Agile Environments and DBAs

Laine Campbell, PalominoDB
Sheeri Cabral, The Pythian Group

Definition of Agile

From m-w.com:
Main Entry: ag·ile
Pronunciation: \ˈa-jəl, -ˌjī(-ə)l\
Function: adjective Etymology:Middle French, from Latin agilis, from agere to drive, act — more at agent Date:1581

1 : marked by ready ability to move with quick easy grace <an agile dancer>

2 : having a quick resourceful and adaptable character <an agile mind>

Definition of Agile

From Wikipedia:

Agile methodologies generally promote a project management process that encourages frequent inspection and adaptation,

Definition of Agile

From Wikipedia:

Agile methodologies generally promote a project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability,

Definition of Agile

From Wikipedia:

Agile methodologies generally promote a project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices that allow for rapid delivery of high-quality software,

Definition of Agile

From Wikipedia:

Agile methodologies generally promote a project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices that allow for rapid delivery of high-quality software, and a business approach that aligns development with customer needs and company goals.

Agile Environments

Rapid Delivery

Time-driven: Release what's done

High number of changes and frequent QA

Functionality from the beginning

Prepare not Predict

What a DBA does, part 1

Defines architecture
Schema design and implementation

Database Configuration

What a DBA does, part 2

Database Security

Backup and Recovery

Data integrity

What a DBA does, part 3

Developer support

Tuning and query optimization

May have QA duties

An Agile Database System

Frequent change, little downtime

Rolling implementations, fallbacks

Requirements change often, unpredictably

Well documented, has change management

Supporting Rapid Change

Adaptive vs. Predictive

Containment

Contingencies

Online Operations In MySQL

Adding/dropping tables

Adding/dropping databases

Dynamic global variables

Online Operations in 5.1

Adding values at end of ENUM/SET

Column rename

Column default value change

Starting and stopping logging

Frequent Production Changes

What is the impact?

Can the code handle it?

New fields, tables, views
    naming conventions, plans

Frequent Production Changes

Cache changes / invalidations

Data amount changes optimization

Catch changes done "out of process"

Frequent Production Changes

Easily create new instances

How do you rollback?

What is your DR plan?

Master/Slave Replication

Promotion procedures

Rolling implementations

Slave cold cache

Does not scale writes

Master/Master Replication

Does not scale writes

App needs to handle live traffic switches

Insert Buffer Merges

sql_log_bin=0

Changing server-ids

Data Synchronization

Non deterministic changes

Statement based vs. Row based

Keep an eye on drift

mk-table-checksum

Splitting Data

ID	Last	First	State	Color
1	Cabral	Sheeri	MA	Blue
2	Campbell	Laine	CA	Red
3	Cosloy	Barry	CT	Brown
4	Wasserman	Maya	NJ	Purple

Vertically Splitting Data

ID	Last	First	State	Color
1	Cabral	Sheeri	MA	Blue
2	Campbell	Laine	CA	Red
3	Cosloy	Barry	CT	Brown
4	Wasserman	Maya	NJ	Purple

Vertically Splitting Data

ID	Last	First	State	Color
1	Cabral	Sheeri	MA	Blue
2	Campbell	Laine	CA	Red
3	Cosloy	Barry	CT	Brown
4	Wasserman	Maya	NJ	Purple

ID	Last	First
1	Cabral	Sheeri
2	Campbell	Laine
3	Cosloy	Barry
4	Wasserman	Maya

ID	State	Color
1	MA	Blue
2	CA	Red
3	CT	Brown
4	NJ	Purple

Horizontally Splitting Data

ID	Last	First	State	Color
1	Cabral	Sheeri	MA	Blue
2	Campbell	Laine	CA	Red
3	Cosloy	Barry	CT	Brown
4	Wasserman	Maya	NJ	Purple

Horizontally Splitting Data

ID	Last	First	State	Color
1	Cabral	Sheeri	MA	Blue
2	Campbell	Laine	CA	Red
3	Cosloy	Barry	CT	Brown
4	Wasserman	Maya	NJ	Purple

ID	Last	First	State	Color
1	Cabral	Sheeri	MA	Blue
2	Campbell	Laine	CA	Red
3	Cosloy	Barry	CT	Brown

ID	Last	First	State	Color
4	Wasserman	Maya	NJ	Purple

Splitting Data

Data design

Code design

Upgrade by sections

Prepare, Not Predict

Avoid generic fields

Flexible datatypes/sizing

Revisit schema, query plans and indexing

Prepare, Not Predict

Defragment!!!

