

# Raspberry Pi 3/4 (Raspberry Pi OS) での Actian Zen v15 Edge 使用方法



株式会社エージーテック

2021年10月27日

## 免責事項

株式会社エージーテックは本書の使用を、利用者またはその会社に対して「現状のまま」でのみ許諾するものです。株式会社エージーテックは、いかなる場合にも本書に記載された内容に関するその他の一切の保証を、明示的にも黙示的にも行いません。本書の内容は予告なく変更される場合があります。

## 商標

© Copyright 2022 AG-TECH Corp. All rights reserved. 本書の全文、一部に関わりなく複製、複写、配布をすることは、前もって発行者の書面による同意がない限り禁止します。

すべての Actian のブランド名は、Actian Corporation の米国およびその他の国における登録商標または商標です。

文中の社名、商品名等は各社の商標または登録商標である場合があります。

Raspberry Pi 3/4 (Raspberry Pi OS) での Actian Zen v15 Edge 使用

最終更新：2022 年 1 月 17 日

## 目次

はじめに . . . . .	4
Raspberry Pi 3 / 4 の設定 . . . . .	5
Actian Zen Edge のインストール . . . . .	6
Actian Zen Edge の環境設定 . . . . .	6
クライアントから Control Center で接続 . . . . .	8
クライアントからのアクセス . . . . .	8
クライアントから Btrieve ファイル参照 . . . . .	8
クライアントから SQL (ODBC) で接続 . . . . .	9
クライアントの Control Center から設定変更 . . . . .	9
Windows の共有にある Btrieve ファイルにアクセス . . . . .	10
Actian Zen Edge ユーザー以外でツールを使用 (設定) . . . . .	11
Actian Zen Edge ユーザー以外でツールを使用 (実行) . . . . .	11
Btrieve API サンプルの実行 . . . . .	11
Python サンプル . . . . .	13
Python 実行用のインタフェースモジュール生成 . . . . .	13
Python サンプルの実行 . . . . .	16
Btrieve 2 の C++プログラムのビルド及び実行の手順 . . . . .	16

## はじめに

Actian Zen で新たに IoT 環境がサポートされました。

本書では、Raspberry Pi 3/4に OS として Raspberry Pi OS をインストールした環境での Actian Zen Edge 設定・使用手順をご案内いたします。

Raspberry Pi OS は、Linux のディストリビューションの一つですが、x86 環境の Linux と若干異なるところがあります。

また、x86 環境と比べリソースに制限があるため、Actian Zen Server/Workgroup の GUI ツール (Control Center 等) は含まれていません。

Raspberry Pi OS で Actian Zen Edge を使用するための設定に加え、Windows 環境から接続して、設定等を行う手順についてもご案内します。

IoT 環境でのデータベースを利用した案件で Actian Zen Edge をご利用いただければ幸いです。

なお、本書では Raspberry Pi OS は 2021 年 10 月現在最新の Raspberry Pi OS GNU/Linux 10 (Buster) を使用して解説します。

## Raspberry Pi 3 / 4 の設定

Raspberry Pi OS 環境で Actian Zen Edge を使用するには、前準備を行っておくと、より簡単に使用できます。

### 1. 準備

設定を容易にできるように、root アカウントにパスワードを設定しログオン可能にします。

```
sudo passwd root
```

Raspberry Pi OS を更新します。

```
sudo apt-get update  
sudo apt-get upgrade
```

root ユーザーに変更します。

```
sudo su
```

### 2. Samba サーバーインストール

Samba をインストールします。設定を簡単に行うには、root でログオンしてください。

```
apt-get install samba
```

samba を有効化

```
systemctl enable smbd  
systemctl start smbd
```

## Actian Zen Edge のインストール

Actian Zen Edge のインストールは root ユーザーでログオンして行います。

1. ファイルを展開します。

カレントフォルダーを変更します。

```
cd /usr/local
```

インストールパッケージを展開します。

```
tar -zxf /home/pi/Zen-Edge-Server-jajp-linux-armhf-yy.yy.zzz.zzz.tar.gz
```

