MySQL:
Past, Present and Future

David Axmark
Co-Founder

MySQL AB Creators of MySQL
Vilnius, Lithuania 2004-03-26

A bit different OpenSource project

- The MySQL project was started to make money (but not only for this!)
- MySQL server has been developed by a company (with paid developers) not by volunteers.
- But we could not have done it without all the help with word of mouth marketing, bug reports, ideas, client and tool development and feedback...

Beginnings

- In the 1990 I was working with the other CoFounders of MySQL doing data warehousing work in Sweden
- Our customers had problems with our home developed tools and wanted us to switch to “standard” tools
  - We tried and hated it
- At the same time the web was growing (1994-1995) so we needed solutions more suited to the web
- So we choose to develop our own commercial OpenSource SQL server with a ODBC interface for windows

Initial Business Idea

- To get a SQL server well know so that we could sell commercial non-GPL licenses and support globally without normal marketing
- We distributed the code under two licenses from the very beginning
  - One “OpenSource”. First a stricter license that forbid some uses and later the GPL (from June 2000). This license requires the customer to distribute **their** source code if they include MySQL in a product.
  - And one “normal” commercial license without any special demands on the customer

Technical goals

- MySQL has always been developed for practical production use
  - We did not care much about Database theory only practical use
- If MySQL crashed in the initial years it was me and Monty (other CoFounder) who had to get out of bed and fix it
- We spent a **lot** of time making MySQL easy to install and use.
  - The 15 Minutes rule: The goal was (and is) to make it possible to install MySQL and try it out in 15 minutes.
- We wrote documentation at the same time we wrote code (uncommon in OpenSource projects)

MySQL Google Numbers & Downloads

- # of Mentions on the web according to Google.com

– MySQL	17.80M	Oracle	13.70M
– DB2	2.62M	Sybase	2.31M

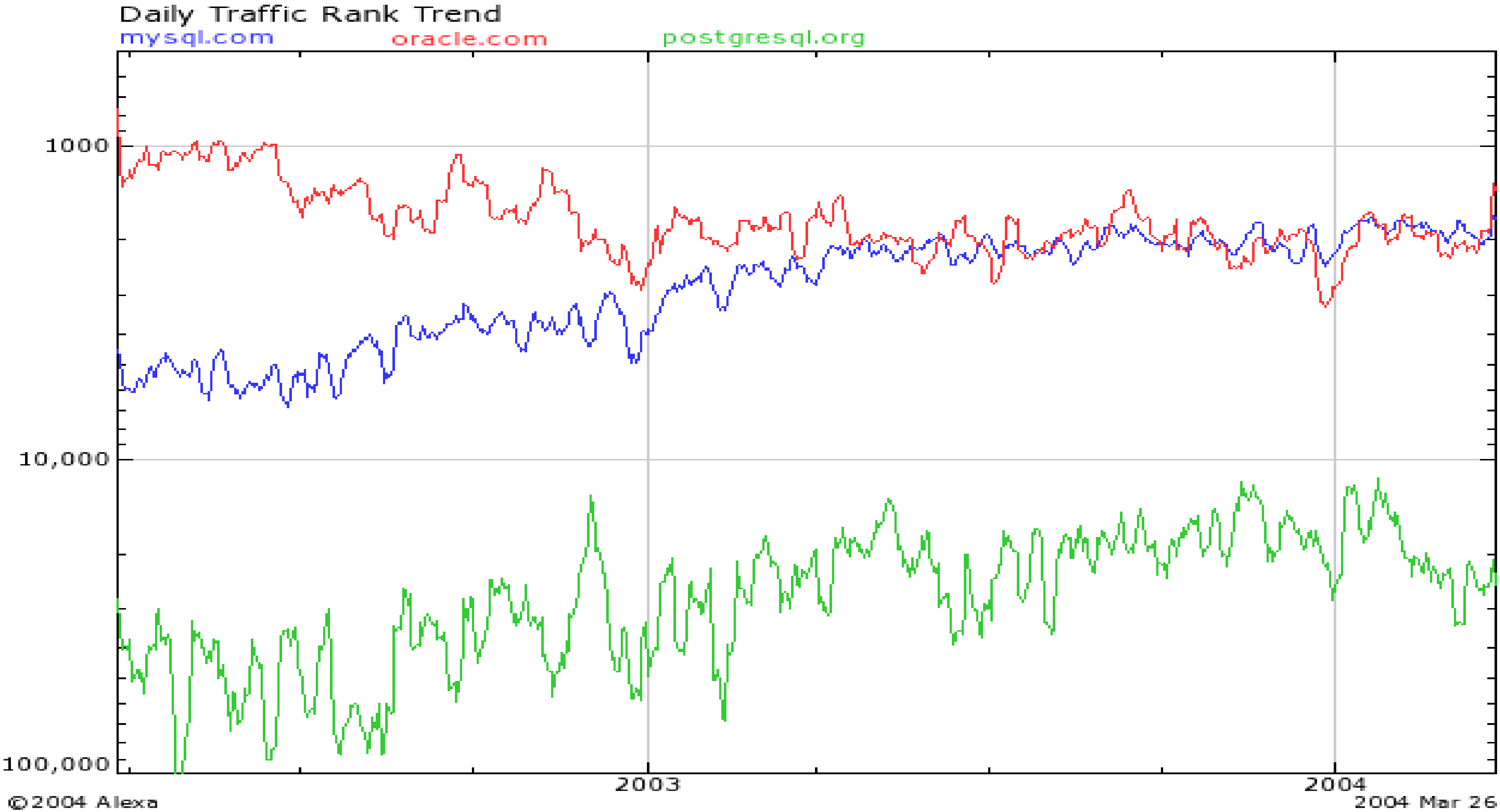
- # of Links to site according to Google.com

– MySQL.com	79.100	Oracle.com	18.500
– IBM.com	82.200	Sybase.com	7.070

- Numbers from 2004-03-28 (Numbers change all the time)
- Over **30k** server downloads per **day** through mysql.com (**1 Million per Month**)
- Forrester research interviewed 140 enterprise companies and found that 31% are using MySQL (and 60% FLOSS)

Alexa (Amazon) Web Stats

A graph of daily web rank of MySQL.com (1793) compared to Oracle.com (1666) and PostgreSQL.org (16215)



Press who has written about MySQL

THE WALL STREET JOURNAL.
REUTERS
KNOW. NOW.
FT FINANCIAL TIMES
World business newspaper
FORTUNE
The New York Times
The Mercury News
BusinessWeek online
CNN
RED HERRING
DOW JONES
The Deal
venturewire

Some Awards and notable events

- 1998 WEBTechniques & CNET: Best affordable DB
- 1998-2003 **Linux Journal**: Readers choice (**6 times**)
- 2000-2001 **Linux Magazine**: Editors choice
- 2001 MySQL AB is formed and takes investment
- 2001 Oracle introduces MySQL migration kit
- 2002 **Linux Journal**: Editors choice
- 2003: Guldmusen for Sweden's best IT Product

Quotes from Press & Analysts

“MySQL AB's use of the Open Source software model is propelling its adoption at a much faster rate than any traditional software sales model could have.”
META Group

“The open source movement has become a major force across the software industry, and MySQL is the world’s most popular open source database.” ***Fortune Magazine***

“CIOs should put MySQL on their radar screen… MySQL is a powerful enough force to alter the database market”
Forrester, Inc.

“Our research indicates that MySQL has provided exactly what it advertised - a fast, low-maintenance, extremely stable database product.”
META Group

“MySQL CEO Marten Mickos, makes a virtue of his product's stripped-down simplicity. “Software shouldn't be glorified,” he says. “We say, 'Let's do this as compactly as possible and sell it at a price that blows the competition away.' ”

The Wall Street Journal

Readers’ Choice Winner international php magazine

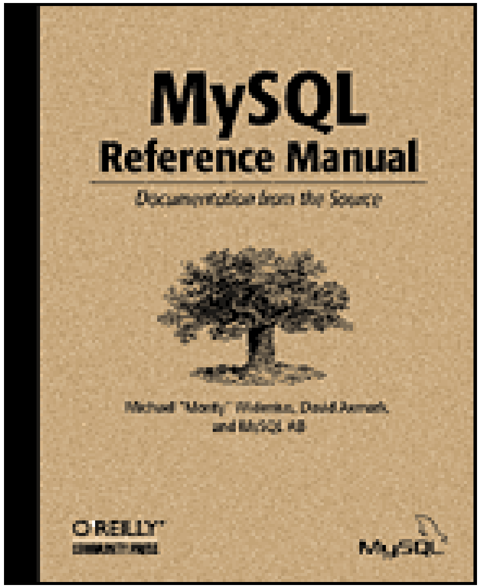
LINUX JOURNAL READERS' CHOICE 2003

2002 EDITORS' CHOICE LINUX

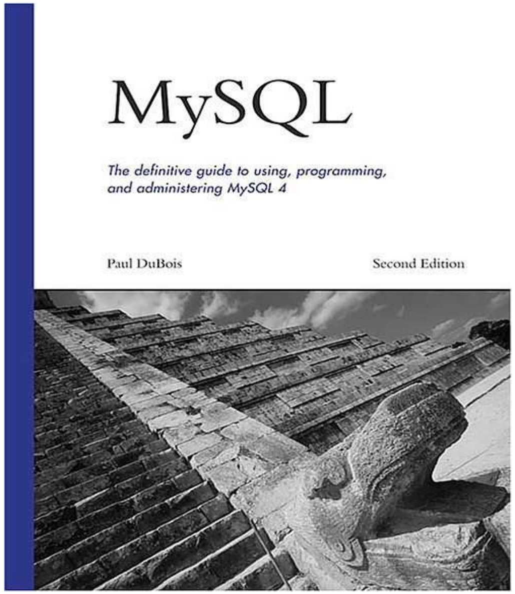
developer.com Product of the Year 2002 Winner

** WINNER ** OSDir.com EDITOR'S CHOICE 2003

Some MySQL Books

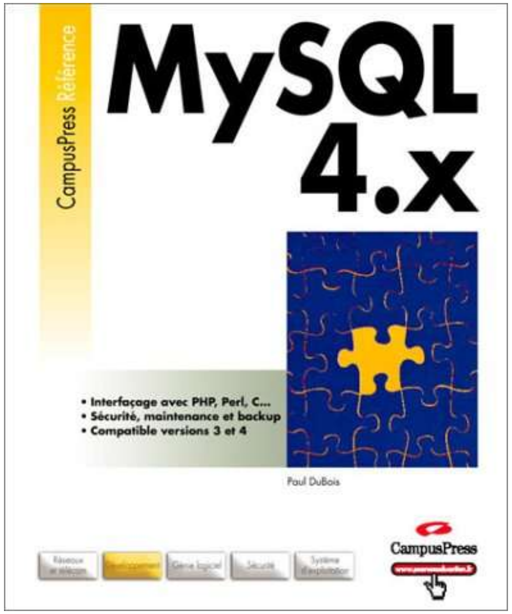

MySQL
Reference Manual
Documentation from the Source
Michael "Monty" Widenius, David Axmark, and MySQL AB
O'REILLY
MySQL

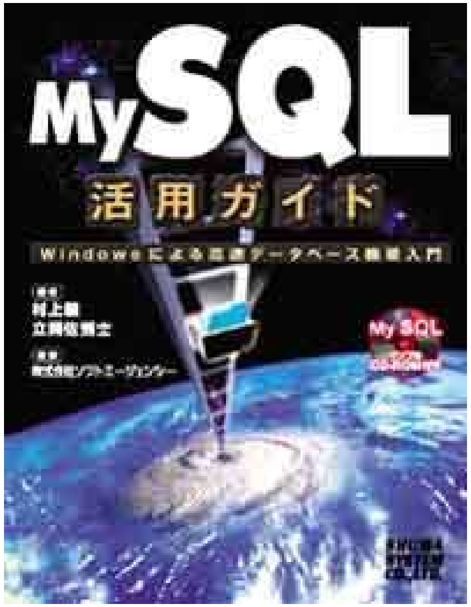
Recommended

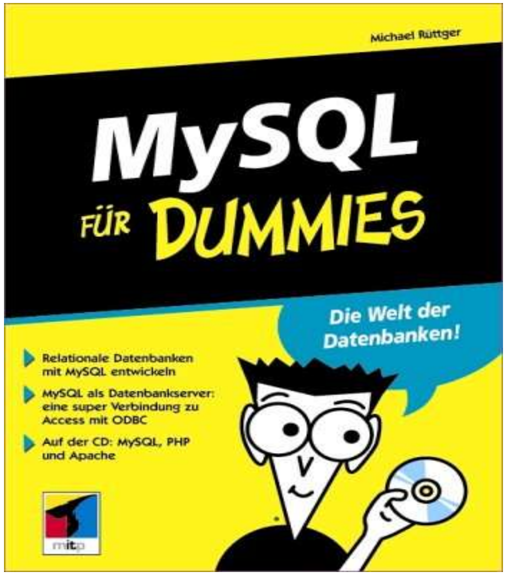

MySQL
The definitive guide to using, programming, and administering MySQL 4
Paul DuBois
Second Edition

