www.spkaa.com Ph: 888-310-4540





Confluence

Upgrading confluence would have been really simple if we wanted to retain the same server. It's a simple process:

- 1. Stop the service
- 2. Backup MySQL database
- 3. Backup confluence application directory
- 4. Backup data directory
- 5. Upgrade any applications that are available to be upgraded
- 6. Install the tar.gz from Confluence
- 7. Start the process
- 8. Wait around 10 minutes for indexes to be rebuilt.

However, we went a slightly different path. This was for 3 reasons:

- 1. Upgrade the OS to Ubuntu 12.04
- 2. 3 instances needed upgrading on the same host
- 3. Keep production running while we test the new release

Here are the steps required during our installation

- 1. Stop the services for each instance
- 2. Backup database from original host
- 3. Backup application directory into a tar.gz
- 4. Backup data directory into a tar.gz
- 5. Start the services for each instance on the production node
- 6. Copy all backups over to the new host
- 7. Restore the databases
 - a. One problem we saw when upgrading is that there's some large BLOBs in one of the WIKI instances requiring special restore commands on the MySQL command line and in the [mysqld] section of my.cnf
 - [mysqld]
 - transaction-isolation=READ-COMMITTED
 - max_allowed_packet=128M

During DB restore, you also need to set the max_allowed_packet on the client side so that both client and server can handle the BLOB:

 $Mysql ---max_allowed_packet=128M - uroot - p \ confluence < confdb.sql$

- 8. Copy all the data directories over to the new application host
- 9. Upgrade any applications that are available to be upgraded
- 10. Install the tar.gz from Confluence
- 11. Upgrade the database connection strings as needed in <datadir>/confluence.cfg.xml <property name="hibernate.connection.password">PASSWORD</property>



SPK and Associates 900 E Hamilton Ave, Ste.100 Campbell, CA 95008

<property name="hibernate.connection.url">jdbc:mysql://localhost/confluence?autoReconnect= true</property>

<property name="hibernate.connection.username">USERID</property></property>

- 12. Start the process
- 13. Wait around 10 minutes for indexes to be rebuilt.
- 14. We also wanted to see this from the production server. A handy way to do this is by using a ProxyPass setup like this:

ProxyPass /newwiki ProxyPassReverse /newwiki http://<NewHost>:8000 http://<NewHost>:8000

This allows us to go to the original host, server1 and see the new application as a subdirectory at this URL:

http://server1.abc.com/newwiki/