※ yy.yy.zzz.zzz は、ビルド番号を表す「15.00.055.000」のような文字列です。

Zen のインストールファイルは/home/pi に置かれていると想定しています。パス名は各自の環境に読み替えてください。

2. スクリプトを実行し環境設定を行います。

カレントディレクトリをスクリプトがある /usr/local/actianzen/etc に変更します。

```
cd actianzen/etc
```

スクリプトを実行します。

```
./preinstall.sh  
./postinstall.sh
```

## Actian Zen Edge の環境設定

Demodata を外部から参照可能なように設定します。

ユーザー test を作成してアクセスが可能なようにします。

※root でログインをして設定を行います。

ユーザー test を作成します。

```
adduser test
```

※メッセージに従いパスワード等を設定してユーザーを作成してください。

パスワード以外は入力しなくても (Enter 入力のみ)、作成できます。

test ユーザーを zen-data グループに追加します。

```
gpasswd -a test zen-data
```

test ユーザーを sudo グループに追加します。

```
gpasswd -a test sudo
```

test ユーザーを samba のユーザーとして登録します。

```
smbpasswd -a test
```

Samba の設定を変更し、/usr/local/actianzen/data フォルダーを共有に設定します。

/etc/samba/smb.conf をエディターで開きます。

次の行を追加します。

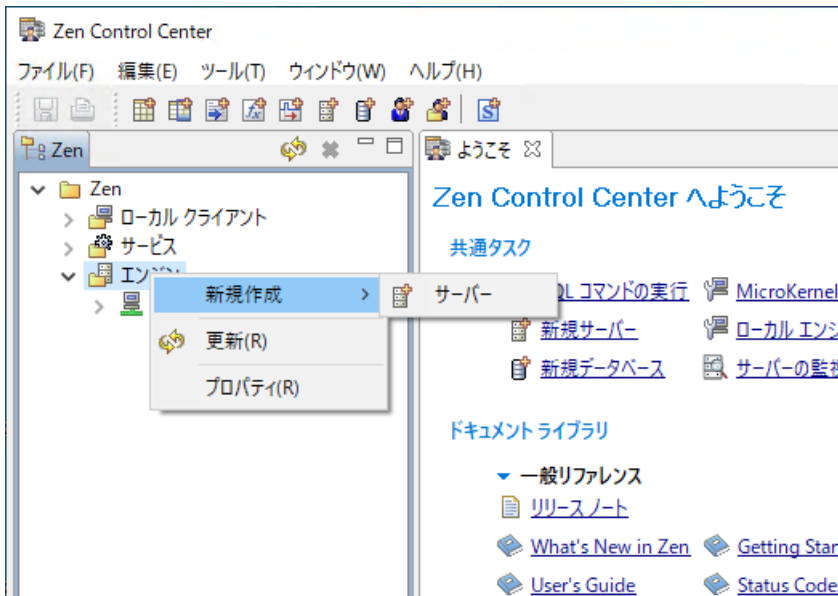
```
[actianzen]
comment = Actian zen data
path = /usr/local/actianzen/data
browseable = yes
writable = yes
```

samba サービスを再起動します。

```
systemctl restart smbd
```

## クライアントから Control Center で接続

クライアントの Control Center から、[エンジン] → [新規作成] → [サーバー] で Raspberry Pi 3 / 4 のマシンを追加します。



※DNS 等で名前解決ができるように設定していない場合、IP アドレスで指定します。

マシン名 (IP アドレス) をダブルクリックして、接続します。

## クライアントからのアクセス

Action Zen Edge 製品に対し、Windows および Linux クライアントから全ての Actian Zen インターフェイス (Btrieve API、DTI、ODBC、ADO、ADO.NET、JDBC) がサポートされています。また、Raspberry Pi OS 上の Actian Zen Edge 製品がインストールされている環境からは、Actian Zen Windows および Linux、他の Actian Zen Edge 製品がインストールされている環境に対し、Btrieve API、DTI、ODBC、JDBC がサポートされています。

