Migration to Open Source Databases

Jutta Horstmann

www.osdbmigration.org

whoami

- Unix/Linux sysadmin
- DBA, developer (Informix)
- DB developer (Oracle)
- Web stuff (MySQL, PostgreSQL)
- Claim to Fame: OpenUsability.org
- Comp Sci Diploma Thesis:

  Migration to Open Source Databases

Agenda

- What's this? → About
- Why migrate? → Pros and Cons
- Where to migrate? → Open Source Databases
- How to migrate? → Workflow: Activities
- What to migrate? → Workflow: Assets

Migration?

*"The **process** of changing from the use of one platform, environment, IT system, etc., to another, esp. in such a way as to **avoid interruptions** in service."*

(Oxford English Dictionary)

Migration: Objectives

The target system:

- same functionality
- extendable
- incorporate all data
- modern hardware, software, architecture

The migration workflow:

- Minimize risk
- Stay on budget
- Deliver in due time
- Minimize downtime

OSDB Migration:Pros and Cons

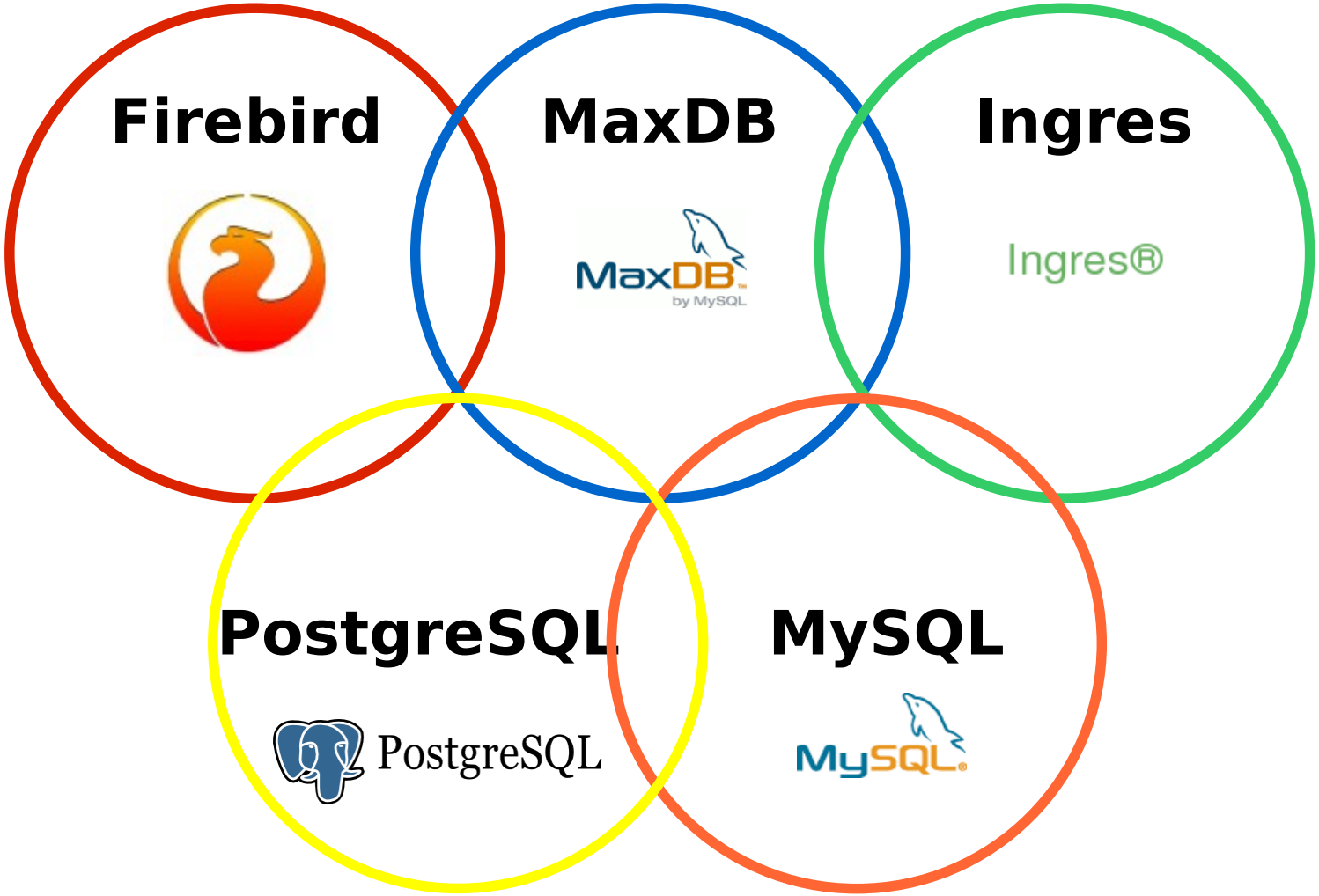
No

- Cost? Time? Effort?
- Lack of roadmap
- Licensing
- ISV Support
- Maintenance
- Accountability
- Features

