Michael Köster - IT Specialist
November 2010

What's new in Informix 11.70?

Featues & Functions

- **90+ new Features & Function in Informix 11.70**
- Split in:
  - Installation
  - Migration
  - Administration
  - Application Development
  - Embeddability
  - OAT
  - Enterprise Replication
  - High-availability
  - Performance
  - Security
  - Warehousing
- Information found in **Informix Information Center 11.70**

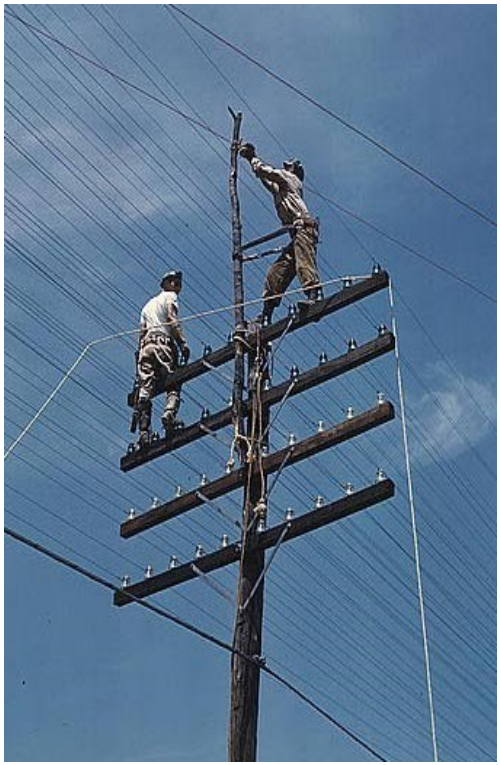
  http://publib.boulder.ibm.com/infocenter/idshelp/v117/index.jsp

  Section Product Overview – Release Information - Informix

Agenda

- **Informix Grid Replication**
- Rolling Upgrades
- Transactional Survival
- DDL Support on all nodes
- Security
  - Mapped User
  - Trusted Context
  - Selective Row Auditing
- Data Warehouse Features

Flexible Grid



- Grid is an extension to Enterprise Replication that allows ER to replicate the execution of statements
- DML/DDL statements, procedures, and functions can be replicated to, then run on grid target servers
- Split in Grid Administration / Execution
- Filter / Conflict Resolution
  - Grouping / Tagging Commands
  - Redoing commands
  - Server State and Response
  - Forcing commands
- Cleanup commands

Agenda

- Informix Grid Replication
- **Rolling Upgrades**
- Transactional Survival
- DDL Support on all nodes
- Security
  - Mapped User
  - Trusted Context
  - Selective Row Auditing
- Data Warehouse Features

Rolling Upgrade

- Ability to convert HDR or RSS pair to ER and setup replicate for every table in the system automatically
  - Uses ERKEY on tables with no primary key
  - *Product upgrade without downtime*
- check & start conversion

```
$ cdr check sec2er -c ol_er1 --print ol_er2
WARNING:  CDR_SERIAL value on ol_er1 can cause collisions.
ERROR:  Server ol_er1 has no group.
ERROR:  Server ol_er1 has no group.
WARNING:  Using the same values for CDR_SERIAL can cause collisions.
FATAL:  SQLHOSTS is not set up correctly for ER.
ERROR:  SQLHOSTS is not set up correctly for ER.
Secondary conversion to ER is not possible.

Errors:0003   Warnings:0002
command failed -- Error while processing cdr sec2er command - see output (225)
Can not switch to syscdr sqlcode:0 isamcode:0
```