## クライアントから Btrieve ファイル参照

ファイルパスに smb.conf で設定した共有名を使用して UNC 形式で指定するか、Btrieve URI 形式で接続します。

※ Btrieve URI でのアクセスでは、Samba の設定は必要ありません。



例：共有名で使用

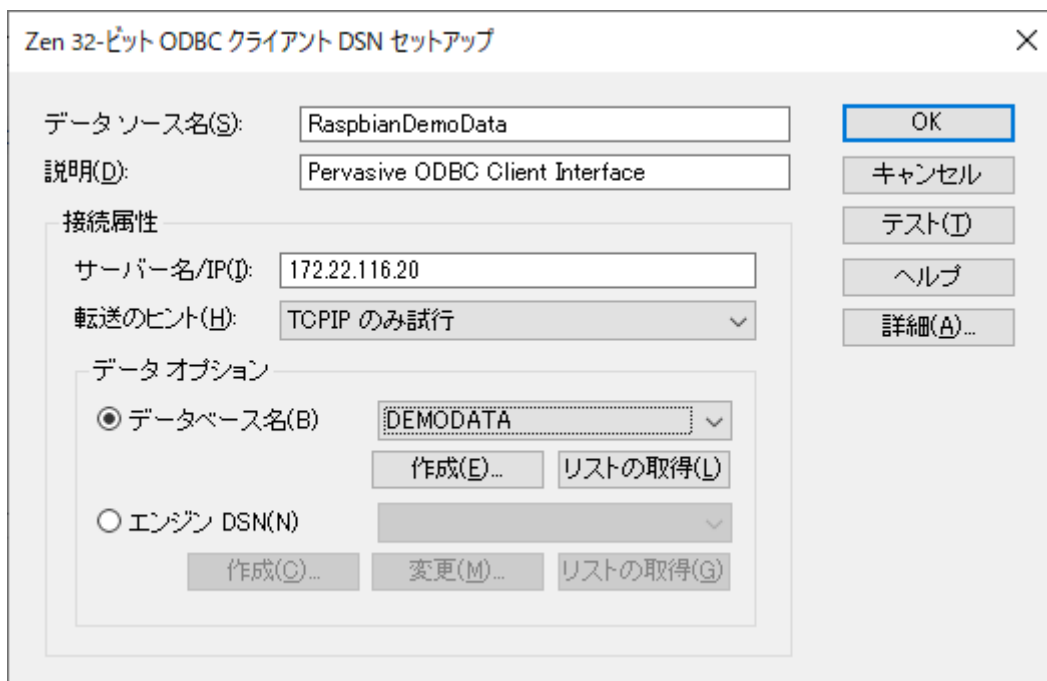
¥¥サーバ名¥actianzen¥demodata¥person.mkd

例：URI で使用

btrv://サーバ名/demodata?table=person

## クライアントから SQL (ODBC) で接続

クライアントから ODBC で接続するには、ODBC Administrator から DSN を追加します。サーバ名に Raspberry Pi OS の IP アドレスを設定し、「リストの取得」を行います。リストボックスにデータベースがリストされますので、該当のデータベースを選択します。



これで、ODBC データソースを指定して SQL アクセスが可能です。

## クライアントの Control Center から設定変更

Action Zen Edge の設定を変更するには、btadmin コマンドで管理者権限を設定しておく必要があります。

btadmin を使用するためには、ツールへのパス、ライブラリへのパスが適切に指定されている必要があります。ここでは、既に設定が行われている zen-svc ユーザーへ切り替えて設定を行います。

```
sudo su
su zen-svc
btadmin -p パスワード a+ ユーザー名
```

この設定を行うことで、Control Center からエンジンのプロパティの変更が可能となります。

## Windows の共有にある Btrieve ファイルにアクセス

Windows の共有をマウントするためのマウントポイントを作成します。

```
mkdir /winzen
```

Windows の共有をマウントします。

```
mount -t cifs //マシン名/共有名 /winzen -o username=ユーザー名
```

pvnetpass を使用し、該当サーバーで有効なユーザーとパスワードを設定しておきます。

```
pvnetpass -a マシン名 -u ユーザー名 -p パスワード
```

butil で Windows の共有にあるファイルにアクセスします。

例：

```
butil -stat /winzen/demodata/person.mkd
```

パスに Btrieve URI を使用することも可能です。

## Actian Zen Edge ユーザー以外でツールを使用（設定）

Actian Zen Edge ユーザー以外で Actian Zen Edge のツールを使用するには、パス等の設定が必要です。

/etc/profile または /home/ユーザー名/.bash\_profile に次の 4 行を追加します。

```
export ACTIANZEN_ROOT=/usr/local/actianzen
export PATH=$PATH:$ACTIANZEN_ROOT/bin:/bin:/usr/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ACTIANZEN_ROOT/lib:$ACTIANZEN_ROOT/bin:/usr/lib
export MANPATH=$MANPATH:$ACTIANZEN_ROOT/man
```

