



# **СЪХРАНЯВАНЕ НА ДАННИ. АВАРИИ НА ДИСКОВЕТЕ. ВЪЗСТАНОВЯВАНЕ ОТ АВАРИИ**

**Пламен Николов, СУ ФМИ**

# Резюме

- ✓ Факти за MySQL
- ✓ Специфични възможности
- ✓ Съхраняване на данни
- ✓ Видове хранилища
- ✓ Групиране – Clustering
- ✓ Справяне с аварии
- ✓ Възтановяване от аварии на дисковете

# Факти за MySQL

MySQL е най-популярната система за управление на бази данни с отворен код. Със своите над 100 милиона копия до момента, надеждност, бързина и улеснено ползване, системата е предпочитана за интернет приложения, Web 2.0 , SaaS, ISV, телекомуникационни компании и много други.

Множество от най-големите световни организации разчитат на MySQL за управление на техните данни: **Amazon, Flickr, Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, Booking.com.**

MySQL поддържа интерфейси за програмиране под множество езици - C/C++, Eiffel, Java, Lisp, Perl, Python, Ruby, но най-често се използва в комбинация с езика PHP. Интерфейс тип ODBC наречен MyODBC позволява на други програмни езици, които поддържат ODBC интерфейс, да комуникират с MySQL база данни, например.

# ВЪЗМОЖНОСТИ

Последната официална версия е от началото на 2009 – MySQL 5.1.32

- ✓ Поддръжка на SSL
- ✓ Независими хранилища (storage engines)
- ✓ Пълна поддръжка на Unicode
- ✓ Транзакции с InnoDB, BDB и Cluster хранилища
- ✓ Поддържа X/Open XA (разпределена обработка на транзакции)
- ✓ Кеширане на запитвания (Query caching)
- ✓ Възстановителни точки (savepoints)
- ✓ Пълнотекстово индексване и търсене
- ✓ Клъстър - разпределена обработка без споделен ресурс чрез **MySQL Cluster**

Платформи: AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows.

# Съхранение на данни

MySQL използва за съхранение на данните така наречените хранилища (Storing engines). Организацията на информацията в тях варира в зависимост от вида на хранилището. В една база данни може да има таблици с различни типове хранилища. По този начин има възможност за избор на най-подходящата комбинация за постигане на максимално бързодействие и висока степен на надеждност.

Различните видове хранилища са разработени за да покриват максимално нуждите при съхранението на съответно различните типове информация.

# Видове хранилища

**MyISAM** – Хранилището което се използва по-подразбиране. Характеризира се с много високо бързодействие и е предпочитан при разработката на уеб приложения, складове на данни и др.

Една MyISAM таблица се съхранява върху диска разделена на три файла. Имената на файловете започват с името на таблицата и имат следните разширения:

- .frm – съхранява формата на таблицата
- .myd – съхранява данните
- .myi – съхранява индексите

## Особености:

- Поддръжка на конкурентно добавяне на записи
- BLOB и TEXT могат да бъдат индексирани
- Не осигурява независимост на данните при транзакции
- Не осигурява референтна цялост

# Видове хранилища

**InnoDB** – Разработен е от Innobase Oy (компания собственост на Oracle). Създаден е с цел подобрене на бързодействието при работата с огромно количество информация. Процесорната ефективност при InnoDB вероятно не може да се сравнява с нито една друга дисково базирана релационна система за управление на база от данни. Таблиците и индексите се съхраняват на няколко файла (или дялове от диска). Таблиците може да са дори по-големи от поддържаения размер от операционната система.

## Особености:

- Независимост на данните при транзакции
- Възможност за връщане на предишно състояние на данните
- Възможност за възтанояване на данните
- Осигурява референтна цялост
- Намален брой на I/O дискови операции

# Видове хранилища

**Memory** – Съхранява цялата информация в RAM паметта за максимално бързодействие. Хранилището е по-познато още като HEAP. Всяка таблица която ползва това хранилище се представя чрез един файл. Името на файле е името на таблицата и разширение **.frm**. Таблиците се съхраняват директно в паметта. Използват се хеш индекси по подразбиране, това ускорява повишава допълнително бързодействието и прави хранилището подходящо за съхранение на временни таблици. Освен хеш индекси, поддържат се и индекси представени с двойчни дървета(B-Tree). Таблицы ползващи това хранилище не може да имат колони от тип **BLOB** или **TEXT**. През цялото си време на съществуване никога данните в таблиците не се съхраняват на диска.

