



“ Platform for **C**onvergence research and **U**nification of **B**ig **E**-resources ”



Developer Manual

Table of Contents

Chapter 1 : Outline	1
Document Outline	1
Chapter 2 : JAVA Installation.....	1
JAVA Installation	1
Chapter 3 : DB Server Settings.....	2
MySQL Installation	2
Chapter 4 : WEB Server Setting	3
Apache Httpd Server Settings.....	3
Chapter 5 : WAS Server Setting	4
Tomcat server installation.....	4
Apache Tomcat Link.....	4
Chapter 6 : Eclipse Installation.....	5
Eclipse (developer PC).....	5
Installation of egit on Eclipse.....	6
Chapter 7 : P-CUBE Project Check Out using GitHub	8
P-CUBE Check Out.....	8
Chapter 8 : P-CUBE Project Installation	13
Project extraction and upload	13
Environment setting file upload of P-CUBE project	15
Chapter 9 : Environment Settings of P-CUBE Project	15
File setting of cube.cfg	15
Setting for log storage.....	17
Environment setting for Tomcat	18
solr settings	19
Setting for Script Launcher execution	19
Setting on usage or not	21

Chapter 1 : Outline

Document Outline

1. This document describes the settings for installation and operation for P-CUBE.
2. P-CUBE is developed with JAVA, and can be operated at any operation system which has JDK installed.
3. This document is a (developer) manual for P-CUBE, version of 1.0.

Chapter 2 : JAVA Installation

JAVA Installation

1. JDK installation is required since P-CUBE has been developed based on JDK.
2. Please download JDK at <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
3. Please install JDK using reference below.

```
# mkdir -p /usr/java
# cd /usr/java
# chmod 700 /tmp/jdk-6u45-linux-x64.bin
# /tmp/jdk-6u45-linux-x64.bin
```

4. Setting an environment variables

```
# export JAVA_HOME=/usr/java/jdk1.6.0_10
# export PATH=$JAVA_HOME/bin:$PATH
```

Chapter 3 : DB Server Settings

MySQL Installation

1. download MySQL from <http://dev.mysql.com/downloads>

2. If you have MySQL already installed, please delete as follows.

```
[local-host]# rpm -qa | grep -i mysql
mysql-5.0.22-2.1.9.1
mysqlclient10-3.23.58-4.RHEL4.1
```

```
[local-host]# rpm -e mysql --nodeps
warning: /etc/my.cnf save as /etc/my.cnf.rpm.save
[local-host]# rpm -e mysqlclient10
```

3. Refer to below and install MySQL package.

```
[local-host]# rpm -ivh MySQL-server-community-5.1.25-0.rhel5.i386.rpm MySQL-client-
community-5.1.72-1.rhel5.x86_64.rpm
```

4. Check on MySQL installation is as follows.

```
[local-host]# /usr/bin/mysql_secure_installation
```

5. MySQL access as root user is as follows.

```
[local-host]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 13
Server version: 5.1.25-rc-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. '\c' to clear the buffer.

mysql>
```

6. The start and the end of MySQL are as follows.

```
[local-host]# service mysql start
```

- Input each DB name, DB username and password to use into spaces for 'DBNAME', 'USERNAME', 'PASSWORD'.

```
mysql> CREATE DATABASE 'DBNAME' DEFAULT CHARACTER SET utf8 COLLATE
utf8_general_ci;

mysql> use mysql;

mysql> insert into user (host,user,password)
values('localhost','USERNAME',password('PASSWORD'));

mysql> insert into user (host,user,password) values('%','USERNAME',password('PASSWORD'));

mysql> insert into db
values('%','DBNAME','USERNAME','y','y','y','y','y','y','y','y','y','y','y','y','y','y','y','y','y');

mysql> insert into db
values('localhost','DBNAME','USERNAME','y','y','y','y','y','y','y','y','y','y','y','y','y','y','y','y','y');

mysql> grant all privileges on . to 'USERNAME'@'%' identified by 'PASSWORD';
```

Apache Httpd Server Settings

- ```
gunzip -d httpd-2_0_NN.tar.gz
tar xvf httpd-2_0_NN.tar
```

- ```
# up2date httpd
```

4. Start of an Apache httpd is as follows.

```
# chkconfig httpd on  
# /etc/init.d/httpd start
```

*Apache Httpd Server is a WEB server used for P-CUBE development and test, and it is not required to use. WEB server can be changed depending on a situation of an institute.

Chapter 5 : WAS Server Setting

Tomcat server installation

1. Download Tomcat from <http://tomcat.apache.org/download-60.cgi>
2. Unzip the file you downloaded as follows.

```
# tar zxvf /tmp/apache-tomcat-6.0.37.tar.gz
```

3. Find a route of Tomcat server and get into it.

```
# cd tomcat6
```

4. The start and stop of server are as follows.

```
# /bin/startup.sh  
# /bin/shutdown.sh
```

Apache Tomcat Link

1. Tomcat connector can be downloaded from <http://tomcat.apache.org/download-connectors.cgi>
2. Unzip the file you downloaded as follows.

```
# tar zxvf /tmp/tomcat-connectors-1.2.37-src.tar.gz  
# cd tomcat-connectors-1.2.37-src
```

```
# cd native
# ./buildconf.sh
```

Install as follows. 'Apache route' means the installation route for Apache.

```
# ./configure --with-apxs=/Apache route/bin/apxs
# ls /Apache route/modules/mod_jk.so
```

Creates a linkage file

```
# /Apache route/conf/workers.properties
# /Apache route/conf/uriworkermaps.properties
```

* Linkage of Apache and Tomcat is only applicable when using Apache as WEB server and Tomcat as WAS server.

Chapter 6 : Eclipse Installation

Eclipse (developer PC)

1. Download Eclipse from <http://www.eclipse.org/downloads> and unzip the file.



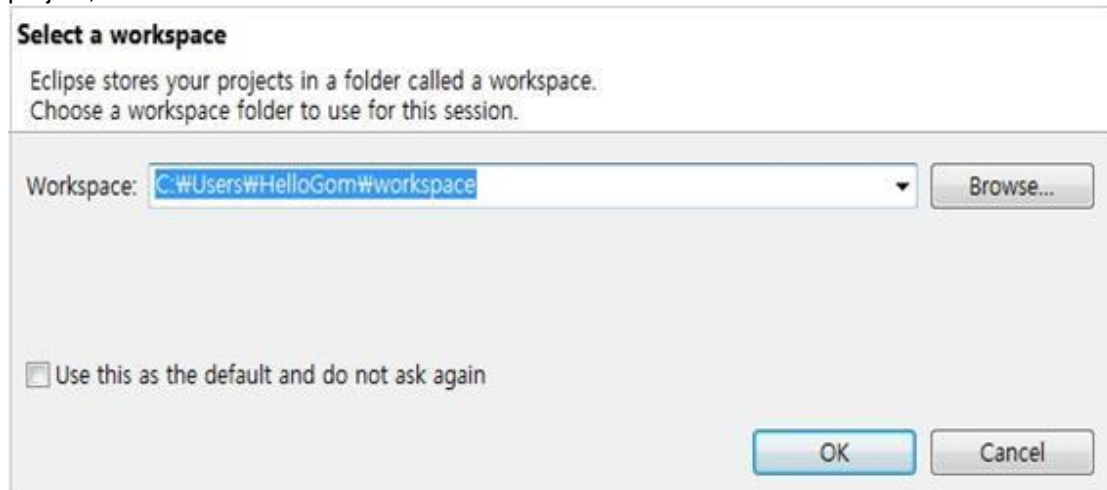
2. Create a shortcut icon of Eclipse execution file, and add to properties as follows.

```
C:\eclipse\eclipse.exe -vmargs -Xms128M -Xmx256M
```

Properties description

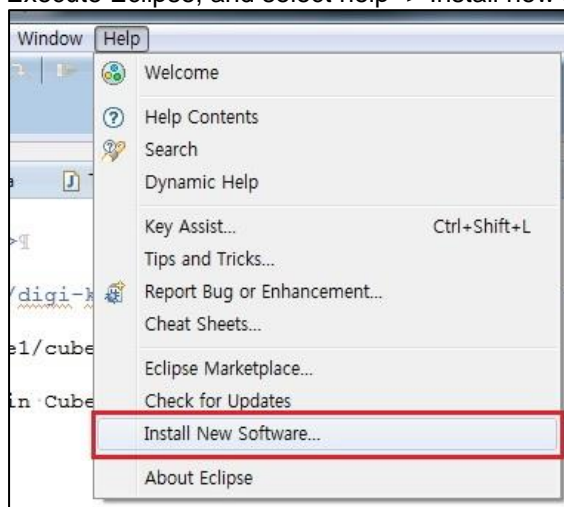
- C:\eclipse\eclipse.exe : Location of Eclipse execution file
- vmargs -Xms128M -Xmx256M : Eclipse memory settings is recommended for securing memory.

3. Executing Eclipse will lead to a workspace setting as follows. After setting up of location for project, click "OK" button.



Installation of egit on Eclipse

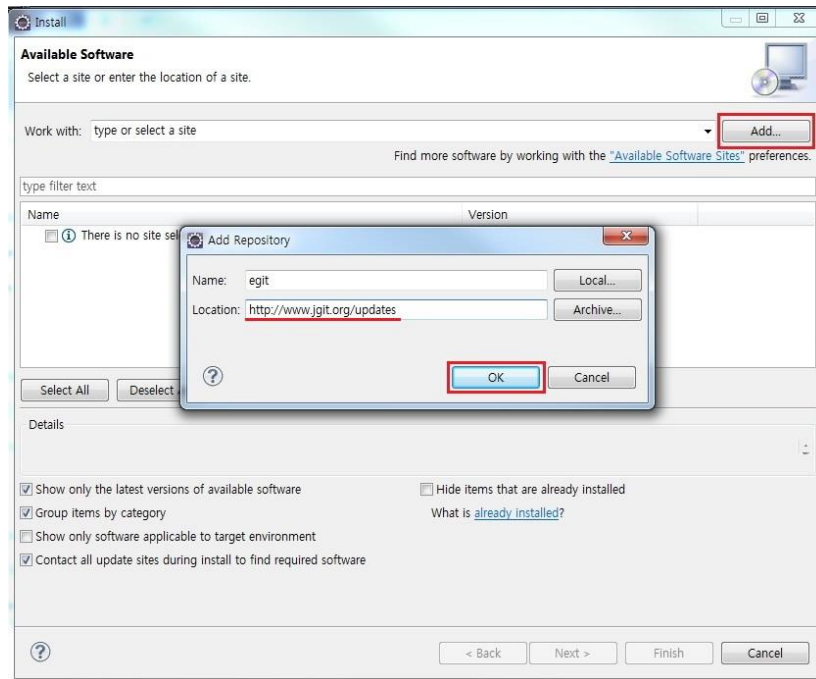
1. Source of P-CUBE is registered at GitHub. For synchronization with source, egit is recommended.
2. Execute Eclipse, and select help -> Install new software above.



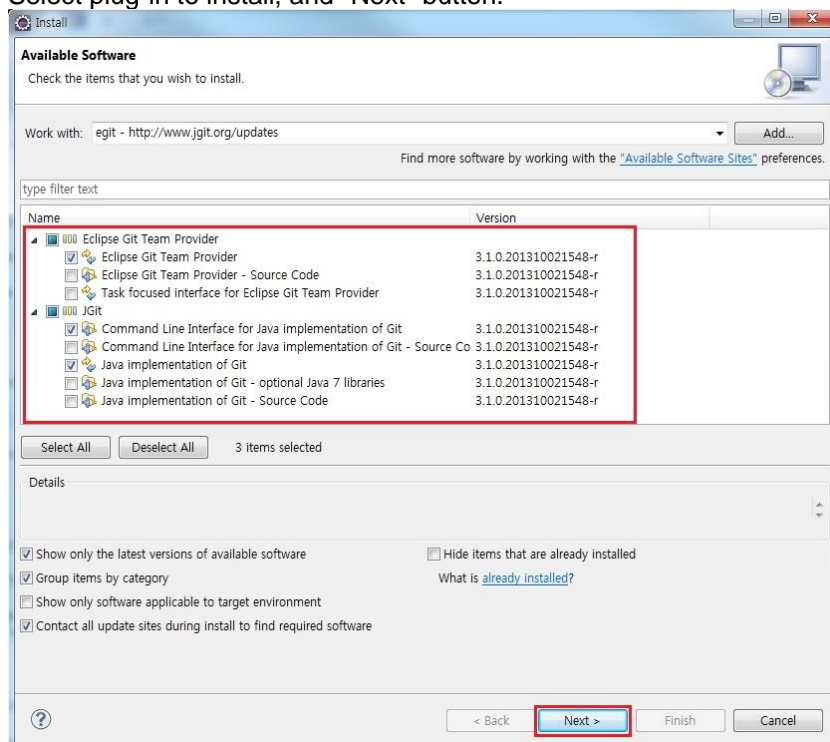
3. Click add to add a site link.

Name : egit

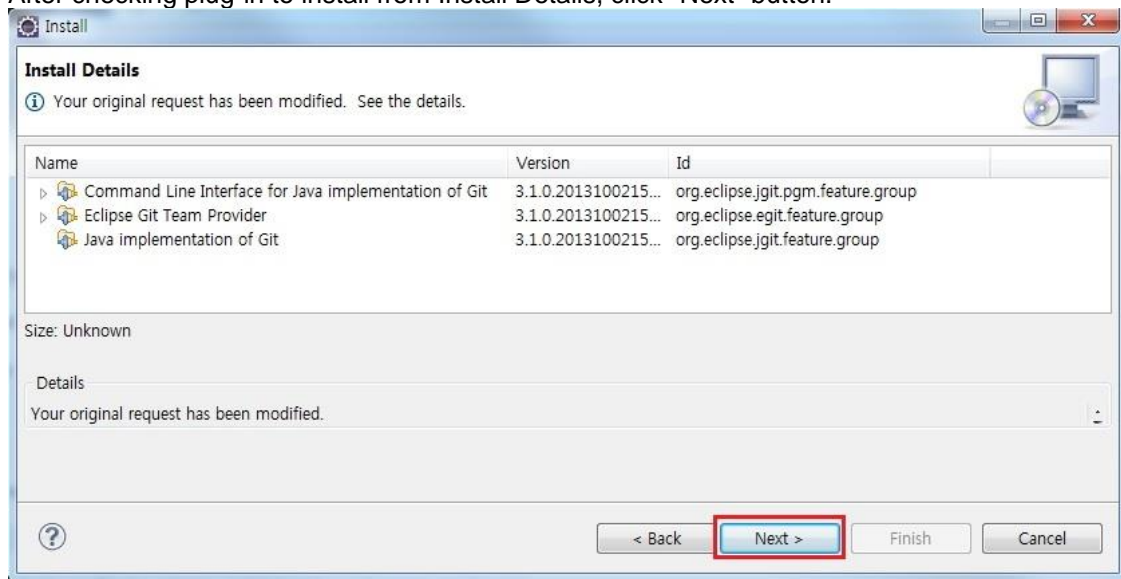
Location : <http://www.jgit.org/updates>



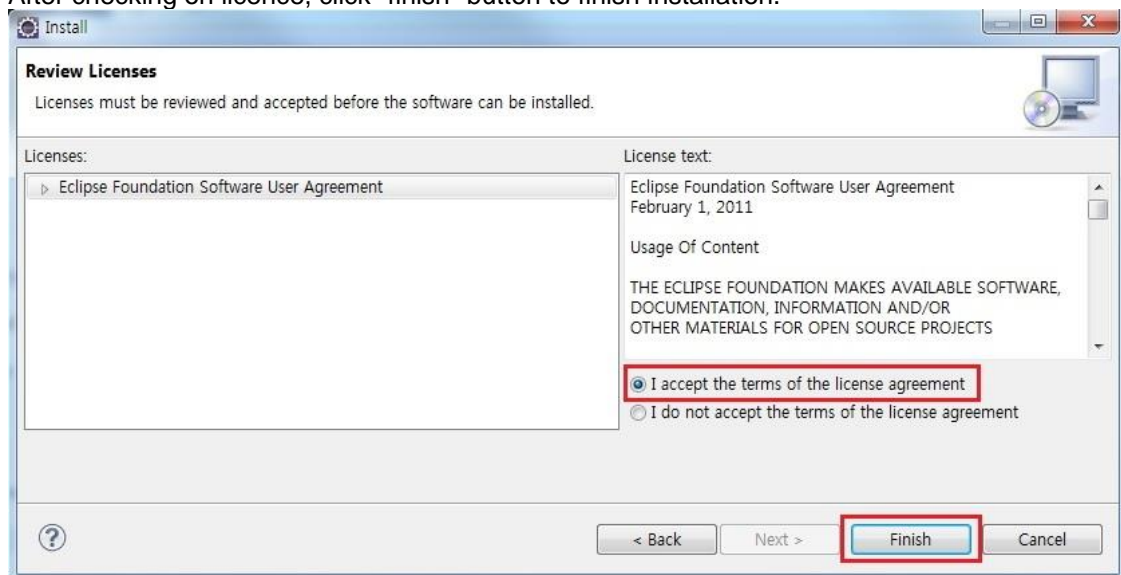
4. Select plug-in to install, and "Next" button.



5. After checking plug-in to install from Install Details, click "Next" button.



6. After checking on licence, click "finish" button to finish installation.



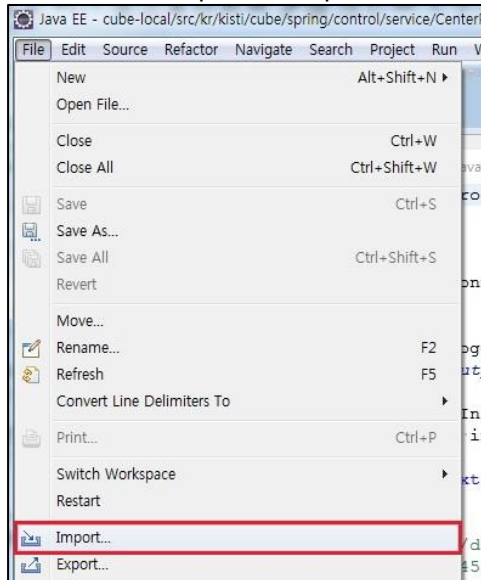
7. If you restart Eclipse, the installation will be completed.

