LISTSERV Maestro Admin Tech Doc 15

Test-Bed Backups

This document is a LISTSERV Maestro Admin Tech Doc. Each admin tech doc documents a certain facet of the LISTERV Maestro administration on a technical level. This document is number 15 of the collection of admin tech docs and explains the topic "Test-Bed Backups".

Last updated for LISTSERV Maestro 9.0-2 on January 24, 2020. The information in this document also applies to later LISTSERV Maestro versions, unless a newer version of the document supersedes it.

All of L-Soft's manuals for LISTSERV are available in ASCII-text format via LISTSERV and in popular word-processing formats via ftp.lsoft.com. They are also available on the World Wide Web at the following URL:

URL: http://www.lsoft.com/manuals.html

L-Soft invites comment on its manuals. Please feel free to send your comments by e-mail to:
MANUALS@LSOFT.COM

Table of Contents

1 Introduction

In certain situations it may seem useful to be able to make a copy of all data in a given LISTSERV Maestro installation and transfer it into a second installation. Usually that is for the purpose of creating a test server on which the effects of a certain version upgrade are to be tested before the same upgrade is applied on the production server, and for this the test server is supposed to be initialized with a copy of the production server's data.

The problem with this requirement is: How to get a copy of the original data into the test system?

The first idea that comes to mind would be, to simply trigger a backup on the original system, and then restore this backup into the test system.

However, this approach has several pitfalls which could, in a worst case scenario, even destroy some of the data in the original production system.

Some of the possible pitfalls are for example:

- If at the moment the backup is triggered the original system contains a mail job which is already in the outbox and scheduled for delivery, and then this backup is restored into the second system, and then the scheduled delivery time arrives, the second system would actually send out the job to the given recipients!

  And since the job is a copy of a real job on the production server, with real recipients, this mailing from the test system would go out to these real recipients, which is usually not desired (and worse, since the same job also still exists on the production server, where it will be mailed out at the same time, the recipients will actually each get two copies of the same mail, i.e. the test server sends out duplicates of the mails which are already being mailed by the production server).

- If the production server installation actually consists of an installation where the LISTSERV Maestro components are distributed over several servers, for example where LUI is installed on one server and TRK is installed on another server, then this means that the information about how the components are distributed will be contained in the backup data too.

  And you would then have to be very careful when restoring this backup into the test server, so that you do not accidentally end up for example with connecting the test server LUI with the TRK of the original production system (or vice versa).

- If the production server installation uses an external system databases, then the connection information for this external database is of course also stored in the data which is part of the backup. And during the backup restoration, the information will therefore also be restored into the test server. Which means that the test server will try to connect to the same external database (and use it as its system database) as the production server does. At worst, the test server could then delete or change data which actually belongs to the production system!

Because of these pitfalls, it is generally not advisable to restore a backup of an existing and running system into a different system. Therefore the idea of restoring a backup from the original server into the test server is not really a good idea either.

To solve this problem, LISTSERV Maestro offers the feature of a **test-bed backup**:

A test-bed backup is similar to a normal backup of a system in so far that it contains all the system data (all user accounts, jobs, reports, tracking information, subscriber data etc.).

However, all the "dangerous" aspects of the data, like information about distributed components, connection information for external databases, scheduled jobs in the outbox, etc. have been changed by the system (when the data was written to the test-bed backup) so that they do no longer pose a risk when the test-bed backup is restored into a test server.

Or in other words: A test-bed backup is as close a copy to the data of the original system as possible, but with all risky information removed or changed.

Information that is removed or changed includes (among others):

- **Component distribution information**. If the original system was distributed over several servers, then the test-bed backup is not aware of this any longer and assumes that all components are on the same server (the "`localhost`").

- **External database information**. If the original system did use an external system databases, then the test-bed backup is not aware of this any longer and assumes that the internal system database is used. If you plan to change this and to actually use an external database similar to your production system, see section 4.

- **Instance-ID information**. The test-backup backup does not contain the original system's instance ID, so that during the first start the test server will generate its own instance IDs.

- **LISTSERV connection host information**. The test-bed backup does not contain information about which LISTSERV host to use.

- **Tracking host information**. The test-bed backup does not contain information about which tracking host name to use, so this information needs to be added manually (via the Administration Hub) after the test-bed backup has been restored to a test server to enable tracking.

- **Scheduled jobs**. The outbox send queue in the test-bed backup is set to "Sending is disabled", so that any queued jobs in the outbox are not automatically delivered by the server into which the test-bed backup is restored. It is therefore necessary to re-open the outbox (via the Administration Hub) on the test server, if you need to actually send any jobs on the test server. However, before re-opening the outbox, make sure to check it for any production jobs still left in it, and delete or revoke those jobs, to stop them from being delivered when the outbox is re-opened.

- And more.

With the risky information removed or reset to appropriate defaults, it is safe to restore the test-bed backup into a test server, without the risk of damaging or impacting the production server from which the backup was initially taken.

The following sections describe how such a test-bed backup can be created on an original system and how it must be restored into a test system.

