

# Table of Contents

<b>Introduction.....</b>	<b>1</b>
<b>Considerations.....</b>	<b>2</b>
<b>Installation and Configuration.....</b>	<b>3</b>
Deploying RPMs via YUM.....	3
Configuration for EMI/gLite Sites.....	3
Handling of CE Software Tags.....	4
Configuration for OSG Sites.....	4
Handling of CE Software Tags.....	5
Job runtime environment.....	5
Configuration for/ARC Site.....	5
<b>Basic testing.....</b>	<b>6</b>
<b>SAM Test.....</b>	<b>7</b>
<b>Deployment.....</b>	<b>8</b>
<b>Warning: Added on May 12, 2015.....</b>	<b>9</b>

# Introduction

CernVMFS is a network file system based on HTTP and optimized to deliver experiment software in a fast, scalable, and reliable way. Files and file metadata are aggressively cached and downloaded on demand. Thereby the CernVM-FS decouples the life cycle management of the application software releases from the operating system. More information can be found on the CVMFS Project page [↗](#) and in Technical Report [↗](#) from the CVMFS developers.

This page gives some CMS specific informations.

# Considerations

CVMFS needs some disk space on the local machine, where the jobs are executed. It is highly recommended to have this cache on a separate disk partition. The minimum of this local cache is 10GB, while 20GB or more are recommended for cache for CMS only. A cache shared among several VOs should provide at least 40GB space.

On each Grid site CVMFS profits very much from a Squid cache server. CMS uses Squids already with Frontier to deliver database constants. These Frontier Squids can be also used for CVMFS. Large sites should consider to deploy additional Squid capacity when they use them for Frontier and CVMFS. The SquidForCMS page gives more recommendations. CVMFS is expected to add only little additional load on the Squids.

# Installation and Configuration

The actual installation of CVMFS is independent of the Middleware. Since mid of October 2012 CVMFS is supported for Scientific Linux 5 and 6. CVMFS can be built from source for different operation systems or might be available from the CVMFS pages [↗](#).

## Deploying RPMs via YUM

You need to root access to the machine to apply this recipe.

Add CVMFS repository:

```
wget -O /etc/yum.repos.d/cernvm.repo http://cvmrepo.web.cern.ch/cvmrepo/yum/cernvm.repo
```

And import the GPG key of the repository:

```
wget -O /etc/pki/rpm-gpg/RPM-GPG-KEY-CernVM http://cvmrepo.web.cern.ch/cvmrepo/yum/RPM-GPG-KEY-Ce
```

Install the required RPMs. Make sure that your OS repositories are enabled, since it is quite likely that additional system RPMs, e.g. `fuse` and `autofs`, need to be installed in order to resolve the dependencies.

```
yum install cvmfs cvmfs-init-scripts
```

## Configuration for EMI/gLite Sites

There are some small differences how to set up CVMFS on different middleware stacks. The following is for EMI/gLite sites

Read the comment lines in `/etc/cvmfs/default.conf` in which order configuration are read in.

You need to configure the CVMFS repositories, that you want to support at the site.

For CMS add the following to `/etc/cvmfs/default.local` (you might need to create the file). Of course you can add other repositories. The CVMFS web page gives more examples [↗](#).

```
# Repositories: cms.cern.ch is vital for CMS, grid.cern.ch provides a Grid-UI and is a recommended
CVMFS_REPOSITORIES=cms.cern.ch,grid.cern.ch
# Make sure $CVMFS_CACHE_BASE has enough space
# Ensure that the file system hosting the cache has an additional 15% free space.
CVMFS_QUOTA_LIMIT=8000
# Increase the size to 10000 or bit more if you can effort it
```

The default cache directory is `/var/cache/cvmfs2`. It is recommended to have a separate partition for it.

Please note that, starting with CVMFS v. 2.1, the CVMFS cache is by default shared among all the enabled CVMFS repositories.

Some settings that apply to the CMS repository only are set in `/etc/cvmfs/config.d/cms.cern.ch.local`.

**Check this carefully. It is vital for jobs to find the local site configuration:**

```
# Important setting for CMS, jobs will not work properly without!
export CMS_LOCAL_SITE=<>
# This only needed if you did not configure Squids in /etc/default.[conf|local]
CVMFS_HTTP_PROXY="http://<Squid1-url>:<port>|http://<Squid2-url>:<port>[|...]"
```

For the setting of the `CMS_LOCAL_SITE` variable you have effectively three options:

1. Use SITECONF files needed by CMS jobs on the worker nodes from CVMFS. In that case, set `CMS_LOCAL_SITE` to your site name, i.e. set `export CMS_LOCAL_SITE=T2_XX_Example` (It may take a while, hours, for changes in your SITECONF area in the GitLab repository to propagate to CVMFS.)
2. (Recommended) Have a configuration management system such as cfengine or puppet keep an up-to-date copy of SITECONF on the local worker node disks, for example in `/etc`. (`export CMS_LOCAL_SITE=/etc/cms/SITECONF/T2_XX_Example`)
3. Keep a SITECONF copy on a shared filesystem and maintain this centrally. This has the downside of making your site depend on NFS or similar technology. (`export CMS_LOCAL_SITE=/nfs/cms/SITECONF/T2_XX_Example`)

**If you have a `cmsset_local[c]sh` configuration, you need to put it into the directory where your local site configuration is sitting. This is of particular importance for DPM sites that need to implement the infamous CMS DPM hack in case they still use RFIO for local file access.**

Let `cmvfs_config` do some configuration for you:

```
cmvfs_config setup
```

Start `autofs` and make it starting automatically after reboot

```
service autofs start
chkconfig autofs on
```

Let `cmvfs_config` do some checks for you:

```
cmvfs_config chksetup
```

This should report errors like wrong setting for the Squids, missing variables and so on.

