

ANCL

ShareTask 5.5

(doc#01)

ジョブ仲介サーバー
インストールマニュアル

アンクル
2014 年 2 月

目次

1	概要	5
2	インストール	6
2.1	インストール概要	6
2.2	インストールメディアのディレクトリ構成	7
2.3	外部パッケージのインストール	7
2.4	その他の注意事項	8
2.5	インストール手順	8
2.5.1	インストール先の決定	8
2.5.2	プログラムのインストール	8
2.5.3	PostgreSQL データベースの作成	9
2.5.4	PostgreSQL の設定	9
2.5.5	Apache の設定	11
2.5.6	Memcached の設定	12
3	ShareTask の設定	13
3.1	システム設定	13
3.1.1	データベース環境の設定 conf/db.conf	13
3.1.2	システム環境の設定 conf/system.conf	13
3.1.3	conf/mime.conf	17
3.1.4	conf/index.html	17
3.1.5	conf/progs.conf	18
3.1.6	ライセンス・カスタマー設定	18
3.1.7	運用設定データベースの初期化	18
3.2	ユーザー登録	18
3.3	ユーザーに管理者権限を与える	19
3.4	同時実行ジョブ数ライセンス	20
4	運用	21
4.1	運用概要	21
4.2	運用詳細	21
4.2.1	PostgreSQL の運用	21
4.3	バックアップをとる	21

4.4 エージェントを制御する	22
4.4.1 エージェントのジョブポーリングを開始する	22
4.4.2 エージェントのジョブポーリングを停止する	22
4.4.3 エージェントの設定を修正する	22

図目次

3.2.1 ユーザー登録画面	19
3.2.2 登録内容確認	19
3.3.1 管理者リスト	19

1 概要

ShareTask は、Web ベースのジョブスケジューラーです。

ユーザーからジョブの登録を受け付けるジョブ仲介サーバーと、ジョブ仲介サーバーから自動的にジョブを取り込み計算する計算エージェントとから構成されます。

ジョブ仲介サーバーは、Web サーバーであり、ジョブキューならびにジョブの蓄積場所（レポジトリ）としてシンプルな機能を果たします。

ユーザーは、ジョブ仲介サーバーと HTTP で通信することにより、ジョブのサブミット、ジョブの状態と計算結果の閲覧、ダウンロードを行うことができます。

本文書は、ジョブ仲介サーバーのインストールと運用設定について説明します。

2 インストール

2.1 インストール概要

このドキュメントは CentOS5.5 へのインストールを前提として記述されていますが、後述のソフトウェア環境が整えられる OS であれば、その他のベンダーによる Linux ディストリビューション、BSD 系 OS、または商用 Unix 環境での動作を制限する要素は特にありません。ただし、動作確認対象外となります。

ShareTask 本体以外のソフトウェアは付属 CD-ROM に含まれておりませんので、別途配布元や epel, rpmforge 等のサードパーティの RPM レポジトリから入手する必要があります。

ShareTask に必要な外部ソフトウェアは以下のものです。
(それぞれが依存しているものは含まれておりません)

- Apache (2.0 以上)
- PostgreSQL (8.1 以上, 8.4 を推奨)
- Memcached (1.4 以上)
- Perl 5.8.8 以上 (5.10 は動作未確認)
- Perl モジュール
 - Jcode
 - Cache::Memcached::Fast
 - Date::Calc
 - DBI
 - DBD::Pg
 - Authen::PAM
 - Net::LDAP
 - Net::NIS
 - Crypt::RSA
 - IO::Socket::SSL
 - JSON
 - JSON::XS
- Tomcat5 (複数で稼動する場合のみ、通常は必要ありません)

2.2 インストールメディアのディレクトリ構成

インストールメディアのディレクトリ構成は次のとおりです。

external

client: Java 関連
server: RPM 関連

sharetask

client: エージェント、コマンドライン関連
doc: ドキュメント
server: ジョブ仲介サーバー関連

2.3 外部パッケージのインストール

必要な外部パッケージ (RPM) については、インストールメディアの

`external/server/rpms_20110901.tar.gz`

に、RedHat EL 5.6 x86_64 を前提として必要となる RPM ファイルを用意してありますので、参考にしてください。

また、その RPM のインストール作業を支援する Makefile を

`external/server/Makefile.RHEL5_6_x86_64`

として用意しましたので、こちらも参考にしてください。

RedHat EL 5.6 x86_64 であれば、これらを組み合わせて、必要なパッケージをインストールできると思いますが、環境によっては、すでにインストールされていたり、異なるバージョンのパッケージがインストールされている場合がありますので、個々のパッケージのインストール状態に応じて RPM インストール操作を行っていただき、前節 2.1 で列挙しましたパッケージ群がインストールされている状態を実現してください。

