MySQL Cluster: ‘What’ and ‘How’.

Keith Hollman
Principal Sales Consultant EMEA
MySQL

keith.hollman@oracle.com

Agenda

1. Introduction
2. Starting: configuration/installation/starting/stopping
3. The simplest way: Cluster Manager.

Oracle MySQL HA & Scaling Solutions

	MySQL Replication	MySQL Fabric	Oracle VM Template	Oracle Clusterware	Solaris Cluster	Windows Cluster	DRBD	MySQL Cluster
App Auto-Failover	✖	✔	✔	✔	✔	✔	✔	✔
Data Layer Auto-Failover	✖	✔	✔	✔	✔	✔	✔	✔
Zero Data Loss	MySQL 5.7	MySQL 5.7	✔	✔	✔	✔	✔	✔
Platform Support	All	All	Linux	Linux	Solaris	Windows	Linux	All
Clustering Mode	Master + Slaves	Master + Slaves	Active/Passive	Active/Passive	Active/Passive	Active/Passive	Active/Passive	Multi-Master
Failover Time	N/A	Secs	Secs +	Secs +	Secs +	Secs +	Secs +	< 1 Sec
Scale-out	Reads	✔	✖	✖	✖	✖	✖	✔
Cross-shard operations	N/A	✖	N/A	N/A	N/A	N/A	N/A	✔
Transparent routing	✖	For HA	✔	✔	✔	✔	✔	✔
Shared Nothing	✔	✔	✖	✖	✖	✖	✔	✔
Storage Engine	InnoDB+	InnoDB+	InnoDB+	InnoDB+	InnoDB+	InnoDB+	InnoDB+	NDB
Single Vendor Support	✔	✔	✔	✔	✔	✖	✔	✔

ORACLE

When to consider MySQL Cluster

- What are the consequences of downtime or failing to meet performance requirements?
- How much effort and $ is spent in developing and managing HA in your applications?
- Are you considering sharding your database to scale write performance? How does that impact your application and developers?
- Do your services need to be real-time?
- Will your services have unpredictable scalability demands, especially for writes ?
- Do you want the flexibility to manage your data with more than just SQL ?

When NOT to consider MySQL Cluster

- Most 3rd party applications
- Long running transactions
- Geospatial indexes
- Huge dataset (>2TB)
- Complex access pattern to data and many full table scans
- When you need a disk based database like InnoDB

Introduction to MySQL Cluster

ORACLE

History of MySQL Cluster ”NDB”

The Network DataBase NDB

- MySQL Cluster aka Network DataBase NDB
- Designed/Developed at Ericcson in late 90’s
- Original design paper: ”Design and Modeling of a Parallel Data Server for Telecom Applications” from 1997 by Michael Ronström
- Originally written in PLEX (Programming Language for EXchanges) but later converted to C++.
- MySQL AB acquired Alzato (owned by Ericsson) late 2003.

History of MySQL Cluster ”NDB”

The Network DataBase NDB

- Databases services back then:
  - SCP/SDP (Service Control/Data Point) in Intelligent Networks.
  - HLR (Home Location Register) for keeping track of mobile phones/users.
  - Databases for network management especially real-time charging information.

History of MySQL Cluster ”NDB”

The Network DataBase NDB

- NDB was designed to:
  - Reliability, the availability class of the telecom databases should be 6. This means that downtime must be less than 30 seconds per year. This means that no planned down time of the system is allowed.
  - Performance, designed for high throughput, linear scalabillity when adding more servers (data nodes) for simple access patterns (PK lookups).
  - Real-time, data is kept in memory and system is designed for memory operations.

MySQL Cluster resumido

Scaling Reads & Writes	Auto-sharding + Multi-master Transactional, ACID-compliant relational database
99.999% Availability	Shared-nothing design, no Single Point of Failure On-Line operations: Scale, Upgrade Schema, etc.
Real-Time Responsiveness	High-load, real-time performance Predictable low latency, bounded access times
SQL & NoSQL APIs	Complex, relational queries + Key/Value Access MySQL, Memcached, C++, Java, JPA, HTTP / REST
Low TCO, Open platform	GPL & Commercial editions Commodity hardware, management & monitoring tools

ORACLE

MySQL Cluster Components

SQL Node

(Applications)

- Standard SQL interface
- Scale out for performance
- Enables Geo Replication

NDB API

(Applications)

- Real-time applications
- C++/Java APIs
- Automatic failover & load balancing

Data Node

(Data Storage)

- Data storage (Memory & Disk)
- Automatic & User defined data partitioning
- Scale out for capacity and performance

MGM Node

(Management)

- Management, Monitoring & Configuration
- Arbitrator for split brain/network partitioning
- Cluster logs

Data Nodes

- Stores data and indexes
  - In memory
  - Non-indexed data possible on disk
  - Contains several blocks, most important, LQH, TUP, ACC and TC.
- Data check pointed to disk “LCP”
- Transaction coordination
- Handling fail-over
- Doing online backup

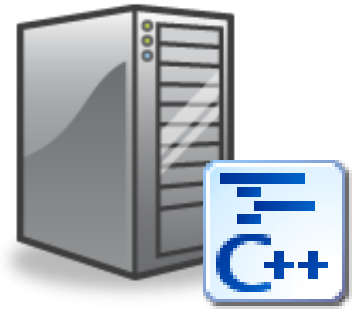
- All connect to each other
- Up to 48
  - Typically 2, 4.

Management Nodes

- Distributing configuration
- Logging
- Monitoring
- Act as Arbitrator
  - Prevents split-brain
- OK when not running
  - Need to start others
- 1 is minimum, 3 too many, 2 is OK

API Nodes

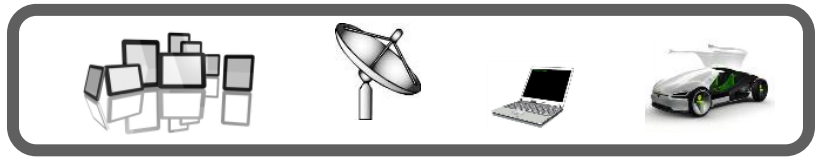
- Applications written using NDB API
  - C/C++/Java
- Fast
  - No SQL parsing
- Examples:
  - NDBCluster storage engine
  - ndb_restore



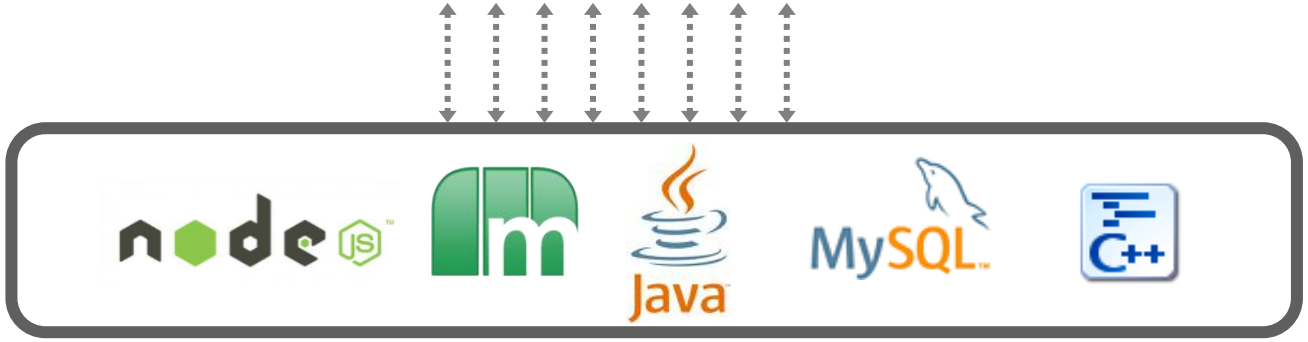
SQL Nodes

- MySQL using NDBCluster engine
  - Is also an API Node
- Transparent for most applications
- Used to create tables
- Used for Geographical Replication
  - Binary logging all changes
- Can act as Arbitrator
- Connects to all Data Nodes