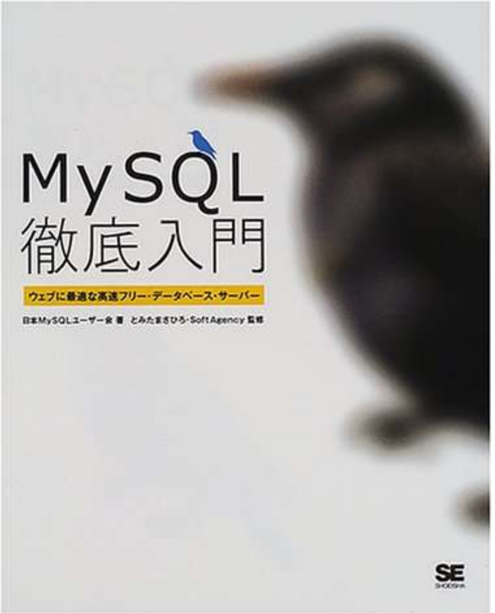
Recommended

DROIT AU BUT
spécial
vrai
débutant
Les livres qui vont @ l'essentiel
CD-ROM inclus
Votre première base de données avec MySQL
Jean Engels
OSMAN EYROLLES MULTIMEDIA


CampusPress Référence
MySQL 4.x
• Interfaçage avec PHP, Perl, C...
• Sécurité, maintenance et backup
• Compatible versions 3 et 4
Paul DuBois
CampusPress



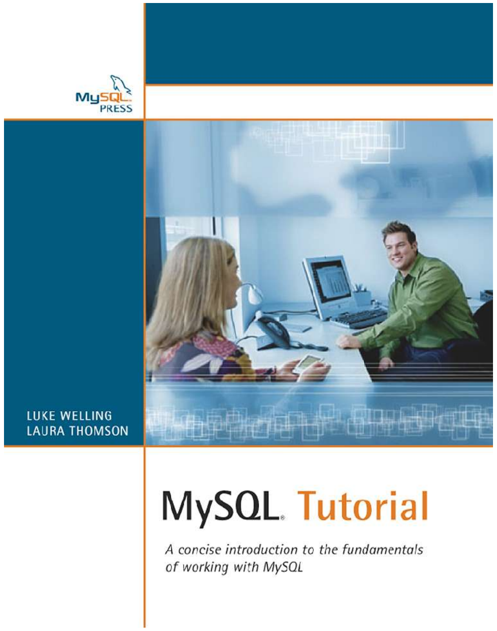

Michael Rüttger
MySQL
FÜR DUMMIES
Die Welt der Datenbanken!
Relationale Datenbanken mit MySQL entwickeln
MySQL als Datenbankserver: eine super Verbindung zu Access mit ODBC
Auf der CD: MySQL, PHP und Apache
mitp

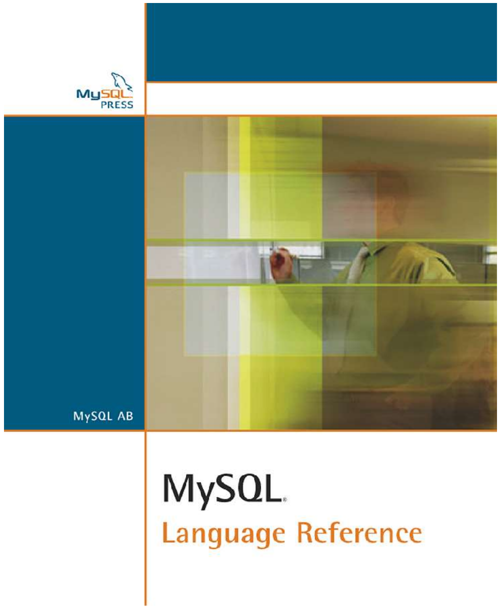


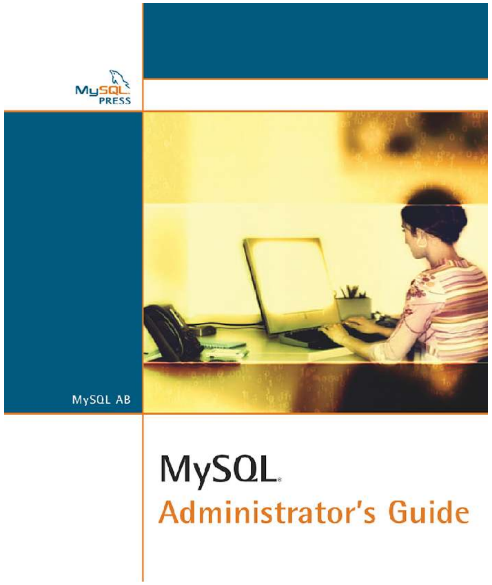
- There exists **LOTS** of MySQL books in many languages!

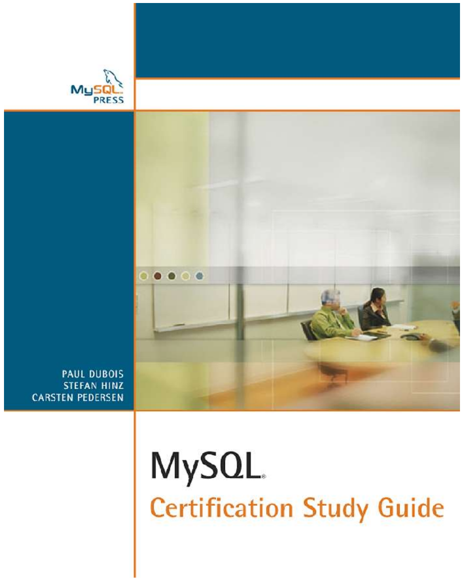
MySQL Press

- MySQL Press was launched in February by MySQL AB and Pearson Education
  - Will help you find the books that are checked by experts
  - Pearson also does x-Press for Cisco, Novell, HP, IBM, MS


MySQL PRESS
LUKE WELLING
LAURA THOMSON
MySQL Tutorial
A concise introduction to the fundamentals of working with MySQL







Associated Press (AP)

Application

Provides news text, audio, video, photos to more than 15,000 media outlets reaching 1 billion people daily worldwide .

Key Business Benefit

AP can handle customer demand for large volumes of information, even during peak transaction loads

Why MySQL? Performance / Low TCO

"The key benefits of using the MySQL database are its perform ance, its stability and its low total cost of ownership. Bottom line: it just works."

Terry Ewing,
Senior Systems Manager
Associated Press (AP)

Yahoo!

Application

MySQL powers over 200 Yahoo! properties around the world including Yahoo! Finance with 260 million rows.

Key Business Benefit

Low down time. Extremely reliable for high-volume, business-critical applications

Why MySQL? Mission-Critical Reliability

“We have used MySQL far more than anyone expected. We went from experimental to mission-critical in a couple of months.”

Jeremy Zawodny,
MySQL Database Expert
Yahoo! Finance

Cox Communications

Application

MySQL powers a huge data warehouse for 6.3 million cable users. It includes 3,600 tables and 2 billion rows of data with 4 million inserts every 2 hours.

Key Business Benefit

Allows Cox to make informed business decisions and greater customer care

Why MySQL? – Best Price / Performance

"In today's market, we are constantly striving to keep costs down, Leveraging the MySQL database allows us to save money without sacrificing performance."

Mark Cotner, Manager
Network Application Development
Cox Communications

NASA

Application

Informing the general public about contract opportunities within NASA

Key Business Benefit

Saves NASA and its partners $4 million annually with 28% faster database

Why MySQL? Proven Reliability

“We've been up and running for 3 years now without any data loss or downtime. That's just incredible.”

John Sudderth,
Senior Computer Scientist
CSC NASA

Sabre Holdings/Travelocity.com

Application

Air Travel Shopping Engine fare search used by Travelocity and others moved from mainframe to Linux, MySQL.

Key Business Benefit

Multi-million dollar savings while supporting million of daily transactions

Why MySQL? Speed and Low TCO

“MySQL ran faster or as fast as any commercial database we tested. It never crashed. It was the fastest to get working.”

Alan Walker,
Systems Architect
Sabre Holdings

US Census Bureau

Application

Award-winning FedStats portal application aggregates leading financial and economic indicators at a glance.

Key Business Benefit

Cost savings from LAMP have enabled greater investment in other areas

Why MySQL? Easy to Deploy

“We chose the MySQL database because of its ease of installation, maintainability, configuration and speed.”

Rachael LaPorte Taylor,

Senior Technology Architect

U.S. Census Bureau

Lufthansa Systems

Application:

Accounting and data management of aviation customers

Key Business Benefit:

Reduced cost of operations with broad deployment of web applications.

Why MySQL? Stability

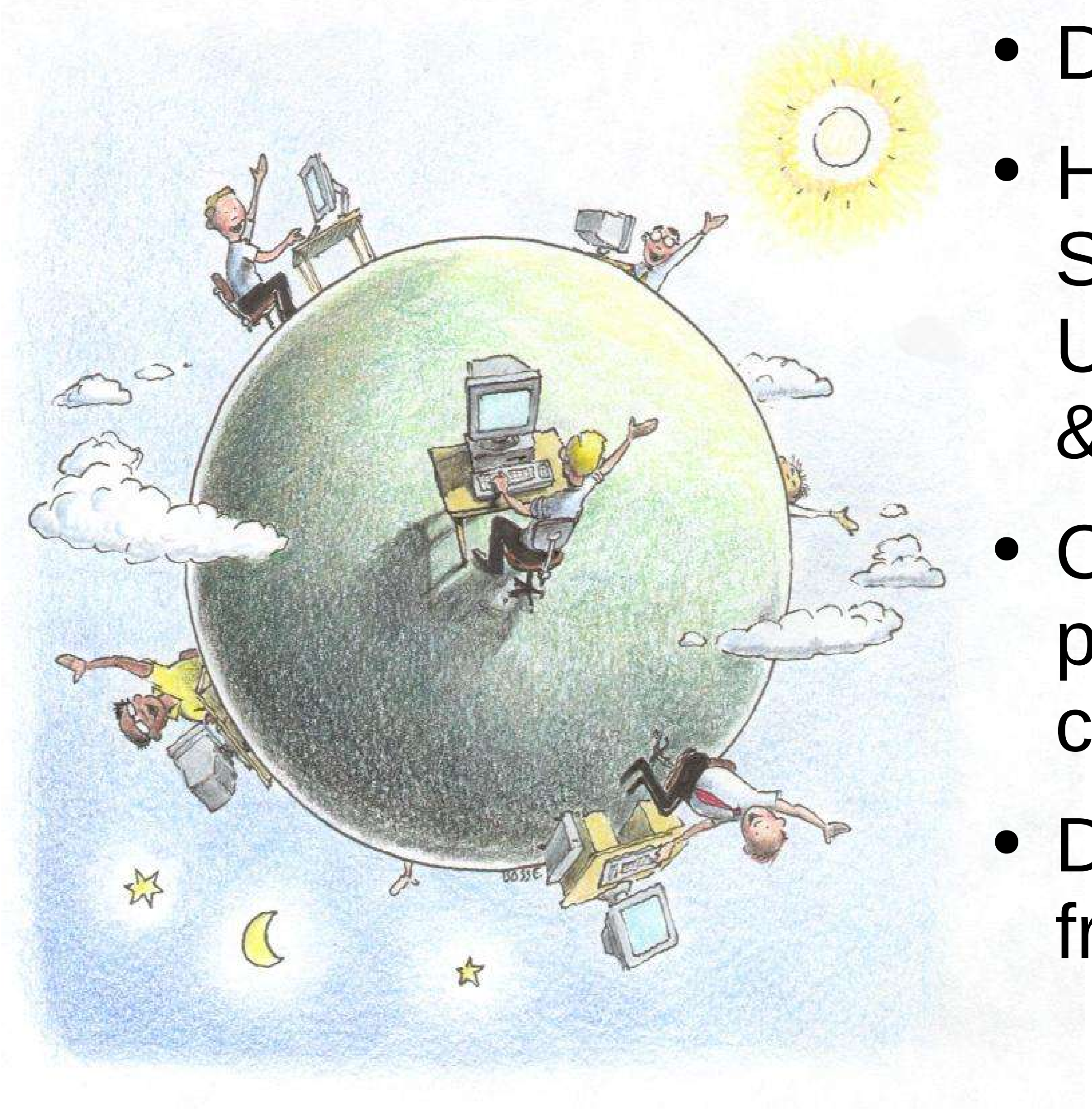
“The database is very stable, especially with NetWare and Linux.”

