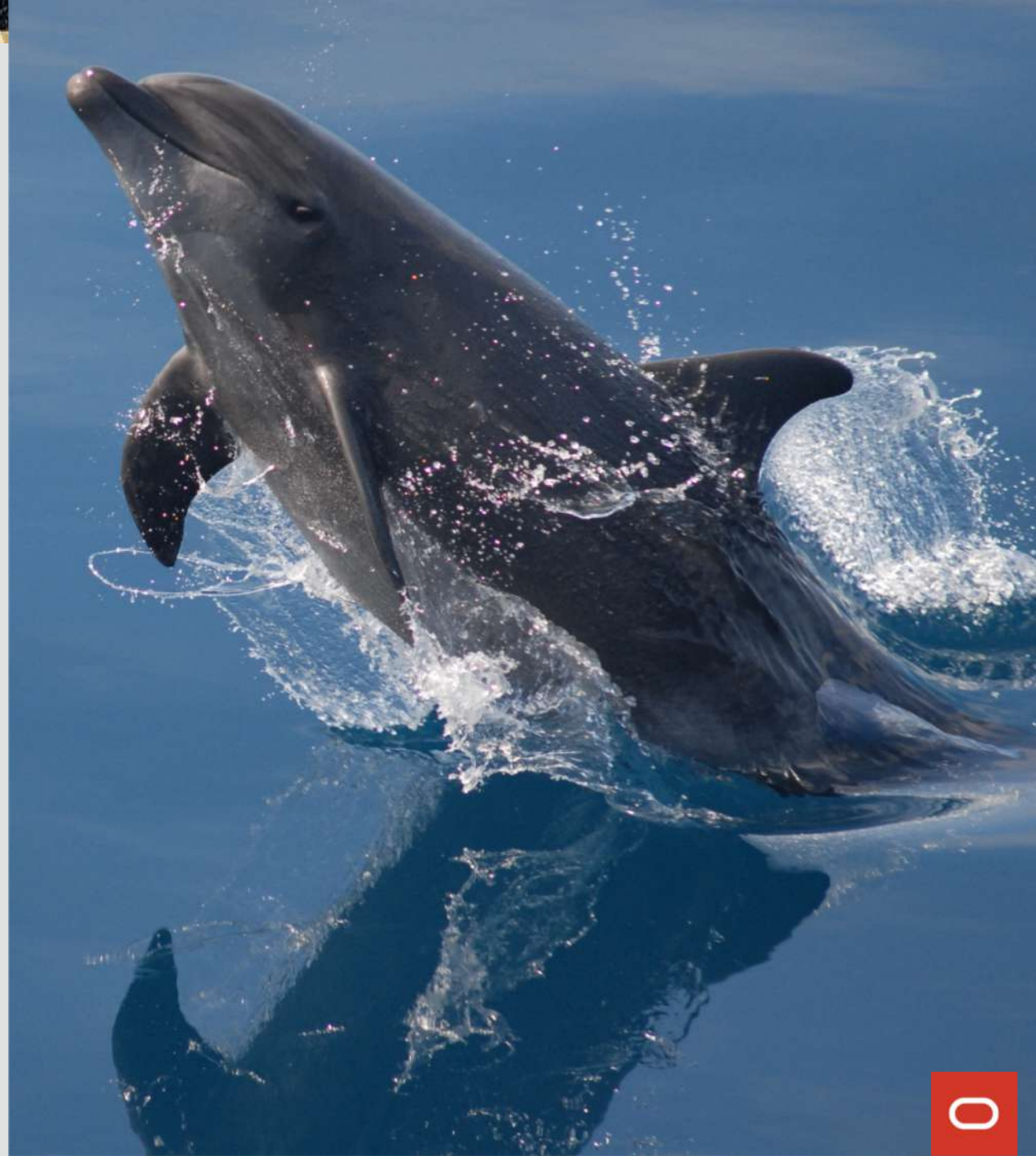
Automatic Upgrade and New Error Logging in MySQL 8.0

Ståle Deraas
MySQL Engineering
Oct 2019

Percona Live Amsterdam



Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Program agenda

Introduction to upgrade

A straight forward upgrade to MySQL 8.0

Upgrade to MySQL 8.0 in detail

Smoother upgrade going forward

Improvements to error logging in MySQL 8.0

ORACLE

Introduction to upgrade

Upgrading in General

Why upgrade the MySQL installation?

- Security concerns
- Performance and Scalability
- New functionality
- Reduce tech debt for the MySQL installation
  - Multiple version upgrade is complex (5.6 -> 5.7 -> 8.0)
  - Eg. deprecated functionality in 5.7, removed in 8.0

Upgrading in General - Biting the Bullet

At some point upgrade needs to happen!

DBA feedback regarding upgrading MySQL

- Reducing risk and cost is key
- Total duration of upgrade should be short
- For customer apps, keep old MySQL behavior by default, change behavior later
- Want to adopt new version gradually
- When switching, downtime should be minimal

ORACLE

Upgrade to MySQL 8.0
The straightforward case

MySQL upgrade 5.7 -> 8.0, smooth sailing!

Read release notes: Conclusion – No problems
Run upgrade_checker

me@siv20$ ./mysqlsh root:@localhost:3307 -e "util.checkForServerUpgrade();"
The MySQL server at localhost:3307 will now be checked for compatibility issues for upgrade to MySQL 8.0...
MySQL version: 5.7.25 - Source distribution

1) Usage of db objects with names conflicting with reserved keywords in 8.0
No issues found

2) Usage of utf8mb3 charset
No issues found

.....
No known compatibility errors or issues for upgrading the target server to MySQL 8 were found.