Avoid/revisit index hints and other hacks

Know what queries you are running (mk-query-digest)

Plan for data management

Finding Problems

Query/database errors

Proactively monitor queries with mk-query-digest

Slow query log aggregation/review

Finding Problems

Application debug mode

Constant feedback to development

Monitor for unplanned schema changes

Testing/QA

Do not have the same problem twice

Performance Testing

Schema diff/storage/reversion

Release notes

People Skills

Be a part of the full lifecycle:
- design
- develop
- test
- implement

Set and meet expectations

encourage discipline

People Skills

Encourage communication and trust

Make others want to come to you for review/questions

Be efficient

Maatkit Query Digest Output (1)

```
# main:4218 28360 Getting a dbh from processlist for --processlist
# 230ms user time, 20ms system time, 14.94M rss, 132.49M vsz
# Overall: 95 total, 34 unique, 8.64 QPS, 2.08x concurrency _______________
#                    total      min      max      avg      95%  stddev  median
# Exec time            23s    102ms       4s    241ms    640ms     67ms    105ms
# Lock time              0        0        0        0        0        0        0
# Time range        1239991238 to 1239991249

# Query 1: 0 QPS, 0x concurrency, ID 0x11B5A1D8DF88E882 at byte 0 _________
#              pct   total      min      max      avg      95%  stddev  median
# Count          2        2
# Exec time     25       6s       2s       4s       3s       4s       2s       3s
# Lock time      0        0        0        0        0        0        0        0
# Users          xxxxxx
# Databases      1 xxxxxxxxx
# Time range 1239991244 to 1239991244
# Query_time distribution
#   10ms
# 100ms
#    1s  ############################################################
#   10s+
```

Maatkit Query Digest Output (2)

```
# Tables
#    SHOW TABLE STATUS FROM `break_production` LIKE 'videos'\G
#    SHOW CREATE TABLE `break_production`.`videos`\G
#    SHOW TABLE STATUS FROM `break_production` LIKE 'approvals'\G
#    SHOW CREATE TABLE `break_production`.`approvals`\G
#    SHOW TABLE STATUS FROM `break_production` LIKE 'feedings'\G
#    SHOW CREATE TABLE `break_production`.`feedings`\G
#    SHOW TABLE STATUS FROM `break_production` LIKE 'feeds'\G
#    SHOW CREATE TABLE `break_production`.`feeds`\G
#    SHOW TABLE STATUS FROM `break_production` LIKE 'customer_content_providers'\G
#    SHOW CREATE TABLE `break_production`.`customer_content_providers`\G
# EXPLAIN
SELECT videos.* FROM `videos`  INNER JOIN approvals ON videos.id=approvals.video_id INNER JOIN
feedings on videos.id=feedings.video_id INNER JOIN feeds on feeds.id=feedings.feed_id INNER JOIN
customer_content_providers ON
feeds.content_provider_id=customer_content_providers.content_provider_id WHERE
(feeds.code='all_videos'
                    AND approvals.status = 1
                    AND videos.transcoding_status = 1
                    AND customer_content_providers.customer_id=31
                    AND videos.publicated_at < '2009-04-17 11:00:42'
                    AND (videos.expiration_date is null or videos.expiration_date > '2009-04-17 11:00:42'))
ORDER BY publicated_at desc LIMIT 10, 5\G
```

Maatkit Query Digest Output (3)

(taken from maatkit online docs)

```
    # Review information
# comments: really bad IN() subquery, fix soon!
# first_seen: 2008-12-01 11:48:57
# jira_ticket: 1933
# last_seen: 2008-12-18 11:49:07
# priority: high
# reviewed_by: xaprb
# reviewed_on: 2008-12-18 15:03:11
```

Thank you!

Questions

Comments

Feedback

Laine - laine@palominodb.com

Sheeri - cabral@pythian.com