ShareTas は、PostgreSQL サーバーと通信するためのパッケージとして、

`postgresql-libs-8.1.x`

での動作実績があります。

`postgresql-libs-8.4.x`

では、動作しないことがありますので、必ず、`postgresql-libs-8.1.x` がインストールされていることを確認してください。ちなみに、

`postgresql-libs-8.1.x`

`postgresql-libs-8.4.x`

この 2 つは共存できます。

2.4 その他の注意事項

SE Linux が有効化されていると、動作が阻害されることがあります。必要に応じて SE Linux の無効化を検討してください。

iptables によるファイアウォール機能が、PostgreSQL との通信を阻害することがあります。iptables の設定を確認してください。

2.5 インストール手順

2.5.1 インストール先の決定

まず、ShareTask のインストール先を決定します。

任意の場所で構いませんが、ここでは `"/home/sharetask "` へのインストールを前提として説明します。

2.5.2 プログラムのインストール

1. インストール先のディレクトリと作業用のディレクトリを作成します。

```
# mkdir /home/sharetask
# mkdir /home/sharetask/tmp
```

2. 付属 CD-ROM 内の tar で圧縮されたプログラムファイルを作業用のディレクトリにコピーし、/home/sharetask へ展開します。

- sharetask-{version}.tar.gz サーバ本体
- sharetask-{version}-agent_conf.tar.gz エージェント設定ファイル

```
# cp {CD-ROM}/sharetask-{version}.tar.gz /home/sharetask/tmp
# cp {CD-ROM}/sharetask-{version}-agent_conf.tar.gz /home/sharetask/tmp
# cd /home/sharetask/tmp
# tar xzvf sharetask-{version}.tar.gz -C /home/sharetask
# tar xzvf sharetask-{version}-agent_conf.tar.gz -C /home/sharetask
```

3. /home/sharetask へ移動し、必要なディレクトリなどを作成します。

展開したサーバプログラムにシンボリックリンクを設定します。

```
# ln -s sharetask-{version} sharetask_server
```


データベース領域用, 計算ファイル用のディレクトリを作成します.

```
# mkdir sharetask_database
# mkdir sharetask_files
# mkdir -p sharetask_configs/agent_conf
```

ディレクトリのオーナーを適切なユーザーに設定します.

```
* httpd(apache)
# chown -R apache: sharetask-{version}
# chown -R apache: sharetask_configs
# chown -R apache: sharetask_files
* PostgreSQL(postgres)
# chown -R postgres: sharetask_database
```

2.5.3 PostgreSQL データベースの作成

1. データベース領域を初期化します.

```
# sudo -u postgres initdb -E EUC-JP --no-locale
/home/sharetask/sharetask_database/data
```

2. 作成したデータベース領域を PostgreSQL の起動スクリプトに指定します.

```
# echo " PGDATA=/home/sharetask/sharetask_database/data " >
/etc/sysconfig/pgsql/postgresql
```

3. PostgreSQL を起動します.

```
# /etc/init.d/postgresql start
```

4. ShareTask データベースを作成します.

```
# cd /home/sharetask/sharetask_server/sql/
# sh CREATE_DATABASE.sh (質問にはすべて"n"と入力する)
# sh CREATE_TABLES.sh (作成結果が出力されます)
```

2.5.4 PostgreSQL の設定

(詳しくは PostgreSQL のマニュアルを参照してください)

1. データベースのパスワード保護

初期状態ではパスワードは設定されていません。パスワードは、psql コマンドで管理者として接続し、SQL を入力して設定します。

```
$ psql -U postgres
psql> ALTER USER postgres WITH PASSWORD 'your_password '
psql> ALTER USER stask_mpi WITH PASSWORD 'your_password '
```

注意：

プロンプトは、実際とは異なるかもしれません。

パスワード文字列は、シングルクォート ' で括ってください。

2. PostgreSQL、ホタト

```
/home/sharetask/sharetask_database/data/pg_hba.conf
```

(データベース接続に対する認証方式を設定します。)

- 誰でも認証なしに使用できるようにするには、認証方式を"trust" に設定します。
- パスワードで接続を保護する場合には、認証方式を"password"または"md5" に変更します。
- 変更後は PostgreSQL の再起動が必要です。

```
/home/sharetask/sharetask_database/data/postgresql.conf
```

(リソースの割り当てや基本動作について設定します)

- マシンリソース等の状況に合わせて、適切なリソースを割り当てます。
- 任意のアドレスで TCP 接続を有効にしてください。
- AUTO VACUUM を有効にするか、別途、cron 等で vacuumdb コマンドを定期的に行うようにしてください。
- 変更後は PostgreSQL の再起動が必要です。

3. OS ブート時の自動起動設定

chkconfig コマンドで自動起動設定を行ってください。

```
# /sbin/chkconfig postgresql on
```

2.5.5 Apache の設定

(詳しくは Apache のマニュアルを参照してください)

1. conf ディレクトリにある httpd-sample.conf を/etc/httpd/conf へ sharetask.conf としてコピーします。

```
# cd /home/sharetask/sharetask_server/conf/
```

```
# cp httpd-sample.conf /etc/httpd/conf/sharetask.conf
```

2. コピーしたファイルを httpd.conf から include するように設定します (httpd.conf は事前にバックアップを取ってください)。

```
# echo "include conf/sharetask.conf" >> /etc/httpd/conf/httpd.conf
```

3. httpd を起動します.

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

4. OS ブート時の自動起動設定

chkconfig コマンドで自動起動設定を行ってください.

```
# /sbin/chkconfig httpd on
```

- 文字化けを起こす場合は, /etc/httpd/conf/httpd.conf のディレクティブを変更してみてください.

```
AddDefaultCharset Off
```

- 起動時の設定は, /etc/sysconfig/httpd で設定します.

例) SSL を有効にする

```
HTTPDOPT="--DSSL"
```

2.5.6 Memcached の設定

(詳しくは Memcached のマニュアルを参照してください)

1. 使用可能なメモリ量を設定します (メモリ 1G 以上を推奨します).

```
/etc/sysconfig/memcached
```

```
CACHESIZE="1G"
```

2. memcached を起動します.

```
# /etc/init.d/memcached start
```

3. OS ブート時の自動起動設定

chkconfig コマンドで自動起動設定を行ってください.

```
# /sbin/chkconfig memcached on
```

3 ShareTask の設定

3.1 システム設定

運用開始前に設定しなければならない項目です。設定ファイルの基本ディレクトリは、`/home/sharetask/sharetask_server/conf/` です。

設定ファイルは基本的に、`key = value` の形式で書かれており、行頭`#`、空行は無視されます。指定のファイルがない場合は、新たに同名のテキストファイルを作成してください (基本ディレクトリ内に `.example` の拡張子のついた雛形が用意されています)。

3.1.1 データベース環境の設定 `conf/db.conf`

```
#データベースホスト
dbhost = localhost
#データベースポート
dbport = 5432
#データベース名
dbname = stask_mpi
#ユーザ名
dbuser = stask_mpi
#パスワード
dbpass = test123
```

3.1.2 システム環境の設定 `conf/system.conf`

基本的な動作モードは、`system.conf` に記述される各種パラメーターで選択します。`system.conf.example` を参考に `system.conf` を作成します。

以下、各パラメーターについて説明します。

DISPAT

エージェントが起動するアプリケーションプログラムを束ねるラッパースクリプト (ディスパッチャーと呼んでいる) の名称を指定します。ディスパッチャーは、与えられた引数から起動すべきアプリケーションを決定し、その実行ファイルを起動するように記述します。ディスパッチャーを使用することによって、エージェント設定ファイルの `define_program` セクションを集約して記述できるので、記述量を大幅に減らせます。

例：

```
define_program {  
    name = dispat  
    executable = /usr/local/sharetask/appl/dispat.sh
```

このプログラムを指定した場合、従来のプログラム名の代わりに、ディスパッチャーの第 1 引数をアプリケーション識別子とみなし、ジョブのリソース使用量集計などにおいてキーとして扱います。

CACHE

待ち時間予測の算出を高速化するために memcached を使用するかどうかを選択します。

PROGS_SELECTOR

ジョブ登録画面で、プログラム名とキューをプルダウンメニュー選択方式に切り替えられます。値 1 でプルダウンメニュー選択方式を指定、値 0 で従来の任意文字列タイプイン方式を指定します。プルダウンメニューの選択肢は、`conf/progs.conf` に記述します。

NO_UPLOAD

このパラメーターの値を 1 に設定すると、ジョブ登録画面の入力ファイル指定が、実行ノード側のファイルシステムのパスを指定するとみなします。つまり、ローカルファイルのアップロードはしないことになります。値 0 では、入力ファイルはブラウザが動いているローカル上のパスであるとみなし、アップロードが行われます。

JOBLIST_SHOW_ALL

一般ユーザが開くジョブリスト画面に、全ユーザのジョブを表示するかどうかを指定します。値 1 で全ユーザのジョブを表示、値 0 で当該ログインユーザのジョブのみを表示します。

AGENT_AUTO_REG

エージェントからサーバーへの初回アクセス時に、サーバーが当該エージェントを自動的に登録し認証情報を発行するかどうかを指定します。値 1 では、新しく起動したエージェントは自動的にサーバーに登録されます。値 0 では、自動登録を許可しないので、管理者用メニューの「エージェント ID の発行」を使用して認証情報ファイルを取得し、これを手動でエージェント側の `/var/spool/sharetask/auth` ディレクトリに配置する必要があります。

AUTH

ユーザー認証方式を選択します。指定できる値は、次の 3 種類です。

sharetask : ShareTask が内蔵するユーザーデータベース
nis : NIS サーバーと連携します
ldap : LDAP サーバーと連携します

MIN_UID

MAX_UID

NIS, LDAP の場合に、ShareTask ユーザとして認識するユーザ ID の最小値と最大値を指定します。この範囲外のユーザ ID のユーザは ShareTask ユーザとして見なされず、ShareTask にログインできません。

NIS.DOMAIN

上記 AUTH で NIS を指定した場合に有効なパラメーターで、NIS のドメイン名を指定します。

NIS_MAP

上記 AUTH で NIS を指定した場合に有効なパラメーターで、NIS のユーザ認証のマップ名を指定します。一般的には、passwd.byname です。

以下は、上記 AUTH で LDAP を指定した場合に有効なパラメーターです。

LDAP_TYPE

LDAP サーバーに管理者権限で問い合わせるかどうかを指定します。

LDAP_HOST

LDAP サーバーのホスト名あるいは IP アドレスを指定します。

LDAP_SSL

LDAP_TLS

LDAP サーバーとの通信の暗号化方式を選択します。

LDAP_SASL

SASL(Simple Authentication and Security Layer) を使用するかどうかを指定します。

LDAP_ATTR_ID

ユーザー ID が納められている属性名を指定します。

LDAP_ATTR_PASSWORD

パスワード文字列が納められている属性名を指定します。

以下は、上記 LDAP_TYPE=admin で、管理者権限での問い合わせを指定した場合に有効なパラメーターです。

LDAP_ADMIN_DN

管理者のドメイン情報 (cn, dc) を指定します。

例: LDAP_ADMIN_DN = cn=Manager,dc=anclab,dc=com

LDAP_ADMIN_PASSWORD

管理者のパスワードを指定します。

LDAP_BASE_DN

ベースとなるドメインを指定します。

例: LDAP_BASE_DN = dc=anclab,dc=com

LDAP_FILTER

ShareTask ユーザーとして許可する範囲を抽出する絞り込み条件を指定します (書式については RFC 2254 を参照)。

例: LDAP_FILTER = (&(|(1=Tokyo)(1=Osaka)))

3.1.3 conf/mime.conf

ジョブ詳細からファイルを閲覧するときの content-type を、標準と違うものに設定します。

例:

```
# .csv ファイルをテキストとしてブラウザで表示させます
csv = text/plain
```

3.1.4 conf/index.html

トップページの右インラインフレーム部分をカスタマイズできます。

HTML で記述します。

注意: <HTML><HEAD><BODY>タグはつけないでください。

3.1.5 conf/progs.conf

system.conf の PROGS_SELECTOR を有効にした場合に必要になるプログラム、キュー、並列度の組み合わせを記述するファイルです。この記述にしたがって、ジョブ登録画面のプログラムとキューの指定のプルダウンメニューが構成されます。書式については、あらかじめ存在するひな形ファイル内のコメントと記述例を参考にしてください。

3.1.6 ライセンス・カスタマー設定

sharetask.user ファイルにあらかじめ設定したライセンス ID を書きます。その後、弊社より発行されたライセンスファイル (.lic .sig のファイル) を license ディレクトリへコピーしてください。

ライセンス ID は、弊社担当までお問い合わせください。

インストールメディアに同梱されている評価用ライセンス

```
{CD-ROM}/sharetask/server/TRIAL_License_20110811.tar.gz
```

を使用する場合は、ライセンス ID は TRIAL です。

3.1.7 運用設定データベースの初期化

運用設定データベース (WebUI から変更します) にデフォルト値をセットします。

```
# cd /home/sharetask/sharetask_server/tools
# perl reset_config.pl
```

3.2 ユーザー登録

ユーザーの登録方法について説明します。この操作には、管理者権限をもつユーザーアカウントが必要です。

ShareTask サーバーの We 画面に、管理者権限をもつユーザーでログインします (初期ユーザーは 'sharetask@anclab.com' パスワードは 'sharetask1234' ログインできない場合は「2.5.3 PostgreSQL データベースの作成」をご確認ください)。

