

OpenOLAP インストールガイド

R 1.2.0

2004 年 10 月



OpenOLAP インストールガイド

R 1.2.0

商標

本書に記載されている社名および商品名は、それぞれ各社の商標または登録商標です。

なお、本文中には TM、® マークは明記していません。

目次

1 前提条件	1
2 インストール方法	2
1.1 インストール方法	2
1.2 MetaTool	2
1.3 Designer / Viewer 共通設定	6
1.4 Designer	6
1.5 Viewer	7
1.6 サンプルデータの取り込み	9
1.7 動作確認	10
1.7.1 Designer	10
1.7.2 Viewer	11
付録：データベースの接続先追加	12

1 前提条件

OpenOLAP を実行する上で、以下の条件を満たしていることが前提条件となります。

マシン	種類	使用製品
サーバー	OS	Fedora Core 1
	データベースサーバー	PostgreSQL 7.4
	WWW サーバー	Apache 2.0
	アプリケーションサーバー	Tomcat 5.0 Tomcat Web Server Connectors
	Java 開発環境	J2SE1.4
クライアント	ブラウザ	Internet Explorer 6.0

※サーバー設定確認事項

1. OpenOLAP でグラフを表示するための設定 (JFreeChart の設定)

Tomcat 実行環境上で、グラフ用 Gif ファイルを作成するために、`$CATALINA_HOME/bin/catalina.sh` ファイルの最初に以下の記述を追加してください。

```
export CATALINA_OPTS="-Djava.awt.headless=true"
```

2. Tomcat4.x から Tomcat5.x へ変更したことに伴う設定追加

一部の日本語が文字化けすることを避けるため、`server.xml` の (使用している Port の) Connector 要素に以下の記述を追加してください。

```
useBodyEncodingForURI="true"
```

例)

```
<Connector port="8009"
           enableLookups="false" redirectPort="8443" debug="0"
           protocol="AJP/1.3" useBodyEncodingForURI="true"/>
```

※詳しくは以下を参照してください。

<http://jakarta.apache.org/tomcat/tomcat-5.0-doc/config/http.html>

2 インストール方法

1.1 インストール方法

インストールは次の順序に行います。

1. OpenOLAP_V1.2.0.zip を任意のディレクトリに解凍します。
2. 「Meta Tool」「Designer」「Viewer」のそれぞれインストールを行います。
3. 必要であればサンプルデータを取り込みます。

1.2 MetaTool

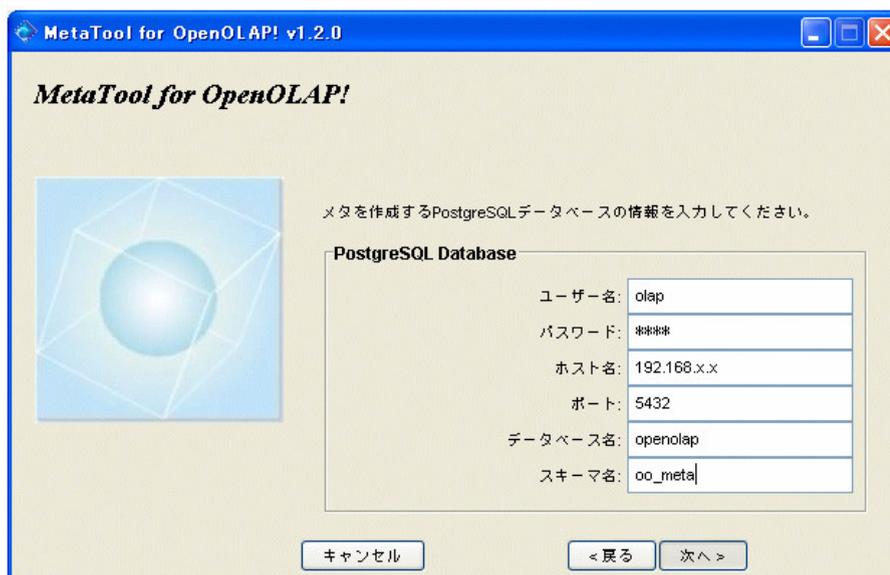
1. OpenOlap¥exe¥metaTool ディレクトリ内の4ファイルを任意ディレクトリ内にコピーします。
 - MetaTool.jar
 - createTimeData.oo
 - createProcedure.oo
 - createMetaTable.oo
2. PostgreSQLにOpenOLAP用のユーザー、データベース、スキーマを作成します（ここではユーザー「olap」、データベース「openolap」、スキーマ「oo_meta」と仮定）。

```
createuser olap
createdb -O olap -E EUC_JP openolap
createlang plpgsql openolap
psql -U olap openolap
create schema oo_meta;
```

