EffectiveMySQL.com

Its all about Performance and Scalability

Successful Scalability Principles - Part I

Ronald Bradford
http://ronaldbradford.com

2011.05

OUTLINE

Necessary Principles

- System Architecture
- Data Availability
- Best Practices
- Being proactive

1

ACTION

Integrated monitoring and instrumentation

EXAMPLE

- **Question:**
- How do you know when your server is down or not accessible?
- **Answer:**
- The users will let us know.

HOW

- Monitoring/Alerting
  - Graphical
  - Historical
  - Necessary
  - Generally missing/incomplete
  - Useless for real-time analysis

REACTIVE

HOW

- Dashboard
  - The state of NOW
  - Near real-time statistics
  - Sampling at 1s/3s/5s
    - e.g. 0.1% of throughput

PROACTIVE

http://rb42.com/monitoring-needs

HOW

PROACTIVE

- Instrumentation
  - Important to business viability
    - e.g. orders per minute (KPI)
    - page load time
  - Seamless implementation
    - i.e. no code changes to view real-time
    - extensible

WHY

Monitoring +
Instrumentation =
Preemptive Analysis

HOW NOT TO

HOW TO

< 1 second

WHY

- Preemptive Analysis
  - Fail early
  - Faster user response
  - Less physical resource impact

TIP

- Have a status website
  - allow for comments (e.g. blog)
- Have a public dashboard website

- **Host them somewhere else!**

ACTION

Seamless automated server deployment

EXAMPLE

Actual Client

- 100+ production database servers
- Full-time DBAs
- No automated software installation
- No single documented installation process
- No version control
- No auditing/verification

HOW

Basics

- Version Control
- Automated Build & Release
- Integrated Monitoring

HOW

Intermediate

- Health Checks
- Runtime config file management
- Benchmarking

HOW

Advanced

- Server request queue
- Automated discovery
- Proactive scaling (up & down)

GOAL

- Consistent and timely provisioning
- Unattended installation with correct and reproducible configuration
- Automated discovery

3

EffectiveMySQL.com - Its all about Performance and Scalability

ACTION

Disaster is inevitable

EffectiveMySQL.com - Its all about Performance and Scalability

EXAMPLE

Question:

Have you ever performed a database recovery?

Answer:

No, why?

EffectiveMySQL.com - Its all about Performance and Scalability

EXAMPLE

Consultant:

Do you know that your daily backups only recover the data up to that time (e.g. 1 am). You know you have lost all your sales(aka $2 million sales today) and data changes since then.

Customer:

No, I didn't know that.

HOW

- Have a DR plan
  - Documented
  - Tested
  - Timed
  - Verified

HOW

- Test under production conditions
  - System Load
  - Database Size
  - End to End

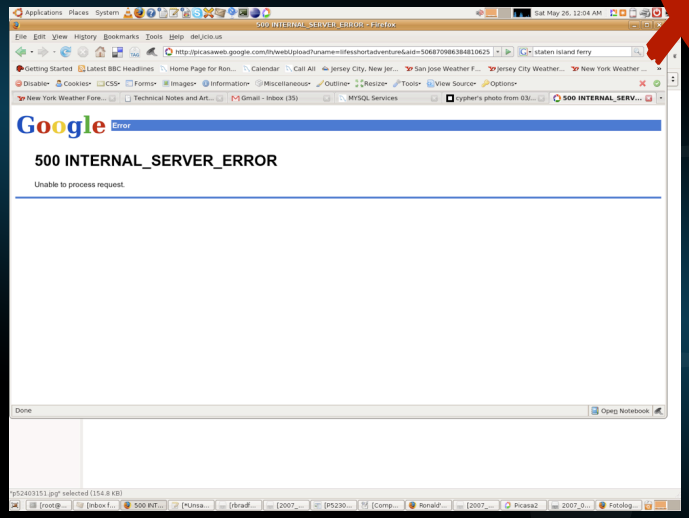
GOAL

- Know and practice for disaster
- Provide confidence to management
- Be as confident about your recovery as your scalability

SIDEBAR!

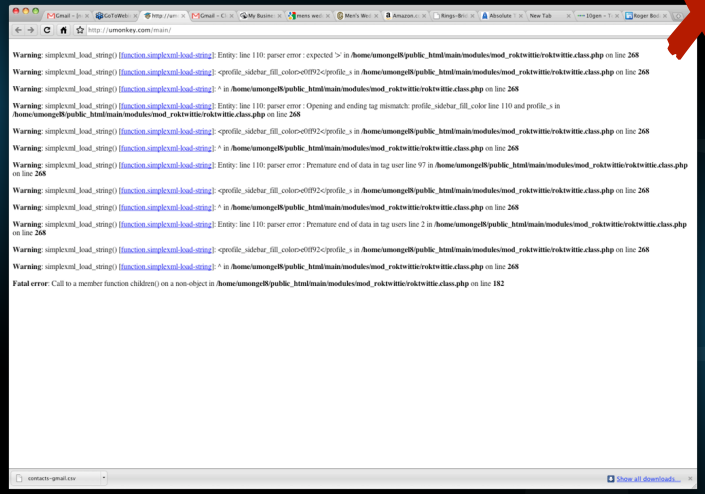
What does your website look like when it's down?