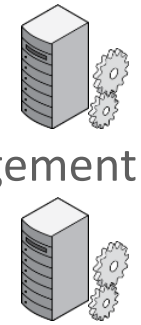
MySQL Cluster Architecture



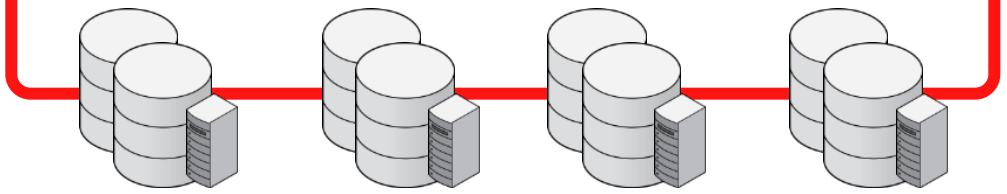
Clients



Application Layer



Management

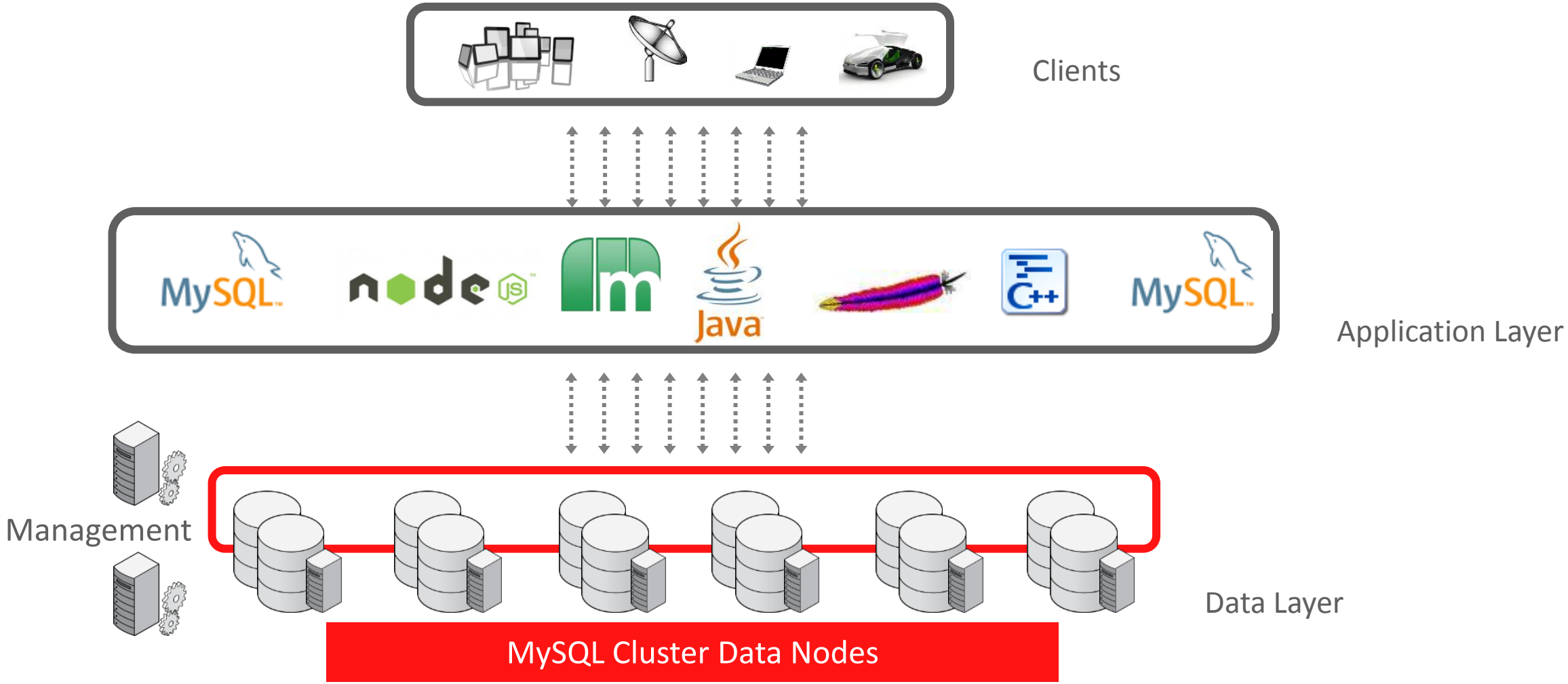


MySQL Cluster Data Nodes

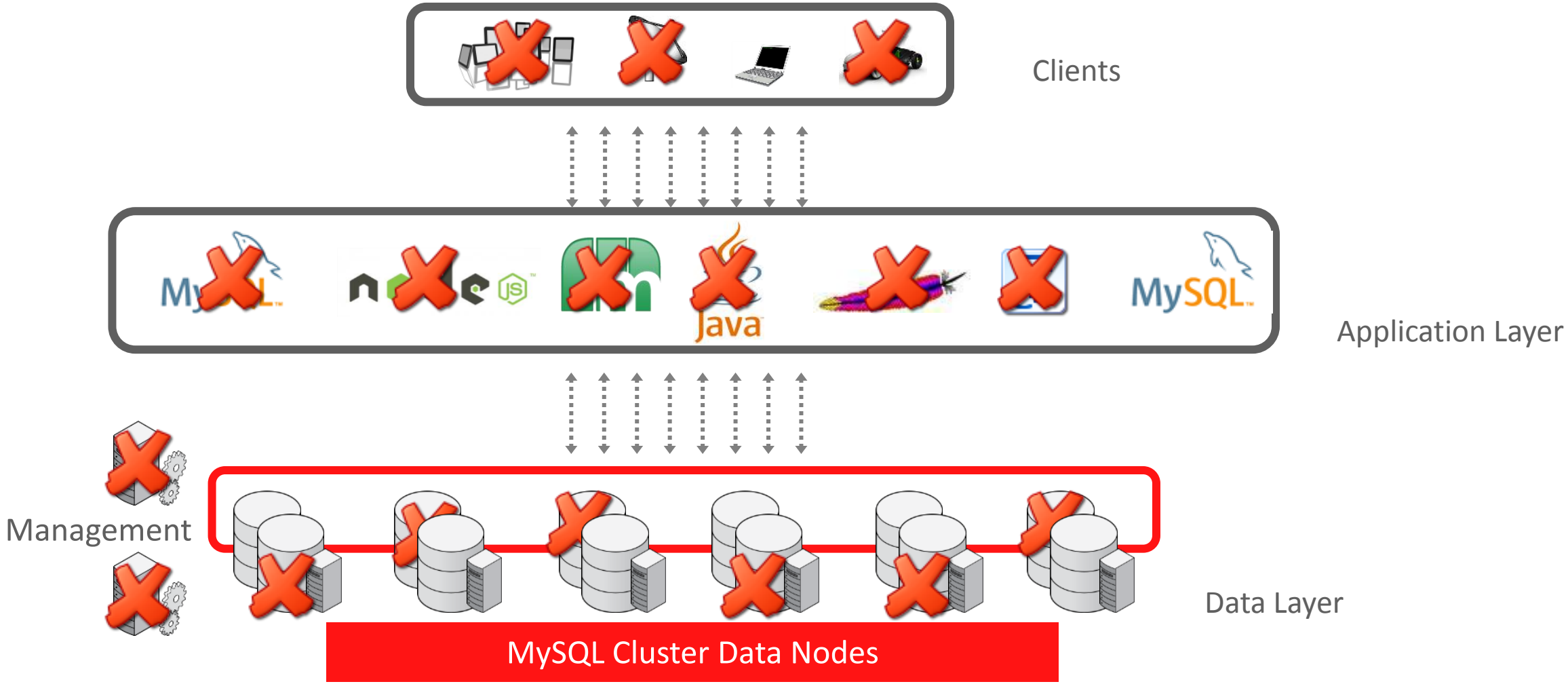
Data Layer

ORACLE

MySQL Cluster Scaling

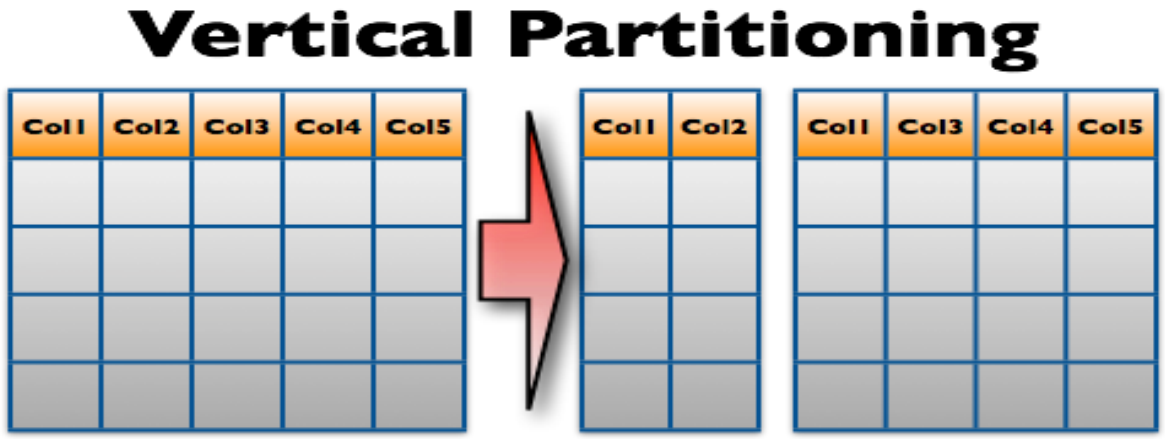


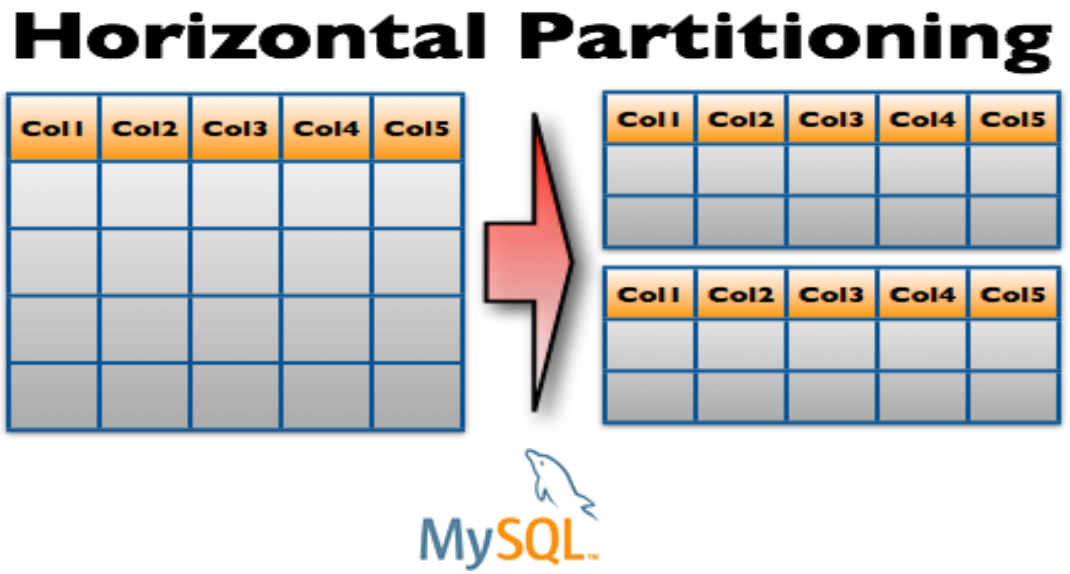
MySQL Cluster - Extreme Resilience



Partitioning I

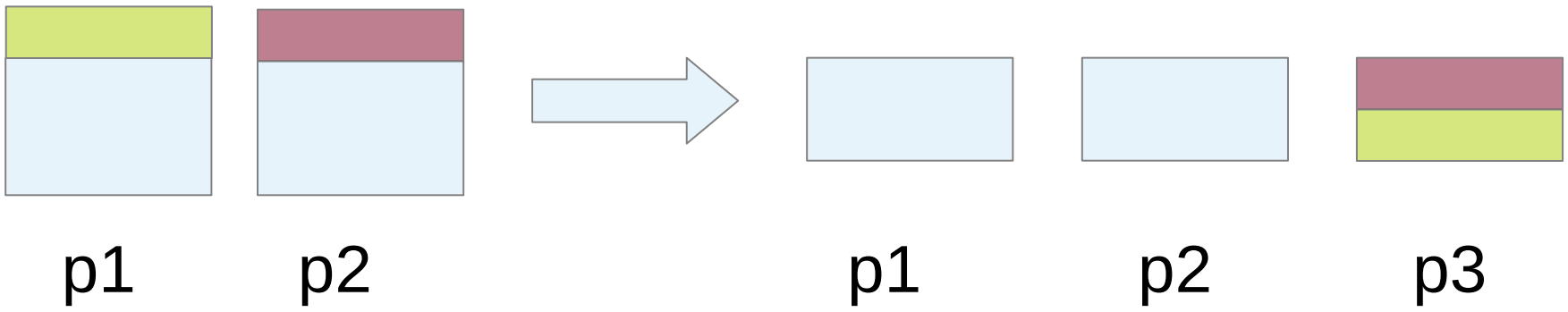
- Vertical Partitioning - 1:1 tables to reduce the size of rows, tables and indexes
- Horizontal Partitioning - 1 table split on multiple tables with different rows





Data Partitioning II

- Data is partitioned on primary key per default
- HASH value of PK, only selective if you provide full PK not “left most”
- Linear hashing, data is only moved away (low impact of reorganize)



Automatic Data Partitioning

Table T1

ID	FirstName	LastName	Email	Phone	Px	Partition
					P1	
					P2	
					P3	
					P4	

- A partition is a portion of a table
- Number of partitions = number of data nodes
- Horizontal partitioning

Automatic Data Partitioning

Table T1

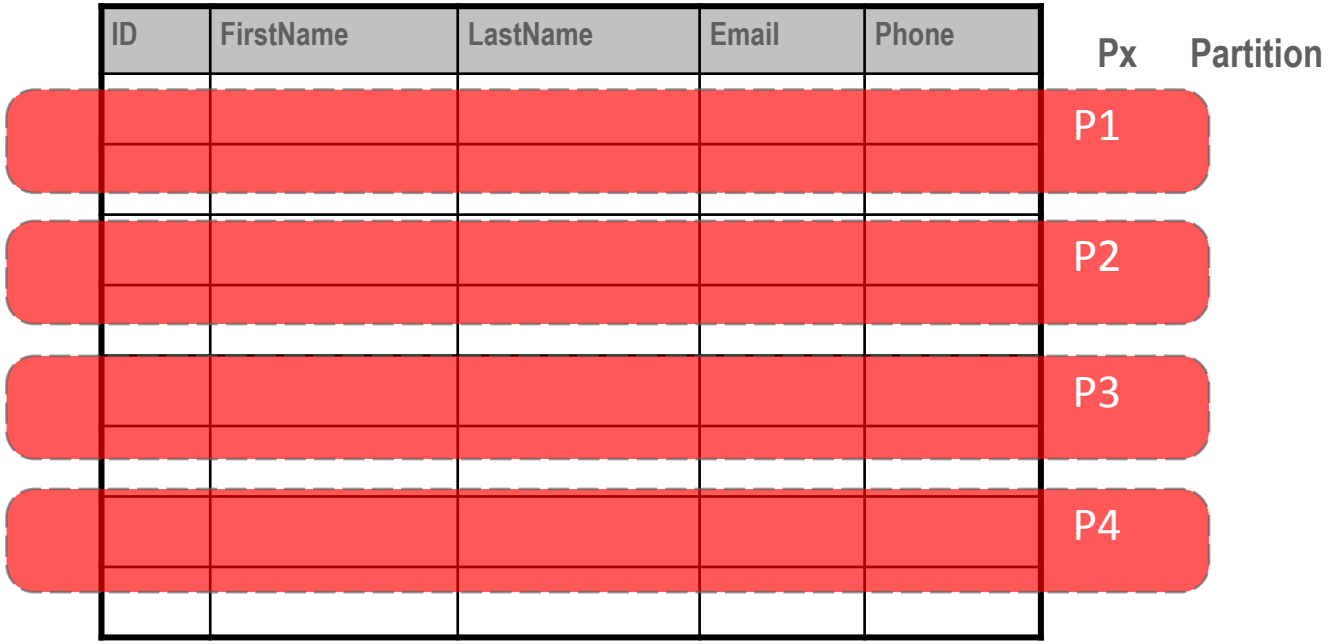
ID	FirstName	LastName	Email	Phone	Px	Partition
					P1	
					P2	
					P3	
					P4	

