

OSC 福岡
2025

ftpex説明資料

2025.11.22

エイチスリーパートナーズ有限公司

Table of Contents

1. 自己紹介
2. ftpexの概要
3. ftpexの使い方
4. ftpexの構造
5. bashのコマンド内コマンド対話処理について
6. ftpexの入手先
7. クロージング

1. はじめに

- ▶ 北の大地のITバイト、tmrhtngwです
- ▶ ftp, sftp, scpの汎用転送ツールを作ってみました
- ▶ bashで[コマンド内コマンド (サブコマンド)] の対話処理のハンドリングに苦労したのでその話をしてみたいと思います。

* やっぱりちょっと恥ずかしいので少し書き換えました

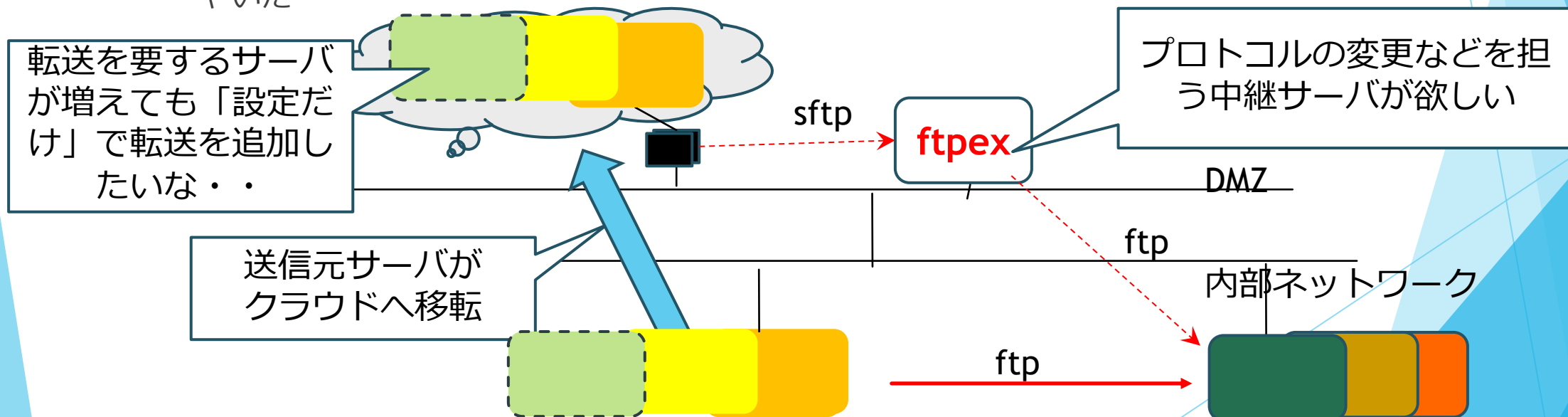


2. ftpex概要

2.1 開発のきっかけ

▶ キッカケ

- ▶ 某お客さんで、オンプレミスのサーバー間でのftpファイル転送ジョブがあった
- ▶ 一方をクラウドにリフトするがその際、送信側がSftpにすることになった
- ▶ 受信側は変えたくないのと、ネットワーク経路が変わる事に伴うセキュリティの都合で、中継し、かつ転送の追加は設定だけで済むもの欲しいなあ・・・と担当者がつぶやいた