MySQL upgrade 5.7 -> 8.0, smooth sailing! Con't

Backup your data directory and install MySQL 8.0
Start the MySQL 8.0 server

In-place upgrade, all upgrade processing happens automatically

Inspect the error log:

```
2019-09-24T10:30:27.672011Z 0 [System] [MY-010116] [Server] /exp/mysqld (mysqld 8.0.18) starting as process 23158
2019-09-24T10:30:27.700032Z 1 [System] [MY-011012] [Server] Starting upgrade of data directory.
2019-09-24T10:30:34.341342Z 2 [System] [MY-011003] [Server] Finished populating Data Dictionary tables with data.
2019-09-24T10:30:36.438459Z 5 [System] [MY-013381] [Server] Server upgrade from '50700' to '80018' started.
2019-09-24T10:30:52.447935Z 5 [System] [MY-013381] [Server] Server upgrade from '50700' to '80018' completed.
2019-09-24T10:30:53.034429Z 0 [System] [MY-010931] [Server] /exp/mysqld: ready for connections. Version: '8.0.18'  socket:
'/tmp/mysql.sock'  port: 3306  Source distribution.
2019-09-24T10:30:53.267799Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: '/tmp/mysqlx.sock' bind-
address: '::' port: 33060
```

Verify that apps and services are working as expected

ORACLE

Upgrade to MySQL 8.0 in detail and how the upgrade_checker helps

Goals for the MySQL 8.0 upgrade experience

Upgrade to be faster and with lower risk

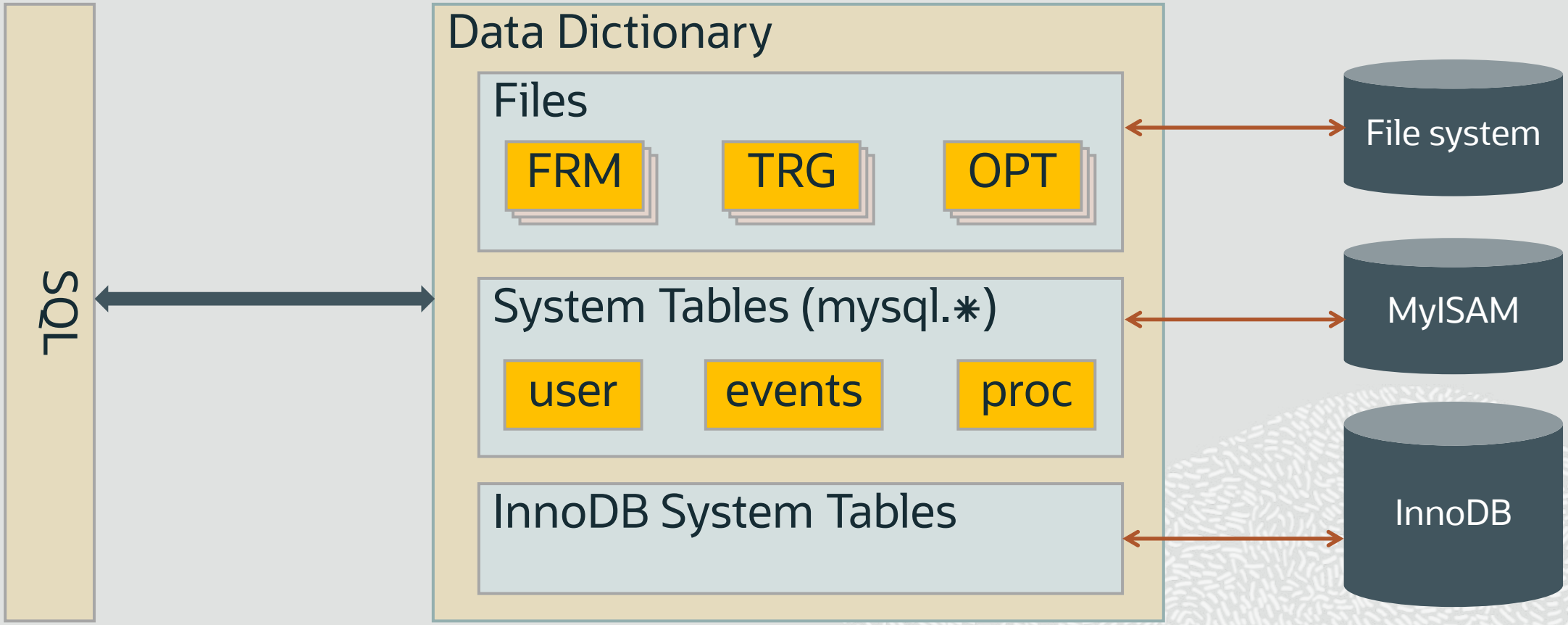
- Eliminate legacy issues with metadata
  - Transition from legacy metadata handling to transactional data dictionary

The upgrade process will produce a consistent data dictionary

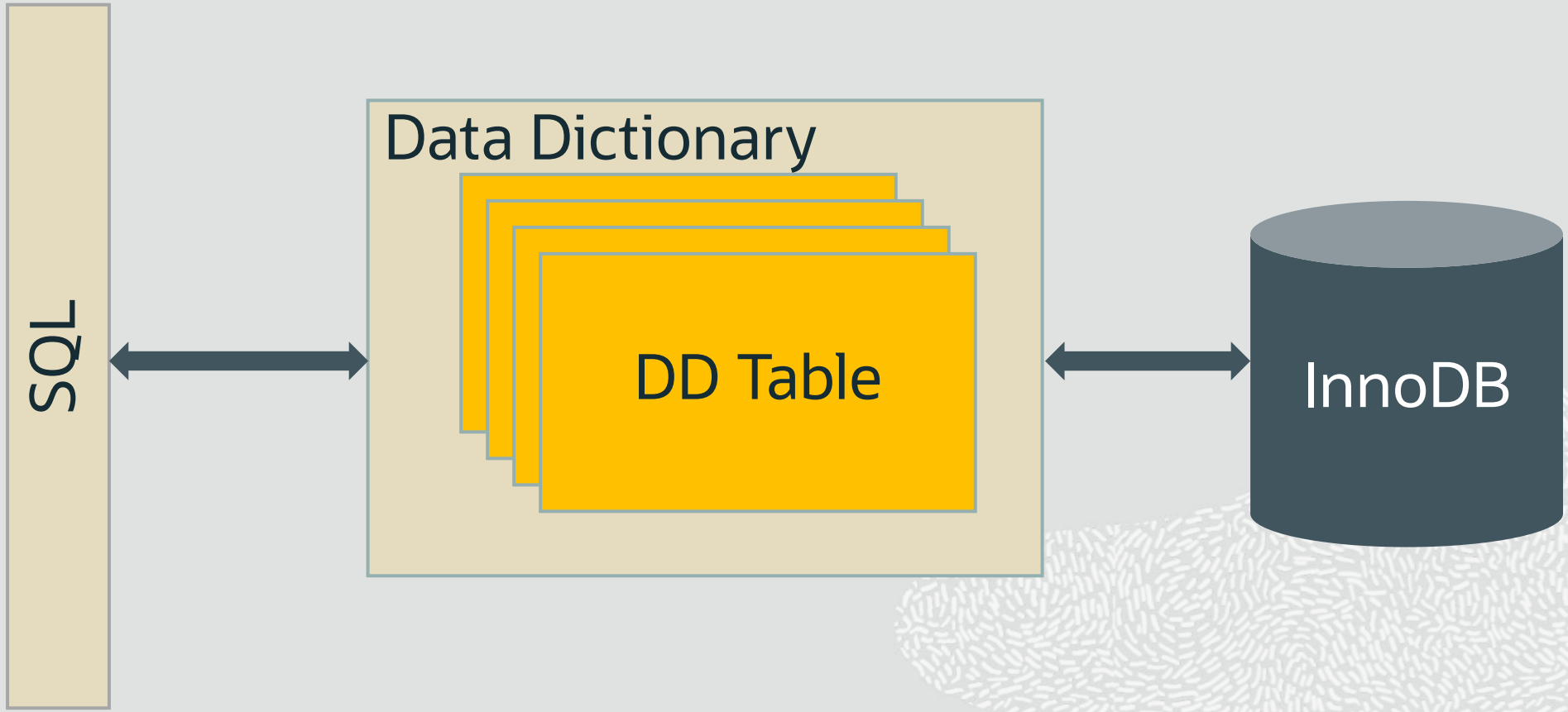
Help DBAs upgrading MySQL

- Better support preparing for the upgrade
  - Added upgrade_checker to the MySQL Shell
- Better support during the upgrade
  - Added upgrade checks to the MySQL server
  - Prohibit legacy issues from entering the MySQL 8.0 metadata store

MySQL Data Dictionary before MySQL 8.0



Transactional Data Dictionary in MySQL 8.0



MySQL 8.0: Relases in 2019

8.0.14 (Jan 2019)

LATERAL DERIVED TABLES
PARALLEL READ OF INDEX
CONSISTENCY LEVELS (GR)

8.0.16 (Apr 2019)

CHECK CONSTRAINTS
AUTO-UPGRADE
AUTO-REJOIN (GR)

8.0.17 (Jul 2019)

CLONE
JSON ARRAY INDEXES
JSON SCHEMA

8.0.18 (Oct 2019)

HASH JOIN
EXPLAIN ANALYZE
...

MySQL 8.0 highlights

Read release notes https://dev.mysql.com/doc/relnotes/mysql/8.0/en/
and https://dev.mysql.com/doc/refman/8.0/en/upgrading.html
Read blogs on https://mysqlserverteam.com

New features in 8.0, and there is a lot of them:
https://mysqlserverteam.com/the-complete-list-of-new-features-in-mysql-8-0/
Deprecations and removals
Defaults changes
upgrade_checker in MySQL shell

MySQL upgrade_checker

- New tool part of MySQL shell
- upgrade_checker checks your MySQL installation readiness for upgrade
- Identify issues preventing upgrade
  - Run the tool on your MySQL installation and make necessary changes
- upgrade_checker is in active development and more checks will be added
  - Part of MySQL update releases

MySQL upgrade_checker, esp 5.7 -> 8.0

- Usage of old temporal type
- Conflicting db object names and reserved keywords
- Usage of utf8mb3 charset
- Use of non-native partitioning
- **Reserved tablenames in mysql schema**
- FK names longer than 64 chars
- Usage of obsolete sql_mode
- ENUM/SET column definitions containing elements longer than 255 characters
- Usage of partitioned tables in shared tablespaces
- Usage of removed functions
- Usage of removed GROUP BY ASC/DESC
- Removed system variables
- System variable with new defaults
- **Inconsistencies after file removal**
- Issues reported by ”check table x for upgrade” command
- New default auth plugin checks

Issues detected by the upgrade_checker: Example

The Transactional Data Dictionary is stored as tables in the mysql schema, and can not conflict with user table names

Detect with SQL, by the upgrade_checker:

```
SELECT TABLE_SCHEMA, TABLE_NAME FROM
INFORMATION_SCHEMA.TABLES
WHERE LOWER(TABLE_SCHEMA) = 'mysql'  and
        LOWER(TABLE_NAME) IN  ( 'catalogs', 'character_sets'....);
```