A fragment is a partition
Number of fragments = # of partitions * # of replicas

Automatic Data Partitioning

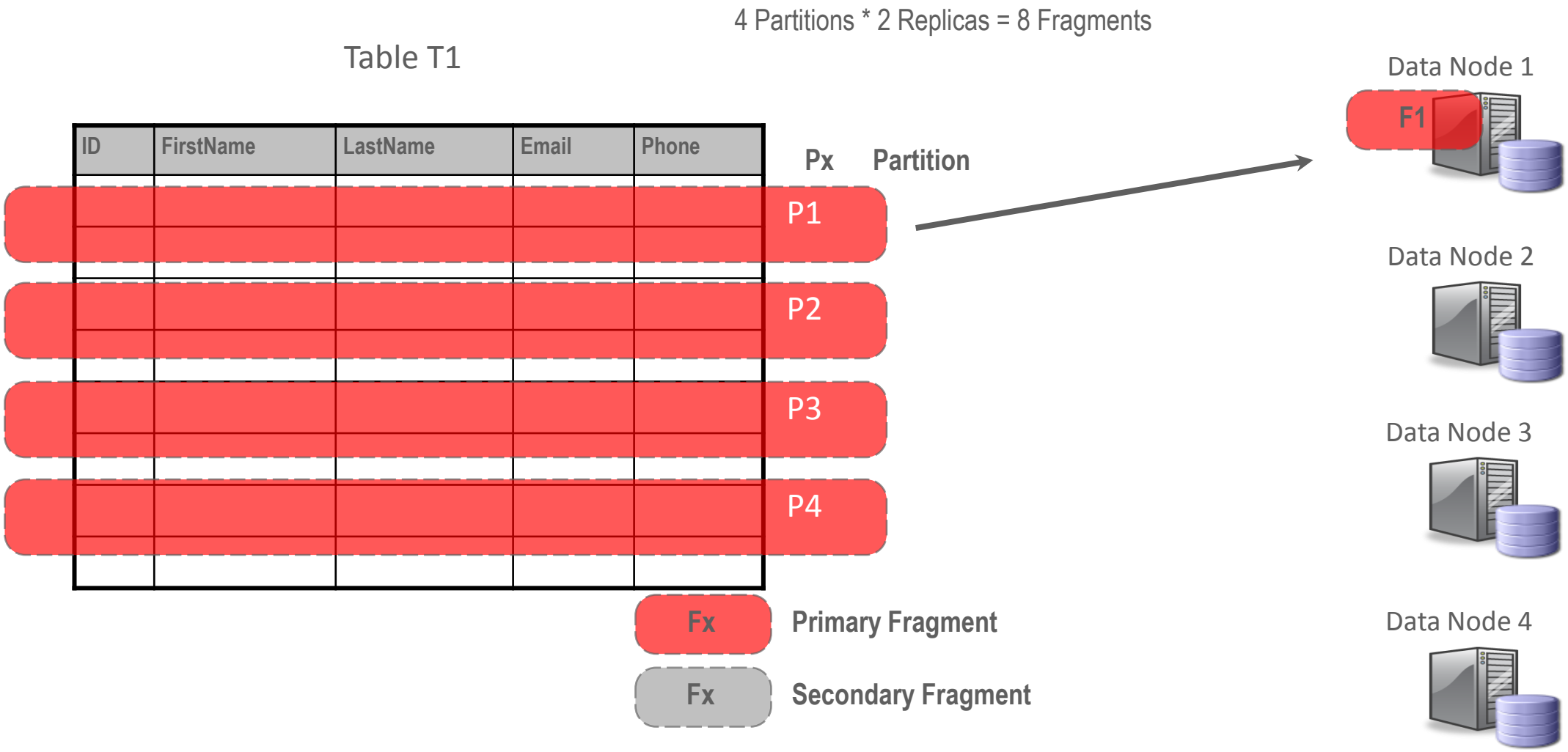
4 Partitions * 2 Replicas = 8 Fragments

Table T1

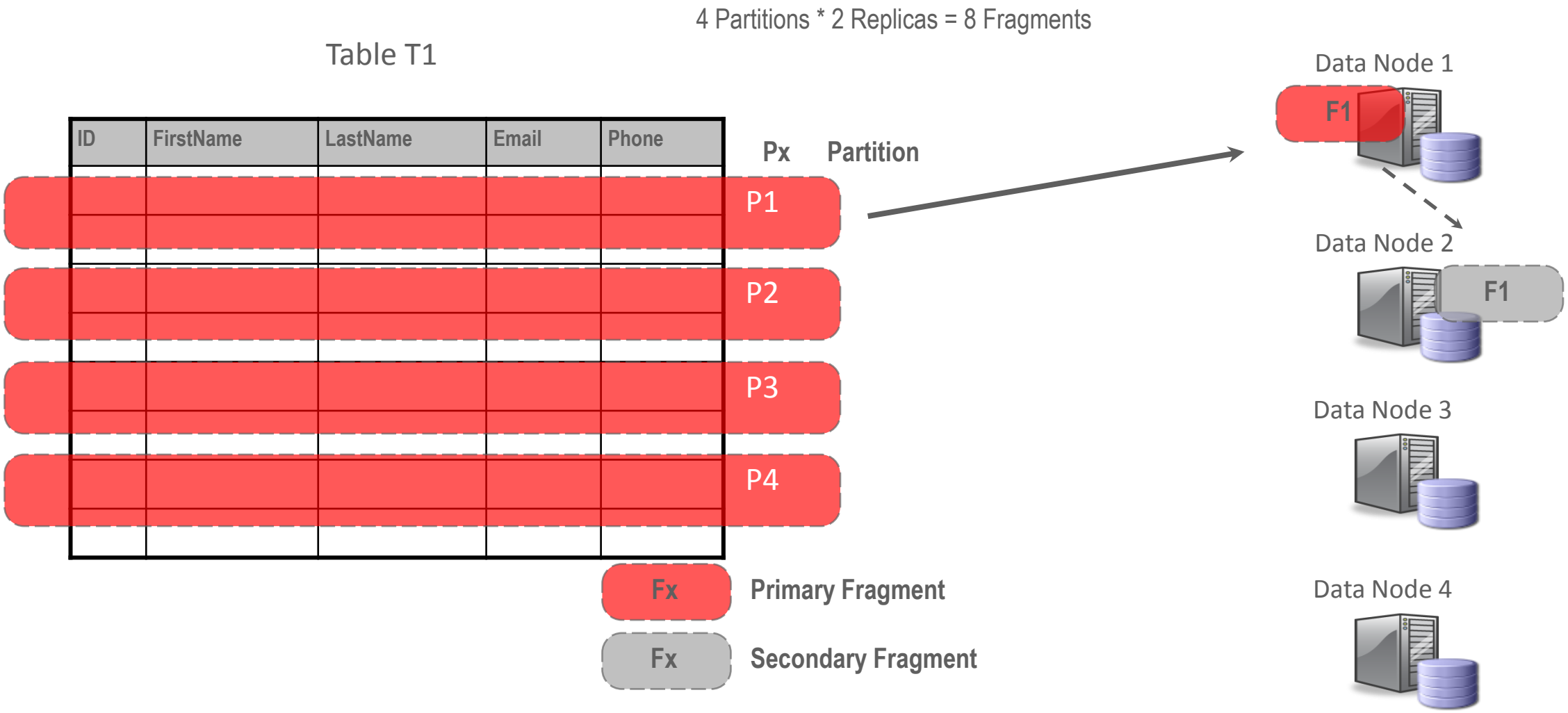


A fragment can be primary or secondary/backup
Number of fragments = # of partitions * # of replicas

