



Vizrt Community Expansion  
**Installation Guide**

3.8.0.130433







Copyright © 2009-2012 Vizrt. All rights reserved.

No part of this software, documentation or publication may be reproduced, transcribed, stored in a retrieval system, translated into any language, computer language, or transmitted in any form or by any means, electronically, mechanically, magnetically, optically, chemically, photocopied, manually, or otherwise, without prior written permission from Vizrt.

Vizrt specifically retains title to all Vizrt software. This software is supplied under a license agreement and may only be installed, used or copied in accordance to that agreement.

### **Disclaimer**

Vizrt provides this publication “as is” without warranty of any kind, either expressed or implied.

This publication may contain technical inaccuracies or typographical errors. While every precaution has been taken in the preparation of this document to ensure that it contains accurate and up-to-date information, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained in this document.

Vizrt’s policy is one of continual development, so the content of this document is periodically subject to be modified without notice. These changes will be incorporated in new editions of the publication. Vizrt may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

Vizrt may have patents or pending patent applications covering subject matters in this document. The furnishing of this document does not give you any license to these patents.

### **Technical Support**

For technical support and the latest news of upgrades, documentation, and related products, visit the Vizrt web site at [www.vizrt.com](http://www.vizrt.com).

### **Last Updated**

13.09.2012





# Table of Contents

<a href="#">1 Introduction</a>	7
<a href="#">2 Installing the Vizrt Community Expansion software</a>	9
<a href="#">3 Vizrt Community Expansion specific setting files</a>	11
<a href="#">4 Database scripts</a>	13
<a href="#">4.1 Example</a>	13
<a href="#">5 Database constants</a>	15
<a href="#">6 Required Plug-ins</a>	17
<a href="#">7 Building and deploying your publication</a>	19
<a href="#">8 3rd party content</a>	21
<a href="#">8.1 Enabling fetching of 3rd party content</a>	21
<a href="#">8.2 Enabling service for creating new 3rd party app services</a>	21
<a href="#">8.3 Distributing the load of 3rd party content fetching</a>	21
<a href="#">8.4 Other 3rd party fetching options</a>	21
<a href="#">8.5 A word of caution</a>	22
<a href="#">9 Troubleshooting</a>	25
<a href="#">9.1 CSRF Security Errors</a>	25





---

# 1 Introduction

Welcome to the Vizrt Community Expansion install guide. Before we start, note that we will use two abbreviations a lot in this guide, namely **ECE** which is the Escenic Content Engine and **VCE**, which is the Vizrt Community Expansion. Furthermore, we use `$VCE_HOME` to mean the location where the VCE bundle has been extracted.

VCE is a module (or plug-in) for the ECE. Before proceeding with this guide, be sure to have your ECE up and running. Installing ECE is extensively documented in the "Escenic Content Engine Installation Guide" and "Escenic Server Administration Guide" available on <http://technet.escenic.com/>







---

## 2 Installing the Vizrt Community Expansion software

Download VCE from <http://technet.escenic.com/>, extract the bundle and create a symlink for it:

```
$ cd /tmp/
$ wget \
  --http-user <technet-user>\
  --http-password <technet-password>\
  http://technet.escenic.com/downloads/community-engine/community-engine-3.8.0.130433.zip

$ cd /opt/escenic/
$ unzip /tmp/community-engine-3.8.0.130433.zip
$ ln -s community-engine-3.8.0.130433 community-engine
```

Then, enter the ECE plugins directory and make it use the community engine symlink you just created.

```
$ cd /opt/escenic/engine/plugins/
$ ln -s ../../community-engine
```

When upgrading, all you need to do is to extract the new bundle in the `/opt/escenic` directory and update the `community-engine` symlink to point to the new release. When you then re-run the assembly tool, it will automatically use the new release as it only relates to the symbolic links.