システム管理 (管理者のみ) というセクションにある

ユーザー登録

という項目をクリックしてください。

フォームの各項目を入力し、送信ボタンをクリックしてください。

登録内容を確認してから登録ボタンをクリックしてください。

これで登録完了です。

ハンドル名	<input type="text"/>
メールアドレス	<input type="text"/>
メールアドレス(確認)	<input type="text"/>
パスワード(8文字以上)	<input type="password"/>
パスワード(確認)	<input type="password"/>

図 3.2.1 ユーザー登録画面

ハンドル名	saito2
メールアドレス	saito2@anclab.com
パスワード(8文字以上)	*****

図 3.2.2 登録内容確認

3.3 ユーザーに管理者権限を与える

初期状態では、追加された全ユーザーに管理者権限が与えられます。制限が必要な場合は、以下の手順に従って、特定のユーザーに管理者権限を設定してください。

システム管理 (管理者のみ) というセクションにある

データ編集メニュー > ユーザー

でユーザー登録情報の一覧を表示し、管理者権限を与えたいユーザーの ID(ユーザー ID) を調べます。

次に、システム管理 (管理者のみ) というセクションにある

システム設定

項目をクリックしてください。

管理者 UID の項目に、管理者権限を与えたいユーザー ID をスペース区切りで列挙します。ここに”0”を指定すると、全ユーザーが管理権限を持ちます。

更新	27 システムのファンクティ	local0
更新	28 管理者UID(スペースで区切って複数列挙可能)	2
更新	29 エージェント設定ファイルディレクトリ	/var/www/html/stash/public/html/admin/agent/conf.d

図 3.3.1 管理者リスト

更新ボタンをクリックしてください。
以上で、設定完了です。

3.4 同時実行ジョブ数ライセンス

弊社より発行されたライセンスファイル (.lic .sig のファイル) を license ディレクトリへコピーしてください。

4 運用

4.1 運用概要

運用の前提とする OS は、CentOS5 です。

必要に応じ、Linux, Apache, PostgreSQL のドキュメントを参照してください。

4.2 運用詳細

4.2.1 PostgreSQL の運用

データベースのパフォーマンスを維持するため、定期的な VACUUM と ANALYZE の実行が強く推奨されます。AUTOVACUUM を設定しない場合は、postgres ユーザーで定期的に次のコマンドを実行してください。

```
$ vacuumdb -a -z
```

または、cron に設定します。

```
# 毎日午前 5:00 に VACUUM と ANALYZE を自動実行
0 5 * * * postgres /usr/bin/vacuumdb -a -z
```

使用状況により、VACUUM が追いつかない場合があります。

エージェントが大量の場合は、VACUUM の回数を増やしてください。

4.3 バックアップをとる

予期せぬ障害に備えて、PostgreSQL データベース、ファイルリポジトリは cron 等を利用し、定期的にバックアップしておく心安心です。

PostgreSQL は、pg_dumpall コマンドを使用し、データベースをまるごとバックアップすると簡単です。

```
例: # pg_dumpall -U postgres | gzip > backup_database.gz
```

ファイルリポジトリは、ディレクトリごと tar で圧縮するとよいでしょう。

```
例: # tar cvzf backup_files.tar.gz /home/sharetask/sharetask_files
```

4.4 エージェントを制御する

計算ノード上のエージェントを一度起動すると、その設定変更、ジョブポーリングの開始と停止は、ジョブ仲介サーバーの Web 画面から中央制御できます。

以下の操作は、ジョブ仲介サーバーの Web 画面に管理者権限ユーザーでログインして行ないます。

Web トップ画面の監視・統計 (管理者のみ) セクションのエージェントコントロールリスト項目をクリックして、エージェントコントロールリスト画面を開いてください。

4.4.1 エージェントのジョブポーリングを開始する

start ボタンをクリックすると、コマンド欄が start に変わります。60 秒ほどで、ステータス欄が operational に変わります。

これで、エージェントがジョブのポーリングを開始しました。

4.4.2 エージェントのジョブポーリングを停止する

stop ボタンをクリックすると、コマンド欄が stop に変わります。60 秒ほどで、ステータス欄が inactive に変わります。

これで、エージェントがジョブのポーリングを停止しました。

4.4.3 エージェントの設定を修正する

表の設定ファイル欄に表示されているファイル名 *.conf をクリックします。画面上部の設定ファイルのテキストボックスにそのファイル名が埋め込まれます。その右側の new/edit ボタンをクリックすると、ポップアップウィンドウとして 編集画面が表示されます。編集して保存してください。

以上