ACTION: Rename tables with conflicting name

ORACLE

Smoother upgrade going forward

Smoother upgrade going forward

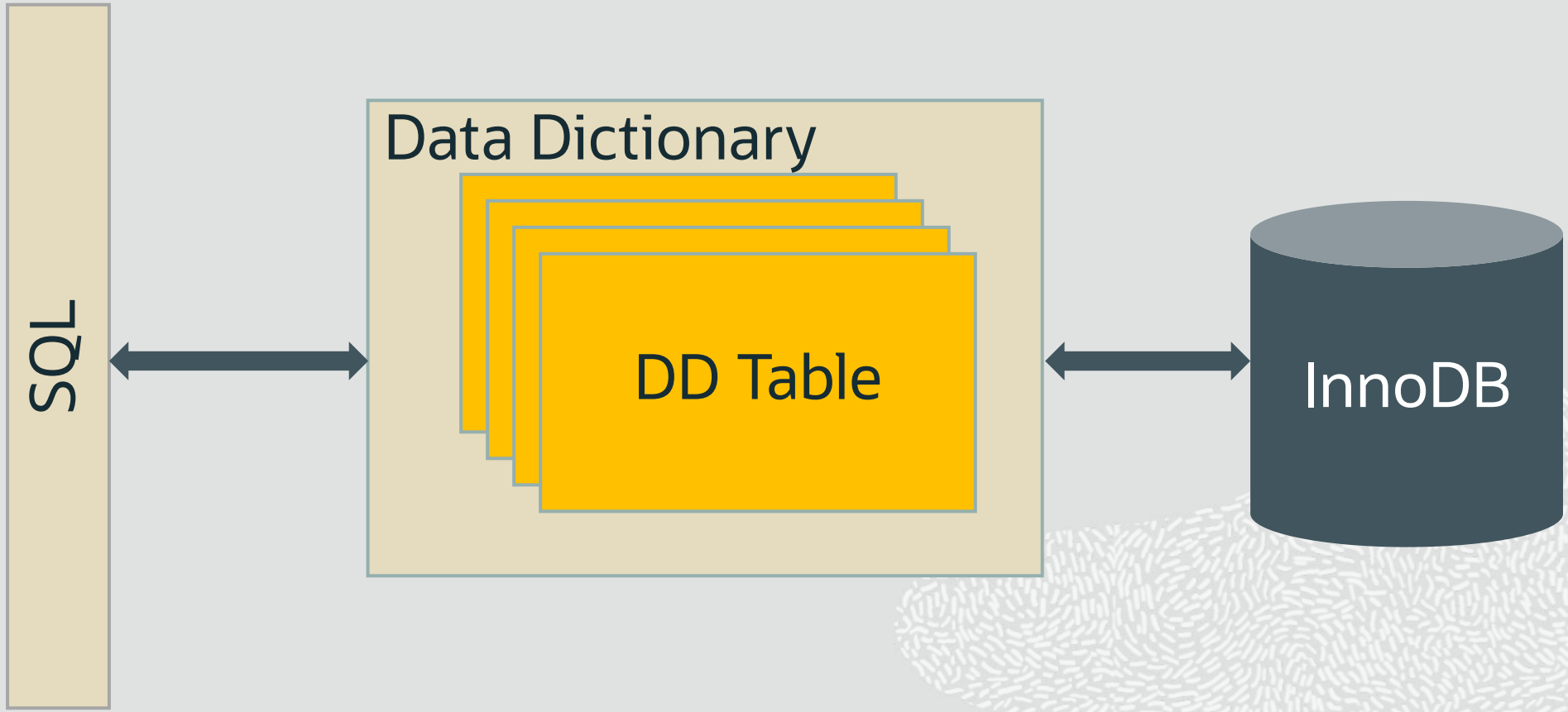
How can we continue improving the upgrade?

- We want to reduce time and risk even further
- Reduce need for manual intervention

Bulk of time spent for in-place upgrade of MySQL

- Harvest metadata for analysis
- Examine all user tables

Transactional Data Dictionary in MySQL 8.0



Upgrade MySQL 8.0 ->

MySQL now stores all metadata in InnoDB (GA)
- Enables fast metadata analysis with SQL-queries

Added metadata for versioning (GA)
- The new mysqld executable knows which version it is upgrading from

Improved protection of metadata, good for security reasons (Partly done)
- Enforced metadata integrity

Remove need for mysql_upgrade client (8.0.16)
- Move functionality to mysqld proper
- Docker/container friendly

Update of help tables (8.0.16)

Upgrade MySQL 8.0 -> the GREAT news con't

The MySQL 8.0 upgrade

1. Stop old MySQL server
2. Change binaries to new MySQL server version
3. Adjust config, my.cnf of new server version
4. Start new MySQL server (once)
   - Analyze metadata and automatically upgrade
   - Speeding up upgrade
   - Container/Docker friendly
5. ~~Run mysql_upgrade to possibly upgrade system tables and user tables~~
   - ~~Potentially time consuming~~
6. ~~Restart MySQL server~~
   - ~~Adds downtime, container/docker unfriendly~~

Upgrade MySQL 8.0 – new option to mysqld

New option --upgrade

--upgrade=AUTO mysqld upgrades anything it determines to be out of date (default option)

--upgrade=NONE mysqld upgrades nothing, exits with an error if anything must be upgraded

Advanced option values

--upgrade=MINIMAL

--upgrade=FORCE

These exist because MySQL currently allows DDL on some system tables

We foresee that MINIMAL and FORCE will go away in a future release

Summary: Upgrade MySQL 8.0 ->

Lowers risk

- The upgrade_checker identifies potential issues
  - In active development
- Metadata integrity

Faster upgrade process

- Fast metadata analysis
- Fast metadata upgrade
- Removal of mysql_upgrade

Simplified upgrade process

- Fewer steps
- Automatic metadata upgrade
- Help tables updated as part of upgrade

ORACLE

Improvements to error logging in MySQL 8.0

Error logging – feedback from community pre MySQL 8.0

- Too verbose by default
- Useful info left out
- Hard to filter
  - Some special filtering options for certain messages
- No identification of subsystem source for error message
- No error codes, so parsing messages is needed to identify the error
- Bootstrap messages might get lost
- Fixed format

Error logging – MySQL 5.7 startup and shutdown output

2019-04-25T14:53:40.062118Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for more details).