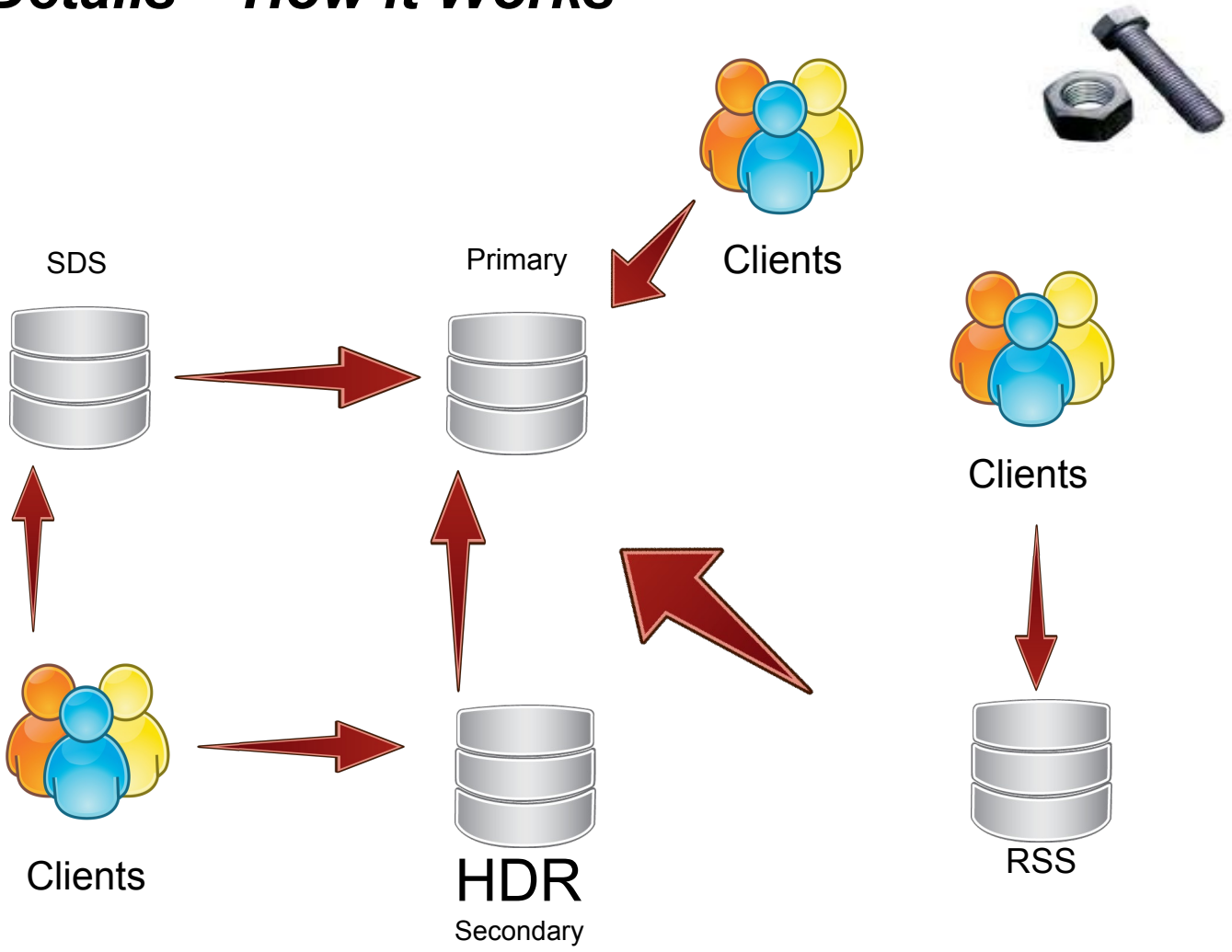
Process for peforming a rolling upgrade

1. Run cdr check sec2er
   - Check if split is possible
2. Run cdr start sec2er
   - Convert HDR/RSS pair into an ER pair
3. Upgrade the secondary to new version
   - ER supports replication between different versions so upgraded server would still be able to replicate with old primary
4. Move applications to old secondary node
5. Reinstantiate old primary using ifxclone
6. Move applications back to old primary

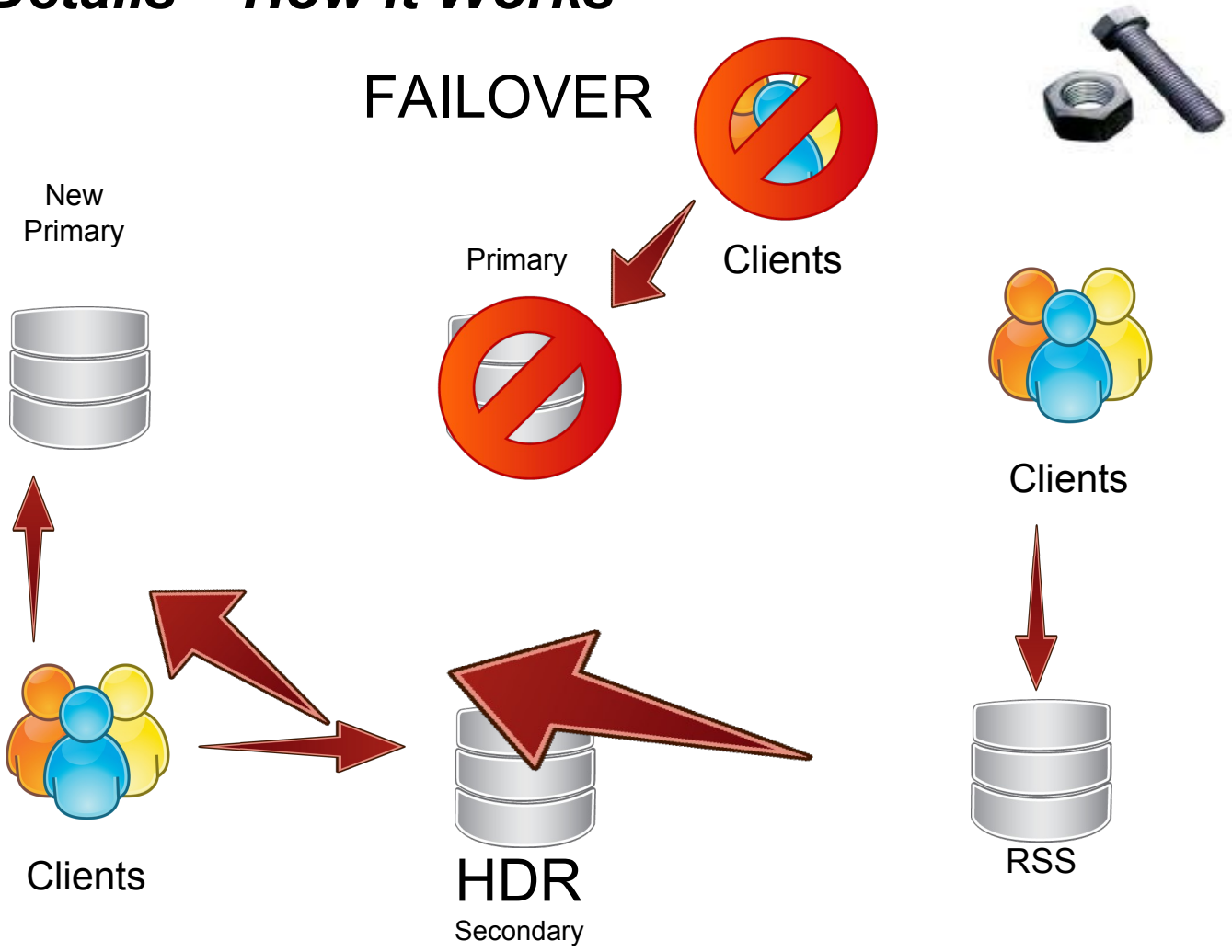
Agenda

- Informix Grid Replication
- Rolling Upgrades
- **Transactional Survival**
- DDL Support on all nodes
- Security
  - Mapped User
  - Trusted Context
  - Selective Row Auditing
- Data Warehouse Features

Feature Details – How it Works



Feature Details – How it Works



Feature Summary – Quick Reference

- onconfig parameter: FAILOVER_TX_TIMEOUT
- Recommendations:
  - Same value for FAILOVER_TX_TIMEOUT across cluster.
  - Order of failover node choices: SDS, HDR, then RSS.
  - If using HDR, setting DRINTERVAL to -1 for synchronous log buffer flushing.