If you do not see error from the checking above, you can do some basic testing.

## Handling of CE Software Tags

Presently used CMS submission tools do not relay on CE software tags any longer. Most tags are removed from the sites to un-load the BDII.

## Configuration for OSG Sites

OSG maintains documentation on the basic install for OSG sites:

<https://twiki.grid.iu.edu/bin/view/Documentation/Release3/InstallCvmfs>

Follow that document for basic install; for `default.local`, the only required repository is `cms.cern.ch`; `grid.cern.ch` is a usefull addition; the remainders are optional.

The CMS repository requires a customization in `/etc/cvmfs/config.d/cms.cern.ch.local` to find the SITECONF directory (the text in red must be customized):

```
# Important setting for CMS, jobs will not work properly without!
export CMS_LOCAL_SITE=<>
# This only needed if you did not configure Squids in /etc/default.[conf|local]
CVMFS_HTTP_PROXY="http://<Squid1-url>:<port>|http://<Squid2-url>:<port>[|...]"
```

For the setting of the `CMS_LOCAL_SITE` variable you have effectively three options:

1. Use SITECONF files needed by CMS jobs on the worker nodes from CVMFS. In that case, set `CMS_LOCAL_SITE` to your site name, i.e. set `export CMS_LOCAL_SITE=T2_XX_Example` (It may take a while, hours, for changes in your SITECONF area in the GitLab repository to propagate to CVMFS.)
2. (Recommended) Have a configuration management system such as cfengine or puppet keep an up-to-date copy of SITECONF on the local worker node disks, for example in `/etc`. (`export CMS_LOCAL_SITE=/etc/cms/SITECONF/T2_XX_Example`)
3. Keep a SITECONF copy on a shared filesystem and maintain this centrally. This has the downside of making your site depend on NFS or similar technology. (`export CMS_LOCAL_SITE=/nfs/cms/SITECONF/T2_XX_Example`)

For **opportunistic OSG sites** (T3\_US\_OSG), option 1 is recommended.

## Handling of CE Software Tags

Presently used CMS submission tools do not rely on CE software tags any longer. Most tags are removed from the sites to un-load the BDII.

## Job runtime environment

CRAB jobs will prefer `$OSG_APP/cmssoft/cms` over `$CVMFS` if `$OSG_APP` is present in the job's runtime environment. WMAgent jobs do not know to look for CVMFS (currently - this is planned in 2013).

In order to get jobs to use CVMFS, you need to symlink `$OSG_APP/cmssoft/cms` to `/cvmfs/cms.cern.ch`.

## Configuration for/ARC Site

To be written and tested

# Basic testing

Testing should work with any user, not only root.

As first and simple test just list the root directory:

```
ls /cvmfs/cms.cern.ch
```

Try to do some basic setup of CMSSW (gLite for the moment):

```
export VO_CMS_SW_DIR=/cvmfs/cms.cern.ch
source $VO_CMS_SW_DIR/cmsset_default.sh

scramv1 list CMSSW
scramv1 project CMSSW CMSSW_5_0_1_patch3
cd CMSSW_5_0_1_patch3/src
cmsenv
```

One important check is test for the local site configuration.

```
ls -l /cvmfs/cms.cern.ch/SITECONF/local/
```

**It MUST point to your siteconfiguration.** If the link is not resolved properly, double check you settings for `CMS_LOCAL_SITE` in `/etc/cvmfs/config.d/cms.cern.ch.local`.

# SAM Test

The software installation at a site is verified by the `swinst` SAM test. This is fully CVMFS aware and reacts accordingly. Some other tests that rely on the SW installation are being prepared for CVMFS. Pre-production versions are running in pre-production NAGIOS/SAM infrastructure.

# Deployment

Some more details are available from the [WLCG CVMFS Task Force page](#)

## Warning: Added on May 12, 2015

The SITECONF from the installed CVMFS client provides only the job configuration files that are mainly intended to be used in the worker nodes or the machines where the CMSSW executable is executed. Namely, script that updates SITECONF in the CVMFS directory from git only updates site-local-config.xml and storage.xml. Although storage\*.xml is used both in PhEDEx and the worker nodes, sites are not encouraged to use the PhEDEx/storage.xml files in the CVMFS client. Instead, it is suggested that the SITECONF for the PhEDEx machines should be maintained independently from the CVMFS SITECONF because T[0-3]\_CO\_SITENAME/PhEDEx directory in the CVMFS client do not have the necessary PhEDEx agent configuration files for sites.

-- ChristophWissing - 20-Apr-2012

---

This topic: CMSPublic > CernVMFS4cms

Topic revision: r24 - 2018-02-02 - RokasMaciulaitis



Copyright &© 2008-2022 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback