



**Firebird
Present Futur
July 2008**

Development

★ Engine Development

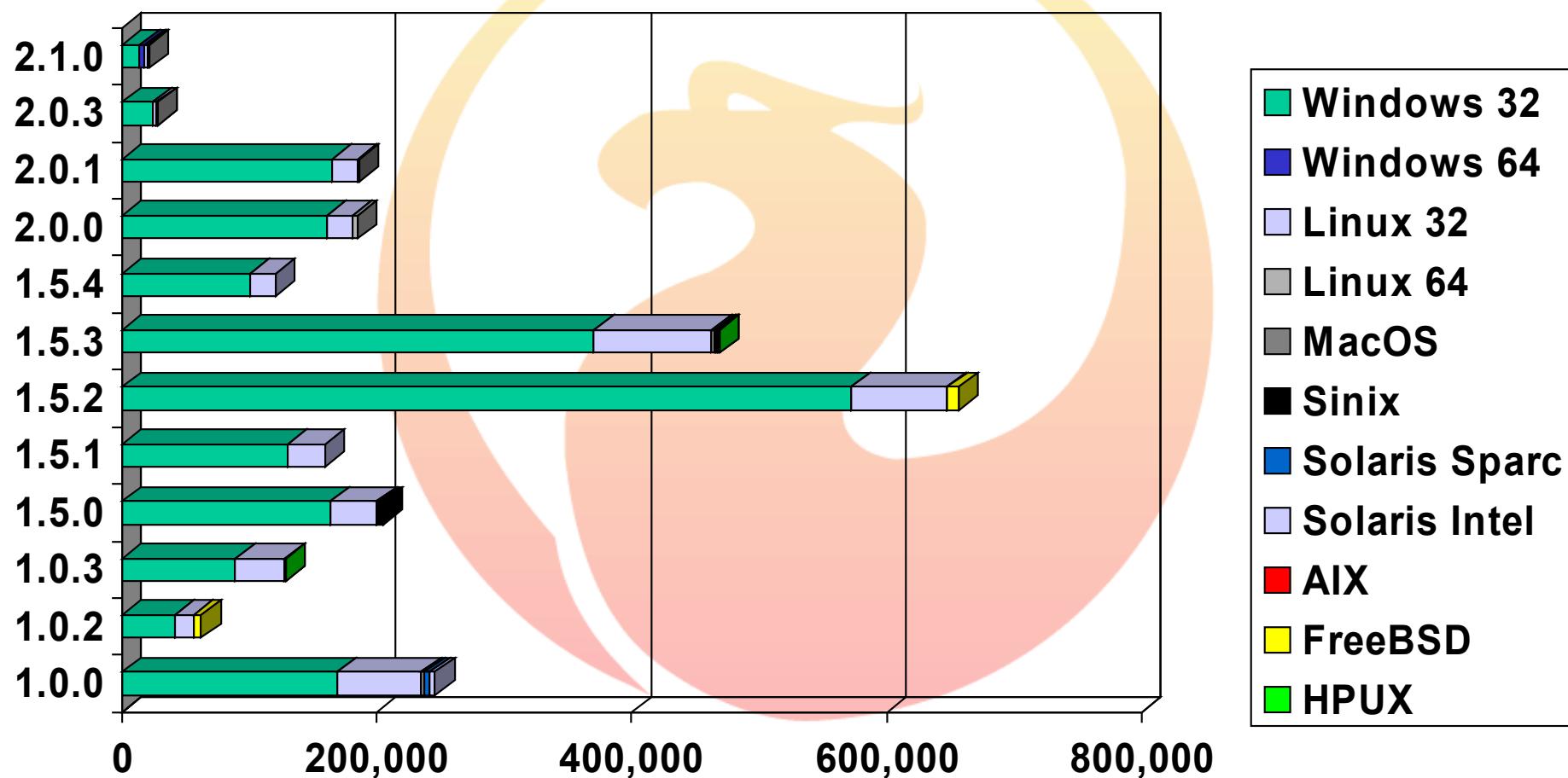
Firebird 2.0.4 third bugfix sub-release

Firebird 2.1.1 released.