3. MetaTool.jar をダブルクリックして Meta Tool を起動させます。



4. [次へ] ボタンをクリックします。
5. 以下の情報を入力し、[次へ] ボタンをクリックします。



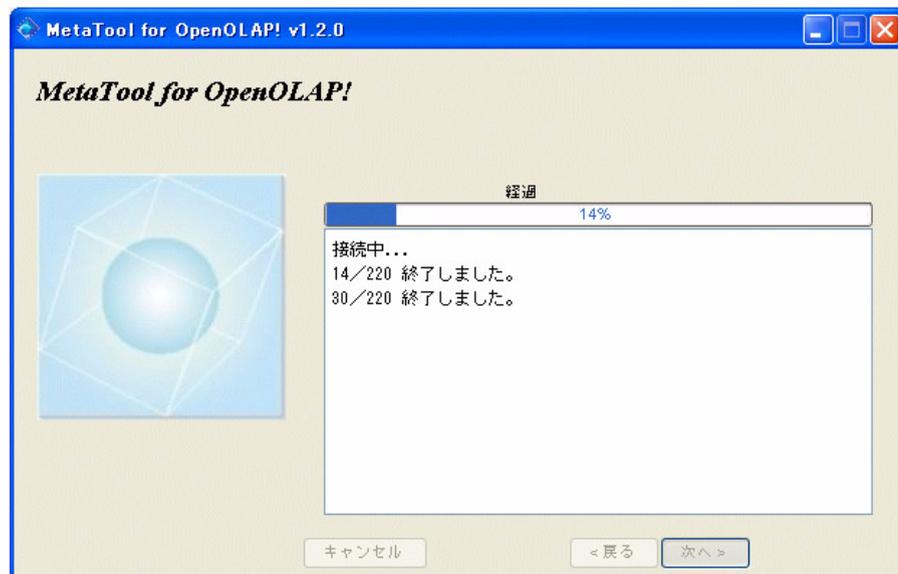
項目名	内容
ユーザー名	PostgreSQL ユーザー
パスワード	PostgreSQL ユーザーのパスワード
ホスト名	PostgreSQL が動作しているホストマシンの名前または IP アドレス
ポート	PostgreSQL のポート番号 (デフォルト「5432」)

項目名	内容
データベース名	PostgreSQL のデータベース名
スキーマ名	PostgreSQL のスキーマ名 (ここでは、ユーザー名「olap」、パスワード「olap」、ホスト名「192.168.x.x」、ポート「5432」、データベース名「openolap」、スキーマ名「oo_meta」を入力する)

6. 入力情報を確認後、[次へ] ボタンをクリックします。



7. メタ情報の作成処理が開始されます。



8. 処理が終了したら [次へ] ボタンをクリックします。



9. [閉じる] ボタンをクリックします。



1.3 Designer / Viewer 共通設定

Postgres JDBC ドライバーをインストールします。

1. OpenOlap¥src¥Viewer¥postgresql742.jar を \$CATALINA_HOME/common/lib に配置します。

1.4 Designer

1. OpenOlap¥exe¥Designer¥designer.war を \$CATALINA_HOME/webapps/ に配置します。
2. \$APACHE2_HOME/conf/workers2.properties を編集します。
(下記設定情報を追加する)

```
[uri:/designer/*]
info=Openolap designer Application.
context=/designerdebug=0
```

3. \$CATALINA_HOME/webapps/designer/login.ini を編集し、環境設定を行います。

[Environment]

DESIGNER_CLASSES_PATH	\$CATALINA_HOME/webapps/designer/WEB-INF/classes の絶対パス
JDBC_DRIVER	PostgreSQL JDBC ドライバの絶対パス