Chapter 7 : P-CUBE Project Check Out using GitHub

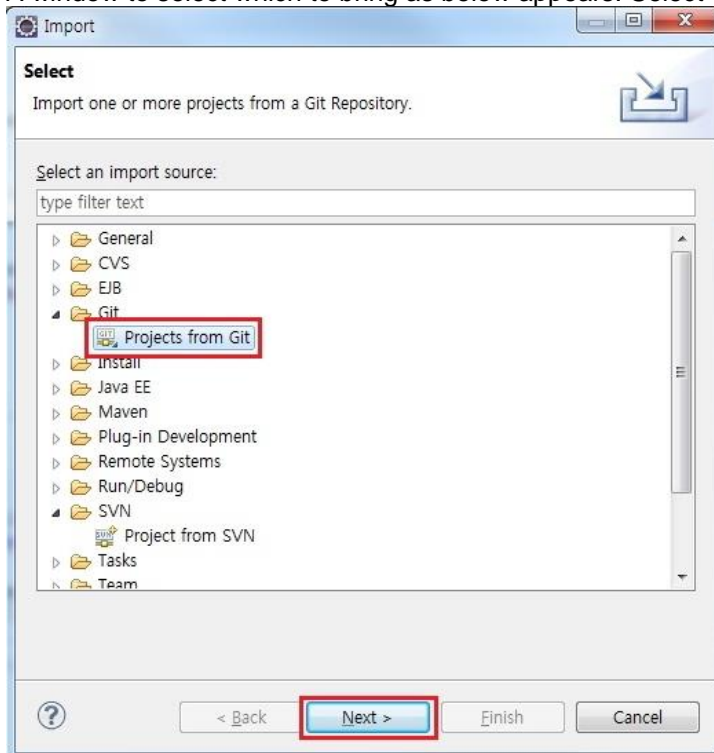
P-CUBE Check Out

1. GitHub as system of dispersed version management is used for shape management of P-CUBE project.

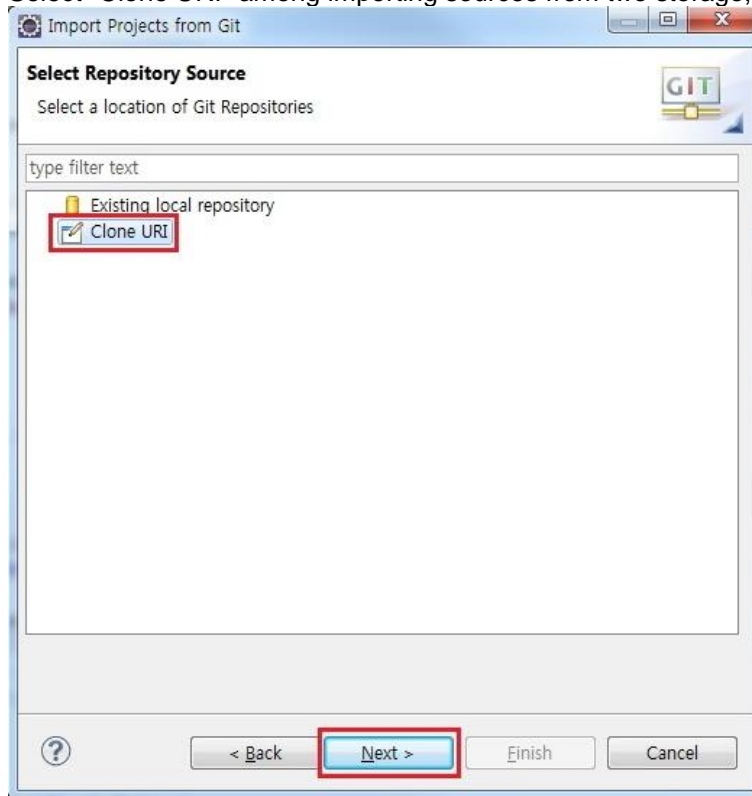
2. Select File -> Import of Eclipse.



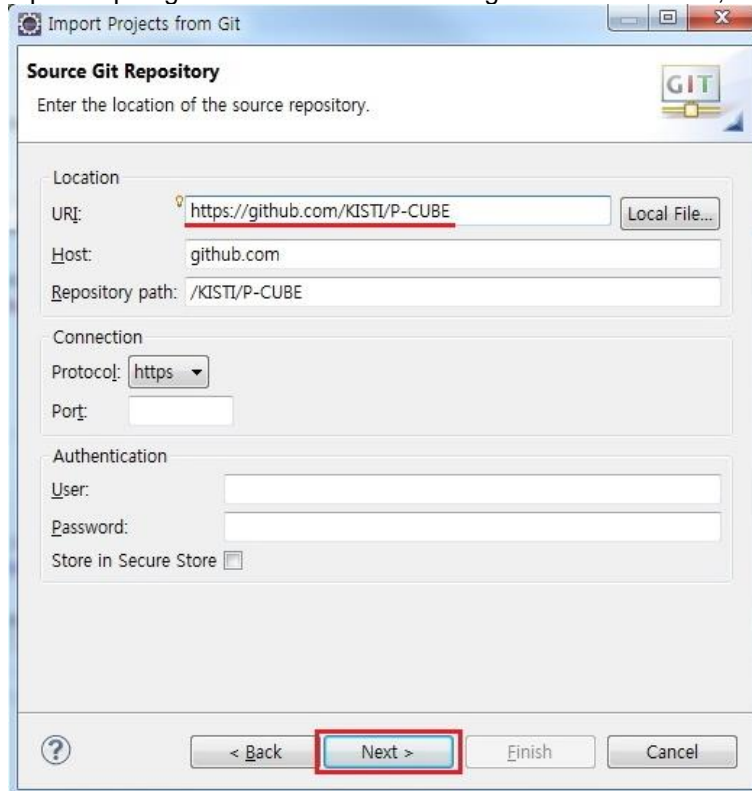
3. A window to select which to bring as below appears. Select "Project from Git" and "Next" button.



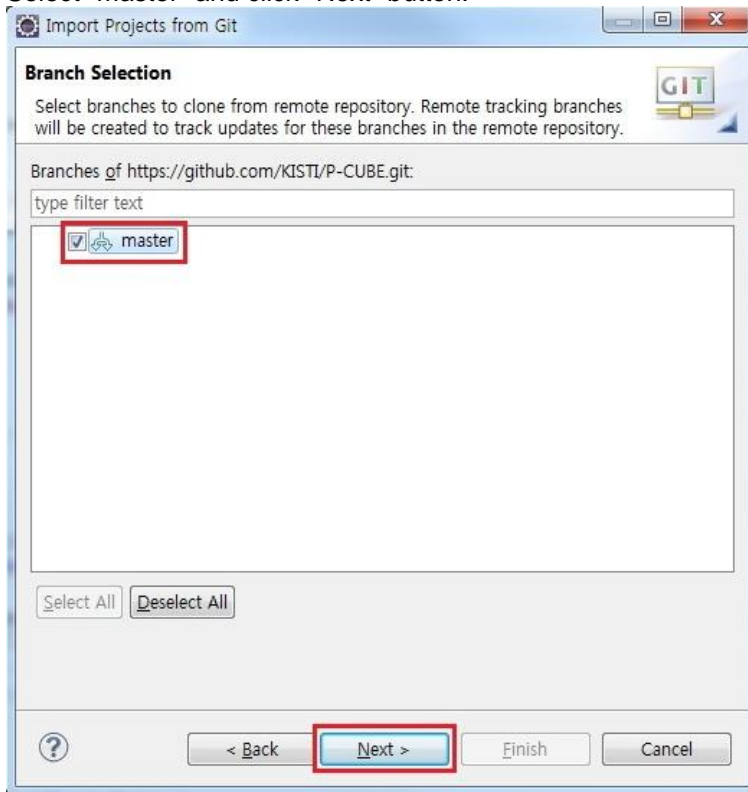
4. Select "Clone URI" among importing sources from two storage, and click "Next" button.



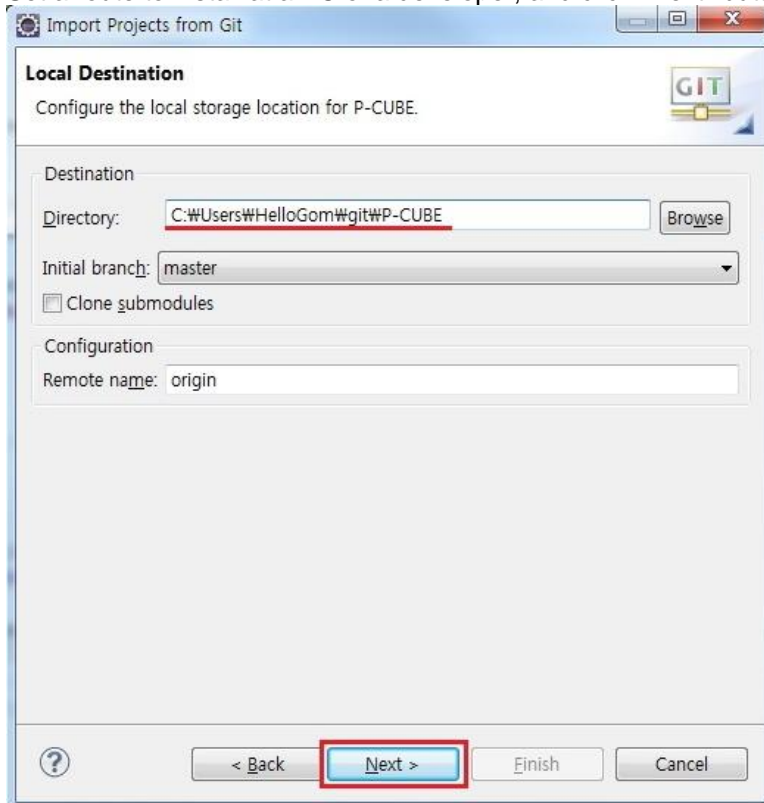
5. Input "https://github.com/KISTI/P-CUBE.git" to URI as below, and click "Next" button.



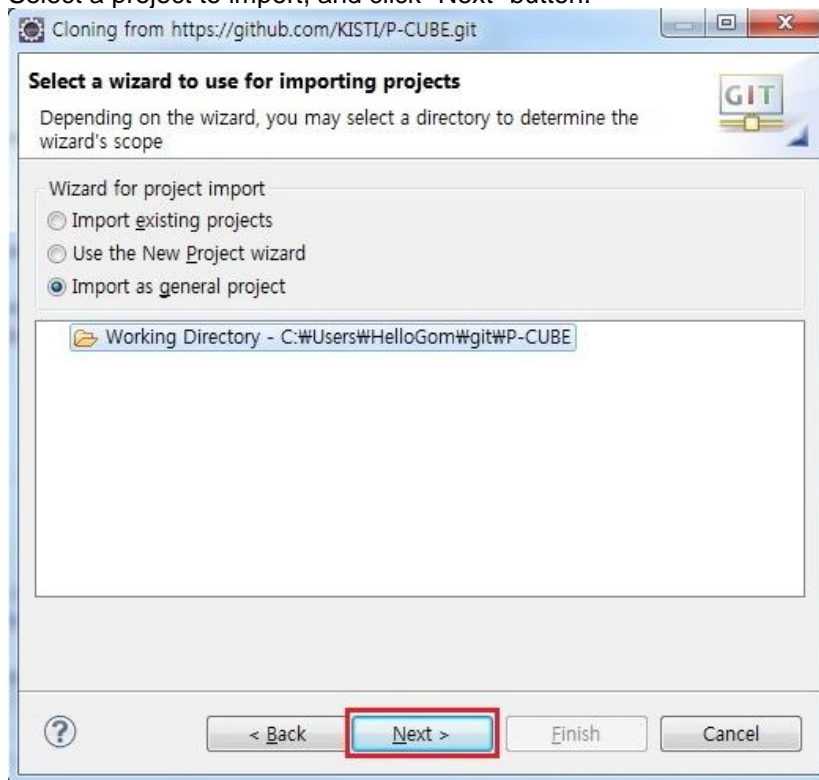
6. Select "master" and click "Next" button.