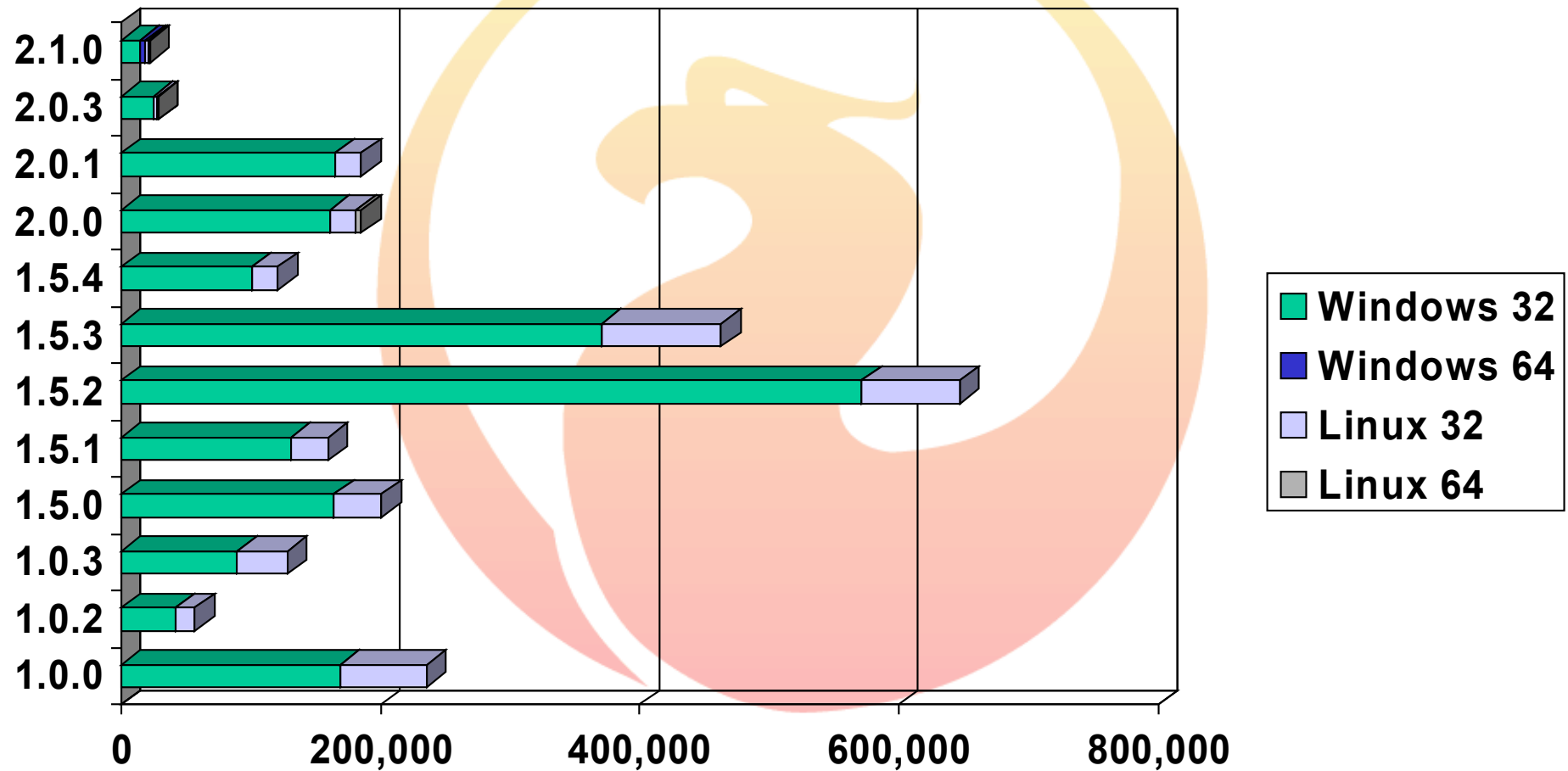
Firebird 2.5 is now alpha

Firebird 1.5.5 was also released

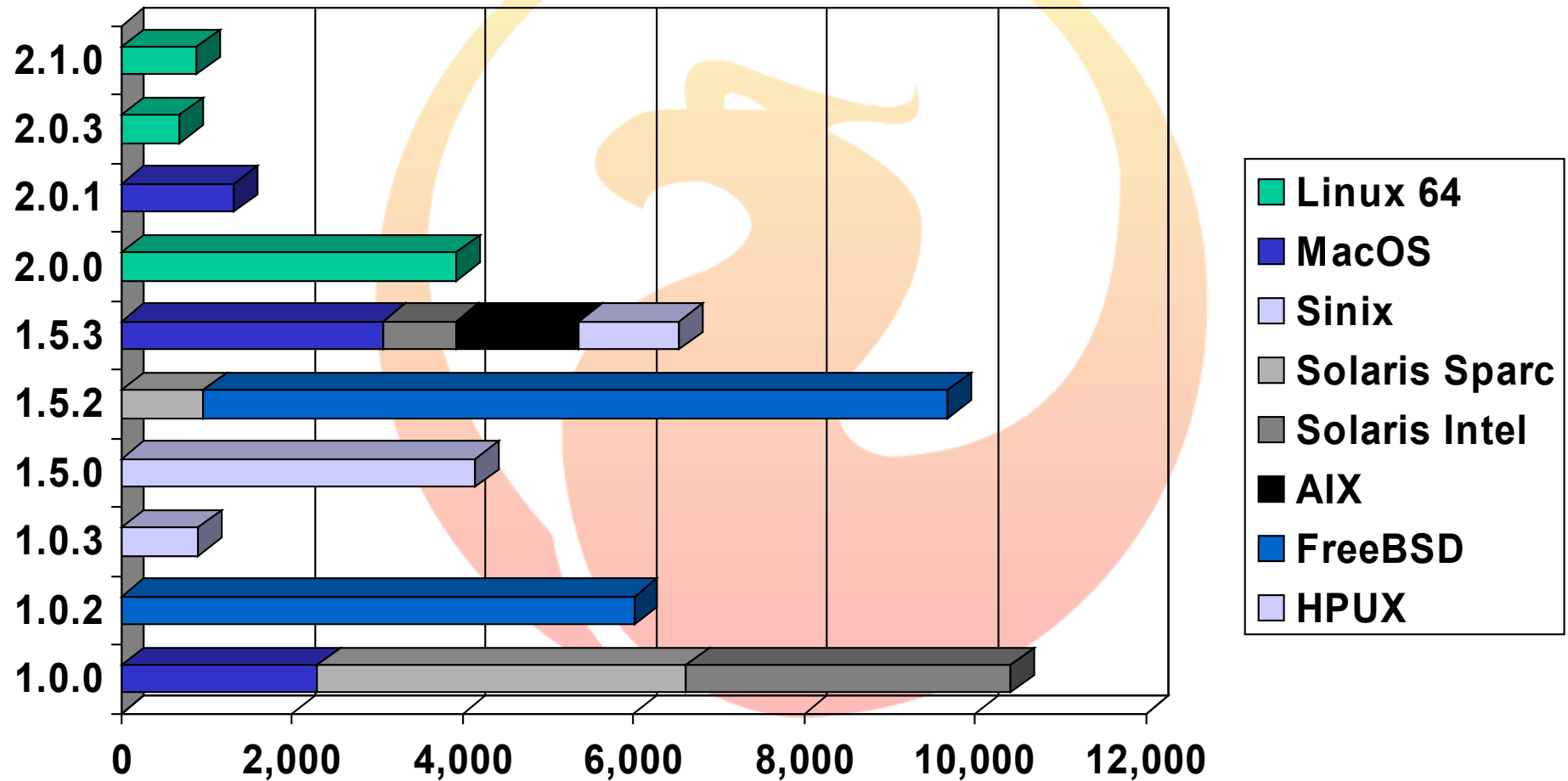
All Downloads



Main Downloads



Other Platforms

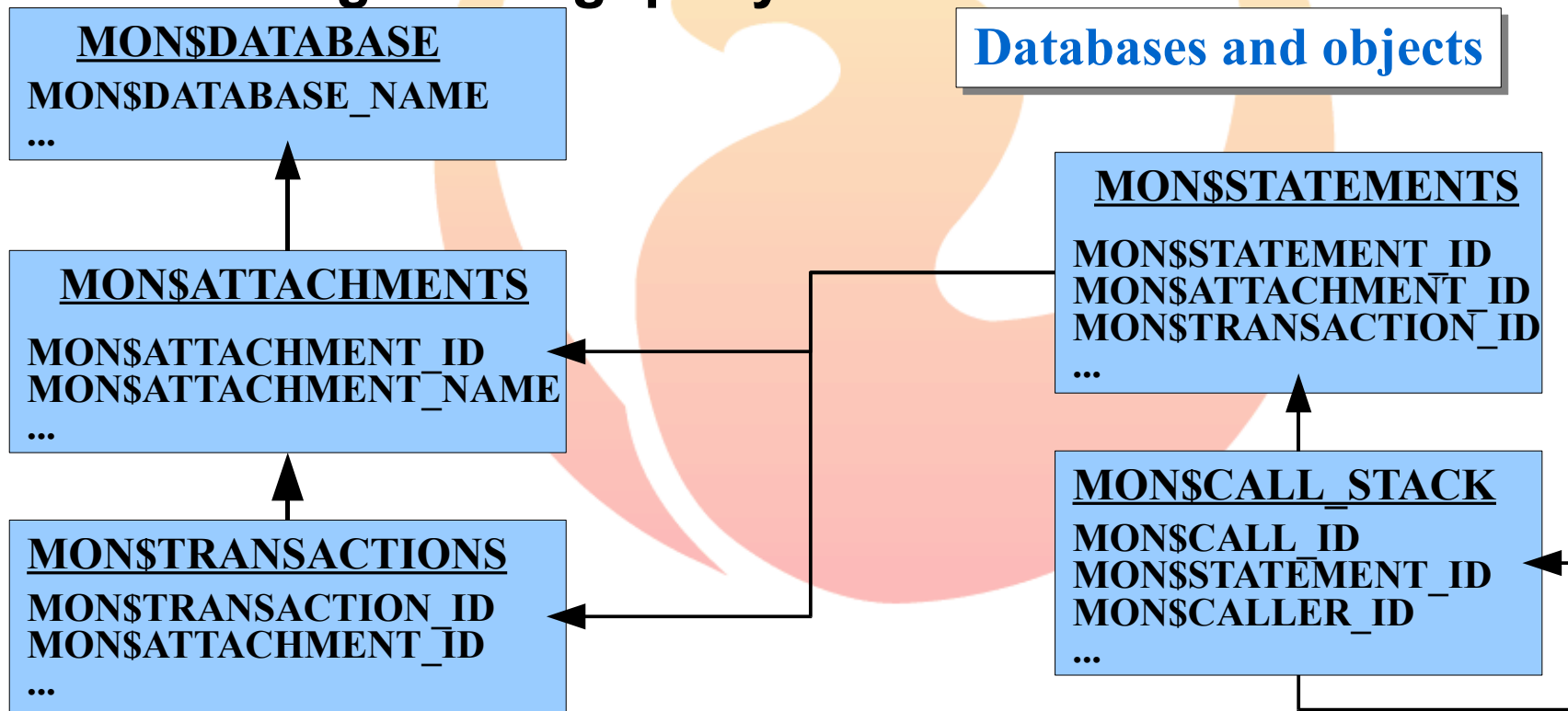


V2.1

- **Global temporary tables**
- **Common table expressions**
- **Monitoring tables**
- **Database triggers**
- **New built-in functions**
- **Win64 build**
- **Trusted Windows authentication**
- **Improved performance (network, sort, ...)**
- **Domains everywhere**
- **Insert or update, merge, list**

Monitoring tables

- Activity snapshot retained till the end of transaction
- Works with both Super and Classic Server
- Regular users restricted to see own attachment
- Kill long running query



Database triggers

- **Database-wide triggers fired on :**
 - CONNECT
 - DISCONNECT