Разбира се, когато сървърът се рестартира, данните в таблиците се губят.



# Видове хранилища

**Archieve** – Осигурява идеално решение за съхранение и четене на големи по обем данни на ограничено дисково пространство. Хранилището поддържа **INSERT** и **SELECT**, но е без **DELETE**, **REPLACE** или **UPDATE**. Също така използва методът на заключване на ред от таблицата при работа с него (row-locking).

Данните се представят чрез файл с разширение **.frm** и **.arz**, съответно за структурата и информацията. По време на оптимизационния процес може да се появят и файлове с разширение **.arn**.

# Видове хранилища

**Merge** – известен още като MRG\_MyISAM, представлява колекция от еднотипни MyISAM таблици която може да се използва като една единствена таблица. Таблиците трябва да имат едни и същи колони и индекси, както и еднаква подредба при последователността на колоните.

При създаване на таблица, ползваща това хранилище, се правят два файла. Файловете са с името на таблицата и разширения съответно **.frm** – съхраняващ формата на таблицата и **.mrg** съдържащ имената на съответно свързаните таблици.

Алтернатива на **Merge** са таблиците разделени на дялове. Разделянето на таблицата на дялове(няколко файла) позволява някои операции да се извършват доста по-ефективно.

# Видове хранилища

**NDBCLUSTER** — Повече познато като (NDB), това хранилище е необходимо на приложенията които се стремят към много висока надеждност и стабилост. Хранилището не се поддържа пряко по MySQL 5.1, има специален продукт наречен MySQL Cluster NDB който поддържа изцяло това хранилище. Операциите които се извършват се дублират като може да се настрои дори още по-голяма степен на надеждност. Последната версия на базата данни поддържаща това хранилище е **MySQL Cluster 7.1.3**

# Видове хранилища

**Federated** — Осигурява възможността да се свърваж няколко MySQL сървъра за да създадат една логическа таблица.

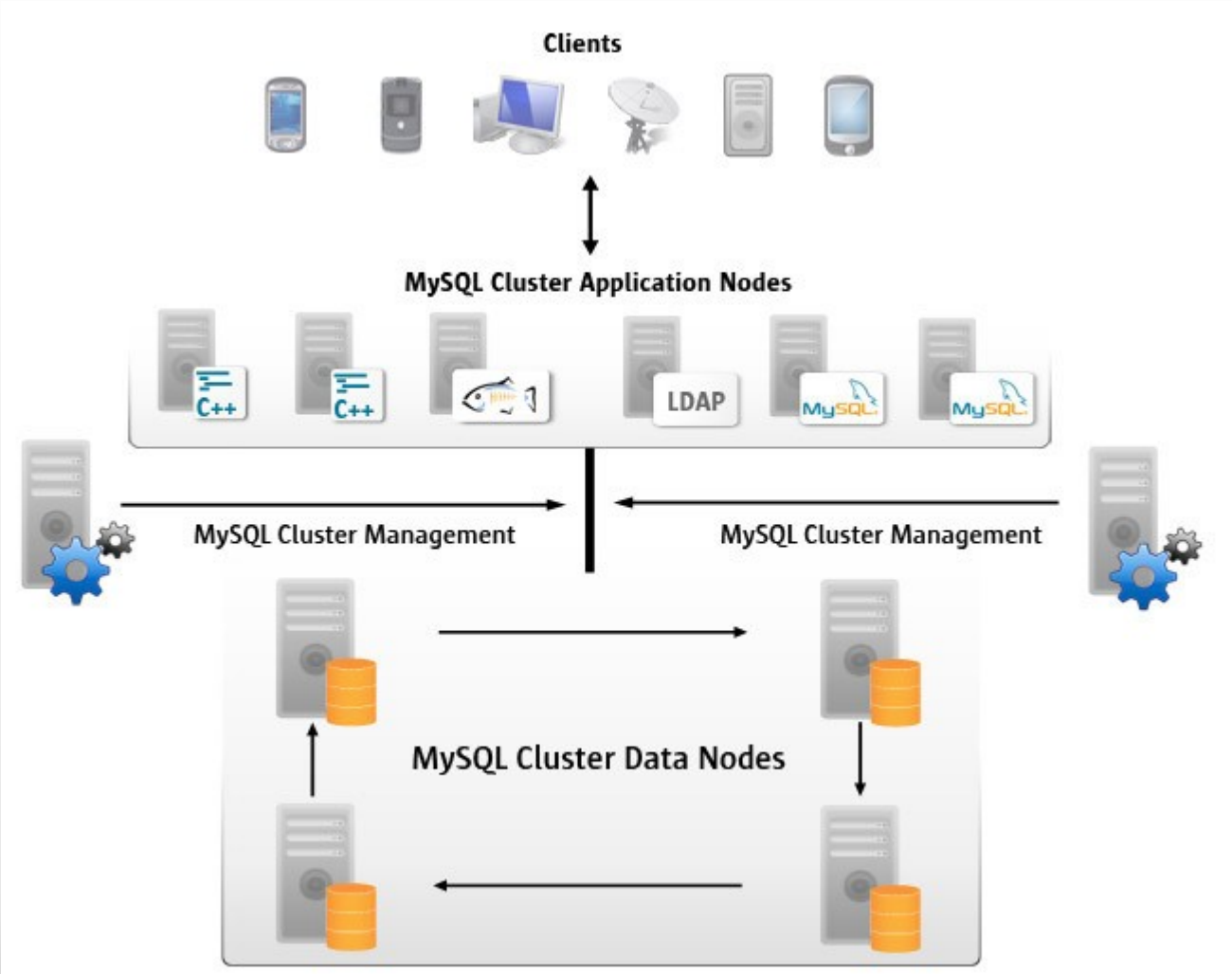
**CSV** — Това хранилище използва за съхранение на данните текстови файлове във формат .csv(comma-separated values). Хранилището улеснява размяната на данни между различни платформи, софтуер и приложения.

