Firebird on the road from v4 to v5

Dmitry Yemanov
dimitr@firebirdsql.org

Firebird Project
www.firebirdsql.org
Firebird Conference 2019
Berlin, 17-19 October
Maintenance

Firebird 2.5

• v2.5.9 was released in June 2019
• >20 bugfixes, 3 improvements
• Discontinued
Maintenance

Firebird 2.5

- v2.5.9 was released in June 2019
- >20 bugfixes, 3 improvements
- Discontinued

Firebird 3.0

- v3.0.4 was released in October 2018
- v3.0.5 is expected in Q4 2019
- >60 bugfixes, 6 improvements
- Supported until v5 is out
Firebird 4.0 progress

Features introduced in Alpha 1

- ODS 13
- 32KB page size
- Metadata names as long as 63 characters
- Incremental restore via nbackup
- Predefined system roles and fine-grained DBA permissions
- Timeout for statements and connections
- Aggregated roles
- SQL SECURITY declaration for PSQL objects
- Batch API
Firebird 4.0 progress

Features introduced in Beta 1

- Built-in logical replication
- Statement-level read consistency in RC transactions
- Intermediate garbage collection
- DECFLOAT data type
- NUMERIC/DECIMAL up to 38 digits
- Support for time zones
- Built-in cryptographic functions
- Encrypted database support for GBAK
- Pooling of external connections
Firebird 4.0 progress

Progress of the past year

- Beta 1 was released in February 2019
- 55 bugs fixed since
- 12 features and improvements added
Firebird 4.0 progress

Progress of the past year

- Beta 1 was released in February 2019
- 55 bugs fixed since
- 12 features and improvements added

Most interesting

- Snapshots shared among transactions
- Built-in functions converting binary string to hexadecimal representation and vice versa
- Lateral derived tables
- RECREATE USER statement
Firebird 4.0 progress

Features being reworked

• Native INT128 instead of DECFLOAT-based integer for longish NUMERIC/DECIMAL data types
• Flexible binding for new data types
• New logic for query restarts in READ CONSISTENCY transactions
• Issues with time zones (regions) and ICU dependency on the client side
• Schema-driven replication customization
Firebird 4.0 progress

Where we stand now

- Beta 2 in Q4 2019
- Release Candidate in Q1 2020
- Final release in Q2 2020
Firebird 4.0 preview

DECFLOAT data type

- Introduced in IBM DB2
- Now part of the standard (SQL:2016)
- Stores decimal values precisely
- DECFLOAT / DECFLOAT(16) / DECFLOAT(34)
- 64-bit or 128-bit storage (IEEE 754)
- Supported by all built-in functions
- Intermediate calculations with 34 decimal digits
DECFLOAT special functions

- COMPARE_DECFLOAT — compare exactly (w/o specials)
  \[ \text{COMPARE_DECFLOAT}(2.17, 2.170) = 2 \]

- NORMALIZE_DECFLOAT — remove trailing zeroes
  \[ \text{NORMALIZE_DECFLOAT}(12.0) = 12 \]
  \[ \text{NORMALIZE_DECFLOAT}(120) = 1.2E+2 \]

- QUANTIZE — scale by pattern
  \[ \text{QUANTIZE}(1234, 9.999) = 1234.000 \]

- TOTALORDER — compare exactly with specials
  \[ -\text{nan} < -\text{snan} < -\text{inf} < -0.1 < -0.10 < -0 < 0 \]
  \[ 0 < 0.10 < 0.1 < \text{inf} < \text{snan} < \text{nan} \]
Firebird 4.0 preview

DECFLOAT management

- Works at the session level
- SET DECFLOAT ROUND <mode>
  
  \textit{CEILING, UP, HALF_UP, HALF_EVEN, HALF_DOWN, DOWN, FLOOR, REROUND}
  
- SET DECFLOAT TRAPS TO <traps-list>
  
  \textit{DIVISION_BY_ZERO, INEXACT, INVALID_OPERATION, OVERFLOW, UNDERFLOW}
NUMERIC / DECIMAL improvements

- Maximum precision is extended to 38 decimal digits
- Backed by native 128-bit signed integer
- INT128 is not directly surfaced (so far)
- Intermediate calculations with either NUMERIC(38) or DECFLOAT(34)
Time zone support