 - TRANSACTION START
 - TRANSACTION COMMIT
 - TRANSACTION ROLLBACK
- **Only SYSDBA or database owner can :**
 - define database triggers
 - switch it off for new connection by :
 - new **isc_dpb_no_db_triggers** tag
 - new **-no_dbtriggers** switch in utilities

Database triggers example

Example of ON CONNECT trigger

```
isql temp.fdb -user SYSDBA -pass masterkey
Database: temp.fdb, User: SYSDBA
SQL> SET TERM ^ ;
SQL> CREATE EXCEPTION EX_CONNECT 'Forbidden !' ^
SQL> CREATE OR ALTER TRIGGER TRG_CONN ON CONNECT
CON> AS
CON> BEGIN
CON>   IF (USER != 'GOOD_USER')
CON>   THEN EXCEPTION EX_CONNECT USER || ' not allowed !';
CON> END ^
SQL> EXIT ^
```

```
isql temp.fdb -user BAD_USER -pass ...
Statement failed, SQLCODE = -836
exception 217
-EX_CONNECT
-BAD_USER not allowed !
-At trigger 'TRG_CONN' line: 5, col: 3
Use CONNECT or CREATE DATABASE to specify a database
SQL> EXIT;
```

Database triggers example

Help ! My ON CONNECT trigger deny all users and even SYSDBA ! Nobody can log in anymore :(What can i do ???

Don't panic :) Use -no_dbtriggers switch in utilities :

```
isql temp.fdb -user SYSDBA -pass masterkey -nodbtriggers
Database: temp.fdb, User: SYSDBA
SQL> ALTER TRIGGER TRG_CONN INACTIVE;
SQL> EXIT;
```

Global temporary tables

- **Standard properties**
 - Common metadata and private data
 - Indices, triggers, uniqueness check and referential integrity
 - Lifetime : attachment or transaction
 - DELETE triggers are not fired on cleanup
- **Firebird's specific**
 - No problem with garbage collection
 - Storage is in temporary files, forced writes off
 - Instant cleanup

Global temporary tables

- **Syntax**

- **CREATE GLOBAL TEMPORARY TABLE**

...