## 3 Vizrt Community Expansion specific setting files

As of VCE 3.0, the `${user.home}/.ndc-gce/environment.properties` configuration file is no longer required or supported. VCE is now configured using a properties file named `CommunityEngine.properties` which is used in spring configuration. This file should be put in one of the following locations:

- `/etc/escenic/engine/instance/${com.escenic.instance}/com/escenic/community/`
- `/etc/escenic/engine/host/${escenic.server}/com/escenic/community/`
- `/etc/escenic/engine/common/com/escenic/community/`
- `${com.escenic.config}/com/escenic/community/`

Example configurations are available under the `$VCE_HOME/misc/siteconfig/` directory.

The properties file `CommunityEngine.properties` is structured as follows:

```
# Hibernate & database configuration

# This property tells hibernate whether to show the SQL executed by hibernate
hibernate.show_sql=false

# This property tells hibernate which syntax to use when it comes to database
# specific SQL queries
hibernate.dialect=org.hibernate.dialect.MySQLDialect

# This property tells Hibernate where to find the JNDI data source
# Please note that different application servers are configured differently.
# Also the name of the DataSource can be different. For example, Tomcat adds
# a prefix 'java:comp/env' to a configured DataSource. So, if the data source
# is configured by the name 'jdbc/ecome', then the JNDI resource location will
# be as follows:
hibernate.connection.datasource=java:comp/env/jdbc/ecome

# messaging configuration
mail.host=your mailhost
mail.default.from=mailfrom@you.com
# format is <publication-id>:<user-name>
mail.adminusers=12:test,1:glace_admin
```

The `Statistics` plug-in can be configured to use a datasource different than the one used by VCE. To configure `Statistics` plug-in to use a separate data source, `StatisticsPlugin.properties` file has to be configured. This file should be put in one of the following locations:

- `/etc/escenic/engine/instance/${com.escenic.instance}/com/escenic/community/statistics/`
- `/etc/escenic/engine/host/${escenic.server}/com/escenic/community/statistics/`
- `/etc/escenic/engine/common/com/escenic/community/statistics`
- `${com.escenic.config}/com/escenic/community/statistics/`

The properties file `StatisticsPlugin.properties` is structured as follows:

```
# Hibernate database configuration for Statistics plug-in
```



```
# This property tells hibernate whether to show the SQL executed by hibernate
hibernate.show_sql=false

# This property tells hibernate which syntax to use when it comes to database
# specific SQL queries
hibernate.dialect=org.hibernate.dialect.MySQLDialect

# This property tells hibernate where to find the JNDI data source
# Please note that different application servers are configured differently.
# Also the name of the DataSource can be different. For example, Tomcat adds
# a prefix 'java:comp/env' to a configured DataSource. So, if the data source
# is configured by the name 'jdbc/ecome', then the JNDI resource location will
# be as follows:
hibernate.connection.datasource=java:comp/env/jdbc/ecome/statistics
```

Please note that as of VCE release 2.6-13, the database configuration no longer supports any configuration other than JNDI configured DataSource.



## 4 Database scripts

VCE comes with database scripts located in the `misc/database/{mysql,oracle}` folder of the bundle.

These scripts should be installed into the same database schema as the ECE is using, using the same DB user and password. However VCE allows the statistics module database to be installed in a separate database server. The database scripts for the statistics module are suffixed with 'stats'.

Please see the ECE install documentation on how to set up the database and ECE database schema. The VCE scripts should be installed in the following order:

On Oracle databases, you must also add `create` and `view` privileges to the DB user:

```
SQL*Plus> grant create any view to eceuser
```

1. `tables.sql`
2. `constants.sql`

On high volume sites, we recommend that a separate database is used for the statistics module, otherwise, the statistics tables may reside in the same database as the other VCE tables. The following scripts belong to the statistics module:

1. `tables-stats.sql`
2. `views-stats.sql`
3. `constants-stats.sql`

### 4.1 Example

If you have installed the software as described in [chapter 2](#), you may follow this example on how to install all the VCE database scripts on a `mysql` database running on a server called `dbhost` with the schema `ecedb`, user name `eceuser` and password `ecepassword`:

```
$ for e1 in tables tables-stats constants constants-stats views-stats; do
  for ele in `find -L /opt/escenic/community-engine/ -name "$e1.sql" | grep mysql`; do
    mysql -u eceuser -pecepassword -h dbhost ecedb < $ele
  done
done
```

The equivalent for Oracle DB would be:

```
$ for e1 in tables tables-stats constants constants-stats views-stats; do
  for ele in `find -L /opt/escenic/community-engine/ -name "$e1.sql" | grep oracle`; do
    sqlplus eceuser/ecepassword @$ele
  done
done
```





---

## 5 Database constants

Roles and permissions are publication specific. Hence, they cannot be created with the standard SQL scripts. You may do this through the Escenic Web Studio interface, but it is a lot faster to do this using SQL. Find your publication ID and create roles for it. In the example below, the publication ID is **1**:

```
insert into GCE_Role (publicationId, name, description)
values (1, 'GUEST', 'Guest');

insert into GCE_Role (publicationId, name, description)
values (1, 'MEMBER', 'Member');

insert into GCE_Role (publicationId, name, description)
values (1, 'PARTNER', 'Partner');

insert into GCE_Role (publicationId, name, description)
values (1, 'MODERATOR', 'Moderator');

insert into GCE_Role (publicationId, name, description)
values (1, 'ADMIN', 'Administrator');

insert into GCE_Role (publicationId, name, description)
values (1, 'SECTION OWNER', 'Section Owner');

insert into GCE_Role (publicationId, name, description)
values (1, 'GROUP MEMBER', 'Group member');

insert into GCE_Role (publicationId, name, description)
values (1, 'GROUP MODERATOR', 'Group moderator');

insert into GCE_Role (publicationId, name, description)
values (1, 'GROUP OWNER', 'Group owner');

insert into GCE_IdGenerator values('roleId', 0);
insert into GCE_IdGenerator values('permissionId', 0);

update GCE_IdGenerator set value=(select max(roleId) + 1 from GCE_Role) where id='roleId';
update GCE_IdGenerator set value=((select max(permissionId) + 1 from GCE_Permission) where
id='permissionId');
```

For all permissions you need that is application dependent you may do the same, inserting values into the **GCE\_Permission** table. These permissions are the ones used in your **security** publication resource file

For example, to add a permission called **rate**, you would do:

```
insert into GCE_Permission (publicationId, name, shortKey, description)
values (1, 'rate', 'rate', 'User may rate content');
```

You also have to run above two lines of update query







---

## 6 Required Plug-ins

A number of VCE functionality requires following plug-ins installed:

**Forum** : The functionality of commenting on a content in a community site has to be implemented using the Forum plug-in. VCE provides history and moderation functionality for the comments created using Forum plug-in.

Required Forum version is **2.3.1** or later.

**Lucy** : This is an Escenic plug-in that adds search capabilities providing an interface to **Apache Solr**. Searching in the **Dashboard** publication requires Lucy plug-in installed.

Required Lucy version is **4.1-4** or later.





---

## 7 Building and deploying your publication

How to create, build and deploy a publication is described in detail in the "Escenic Content Engine Installation Guide", available on <http://technet.escenic.com/engine/43/guides/>






---

## 8 3rd party content

---

### 8.1 Enabling fetching of 3rd party content

Add the following to your Nursery configuration, e.g.: `/etc/escenic/engine/common/Initial.properties`

```
service.800-third-party-application-fetcher=/com/escenic/community/thirdparty/fetcher/ThirdPartyApplicationFetcher
```

---

### 8.2 Enabling service for creating new 3rd party app services

Before creating a 3rd party app service, a Nursery service `ApplicationServiceArticleEventListener` needs to be enabled. This Nursery service creates a dedicated section for new 3rd party app services. Without this section, the 3rd party content fetching will not work. To enable this service, set the following in `/com/escenic/community/thirdparty/ApplicationServiceArticleEventListener.properties`:

```
enabled=true
```

This service should only be enabled on one ECE instance in a multi-server environment.