Actian Zen の ライブラリモジュールが呼び出せるよう /etc/ld.so.conf.d/ に拡張子が conf で任意の名前のファイル（例えば zenlib.conf）を作成し、次の 3 行を追加して保存後、ldconfig コマンドを実行します。

```
/usr/local/actianzen/lib
/usr/local/actianzen/bin
/usr/lib
```

```
ldconfig
```

## Actian Zen Edge ユーザー以外でツールを使用（実行）

例：

```
butil -stat /usr/local/actianzen/data/DEMODATA/Person.mkd
```

## Btrieve API サンプル の実行

C 言語サンプル (btrsamp.c) のコンパイル、実行は次の手順で行います。

1. サンプルが含まれる Btrieve API SDK のダウンロード

次の WEB ページから Btrieve API SDK をダウンロードします。

[https://www.agtech.co.jp/actian/support/reference/sdk/access\\_methods/btrieve/](https://www.agtech.co.jp/actian/support/reference/sdk/access_methods/btrieve/)

Raspbian 環境では、Linux 用の SDK を使用します。

Zen-SDK-Btrieve-linux-noarch-yy.yy.zzz.zzz.tar.gz

2. ダウンロードした SDK を展開します。

- ・ /usr/local/actianzen をカレントにし、tar コマンドで展開します。

```
cd /usr/local/actianzen
tar -zxvf /usr/local/Zen-SDK-Btrieve-linux-noarch-yy.yy.zzz.zzz.tar.gz
```

※ファイルを /usr/local にダウンロードした前提で記載しております。

次のフォルダーが作成されます。

Zen-SDK-Btrieve-000-linux-noarch-yy.yy.zzz

```
|
├ doc
|
├ intf
|
└ samples
```

便宜上フォルダー名を変更します。

```
mv Zen-SDK-Btrieve-000-linux-noarch-yy.yy.zzz SDK
```

3. サンプルをコンパイルします。

サンプルのフォルダーに移動し、make all を実行します。  
詳細は README ファイルをご参照下さい。

```
cd /usr/local/actianzen/SDK/samples/btrieve  
make all
```

4. サンプルを実行します。

```
./btrsamp
```

## Python サンプル

Python サンプルは、SWIG で生成したインタフェースモジュールを使用し、Btrieve2 API を呼び出します。

サンプルを実行する前に、SWIG でインタフェースモジュールの生成が必要です。

## Python 実行用のインタフェースモジュール生成

1. Btrieve2 API SDK をダウンロード

次の WEB ページから Btrieve2 API SDK をダウンロードします。

[https://www.agtech.co.jp/actian/support/reference/sdk/access\\_methods/btrieve/](https://www.agtech.co.jp/actian/support/reference/sdk/access_methods/btrieve/)

Raspberry Pi OS 環境では、Linux 用の SDK を使用します。

Zen-SDK-Btrieve2API-linux-noarch-yy.yy.zzz.zzz.tar.gz

※この手順では、ファイルを/usr/local にダウンロードしています。

2. 下記のデモプログラム用の SDK フォルダーがない場合は、以下のコマンドで作成して下さい。

```
mkdir /usr/local/actianzen/SDK
```

- ダウンロードした SDK を展開します。

/usr/local/actianzen/SDK をカレントにし、tar コマンドで展開します。

```
cd /usr/local/actianzen/SDK
tar -zxf /usr/local/Zen-SDK-Btrieve2API-linux-noarch-yy.yy.zzz.zzz.tar.gz
```

※ファイルを /usr/local にダウンロードした前提で記載しております。

次のフォルダーが作成されます。

```
Zen-SDK-Btrieve2API-linux-noarch-yy.yy.zzz.zzz
|
├ doc
|
├ intf
|
└ samples
```

便宜上フォルダー名を変更します。

```
mv Zen-SDK-Btrieve2API-linux-noarch-yy.yy.zzz.zzz Btrieve2API
```