- **[ON COMMIT <DELETE | PRESERVE> ROWS]**

- **ON COMMIT DELETE ROWS**

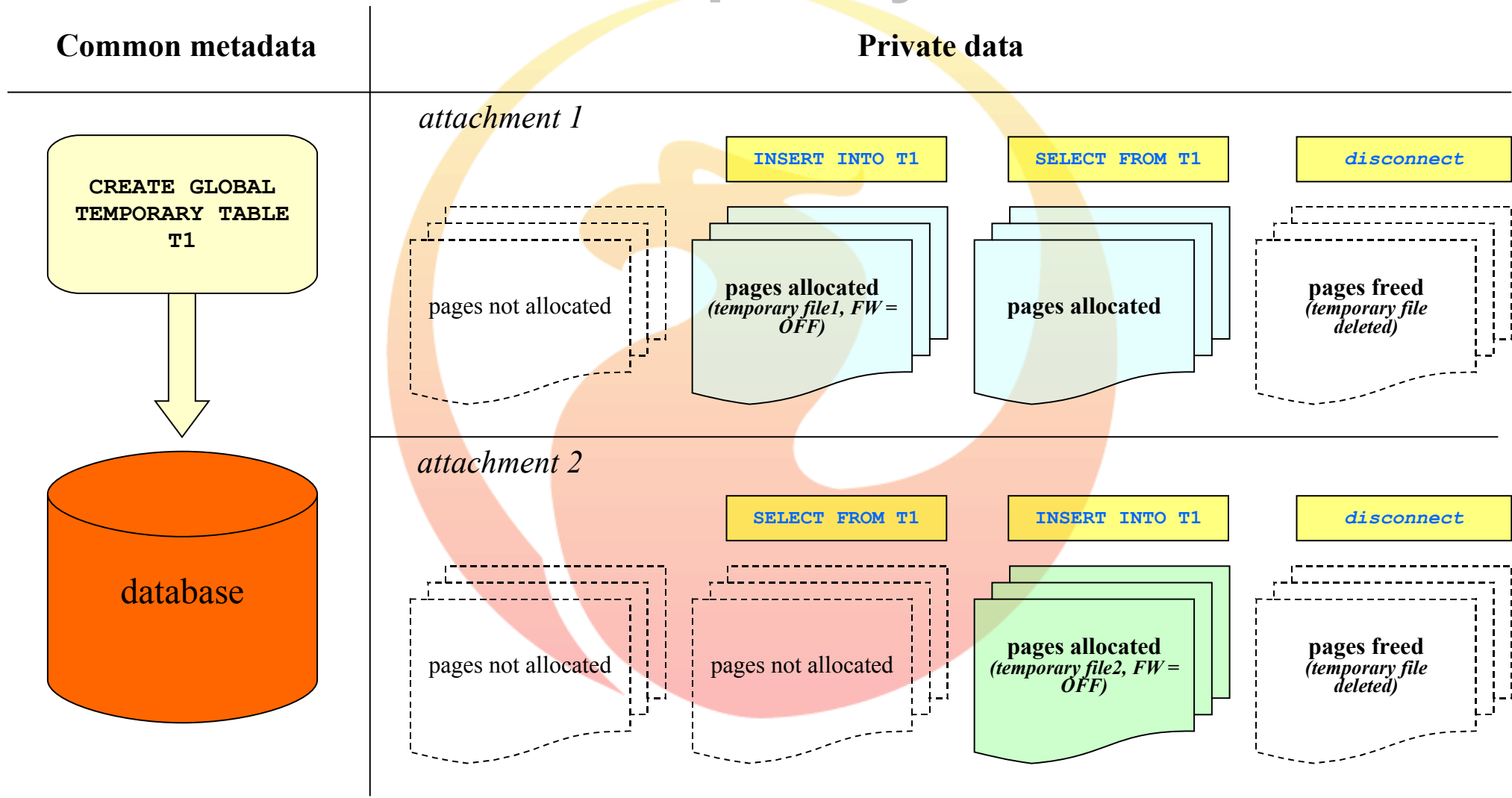
data lifetime - transaction

- **ON COMMIT PRESERVE ROWS**

data lifetime - attachment

- **By default - DELETE ROWS**

Global temporary tables



Common table expressions

• Syntax

- **WITH [RECURSIVE]** *-- new keywords*
 - **CTE_A** *-- first table expression's name*
 [(a1, a2, ...)] *-- fields aliases, optional*
 AS (SELECT ...), *-- table expression's definition*
 - **CTE_B** *-- second table expression*
 [(b1, b2, ...)]
 AS (SELECT ...),
 - ...
- **SELECT ...** *-- main query, used both*
FROM CTE_A, CTE_B, *-- table expressions*
TAB1, TAB2 *-- and regular tables*
WHERE ...

Common table expressions

Example quartile (non recursive)

```
SET TERM !!;
EXECUTE BLOCK
AS
DECLARE VMESURE DOUBLE PRECISION;
DECLARE VMYRANK BIGINT = 0;
BEGIN
  FOR SELECT MESURE FROM TNODE_VAL WHERE ID=100 ORDER BY MESURE ASC INTO :VMESURE
  DO
  BEGIN
    VMYRANK = VMYRANK+1;
    INSERT INTO TNODE_TEMP (MYRANK, MESURE) VALUES (:VMYRANK, :VMESURE);
  END
END !!
```

```
WITH C (MAXI) AS (SELECT MAX(MYRANK) FROM TNODE_TEMP)
SELECT E1.MESURE Q1, E2.MESURE Q2, E3.MESURE Q3, E4.MESURE Q4
FROM TNODE_TEMP E1
  JOIN C ON E1.MYRANK=CEIL(C.MAXI/4)
CROSS JOIN
  TNODE_TEMP E2
CROSS JOIN
  TNODE_TEMP E3
CROSS JOIN
  TNODE_TEMP E4
WHERE E2.MYRANK=CEIL(C.MAXI/2)
  AND E3.MYRANK=CEIL(C.MAXI*3/4)
  AND E4.MYRANK=CEIL(C.MAXI)
```