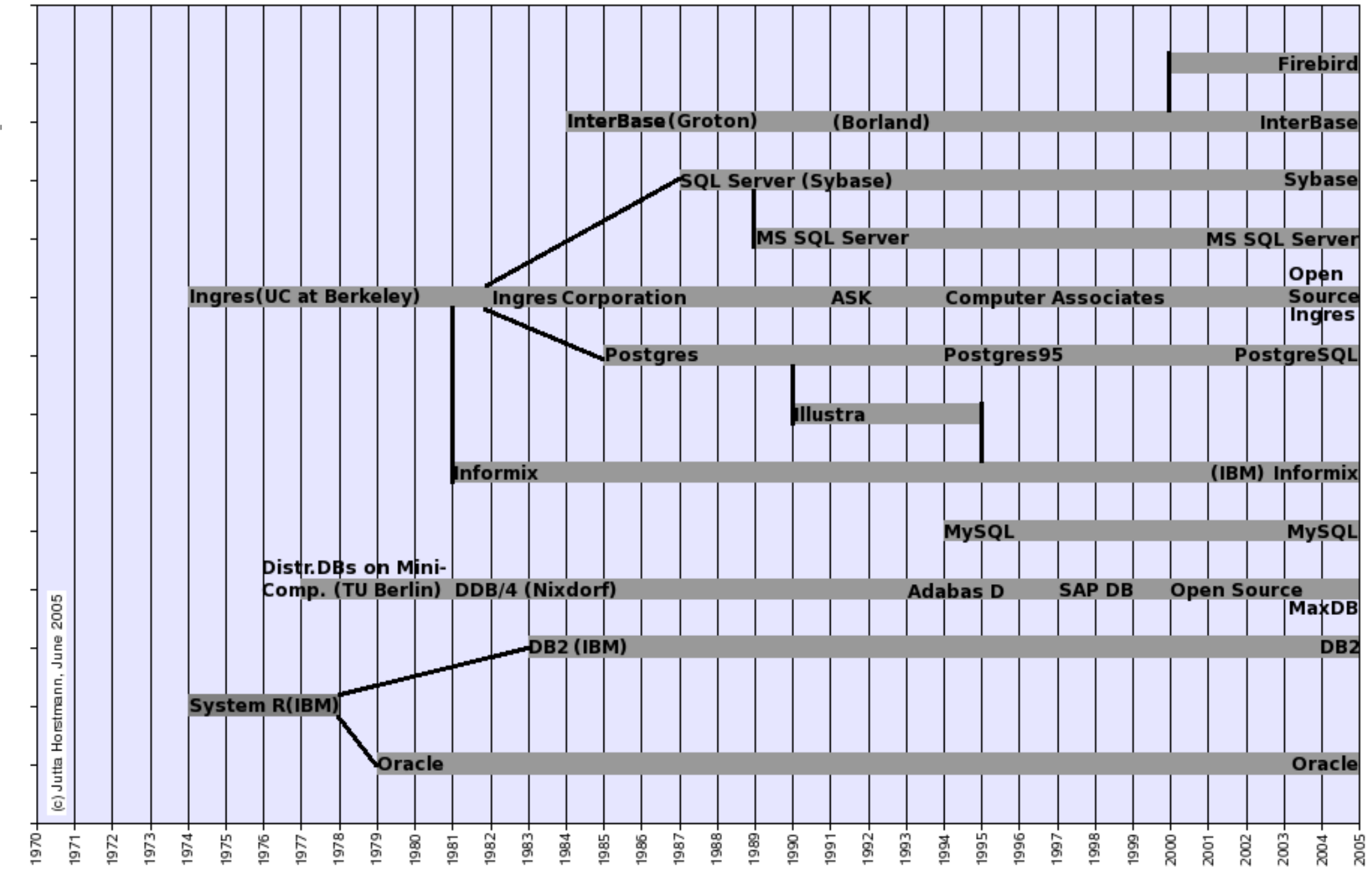
Yes

- Features?
- TCO
- Open Source
  - Code
  - Community
- Standards
- Security
- Independence

Where to migrate?



RDBMS Timeline



Enterprise-Level?