**Blackhole** — Това хранилище приема, но не запазва никакви данни. Винаги връща празен резултат. Представя се само с **.frm** файл, без файл за данни.

**Example** — Това хранилище не прави нищо. Фунцкията му е да служи като пример за създаване на нови хранилища за MySQL от разработчиците.

# Групиране чрез MySQL Clustering

**MySQL Cluster** е технология, която осигурява групиране на данни без споделени части. За първи път е включена в MySQL 4.1 версия през ноември 2004 година. Тя е предназначена да осигури висока надеждност и висока производителност, като същевременно позволява почти линейна скалируемост. MySQL клъстер се осъществява чрез допълнително хранилище на разположение в рамките на MySQL наричао NDB или NDBCLUSTER ("NDB" означава база данни в мрежата).



# Справяне с аварии

MySQL предлага множеството стратегии за справяне с аварии:

- **MySQL NDB Clustered Storage Engine** – използване на хранилището

<http://dev.mysql.com/doc/refman/5.1/en/mysql-cluster-overview.html>

- **Continuent** – комерсиално приложение

<http://www.continuent.com/>

- **Sequoia**(по известно като C-JDBC от ObjectWeb)

<https://forge.continuent.org/projects/sequoia>

- **DRBD for MySQL High Availability**

<http://www.mysql.com/products/enterprise/drbd.html>

- **Linux Heartbeat / High Availability**

<http://www.linux-ha.org>

- **MySQL Pool-Poller**

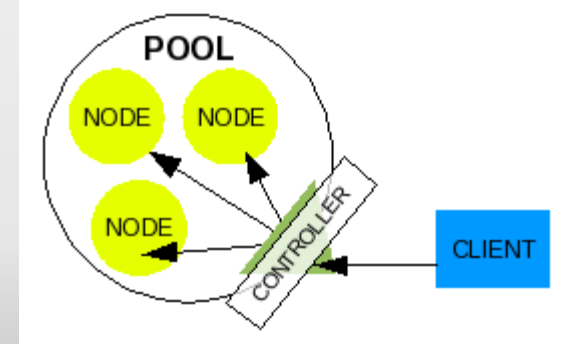
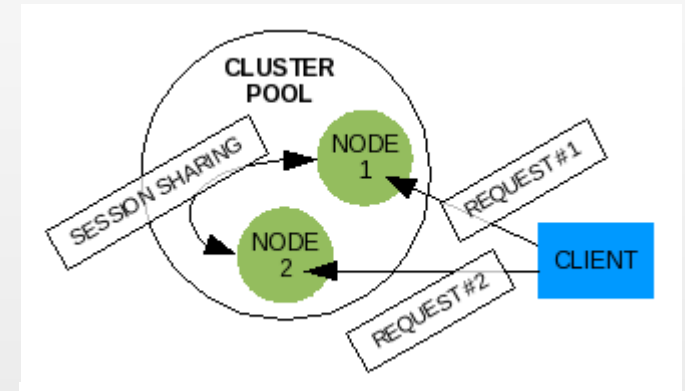
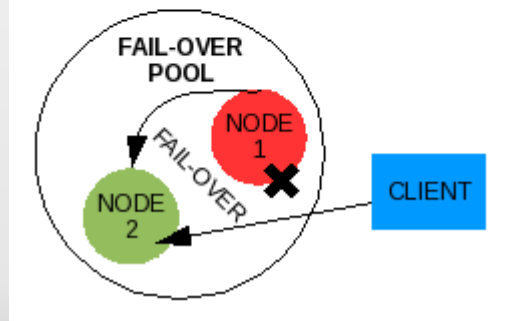
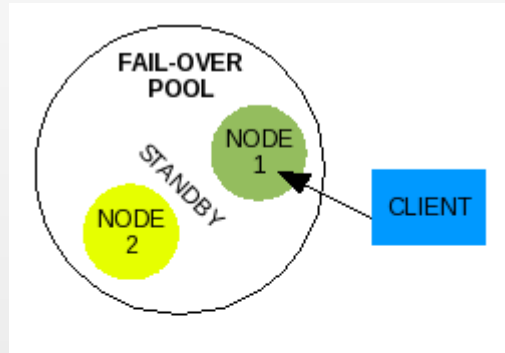