2019-04-25T14:53:40.062355Z 0 [Note] --secure-file-priv is set to NULL. Operations related to importing and exporting data are disabled

2019-04-25T14:53:40.062726Z 0 [Note] D:\mysql\mysqld.exe (mysqld 5.7.25-debug) starting as process 8160 ...

2019-04-25T14:53:40.081310Z 0 [Note] InnoDB: !!!!!!!! UNIV_DEBUG switched on !!!!!!!!!

2019-04-25T14:53:40.082415Z 0 [Note] InnoDB: Mutexes and rw_locks use Windows interlocked functions

2019-04-25T14:53:40.083751Z 0 [Note] InnoDB: Uses event mutexes

2019-04-25T14:53:40.084361Z 0 [Note] InnoDB: _mm_lfence() and _mm_sfence() are used for memory barrier

2019-04-25T14:53:40.085487Z 0 [Note] InnoDB: Compressed tables use zlib 1.2.11

2019-04-25T14:53:40.089203Z 0 [Note] InnoDB: Number of pools: 1

2019-04-25T14:53:40.089872Z 0 [Note] InnoDB: Not using CPU crc32 instructions

2019-04-25T14:53:40.093530Z 0 [Note] InnoDB: Initializing buffer pool, total size = 128M, instances = 1, chunk size = 128M

2019-04-25T14:53:40.269813Z 0 [Note] InnoDB: Completed initialization of buffer pool

2019-04-25T14:53:40.338223Z 0 [Note] InnoDB: Highest supported file format is Barracuda.

2019-04-25T14:53:40.363285Z 0 [Note] InnoDB: Log scan progressed past the checkpoint lsn 1317206

2019-04-25T14:53:40.363758Z 0 [Note] InnoDB: Doing recovery: scanned up to log sequence number 1317215

2019-04-25T14:53:40.364156Z 0 [Note] InnoDB: Database was not shutdown normally!

2019-04-25T14:53:40.364542Z 0 [Note] InnoDB: Starting crash recovery.

2019-04-25T14:53:40.725588Z 0 [Note] InnoDB: Removed temporary tablespace data file: "ibtmp1"

2019-04-25T14:53:40.726076Z 0 [Note] InnoDB: Creating shared tablespace for temporary tables

Error logging – MySQL 5.7 startup and shutdown output

2019-04-25T14:53:40.726636Z 0 [Note] InnoDB: Setting file '.\ibtmp1' size to 12 MB. Physically writing the file full; Please wait ...
2019-04-25T14:53:40.864010Z 0 [Note] InnoDB: File '.\ibtmp1' size is now 12 MB.
2019-04-25T14:53:40.878031Z 0 [Note] InnoDB: 96 redo rollback segment(s) found. 96 redo rollback segment(s) are active.
2019-04-25T14:53:40.878520Z 0 [Note] InnoDB: 32 non-redo rollback segment(s) are active.
2019-04-25T14:53:40.889152Z 0 [Note] InnoDB: Waiting for purge to start
2019-04-25T14:53:40.940950Z 0 [Note] InnoDB: 5.7.25 started; log sequence number 1317215
2019-04-25T14:53:40.945300Z 0 [Note] InnoDB: Loading buffer pool(s) from D:\mysql\data\ib_buffer_pool
2019-04-25T14:53:40.945926Z 0 [Note] Plugin 'FEDERATED' is disabled.
2019-04-25T14:53:40.999322Z 0 [Warning] Failed to set up SSL because of the following SSL library error: SSL context is not usable without certificate and private key
2019-04-25T14:53:41.002994Z 0 [Note] Server hostname (bind-address): '*'; port: 3306
2019-04-25T14:53:41.003573Z 0 [Note] IPv6 is available.
2019-04-25T14:53:41.003925Z 0 [Note] - '::' resolves to '::';
2019-04-25T14:53:41.004402Z 0 [Note] Server socket created on IP: '::'.
2019-04-25T14:53:41.015464Z 0 [Note] InnoDB: Buffer pool(s) load completed at 190425 16:53:41
2019-04-25T14:53:41.062328Z 0 [Note] Event Scheduler: Loaded 0 events
2019-04-25T14:53:41.064056Z 0 [Note] D:\mysql\mysqld.exe: ready for connections.
Version: '5.7.25-debug' socket: '' port: 3306 Source distribution
2019-04-25T14:54:09.295492Z 0 [Note] D:\mysql\mysqld.exe: Normal shutdown
2019-04-25T14:54:09.296374Z 0 [Note] Giving 0 client threads a chance to die gracefully

