



# CEM Upgrade – avoiding pitfalls

.v04 August 2012

When upgrading a CEM Environment, very often the TSO is confronted to many little issues that can fast sum up to a database corruption and long downtime as the DB needs to be restored.

It is advised to check this document, and think about the proper upgrade path to take.

Note – these apply to CEM 4.x.x to CEM 4.5.6.x as CEM 4.5.x to APM9 upgrades.

## CEM to CEM Upgrade

The following steps can be taken to minimize any unwanted surprise !

- Make a backup of all important files. When talking about important files, the tess-customer.properties file as the uploaded SSL keys on the TIM's are also an issue. To address this part – a Field-pack cem\_backup-1.17.2-2.noarch.rpm has been written. Just install it on TIM and TESS, and call it using cem\_backup.sh once. All important files will now be in the user root's Home-directory under ~/backup  
Note that the Database backup is only a link to the latest nightly backups (as it can become very big – this is the best way to point to it and not waste disk-space).
- Copy these backup files on another physical System !
- Make absolutely sure that the customer didn't add any other user as by example a “guest” user to the DB. The DB Upgrade process will fail miserably if it exists !
- Next is to check the README file provided by the new release of CEM that is supposed to replace the old release. Check the ReadMe.htm file provided in the document pack, and understand what it means.  
By example For the migration from CEM 4.2.1 to 4.5.6.1 – the database changes are so important that the update can take several days. Hence – this is something to take into account.
- Check the actual Database size, and (as stated in the ReadMe.htm – the sizes of the ts\_defects and ts\_defects\_meta\_values fields. On a TESS, issuing the following command would provide you a good overview on the DB content.

```
[root@tess ~]# PGUSER=admin PGPASSWORD="quality" psql -t -d cemdb -c 'SELECT relname, reltuples, relpages * 8 / 1024 AS "MB"
FROM pg_class ORDER BY relpages DESC;' | egrep "ts_defects |ts_defect_meta_values "
```

|                       |             |      |
|-----------------------|-------------|------|
| ts_defect_meta_values | 1.90366e+07 | 2693 |
| ts_defects            | 386284      | 87   |

With that data – you can decide if you need to start early morning, or late at night.

- Best is to actually install the “screen” program on the TESS, and start lynx from within screen. Once done, and the database conversion is running, you can detach the screen process (<Ctrl>-d)from the console, log out, go home, and come back in the next morning to reattach the screen process (screen -r) to see if it's finished.

## CEM to APM or Introscope to APM Upgrade

There are some other issues that can come up when going the APM Way.

- Check under which user the previous installation has been executed. If it has been done as user root – you have to do the upgrade process also as user root. If not – eventually it will work with the previously used user. Up to now though – performing the upgrade using the root user is the one with the fewest head-ache to expect.  
In our experience, running the install as a user called “wily” laid down the EM ok (of course) and even installed postgres successfully (running as wily). However, it failed on installing the ODBC drivers that are used by the python upgrade scripts. (You would never know that by looking.) The error we saw was “odbcinst: SQLWritePrivateProfileString failed with General error request failed.” Therefore, either run the DB install as root or create as a postgres user. Keep in mind that you won’t get init scripts for postgres installed to /etc/init.d/postgresql-8.4 if you don’t do it as root, either, which could be important on reboot.
- If you choose to run the entire install as root (we tried via sudo permissions) then the whole install tree will be owned as root, so be prepared to chown the tree (but not postgres in that tree!) If you don’t have root access yourself and don’t chown the tree, you won’t be able to update configs, write logs, etc. It can be very time consuming to work out, which is why I think you should consider postgres on its own and Introscope on its own, saving yourself the permissions mess.
- If you have a cluster and want to install the DB on a collector machine, the installer will not permit you to do that at the same time as you install/upgrade the collector itself. For some reason, the DB-install screen only comes up with you choose MOM, not collector. Only way to get past this is to either install the DB by itself in a first pass and then run the installer again for the collector install/upgrade. Or you could tell it that it’s a MOM and put in garbage entries in the “where’s your collector” screen, then go into the properties file and remove that stuff and reset it to a collector afterwards. That’ll allow you to get to the option to actually install postgres.
- In case you want to migrate a TESS DB to an APM9 system running under Windows – you can not perform the DB Upgrade under Windows. Means – you have to upgrade the DB first under Linux, then copy the DB over to windows and use it there. The DB of the TESS needs to be at least 4.5.4 – so in case you have an older release – you have to upgrade the TESS to the latest release first.
- If you use the DbBackup script to create a backup copy of the PostgreSQL database on Windows or when you perform a Configuration Import, you may see the following error message displayed when you run the DbRestore script to restore the database:

```
ERROR: language "plpgsql" already exists
```

This error message is displayed when the DbRestore program attempts to install the procedural language plpgsql on a computer where it already exists. The error message is informational and does not prevent the database from being restored. If you see this error message when restoring the database from a backup, you can safely ignore it.

- If going from pre-9.0.6 to post-9.0.6 release, you must install a fresh version of Postgres and then start moving data from the old to the new depending on what data you want to preserve.  
If upgrading a DB running on CEM 4.5.6 to APM 9.1.1 – you will have to perform the following steps:
  1. Backup the database (schema Version 4.5.6) to a file cemdb-sv456-psql835.dbbakup
  2. Stop the TESS psql835
  3. Install the DB Only of APM 9.0.6 or later psql-84-32bit

4. restore the cemdb-sv456-psql835.dbbbackup to the psql-84-32bit
  5. An architecture change will take place – so you need to again take a backup of the DB. Do this as cemdb-sv906-psql8432bit.dbbbackup
  6. copy the file over to the new DB Server which will be running on a 64Bit architecture.
  7. install the DB of APM 9.1.1 – 64it DB of psql-8.4-64bit. It will install an empty cemdb.
  8. Import the cemdb-sv906-psql84-32bit.dbbbackup into the DB psql-84-64bit
  9. Restart the APM Installer, and upgrade not the DB to the 9.1.1 Version.
- You are done.