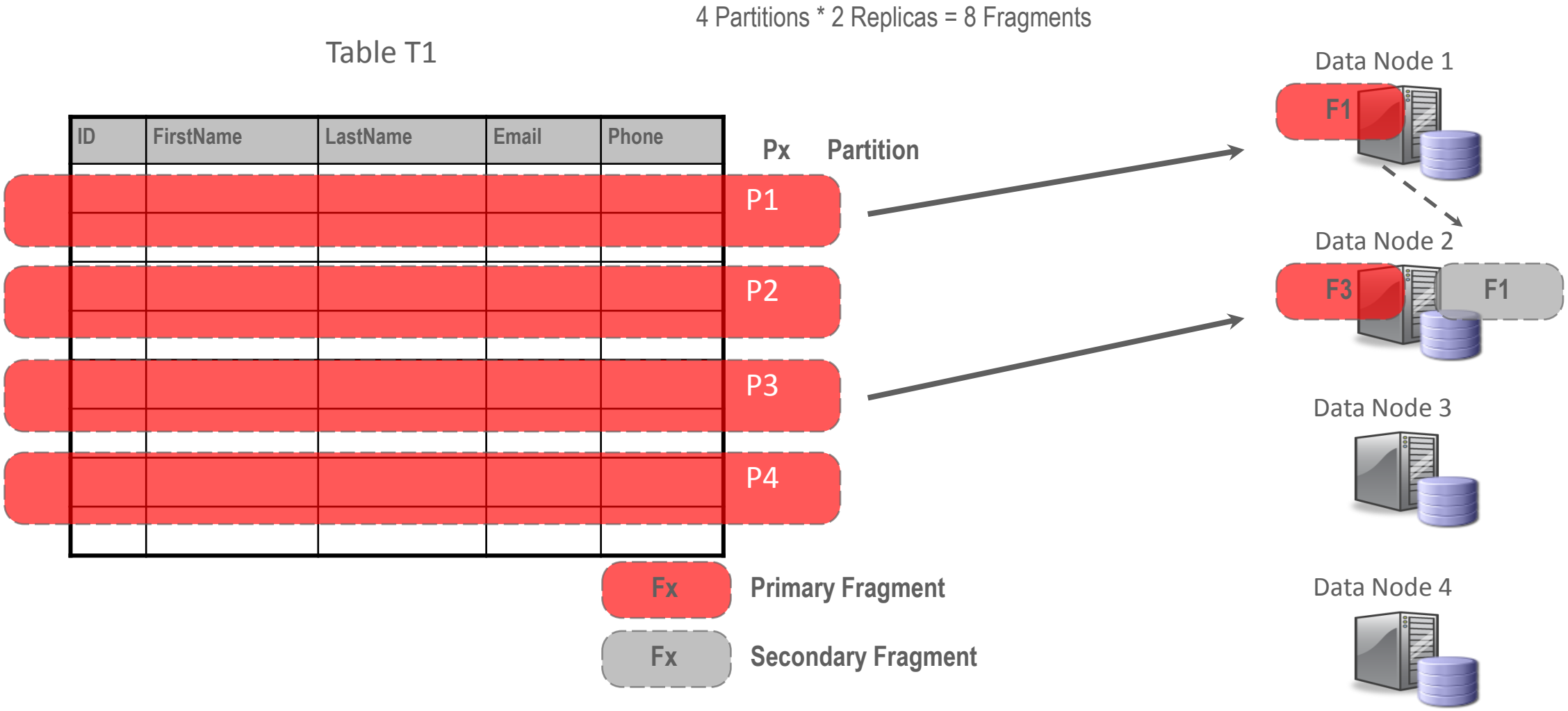
Automatic Data Partitioning



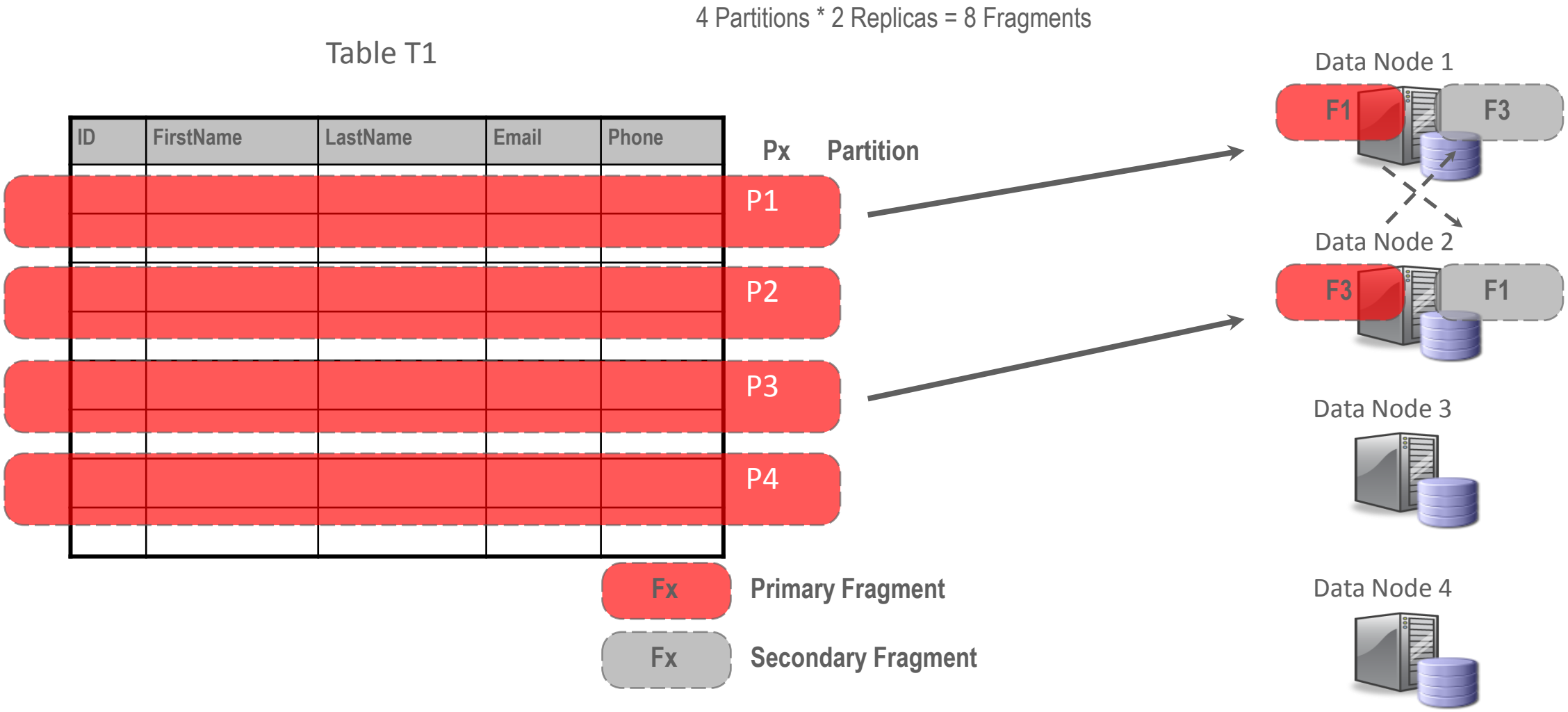
Automatic Data Partitioning



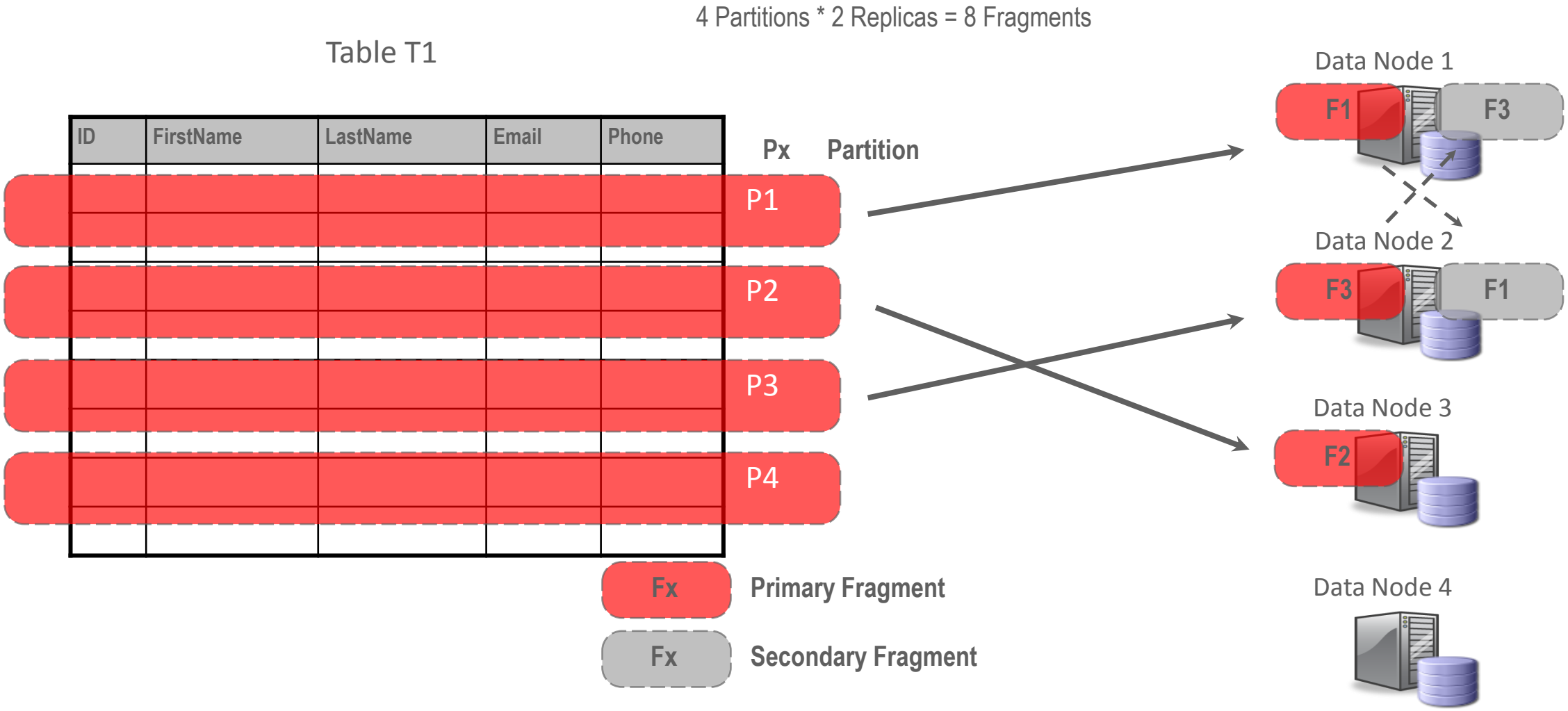
Automatic Data Partitioning



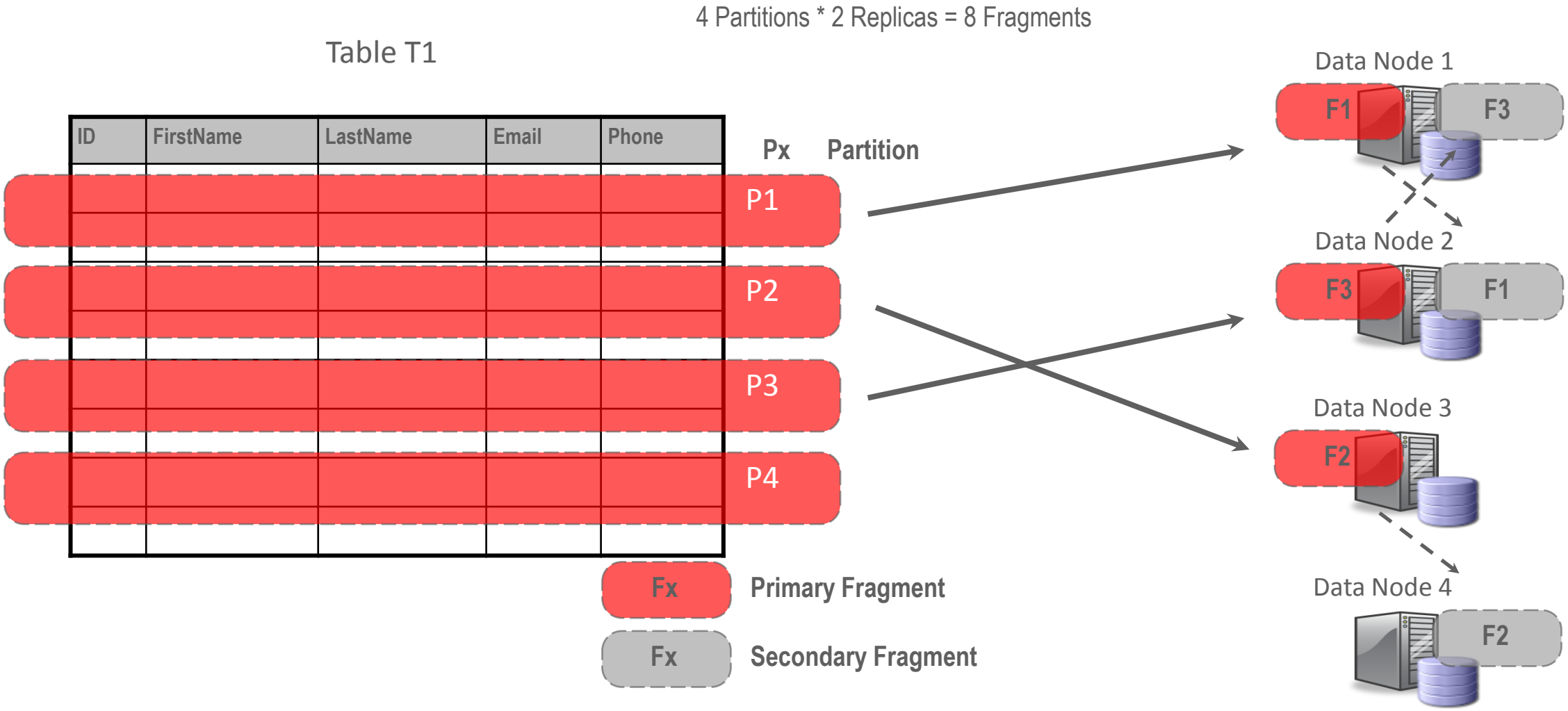
Automatic Data Partitioning



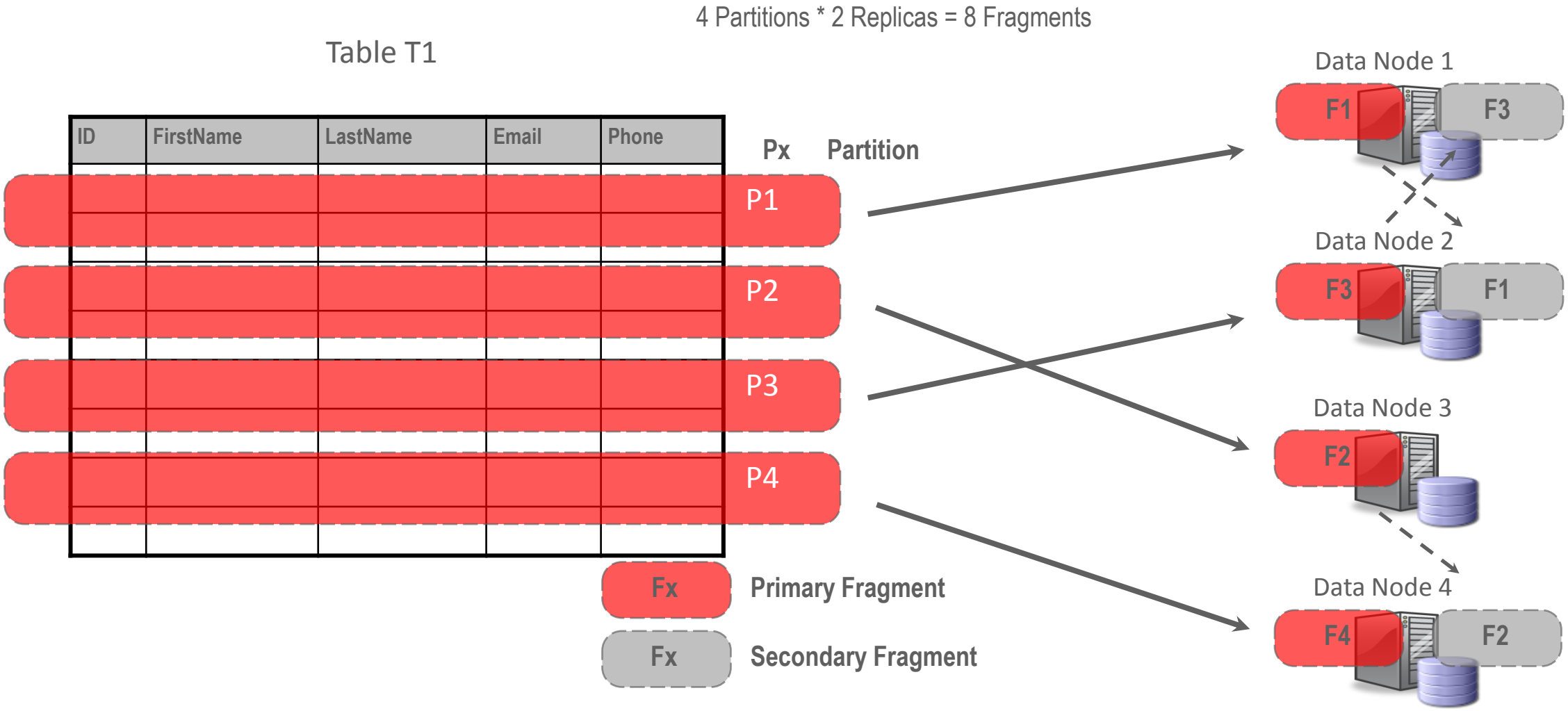
Automatic Data Partitioning



