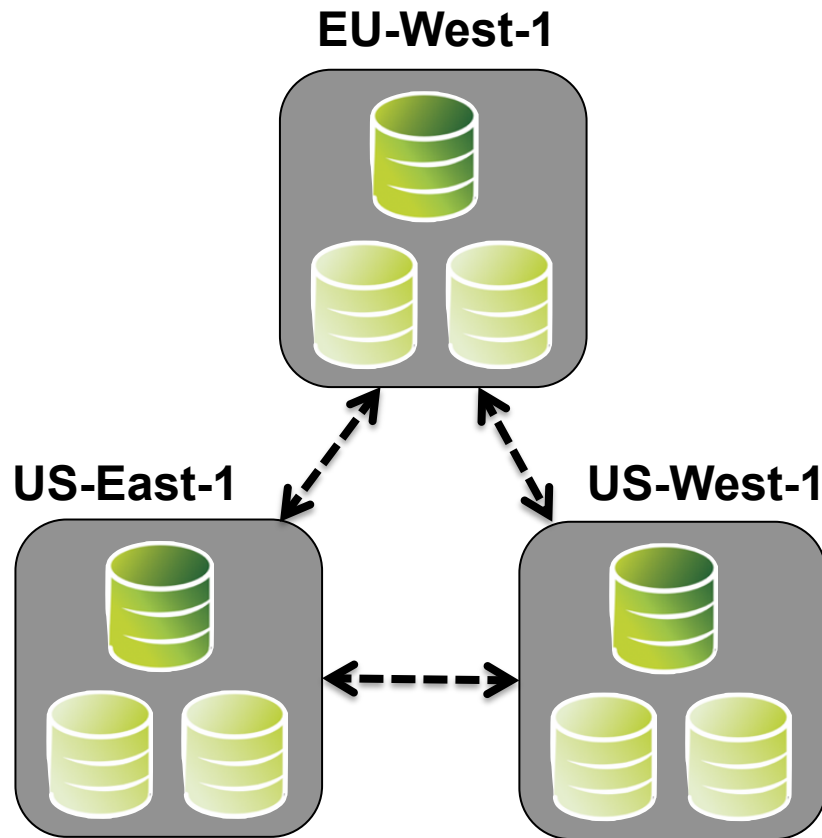




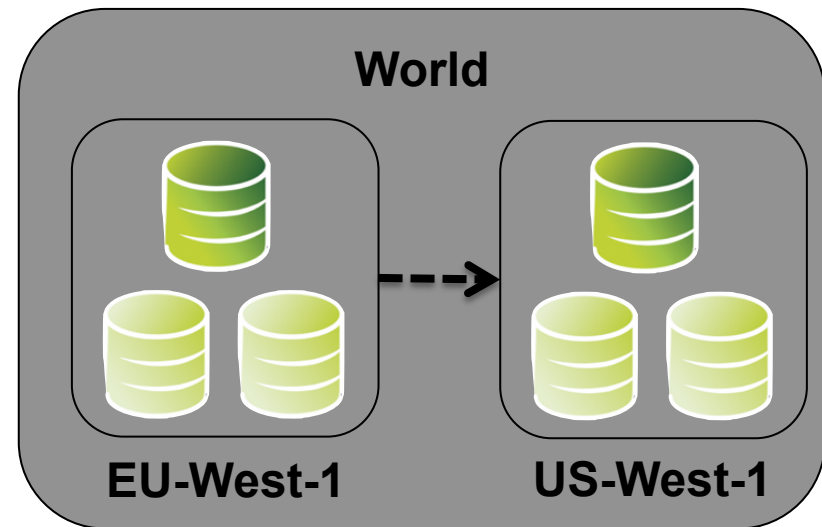
Demonstration

Peter Boros

Demo Architecture



Multi-Master Topology



Composite Data Service



Practical steps to multi-data center MySQL

Robert Hodges

Setting Up Multi-DC Clustering

- Tungsten tpm – Sets up local clusters and replication links between them
 1. Put tungsten.ini file on host
 2. Download software (rpm or tarball)
 3. Run 'tpm install'

Ini File for Composite Service

```
[defaults]
home-directory=/opt/continuent
user=tungsten
replication-user=tungsten
replication-password=password
application-user=sbtest
application-password=sbtest
mysql-connectorj-path=/home/tungsten/mysql-connector-java-5.1.26-
bin.jar
start-and-report=true
[eu]
master=tw-eu1
slaves=tw-eu2,tw-eu3
connectors=tw-eu1,tw-eu2,tw-eu3
[west]
master=tw-usw1
relay-enabled=true
relay-source=eu
slaves=tw-usw2,tw-usw3
connectors=tw-usw1,tw-usw2,tw-usw3
[world]
composite-datasources=eu,west
```

Composite Data Services – Pros and Cons

- Key benefits:
 - No conflicts
 - No app changes
 - Failover is transparent
 - Works with statement replication
- Things to consider and test:
 - High latency writes to primary master
 - Managing unplanned site failover
 - Potential coupling between sites

Multi-Master Data Services – Pros and Cons

- Key benefits
 - Sites are completely uncoupled
 - No failover!
 - Minimal write latency
- Things to consider and test carefully:
 - Potential for conflicts
 - App changes often required
 - Row replication strongly recommended

Workloads that handle multi-master very well

- Append-only data
- Tables that use only auto-increment keys
- Sharded apps where writes to each shard are in a particular region (system of record)

Where to get started

- Cluster and replication documentation
 - <https://docs.continuent.com>
- Software
 - <https://code.google.com/p/tungsten-replicator/>
 - <http://www.continuent.com/downloads/software>
- Percona and Continuent Websites

Thank You!

Peter Boros

peter.boros@percona.com

Robert Hodges

robert.hodges@continuent.com