## CEM to APM Upgrade example – where the DB runs under Windows

Your Wily CEM installation must be version 4.5 to 4.5.4 to upgrade to CA Wily APM 9.0. If your installation's version is prior to version 4.5, you must first upgrade to version 4.5 before proceeding. For example, if you need to upgrade from 4.0.x, first upgrade to 4.2.x and then to 4.5.x, using the documentation provided in those releases.

**IMPORTANT!** If your Wily CEM version is 4.5.5 or later, you must upgrade instead to the first service pack release after CA Wily APM 9.0.

Note – the exercise is actually to migrate the DB Content from Postgresql 8.3.x to 8.4.x. The issues are as follows:

- a. CEM 4.5.x has Postgresql 8.3 installed
- b. APM 9.0.5.x has Postgresql 8.4 as requirement and provides it
- c. APM 9.0.x with x < 5 has only upgrade scripts for CEM 4.5.4.x
- d. APM 9.0.5 or higher has upgrade scripts for CEM 4.5.6.0 (NOT 4.5.6.1 !!! - a workaround exists though)

However – the exercise consists on these 2 logical steps:

1. Copy the DB content from Postgresql 8.3 over to Postgresql 8.4
2. Convert the database structure from CEM 4.5.x to APM9.0.x format

Under Windows – we can only import an already converted Database. So we need to do all the work on the TESS, or another Linux system !

On the TESS:

1. Backup the 4.5.2 Postgres DB 8.3 - cem45old.backup
2. Upgrade the 4.5.2 TESS to 4.5.4/6 using the 3 upgrade TESS images (updateonce, third-party, tесс)
3. Backup the 4.5.4/6 Postgres DB 8.3 - cem45x.backup
4. Stop the TESS 4.5.4/6 Postgres DB 8.3 using: "service postgresql stop"
5. Install APM9 on the Linux appliance under /opt - make sure you install a plain Postgres DB 8.4 (Database Only installation is enough). Don't start the Postgres DB 8.4 yet. We only need the dbbackup/dbrestore scripts at this time ! Stop it if required !
6. start the TESS 4.5.4/6 Postgres DB 8.3 using: "service postgresql-8.3 start"  
(Service name got renamed when installing APM9).
7. Use dbbackup to backup the actual Database from the 4.5.6 CEM TESS:  
/opt/Introscope9.0.x.x/install/database-scripts/unix/dbbackup-postgres.sh (adapt the options)  
name it: cem45x.backup
8. Stop the old DB Postgres 8.3 database. In the /etc/init.d directory - you'll find several scripts for Postgresql. The old release is: postgresql-8.3 - execute it with the stop argument: "service postgresql-8.3 stop"
9. Start the new Postgres DB 8.4 with "service postgresql start"
10. Import the backup-database cem45x.backup into the new Postgres DB 8.4 with:  
/opt/Introscope9.0.x.x/install/database-scripts/unix/dbrestore-postgres.sh (adapt the parameters again). The schema etc. will be adapted to APM 9.0.x.x structure - so no more manipulation to be done on that side.
10. Use dbbackup to backup the actual Database from the APM 9.0.x.x Postgres DB 8.4:  
/opt/Introscope9.0.x.x/install/database-scripts/unix/dbbackup-postgres.sh (adapt the options)  
Create a apm90x.backup

Done with the TESS and Database conversion for use with APM9.

#### On the APM Postgres 9.0.5.0 Windows 2003:

1. Copy the freshly create file apm90x.backup over to the windows machine.
2. Use the batch files in:  
C:\Program Files (x86)\CA Wily\Introscope9.0.x.x\install\database-scripts\windows\dbrestore-postgres.bat using the apm90x.backup under windows to import/restore the converted DB data.
3. Now you have a 8.4 Postgres DB on Windows 2003 in a 9.0.x.x format with the data from the 4.5.x CEM

## Changes on CEM 4.5.6.x/9.0.5.x to APM 9.0.7.2 upgrade

Use the missing parts from CEM to APM migration – as these will be required.

- 1) Install APMDB 9.0.7.2 (Can be a new machine or the same if you can separate the directories)
- 2) Upgrade schema from 4.5.6.0/9.0.5.x to 9.0.7.2 (upgradeschema.sh used is from 9.0.7.2 installation)

```
./upgradeschema.sh -databaseName cemdb -databaseType Postgres -desiredVersion 9.0.7.2 -host 10.15.81.33  
-is64bit 64 -password quality -port 5432 -scriptsDir /opt/wily/Introscope9.0.7.2/install/database-  
scripts/ -user admin
```

- 3) Backup the upgraded database

```
./dbbackup-postgres.sh cemdb /etc/wily/cem/data/dbbackups/ quality 1
```

- 4) Restore the db into the new APMDB v9

```
./dbrestore-postgres.sh twilya11 /opt/wily/database/ postgres Quality00! cemdb admin Quality00! 5432  
/opt/wily/appoggio/cemdb_20111129\@130859.dbbackup
```

## Credits & Acknowledgements

Thanks to Hallett German for reviewing the document and making suggestions.

Thanks to Chris Kline for the head-ache he had during APM upgrades and sharing the notes.

Thanks to Polly Spangler for testing the upgrade path.

Joerg Mertin <[joerg.mertin@ca.com](mailto:joerg.mertin@ca.com)> – EMEA APM CoE SWAT