Common table expressions

Example chronological serie (recursive)

```
WITH RECURSIVE CTETemps (madata) AS
(
  SELECT date '2004-02-01' maDate FROM RDB$DATABASE
  UNION ALL
  SELECT maDate + 1
  FROM CTETemps
  WHERE maDate + 1 < '2004-03-01'
),
CTEVAL (madata,moyenne,nombre) AS
(SELECT CAST(TNODE_VAL.DATE_MESURE AS DATE), Avg(TNODE_VAL.Mesure) , COUNT(*)
FROM TNODE_VAL
WHERE TNODE_VAL.ID=100
AND CAST(TNODE_VAL.DATE_MESURE AS DATE) BETWEEN '2004-02-01' AND '2004-02-29'
GROUP BY 1)
SELECT CTETemps.madata, COALESCE(CTEVAL.Moyenne,'N/A') Moyenne ,CTEVAL.nombre
FROM CTETemps
LEFT JOIN CTEVAL ON CTETemps.madata=CTEVAL.madata
```


Domains everywhere

•Syntax

```
data_type ::= <builtin_data_type>  
| <domain_name>  
| TYPE OF <domain_name>
```

- Using plain <domain_name> data type inherits domain constraints (CHECK, NOT NULL) and default value
- Using new keyword **TYPE OF** only data type from domain definition is inherited.
- Engine track dependency between domain and stored procedure or trigger where its used and not allows to drop such domains
- When a domain is altered engine don't recompile dependent objects
- Dependent objects will take up new domain definition only after reload into metadata cache

Update or insert

•Syntax

- **UPDATE OR INSERT** INTO <table> [(col1, ..., colN)]
VALUES (val1, ..., valN)
- [**MATCHING** (mc1, ..., mcM)] *-- new keyword **MATCHING**, optional*
- [**RETURNING** (ret1, ..., retR) *-- optional RETURNING ... INTO*
[INTO :var1, ..., :varR]] *-- clause (introduced in FB 2.0)*

Update or insert rules

- If optional clause **MATCHING** is omitted then primary key is used to match table rows against replaced values
- If **RETURNING** clause is used then no more than one row must correspond to **MATCHING** criteria
- “**IF (ROWCOUNT = 0) ...**” construction can be used in PSQL only while **UPDATE OR INSERT** can be used directly as it is usual SQL statement
- User must have both **INSERT** and **UPDATE** privileges on table
- If row was inserted then **INSERT** trigger fired else fired **UPDATE** trigger

Merge

•Syntax

- **MERGE** INTO <table> *-- new keyword **MERGE***
 - **USING** <table_or_join> *-- source of data for merging*
 - **ON** <search_condition> *-- how to find target row*
 - [**WHEN MATCHED THEN** *-- UPDATE query specification*
UPDATE SET col1 = val1, ..., colN = valN]
 - [**WHEN NOT MATCHED THEN** *-- INSERT query specification*
INSERT [(col1, ..., colN)] **VALUES** (val1, ..., valN)]
- Both **INSERT** and **UPDATE** query specifications are optional but at least one of them must be defined

Merge example

```

CREATE TABLE AMOUNTS
(
  GOODID INT,
  SUMMA NUMERIC(18, 4),

  CONSTRAINT PK_AMOUNTS
    PRIMARY KEY (GOODID)
);

CREATE TABLE ENTRIES
(
  GOODID INT,
  DT DATE,
  DBT NUMERIC(18, 4),
  CRD NUMERIC(18, 4),

  CONSTRAINT PK_ENTRIES
    PRIMARY KEY (GOODID, DT)
);

```

```

MERGE INTO AMOUNTS USING ENTRIES E
  ON GOODID = E.GOODID
  WHEN MATCHED
    THEN UPDATE SET SUMMA = SUMMA + E.DBT - E.CRD
  WHEN NOT MATCHED
    THEN INSERT (GOODID, SUMMA)
    VALUES (E.GOODID, E.DBT - E.CRD)

```

ENTRIES			
GOODID	DT	DBT	CRD
1	01.01.2006	10.00	0.00
2	03.01.2006	0.00	4.00
2	02.01.2006	10.00	0.00
2	01.01.2006	15.00	0.00
1	05.01.2006	0.00	3.00

AMOUNTS	
GOODID	SUMMA
1	7.00
2	21.00

New built-in functions

• Mathematical

- ABS, MOD, SIGN, CEIL, CEILING, FLOOR, ROUND, TRUNC, PI
- LOG, LOG10, LN, POWER, EXP, SQRT
- COS, COSH, ACOS, SIN, SINH, ASIN, TAN, TANH, ATAN, ATAN2, COT

• Bit logic

- BIN_AND, BIN_OR, BIN_SHL, BIN_SHR, BIN_XOR

• Datetime arithmetics

- DATEADD, DATEDIFF

• String manipulations

- ASCII_CHAR, ASCII_VAL
- LEFT, RIGHT, LPAD, RPAD, POSITION, REPLACE, OVERLAY, REVERSE

• Other

- DECODE, MAXVALUE, MINVALUE
- GEN_UUID, HASH, RAND
- LIST (String aggregating function)

Engine Development

Core Development

Dimitry Yemanov
(Project Co-ordinator)