Error logging – MySQL 5.7 startup and shutdown output

2019-04-25T14:54:09.299242Z 0 [Note] Forcefully disconnecting 0 remaining clients
2019-04-25T14:54:09.299522Z 0 [Note] Event Scheduler: Purging the queue. 0 events
2019-04-25T14:54:09.300378Z 0 [Note] Binlog end
2019-04-25T14:54:09.303359Z 0 [Note] Shutting down plugin 'ngram'
2019-04-25T14:54:09.303596Z 0 [Note] Shutting down plugin 'partition'
2019-04-25T14:54:09.303934Z 0 [Note] Shutting down plugin 'BLACKHOLE'
2019-04-25T14:54:09.304263Z 0 [Note] Shutting down plugin 'ARCHIVE'
2019-04-25T14:54:09.304448Z 0 [Note] Shutting down plugin 'PERFORMANCE_SCHEMA'
2019-04-25T14:54:09.304740Z 0 [Note] Shutting down plugin 'MRG_MYISAM'
2019-04-25T14:54:09.304935Z 0 [Note] Shutting down plugin 'MyISAM'
2019-04-25T14:54:09.305134Z 0 [Note] Shutting down plugin 'INNODB_SYS_VIRTUAL'
2019-04-25T14:54:09.305610Z 0 [Note] Shutting down plugin 'INNODB_SYS_DATAFILES'
2019-04-25T14:54:09.306178Z 0 [Note] Shutting down plugin 'INNODB_SYS_TABLESPACES'
2019-04-25T14:54:09.306400Z 0 [Note] Shutting down plugin 'INNODB_SYS_FOREIGN_COLS'
2019-04-25T14:54:09.306625Z 0 [Note] Shutting down plugin 'INNODB_SYS_FOREIGN'
2019-04-25T14:54:09.306837Z 0 [Note] Shutting down plugin 'INNODB_SYS_FIELDS'
2019-04-25T14:54:09.307046Z 0 [Note] Shutting down plugin 'INNODB_SYS_COLUMNS'
2019-04-25T14:54:09.307294Z 0 [Note] Shutting down plugin 'INNODB_SYS_INDEXES'
2019-04-25T14:54:09.307506Z 0 [Note] Shutting down plugin 'INNODB_SYS_TABLESTATS'

Error logging – MySQL 5.7 startup and shutdown output

```
[.....]
2019-04-25T14:54:09.311311Z 0 [Note] Shutting down plugin 'INNODB_CMP_RESET'
2019-04-25T14:54:09.311523Z 0 [Note] Shutting down plugin 'INNODB_CMP'
2019-04-25T14:54:09.311719Z 0 [Note] Shutting down plugin 'INNODB_LOCK_WAITS'
2019-04-25T14:54:09.311933Z 0 [Note] Shutting down plugin 'INNODB_LOCKS'
2019-04-25T14:54:09.312134Z 0 [Note] Shutting down plugin 'INNODB_TRX'
2019-04-25T14:54:09.312330Z 0 [Note] Shutting down plugin 'MEMORY'
2019-04-25T14:54:09.312516Z 0 [Note] Shutting down plugin 'CSV'
2019-04-25T14:54:09.312699Z 0 [Note] Shutting down plugin 'sha256_password'
2019-04-25T14:54:09.312908Z 0 [Note] Shutting down plugin 'mysql_native_password'
2019-04-25T14:54:09.313281Z 0 [Note] Shutting down plugin 'binlog'
2019-04-25T14:54:09.313478Z 0 [Note] Shutting down plugin 'InnoDB'
2019-04-25T14:54:09.313756Z 0 [Note] InnoDB: FTS optimize thread exiting.
2019-04-25T14:54:09.314817Z 0 [Note] InnoDB: Starting shutdown...
2019-04-25T14:54:09.419314Z 0 [Note] InnoDB: Dumping buffer pool(s) to D:\mysqldata\ib_buffer_pool
2019-04-25T14:54:09.420990Z 0 [Note] InnoDB: Buffer pool(s) dump completed at 190425 16:54:09
2019-04-25T14:54:10.518640Z 0 [Note] InnoDB: Shutdown completed; log sequence number 1317243
2019-04-25T14:54:10.519125Z 0 [Note] InnoDB: Removed temporary tablespace data file: "ibtmp1"
2019-04-25T14:54:10.521772Z 0 [Note] D:\mysql\mysqld.exe: Shutdown complete
```

Error logging – rewritten in MySQL 8.0

- Pluggable, so more flexibility wrt to log writers and filtering
- All messages to error log have unique error codes, starting from 10000
- New «system» message category, used for messages that are not errors, yet server state changing events that will always be visible in the error log
- Added info, like versioning info at shutdown, who initiated the shutdown ++
- Filtering
  - Default internal: Verbosity option + Suppress sysvar option that can take list of error codes
  - Optional advanced filtering component «Dragnet»
- Log writers (sinks)
  - Default classic
  - Optional JSON
  - Optional syseventlog

Error logging – MySQL 8.0 startup and shutdown output

2019-09-25T14:36:57.642120Z 0 [System] [MY-010116] [Server] D:\mysql\mysqld.exe (**mysqld 8.0.18**) starting as process 14640

2019-09-25T14:37:06.932404Z 0 [System] [MY-010931] [Server] D:\mysql\mysqld.exe: ready for connections. Version: '8.0.18' socket: '' port: 3306 Source distribution.

2019-09-25T14:37:07.014038Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060

[...]

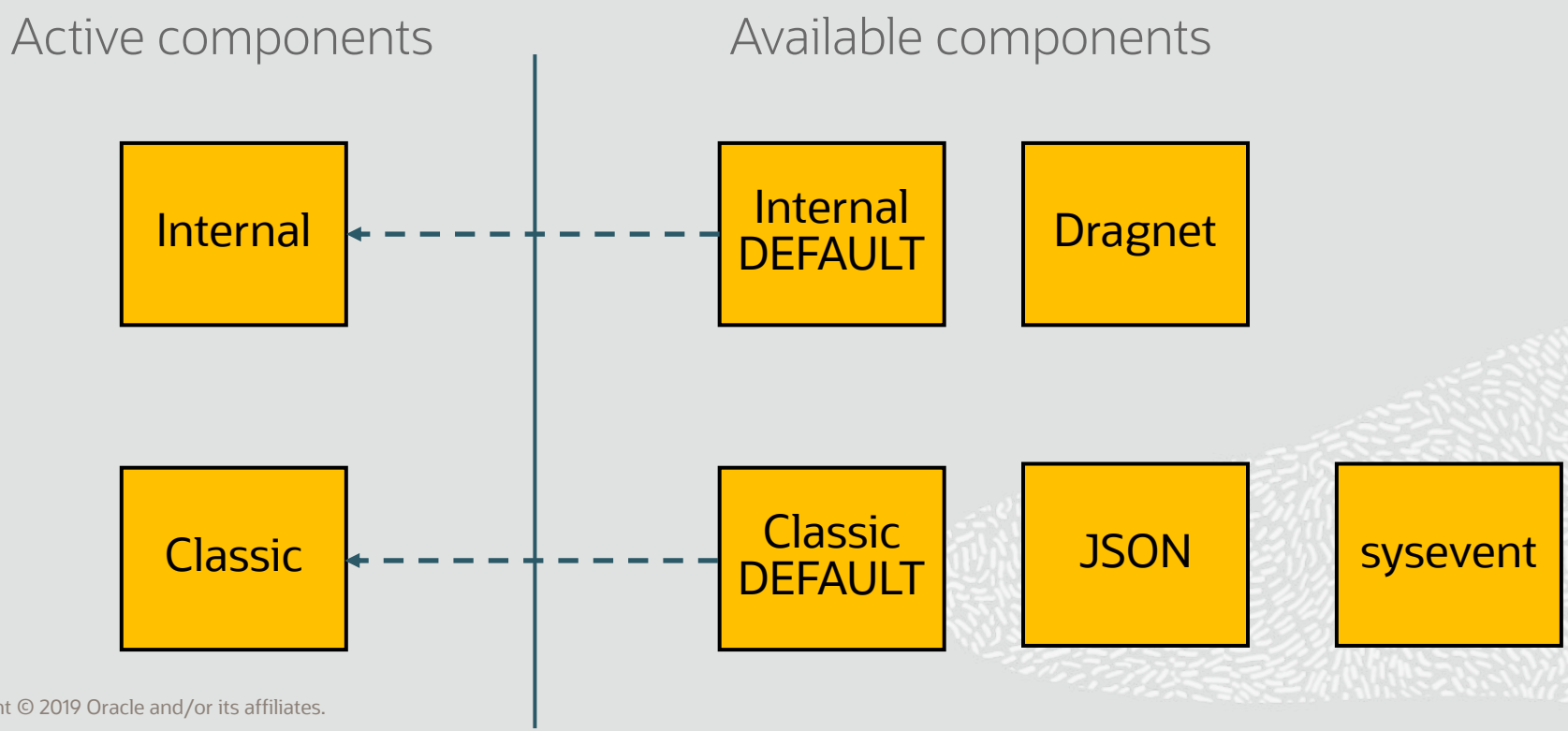
2019-09-25T14:37:33.192903Z 8 [System] [MY-013172] [Server] Received SHUTDOWN from **user root.** Shutting down mysqld (Version: 8.0.18).

2019-09-25T14:37:33.194325Z 0 [System] [MY-013105] [Server] D:\mysql\mysqld.exe: Normal shutdown.

2019-09-25T14:37:33.661078Z 0 [System] [MY-010910] [Server] D:\mysql\mysqld.exe: Shutdown complete (**mysqld 8.0.18**) Source distribution.

Error logging – pluggability in MySQL 8.0

One active filter component and one active writer/sink component



Error logging – Filtering in MySQL 8.0

System variables for default filtering

- «log_error_verbosity»
  - Default value 2 for which SYSTEM + ERROR + WARNING messages are printed
- «log-error-suppression-list» (MySQL 8.0.13)
  - List of error codes to suppress

Example: Log all messages, and suppress some

- -SET GLOBAL log_error_verbosity=3
- -SET GLOBAL log_error_suppression_list='ER_PARSER_TRACE, MY-010001, 10002'

Error logging – Advanced filtering in MySQL 8.0

log_filter_dragnet

-Filter language, rich in nature, see https://dev.mysql.com/doc/refman/8.0/en/error-log-rule-based-filtering.html

-Filter rule fields: time, err_code, prio/severity, subsystem ++

-Filter actions: drop, throttle + set/unset field

Loaded as a component

Changing the error logging stack to use the dragnet filtering component:

```
INSTALL COMPONENT 'file://component_log_filter_dragnet';
SET GLOBAL log_error_services = 'log_filter_dragnet; log_sink_internal';
```

Error logging – Advanced filtering in MySQL 8.0

Filter to print maximum one INFORMATION message per min:

```
SET GLOBAL dragnet.log_error_filter_rules = 'IF prio == INFORMATION THEN throttle
1/60.';
```

Filter to print a given message no more than 5 times per 10 min:

```
SET GLOBAL dragnet.log_error_filter_rules = 'IF err_code == MY-010035 THEN throttle
5/600.'
```

Combining filters:

```
SET GLOBAL dragnet.log_error_filter_rules = 'IF prio == INFORMATION THEN throttle
1/60.
IF err_code == MY-010035 THEN throttle 5/600.'
```

Error logging – Writers in MySQL 8.0

Default writer is «classic», log to file
Optional writer components, JSON and syseventlog

Configuring the syseventlog writer:

-INSTALL COMPONENT 'file://component_log_sink_syseventlog';
-SET GLOBAL log_error_services = 'log_filter_internal; log_sink_syseventlog';

Note that SYSTEM messages are converted to INFORMATION messages in syslog and Windows eventlog

Error logging – JSON writer output in MySQL 8.0

The JSON logger logs all available information


```
{ "prio" : 0,
 "err_code" : 10116,
 "source_line" : 4340,
 "source_file" : "mysqld.cc",
 "function" : "init_common_variables",
 "msg" : "D:\\mysql\\mysqld.exe (mysqld 8.0.18) starting as process 11876",
 "time" : "2019-09-26T14:51:29.008063Z",
 "err_symbol" : "ER_STARTING_AS",
 "SQL_state" : "HY000",
 "subsystem" : "Server",
 "buffered" : 1556290289008063,
 "label" : "System"
}
```

Error logging – summary MySQL 8.0

How is the feedback from the community addressed?