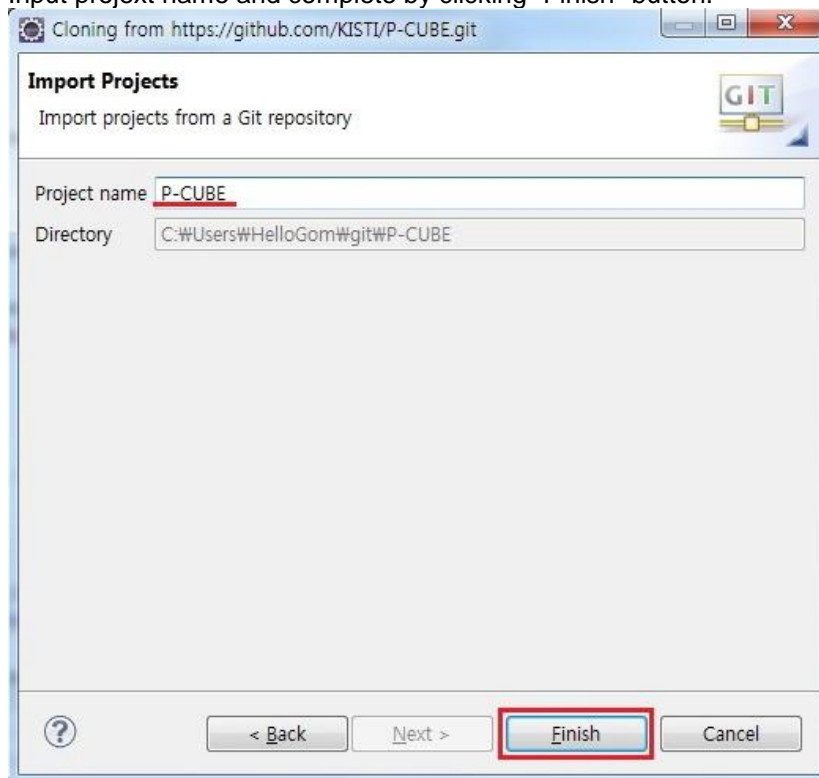
7. Set a route to install at a PC of a developer, and click "Next" button.



8. Select a project to import, and click "Next" button.



9. Input project name and complete by clicking "Finish" button.



10. Project creation is completed.

11. Check out OAI Project in the same way with CUBE.

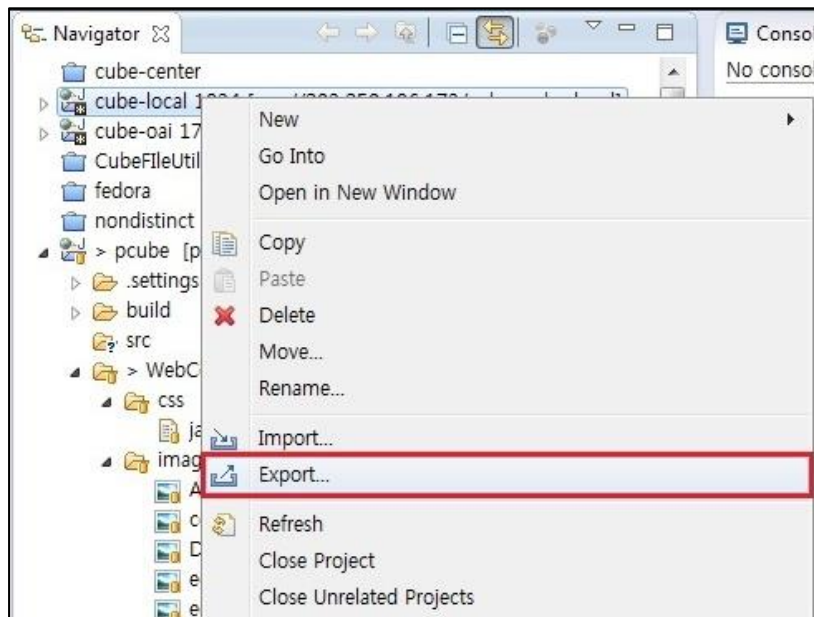
OAI Project is information harvesting and open project based on OAI-PMH, international standard protocol. OAI project is necessary for information harvestment and application.

URI of number 5 is "https://github.com/KISTI/P-CUBE-OAI.git".