Claudio Valderrama

Vlad Horsun

Alex Peshkov

Adriano dos Santos
Fernandes

Arno Brinkman

Alex Krutyakov

Assisted by

Paul Reeves

Helen Borrie

Pavel Cisar

Philippe Makowski

Paul Beach

Jaybird



- ★ **Developers**

 - Roman Rokytskyy (Project Admin)**

 - Evgeny Putilin**

 - Gabriel Reid**

- **Development**

 - Latest V2.1.5 June 2008**

 - V2.2.2 in development with test builds**

.NET Provider

*** Developers**

Jiri Cincura (Project Admin)

Alessandro Petrelli

Le Roy Arnaud

Rick Hoover

• Development

ADO .Net Provider, Win32 & Mono

Nov 2007 - V2.1 Released

Mar 2008 V2.5 Alpha 2 - towards support for New Entity Framework (.NET 3.5 & 2.0)

Python driver

★ Developers

Pavel Cisar (Project Admin)

• Development

KinterbasDB 3.2.1 release July 2008 (maintenance release)

Future : rewrite in pure Python using the wire protocol instead using client library

Documentation

★ Developers

- **Paul Vinkenoog, Helen Borrie, Norman Dunbar, Geoff Worboys, Frank Ingermann, Sergey Mereutsa and translators**

• Development

- **New doc : Firebird 1.5 Language reference. Other volumes will follow, each encompassing a macro release. Translations are under way.**

Organisational

- ★ **Releasing “Often”**
- ★ **Regular Snapshot builds for Linux x86, x84, Windows**
- ★ **High usage of Jira bug tracker**
- ★ **Increase in commitments of developer hours**
- ★ **One new developer joined the Core team**

Bug Tracker



[HOME](#) [BROWSE PROJECT](#) [FIND ISSUES](#) [PLANNING BOARD](#)

QUICK SEARCH:

[Log In](#)  

Report: [Created vs Resolved Issues Report](#)

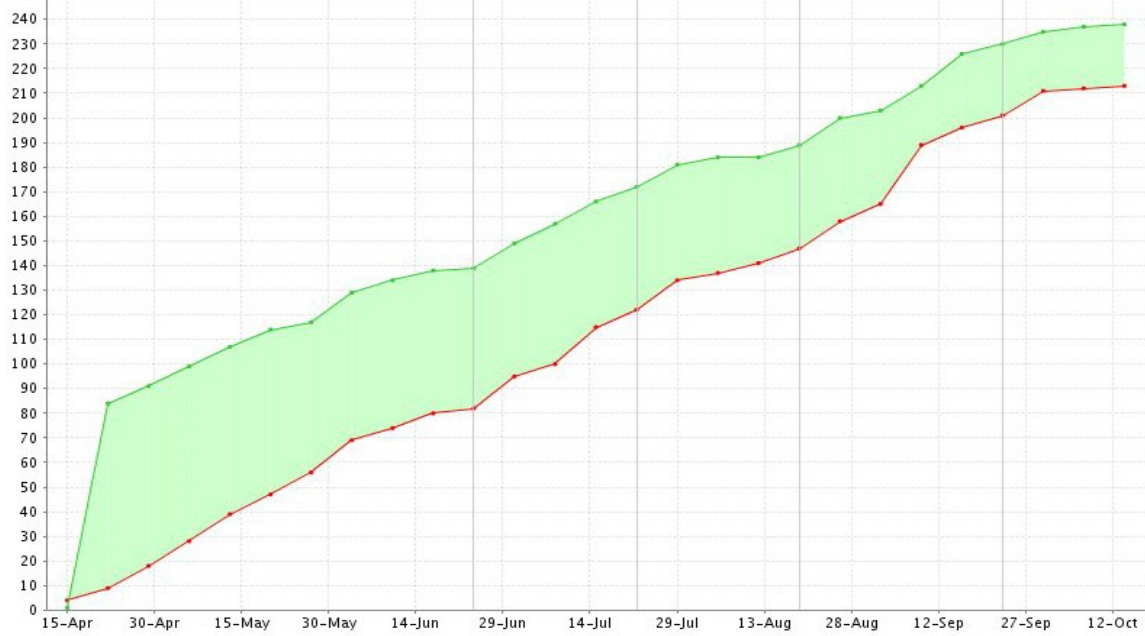
Description:

A report showing issues created vs issues resolved.

Filter: **Core engine - Bugs only**

Chart

This chart shows the number of issues **created** vs the number of issues **resolved** in the last **180** days..