Requirements:

- Advanced data integrity mechanisms
- Advanced database objects
- Advanced SQL features
- Advanced features, tools, support for
  - Performance and scalability
  - (High) Availability
  - Security

OSDB Features

	Firebird 1.5	Ingres r3	MaxDB 7.5	MySQL 4.1	PostgreSQL 8.0
Advanced Indexing	☹	😐	😐	😐 MyISAM	✓
GIS support	☹	✓	☹	✓ MyISAM	✓
MVCC	✓	☹	☹	✓ InnoDB	✓
Two phase commit	✓	✓	☹	v.5.0 ☹	v.8.1
UD Functions	✓	✓	v.7.6	v.5.0	✓
UD data types	☹	✓	☹	☹	✓
Updatable Views	✓	☹	✓	v.5.0	✓

Performance / Scalability

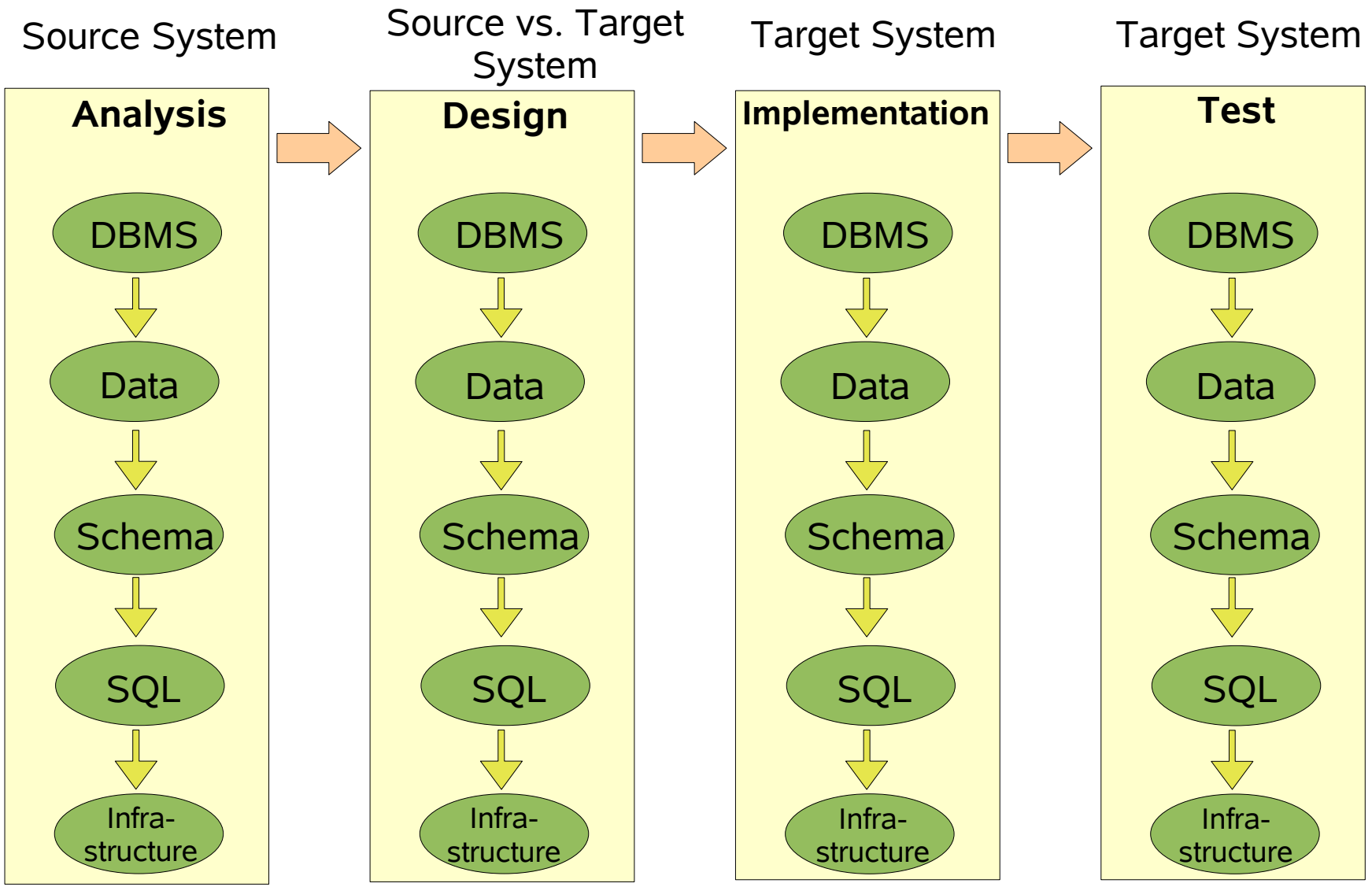
	Firebird 1.5	Ingres r3	MaxDB 7.5	MySQL 4.1	PostgreSQL 8.0
Tablespaces	☹	☹	☹	✓	✓
Table Partitioning	☹	✓	☹	☹	v.8.1
Parallelization	☹	✓	✓	☹	☹
Built-In Clustering	☹	✓	✓	✓	☹
Built-In Load Balancing	☹	✓	☹	✓	☹

How to migrate?