Antonio Mastrolorito
Engineering Directory Services
Lufthansa Systems

Other References

- Google.com (Web - for ad system)
- Slashdot.org, Freshmeat.net (Web)
- Cisco, Nokia, Ericsson (Embedded)
- Human Genome Project, Sanger Inst., Cambridge
- Danish Center for Biological Sequence Analysis
- Cybozu (Exchange leader), Rakuten (Japans Amazon/EBay), Sony, NEC

MySQL AB

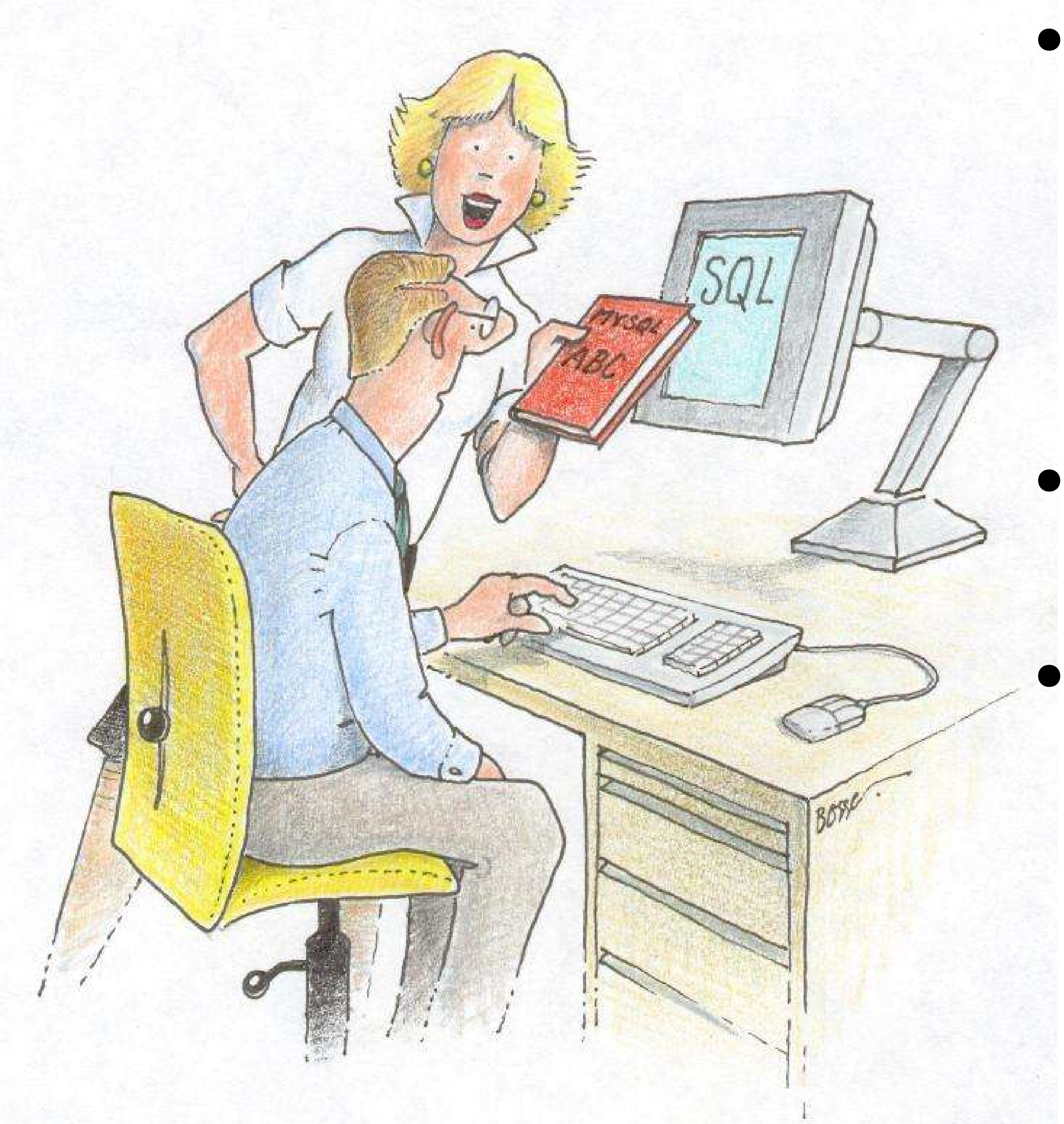


- Distributed company
- Head office in Uppsala, Sweden, Also offices in US (2), France, Finland & Germany
- Currently more than 120 people in about 18 countries
- Developers working from home

MySQL AB & Making Money

- **Commercial** (non GPL) **Licenses**
  - The GPL version contains more than the commercial!
  - Short rule: “If you are free (libre) we are free, if you are proprietary we are commercial”
  - Our most important source of income
- **Support** Agreements
  - Up to 24 by 7 phone support
- **Training**
- **Consulting**
- **Certification**

Training



- Advanced MySQL training is provided by MySQL AB. In many languages by partners
- Public classes or Inhouse
- mysql.com/training

MySQL Users Conference

- We hosted the first MySQL users conference in April (San Jose, California)
  - Big success with over 500 people
- Talks from everyone who matters in the MySQL world, including MySQL management, core developers, large users and industry analysts
- **Next MySQL conference 14-16 April, Orlando, Florida, USA**

SAP Deal

- We announced a partnership with SAP last year. (SAP is a huge ERP vendor with more than $7 billion in revenue)
  - MySQL AB has taken over SAP-DB and named it MaxDB.
  - MySQL AB will sell licenses and support for MaxDB
  - All key MaxDB features will be incorporated into a merged product that will run SAP R/3
  - Development will be done from the MySQL codebase
    - This is needed since SAP-DB is mostly written in Pascal
  - Experienced SAP-DB developers are helping us with the development of the combined version

Investment

- MySQL AB closed a investment round of 13.3 Million EURO last year
- The lead investor was Benchmark Capital (who also invested in RedHat and eBay)
- We will spend the money on expansion in Marketing, Sales and speeding up Development!
  - – Our first senior marketing person ever started in November
- And we will spend it slowly since we have come this far by keeping costs low (and performance hight!). And we plan to keep it that way!

Documentation

- About 800+ pages, updated as new code is written
  - We write code and documentation at the same time (and we have done that from the start!)
- Printed versions in English & German. Russian, French and Japanese are available on the web.
- We now have 3 full-time documentation writers. Including Paul DuBois who both has written some of the very best MySQL books

My (picture from 1999)

MySQL Features

- An extended subset of ANSI SQL:2003.
- Versioning ACID transaction support (InnoDB)
- Numerous extensions like SELECT .... LIMIT 10,10
- Few limits: disk, memory and OS set the limits
- Handles large datasets (hundreds of GB)
- Can be extended with User Defined Functions(UDF) in C
- Embeddable engine (MySQL server as a library. New in 4.0)
- Full text search
- Support for > 4GB tables and easy BLOB support
- Master-Slave Replication for data redundancy
- Per table choice of backend (with/without transactions)

Language Support

- **C**
- PHP
- Java (JDBC)
- Ruby
- Guile
- ADA
- Dylan
- Common Lisp
- Lasso
- Pike
- Perl
- Python
- C++
- TCL
- Delphi
- Rexx
- ODBC
- .NET
- Cobol
- And lots more ...

Supported platforms

Native Threads (Multi CPU)

Linux (x86, PPC, Sparc ...)

Windows 95, 98, NT, 2k, XP

Apple MacOS X

Sun Solaris 2.5+

Novell Netware

DEC (Tru64) UNIX

BSDI 3+

FreeBSD 3+

IBM OS/2

SCO Unixware 7

SGI Irix 6.x

HPUX 11+

IBM AIX 4+

QNX

(OpenVMS)

Emulated threads (Single CPU)

BSDI 2.x

FreeBSD 2.x

HPUX 10.20

NetBSD Intel/Alpha

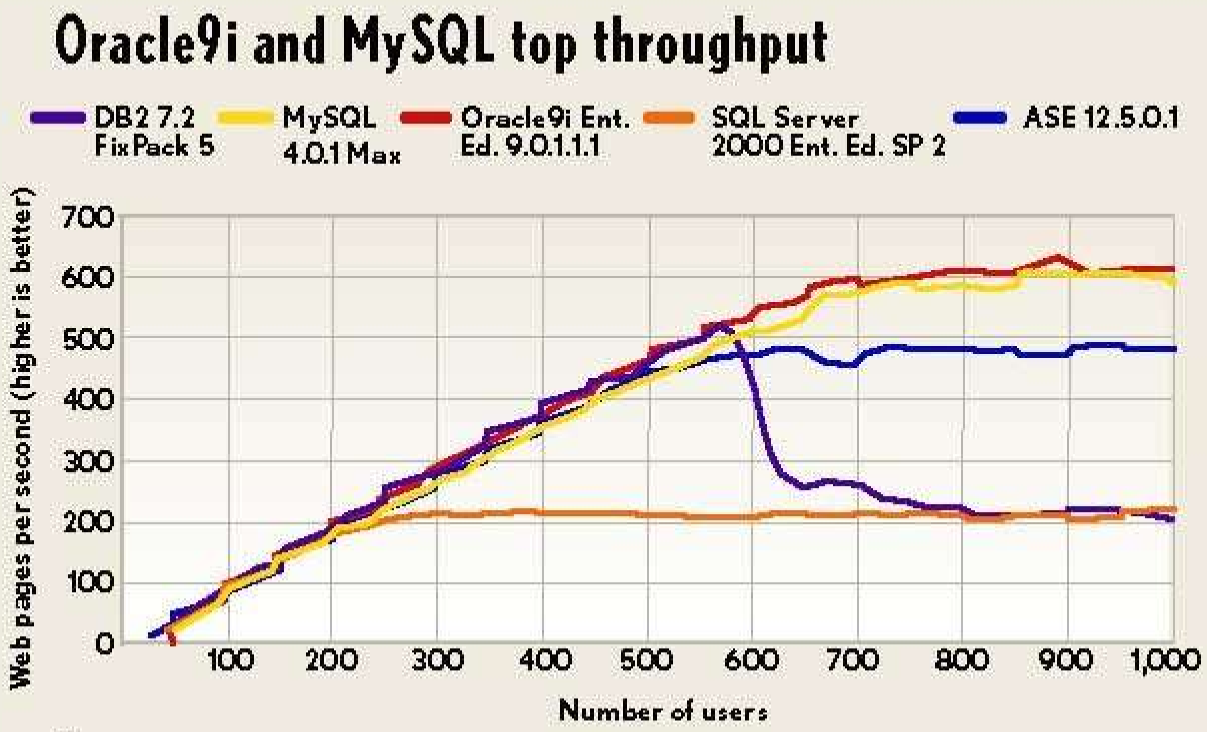
OpenBSD 2.x

Sun SunOS 4

-SCO Openserver (No GPL release)

(AmigaOS)