- Too verbose by default (GA)
- Useful info left out (GA ->)
- Hard to filter (GA + 8.0.13)
  - Some special filtering options for certain messages
- No identification of subsystem source for error message (GA)
- No error codes, so parsing messages is needed to identify the error (GA)
- Bootstrap messages might get lost (GA ->)
- Fixed format (GA)

Try for yourself!

Downloadable 8.0.17 and soon 8.0.18!

http://dev.mysql.com

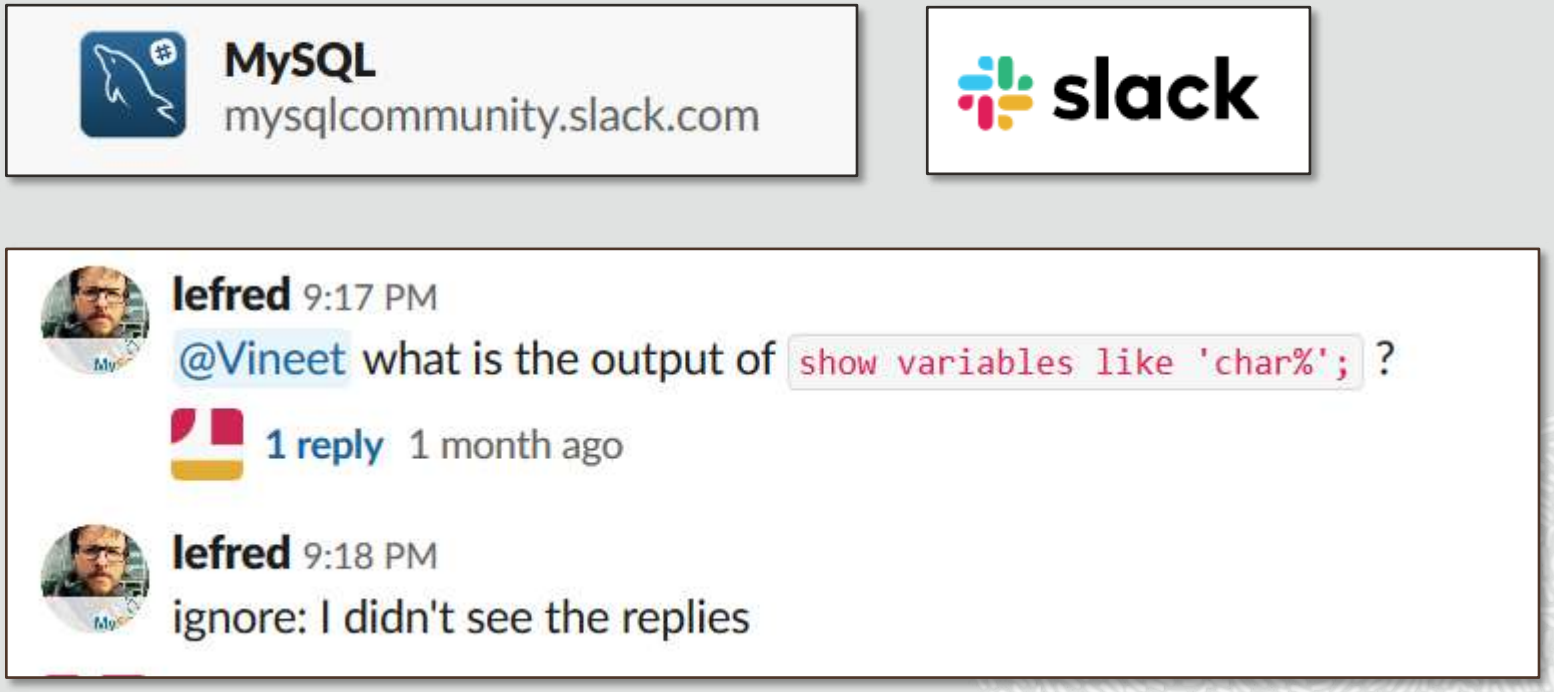
Enjoy and give us your feedback!

Thank you for listening

http://mysqlserverteam.com

Join us on MySQL Community Slack

https://lefred.be/mysql-community-on-slack/



Follow us on Social Media

https://www.facebook.com/mysql

https://twitter.com/mysql

https://www.linkedin.com/company/mysql

Thank you!

Ståle Deraas

Software Development Director
MySQL team, Oracle

MySQL Upgrade Resources

https://dev.mysql.com/doc/refman/8.0/en/upgrading-strategies.html#upgrade-prerequisites
https://dev.mysql.com/doc/refman/8.0/en/upgrading-from-previous-series.html
https://dev.mysql.com/doc/refman/8.0/en/upgrading-what-is-upgraded.html
https://mysqlserverteam.com/inplace-upgrade-from-mysql-5-7-to-mysql-8-0/
https://mysqlserverteam.com/upgrading-to-mysql-8-0-here-is-what-you-need-to-know/
https://mysqlserverteam.com/mysql-shell-8-0-4-introducing-upgrade-checker-utility/
https://mysqlserverteam.com/upgrading-to-mysql-8-0-with-spatial-data/
https://mysqlserverteam.com/whats-new-in-mysql-8-0-generally-available/
https://mysqlserverteam.com/upgrading-your-mysql-server-farm/