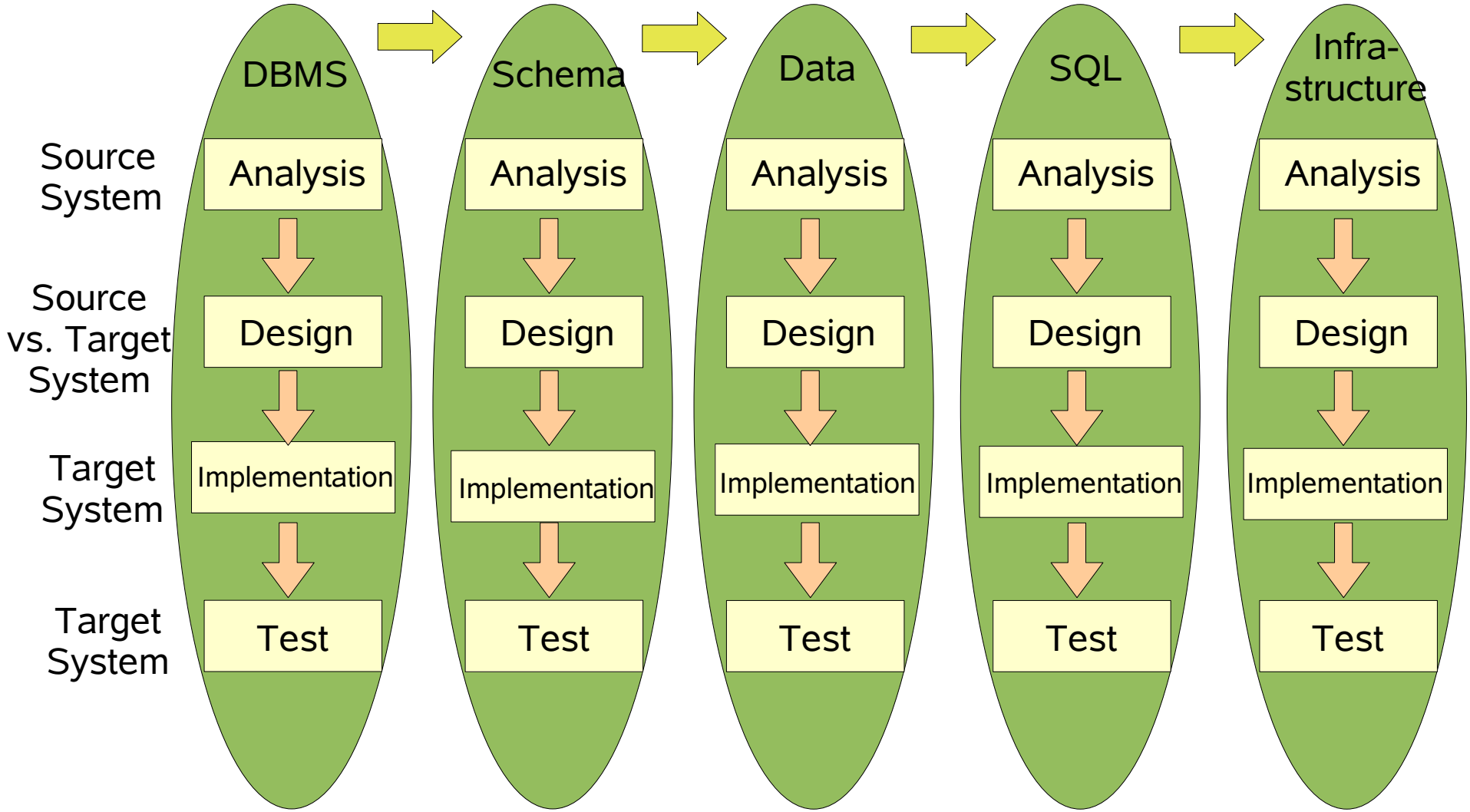
Database Migration Workflow

- Activity-Centered?
- Asset-Centered?
- Proposal: Mixed-Model Workflow
- Walk-Through

