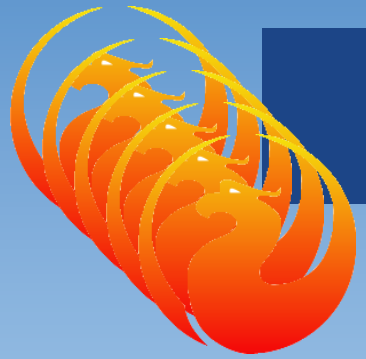
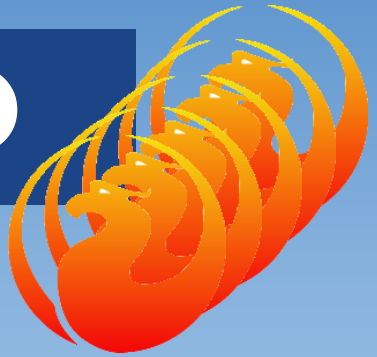
DelphiDay italian conference

PIACENZA EDITION

2019 - XVIII Edizione



FIREBIRD 4.0



It's time to replicate! replicate! replicate!

wintech italia

embarcadero reseller partner

Fabio Codebue

P-SOFT

P-Soft

www.p-soft.biz

www.firebirdsql.it

@fcodebue

linkedin.com/in/fcodebue

@delphiforce

f.codebue@p-soft.biz

AGENDA

- → Firebird 4.0 news
- → Firebird replication
- → Replication sample

Firebird news

Support for international time zones

Time zone support from Firebird 4.0 onward consists of

- **data types**

  TIME [{ WITHOUT | WITH } TIME ZONE]

  TIMESTAMP [{ WITHOUT | WITH } TIME ZONE]

- **expressions and statements** to work with time zones

  CURRENT_TIME and CURRENT_TIMESTAMP are changed

  they now return

  TIME WITH TIME ZONE and TIMESTAMP WITH TIME ZONE

- **conversion** between data types without/with time zones
- **prefaced Implicit Date/Time Literals Now Rejected**

Built-in functions FIRST_DAY and LAST_DAY

FIRST_DAY Returns a date or timestamp (as appropriate) with the first day of the year | month | week of a given date or timestamp value.

```
FIRST_DAY( OF {YEAR|MONTH|WEEK} FROM <date_or_timestamp>)
```

LAST_DAY Returns a date or timestamp (as appropriate) with the last day of the year | month | week of a given date or timestamp value.

```
LAST_DAY( OF {YEAR|MONTH|WEEK} FROM <date_or_timestamp> )
```



New way to capture the database snapshot

SNAPSHOT TRANSACTION

- sees the database state only as it was at the moment it started (TIP)

READ COMMITTED TRANSACTION

- sees the database state requesting TIP

READ COMMITTED READ CONSISTENCY TRANSACTION

- ??????????????

Pooling of external connections

- **To avoid delays when external connections** are being established frequently, the external data source (EDS) subsystem has been augmented

```
ALTER EXTERNAL CONNECTIONS POOL { <parameter
                variants> }
```


change are not persistent.

Pooling of external connections

Sets the maximum number of idle connections

```
ALTER EXTERNAL CONNECTIONS POOL SET SIZE <int>
```

Sets the lifetime of an idle connection, from 1 second to 24 hours. The <time_part> can be SECOND |MINUTE | HOUR.

```
ALTER EXTERNAL CONNECTIONS POOL SET LIFETIME <int> <tp>
```

Closes all idle connections and instigates dissociation of all active connections immediately they become unused

```
ALTER EXTERNAL CONNECTIONS POOL CLEAR ALL
```

Close an expired idle connections

```
ALTER EXTERNAL CONNECTIONS POOL CLEAR OLDEST
```

change are not persistent.

Extended length of metadata identifiers

- The **maximum length of objects names** from this version forward is **63 characters** (Previous length of objects names 31 bytes)
- Double quotes are not counted
- firebird.conf and/or databases.conf

  MaxIdentifierByteLength: Sets a limit for the number of bytes allowed in an object identifier.