EWeek Benchmark



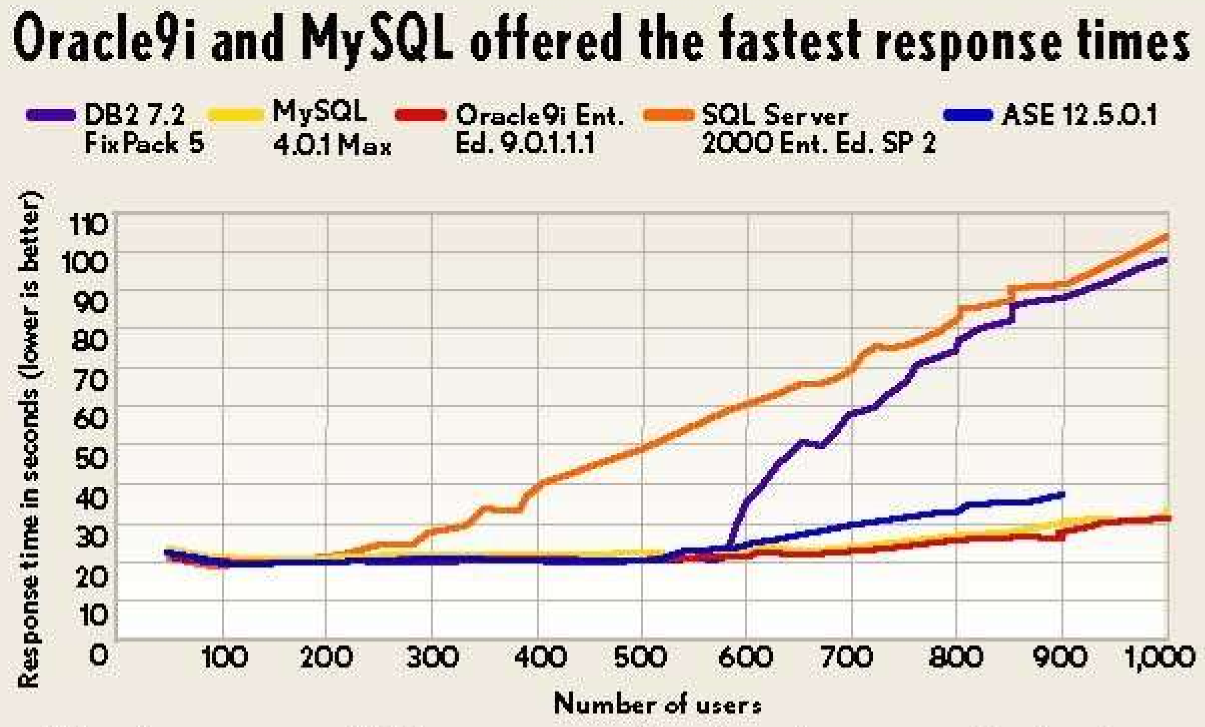
Oracle9i and MySQL top throughput

Throughput is in returned Web pages per second from the application server. Number of users is number of concurrent Web clients driving the load. Response time is the time to complete the six bookstore user action sequences, weighted by frequency of each sequence in the mix. All tests were conducted on an HP NetServer LT 6000r with four 700MHz Xeon CPUs, 2GB of RAM, a Gigabit Ethernet Intel Corp. Pro/1000 F Server Adapter and 24 9.1GB Ultra3 SCSI hard drives used for database storage.

- Done January 2002
- According to EWeek the first independent full scale test of RDBMS in 7 years.

MySQL (4.0), Oracle (9i), IBM DB2 (7.2), Microsoft SQL Server (2000 SP2) and Sybase (12.5) was tested

EWeek Latency



Oracle9i and MySQL offered the fastest response times

Throughput is in returned Web pages per second from the application server. Number of users is number of concurrent Web clients driving the load. Response time is the time to complete the six bookstore user action sequences, weighted by frequency of each sequence in the mix. All tests were conducted on an HP NetServer LT 6000r with four 700MHz Xeon CPUs, 2 GB of RAM, a Gigabit Ethernet Intel Corp. Pro/1000 F Server Adapter and 24 9.1GB Ultra3 SCSI hard drives used for database storage.

This shows the response times of the different databases.

It better to be at the bottom

MySQL is the yellow line under the Red Oracle line

The test was a virtual web bookstore with **50 Million** books. It used a few large Windows servers and about 50 clients machines. Source code to the tests have been published!

Citations from the EWeek article

- "The Oracle and MySQL drivers had the best combination of a complete JDBC feature set and stability."
- "Of the five databases we tested, only Oracle9i and MySQL were able to run our Nile application as originally written for 8 hours without problems."
- "SQL Server and MySQL were the easiest to tune, and Oracle9i was the most difficult because it has so many separate memory caches that can be adjusted."

Storage Engines

- A storage engine is a low-level data storage
- This allows choice locking and speed trade offs per table
- Main choices:
  - MyISAM Non-transaction, fast for R or W
  - InnoDB Fully transactional with row-level locking Multi-Versioning without any need for vacuum
  - Heap In-memory hash or b-tree
  - Cluster Multi machine synchronous clustering with fast fail over (1-2 sec). Release in April (at Conference)
- `CREATE TABLE(key int, value char(10), PRIMARY INDEX key)`**`TYPE=HEAP`**`;`

Storage Engine: MyISAM

- Static, dynamic and compressed (read-only) row formats but no transactions
- Text and compressed indexes
- Data and indexes in separate files
- Fast read/write performance but low r/w concurrency
- Extremely good concurrency in the slow select and insert at end case (logs)
- Int & External check and repair functions
- Especially useful for data warehousing,logging & web

Storage Engine: InnoDB

- Full transactions (ACID) with row level locking (state of the art transactions!)
- Consistent reads (Default isolation level repeatable-reads) using Oracle style Multi Version Concurrency Control (MVCC) without need for vacuum
- Better concurrency than MyISAM for read/write on the same table
- Uses table spaces instead of individual files
- Is used in under extreme load at users like Yahoo, Slashdot and Google

Storage Engine: Heap/InMemory

- Completely in Memory with very fast **hash** based indexing
- Useful for
  - Temporary tables
  - Lookup tables
- Bad for range queries in 4.0 (only hash)
  - Fixed by introducing btree indexes in 4.1
  - Per index choice of hash or b-tree

Tips and Tricks: Temporary tables

- Temporary tables
  - Stays around until you drop them or your thread dies
  - Can be in memory (`TYPE=heap`), on disk (`TYPE=MyISAM` or `TYPE=InnoDB`)
  - `CREATE TEMPORARY TABLE type=MyISAM` ... `SELECT a+b as isum,` ...
  -
    ```
    CREATE TEMPORARY TABLE sums
    (PRIMARY KEY (id),
     INDEX (total),
     comment TEXT)
    SELECT t1.id, SUM(t2.number) AS total FROM
    t1,t2 WHERE t1.id = t2.id GROUP BY t1.id;
    ```

Active Versions

- Much bigger developer team (3 x 2002 team) -> More parallel version trees (Merging with BitKeeper)
- 3.23 Old Production Release
  - Will be slowly phased out with only important bug fixes
- **4.0 Current Production Release**
  - Default recommendation. **Upgrading recommended**
- 4.1 Stabilizing Development Release
  - Binaries out, Recommended for new development
- 5.0 Development release
  - Binaries out. For checking out stored procedures

4.0: Multiple table DELETE

- Allows you to delete rows in multiple tables bases of conditions from multiple tables
- Like saying that you want to delete all rows found by a SELECT
- Syntax inspired by MS Access
- `DELETE t1,t2 FROM t1,t2,t3 WHERE t1.id=t2.id and t2.id=t3.id`

4.0: Dynamic Setting of Variables

- Change options on the fly (per thread and globally)
- `SET GLOBAL SORT_BUFFER=1*1024*1024`
  - This will set the sort buffer for all new connections
- `SET SESSION SORT_BUFFER=32*1024*1024`
  - Will set the same buffer but only for the current connection
- `SELECT @@GLOBAL.SORT_BUFFER`
  - Returns the default size of the variable

4.0: Query Cache

- Improves the speed of queries on read mostly tables. Many websites has some query that will run many times a second
- 2-10x speed improvement for web sites very common
- Does not change semantics at all!
- `SET @@GLOBAL.QUERY_CACHE_SIZE=128*1024*1024`
  - Turns the Query Cache on with 128 MB memory

4.0: Handler (Navigational) Interface

- Used when porting old database application with a navigational (direct ISAM) interface
- Also very useful when you are doing a user interface that navigates a lot of data
- This interface gives "Dirty" reads (but no other MySQL commands are affected!)
- `HANDLER t_name OPEN;`
- `HANDLER t_name READ i_name > ("Banana") LIMIT 30;`
- `HANDLER t_name READ i_name PREVIOUS;`

4.0: Boolean Full Text Search

- A second way to do full text search in addition to the natural language search.

-
  ```
  SELECT * FROM articles WHERE MATCH (title,body)
  AGAINST ('+apples -bananas' IN BOOLEAN MODE);
  ```
  - Find all records with the word "apples" but not the word "bananas" using a full text index
- Faster bulk updates of Full Text Indexes
- Only for MyISAM (no InnoDB Full Text indexes yet)

4.0: Other Features

- `SELECT SQL_CALC_FOUND_ROWS ... LIMIT 10`
- `SELECT FOUND_ROWS();`
  - Make it possible to display the total number of rows when you have a `LIMIT` clause.
  - So a web search box result can show "10 out of 124 items shown" with out doing the query twice

-
  ```
  GRANT .. MAX_QUERIES_PER_HOUR=#
           MAX_UPDATES_PER_HOUR=#
           MAX_CONNECTIONS_PER_HOUR=#
  ```
  - Limit a user (Very useful for ISP use)

4.0: Some smaller features

- Faster `SELECT COUNT(DISTINCT ...) ...`
- Faster bulk loading of data
- A fast turbo Boyer-More string search for LIKE queries
- Hash function: `SELECT SHA1("foo")`
  - Use instead of `PASSWORD()`for your own uses (since `PASSWORD()` will change!)
- Removing all the rows in a table
  - `TRUNCATE TABLE table_name;`
- `CREATE TABLE foo DATA DIRECTORY="/path/to/dir" INDEX DIRECTORY="/path/to/dir"`
- Multi threaded index rebuilding

4.1: Sub Queries

- `SELECT row1 FROM table1 WHERE a=(SELECT b FROM table2)`
- `SELECT ... WHERE (a,b) = (SELECT c,d FROM ...)`
- `SELECT ... WHERE EXISTS (SELECT * FROM...)`
- `SELECT...WHERE a IN (select b from table t1)`
- `SELECT...WHERE a > ANY (select b from table t1)`
  - Sub-Queries

4.1: GIS Support

- Multidimensional data support (OpenGIS)
  - 2d and 3d data
  - New point, line, multipoint, polygon, etc datatypes
- `CREATE TABLE g1 (p1 GEOMETRY);`
- `ALTER TABLE g1 ADD p2 POINT;`
- `CREATE TABLE g (g GEOMETRY NOT NULL, SPATIAL INDEX(g));`
- `SELECT fid,AsText(g) FROM g WHERE MBRContains (GeomFromText('Polygon((30000 15000,31000 15000,31000 16000,30000 16000,30000 15000))'), g);`
- `20 rows in set (0.00 sec) [Without index (0.46 sec)]`

4.1: Character Set Support

- Follows the ANSI SQL Standard
- Makes it possible to have columns in different languages and get
  - Correct sorting & String compare
- This is done by defining columns in a `CHARACTER SET` and/or using `COLLATE` statements
- UNICODE is supported (ucs2 and utf8)
- Latin-7 Support for Latvian (but no local error messages)
- Set defaults per Server, Database and Table

4.1: Usage Examples

```
•CREATE TABLE T1 (
a CHAR(10) CHARACTER SET ujis,
b CHAR(5)  CHARACTER SET latin1 COLLATE
latin1_swedish_ci );

•SELECT * FROM T1 WHERE
       a = 'タナカ';

•SELECT * FROM T1 WHERE
       a = _sjis 'すずき' ORDER BY a
COLLATE sjis;