- TIME[STAMP] WITHOUT TIME ZONE
- TIME[STAMP] WITH TIME ZONE
- Extra two bytes for time zone information
- Time zone can be defined via the region name, e.g. '10:00 America/Los_Angeles'
or via the hours:minutes displacement from GMT, e.g. '10:00 -08:00' or '10:00 -08'
- Time / timestamp part is stored (compared, sorted) as UTC
- Time zone is stored «as is»
Firebird 4.0 preview

Time zone support

- AT expression translates the value to another time zone
  
  ```
  select current_timestamp at time zone 'America/Sao_Paulo'
  or
  select timestamp '2018-01-01 12:00 GMT' at local
  ```

- EXTRACT is extended
  
  ```
  TIMEZONE_HOUR and TIMEZONE_MINUTE
  ```

- Changed CURRENT_TIME and CURRENT_TIMESTAMP to return UTC time

- Added LOCALTIME and LOCALTIMESTAMP to return legacy (WITHOUT TIME ZONE) time
Time zone management

- Default time zone for all sessions can be defined in firebird.conf (DefaultTimeZone setting)
- Time zone can be overridden per session (via DPB)
- Time zone can be altered at runtime
  \[
  \text{SET TIME ZONE } <tz>
  \]
- Time zone can be reset at runtime
  \[
  \text{SET TIME ZONE LOCAL}
  \]
Further development

User demands

• Release often — publicity is vitally important
• **BUT** nobody wants to upgrade often, due to
  • Production downtime (backup/restore duration)
  • Testing efforts (migration difficulties)
Further development

User demands

• Release often — publicity is vitally important
• **BUT** nobody wants to upgrade often, due to
  • Production downtime (backup/restore duration)
  • Testing efforts (migration difficulties)

Problems

• Limited resources
• Aging codebase
• Feature-based release schedule
• **THUS** long intervals between releases
Further development

Possible solution

- Release in smaller steps
  - Less features per release
  - More frequently
- Adopt the «tick-tock» idea
  - Major release: architectural
  - Minor release: technological
- Simplify upgrade paths
Further development

Major release

- Code refactoring
- Major ODS change (and thus b/r requirement)
- A few unique features + basement for others
- Incompatibilities are possible (although minimized)
- Once per 2-3 years
Further development

Minor release

• Minor ODS changes are possible
• Straightforward upgrade option (no b/r requirement)
• A few features based on the current ODS
• Maximum backward compatibility
• Quite stable, no Alpha / Beta stages
• Once per 9-12 months
Further development

Minor ODS management (example)

- Firebird 4.2 introduces ODS 13.2, but can work with ODS 13.0 (created by v4.0) and ODS 13.1 (created by v4.1)
- In the case you need ODS 13.2 features, you upgrade ODS manually
- gfix -upgrade instead of backup/restore
- Upgrade is almost immediate (but exclusive DBA access is required)
- Downgrade is possible via backup/restore
Further development

Possible schedule (based on 12/24 months intervals)
Firebird 4.x evolution

Replication is not a simple «here it is» feature, rather a start of the long way forward

- Multiple publications/subscriptions scenarios
- Bi-directional (multi-master) replication
- Journal compression / encryption
- Multi-threaded applying on the replica side
- etc
Firebird 4.x evolution

Other changes

- Mostly finished but not yet committed features (e.g. TRUNCATE TABLE)
- Some unique features from HQbird and RedDatabase that can be backported into ODS13
- Optimizer improvements
- Monitoring / tracing improvements
Planning Firebird 5.0

Top-voted tickets from the tracker

1) CORE-743     Job / Task Scheduler
2) CORE-720     Local temporary tables
3) CORE-776     Database Links
4) CORE-808     Add support for INTERSECT and EXCEPT data set operators
5) CORE-796     Create Table as Select ....
6) CORE-659     GIS implementation (opengis)
7) CORE-734     Full-Text indexing
8) CORE-5148    Support native JSON datatype for columns as MySQL / PostgreSql
9) CORE-738     Add support for SQL Schemas
10) CORE-670    New database object - Constants
Planning Firebird 5.0

Under the hood

- ODS 14
- Longer (unlimited) record size
- Better on-disk compression
- New BLR format w/o context limitations
- Indexing improvements
Planning Firebird 5.0

Technical Task Group

- Firebird Foundation sponsors
- Project administrators
- Core developers
Planning Firebird 5.0

Technical Task Group

- Firebird Foundation sponsors
- Project administrators
- Core developers

Decision making

- List of suggested features
- Voting
- Planning board
Questions?