

# インストールマニュアル

## RDBS

発行日 2025年3月31日  
公立大学法人会津大学  
株式会社東日本計算センター

# 目次

1. 本書について .....	1
1.1. インストール環境 .....	1
2. 導入準備 .....	2
2.1. オプション .....	3
2.2. bash 実行時のエラー .....	3
3. Java の導入 .....	4
4. Apache2 および Tomcat の導入 .....	4
4.1. Apache2 のインストール .....	4
4.2. Apache2 の構成設定 .....	5
4.3. Tomcat のインストール .....	6
5. Apache と Tomcat の連携 .....	7
5.1. 仮想ホスト設定ファイル .....	7
5.2. アプリケーションのデータ格納パス設定 .....	8
6. MongoDB の導入 .....	9
6.1. インストール .....	9
7. PostgreSQL の導入 .....	10
7.1. インストール .....	10
7.2. PostgreSQL ユーザーのパスワード設定 .....	10
8. DB・テーブル作成 .....	11
8.1. MongoDB の DB と collection の作成 .....	11
8.2. PostgreSQL のロール設定、DB、テーブルの作成 .....	11
9. WEB アプリのデプロイ .....	12
9.1. デプロイの実行 .....	12

# 1. 本書について

本書はロボットデータベースシステム(RDBS)の動作環境の構築手順を記載します。  
本システムの操作に関しては「ユーザーズマニュアル\_RDBS」を参照してください。

## 1.1. インストール環境

本書では以下の環境を想定しています。

表 1-1 動作環境一覧

環境	バージョン	補足	
OS	Ubuntu	20.04 LTS	-
CPU	Core i7 相当以上	-	-
メモリ	8GB 以上推奨	-	-
ストレージ	100GB 以上推奨	-	-
開発言語	Java	1.8.0_191	-
ミドルウェア	Apache	2.4 系	Apache License 2.0
	Tomcat	9.0 系	Apache License 2.0
	PostgreSQL	14 系	PostgreSQL License
	MongoDB	4.4 系	Server Side Public License(SSPL)
フレームワーク	spring framework	5.3.18	Apache License 2.0
依存ライブラリ	spring-data-mongodb	3.2.6	Apache License 2.0
	OpenJDK	9.0.4	GPL License
	Lombok	1.18.8	MIT License
	apache.logging.log4j	2.17.1	Apache License 2.0
	JDBC Driver	42.2.0	BSD

## 2. 導入

各種ライブラリ等を導入します。導入するライブラリは本システムの環境構築時点となります。ライブラリの Version 変更を実施する際はライブラリ導入後にアップデートを行ってください。

各種ライブラリを導入するために RDBS\_installer.zip を任意の場所に配置し、解凍してください。解凍すると以下のファイルが展開されます。

表 2-1 RDBS\_installer ファイル構成

RDBS_installer	
└── RDBS.war	: RDBS アプリケーションパッケージファイル
└── application_deploy.sh	: アプリケーションデプロイ用 bash ファイル
└── envrionment.sh	: 環境構築用 bash ファイル
└── mongo_setting.sh	: MongoDB DB 作成用 bash ファイル
└── postgres_setting.sh	: PostgreSQL DB 作成用 bash ファイル

展開されたファイルには実行権限が付与されていない場合がありますので以下のコマンドを実行し、権限を変更してください。

```
$ cd 展開されたディレクトリのパス
$ sudo chmod 755 *
```

展開されたディレクトリ内にある”envrionment.sh”を下記のコマンドで実行することで各種ライブラリのインストールおよび設定処理を開始します。

bash 実行中はコンソールの出力内容に従ってライブラリの導入を行ってください。

```
$ sudo ./envrionment.sh
```

## 2.1. オプション

bash ファイルを実行する際に以下のオプションを設定することができます。

表 2-2 envrionment.sh ファイルの実行オプション

オプション	説明	記載例
-p, --proxy, --prosy=	ネットワークにプロキシを使用している場合に設定するオプション	例 1)-p 192.168.0.1:8080 例 2)--proxy 192.168.0.1:8080 例 3)--proxy=192.168.0.1:8080

## 2.2. bash 実行時のエラー

bash ファイルを実行した際に以下のエラーが出力される場合があります。エラーの内容を確認し、オプションやネットワーク環境を確認してください。

表 2-3 envrionment.sh ファイルの実行エラー

エラー	説明
Undefined options	不明なオプションが設定された際に出力されるエラーです。
Undefined argument	不明な引数が設定された際に出力されるエラーです。
No internet connection found	インターネットに接続が出来なかった際に出力されるエラーです。