Activity-Centered Workflow



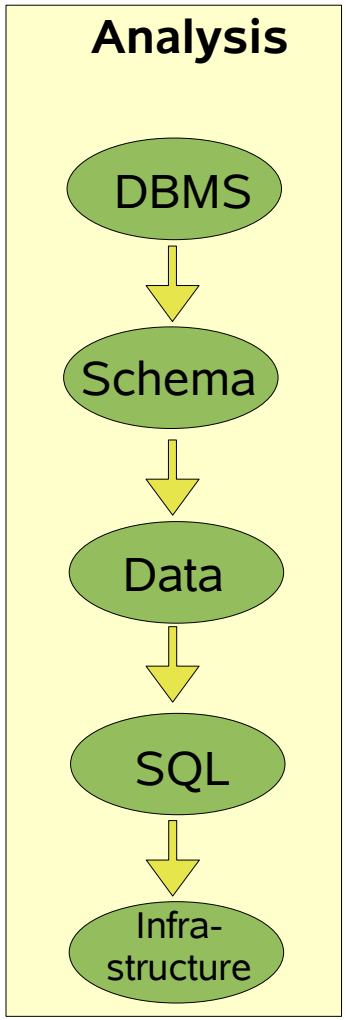
Asset-Centered Workflow



Mixed-Model Workflow

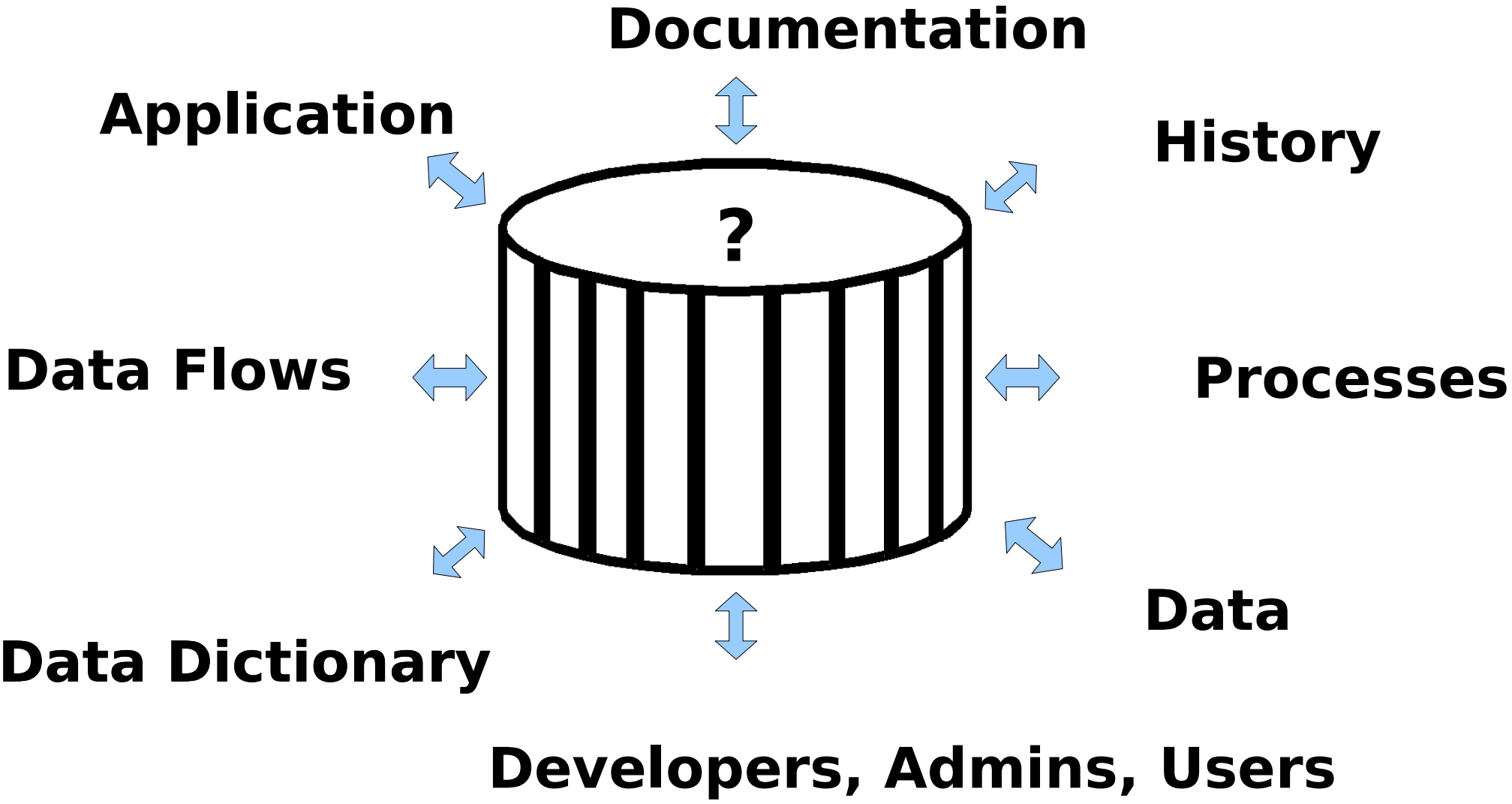
1. Analyze the whole source system (all assets).
2. Migrate the DBMS software (design, implement, test).
3. Migrate the schema(s) (design, implement).
4. Migrate test data (design, implement, test).
5. Test the migrated schema(s) with the test data.
6. Migrate the client SQL (design, implement, test).
7. Migrate the database system's infrastructure (d, i, t).
8. Cut-over: Full data migration.
9. Final testing and evaluation.