[Connect]※コネクト情報は複数記述可

LIST_NAME	任意接続名 (ログイン画面のリスト表示に使用)
USER_NAME	Meta Tool で指定したユーザー (olap)
SCHEMA_NAME	Meta Tool で指定したスキーマ (oo_meta)
CONNECT_SOURCE	jdbc:postgresql://ホスト名:ポート番号/接続先データベース名 (openolap)

設定例

[Environment]

```
DESIGNER_CLASSES_PATH=/usr/local/tomcat/webapps/
designer/WEB-INF/classes
JDBC_DRIVER=/usr/local/tomcat/webapps/designer/WEB-
INF/classes/majolica/postgresql742.jar
```

[Connect 1]

```
LIST_NAME=メタ1
USER_NAME=olap
SCHEMA_NAME=oo_meta
CONNECT_SOURCE=jdbc:postgresql://192.168.X.XX:5432/
openolap
```

1.5 Viewer

- viewer.war を \$CATALINA_HOME/webapps/ に配置します。
- \$APACHE2_HOME/conf/workers2.properties を編集します。
(下記設定情報を追加する)

```
[uri:/viewer/*]
```

```
info= Openolap viewer Application.
context=/viewer
debug=0
```

- \$CATALINA_HOME/conf/server.xml を編集し、TOMCAT のコネクションプーリングの設定を行います。

url	jdbc:postgresql:// ホスト:ポート番号/接続先データベース名 (openolap)
username	Meta Tool で指定したユーザー (olap)
password	Meta Tool で指定したパスワード (olap)

設定例

```
url:jdbc:postgresql://192.168.X.XX:5432/openolap
username:olap
password:olap
```

例) < Host name="localhost" > タグ内に下記を記述する

```
<DefaultContext>
  <Resource name="jdbc/WroxTC41" auth="Container"
  type="javax.sql.DataSource" />
  <ResourceParams name="jdbc/WroxTC41">
    <parameter>
      <name>driverClassName</name>
      <value>org.postgresql.Driver</value>
    </parameter>
    <parameter>
      <name>url</name>
      <value>jdbc:postgresql://192.168.X.XX:5432/openolap</value>
    </parameter>
    <parameter>
      <name>username</name>
      <value>olap</value>
    </parameter>
    <parameter>
      <name>password</name>
      <value>olap</value>
    </parameter>
    <parameter>
      <name>maxActive</name>
      <value>10</value>
    </parameter>
    <parameter>
      <name>maxIdle</name>
      <value>5</value>
    </parameter>
    <parameter>
      <name>maxWait</name>
      <value>100</value>
    </parameter>
  </ResourceParams>
</DefaultContext>
```

4. \$CATALINA_HOME/webapps/viewer/WEB-INF/classes/openolap/viewer/common/ viewer.properties を編集し、環境設定を行います。

[DB 接続情報]

PostgresDAOFactory.meta	Meta Tool で指定したスキーマ (oo_meta)
-------------------------	-------------------------------

設定例

PostgresDAOFactory.meta= oo_meta

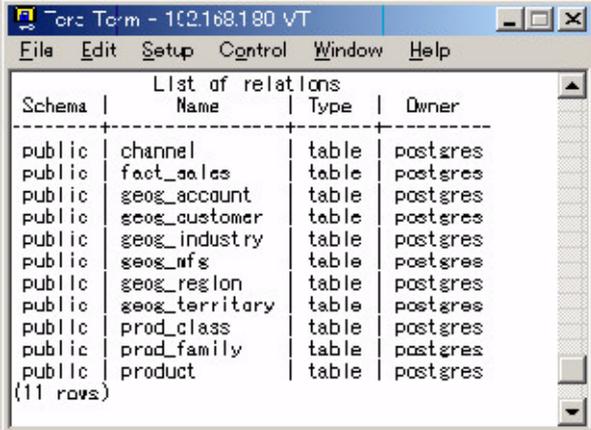
1.6 サンプルデータの取り込み

OpenOLAP マニュアルで使用されているサンプルデータを取り込むには、以下の手順でデータをインポートします。

(データベース名を openlap と仮定する)