### 3. Java の導入

Java アプリケーションの実行に必要な JRE(Java Runtime Environment)と Java アプリケーションの開発に必要な JDK(Java Development Kit)をインストールします。

以下の内容がコンソールに出力された際に”y”を入力し、Enter を押下してください。

```
Start installing java? [y/n]:
```

Java のインストールが完了すると Java の Version が出力されます。

```
jdk-9.0.4/man/man1/unpack200.1
jdk-9.0.4/man/man1/wsgen.1
jdk-9.0.4/man/man1/wsimport.1
jdk-9.0.4/man/man1/xjc.1
jdk-9.0.4/release
update-alternatives: /usr/bin/jar (jar) を提供するために自動モードで /opt/jdk-9.0.4/bin/jar を使います
update-alternatives: /usr/bin/javac (javac) を提供するために自動モードで /opt/jdk-9.0.4/bin/javac を使います
Java installation complete
update-alternatives: /usr/bin/java (java) を提供するためにマニュアルモードで /opt/jdk-9.0.4/bin/java を使います
openjdk version "9.0.4"
OpenJDK Runtime Environment (build 9.0.4+11)
OpenJDK 64-Bit Server VM (build 9.0.4+11, mixed mode)
```

図 3-1 Java のインストール結果

### 4. Apache2 および Tomcat の導入

Apache2 および Tomcat のインストール、各種構成の設定を行います。

#### 4.1. Apache2 のインストール

以下の内容がコンソールに出力された際に”y”を入力し、Enter を押下すると Apache2 と Tomcat のインストールが実行されます。

```
Start installing apache2 and tomcat? [y/n]:
```

Apache2 のインストールが完了すると以下の内容がコンソールに出力されます。また、Tomcat のインストールは Apache2 の構成設定後に実行されます。

```
man-db (2.9.1-1) のトリガを処理しています ...
libc-bin (2.31-0ubuntu9) のトリガを処理しています ...
Apache2 installation complete
Server version: Apache/2.4.41 (Ubuntu)
Server built: 2023-01-23T18:36:09
```

図 4-1 Apache2 のインストール結果

## 4.2. Apache2 の構成設定

コンソールに以下の Apache2 の構成設定要求が出力された際に”y”を入力し、Enter を押下することでサーバーの構成設置を行います。

クライアントに不要なサーバー情報を公開しない設定(表 4-1)やディレクトリインデックスのファイル指定(表 4-2)を変更します。

```
Setup apache2 configuration? [y/n]:
```

表 4 - 1 security.conf の変更箇所

行数	変更前	変更後
25	ServerTokens OS	ServerTokens Prod
36	ServerSignature On	ServerSignature Off

表 4 - 2 dir.conf の変更箇所

行数	変更前	変更後
2	DirectoryIndex ¥ index.html index.cgi ¥ index.pl index.php ¥ index.xhtml	DirectoryIndex ¥ index.html index.php

### 4.3. Tomcat のインストール

Apache2 の構成設定後、Tomcat のインストール処理が実行されます。インストールが完了すると、コンソールにインストールした Tomcat の Version が出力されます。

```
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.64/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/apache-tomcat-9.0.64/bin/bootstrap.jar:/opt/apache-tomcat-9.0.64/bin/tomcat-juli.jar
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS: --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
Server version: Apache Tomcat/9.0.64
Server built: Jun 2 2022 19:08:46 UTC
Server number: 9.0.64.0
OS Name: Linux
OS Version: 5.15.0-67-generic
Architecture: amd64
JVM Version: 9.0.4+11
JVM Vendor: Oracle Corporation
Check the Tomcat installation result by visiting the URL: http://localhost:8080
```