Analyze the whole system

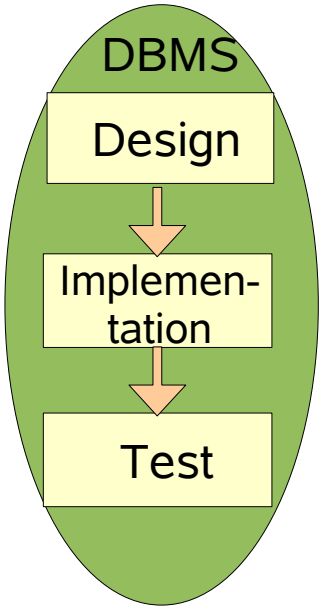


- How is the software configured?
- Complexity, quality of database design?
- How much data? Which condition?
- SQL isolated? Standard compliant?
- Tools, Policies, Tuning ...?

Where to look? Who to ask?

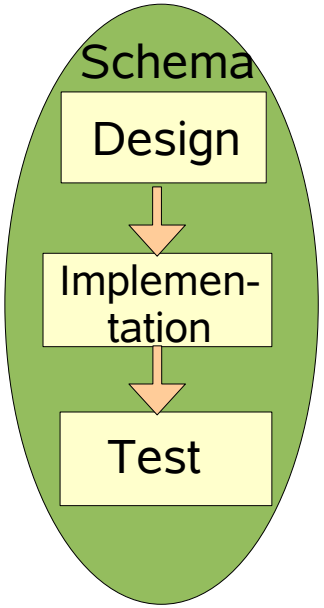


Migrate the DBMS software



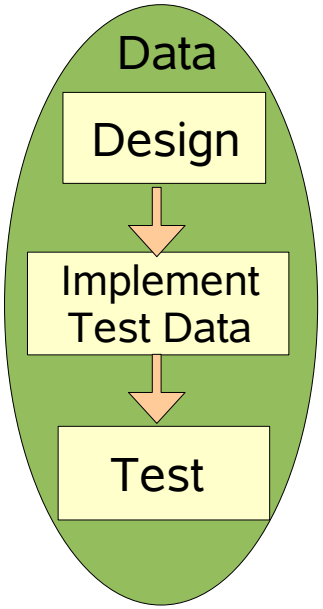
1. Choose Open Source DBMS
2. Devise mappings for configurations
3. Install & configure new software
4. Validate the working of the DBMS by using test databases and tools

Migrate the schema(s)



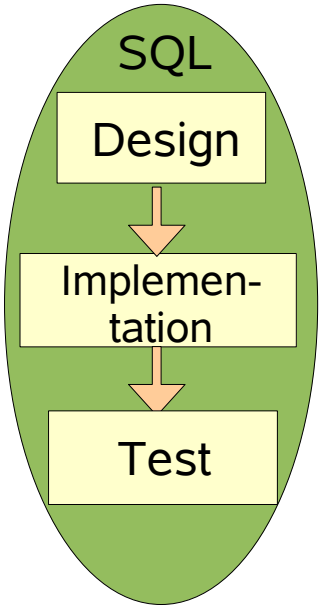
1. Provide abstraction (logical, conceptual?)
2. Devise mappings to target system
3. Implement by loading converted schema into target system
4. Validate the schema's correctness based on loading and querying test data.

Migrate Test Data



1. Correction and “cleansing” of “dirty data”
2. Devise conversions and mappings
3. Transfer some test data:
   a) Extraction
   b)Transformation
   c) Load
4. Test: Errors at load? Query data!

Migrate Client SQL

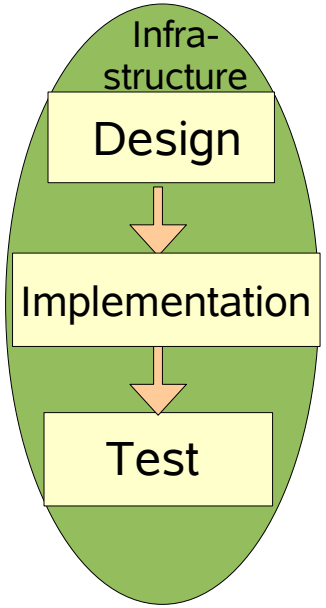


Database connectivity

- Change ODBC/JDBC driver
- Switch client
- Client source available: Migrate!