•SELECT * FROM T1 WHERE
        b = 'bar' ORDER BY a;
```

4.1: Prepared Statements

- Used to issue the same query many times with new data
- Done in 3 steps
  - Prepare a query by doing a PREPARE on the query with place holders (“?”)
    - Set the place holders to values
    - Execute the query
    - And repeat the last 2 steps any number of times witch new values for the placeholders

4.1: JDBC Prepared Statement Code

```
PreparedStatement pstmt =
  conn.prepareStatement(“INSERT INTO foo
  VALUES (?, ?)”);

# Repeated part
pstmt.setDouble(1, 12345.678);
pstmt.setInt(2, 320);
int updateCount = pstmt.executeUpdate();
# End of repeated part

pstmt.close();
```

4.1: Smaller features I

- `UPDATE t1,t2,t3 SET t1.c1=Val,t2.c2=val2  WHERE t1.id=t2.id and t2.id=t3.id`
  - Multiple table `UPDATE`
- `SELECT * FROM table1, (SELECT b from table2) WHERE ...`
  - Inline views / Derived tables

4.1: Smaller Features II

- `INSERT INTO tname SET key=1, a=1 ON DUPLICATE KEY a=a+1;`
  - Insert that is changed to an update if the data already exists
- `LOAD INDEX ... INTO CACHE`
  - Pre-loading of MyISAM index data into the cache
- `SET QUERY_TIMEOUT=5`
  - Query timeouts (or time limits). Syntax may still change
  - Implemented with one extra watcher thread

4.1: Smaller features III

- Secure MySQL client-server connections with SSL
- Replication using SSL for safe replication over a public (Internet) connection
- `SHOW WARNINGS` to get warnings for previous statement (like after a `LOAD DATA INFILE`)
- New functions: `UPDATE ... a=COMPRESS(data); SELECT UNCOMPRESS(data), UNCOMPRESSED_LENGTH(data)`
  - Uses zlib (gzip) library

4.1: Smaller features IV

- Online help for SQL syntax in the command line client
- `CONVERT()` can convert between character sets
- `INSERT INTO customer VALUES (1,2);` `INSERT INTO sale VALUES (LAST_INSERT_ID(),2);`
  - Multiple commands in a single statement. Turned off by default for safety

5.0: Stored procedures

- Stored Procedures
  - First version with ANSI-99 PSM Syntax (they changed the name to “Persistent Stored Modules”)
- We plan to do a interface for external languages
  - First supported example language will probably be Perl or PHP. Later also Java
  - We will provide the tools to support other languages like Python, Ruby
  - We plan to support both one and two processes
- The PSM version works and and binaries are available.
- FYI: IBM DB2 also uses this syntax

5.0: PSM Stored Procedure Example

- CREATE PROCEDURE withdraw(
  - p_amount DECIMAL(6,2),
  - p_tellerid INTEGER,
  - p_custid INTEGER)
- MODIFIES SQL DATA
- BEGIN ATOMIC
  - UPDATE customers SET balance=balance - p_amount;
  - UPDATE tellers
    - SET cashonhand=cashonhand - p_amount
    - WHERE tellerid = p_tellerid;
  - INSERT INTO transactions
    - VALUES (p_custid, p_tellerid, p_amount);
- END

5:0: Smaller features

- Referential Integrity/Foreign keys for MyISAM (already available for InnoDB)
- On line backup of MyISAM tables
- New column types:
  - `BIT`
  - True VARCHAR (no space trimming)
  - ARRAY
- Extend the warning/info system
  - Notify sysadmin when something (may) goes wrong
  - Give feedback while a slow command is running

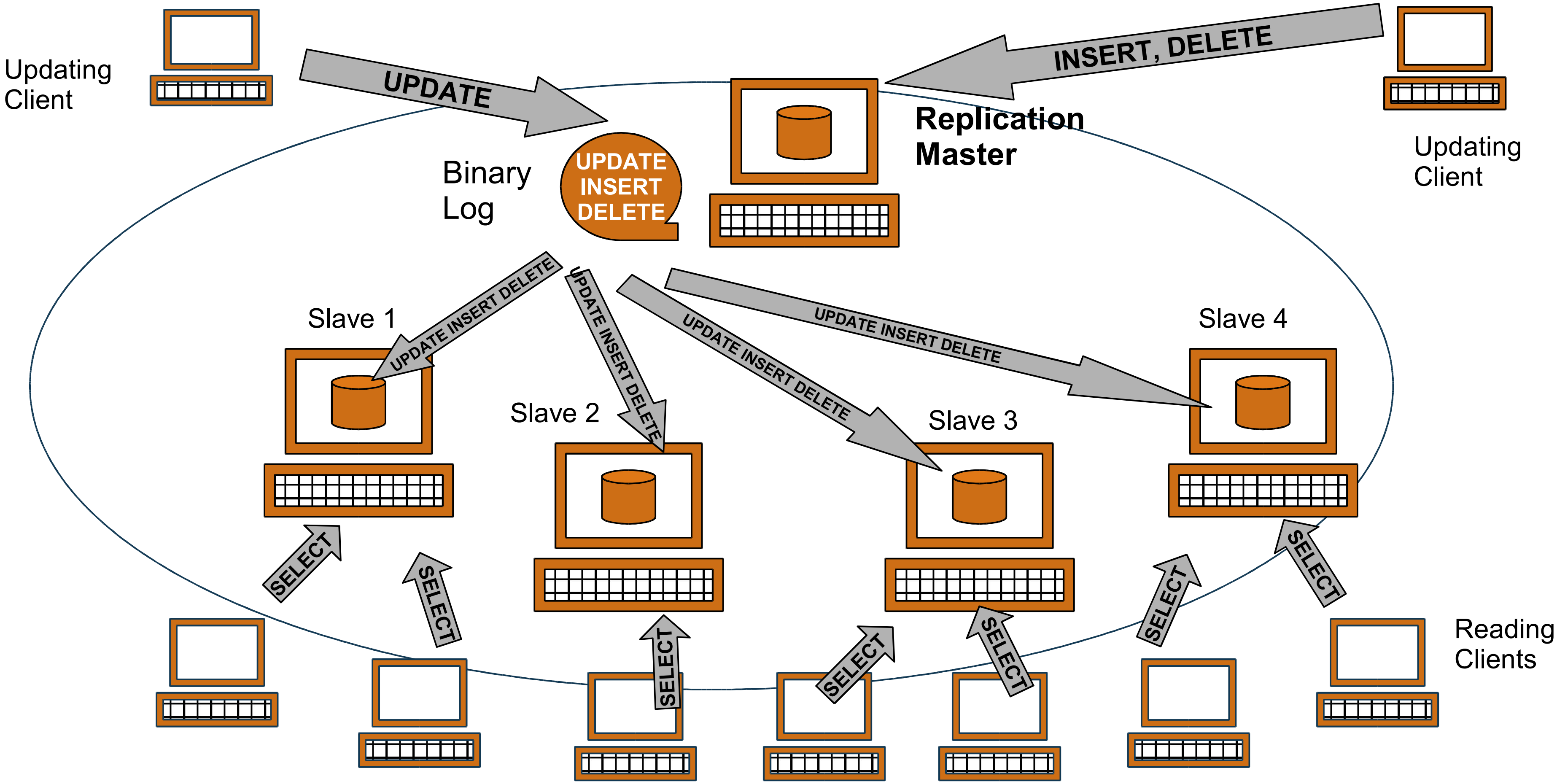
Why Use Replication

- Scale Read-Type Queries