- **What transaction survival won't do**
- Transactions of clients connected to original primary do not survive
- Does not allow you to restart the original primary and have transactions from the existing secondaries resume work

Agenda

- Informix Grid Replication
- Rolling Upgrades
- Transactional Survival
- **DDL Support on all nodes**
- Security
  - Mapped User
  - Trusted Context
  - Selective Row Auditing
- Data Warehouse Features

Feature Summary – Quick Reference

- This feature allows sessions that are connected to any type of updatable secondary to perform DDL SQL statements.
- Still DDL commands that are not work, e.g. create database (with no logging), Create/Drop xadatasource etc.

- If UPDATABLE_SECONDARY is not set to a value > 0 in the onconfig file the following error codes are returned:
  - SQL error code: -26097 Operation is not valid on a secondary server.
  - ISAM error code: 140: ISAM error: operation illegal on a DR Secondary.

Agenda

- Informix Grid Replication
- Rolling Upgrades
- Transactional Survival
- DDL Support on all nodes
- **Security**
  - **Mapped User**
  - Trusted Context
  - Selective Row Auditing
- Data Warehouse Features

Feature Summary – Quick Reference

- Authenticate external users without an OS account
  - Database users no longer need
    - An account Look-up in the local OS
    - Enable password properties for external authenticators
- The DBSA “maps” an external user to:
    - An existing OS user or
    - A database defined UIG/GID pair
- Uses an extension to the GRANT/REVOKE SQL statement
- Requires PAM or SSO authentication
- Traditional password based authentication is still available

Examples:

```
GRANT ACCESS TO user1 PROPERTIES USER ravik;
GRANT ACCESS TO user2 PROPERTIES UID 100, GROUP (200);
GRANT ACCESS TO user3 PROPERTIES USER ravik, HOME '/home/user4”;
GRANT ACCESS TO user4 PROPERTIES USER ravik AUTHORIZATION (dbsa);
GRANT ACCESS TO PUBLIC PROPERTIES USER ravik;
```

System Tables

- New system tables in 'sysuser' database
  - SYSUSERMAP
  - SYSSURROGATES
  - SYSSURROGATEGROUPS
- DBSA should use the GRANT ACCESS TO / REVOKE ACCESS FROM statements to manage the system tables as there are cross-references !

Agenda

- Informix Grid Replication
- Rolling Upgrades
- Transactional Survival
- DDL Support on all nodes
- **Security**
  - Mapped User
  - **Trusted Context**
  - Selective Row Auditing
- Data Warehouse Features

Feature Summary – Quick Reference

- This feature allow to reuse a connection for a different user without the need to establish a new connection

- Why?
  - In a 3-tiers architecture, the middle-tier's user must have all the privileges needed to execute all the requests from all users
  - There is a security issue of accessing resources on behalf of users if the middle-tier's user is compromised
  - There is a lost of information when auditing needs to distinguish end-users from middle-tier's user
  - Establishing new connection may suffer performance drop which is also the case when using connection pool

Feature Details – Trusted Context Object

- A Trusted Context is a database object that defines a set of properties for a connection that when met, allow that connection to be a "**trusted connection**" with special properties.
  - The connection must be established by a specific user.
  - The connection must come from a trusted client machine.
  - The port over which the connection is made must have the required encryption.
- If these criteria are met, the connection will allow changes in userid and privileges as defined in the trusted context.

Step 1: Create Trusted Context Object
Step 2: Establish Trusted Connections
Step 3: Switch Connections

Feature Details – Trusted Context Creation

- New SQL statement to create a trusted context

```
CREATE TRUSTED CONTEXT mytcx
BASED UPON CONNECTION USING SYSTEM AUTHID yob
DEFAULT ROLE employee
ATTRIBUTES (ADDRESS 'linx.swglab.fr.ibm.com')
WITH USE FOR PUBLIC WITHOUT AUTHENTICATION ENABLE
```

- Switch over a trusted connection

EXEC SQL SET SESSION AUTHORIZATION to “joe”

  - Audit records will show the switched user as the originator of the operations
  - Need to commit or rollback before switching to a new user when using transactions

Agenda

- Informix Grid Replication
- Rolling Upgrades
- Transactional Survival
- DDL Support on all nodes
- **Security**
  - Mapped User
  - Trusted Context
  - **Selective Row Auditing**
- Data Warehouse Features

Feature Details – SRLA Setup

- There is a new parameter called ADTROWS in adtcfg file
  - 0: old behavior i.e. no changes in row level auditing (default)
  - 1: SRLA is enabled and only "audit" enabled tables will generate row-level audit records.
  - 2: SRLA + include integer-primary key in the audit records
- The feature can be turned on dynamically
- New SQL statements

  CREATE TABLE {existing syntax} | with AUDIT
  ALTER TABLE {existing syntax} [add | drop] AUDIT
- Anyone with RESOURCE or DBA permission can either
- Only a DBSSO can « DROP AUDIT » on a table.