  MaxIdentifierCharLength: Sets a limit for the number of characters allowed in an object identifier

It is an integer, default to 252 bytes,

*63 characters * 4, 4 being the maximum number of bytes for each character.*

Configurable time-outs

- Configurable timeouts for running SQL statements and for idle connections (sessions).
- **An idle session timeout allows a user connection to close automatically after a specified period of inactivity**.

```
SET SESSION IDLE TIMEOUT <value> [HOUR|MINUTE|SECOND]
```

If the time unit is not set, it defaults to MINUTE.

- By default, the **idle timeout is not enabled.**
- No minimum or maximum limit is imposed but a reasonably large period, such as a **few hours, is recommended.**

Numerics

Extended precision for numerics

- Fixed point numerics with precision up to 34 digits

```
NUMERIC ( P [, S] ) e DECIMAL ( P [, S] )
where P is precision (P <= 34
```

- New ***DECFLOAT data type***

DECFLOAT is an SQL:2016 standard-compliant numeric type that **stores floating-point numbers precisely**,

FLOAT or DOUBLE PRECISION that provide a binary approximation of the purported precision.

```
DECFLOAT(P)
```

Enhanced system privileges

Enables granting and revoking special privileges for regular users to perform tasks that have been historically limited to SYSDBA only

- Run utilities such as gbak, gfix, nbackup and so on
- Shut down a database and bring it online
- Trace other users' attachments
- Access the monitoring tables
- Run management statements

GRANT ROLE TO ROLE

Cumulative roles

roles emebedded within roles

```
GRANT [DEFAULT] <role name> TO [USER | ROLE] <user/role
                name> [WITH ADMIN OPTION];
```

```
REVOKE [DEFAULT] <role name> FROM [USER | ROLE]
     <user/role name> [WITH ADMIN OPTION];
```

Window functions extensions

Add to the OVER clause for Window functions (now supports not just the sub-clauses PARTITION and ORDER subclauses)

- FRAMES
- WINDOWS WITH NAMES
- ranking functions
  - PERCENT_RANK : is a ratio of RANK to group count.
  - CUME_DIST : is the cumulative distribution of a value in a group.
  - NTILE: takes an argument and distributes the rows into the specified number of groups

FILTER Clause for Aggregate Functions

The **FILTER clause extends aggregate functions** (sum, avg, count, etc.) by an additional WHERE clause

```
select count(*) filter (where status = 'A') status_a,
          count(*) filter (where status = 'E') status_e
from data;
```

Sintax

```
aggregate_function [FILTER (WHERE <condition>)] [OVER (<window>)]
```



Enhanced RETURNING clause in DML

- supports **RETURNING * syntax**, and variants,
- to return a complete set of field values after committing a row that has been inserted, updated or deleted

```
INSERT INTO T1 (F1, F2) VALUES (:F1, :F2) RETURNING *
DELETE FROM T1 WHERE F1 = 1 RETURNING *
UPDATE T1 SET F2 = F2 * 10 RETURNING OLD.*, NEW.*
```

Monitoring Status of Attachments

Compression and encryption status of a connection are now available in the monitoring table MON$ATTACHMENTS:

- **MON$WIRE_COMPRESSED**

  wire compression enabled = 1, disabled = 0

- **MON$WIRE_ENCRYPTED**

  wire encryption enabled = 1, disabled = 0

Changes in the Firebird Engine

- **Extended Maximum Page Size**
  - The maximum page size for databases created under ODS 13 has been extended from 16 Kb to 32 Kb.
- **External Functions (UDFs) Feature Deprecated**
  - The default setting for the configuration parameter *UdfAccess* is NONE.
  - now require explicit configuration to *Restrict UDF*
  - The UDF libraries (ib_udf, fbudf) are no longer distributed in the installation kits
  - Most of the functions in the libraries previously distributed in the shared (dynamic) libraries *ib_udf* and *fbudf* had already been replaced with built-in functional analogs.
  - A few remaining UDFs have been replaced with either analog routines in a new library of UDRs named *udf_compat* or converted to stored functions.

replication

Last update firebird 4.0.0 beta 1

Firebird Replication

- **uni-directional**: master-slave
- **logical replication**: record-level replication

Events that are tracked for replication include

- inserted/updated/deleted records
- sequence changes
- DDL statements

Replication is transactional and **commit order is preserved**.