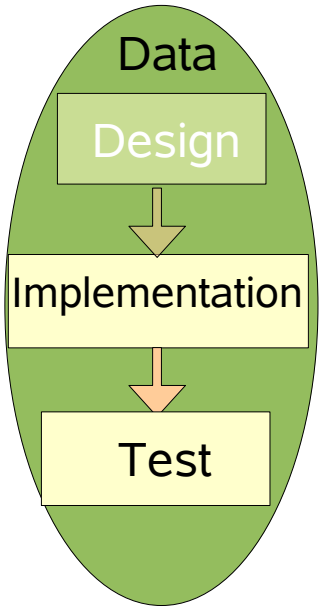
1. Provide abstraction (Standard SQL) OR
2. Convert directly to target SQL syntax
3. Test statements against schema and test data

Migrate Infrastructure



- Tools: Administration, Development, Design
  - Change ODBC/JDBC driver
  - Switch client
- Administrative Tasks
  - Gather policies and jobs
  - Implement them in the target system way

Full Data Migration (Cut-Over)



1. Strategy:

   a) Cold Turkey?

   b) Chicken Little?

   c) Butterfly?

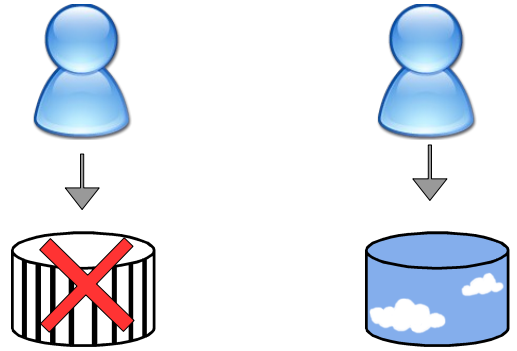
2. Data Transfer:

   Extraction, Transformation, Load

3. Test: Errors at load? Query data!

“Cold Turkey”

- Take source system offline
- Extract data
- Transform data
- Load data into target system
- Take target system online

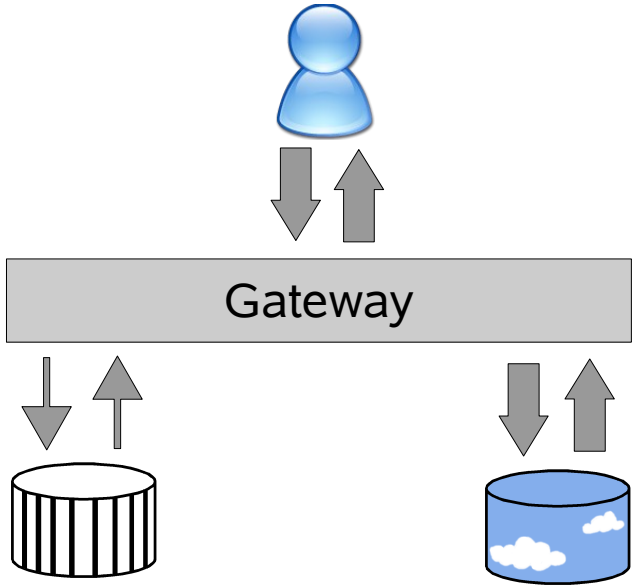


Problems:

- How long offline? How much data?
- Fallback?

“Chicken Little”

- Source and target system operate in parallel
- “Gateway” coordinates and maps queries
- Incremental data migration