図 4-2 Tomcat のインストール結果

また、Web ブラウザで”http://localhost:8080/”にアクセスすると Tomcat がインストールされていることを確認出来ます。

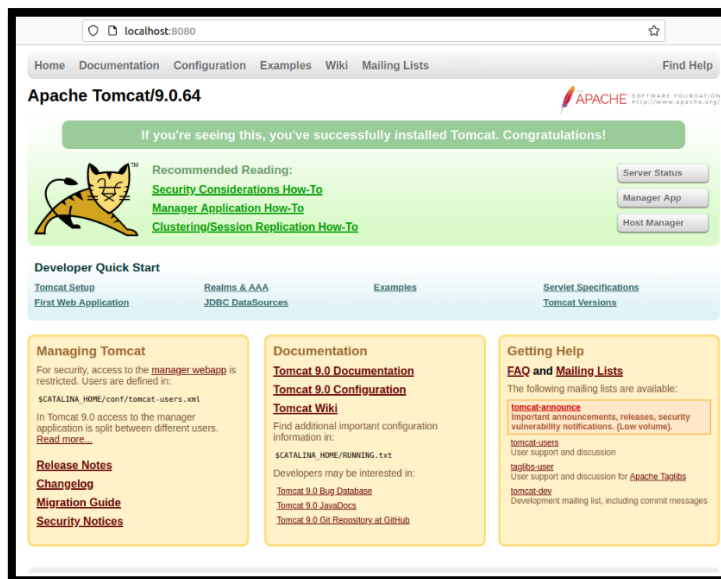


図 4-3 Tomcat 接続画面



## 5. Apache と Tomcat の連携

現状の設定では Apache と Tomcat が独立して動作するため、リクエストを Apache で待ち受けて Tomcat に渡すように設定を変更します。

コンソールに以下の連携設定要求が出力された際に"y"を入力し、Enter を押下することで連携の設定を開始します。

```
Setup apache2 and tomcat linkage configuration? [y/n]:
```

### 5.1. 仮想ホスト設定ファイル

仮想ホスト設定ファイルを生成します(ファイル: /etc/apache2/sites-available/uoa\_service.conf)。

ServerName と ServerAdmin の変更が必要な場合は bash 処理が終了後、記載内容を変更してください。

```
<VirtualHost *:80>
    ServerName localhost:8080
    ServerAdmin admin@uoa.com
    <Location />
        ProxyPass ajp://localhost:8009/
        Order allow,deny
        Allow from all
    </Location>
    ErrorLog /var/log/apache2/error.log
    # Possible values include: debug, info, notice, warn, error, crit,
    # alert, emerg.
    LogLevel warn
    CustomLog /var/log/apache2/access.log combined
</VirtualHost>
```

図 5 - 1 uoa\_service.conf の内容

## 5.2. アプリケーションのデータ格納パス設定

Apache と Tomcat の連携設定時に RDBS のデータ格納パス設定およびディレクトリの生成が実行されます。

```
<!-- Access log processes all example.
Documentation at: /docs/config/valve.html
Note: The pattern used is equivalent to using pattern="common" -->
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log" suffix=".txt"
pattern="%h %l %u %t &quot;%r&quot; %s %b" />
<Context path='/data' docBase='/var/database/uoa/data' />
</Host>
</Engine>
</Service>
</Server>
```

図 5 - 2 server.xml の変更内容

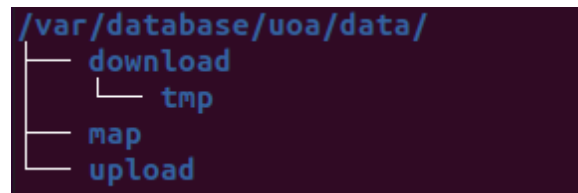


図 5 - 3 RDBS のデータ格納ディレクトリ構成

## 6. MongoDB の導入

### 6.1. インストール

以下の内容がコンソールに出力された際に”y”を入力し、Enter を押下すると MongoDB のインストールが実行されます。

```
Start installing MongoDB? [y/n]:
```

コンソールに以下の内容が出力された際に”y”を入力し、Enter を押下すると MongoDB のインストールが実行されます。

インストール完了すると MongoDB の Version を固定するかどうか要求されます。固定する場合は”y”、固定しない場合は”n”を入力し Enter を押下してください。

```
Hold MongoDB version? [y/n]:
```

Version の固定設定が完了すると MongoDB の Version が出力されます。

```
MongoDB shell version v4.4.15
Build Info: {
  "version": "4.4.15",
  "gitVersion": "bc17cf2c788c5dda2801a090ea79da5ff7d5fac9",
  "opensslVersion": "OpenSSL 1.1.1f 31 Mar 2020",
  "modules": [],
  "allocator": "tcmalloc",
  "environment": {
    "distmod": "ubuntu1804",
    "distarch": "x86_64",
    "target_arch": "x86_64"
  }
}
Mongodb installation complete
```