  - Allows you to scale the amount of **`SELECT`** queries able to be performed
- Redundancy
  - Allows for fail-over server in case of crash on the master
  - Needed for true 24/7 capabilities
- Backup
  - Allows for the ability to take backups without affecting the operations of the master
  - Especially useful with MyISAM tables until hot backup is implemented in MySQL 5.0

Current Users of Replication

- Yahoo!
  - Large scale replication across multiple datacenters worldwide
- Mobile.de
  - Over 60+ servers in active replication clusters
  - Processes 20k+ queries per second
  - Largest online Autodealer in Germany
- Slashdot.org
  - Popular Open Source news site
  - Very high traffic with mostly dynamic content
- Google
  - Used in many internal systems

Replication Overview



Creators of MySQL

Replication Details

- MySQL supports **one-way** replication internally
  - Single server designated as master
  - Many servers can act as slaves to single master
  - Slaves can also be the master for other slaves
  - Currently each slave can only have a single master
  - Replication is done **asynchronously**
    - Master doesn’t wait for slave to confirm changes

- **All** updates must be done on the **Master** server
  - Otherwise the master and slave will get **out of sync**, which effectively breaks the replication process

How Does It Really Work

- Master server keeps track of all updates in Binary Update Log
  - Slave requests to read the binary update log
  - Master acts in a **passive** role, not keeping track of what slave has read what data

2. Upon **connecting** the slaves do the following:
   - The slave **informs** the master of where it left off
   - It then **catches up** on the updates
   - And then **waits** for the master to notify it of new updates

Replication "Breaking"

- Replication will "break"
  - If master and slave gets out of sync (disc full etc)
  - If updates to data occur on the slave
    - There is a state now where only a special user can write
  - Noticable in **`SHOW SLAVE STATUS`** command
  - Check error log for disconnections and issues with replication
- How to fix it
  - Resync slave with master manually
  - **`SET GLOBAL SQL_SLAVE_SKIP_COUNTER=1`**
    - Slave will skip next event in binlog

Different Topographies

- Replication in MySQL can be used in many different layouts
  - Normal tree layout (detailed previously)
  - Single slave
  - Relay Slave layout
  - Circular Replication

Two-way Replication

- MySQL does **not** support two way internal replication on the same data
- As long as you replicate separate databases or tables then you are safe
- Replicating the same table data both ways will lead to race conditions
  - Auto_increment, unique keys, etc.. could cause problems
- You don’t gain many advantages with two-way replication so it is highly recommended you use one-way

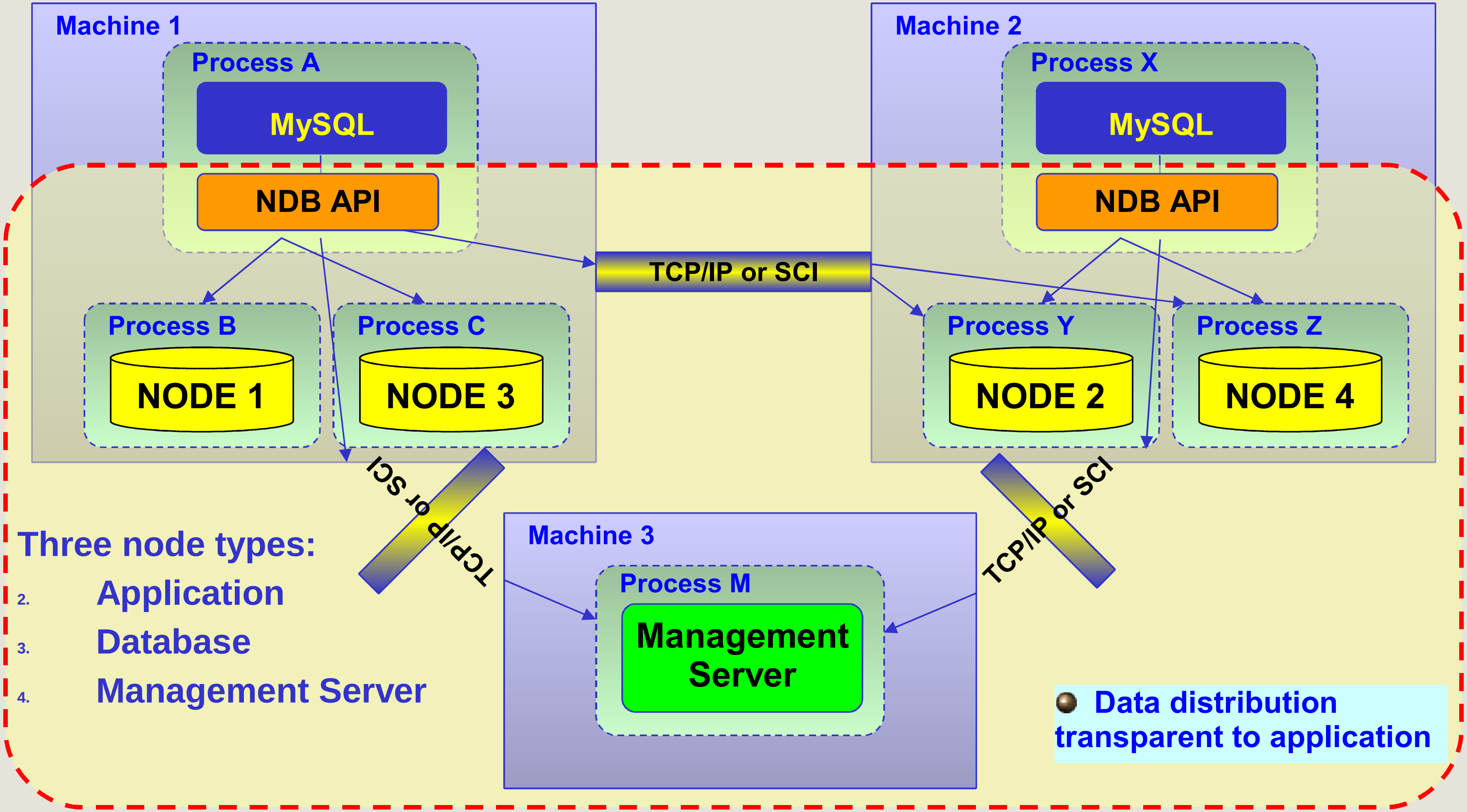
Features Upcoming

- Multimaster replication
  - Slave can have more than a single master
  - Useful for creating a data warehouse from many servers
- Two-way replication
  - True master-master replication
  - Two-phase commit

Storage Engine: MySQL Cluster

- We acquired a cluster database company called Alzato last Summer