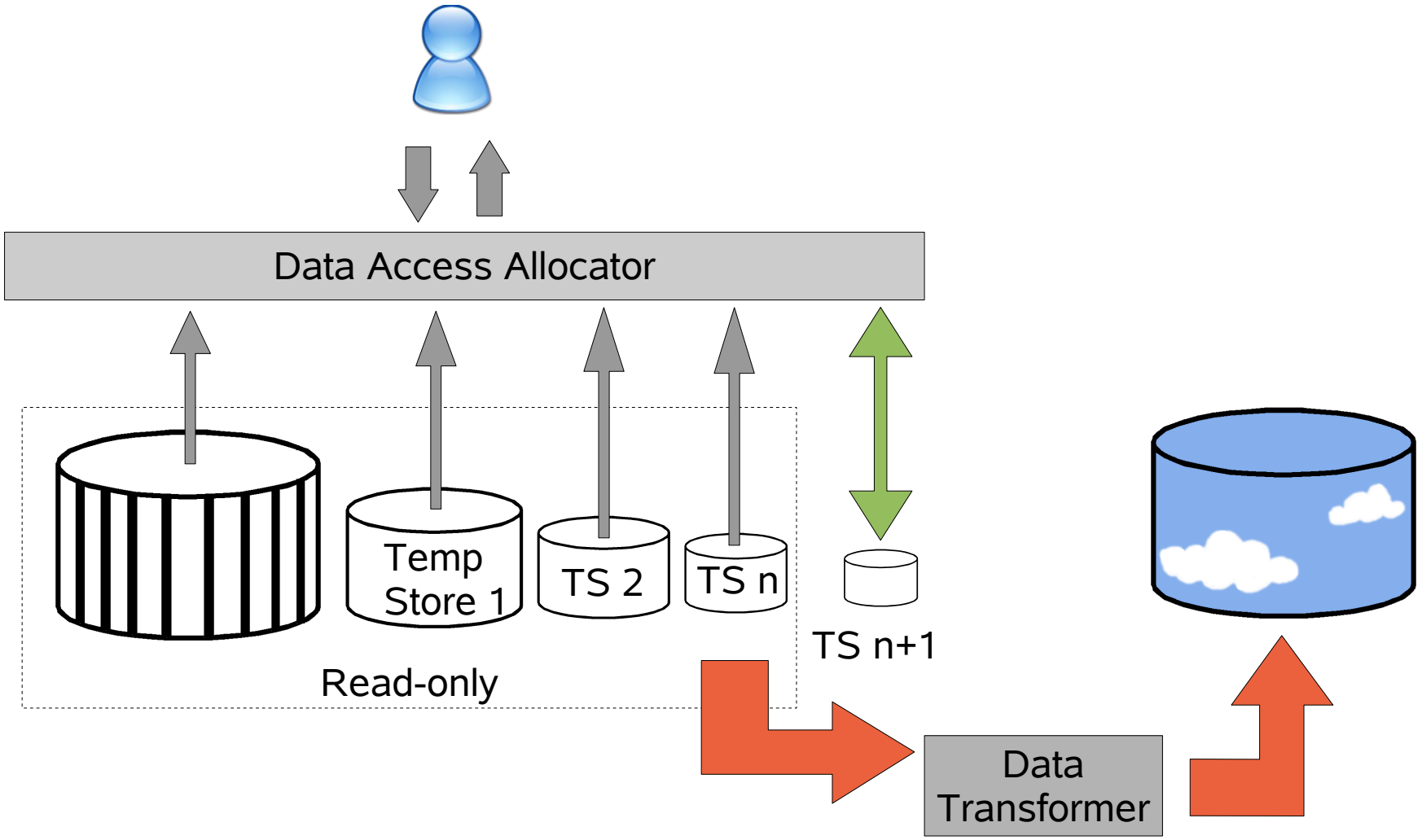
Problem:

- Complex implementation
- Update consistency?

“Butterfly”

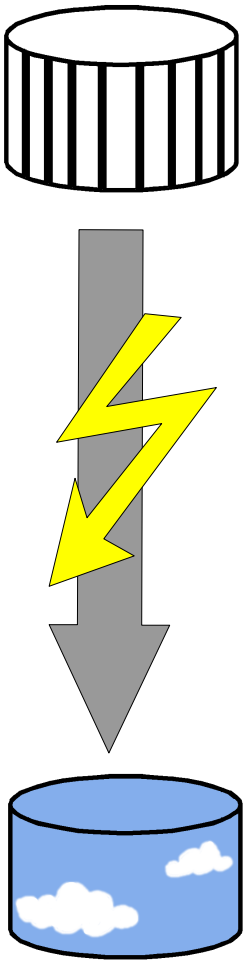
- Only source system online
- Data gets frozen read-only
- Store manipulation results in “Temp Store 0”
- Transfer frozen data
- Freeze TS0, store manipulation results in TS1
- Transfer frozen data ... and so on
- “Data Access Allocator coordinates and maps queries
- Problem: Complex implementation!

“Butterfly”


Data Access Allocator
Temp Store 1
TS 2
TS n
TS n+1
Read-only
Data Transformer

Summary: Problems

- Different SQL implementations
- User-defined data types/functions
- Stored Procedures
- Proprietary interfaces
- Flawed schema design
- Differences in Data Types
- “Dirty Data”
- Reserved Words



Help! - Migration Tools

- Automatization
- Integrity violation detection
- Re-Use
- Knowledge on source and target system
- Documentation
- Code Generation
- Script Scheduling
- Validation
- Speed!

Migration Tools, Examples

- Closed Source:
  - SQLWays (Ispirer). Only Windows.
    - Source: any, Target: MySQL, PostgreSQL
  - ProgressionDB (Versora). Linux, Windows.
    - MSSQL -> MySQL, PostgreSQL, Ingres
- Open Source:
  - Ingres Mio. $ Challenge (CA): shift2ingres
  - MySQL Migration Toolkit (MySQL AB)

Summary

Migration to Open Source Databases ?

- Several Open Source options
- Migration is an ambitious project
- Workflow and results depending on state of source system
- Automatize the process, where possible!

Questions?