Replication can **track changes** either **in all tables, or subset of tables**.

Any table that is to be replicated **must have a primary key** or, at least, **a unique key**.

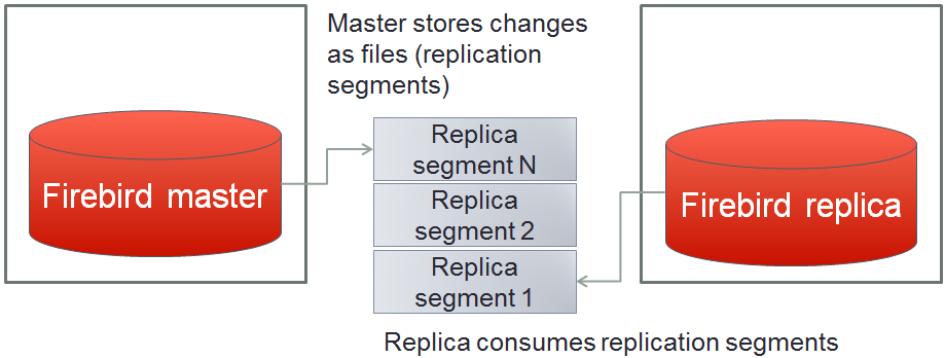
FB Rep: Synchronous mode

Replication Modes : **Synchronous mode**

- In synchronous replication, the primary (master) database is permanently connected to the replica (slave) database(s) and changes are replicated immediately.
- Effectively the **databases are in sync after every commit**, which could have an impact on performance due to additional network traffic and round-trips.
- Although some recent uncommitted changes may be buffered, they are **not transmitted until committed.**
- More than one synchronous replication can be configured

FB Rep: replication modes

Replication Modes : Asynchronous modes



1. Changes are written into local journal files
2. local journal files are transferred over the wire
3. Once arrive was applied to the replica database.
4. The **impact on performance is much lower**,
5. We have a delay—**replication lag**— while changes wait to be applied to the replica database

FB Rep: Access modes

Access modes for replica databases

- **read-only**
  - only queries that do not modify data are allowed
  - Modifications are limited to the replication process only.
  - Global temporary tables can be modified, as they are not replicated
- **read-write**
  - allows execution of any query
  - potential conflicts must be resolved by users.

FB Rep: Journalling

Journalling

Replicated changes are written into the journal which consists of *multiple files*, known as **replication segments**

The Firebird server writes segments continuously, one after another.

segment sequence: *a unique number* which is generated sequentially identifying *replication segment*

The global sequence counter is stored inside the replicated database and is reset only when the database is restored from backup.

FB Rep: error log reporting

Error Reporting

- **All replication errors and warnings** (such as detected conflicts) are written into the replication.log file.
- It may also include detailed descriptions of the operations performed by the replicator.

Log file location

- The replication.log file is stored in the Firebird log directory.
- By default, the Firebird log directory is **the root directory of the Firebird installation**.

FB Rep: setting up

Setting Up the Master Side

- Replication is configured using a single configuration file, **replication.conf**
- All parameters are applied per database setting
- To apply any changes to the master-side settings, *all users must be reconnected*

Setting Up the Replica Side

- The same replication.conf file is used for setting up the replica side.
- Setting the parameter *log_source_directory* specifies the location that the Firebird server scans for the transmitted segments.
- In addition, the DBA may specify explicitly which source database is accepted for replication, by setting the parameter *source_guid.*


Go to
sample

FB Rep: Origin side setting

buffer_size

- **Size of the local buffer used to accumulate changes** that can be deferred until the transaction commit/rollback.
- The **bigger this value the less disk access concurrency** (related to log IOPS) happens.
- For synchronous replication, it also affects number of network round-trips between primary and replica hosts.
- A **larger buffer** costs a *longer replication checkpoints* (**delay to synchronize the original database with its replica at commit**).

```
buffer_size = 1048576 # 1MB
```

FB Rep: Origin side setting

include_filter

- Pattern (regular expression) that defines what tables must be included into replication. **By default, all tables are replicated.**

```
include_filter =
```

exclude_filter

- Pattern (regular expression) that defines what tables must be excluded from replication.
- **By default, all tables are replicated.**

```
exclude_filter =
```