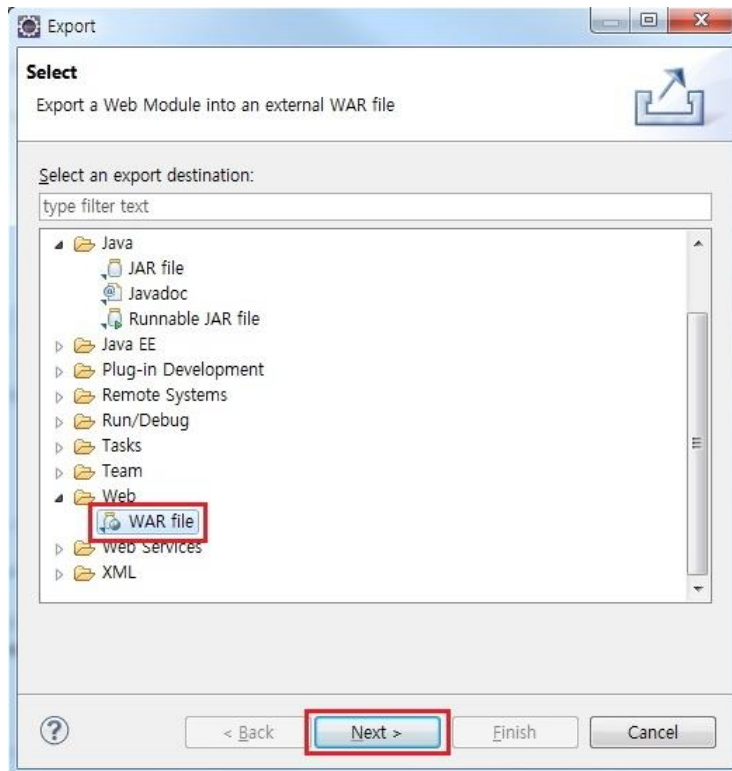
Chapter 8 : P-CUBE Project Installation

Project extraction and upload

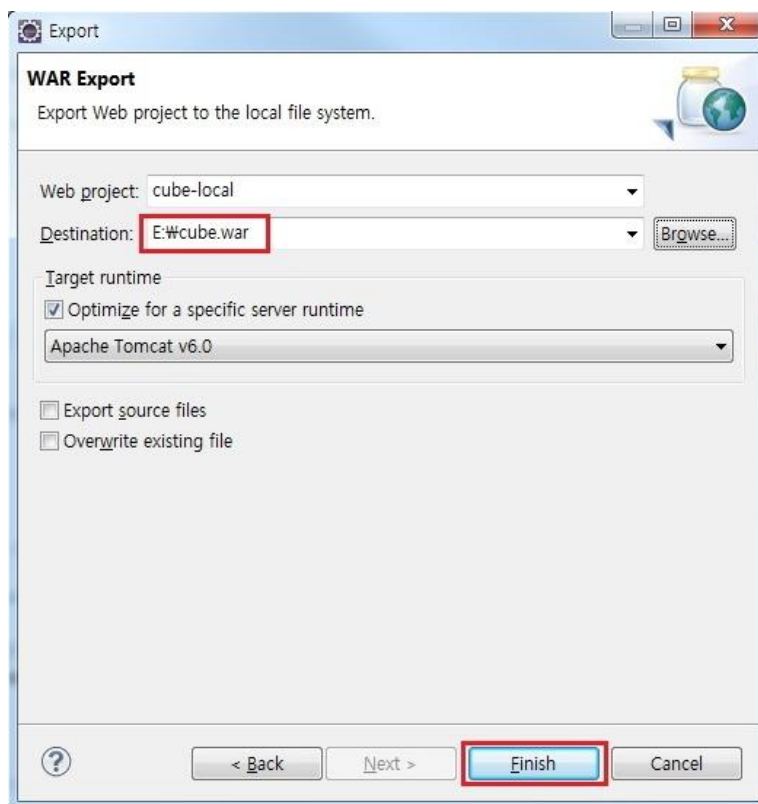
1. After clicking on the right mouse button on "Cube" of Eclipse and "OAI" project, click "Export".



2. On Export list, select Web / WAR file and click "Next >" button.



3. After saving the route and name of the file created, click "finish" button.



Save the name of Export WAR file of "CUBE" as cube.war.

Save the name of Export WAR file of "OAI" as oai.war.

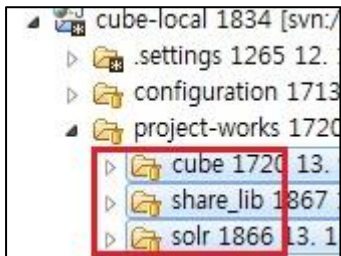
4. Execute FTP program and connect to a server.
5. Please upload "cube.war", and "oai.war" file exported to a webapps route in Tomcat.

/tomcat6/webapps/

Environment setting file upload of P-CUBE project

A resource route of a server for uploading an environment setting folder below is used for 'data/home/digi-kor/'. From this time forward, we will use a directory which saves an environment setting file named 'resource route'. A resource folder is changable depending on environment of each institute.

1. Please connect to a server, executing FTP program.
2. Copy 'cube', 'solr', 'share_lib' folders in 'project-works' folder from "CUBE" of Eclipse to the resource route of a server.



3. Please start Tomcat again for project creation using war file.

Chapter 9 : Environment Settings of P-CUBE Project

File setting of cube.cfg

Environment setting is a setting for files uploaded to server. Please edit a route and a file carefully.

1. Proceed a file setting of cube.cfg which is a file for 'CUBE' basic setting in 'cube/config' route of 'resource route'.

Setting for required management information

cube.dir : cube folder route of 'resource route'. E.g.) /data/home/digi-kor/cube/

cube.hostname : It is a HOST address for CUBE. E.g.) mdr.kisti.re.kr

cube.baseUrl : It is a URL address for CUBE. E.g.) http://www.datacite.kr

cube.name : It is a name of a site for CUBE. E.g.) P-CUBE

db.url : A route for server MySQL. E.g.) jdbc:mysql://127.0.0.1/cube?characterEncoding=utf-8

db.driver : It is a DB driver. E.g.) core.log.jdbc.driver.MysqlDriver

db.username : Username of DB.

db.password : It is a password of DB user.

handle.prefix : It is an unique string used for distinguishing system. E.g.) kisti_kdr

Input prefix of system given by CNRI to use for handle.prefix. Functions for linkage to CNRI are excluded from P-CUBE, so please refer to other sources.

Setting server information for sending an e-mail.

mail.server : It is a name for a server when sending an e-mail with a name of e-mail server.

mail.server.username : E-mail Address, An e-mail address for sending an e-mail.

mail.server.password : E-mail password, it is a password for sending e-mail address.

mail.admin : It is an e-mail address of a administrator which receives an alarm e-mail when warning of monitoring occurs.

Settings related google analytics

xmlui.google.analytics.change.key : It is a google analytics key value. You can get site statistics information after registering to google analytics. If you do not wish to use the function of google analytics, please leave it empty.

Settings related google chart

google.chart.username : An e-mail address which has an authority for searching websites registered to google analytics.

google.chart.password : A password which has an authority to search google analytics.

google.chart.instituteCode : It is used as linkage authority code of google analytics for access statistics of statistics menu.

NDSL openAPI search settings

ndsl.openapi.key : ndsl.openapi.key : It is key value for NDSL linkage. It is used for showing information of thesis similar to a related item of detail item view. Issue key at <http://nos.ndsl.kr>

System monitoring settings

cube.system.monitoring.use : Usage of system monitoring or not. It is used for administrators to check user information on CPU, MEMORY, Storage and etc. The basic setting value is 'N'.

Compression sending setting

rapidant.use.yn : Availability of rapidant applet for uploading a file when registering an item. The basic setting value is 'N'.

subject classification setting

subject.category.yn : subject.category.yn : Availability of subject classification selection and preferred subject classification of a user. The basic setting value is 'Y'.

Triple Store (Mulgara) Setting

mulgara.use.yn : Availability of mulgara which shows lists of related information items of detailed item window at the bottom of the page. The basic setting value is 'N'.

Setting Item Exporting

Set exporting method and order of exporting item according to the method, as exporting information of item at item detailed screen.

itemexport.count : The number of exporting methods. The number should be same as the number of exporting methods to be set.

itemexport.1 : The number after the 'itemexport.' defines order displayed at screen. Add by increasing the number to input several methods. Setting content should input the name of exporting method and write format. Format should be written as 'element name|displayed name on screen', arranged and differentiated with comma.