SIDEBAR



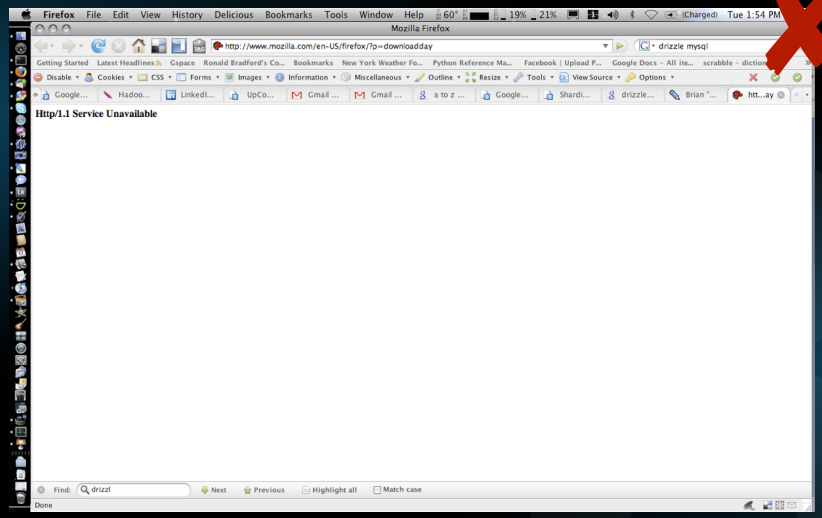
EffectiveMySQL.com - Its all about **Performance** and **Scalability**

SIDEBAR



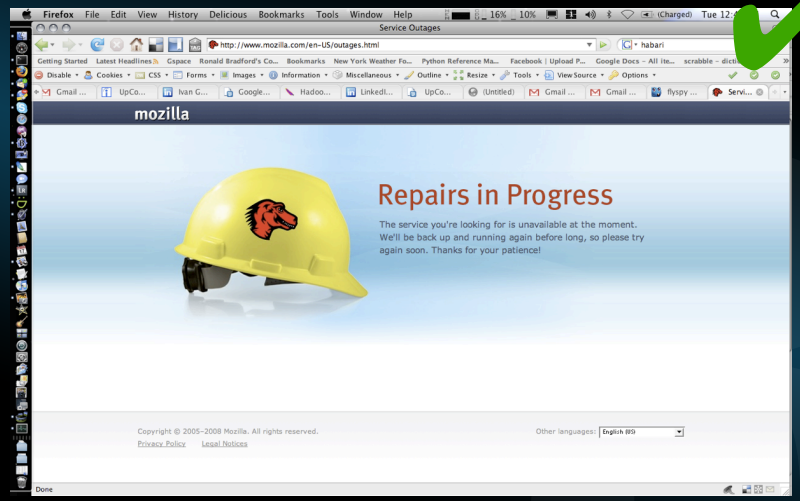
EffectiveMySQL.com - Its all about **Performance** and **Scalability**

SIDEBAR

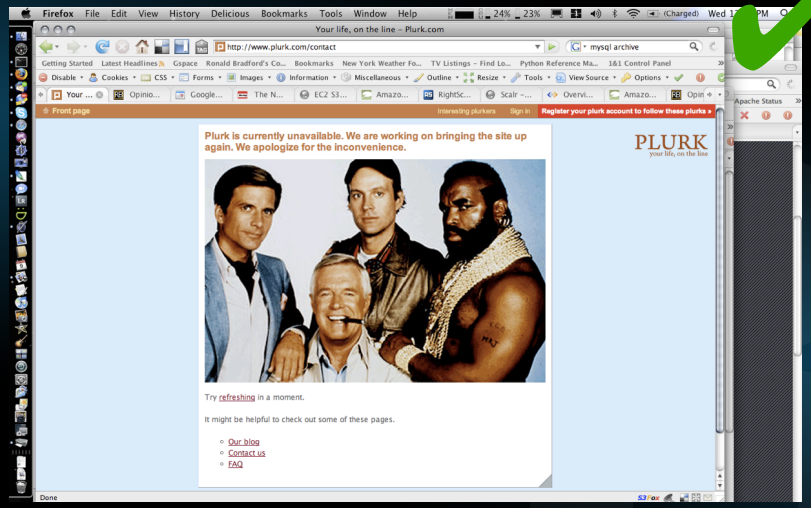


EffectiveMySQL.com - Its all about **Performance** and **Scalability**

SIDEBAR



SIDEBAR



4

ACTION

Application

Programming

Interface

EXAMPLE

- Public API
- Application web site
- Batch processes

3 access paths to same data with different business rules

WHY

- One code path for business functionality
- Implied business documentation
- Enforced data exchange standard
- **Testability**

GOAL

- Technology independence
- Business specification dependence (API)
- Stress testable

REQUIREMENTS

- Input Specification
- Output Specification
- Standard for data communication
- Throughput needs
- Latency requirements

SIDEBAR!