- SWIG および Python がまだインストールされていない場合は、インストールします。

```
apt-get install swig
apt-get install python3 python3-dev
```

- Btrieve 2 SDK が入っているディレクトリをカレントに変更します。

```
cd Btrieve2API
```

6. btrievePython.cpp を生成します。

```
swig -c++ -c++ -include -o btrievePython.cpp swig/btrievePython.swig
```

7. Btrieve 2 の Python 用インタフェースモジュール (btrievePython.py と \_btrievePython.so) をビルドします。

```
g++ -c -fPIC -include -I/usr/include/python3.7 btrievePython.cpp  
g++ -shared btrievePython.o -o _btrievePython.so -L/usr/local/actianzen/lib -lbtrieveC++
```

上記で使用されている Python へのパスは異なる場合があることに注意してください。

8. デモプログラム用のフォルダーを作成。

```
mkdir /usr/local/zendemo
```

9. /usr/local/zendemo フォルダーのパーミッションで「内容の変更」に「すべて」を設定します。

```
chmod 777 /usr/local/zendemo
```

10. Btrieve 2 SDK に含まれている Python サンプルと必要なファイルをそのフォルダー内にコピー。

```
cd /usr/local/zendemo  
cp /usr/local/actianzen/SDK/Btrieve2API/samples/btest.py .  
cp /usr/local/actianzen/SDK/Btrieve2API/btrievePython.py .  
cp /usr/local/actianzen/SDK/Btrieve2API/_btrievePython.so .
```

11. 必要なロードライブラリを見つけるために、zen-svc のユーザーに変更する、または LD\_LIBRARY\_PATH に /usr/local/actianzen/lib を追加します。

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/actianzen/lib
```

12. Python テストプログラムを実行します。

```
python3 btest.py 9
```

出力 "record: (9, 81, 3.0)" が表示されます。

ノート：このテストプログラムの 12 行目 (import btrievePython) でこの手順で作成された Btrieve 2 API インタフェースモジュールを読み込んでいます。

## Python サンプルの実行

次の WEB ページの「サンプルプログラム」のところから `pythondemo.tar.gz` をダウンロードして展開します。

[https://www.agtech.co.jp/actian/support/reference/sdk/access\\_methods/btrieve/#sample](https://www.agtech.co.jp/actian/support/reference/sdk/access_methods/btrieve/#sample)

```
tar -zxf pythondemo.tar.gz
```

展開したファイルを `/usr/local/zendemo` フォルダにコピーします。

```
cp /usr/local/zendemo/pythondemo/*.* .
```

次のコマンドでサンプルを実行します。

```
python3 linux_python_sample.py
```

## Btrieve 2 の C++プログラムのビルド及び実行の手順

Btrieve 2 の C++のプログラムは下記の手順でビルド及び実行できます。いくつかの Btrieve 2 の C++サンプルプログラムは Actian Zen の Btrieve 2 API 説明書の「各種例」のところ

([https://www.agtech.co.jp/products/actian/docs\\_portal/Zen/btrieve2v15/html/examples.html](https://www.agtech.co.jp/products/actian/docs_portal/Zen/btrieve2v15/html/examples.html)) にあります。



この手順は上記の「Python サンプル」の手順が使用した/usr/local/zendemo フォルダを再利用します。

1. [https://www.agtech.co.jp/products/actian/docs\\_portal/Zen/btrieve2v15/html/examples.html](https://www.agtech.co.jp/products/actian/docs_portal/Zen/btrieve2v15/html/examples.html) から btest.cpp サンプル C++ プログラムをダウンロードし、テキストファイル (btest.cpp) として /usr/local/zendemo に保存します。

```
cd /usr/local/zendemo
cp /usr/local/actianzen/SDK/Btrieve2API/include/btrieveC*.h .
```

2. 下記のコマンドでビルドします。

```
g++ -Wall -o "btest" "btest.cpp" -L/usr/local/actianzen/lib -lbtrieveCpp -Wl,-rpath-link /usr/local/actianzen/lib
```

3. テストプログラムを実行します。

```
./btest 9
```

出力 "record: (9, 81, 3.000000)" が表示されます。