- This technology will provide true clustering for MySQL through a special storage engine that connects to a cluster of machines.
- In memory database with fast automatic failover (complete failover in 1-2 sec)
- Full transaction support
- Needs about 2xdata memory in whole cluster
- Good scalability

Example Configuration: Process Diagram

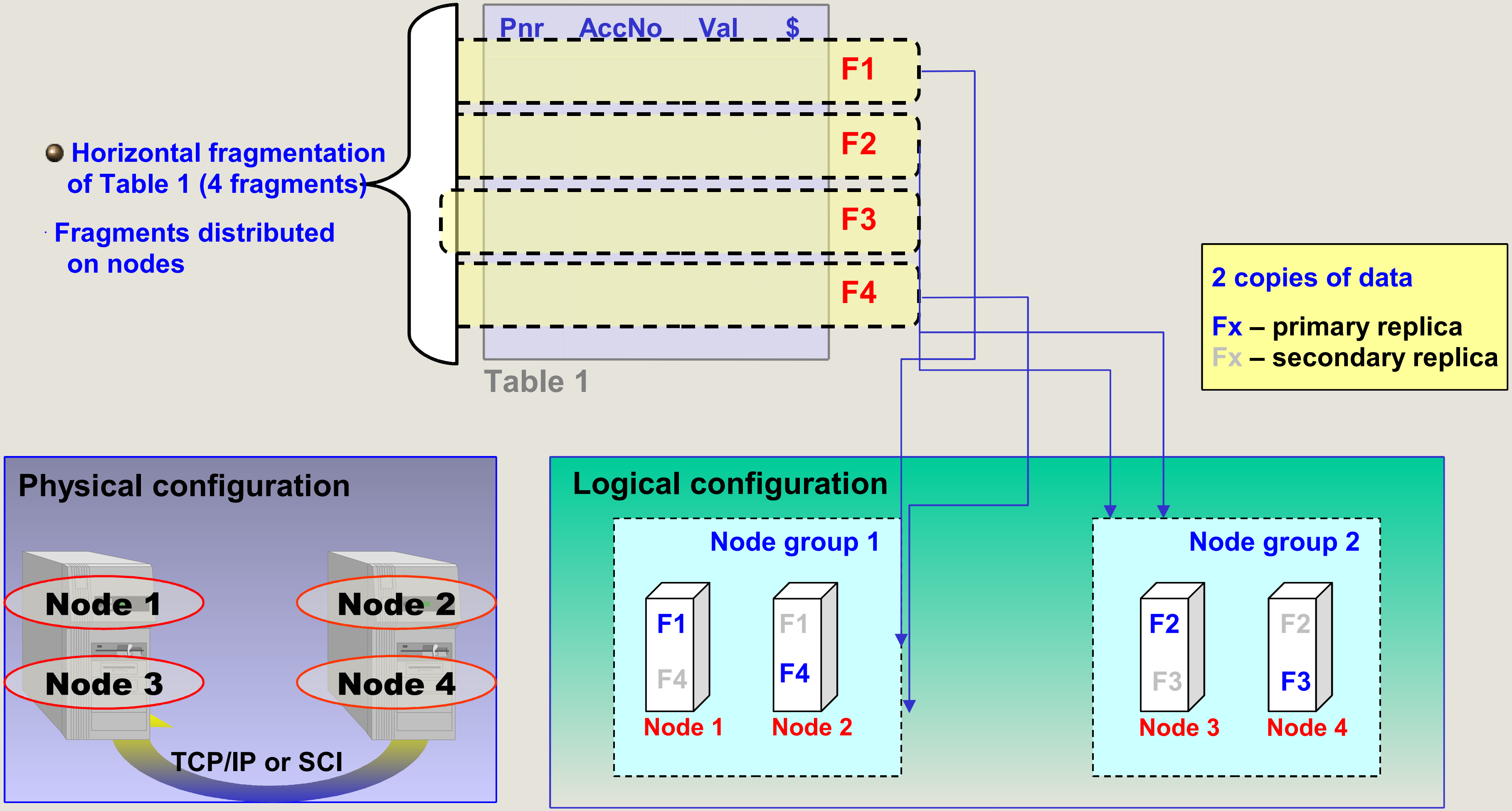


Three node types:

2. Application
3. Database
4. Management Server

- Data distribution transparent to application

Data distribution



- Horizontal fragmentation of Table 1 (4 fragments)
- Fragments distributed on nodes

2 copies of data

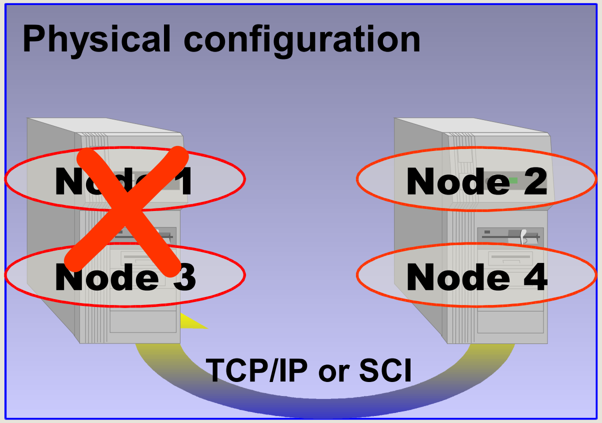
Fx – primary replica
Fx – secondary replica

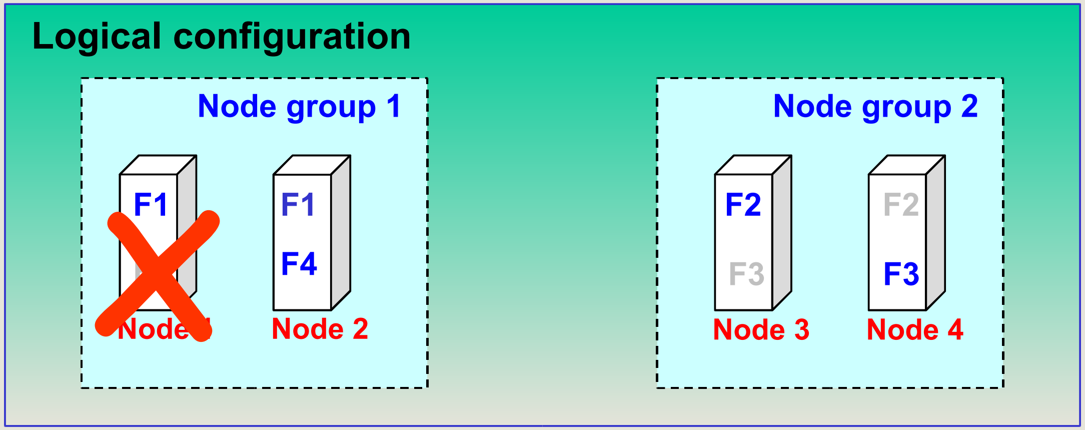
NDB Cluster : 4 node configuration on 2 dual processor machines

Single Node Failure

What happens if *node 1* fails?

- **Detection of node failure. Primary fragment F1 handled by node 2**
  - Switchover times dependent on underlying OS, e.g., OSE Delta (10s of milliseconds), Solaris & Linux (100 ms – 10s of seconds depending on configuration)
- **Automatic restart and recovery of node 1**
  - Node 1 recovers F1 and F4 from node 2 and rejoins the cluster.

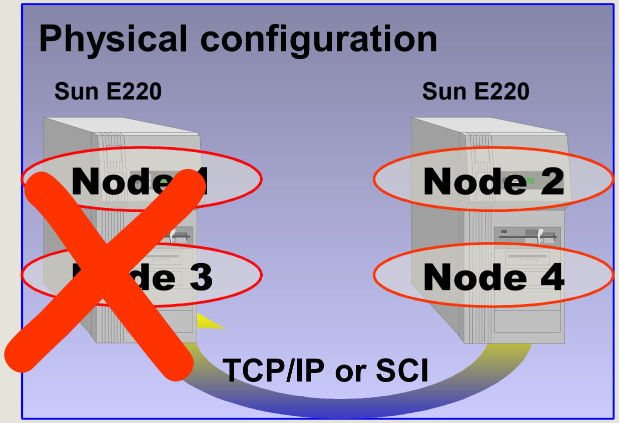


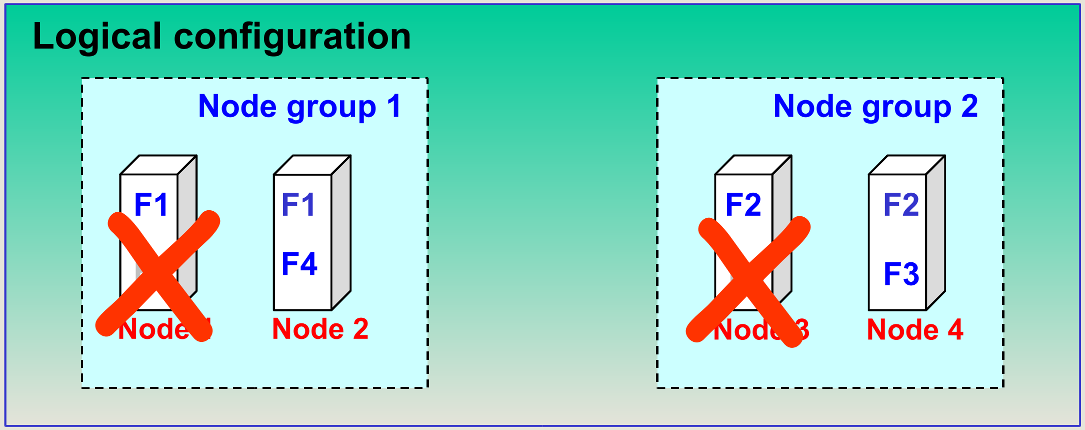


Multiple Node Failure

What happens if *machine 1* fails?

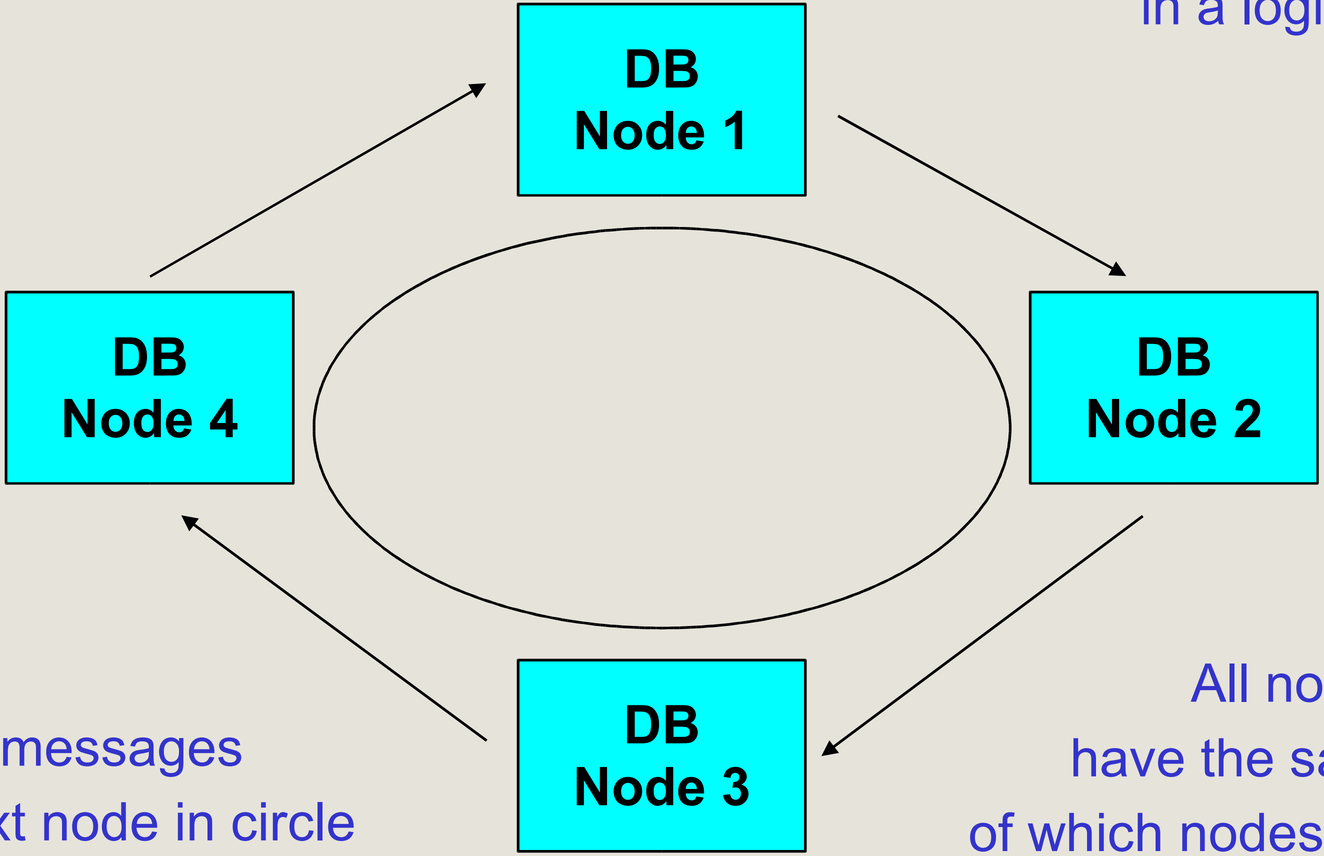
- **Detection of node failure. Primary fragment F1 handled by node 2, primary fragment F2 by node 4**
  - Database available as long as at least one node is alive per node group
- **Automatic restart and recovery of node 1 and node 3**
  - During the recovery process, there is degradation in performance of the cluster





Failure detection: Heartbeats, lost connections

Nodes organized
in a logical circle



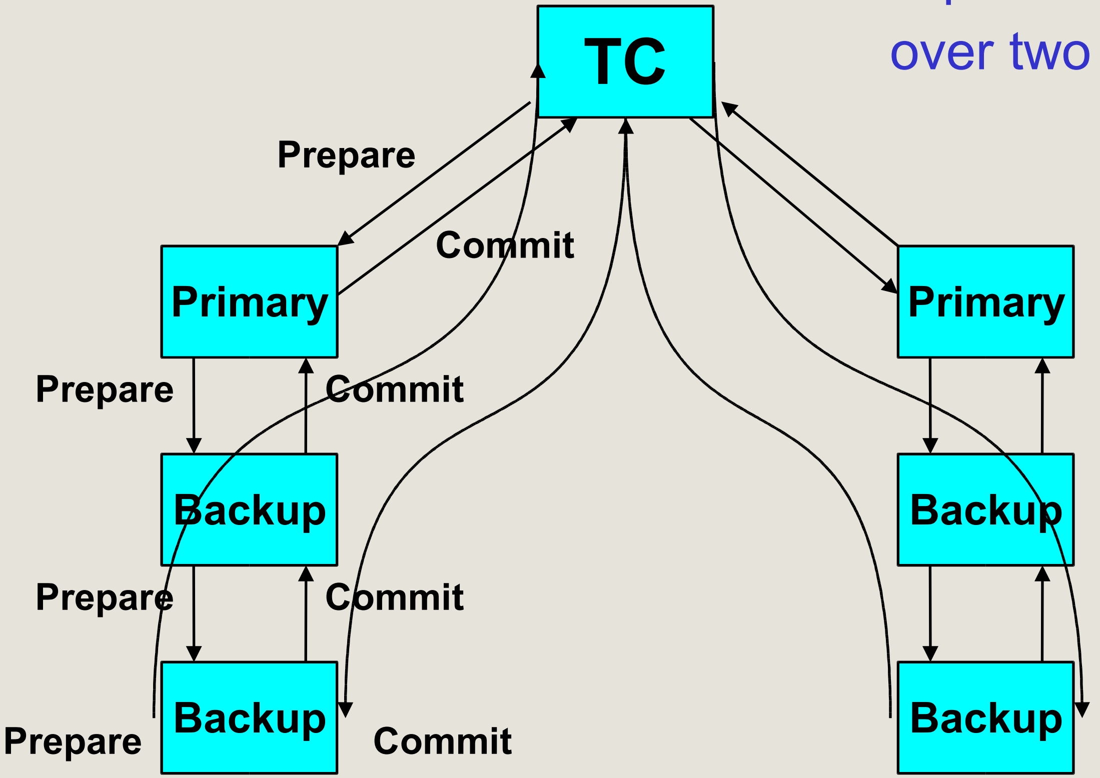
Heartbeat messages
sent to next node in circle

All nodes must
have the same view
of which nodes are alive

Synchronous Replication: Low failover time

messages = 2 x fragments x (replicas + 1)

Example showing transaction over two fragments using three replicas

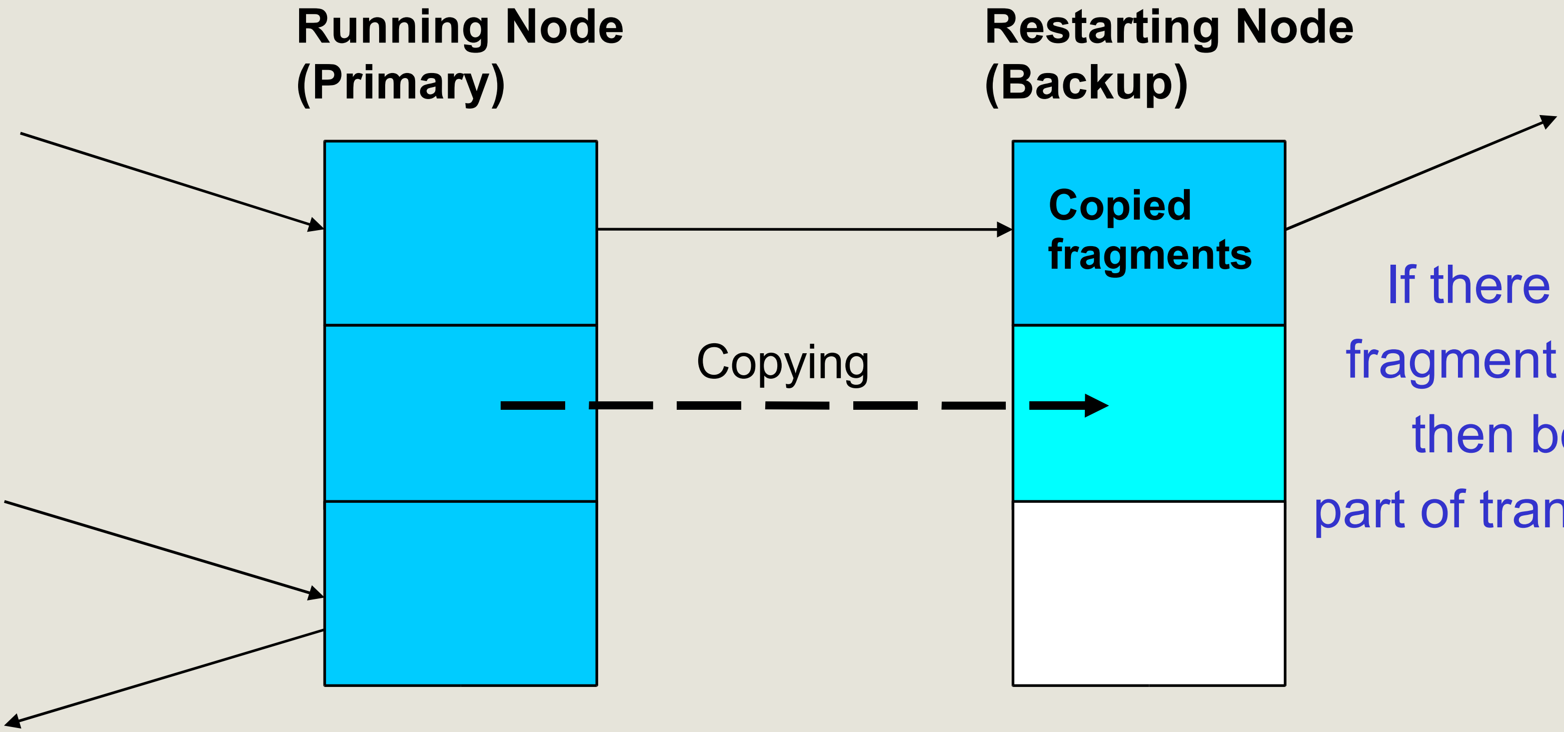


1. Prepare F1 2. Commit F1 1. Prepare F2 2. Commit F2

Single Node Recovery

TC coordinates fragment operations

Example with two replicas



If there are two fragment copies, then both are part of transaction

Update, ***insert***, and ***delete***

Read operations only read from one replica

Development Road Map

- Implement what is needed to run SAP R/3
  - That means almost all of standard SQL
  - We are working with SAP experts. Both application and and database developers (SAP-DB).
- Implement the core parts of the SQL-2003 standard
  - Includes some XML support
- Do useful nonstandard extensions
  - Data warehousing extensions
  - Fixes for huge loads (like preloading of indexes and the query cache)

Windows: ODBC and .Net

- MySQL is actively developed and tested on Windows
  - MySQL has a true high performance native Windows port
- MyODBC and .Net drivers active developed by MySQL AB
- ODBC level 3.5 supported
  - Transaction support
  - Cursor support
  - Lots of small annoying long lived quirks fixed

Java: JDBC

- MySQL has a very high quality JDBC driver
- Developed by MySQL AB with