図 6-1 MongoDB のインストール結果

## 7. PostgreSQL の導入

### 7.1. インストール

コンソールに以下の内容が出力された際に”y”を入力し、Enter を押下すると MongoDB のインストールが実行されます。

```
Start installing PostgreSQL? [y/n]:
```

以下の内容がコンソールに出力された際に”y”を入力し、Enter を押下すると PostgreSQL のインストールが実行されます。

インストール完了すると PostgreSQL の Version が出力されます。

```
Executing: /lib/systemd/systemd-sysv-install enable postgresql  
psql (PostgreSQL) 14.7 (Ubuntu 14.7-1.pgdg20.04+1)  
Postgresql installation complete
```

図 7-1 PostgreSQL のインストール結果

### 7.2. PostgreSQL ユーザーのパスワード設定

PostgreSQL のインストールが完了すると postgres ユーザー(OS ユーザー)のパスワード設定を実行します。コンソールに下記の内容が出力されたらパスワードを入力し、設定を行ってください。

```
Set password for PostgreSQL admin user  
admin name : password  
postgres :
```

## 8. DB・テーブル作成

### 8.1. MongoDB の DB と collection の作成

RDBS で使用する DB と collection の作成を行います。”mongodb\_setting.sh”を以下のコマンドで実行してください。

```
$ sudo ./mongo_setting.sh
```

### 8.2. PostgreSQL のロール設定、DB、テーブルの作成

postgres ロール(PostgreSQL ユーザー)のパスワードを設定および RDBS で使用するデータベースとテーブルの作成を行います。”postgres\_setting.sh”を以下のコマンドで実行してください。

- (1) postgres\_setting.sh を実行してください。

```
$ sudo ./postgres_setting.sh
```

- (2) bash ファイルを実行すると postgres ロール(PostgreSQL ユーザー)のパスワード設定を要求されます。パスワードの設定を行う場合は”y”を入力し、Enter を押し、パスワードを入力してください。

```
Start setting up a PostgreSQL admin role? [y/n]: y
Set password for PostgreSQL admin role
role name : password
postgres :
```

- (3) パスワードの設定が完了すると DB とテーブルを作成します。DB が作成されるとコンソールに DB の状況が出力されます。

```
ALTER ROLE
Set a password for the PostgreSQL admin role
PostgreSQL: DB and table create start
          テータベース一覧
名前 | 所有者 | エンコーディング | 照合順序 | Ctype(変換演算子) | アクセス権限
-----+-----+-----+-----+-----+-----
postgres | postgres | UTF8 | ja_JP.UTF-8 | ja_JP.UTF-8 | =c/postgres +
template0 | postgres | UTF8 | ja_JP.UTF-8 | ja_JP.UTF-8 | postgres=CTc/postgres +
template1 | postgres | UTF8 | ja_JP.UTF-8 | ja_JP.UTF-8 | =c/postgres +
          postgres=CTc/postgres
uoa_2nd_db | uoa_guest | UTF8 | ja_JP.UTF-8 | ja_JP.UTF-8 |
(4 行)

ALTER DATABASE
CREATE TABLE
ALTER TABLE
PostgreSQL: DB and table create completed
```

図 8-1 PostgreSQL の DB、テーブルの作成結果

## 9. WEB アプリのデプロイ

### 9.1. デプロイの実行

WEB アプリケーションのデプロイを開始するには”application\_deploy.sh”を以下のコマンドを実行してください。bash ファイルを実行する際はデプロイする RDBS.war ファイルを”application\_deploy.sh”と同じディレクトリに配置するか、RDBS.war ファイルのパスを引数に指定し、bash ファイルを実行してください。

```
$ sudo ./application_deploy.sh
```

```
$ sudo ./application_deploy.sh /home/uoa/src/RDBS.war
```

bash ファイルを実行すると以下の Log がコンソール上に出力され WEB アプリケーションのデプロイが完了します。

```
robot-db@robotdb-VirtualBox:~/src$ sudo ./application_deploy.sh
RDBS deploy start
RDBS deploy end
robot-db@robotdb-VirtualBox:~/src$
```

図 9-1 WEB アプリケーションのデプロイ実行結果

## 著作権

本文書の著作権は公立大学法人 会津大学に帰属します。