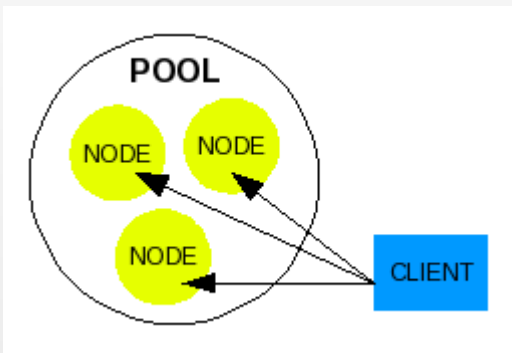
<http://www.codepin.org/project/mpp>

- **MySQL Proxy**

[http://forge.mysql.com/wiki/MySQL\\_Proxy](http://forge.mysql.com/wiki/MySQL_Proxy)

# Справяне с аварии

Изброените методи предлагат разнообразни решения за справяне с проблеми, ползвайки управление на състоянията(State Management).





# Възтановяване от аварии

Различните хранилища предлагат различни възможности за възтановяване на данните при системна авария.

**MyISAM** – таблиците може да се поправят чрез инструментите *mysqlcheck* или *myisamchk*.

**InnoDB** – това хранилище е най-походящо и предлага най-много възможности за възвръщане на данните след авария. За да може да се ползват тези възможности трябва MySQL сървъра да е стартиран със опцията *-log-bin*, да се правят периодични пълни резервни копия чрез *mysqldump* командата, също така и периодични допълнителни резервни копия чрез *mysqladmin*.

# Възтановяване от аварии(InnoDB)

Нека разгледаме някои от възможните случаи за аварии:

1. Авария в операционната система
2. Авария в захранването
3. Проблем с файловата система
4. Хардуерен проблем

При случаите **1)** и **2)** след рестартиране на сървъра, поради липсата на отделни части, данните не са консистентни. InnoDB чете логовете, намира чакащите, завършените или не транзакции и автоматично отменя тези които не са извършени, отразява завършените.

При случаите **3)** и **4)** MySQL сървъра няма да се рестартира коректно. Трябва да се форматира диска и да се възтанови резервно копие от вече наличните.

??



**Въпроси?**