Porting Hall of Fame

Firebird 2.x

Linux AMD64 - Alex Peshkov

Linux MIPSEL - Fabrice Aeschbacher

Linux PPC - Damyan Ivanov

Win64 (AMD & EM64T) - Dimitry Yemanov

FreeBSD - Sergey Skvortsov

MacOS - Paul Beach

Solaris Intel and Sparc - Paul Beach & Alex Peshkov

HPUX (ongoing) - Paul Beach & Alex Peshkov

Other Highlights

- ★ **SourceForge Community Choice award in 2 categories (Best Project for Enterprise and Best User Support) August 2007**
- ★ **Actively maintained Firebird FAQ site started by David Richmond at www.firebirdfaq.org**
- ★ **Dec 2007 Sourceforge project of the month**

Other Highlights

- ★ **Firebird 2.x fully ported to Mac Intel and native platforms. 64bit for Intel, 64bit for PPC coming soon**
- ★ **Firebird 2.0.x Solaris Classic (Intel & Sparc) ports available. Work continues on 64bit**
- ★ **Firebird 2.0.x HPUX Classic 64bit**

Resource Donations

- ★ **Donation of a Mac for builds and testing**
- ★ **Broadview Systems continues to provide hosting and on-site management of web, tracker and mail servers**
- ★ **Can you help?**

Firebird's Future

Current situation

- ★ **3 versions under active development**
- ★ **Pressure to continue support for V.1.5**
- ★ **Relatively small team**
- ★ **More funding and assistance is always needed**

V2.5

- **Scalable multi-threading with per-connection caches**
- **Optimised lock management**
- **Monitoring extensions**
- **ALTER VIEW statement**
- **Support for regular expressions in SQL**
- **Autonomous transactions**
- **User account maintenance via SQL**
- **New built-in functions**
- **Queries against external databases**
- **Audit/logging facilities**

New threading architecture

- **For superserver :**
 - threads distributed evenly to available processors according the database clients attach to
- **For Classic :**
 - Now multi-threaded, even on Posix
- **Superclassic :**
 - Multi threaded mode with connection pool
 - performs better than Classic
 - use fewer kernel ressources
 - does not require [x]inted
 - but still cache per connexions

Regex Search using SIMILAR TO

- **Syntax :**

<similar predicate> ::=

- **<value> [NOT] SIMILAR TO <similar pattern>**
[ESCAPE <escape character>]

```
create table department (
  number numeric(3) not null,
  name varchar(25) not null,
  phone varchar(14)
  check (phone similar to '\\([0-9]{3}\\) [0-9]{3}\\-[0-9]{4}' escape '\\')
);

insert into department
  values ('000', 'Corporate Headquarters', '(408) 555-1234');
insert into department
  values ('100', 'Sales and Marketing', '(415) 555-1234');
insert into department
  values ('140', 'Field Office: Canada', '(416) 677-1000');

insert into department
  values ('600', 'Engineering', '(408) 555-123'); -- check constraint violation

select * from department
  where phone not similar to '\\([0-9]{3}\\) 555\\-%' escape '\\';
```

Autonomous transaction

- **Syntax :**
IN AUTONOMOUS TRANSACTION DO
<simple statement | compound statement>

```
SET TERM ^ ;  
CREATE OR ALTER TRIGGER TRG_CONN ON CONNECT  
AS  
BEGIN  
  IF (USER != 'GOOD_USER') THEN  
    BEGIN  
      IN AUTONOMOUS TRANSACTION  
      DO  
      BEGIN  
        INSERT INTO LOG (LOGDATE,MSG) VALUES (CURRENT_TIMESTAMP, 'Connection rejected');  
      END  
    EXCEPTION EX_CONNECT USER || ' not allowed !';  
  END  
END ^
```

External queries

- **Syntax :**

- `[FOR] EXECUTE STATEMENT <query_text> [(<input_parameters>)]`
- `[ON EXTERNAL [DATA SOURCE] <connection_string>]`
- `[WITH {AUTONOMOUS | COMMON} TRANSACTION]`
- `[AS USER <user_name>]`
- `[PASSWORD <password>]`
- `[INTO <variables>]`

Monitoring extension

- **Two more monitoring tables :**
 - **MON\$MEMORY_USAGE**
 - **MON\$CONTEXT_VARIABLES**

Top 10 statements ranked according to their memory usage

```
SELECT FIRST 10
  STMT.MON$ATTACHMENT_ID,
  STMT.MON$SQL_TEXT,
  MEM.MON$MEMORY_USED
FROM MON$MEMORY_USAGE MEM
NATURAL JOIN MON$STATEMENTS STMT
ORDER BY MEM.MON$MEMORY_USED DESC
```


Terminating a client

- **Now you can also terminate client session**

To cancel all clients except the current connection

```
DELETE FROM MON$ATTACHMENTS  
WHERE MON$ATTACHEMENT_ID <> CURRENT_CONNECTION
```

Roadmap



- **July 2008**
 - 2.1.1, 2.5 alpha
- **October 2008**
 - 2.5 RC1
- **Q4 2008**
 - 2.5, 2.0.5, 2.1.2, 3.0 alpha

Firebird 3.0

- **External (e.g. Java) procedures**
- **Effective multi-treading with shared cache**
- **Compiled statement cache**
- **Per-database configuration**
- **Cross-database queries**
- **External (e.g. Java) procedures**
- **New SQL features**



Thank You