- Our main developer is now on the JDBC experts group that develops the next version of the standard.
- [Thanks to Java since MySQL would not get a fixed Linux Kernel scheduler without them. MySQL also likes to run with many threads!]

MySQL Benchmarks

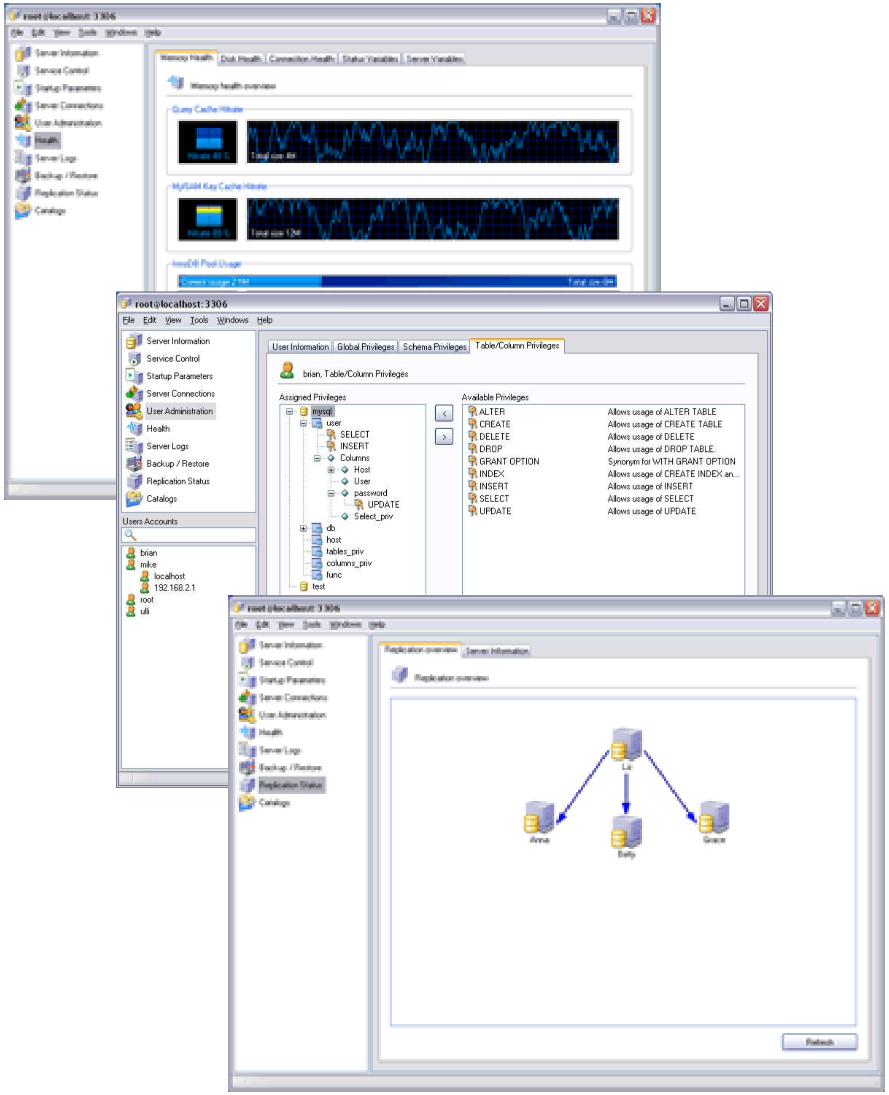
- Done to find speed of individual operations
  - Written in Perl using DBI for database independence
  - Single thread and small data size
  - We used it to compare MySQL with MySQL and also other databases
  - Shows some of the architecture choices of different databases
- Very good for the above but horrible as a generic benchmark.
- Currently no results on MySQL.com since they are to much out of date. New results will be published soon.

Crash-Me database test

- Written to find supported features in a Database from a known list.
- Does try to find limits by repeating “larger and larger” queries until something go wrong
- Tests for things that is very hard to find in manuals
  - Maximum size of: joins, strings, column and other names, number/depth of expressions.
  - Also which standard functions/types exists.
- New updated version is published on MySQL.com
- Crash-Me does test for known **syntax not features**
  - If a feature is done with non standard syntax we miss it

MySQL Administrator

- Easy, graphical management
- Replaces MySQL-CC
- Available Q2 2004



ALTER	Allows usage of ALTER TABLE
CREATE	Allows usage of CREATE TABLE
DELETE	Allows usage of DELETE
DROP	Allows usage of DROP TABLE.
GRANT OPTION	Synonym for WITH GRANT OPTION
INDEX	Allows usage of CREATE INDEX an...
INSERT	Allows usage of INSERT
SELECT	Allows usage of SELECT
UPDATE	Allows usage of UPDATE

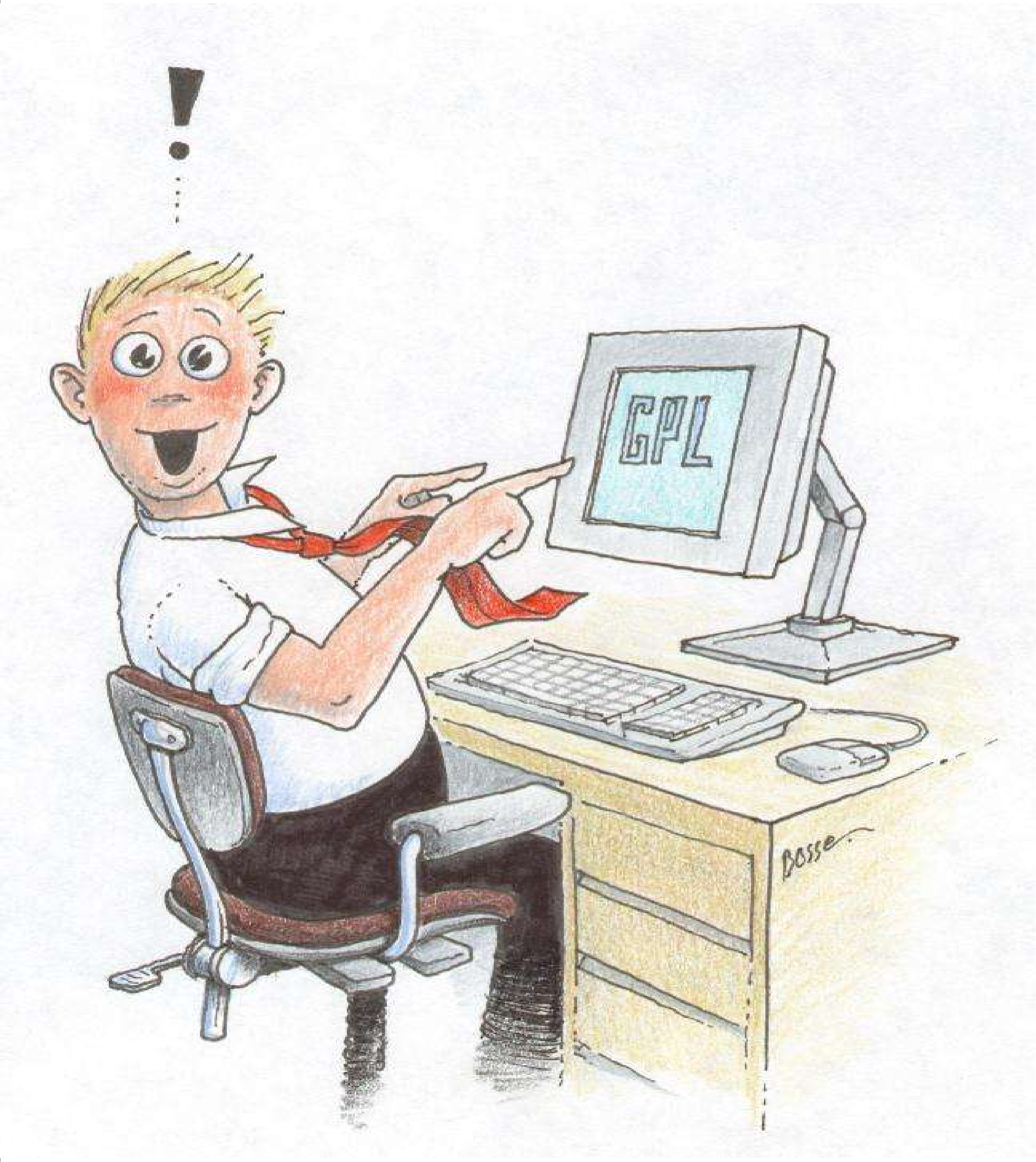
Free Databases are/get Better I

- Freedom & Independence!
  - You have the ultimate documentation, the source! You do not have to depend on limited documentation from any vendor to learn
- People invest more time on good bug reports when the possibility of a quick fix exits. **Repeatable bug reports are worth as much as code!**
- A lot of testing since you get more users even for advanced features (Replication, Full text search and Geographical Data are some examples)
- We can hire people who already know the code

Work at MySQL?

- We are always looking for experienced C/C++ developers. The MySQL server is written in C with some (just a few) features from C++
- If you want to apply send a email with a CV to jobs@mysql.com
- And if you want us to notice it do spend time looking at the MySQL server code before!

Why OpenSource is Better

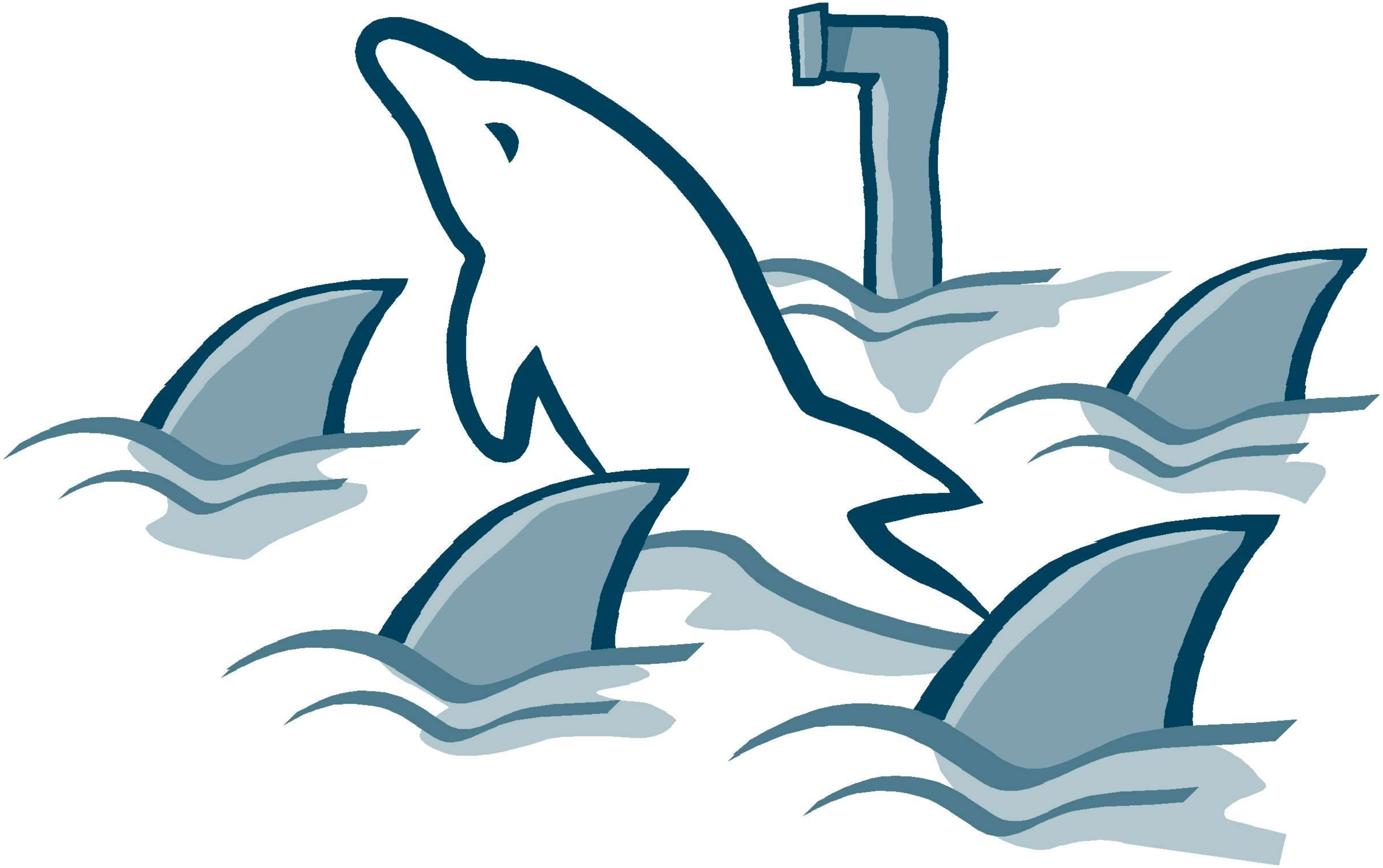


- Money stays in the local economy (instead of being payed abroad)
- Lots of code gets written by others (for MySQL especially in the database client & tools area)
- Security is not by obscurity, No hidden hooks in the code
- Result: Low Total Cost of Ownership

No Software Patents!

- **MySQL AB is officially against Software Patents**
- Software Patents are a threat against Free Software!
- I did talk on in the beginning of May at the European Parliament Brussels at a conference against Software Patents
- I hope **you** will work against Software Patents also!

Do Differently



Questions!

Now it is your turn to ask **questions!**

For more information about MySQL see:

MySQL.com