1. OpenOLAP¥sample_data¥mart.dmp を任意のディレクトリに配置します。
2. 上記ディレクトリで、PostgreSQL の下記コマンドを発行します。

```
pg_restore -U olap -d openlap -O mart.dmp
```
3. psql openlap で PostgreSQL に接続し、¥d コマンドを発行して以下 11 テーブルが作成されていることを確認します。



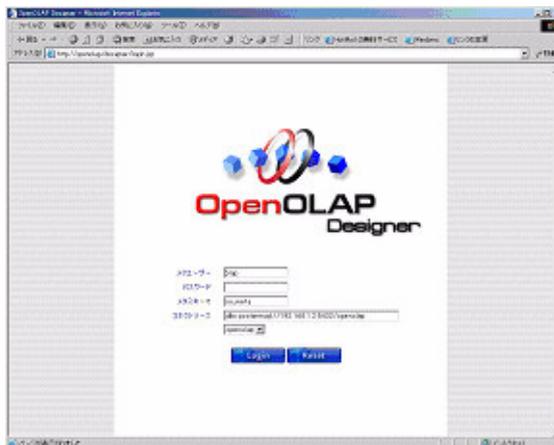
```
Term - 102.168.180 VT
File Edit Setup Control Window Help
List of relations
Schema | Name | Type | Owner
-----+-----+-----+-----
public | channel | table | postgres
public | fact_sales | table | postgres
public | geog_account | table | postgres
public | geog_customer | table | postgres
public | geog_industry | table | postgres
public | geog_mfg | table | postgres
public | geog_region | table | postgres
public | geog_territory | table | postgres
public | prod_class | table | postgres
public | prod_family | table | postgres
public | product | table | postgres
(11 rows)
```

1.7 動作確認

1.7.1 Designer

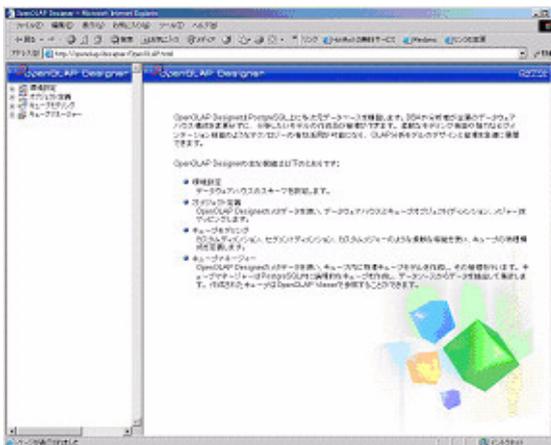
1. ブラウザを立ち上げ、下記のアドレスにアクセスし、ログイン画面が表示されることを確認します。

http://<ホスト名>/designer/login.jsp



2. ログイン画面でユーザー名、パスワード、メタスキーマ、接続ソースを入力後、ログインボタンを押下し、ログインができることを確認します。

項目	入力内容
ユーザー名	Meta Tool で指定したユーザー (olap)
パスワード	Meta Tool で指定したパスワード (olap)
メタスキーマ	Meta Tool で指定したスキーマ (oo_meta)
接続ソース	jdbc:postgresql://ホスト名:ポート番号/接続先データベース名 (openolap)