FB Rep: Origin side setting

log_directory

- Directory to store replication log files

```
log_directory =
```

log_file_prefix

- Prefix for replication log file names. It will be automatically suffixed with an ordinal sequential number.
- If not specified, *database filename (without path) is used as a prefix.*

```
log_file_prefix
```

FB Rep: Origin side setting

log_segment_size

- Maximum allowed size for a single replication segment.

```
log_segment_size = 16777216 # 16MB
```

log_segment_count

- Maximum allowed number of full replication segments.
- Once this limit is reached, the replication process is temporarily delayed to allow the archiving to catch up.
- If any of the full segments is not archived during one minute, the replication fails with an error.
- Zero means an unlimited number of segments pending archiving.

FB Rep: Origin side setting

log_group_flush_delay

- Delay, in milliseconds, to wait before the changes are synchronously flushed to the log (usually at commit time).
- This allows multiple concurrently committing transactions to amortise I/O costs by sharing a single flush operation.
- Zero means no delay, i.e. "group flushing" is disabled.

```
log_group_flush_delay = 0
```

FB Rep: Origin side setting

log_archive_directory

- **Directory for the archived log files.**
- **Directory to store archived replication segments.**
- It also defines the **$(archpathname)** substitution macro (see below).

FB Rep: Origin side setting

log_archive_command

- Program (complete command line with arguments) that is executed when some replication segment gets full and needs archiving.
- This program MUST return zero ONLY if archiving has been performed successfully.
- Special predefined macros are available:
- $(logfilename) - file name (without path) of the log segment being archived
- $(logpathname) - full path name of the log segment being archived

  same as log_directory + $(logfilename)
- $(archpathname) - suggested full path name for the archived segment

  same as log_archive_directory + $(logfilename)
- Simplest configuration is to use standard OS commands for archiving, e.g.:

```
Linux: "test ! -f $(archpathname) && cp $(logpathname) $(archpathname)"
Windows: "copy $(logpathname) $(archpathname)"
```

FB Rep: Origin side setting

log_archive_timeout

- Timeout, in seconds, to wait until incomplete segment is scheduled for archiving.
- It allows to minimize the replication gap if the database is modified rarely.
- Zero means no intermediate archiving, i.e. segments are archived only after reaching their maximum size

  (defined by log_segment_size).

```
log_archive_timeout = 60
```

FB Rep: Origin side setting

sync_replica

- Connection string to the replica database (used for synchronous replication only)
- Multiple entries are allowed (for different synchronous replicas)

```
[<login>:<password>@]<DB connection string>

server2:replica
john:smith@server2:replica
server2:/my/replica/database.fdb
john:smith@server2:/my/replica/database.fdb
```

FB Rep: Replica side setting

log_source_directory

- Directory to search for the log files to be replicated

```
log_source_directory =
```

source_guid

- Filter to limit replication to the particular source database (based on its GUID).
- Expected format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}

```
source_guid = as862PD4-2019-0ed2-58a8-12DE5d2shP5S
```

FB Rep: Replica side setting

verbose_logging

- If enabled, replication.log contains the detailed log of operations performed by the replication server.
- **Otherwise (by default), only errors and warnings are logged**

```
verbose_logging = false
```

FB Rep: Replica side setting

apply_idle_timeout

- **Timeout (in seconds) to wait before scanning for the new replication segments.**
- It's used to pause the replication server when all existing segments are already applied to the replica database and there are no new segments in the specified directory.

```
apply_idle_timeout = 10
```

FB Rep: Replica side setting

apply_error_timeout

- **Timeout (in seconds) to wait before retrying to apply the queued segments after error.**
- It's used to pause the replication server after some critical error happened during replication.
- In this case, the server disconnects from the replica database, sleeps for the specified timeout, then reconnects back and tries to re-apply the latest segments from the point of failure

```
apply_error_timeout = 60
```


Go deeper
& enjoy!

GRAZIE…

Sostienici, diventa un associato …
https://firebirdsql.org/en/membership/

Fabio Codebue

P-Soft

www.p-soft.biz

www.firebirdsql.it

@fcodebue

linkedin.com/in/fcodebue

@delphiforce

f.codebue@p-soft.biz