

# **Backup and recovery benchmarks for MyISAM and InnoDB engines with Zmanda Recovery Manager for MySQL.**

*By Dmitri Joukovski and Shailen Patel*

*"As MySQL gains widespread adoption and moves more broadly into the enterprise, ZRM for MySQL addresses the growing need among database administrators to protect their digital assets with a comprehensive backup and recovery solution. Zmanda is a valued member of MySQL's ecosystem, and their open source backup solutions are an excellent complement to MySQL's open source database offerings."*

***Marten Mickos,  
CEO of MySQL AB.***

---

## **Abstract**

This document provides backup and recovery benchmarks for MyISAM and InnoDB engines with logical, raw and snapshot methods of backup available in ZRM for MySQL.

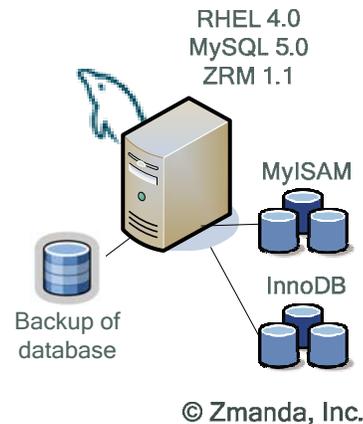
Please send your comments to [\*feedback@zmanda.com\*](mailto:feedback@zmanda.com)

The MySQL database has become the world's most popular open source database because of its consistent fast performance, high reliability and ease of use. The DBAs also expect performance, robustness and ease of use from a backup solution for MySQL. Often times we are asked what performance to expect from Zmanda Recovery Manager (ZRM) for MySQL. This document describes our first results in measuring ZRM backup and recovery performance.

## Testing configuration

2 x 2.8Ghz/800 FSB Xeon server  
4GB ECC/Registered 333Mhz RAM  
9500S-12MI 3Ware SATA Controller  
230 GB RAID 1 system array  
2.27 TB RAID 0 data array  
All Disks 7200 RPM

Operating System - Red Hat Enterprise Linux AS 4  
with 2.6.12 kernel  
MySQL version 5.0  
MyISAM and InnoDB storage engines  
Version 1.1 of Enterprise Edition ZRM for MySQL



According to 3Ware, the 9500 SATA controller supports a transfer rate of approximately 400 MB/s. To benchmark ZRM against file system performance, we measured that our test server was able to copy files with an average transfer rate of 37 MB/s.

To measure ZRM performance for backup and recovery in real life scenarios, we used Netflix databases available as part of the Netflix Prize project targeted on improving the accuracy of movie recommendations. For both InnoDB and MyISAM storage engines we used databases with sizes of 1000, 2000 and 4000 MB to evaluate if transfer rate changes with the size of the database. Each database had two tables, one with movie titles and another one with movie ratings. We calculated the transfer rates by dividing the size of the original database by the amount of total time taken for backup or recovery. All observed transfer rates are provided in a table on page 4.

## Discussion of results

We started our testing by instructing ZRM to do logical backups using *mysqldump*. Logical backups provide more flexibility for recovery because the backup file is a text file with all MySQL statements needed to re-create both the database schema and contents. Logical backup using *mysqldump* works for all storage engines except the NDB engine (used for MySQL clustering). The biggest advantage of logical backup is flexibility for a database recovery. You can restore logical backups of MySQL to other architecture and even to another database.

Transportability of logical ZRM backup images makes ZRM a convenient tool for migration. For example, you can move your MySQL data:

- From MySQL on Solaris to MySQL on Linux
- From one storage engine to another
- From a 32-bit server to a 64-bit server
- From one managed hosting provider to your datacenter or another provider with a different MySQL configuration.

Of course, there is price to pay for such flexible recovery. Restoring data from the logical backups takes a long time since you have to read and replay back all MySQL statements. For the MyISAM engine the observed transfer rate for backup was close to 5 MB/s, but the restore time was only 2 MB/s. For the InnoDB engine the transfer rate for backup was 8 MB/s and the restore rate was 2 MB/s.

ZRM for MySQL provides a wealth of information about the state of the MySQL server during backup. For example, ZRM reports to you how long did it take to flush memory buffers, or how long was the READLOCK. During logical backup, for both engines the READLOCK time was 95% of the total backup time. This is a crucial measure of the impact that backup has on your application. The shorter the READLOCK time the better for an application and users.

Another downside of logical backups is that it could be difficult to predict the size of the backup file. Depending on the type of data and your database schema, the size of the logical backup could be larger than the database itself. In our testing, for the MyISAM engine the backup file was 30% larger than the database file. One remedy is that since logical backup is basically a text file you can usually get a decent compression. Interestingly, the logical backup file (uncompressed) for InnoDB engine was 40% smaller than the database file.

Our next step was to measure raw backup and recovery performance by instructing the ZRM to use *mysqlhotcopy* available only for MyISAM and ARCHIVE engines. Raw backup provides a consistent copy of a database with your backup being a binary file. The advantages of raw backups over logical backups are:

- Backup and especially recovers are much faster. In our testing the backup transfer rate was 26 MB/s, or more than 5 times faster than the logical backup. The recovery transfer rate was 40 MB/s, or more than 20 times faster than logical recovery.
- You will always know the size of your backup since it is just a copy of a database.
- READLOCK time was only 60% of the total backup time.