E.g.) datacite Format:contributor|Writer, title|Title, publisher|Publisher, identifier.uri|URL

Setting for log storage

log4j is a java based logging utility and saves log history as a file.

1. Please proceed log4j.properties file settings, which is "CUBE" log setting file in 'cube/config' route of 'resource route'.

The three settings below are routes for log file of CUBE to be create.

log4j.appender.A1.File

log4j.appender.A2.File

log4j.appender.A3.File

Environment setting for Tomcat

WAS Server was described based on Tomcat. When using other kinds of WAS server, please refer to a product manual.

1. Change DB route (including DB name), username, password in **dataSource** bean of WEB-INF/classes/config/applicationContext.xml file of "OAI" and "CUBE" project from webapps folder of Tomcat.

```
<property name="url">
  <value>jdbc:mysql://127.0.0.1/cube?characterEncoding=utf-8</value>
</property>
<property name="username">
  <value>__</value>
</property>
<property name="password">
  <value>__</value>
</property>
```

2. Change config file route, username, and password for **cube-config**, **cube.dir** of WEB-INF/web.xml file for "CUBE", "OAI" project in webapps folder of Tomcat.

```
<context-param>
  <description>
    The location of the main Cube configuration file
  </description>
  <param-name>cube-config</param-name>
  <param-value>/data/home/digi-kor/cube/config/cube.cfg</param-value>
</context-param>
<context-param>
  <description>
    The location of the main Cube configuration file
```

```

</description>
<param-name>cube.dir</param-name>
  <param-value>/data/home/digi-kor/cube</param-value>
</context-param>

```

Change fields as below one more time for OAI project.

```

<context-param>
  <param-name>properties</param-name>
  <param-value>/data/home/digi-kor/cube_dev/config/oaicat.properties</param-value>
  <description>
    The location of the oaicat.properties file containing
    datapairs used to initialize the OAI repository software.
  </description>
</context-param>

```

solr settings

1. Create solr.xml file to conf/catalina/localhost in Tomcat.

Contents of a file (Please input correctly, fitting for solr folder route of 'resource route')

```

<Context docBase="/data/home/digi-kor/solr_home/solr-4.3.1.war" debug="0"
crossContext="true" path="/solr">
  <Environment name="solr/home" type="java.lang.String"
value="/data/home/digi-kor/solr_home/cube" override="true"/>
</Context>

```

2. Add solr route to 'common.loader' of catalina.properties file on conf route in Tomcat.

Add after the text written, fitting for the route.

./home/solr/ext/*.jar

Setting for Script Launcher execution

1.
 1. Create a jar file from the route of 'webapps/cube/WEB-INF/classes' in Tomcat. The created file is added to cube/ lib route in 'resource folder'.

Creates as below, after accessing a route on server command window.

```
# jar cvf pcube.jar kr config spring
```

2. Copy all the library file from 'tomcat6/webapps/cube/WEB-INF/lib' route of server, and add to the route for 'cube/lib' of 'resource folder'.
3. Copy 'servlet-api.jar' file of 'tomcat6/lib' route of a server and add to 'cube/lib' route of 'resource folder' by copying.
4. Instructions for Script Launcher

Script Launcher simply performs with prompt command. It is set for using functions of DB initialization, administrator creation, harvesting, and index of MySQL at P-CUBE.

Launcher only those registered at 'cube/config/launcher.xml' is applicable.

Use Script Launcher with command in 'cube/bin' folder of command window for server.

The below shows a few important instruction among registered launcher.

database checking (DB linkage test)

```
test-database
```

init database (DB initialization using sql file)

```
init-database <sql_file>
```

create administrator (Create Administrator)

```
create-administrator -e email -f first_name -l last_name -c language -p password
```

harvest (Harvesting from external source)

```
PING OAI server : harvest -g -s oai_source -i oai_set_id
```

```
RUNONCE : harvest -o -e eperson(userEmail) -c collectionId -t harvest_type -a oai_source -i oai_set_id -m metadata_format
```

```
SETUP : harvest -s -c collectionId -t harvest_type -a oai_source -i oai_set_id -m metadata_format
```

```
RUN : harvest -r -e eperson(userEmail) -c collectionId ( i.e) pcube harvest -r -e admin@kisti.re.kr -c 7 )
```

```
START harvest scheduler : harvest -S
```

```
RESET all harvest status : harvest -R
```

```
PURGE all harvestable collections : harvest -p -e eperson(userEmail)
```

Index (Index task)

Index initialization : index-init

Index update : Index-update

update-handle-prefix (handle prefix value change en bloc)

update-handle-prefix <old handle> <new handle>

Setting on usage or not

1. Use of system monitoring

Edit file '.bash_profile' in server root route. LD_LIBRARY_PATH add server route.

```
# vi .bash_profile
```

```
# export LD_LIBRARY_PATH=/data/home/share_lib
```

```
# source .bash_profile
```

2. Use of compression and transmission

jsvc installation

It sets rapidantServerDaemon.sh file in cube folder.

RAPIDANT_HOME : It is a route for installing compression sending server.

RAPIDANT_USER : It is a user name of a server. (Access user of server prompt)

JAVA_HOME : A route of JDK installed.

JSVC_PATH : A route of jsvc installed.

COMMONS_DAEMON_PATH : A route of commons-daemon file.

\$JSVC_PATH : Setting port number for '-p' of rapidant.RapidantDaemon.

3. Use of Triple store(Mulgara)

Edit 'mulgara.data.name' of 'cube.cfg' file as a name of server for data building setting.

```
mulgara.data.name = cube
```

Execute a file of cube/config/mulgara-2.1.13.jar from server.

```
# cd cube/config
```

```
# nohup java -jar mulgara-2.1.13.jar
```

4. Add code group

Create group of code used for P-CUBE such as item submission, user registration and etc.

Input as below into db command window. Input each code group name, code group explanation, and an English name of code group into 'TYPE_NAME', 'TYPE_NOTE', 'TYPE_ENG_NAME'

```
mysql> INSERT INTO INPUT_PARAM_TYPE (TYPE_ID, TYPE_NAME, TYPE_NOTE,  
TYPE_ENG_NAME) (SELECT IFNULL(MAX(TYPE_ID),0)+1, 'TYPE_NAME', 'TYPE_NOTE',  
'TYPE_ENG_NAME' FROM INPUT_PARAM_TYPE)
```