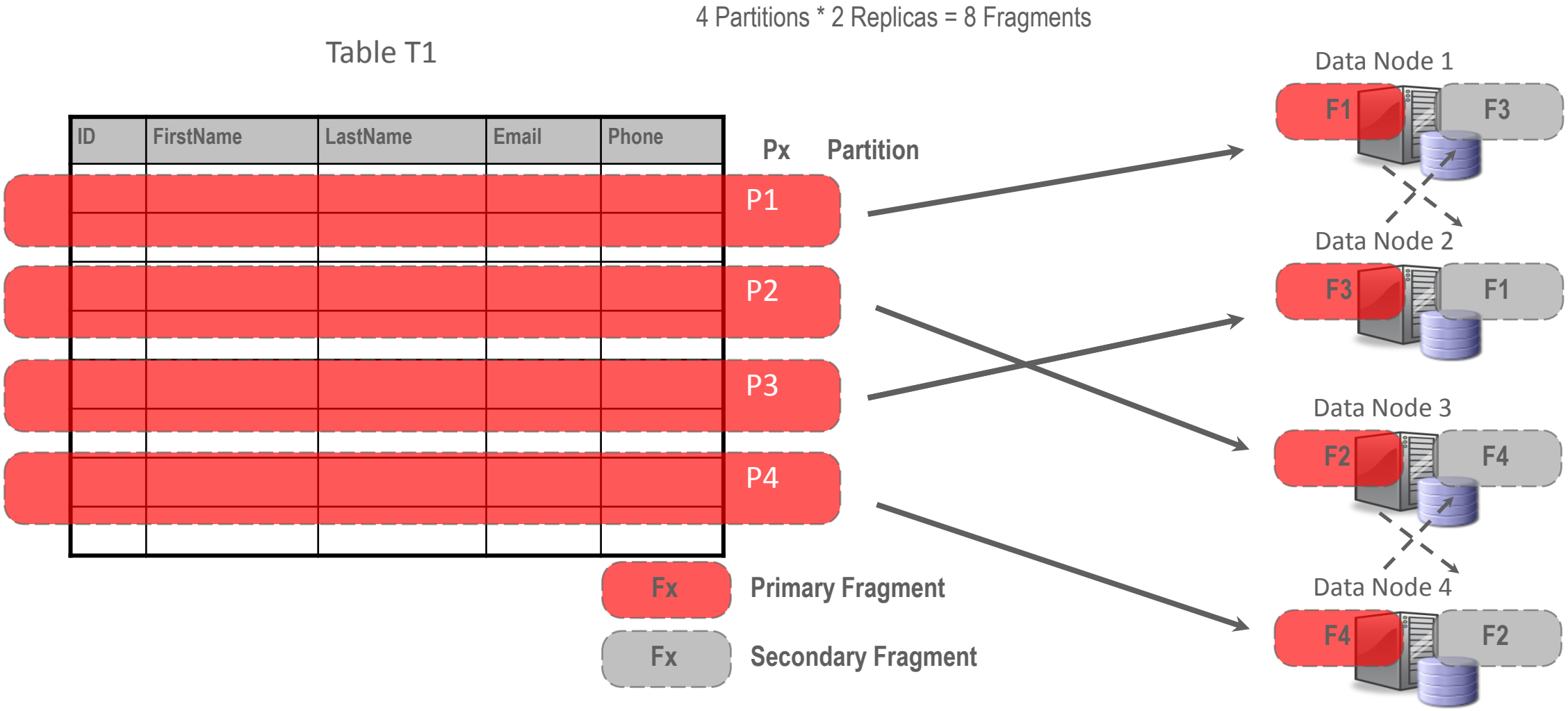
Automatic Data Partitioning



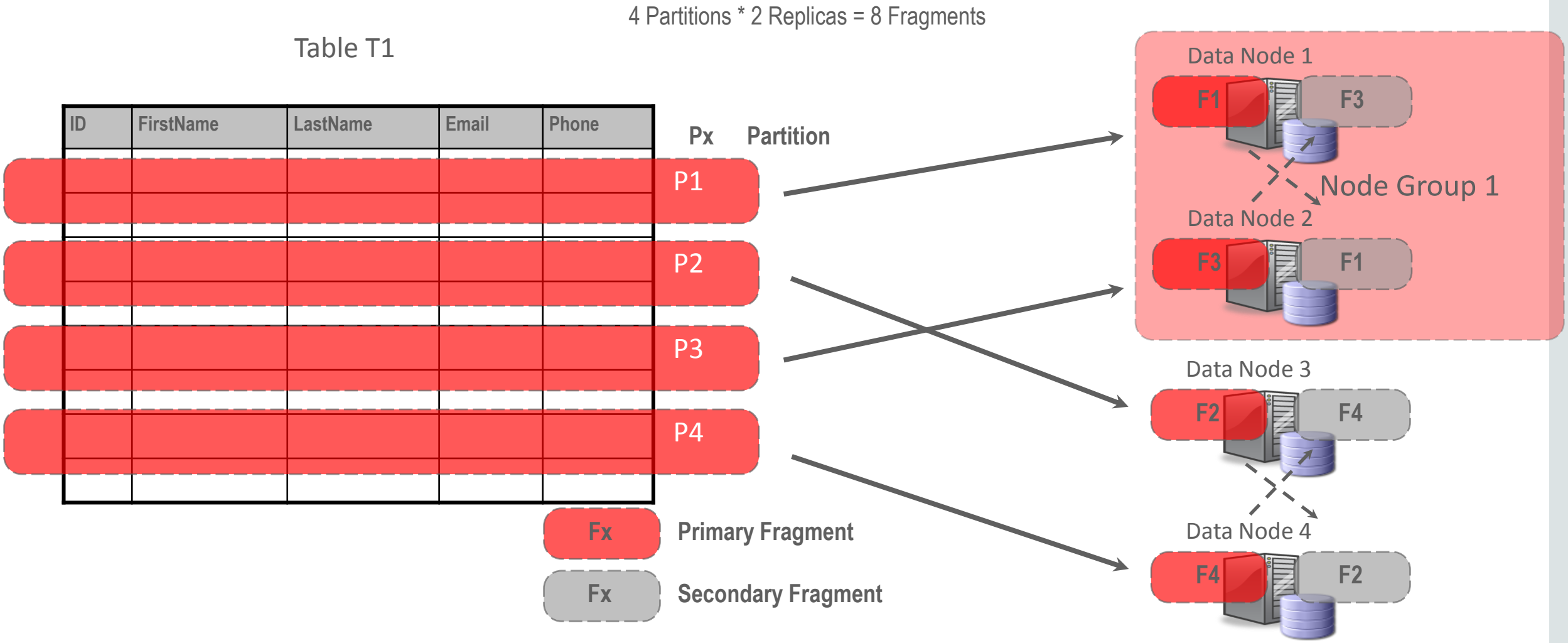
Automatic Data Partitioning



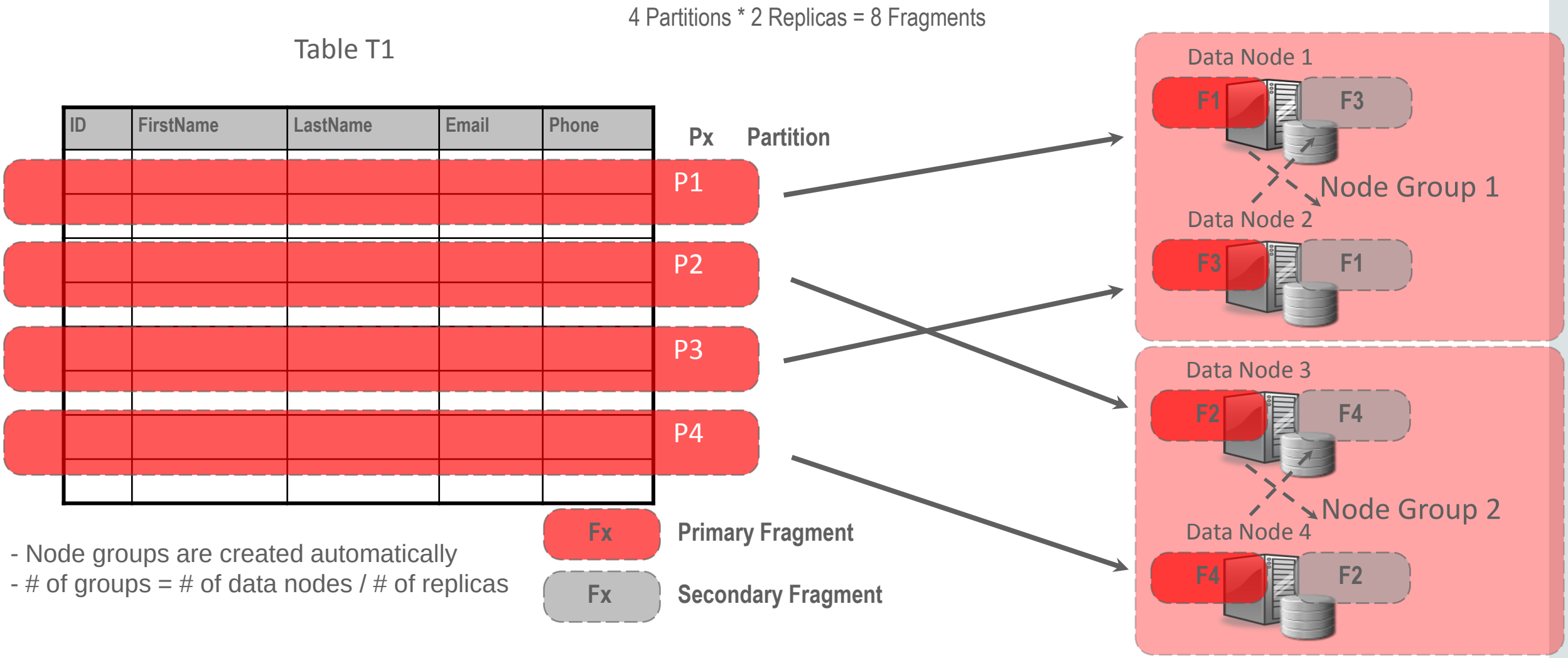
Automatic Data Partitioning



Automatic Data Partitioning

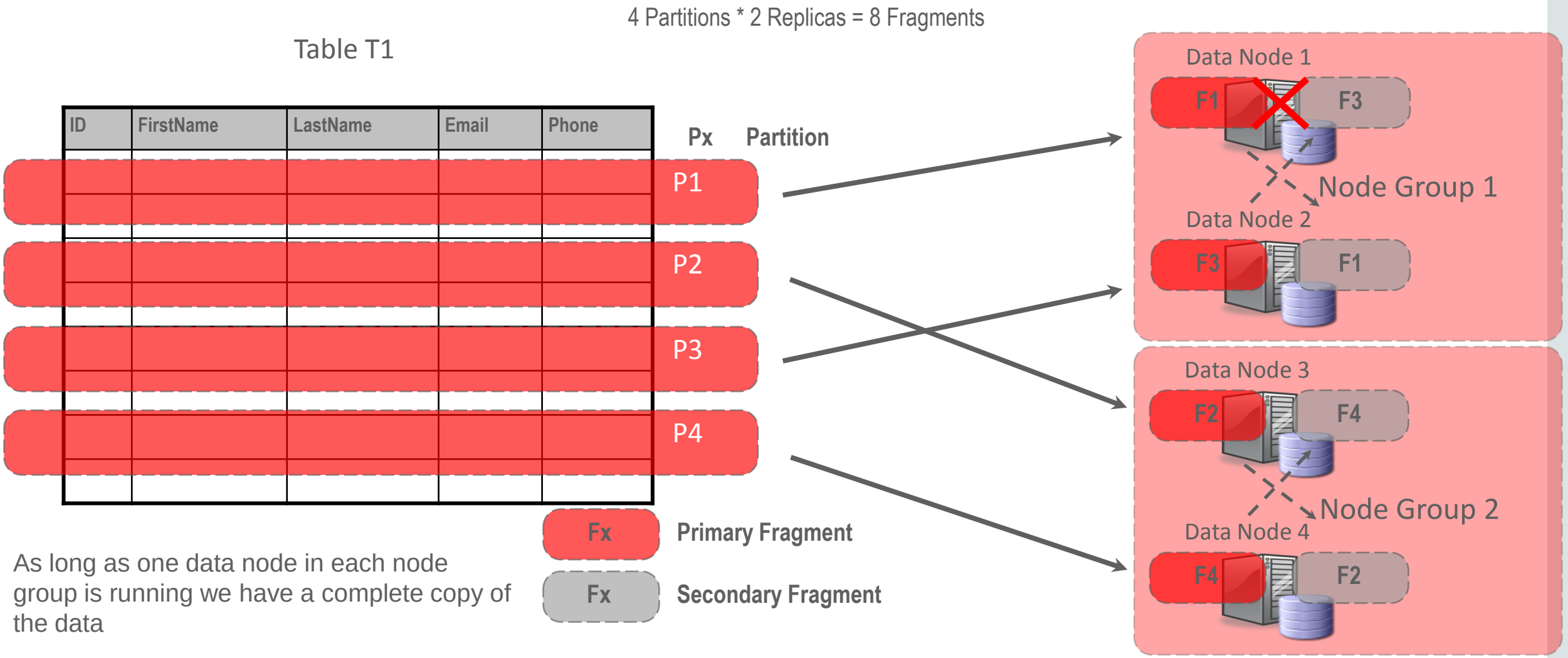


Automatic Data Partitioning



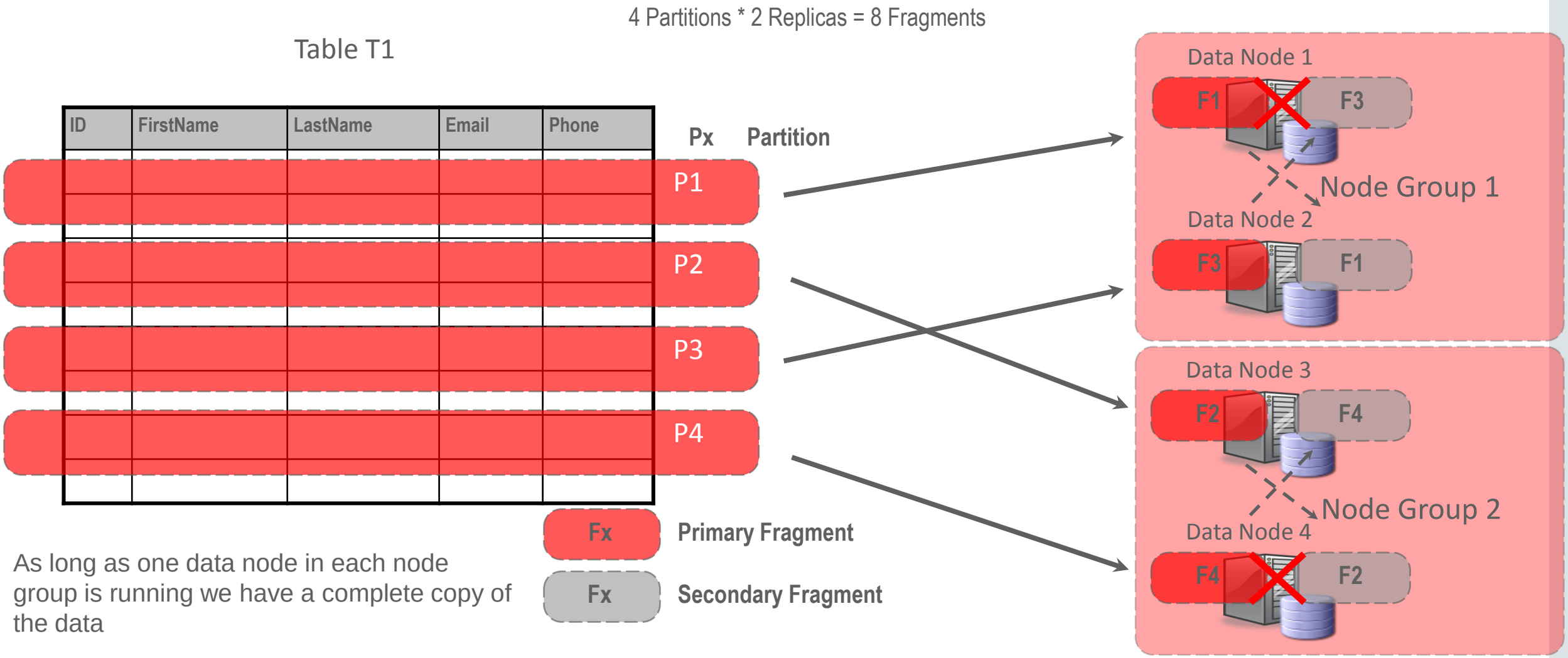
- Node groups are created automatically
- # of groups = # of data nodes / # of replicas

Automatic Data Partitioning



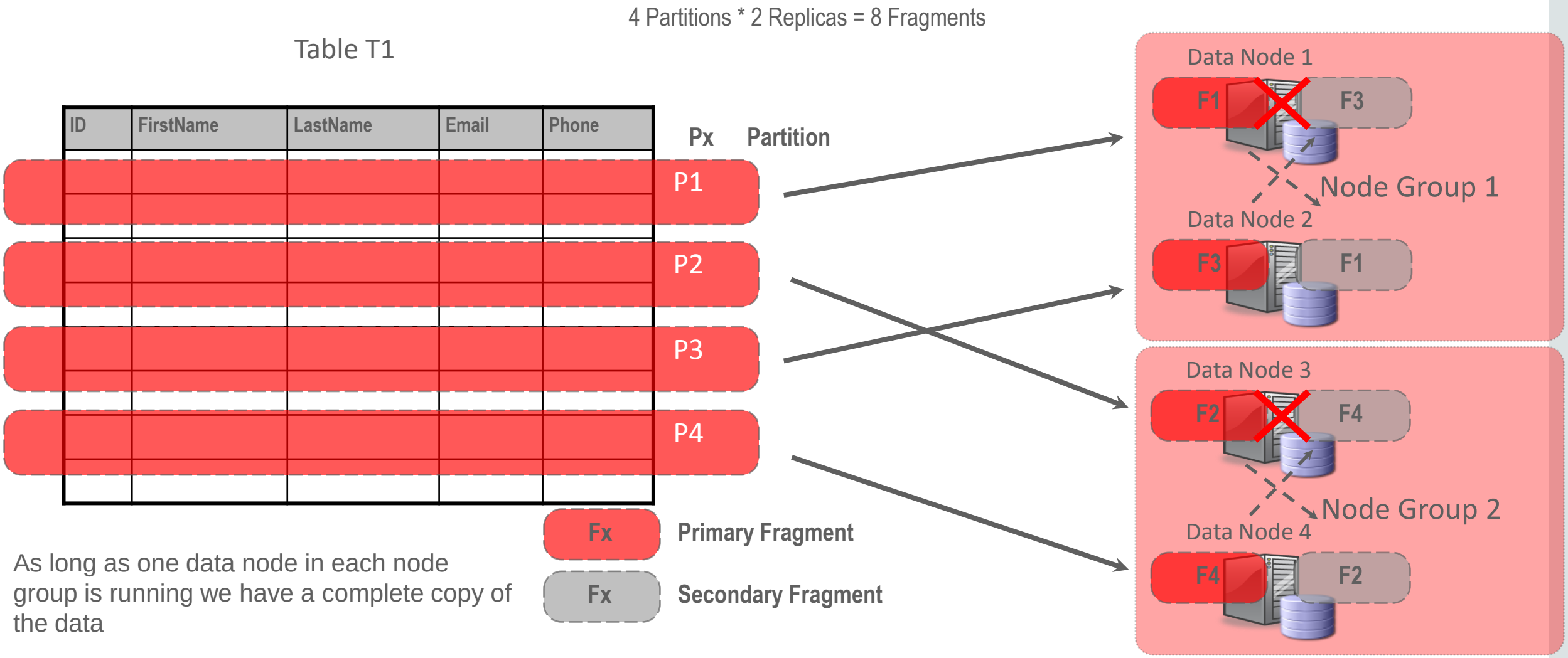
As long as one data node in each node group is running we have a complete copy of the data

Automatic Data Partitioning



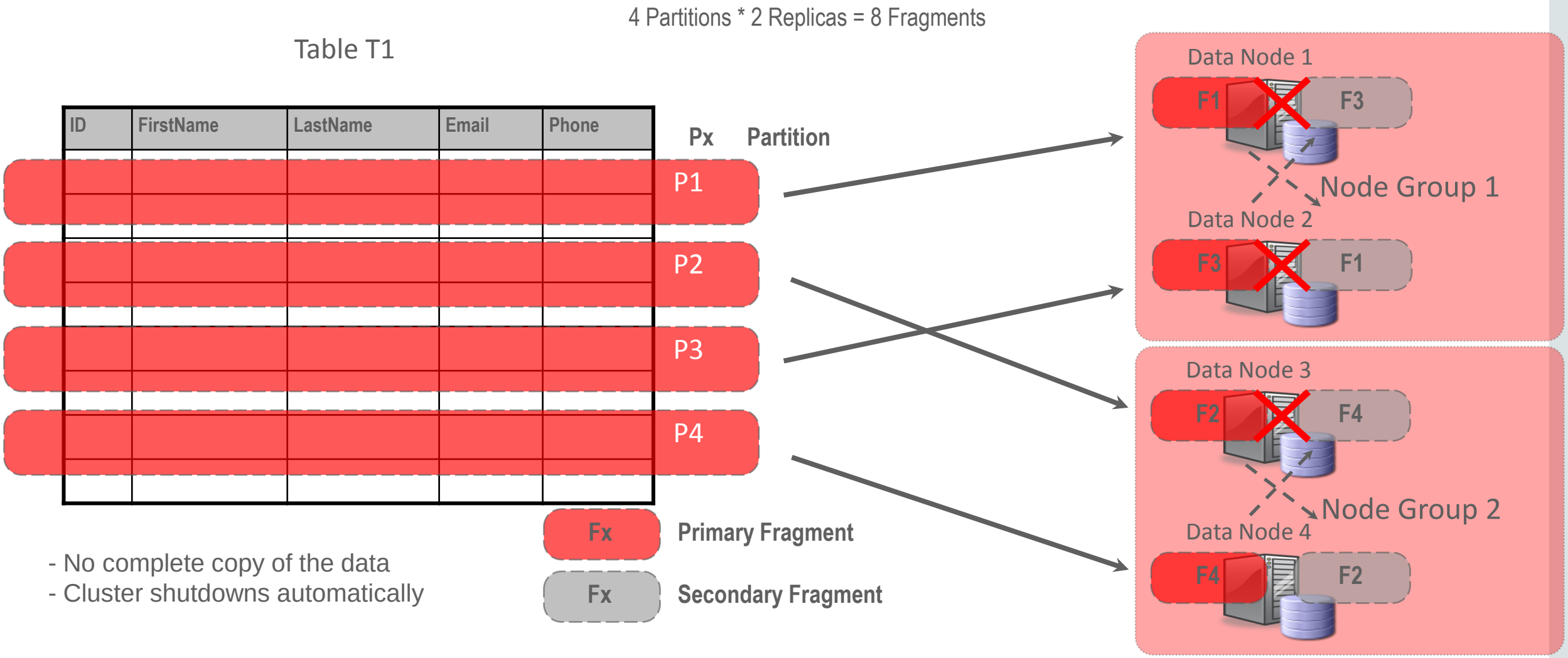
As long as one data node in each node group is running we have a complete copy of the data

Automatic Data Partitioning



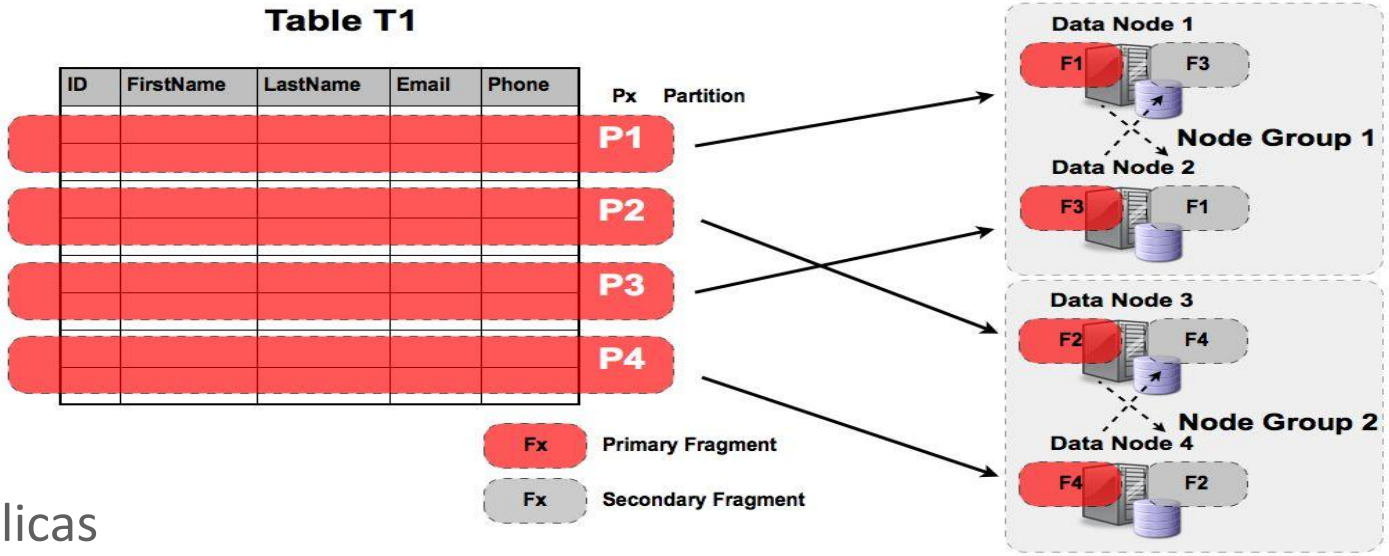
As long as one data node in each node group is running we have a complete copy of the data

Automatic Data Partitioning



Data Partitioning III

- Partition
  - Horizontal partitioning
  - A portion of a table, each partition contains a set of rows
  - Number of partitions == LQH
- Replica
  - A complete copy of the data
- Node Group
  - Created automatically
  - # of groups = # of data nodes / # of replicas
  - As long as there is one data node in each node group we have a complete copy of the data



Agenda

1. Introduction
2. Starting: configuration/installation/starting/stopping
3. The simplest way: Cluster Manager.

MySQL Cluster Installations

- Community / GPL
  - http://dev.mysql.com/downloads/cluster
- Carrier Grade Edition
  - http://edelivery.oracle.com
  - http://support.oracle.com

- Tarball
  - Manual “hands-on” install.
- RPM, DEB, PKG, DMG, MSI
  - O.S. specific install packages.
  - Specific paths, packages (server, client, etc.)
- Source code
  - Compile your own version.
    - `cmake .`
    - `make`
    - `make install`

MySQL Cluster Configuration

- All nodes are defined in config.ini
  - Devided into sections using “[…]”, sections for;
    - data nodes ndb{mt}d
    - Management nodes [ndb_mgmd]
    - MySQL servers [mysqld]
    - API nodes [ndbapi]
    - If keyword default is added to section values are applied to all
  - Management node is started with a config.ini. Should be same for all management nodes in system.
- MySQL API nodes are configued both in config.ini and have dedicated configuration file my.cnf

Configuration “config.ini”
Typical parameters

- `IndexMemory` and `dataMemory`.
- Set `MaxNoOfExecutionThreads` <= #cores
  - Otherwise contention will occur → unexpected behaviour.
- `RedoBuffer=32-64M`
  - If you need to set much it higher → your disks are probably too slow
- `FragmentLogFileSize=256M`
- `NoOfFragmentLogFiles`= 6 x DataMemory (in MB) / (4x 256MB)
  - Most common issue – customers never configure large enough redo log
- Data and index memory to cope with size of database/indexes
- `LockPagesInMainMemory=1`
- `MaxNoOf*`

Configuration
Disk-based tables

- Use Disk Data tables for
  - Simple accesses (read/write on PK)
  - Same for InnoDB – you can easily get IO BOUND (iostat)
- Set `DiskPageBufferMemory=3072M`
  - is a good start if you rely a lot on disk data – like the Innodb_Buffer_Pool, but set it as high as you can!
- Increased chance that a page will be cached
  - `SharedGlobalMemory=384M-1024M`
- `UNDO_BUFFER=64M` to `128M` (if you write a lot)
  - You cannot change this BUFFER later!
- Specified at LOGFILE GROUP creation time
  - `DiskIOThreadPool=[ 8 .. 16 ]`

MySQL Cluster Starting / Stopping
config.ini

```
[ndb_mgmd default]
ArbitrationRank                 =1
DataDir                         =/opt/mysql/735/mgmd_data

[ndb_mgmd]
hostname                        =khollman-es
NodeId                          =1

[ndbd default]
noofreplicas                    =2
DataDir                         =/opt/mysql/735/ndbd_data
DataMemory                      =20M
IndexMemory                     =10M
DiskPageBufferMemory            =4M
StringMemory                    =5
MaxNoOfConcurrentOperations     =2K
MaxNoOfConcurrentTransactions   =2K
SharedGlobalMemory              =500K
LongMessageBuffer               =512K
MaxParallelScansPerFragment     =16
MaxNoOfAttributes               =1000
MaxNoOfTables                   =20
MaxNoOfOrderedIndexes           =20
ODirect                         =TRUE
HeartbeatIntervalDbDb           =500
HeartbeatIntervalDbApi          =500
StopOnError                     =1
TransactionInactiveTimeout      =500
TransactionDeadlockDetectionTimeout = 1200
LockPagesInMainMemory           =2

[ndbd]
hostname                        =khollman-es
datadir                         =/opt/mysql/735/ndbd_data
nodeid                          =3

[mysqld default]

[mysqld]
NodeId                          =10

[mysqld]
NodeId                          =11

[NDBAPI]
NodeId                          =12

[NDBAPI]
NodeId                          =13
```

MySQL Cluster Starting / Stopping

my.cnf

```
[client]
socket                          =/tmp/mysql_7351.sock

[mysql]
prompt                          =\R:\m \d>\_
no-beep

[mysqld]
ndbcluster
datadir                         =/opt/mysql/735/data
ndb-connectstring               =khollman-es:1186
user                            =mysql
port                            =7351
socket                          =/tmp/mysql_7351.sock
general-log                     =1
log-output                      =FILE
log-error                       =khollman-es_7351.err
slow-query-log                  =1
max_connections                 =20
innodb_log_buffer_size          =8M
innodb_buffer_pool_size         =64M
innodb_log_file_size            =16M
innodb_flush_log_at_trx_commit  =2
innodb_file_per_table           =1
innodb_data_home_dir           =/opt/mysql/735/data
innodb_data_file_path
  =ibdata1:50M;ibdata2:50M:autoextend

[mysql_cluster]
ndb-connectstring               =khollman-es:1186
```

MySQL Cluster Starting / Stopping

- First time & when configuration changes are needed: **`--INITIAL`**
  - When the config.ini changes, no need to do a complete shutdown. Restart ndb_mgmd with `--INITIAL` to clean the cached config information. And then just restart the datanodes (without --initial)

- Starting

```
# ndb_mgmd -f config.ini --config-dir=/usr/local/mysql-cluster/conf --INITIAL
# ndbd --INITIAL  | ndbmtd -c localhost:1186 --INITIAL  |  ndbd –n (nostart)
# scripts/mysql_install_db --defaults-file=my.cnf --user=mysql
# mysqld_safe --defaults-file=my.cnf --user=mysql
```

MySQL Cluster Starting / Stopping

- Starting
  - `ndb_mgmd -f config.ini --config-dir=/usr/local/mysql-cluster/conf`
  - `ndbd | ndbmtd -c localhost:1186`
  - `mysqld_safe --defaults-file=my.cnf`
- Stopping
  - `ndb_mgm -e shutdown`
  - `ndb_mgm -e` [mgmt node & datanode] `1 | 3 | 4 stop`
  - `ndb_mgm -e` [mgmt node & datanode] `1 | 3 | 4 restart`
  - `mysqladmin --defaults-file=my.cnf -uroot shutdown`

MySQL Cluster Starting / Stopping
nostart

```
ndbmtd --ndb-nodeid=3 -n

ndbmtd --ndb-nodeid=4 -n

ndb_mgm -e show
Cluster Configuration
---------------------
[ndbd(NDB)]   2 node(s)
id=3   @127.0.0.1  (mysql-5.6.17 ndb-7.3.5, not started)
id=4   @127.0.0.1  (mysql-5.6.17 ndb-7.3.5, not started)

ndb_mgm -e "all start"

ndb_mgm -e show
Cluster Configuration
---------------------
[ndbd(NDB)]   2 node(s)
id=3   @127.0.0.1  (mysql-5.6.17 ndb-7.3.5, starting, Nodegroup: 0, *)
id=4   @127.0.0.1  (mysql-5.6.17 ndb-7.3.5, starting, Nodegroup: 0)
```