IMPORTANT: The test-bed backup is **not** meant to be used to clone the production server onto a second server that is then used as a second production server, or that is then used as a replacement of the original production server. A restored test-bed system cannot assume this role, because among the data that has been deleted in the test-bed backup are also the LISTSERV connection settings. So the test-bed system does not have a connection to LISTSERV, and on such a test-bed system, this connection data can also not be entered manually, i.e. a test-bed system remains permanently without a LISTSERV connection. That in turn means that on a test-bed system, all the features that require a LISTSERV connection cannot be used (most notably mail job deliveries, test mailings, bounce reporting and advanced subscriber lists). Which is the reason why such a system is not fit to be used as a production system.

2 Creating a Test-Bed Backup on the Original System

To create a test-bed backup on the original system, log into the Administration Hub of this original installation, then select "Global Settings" → "Administration Hub → "General Administration" and click on "Create Test-Bed Backup".

Once triggered, the test-bed backup proceeds just like a normal backup. The resulting backup ZIP files are stored into the same backup folders as the normal backup files are, but are marked as "`TestBedBackup`" in the filename.

Just like with a normal backup, a test-bed backup consists of the backups of all three components, i.e. a complete test-bed backup encompasses all three backup ZIP files.

For each test-bed backup that is created, a backup report is written to the log folder of the Administration Hub component ("`[install_folder]/hub/logs`"), just like for a normal backup. The content of this report is also very similar to a normal backup report, only that it clearly states that this backup is a test-bed backup (and thus contains data which has been changed slightly, in comparison to the original data, see above).

Similarly to a normal backup, a test-bed backup is only valid (i.e. contains complete and restorable data) if the last line of the report reads:

```
The test-bed-backup was completed successfully
```

(If you do not see this line, then the backup was not successful in some way and should not be used during the restore step described in the next section.)

3 Restoring a Test-Bed Backup into the Test System, using the Internal Database

A test-bed backup is restored exactly like a normal backup, so please see the section about restoring a backup for details.

Just like with a normal backup, it is important to restore **all three** components and also to make sure that the backup data of the three components which are restored actually was written by the same backup (or test-bed backup in this case). As normally, this verification is done by verifying the backup ID (Test-Bed-Backup-ID) in the filenames of the three backup ZIP files.

Of course the test-bed backup is restored not on the original system where it was created, but on a different system, the "test server" system. It is also important to know that a test-bed backup by default is restored into a test server which was installed on a single server, including the internal database, i.e. which meets the following requirements:

- All three components (HUB, LUI and TRK) are installed on the same server.

- The internal MySQL database is also installed, to be used as the system database. (This is necessary even if you plan to not use the internal MySQL database, see next section.)

These requirements are most easily met if you install a fresh installation of LISTSERV Maestro on the test server and simply select all three components plus the internal system database when running the setup wizard.

If you have such a test server system and have also identified the test-bed backup you want to restore, then restore it by simply following the normal backup restoration description.

4 Restoring a Test-Bed Backup into the Test System, using an External Database

There are some scenarios where you do not want to restore the test-bed backup into the default MySQL internal database:

- Your production system is running with a non-MySQL database configured by your own staff and you want to perform a test-bed backup restoration that uses the same database vendor as your production system, for example to make sure that a potential production backup restoration in the future has also been tested using the same database vendor.

- Your production system is running successfully with data that causes problems in MySQL due to limitations that exist with MySQL, most notable record size limitations and limitations with regard to how many indexes can be created on a database table. In such a case, trying to restore a test-bed backup using the default internal MySQL database fails, so you must take action.

To use an external database to restore a test-bed backup, the following steps are necessary (described for Windows only, the steps for a non-Windows system are different):

1. Restore the test-bed backup data files as described above, but do not yet start the services.

2. If it is running, stop the "L-Soft MySQL" service and only start the "L-Soft Tomcat" service. By starting up like this, you force LUI to run into an error when attempting to restore the test-bed backup data into the internal database. This is the expected behavior, you will notice appropriate errors being logged in the LUI log file, but those can safely be ignored.

3. Making use of the fact that the HUB component is running and accepts browser connections, log in with the "admin" account to the HUB. **Note**: You must use the password that was present in your production system when the test-bed backup was written, this password is contained in the HUB backup data that was restored successfully in the step above. In the settings for the LUI component, you will notice that neither a database plugin is registered nor an external system database connection for LUI is configured. This is a special property of a test-bed backup. You now register the same database plugin that is in use in your production system. Special care is necessary when supplying the external system database connection: Your staff should at this point have prepared a special database (or schema, as would be appropriate for Oracle) that is ready to accept connections. Supply the connection settings for this new database, taking extra care **NOT** to supply the settings for your production system. In many cases the settings are deceivingly similar, so **please be careful here**!

4. By default, a test-bed backup system does not need external database vendor driver JDBC jar files, but in your case, you now must copy the appropriate files to the LIB folder of your test system, just as would be the case for your production system.

5. Now the HUB settings and the contents of the LIB folder are ready for an external database connection. You now must once again prepare the LUI component to restore from the LUI part of the test-bed backup. **Do NOT restore HUB or TRK again, ONLY prepare this for the LUI components, this is important**!

6. With all preparations in place, you are now ready to finally restart the "L-Soft Tomcat" service. This will start HUB and TRK normally, but will cause LUI to once again read the data from its test-bed backup ZIP file, only this time using the external system database connection you prepared in the HUB. When you monitor the LUI log, you should see that your prepared external database is used as target for the backup data restoration.