Both logical (via *mysqldump*) and raw (via *mysqlhotcopy*) backups provide so-called "warm backup", meaning you don't have to shut down the MySQL server for backup, but all tables are locked during backup and users can't enter their data. If your MySQL server runs on Linux with Logical Volume Manager (LVM), you can perform another type of raw backup by taking advantage of LVM2 snapshot functionality managed by ZRM. Backup via snapshots works well for all storage engines, but it works best for transactional engines such as the InnoDB and the upcoming Falcon storage engine. For transactional engines it provides true "hot backup", meaning that during backup the READLOCK time is practically zero. The performance data for

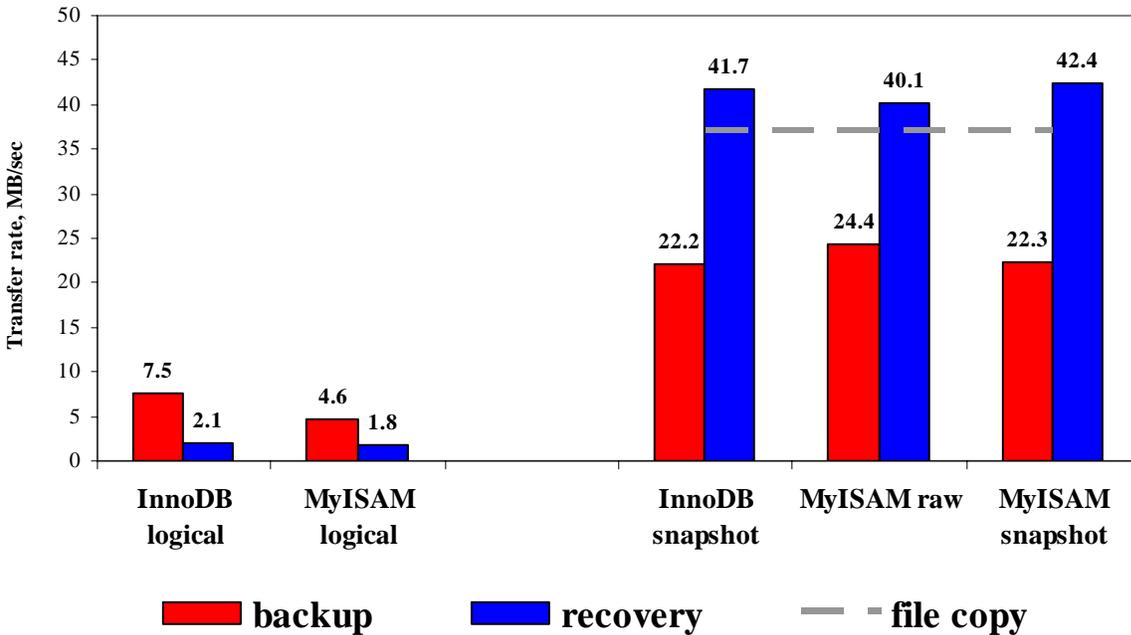
backup and recovery via snapshots are comparable to backup and recovery via *mysqlhotcopy*. Raw backups can be recovered ONLY to the same version of MySQL server on the same operating system as the original data. You should consider that when choosing raw vs. logical backup.

In our testing we did not find any significant difference in backup and recovery transfer rates for different sizes of the database.

### ZRM backup and recovery transfer rates for MyISAM and InnoDB engines

Engine	Size of DB (MB)	ZRM logical backup & recovery			ZRM raw backup & recovery					
		Backup Size (MB)	Backup (MB/s)	Restore (MB/s)	mysqlhotcopy			LVM snapshot		
					Backup Size (MB)	Backup (MB/s)	Restore (MB/s)	Backup Size (MB)	Backup (MB/s)	Restore (MB/s)
MyISAM	1000	1320	4.9	1.8	1001	27.8	38.5	1001	26.3	38.5
	2000	2570	4.7	1.9	1940	26.3	41.7	1940	24.0	39.6
	4000	5230	4.6	1.8	3940	24.4	40.0	3940	22.3	42.4
		READLOCK time 95% of backup time			READLOCK time 60% of backup time			READLOCK time is 0		
InnoDB	1000	641	8.3	2.0	mysqlhotcopy is not available for InnoDB			1170	22.2	31.3
	2000	1160	8.2	2.2				1990	25.3	37.7
	4000	2500	7.5	2.0				4006	22.2	41.7
		READLOCK time 95% of backup time						READLOCK time is 0		

The graph on page 5 provides the visual overview of all observed transfer rates for a 4000 GB database. The grey dash line shows the ‘internal benchmark’ of 37 MB/s, the transfer rate for copying the database file with a *cp* command.



How to use the provided results? If you have hardware similar to what we used in our testing, you can estimate backup and recovery time for your database. Let's say you have 10 GB in MyISAM tables. Chances are that it will take you approximately 40 minutes to do full logical backup (10,000 MB / 4.7 MB/s = 2128 seconds = 35.5 minutes) . It will take more than an hour and half to restore your database. However, it will take only 7 minutes for a backup of such a database using ZRM with snapshots, and your recovery time will be only 4 minutes. Please remember though, that in real life most of the time you perform incremental backups of your database when you backup ONLY data changed after the previous backup, usually less than 5% of total database size. ZRM for MySQL supports incremental backups for all backup methods and for all storage engines.

### Future plans

The provided results are our first step in measuring the performance of ZRM for MySQL when there is no user activity. We do understand that DBAs are concerned with backup performance when MySQL is used. That is why we will continue to measure ZRM performance including conditions with heavy user activity. We are very interested to hear about your experience with ZRM for MySQL. Please drop us a line [feedback@zmanda.com](mailto:feedback@zmanda.com) or post a message on our forums <http://forums.zmanda.com>

Additional information about ZRM for MySQL is available at <http://www.zmanda.com/backup-mysql.html>. The User documentation is available on wiki at <http://mysqlbackup.zmanda.com/>. Community edition of ZRM can be downloaded from <http://www.zmanda.com/downloads.html>