Testing is not about what works, it's about breaking your software

ACTION

Support different levels of data availability

HOW

Data can be:

- Read & Write
- Read
- Cached
- No Access

EXAMPLE

Development Team:

We need a maintenance window for software upgrades and new releases.

EXAMPLE

Management Team:

No Downtime

EXAMPLE

Development Team:

But we need this to fix problems and improve performance.

EXAMPLE

Management Team:

No Downtime

EXAMPLE

What is your definition of no downtime?

WHY

- Support downtime
  - Software upgrades
- Support disasters
- Reduce high load
- Support partial functionality

GOAL

- Meet the business needs of availability
- There are ways to give greater perception of availability

ACTION

Support different scalability principles

HOW

- Read Scalability
- Write Scalability
- Caching

HOW

- Depends on your R/W ratio
- Depends on your business needs
- Depends on rate of data change (caching)

ISSUES

- Consistency
- Data Interoperability

GOAL

- Support large scale growth with appropriate software architecture
- Minimize human interaction
- Ongoing review and improvement

SIDEBAR!

Availability options & scalability principles = scalable datastore

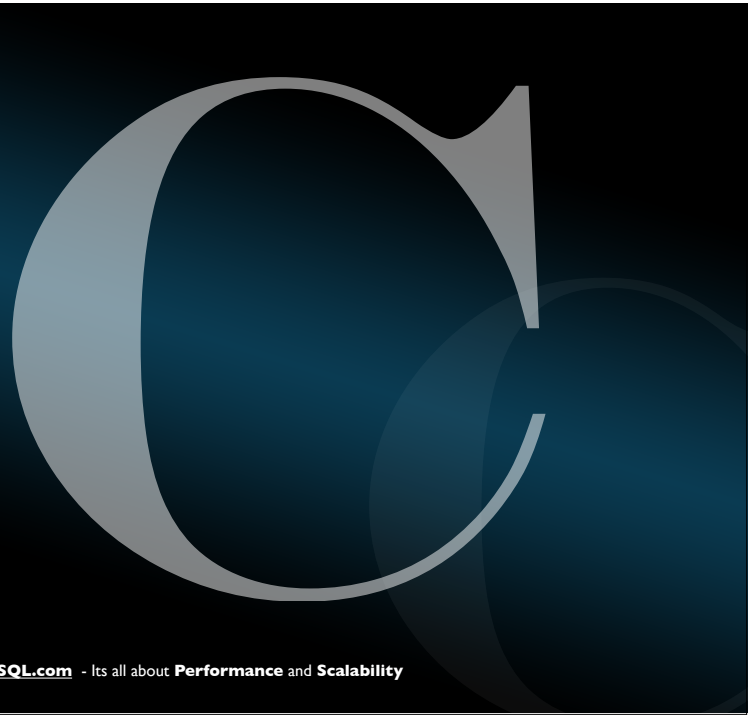
WHAT'S NEXT

I only had one hour!

WHAT'S NEXT

- Reduce deployment time
- The right business metrics
- Team dynamics
- Managing risk
- Continual improvement
- Data interoperability
- ...

- **Asynchronous**
- **Write once data**
- **Consistency**



CONCLUSION

- Monitoring. Before, during and after NOW.
- You may not be able to predict the future but you can preempt the future.
- Operate below 90%. That 10% is your insurance policy. Invest in insurance.

CONCLUSION

- Does your business exist without your data?
- How long can your site be unavailable before customers go elsewhere?

CONCLUSION

- Support 3 levels of real time data access
  - Read/Write, Read and no access
- Support 3 different aspects of scalability
  - Read, Write and Caching

CONCLUSION

- If you can't drive without a mouse, find somebody that can.

REFERENCE

Presentation

- The most common MySQL scalability mistakes, and how to avoid them.

REFERENCE

1. My website is slow?
2. I want to add new H/W. How do I change my application to support this?
3. The database is slow. My webpage takes five seconds to load.
4. Why is my database executing so many qps?
5. My server has crashed with a hard drive failure
6. My replication slave can't keep up?
7. I can't access my website?
8. 8. Why is my database so large?
9. My website seems to freeze or responds randomly?


$Em=ps^n$
Ronald Bradford
http://effectiveMySQL.com