---

### 8.3 Distributing the load of 3rd party content fetching

You may want to distribute the load of fetching content among your ECE servers. To do this, set the ECEs to only fetch a certain set of app services and/or publications. Edit your host specific configuration, e.g. in `/etc/escenic/engine/hosts/myhost/com/escenic/community/thirdparty/fetcher/ThirdPartyApplicationFetcher.properties`

```
appServicesIncluded=flickr,picasaweb,last.fm
publicationNamesIncluded=mysite
```

---

### 8.4 Other 3rd party fetching options

In addition to the parameters for narrow down the scope of the fetching mentioned above, these may be of interest to you:

Parameter name	Description
interval	Time in seconds between each time VCE will start a new batch of fetching

Parameter name	Description
	content. Try to keep this as high as possible to not grind external services to a halt, or being locked out from 3rd party services.
delay	Delay in seconds before the first batch. The default should be fine, but if your app server is particularly slow to start up, increase this one. Five minutes (60 * 5) should be fine.
appServiceThreadPoolSize	The size of the initial fetcher pool (there are two pools). You may set this variable to the number of 3rd party app services you have. There is no need to set it any higher.
userAppThreadPoolSize	<p>The number of thread to use for each app service to fetch user applications. The default value should be fine for most cases, but you may want to increase it to make fetching go faster. If you increase it, watch the number of open files the app server keeps. One quick way of doing this on a Linux system is:</p> <pre>\$ ls /proc/&lt;app server PID&gt;/   wc -l</pre> <p>It is important that the Unix user that starts the app server is allowed to open enough file handles. If you are getting exceptions like:</p> <pre><b>SEVERE: Socket accept failed</b> <b>org.apache.tomcat.jni.Error: Too many open files</b></pre> <p>You must increase the file handle size. To see the current number of file handles a process can open (applicable at least on NetBSD, FreeBSD and GNU/Linux), do:</p> <pre>\$ ulimit -n</pre> <p>Refer to your OS documentation on setting this permanently.</p>

## 8.5 A word of caution

Different 3rd party service providers have different rules on using their APIs (including feed URLs) for fetching content. Some have restrictions on the number of requests a client (i.e. the public IP of your service) can request per hour or day. Others have restrictions on using their services for commercial



use. For more information, please refer to the API documentation of the specific service you want to make available to your site's users.

If you are experiencing problems with some 3rd party services, the solution may be to develop a proxy server that VCE relates to. This proxy deals with distributing the requests among several public IPs or provides API key information for the third party service. This is more or less the way Facebook and iGoogle application works. These sites cloak (using IFRAMES) the app which is hosted on a dedicated server entirely. The only thing that makes it blend in with the mother web site (Facebook and iGoogle) is the skinning of the contents (and often the lack of IFRAME scrollbars). Depending on the 3rd party apps you want to make available on your site, you may or may not need to develop a proxy service for (some of) them.

You will find that many services will work effortlessly without any additional proxy layer, whereas others will pose problems as the user base increases. Reading through the 3rd party service API documentation will help you in making the right measures to ensure seamless integration.







---

## 9 Troubleshooting

This section contains descriptions of common problems and solutions to them.

---

### 9.1 CSRF Security Errors

#### Problem

Using Tomcat 7, every request from the Vizrt Community Expansion is rejected as a potential **CSRF (cross-site request forgery)** attack, resulting in this session error message:

```
A request has been denied as a potential CSRF attack
```

#### Solution

In Tomcat 7 the `useHTTPOnly` option is set to `true` by default, and this setting is not supported by the Vizrt Community Expansion's qualification module. To fix the problem, edit `Context.xml` in your Tomcat installations `conf` directory and set `useHttpOnly` to `false`:

```
<Context useHttpOnly=false >
```