Agenda

- Informix Grid Replication
- Rolling Upgrades
- Transactional Survival
- DDL Support on all nodes
- Security
  - Mapped User
  - Trusted Context
  - Selective Row Auditing
- **Data Warehouse Features**

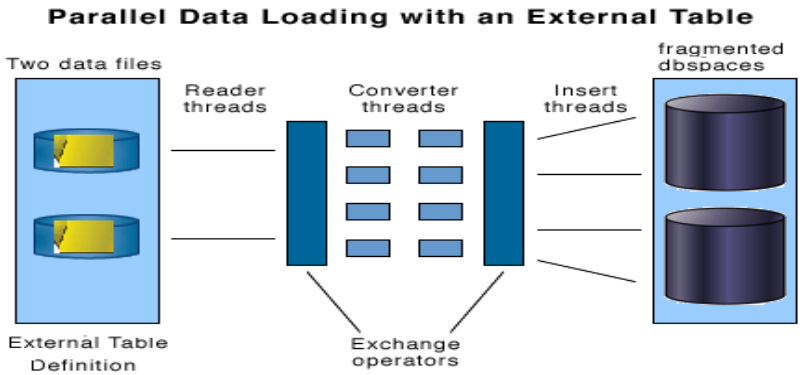
Data Warehouse Features

- **External Tables**
- Intervall Fragmentation
- Fragment Level Statistics
- Storage Provisioning
- Multi Index Scans
- Pushdown Hash Join

External Tables

Performance

- Faster load and unload of large dataset.
- Internal tests shows up to 2x for unload and up to 3x for load of large tables over existing utilities



Ease of Use

- External table can be used in an SQL statement in place of a regular table.
- External table can be used in Stored procedure for load and unload
- No need of DBA privilege to do Load/Unload using external tables

Example #1: Creation of an External Table

```
CREATE EXTERNAL TABLE empdata
( empname char(40), empdoj date)
USING (DATAFILES ("DISK:/work/empdata.unl“), FORMAT "DELIMITED",
REJECTFILE "/work/errlog/empdata.rej", MAXERRORS 100, DELUXE);
```

Example #2: Creation of an External Table using SAMEAS

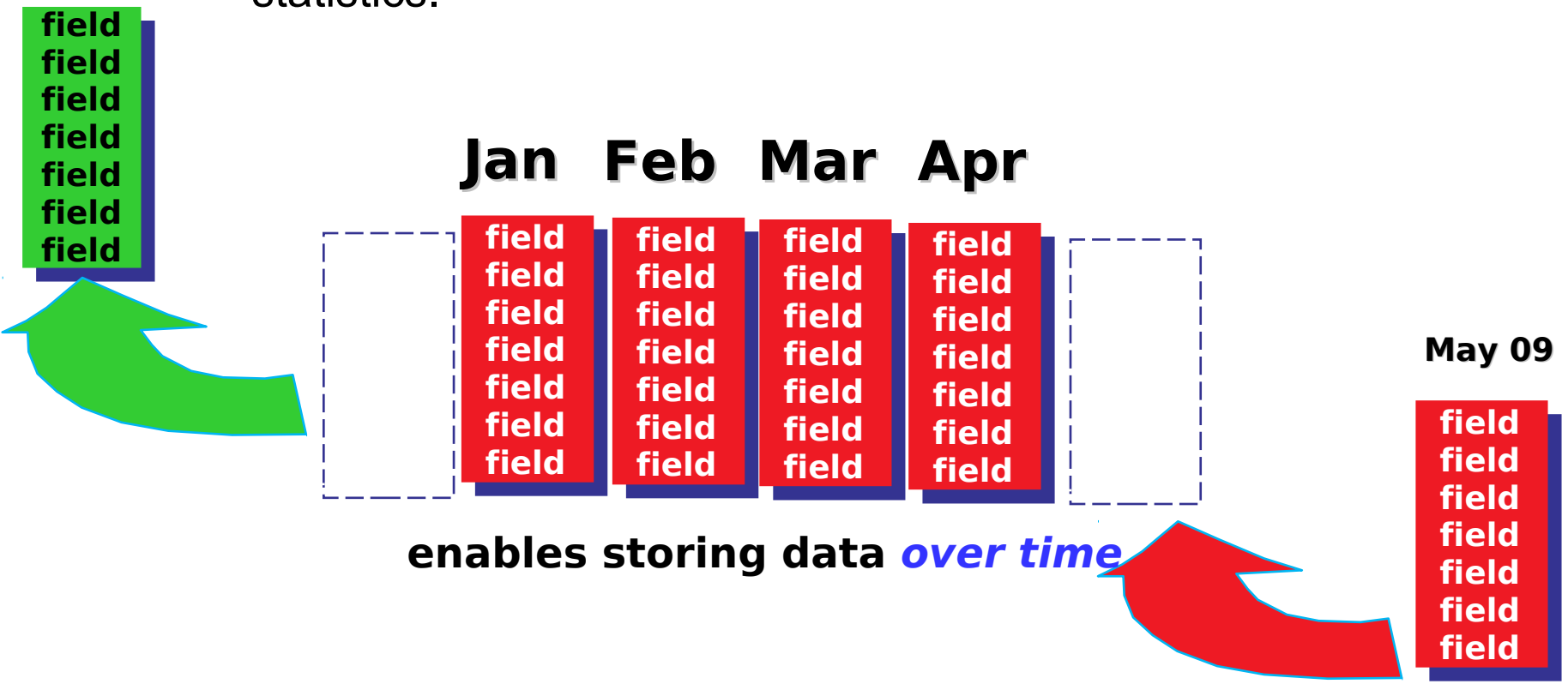
```
CREATE EXTERNAL TABLE emp_ext SAMEAS empdata
USING (DATAFILES ("DISK:/work/empdata2.unl"), REJECTFILE"/work/errlog/empdata2.rej“, DELUXE);
```

Data Warehouse Features

- External Tables
- **Intervall Fragmentation**
- Fragment Level Statistics
- Storage Provisioning
- Multi Index Scans
- Pushdown Hash Join

Interval Fragmentation

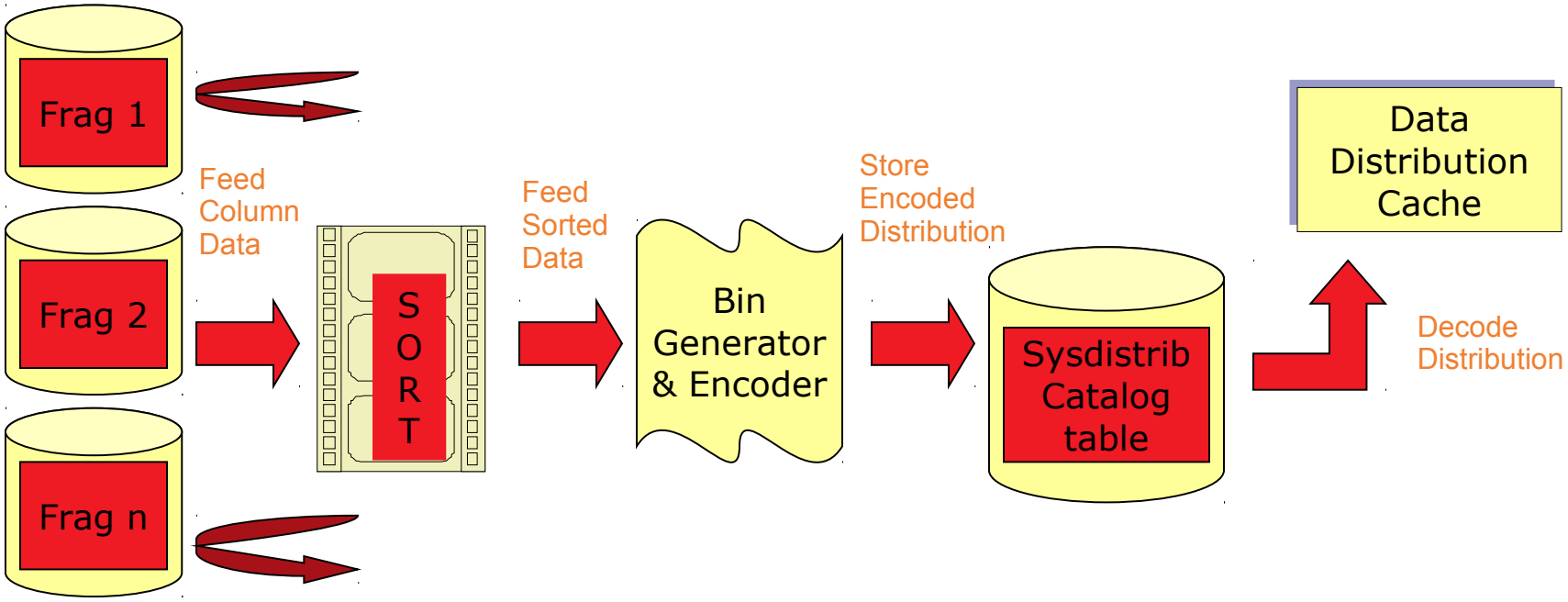
- Time-cyclic data management (roll-on, roll-off)
- Attach and detach online without requiring exclusive lock and access to the table
- Automatically kicks off background process to recollect statistics.



Data Warehouse Features

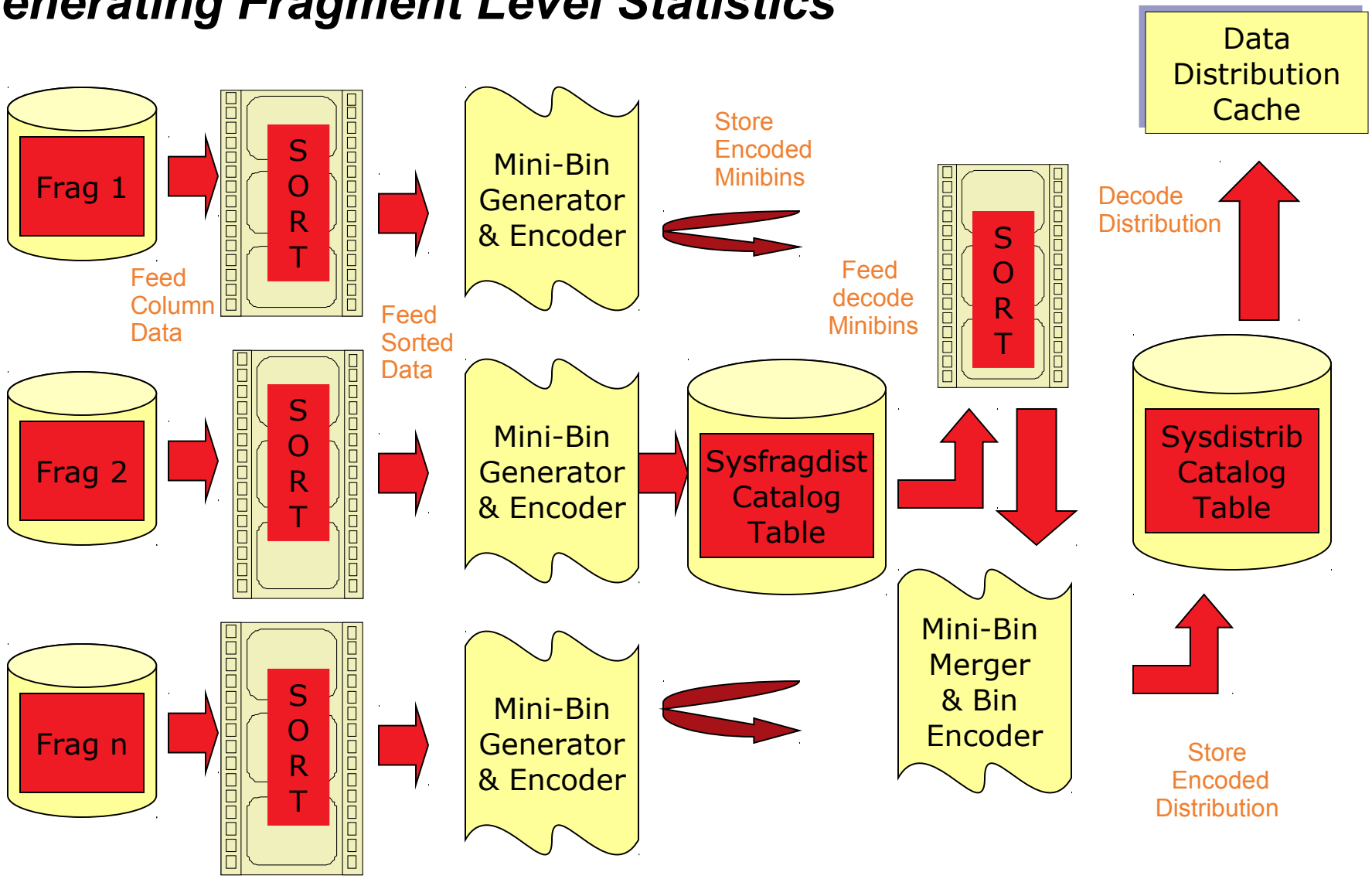
- External Tables
- Intervall Fragmentation
- **Fragment Level Statistics**
- Storage Provisioning
- Multi Index Scans
- Pushdown Hash Join

Generating Table Level Statistics



- Distribution created for entire column dataset from all fragments.
- Stored in sysdistrib with (tabid,colno) combination.
- Dbschema utility can decode and display encoded distribution.
- Optimizer uses in-memory distribution representation for query optimization.

Generating Fragment Level Statistics



Data Warehouse Features

- External Tables
- Intervall Fragmentation
- Fragment Level Statistics
- **Storage Provisioning**
- Multi Index Scans
- Pushdown Hash Join

Storage Provisioning

- Automatic expansion of dbspaces, temporary dbspaces, sbspaces, temporary sbspaces, and blobspaces.

- Manual / Automatic
- Chunk extension / creation

- "Out-of-space" errors are significantly reduced.

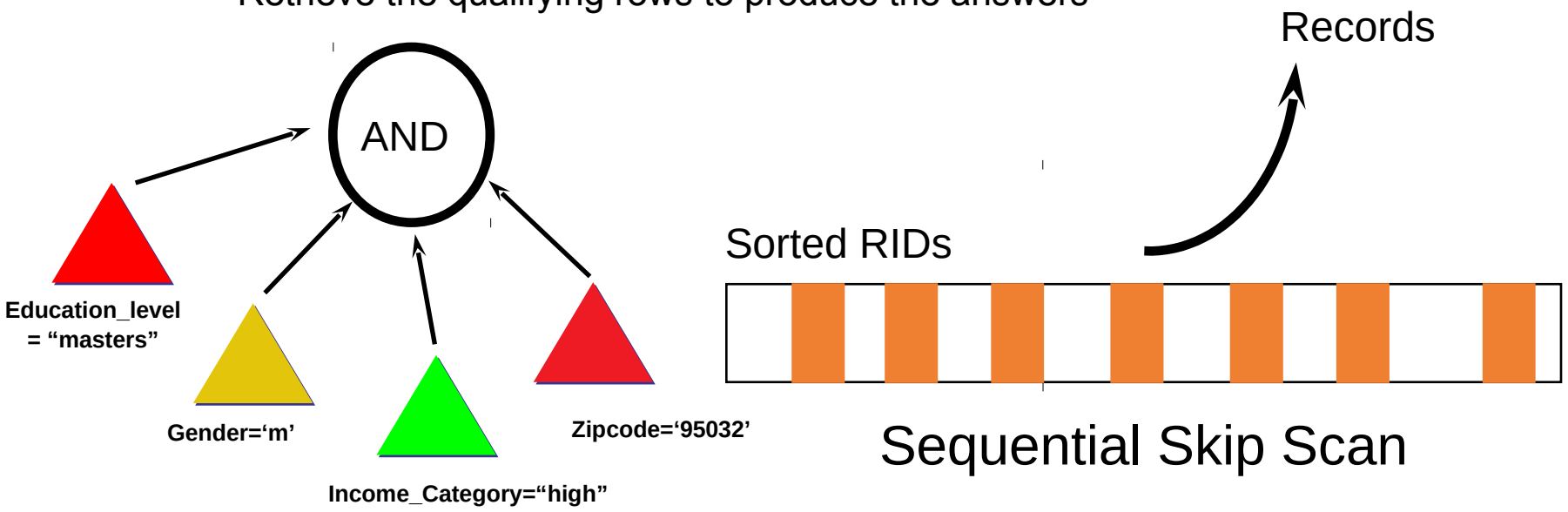
Data Warehouse Features

- External Tables
- Intervall Fragmentation
- Fragment Level Statistics
- Storage Provisioning
- **Multi Index Scans**
- Pushdown Hash Join

Multi Index Scan

- **Traditional Method:**
  - Evaluates the most selective constraint
  - Generates a list of rows that qualify, and
  - Evaluate the remaining constraints for each of the rows generated above which will produce the answer to the query

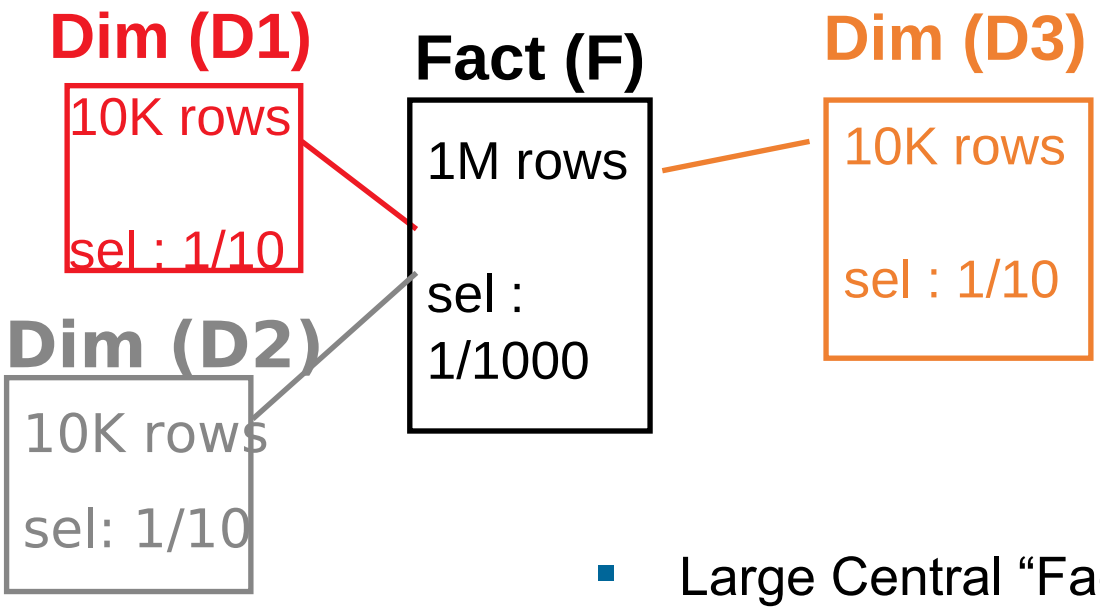
- **Multi Index Scan Method:**
  - Evaluate each constraint by using a different B-tree index on each attribute results in a list of rows that qualify for each constraints.
  - Merge the lists to form one master list that satisfies all the constraints
  - Retrieve the qualifying rows to produce the answers



Data Warehouse Features

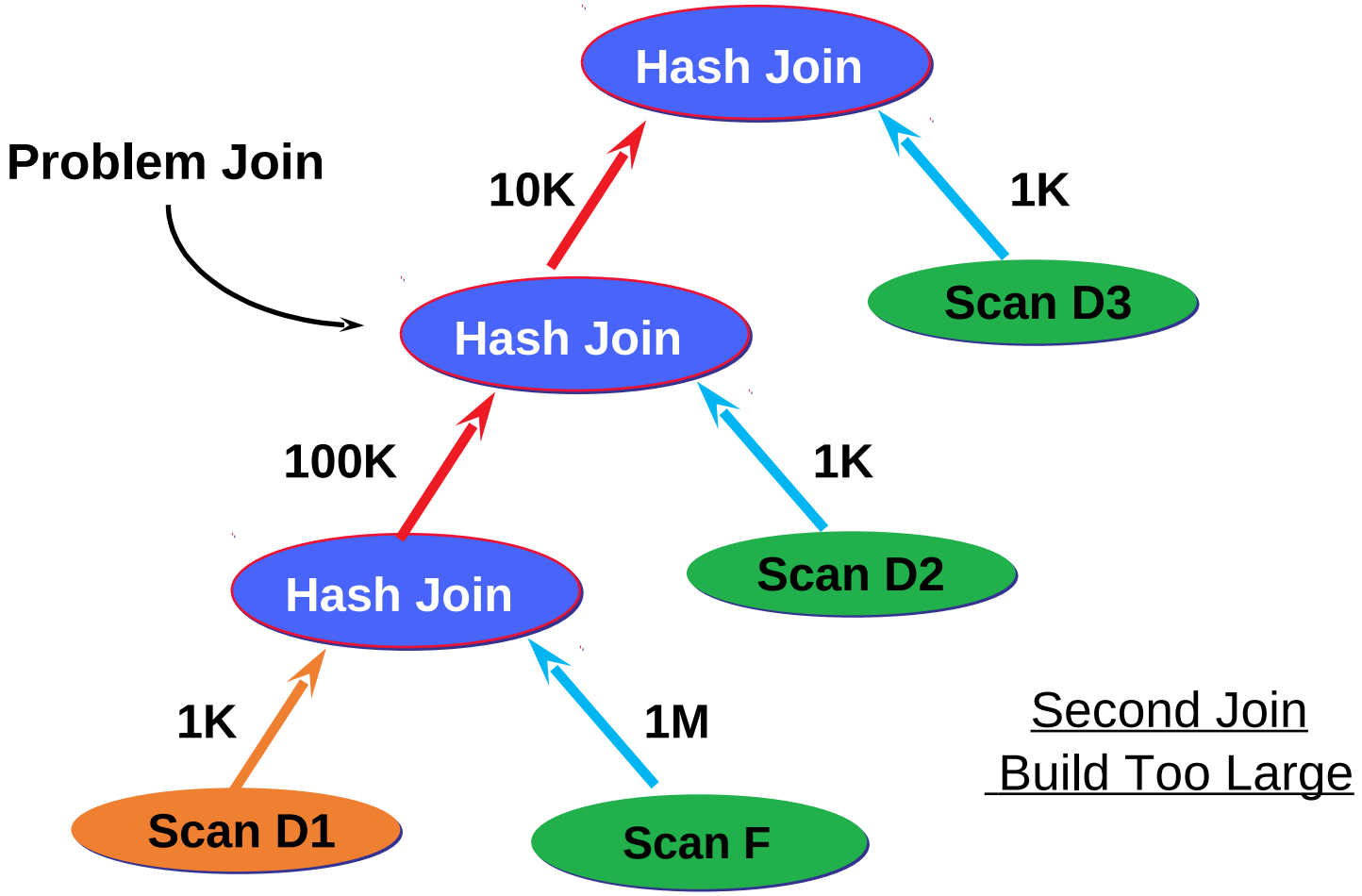
- External Tables
- Intervall Fragmentation
- Fragment Level Statistics
- Storage Provisioning
- Multi Index Scans
- **Pushdown Hash Join**