1.7.2 Viewer

1. ブラウザを立ち上げ、下記のアドレスにアクセスし、ログイン画面が表示されることを確認します。

`http://<ホスト名>/viewer/login.jsp`



2. ログイン画面でデフォルトのユーザー名、パスワードを入力後、ログインボタンを押下し、ログインができることを確認します。

〈一般ユーザー〉

ユーザー名 : user

パスワード : user

〈ゲストユーザー〉

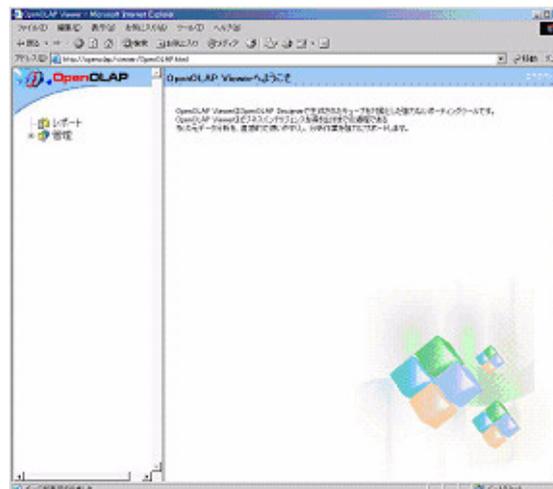
ユーザー名 : guest

パスワード : guest

〈管理者ユーザー〉

ユーザー名 : admin

パスワード : admin



付録：データベースの接続先追加

Postgres 上に複数のデータベースがあり、1つの OpenOLAP Viewer でそれぞれ接続したい場合、ログイン画面に接続するデータベースを切り替えるリストボックスを表示させることができます。

1. login.ini の編集

\$CATALINA_HOME/webapps/viewer/login.ini の以下の項目を編集します。

値	説明
USE_LIST	データベースを切り替えるかどうかを設定します。 [true] ログイン画面に [接続先] リストボックスが表示されます。 [false] データベースの切り替えはできません (デフォルト設定)。 server.xml、web.xml、viewer.properties で設定したデータベースへ接続します。
LIST_NAME	[接続先] リストボックスで表示する接続先名を設定します。
SCHEMA_NAME	接続先のスキーマを設定します。
POOL_NAME	接続先のデータベース名を設定します。

【設定例】

```
[Common]
USE_LIST=true

[Connect1]
LIST_NAME= 開発環境
SCHEMA_NAME=dev
POOL_NAME=jdbc/WroxTC41

[Connect2]
LIST_NAME= 本番環境
SCHEMA_NAME=oo_meta
POOL_NAME=jdbc/RelDB
```

2. server.xml の編集

<ResourceParams> タグの属性に login.ini で指定した POOL_NAME を指定します。

【設定例】

```
<DefaultContext>
  <Resource name="jdbc/WroxTC41" auth="Container"
  type="javax.sql.DataSource"/>
  <ResourceParams name="jdbc/WroxTC41">
    <parameter>
      <name>driverClassName</name>
      <value>org.postgresql.Driver</value>
    </parameter>
  </ResourceParams>
</DefaultContext>
```

1つ目の接続先

```
        <name>url</name>
        <value>jdbc:postgresql://192.168.X.XX:5432/dev</value>
    </parameter>
    <parameter>
        <name>username</name>
        <value>dev</value>
    </parameter>
    <parameter>
        <name>password</name>
        <value>dev</value>
    </parameter>
    <parameter>
        <name>maxActive</name>
        <value>50</value>
    </parameter>
    <parameter>
        <name>maxIdle</name>
        <value>20</value>
    </parameter>
    <parameter>
        <name>maxWait</name>
        <value>20000</value>
    </parameter>
</ResourceParams>
<Resource name="jdbc/RelDB" auth="Container"
type="javax.sql.DataSource"/>
<ResourceParams name="jdbc/RelDB">
    <parameter>
        <name>driverClassName</name>
        <value>org.postgresql.Driver</value>
    </parameter>
    <parameter>
        <name>url</name>
        <value>jdbc:postgresql://192.168.X.XX:5432/openolap</
value>
    </parameter>
    <parameter>
        <name>username</name>
        <value>olap</value>
    </parameter>
    <parameter>
        <name>password</name>
        <value>olap</value>
    </parameter>
    <parameter>
        <name>maxActive</name>
        <value>50</value>
    </parameter>
    <parameter>
        <name>maxIdle</name>
        <value>20</value>
    </parameter>
    <parameter>
        <name>maxWait</name>
        <value>20000</value>
    </parameter>
</ResourceParams>
</DefaultContext>
```

2つ目の接続先

3. web.xml の編集

<res-ref-name> に login.ini で指定した POOL_NAME を指定します。

【設定例】

```
<resource-ref>  
  <res-ref-name>jdbc/WroxTC41</res-ref-name>  
  <res-type>javax.sql.DataSource</res-type>  
  <res-auth>Container</res-auth>  
</resource-ref>  
<resource-ref>  
  <res-ref-name>jdbc/RelDB</res-ref-name>  
  <res-type>javax.sql.DataSource</res-type>  
  <res-auth>Container</res-auth>  
</resource-ref>
```

1つ目の接続先

2つ目の接続先