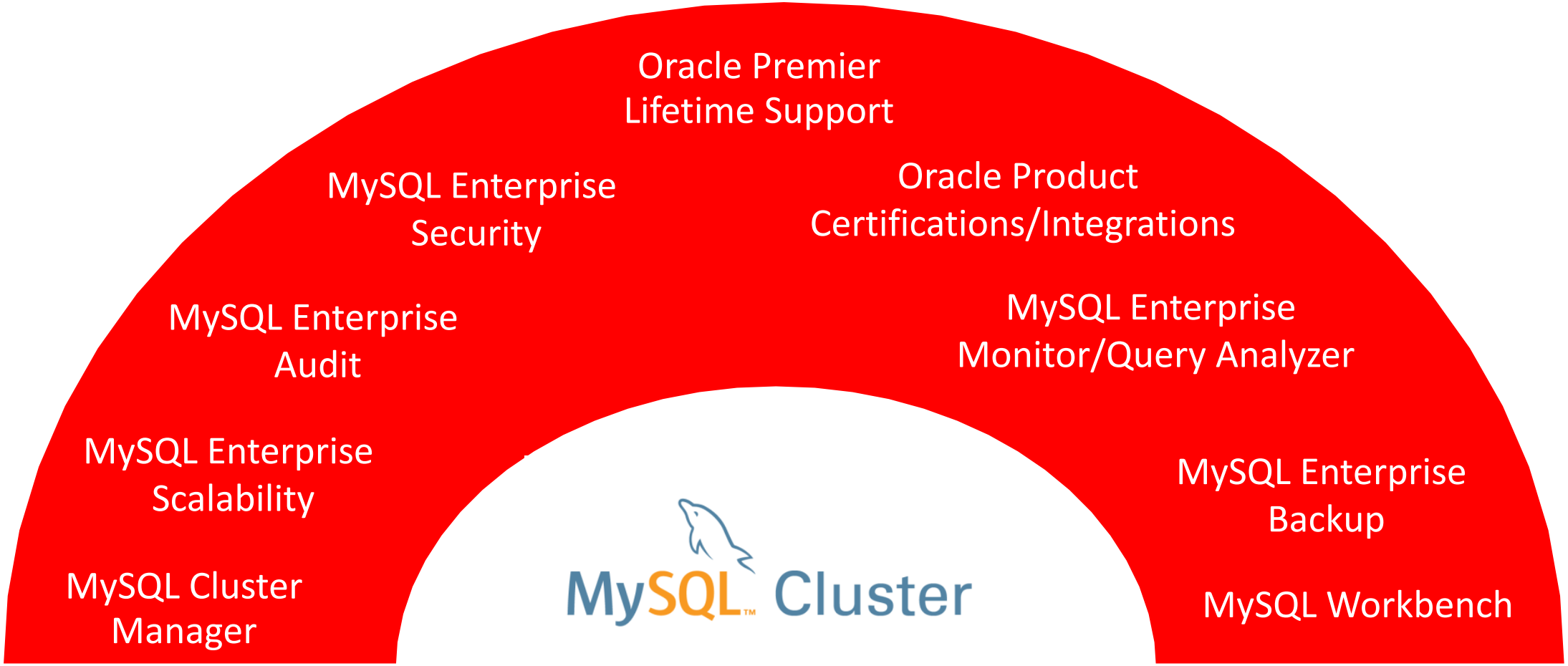
Agenda

1. Introduction
2. Starting: configuration/installation/starting/stopping
3. The simplest way: Cluster Manager.

The simplest way: Cluster Manager

MySQL Cluster CGE

Highest Levels of Performance, Security & Availability



ORACLE

Enhancing DevOps Agility, Reducing Downtime

MySQL Cluster Manager

Automated Management

- Start / Stop node or whole cluster
- On-Line Scaling
- On-Line Reconfiguration
- On-Line Upgrades
- On-Line Backup & Restore
- Import Running Cluster

Self-Healing

- Node monitoring
- Auto-recovery extended to SQL + mgmt nodes

HA Operations

- Cluster-wide configuration consistency
- Persistent configurations
- HA Agents

ORACLE

MySQL Cluster Manager: Creating a Cluster I

- Download MySQL Cluster Manager (MCM) + MySQL Cluster:
  - MCM 1.3.4, choose the package*without* cluster, that should be about 20Mb (download from https://edelivery.oracle.com/)
  - MySQL Cluster 7.4.6 (download from http://downloads.mysql.com/archives/get/file/ mysql-cluster-gpl-7.4.6-linux-glibc2.5-x86_64.tar.gz)

- Install and Configure MySQL Cluster Manager
  - The MCM and Cluster binaries have to be installed in all the servers where any NDB process is running (ndbd, mysqld and ndb_mgmd).
  - `mkdir MCM`
  - `cd MCM`
  - `tar xzf /ruta/a/ mcm-1.3.4-linux-glibc2.5-x86-64bit.tar.gz`
  - `mv mcm-1.3.4-linux-glibc2.5-x86-64bit/mcm1.3.4 .`
  - `rmdir mcm-1.3.4-linux-glibc2.5-x86-64bit.tar.gz`
  - `cp mcm1.3.4/etc/mcmd.ini .`
  - (need to change: `manager-directory = /home/<user>/MCM/mcm_data`)

MySQL Cluster Manager: Creating a Cluster II

- Install Cluster binaries
  - `tar xzf /ruta/a/ mysql-cluster-gpl-7.4.6-linux-glibc2.5-x86_64.tar.gz`
  - `mv mysql-cluster-gpl-7.4.6-linux-glibc2.5-x86_64 cluster-746`
- Start the MCMD daemon (as o.s. user 'mysql')
  - `./mcm1.3.4/bin/mcmd --defaults-file=./mcmd.ini –daemon`
- Configuring the Cluster
  - Connect to MCM and configure cluster (only from one site):
  - Start mcm client: `./mcm1.3.4/bin/mcm`
  - (you will need the mysql CLI executable in the PATH variable, ej. `/usr/local/mysql/bin/mysql` as the mcm CLI is based around this executable)

MySQL Cluster Manager: Creating a Cluster III

```
mcm> list commands
mcm> create site --hosts=127.0.0.1 mysite;
mcm> list sites;
mcm> list hosts mysite;
mcm> add package --basedir=/home/<user>/MCM/cluster-746
  cluster746;
mcm> list packages mysite;
mcm> create cluster --package=cluster746 --
  processhosts=ndb_mgmd@127.0.0.1,ndbmtd@127.0.0.1,ndbmtd@1
  27.0.0.1 mycluster;
mcm> add process --
  processhosts=mysqld@127.0.0.1,mysqld@127.0.0.1 mycluster;
mcm> add process --
  processhosts=ndbapi@127.0.0.1,ndbapi@127.0.0.1 mycluster;
mcm> add process --
  processhosts=ndbapi@127.0.0.1,ndbapi@127.0.0.1 mycluster;
mcm> get -d port:mysqld mycluster;
mcm> set port:mysqld:51=3307 mycluster;
```

- Start the Cluster:

```
mcm> show status -r mycluster;
mcm> start cluster mycluster;
mcm> show status -r mycluster;
```

Hot Database Migration
Requirements

- Use MySQL Cluster Replication Slave.
  - Slave's can have a superior version than Master.
- Rolling Upgrade. (`--initial` required)

- If you want to use MySQL Cluster Manager:

```
mcm> add package --basedir=/opt/MCM_LAB/cluster-736 cluster736;
mcm> upgrade cluster --package=cluster736 mycluster;
```

How Does MySQL Cluster Manager Help?
Initiating upgrade from MySQL Cluster 7.0 to 7.3

MySQL Cluster Manager

Before MySQL Cluster Manager

- 1 x preliminary check of cluster state
- 8 x ssh commands per server
- 8 x per-process stop commands
- 4 x scp of configuration files (2 x mgmd & 2 x mysqld)
- 8 x per-process start commands
- 8 x checks for started and re-joined processes
- 8 x process completion verifications
- 1 x verify completion of the whole cluster.
- Excludes manual editing of each configuration file.

Total: 46 commands -
2.5 hours of attended operation

With MySQL Cluster Manager

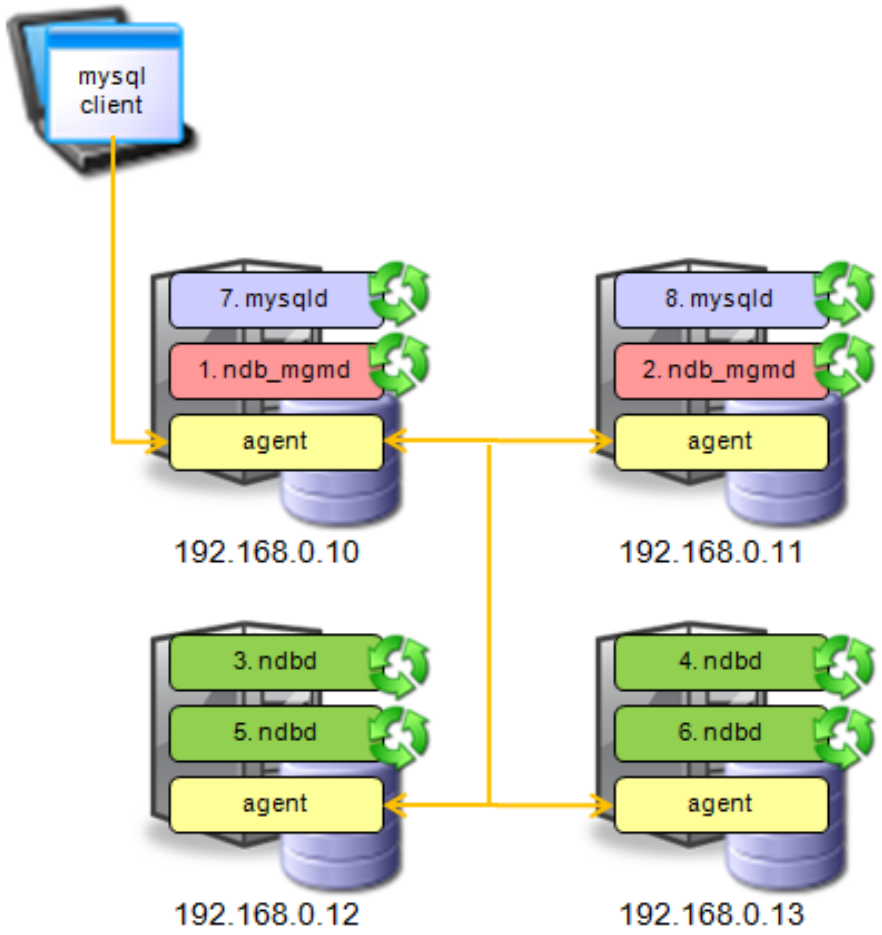
upgrade cluster --package=7.3 mycluster;

Total: 1 Command -
Unattended Operation

ORACLE

MCM: Upgrade Cluster

MySQL Cluster Manager




```
mcm> upgrade cluster --package=cluster736
mycluster;
```

ORACLE

MySQL Cluster – Users & Applications

Telecoms

- Subscriber Databases (HLR / HSS)
- Service Delivery Platforms
- VAS: VoIP, IPTV & VoD
- Mobile Content Delivery
- Mobile Payments
- LTE Access

Web & Enterprise

- High volume OLTP
- eCommerce
- User Profile Management
- Session Management & Caching
- Content Management
- On-Line Gaming

http://www.mysql.com/customers/cluster/

Alcatel·Lucent

COMPANY OVERVIEW

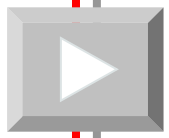
- Leading provider of communications platforms, solutions & services
- €15.2bn Revenues (2009), 77k employees across 130 countries

CHALLENGES / OPPORTUNITIES

- Converged services driving migration to next generation HLR / HSS systems
- New IMS platforms for Unified Communications
- Reduce cost per subscriber and accelerate time to value

SOLUTIONS

- MySQL Cluster Carrier Grade Edition
- MySQL Support & Consulting Services



CUSTOMER PERSPECTIVE

"*MySQL Cluster won the performance test hands-down, and it fitted our needs perfectly. We evaluated shared-disk clustered databases, but the cost would have been at least 10x more.*"

-- François Leygues, Systems Manager

RESULTS

- Scale out on standard ATCA hardware to support 60m+ subscribers on a single platform
- Low latency, high throughput with 99.999%+ availability
- Enabled customers to reduce cost per subscriber and improve margins
- Delivered data management solution at 10x less cost than alternatives

ORACLE

COMPANY OVERVIEW

- Leading telecoms provider across Europe and Asia. Largest Nordic provider
- 184m subscribers (Q2, 2010)

CHALLENGES / OPPORTUNITIES

- Extend OSS & BSS platforms for new mobile services and evolution to LTE
- OSS: IP Management & AAA
- BSS: Subscriber Data Management & Customer Support

SOLUTIONS

- MySQL Cluster
- MySQL Support Services

CUSTOMER PERSPECTIVE

"Telenor has been using MySQL for fixed IP management since 2003 and are extremely satisfied with its speed, availability and flexibility. Now we also support mobile and LTE IP management with our solution. Telenor has found MySQL Cluster to be the best performing database in the world for our applications."

- Peter Eriksson, Manager, Network Provisioning

RESULTS

- Launch new services with no downtime, due to on-line operations of MySQL Cluster
- Consolidated database supports Subscriber Data Management initiatives
- MySQL Cluster selected due to 99.999% availability, real time performance and linear scalability on commodity hardware

ORACLE

Questions?

ORACLE

Next Steps

MySQL Cluster

Learn More

- www.mysql.com/cluster
- MySQL Curriculum: http://oracle.com/education/mysql

Try it Out

- dev.mysql.com/downloads/cluster/

Let us know what you think

- forums.mysql.com/list.php?25
- ted.wennmark@oracle.com
- keith.hollman@oracle.com

ORACLE


ORACLE®