Typically Star Schema

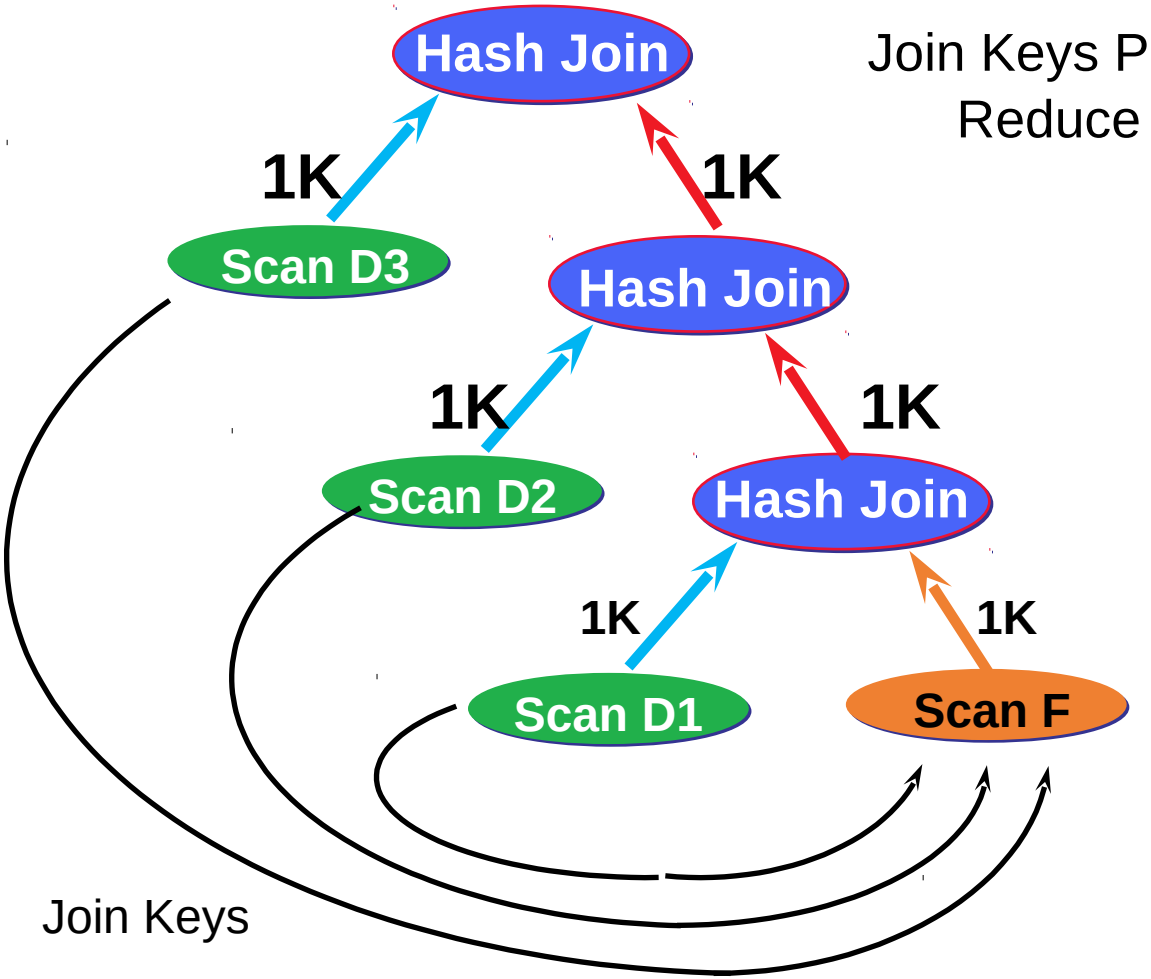


- Large Central “Fact” table
- Smaller “Dimension” tables
- Restrictions on Dimension tables
  - assume independence
- Small fraction of Fact table in result

Taditional Hash Joins



Pushdown Hash Join Solution



Join Keys Pushed Down to Reduce Probe Size

Multi Index Scan of Fact Table using Join Keys and Single-Column Indexes

Michael Köster - IT Specialist
November 2010

What's new in Informix 11.70?