* 講演時のQ&Aの結果、加筆した部分があります（転送対象サーバの追加について）

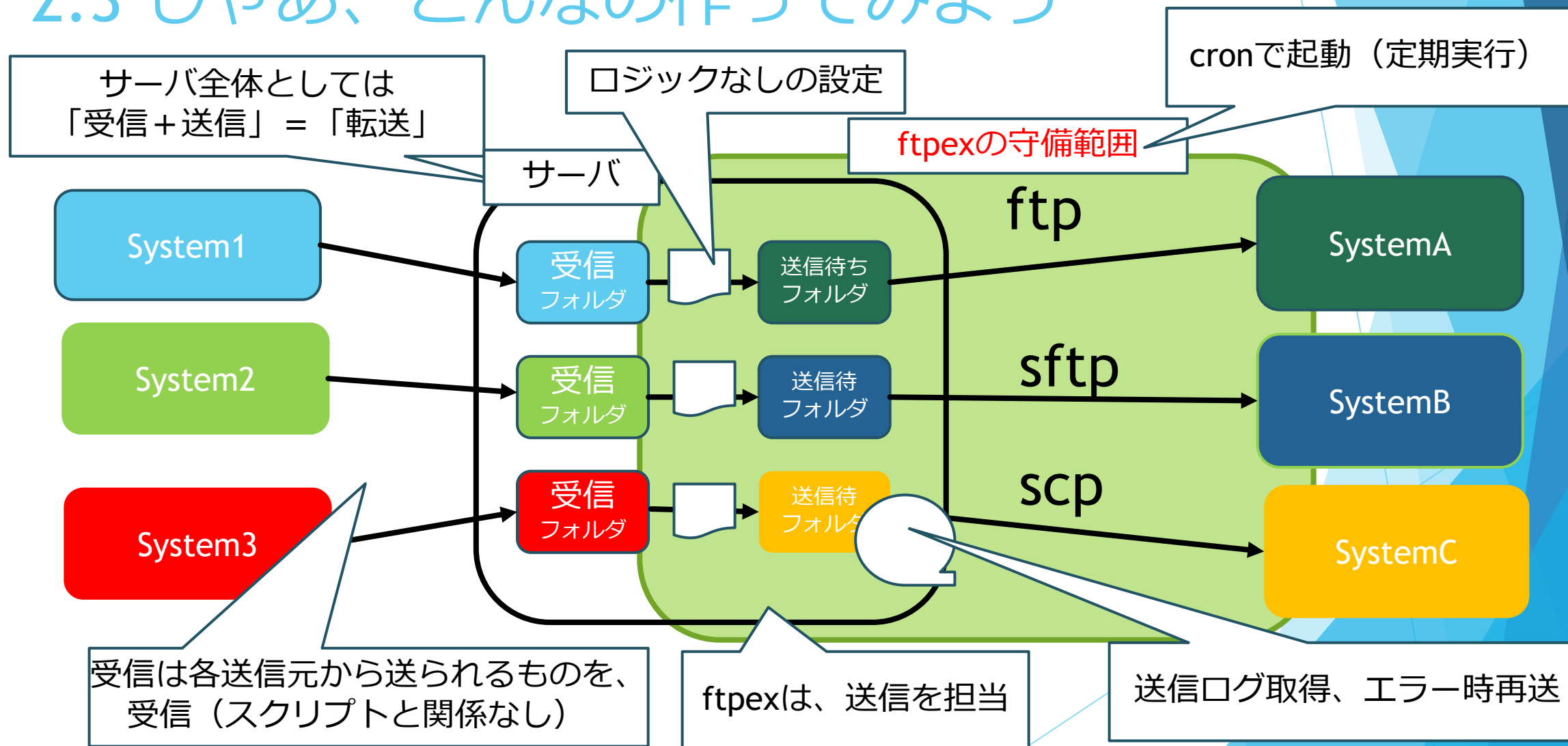
2. ftpex概要

2.2 要望機能

- ▶ 期待された機能
 - ▶ シンプルにsftpで受信したものをftpで送信しなおす（できればsftp, scpにも対応）
 - ▶ 送信のログを残すこと
 - ▶ 失敗の際に自動でリトライすること
 - ▶ 送信・受信の組合せを「ロジックを書かずに設定だけで」増やせ、設定値（コード）で管理できること（**laC**: Infrastructure as Code)

2. ftpex概要

2.3 じゃあ、こんなの作ってみよう



3. ftpexの使い方

3.1 ロジック不要、設定ファイル書くだけ

ファイル例: /home/ftpex/config/sys01.cfg

"sys01"

#System for: Test System01

MODE="ftp" #"ftp" or "sftp" or "scp"

SUB="sys01"

USER="ftpuser01"

PSWD="password"

DEST="192.168.0.3"

SUBDIR="receive"

#SUBDIR="."

RCV="/chroot/sys01/receive_dir"

FTPMODE="on" #only for FTP. Passive="on", Active="off"

PORT="22" #Default: ftp 21, sftp, scp 22

IDENTITYFILE="/home/ftpex/.ssh/id_ed25519" #identity file path if needed

送信受信の1ペアごとに設定ファイルを書く

ftp, sftp, scpを選択

送信先サーバ。IPアドレスかホスト名

SUBDIRは、転送先サーバでログインしたディレクトリからcdでカレントを移動する先。異動しない場合は"."(ピリオド)

RCVは、ftpexのサーバ（転送用サーバ）でファイルを受け取る受信ディレクトリ

4. ftpexの構造

4.1 技術仕様

▶ 動作環境

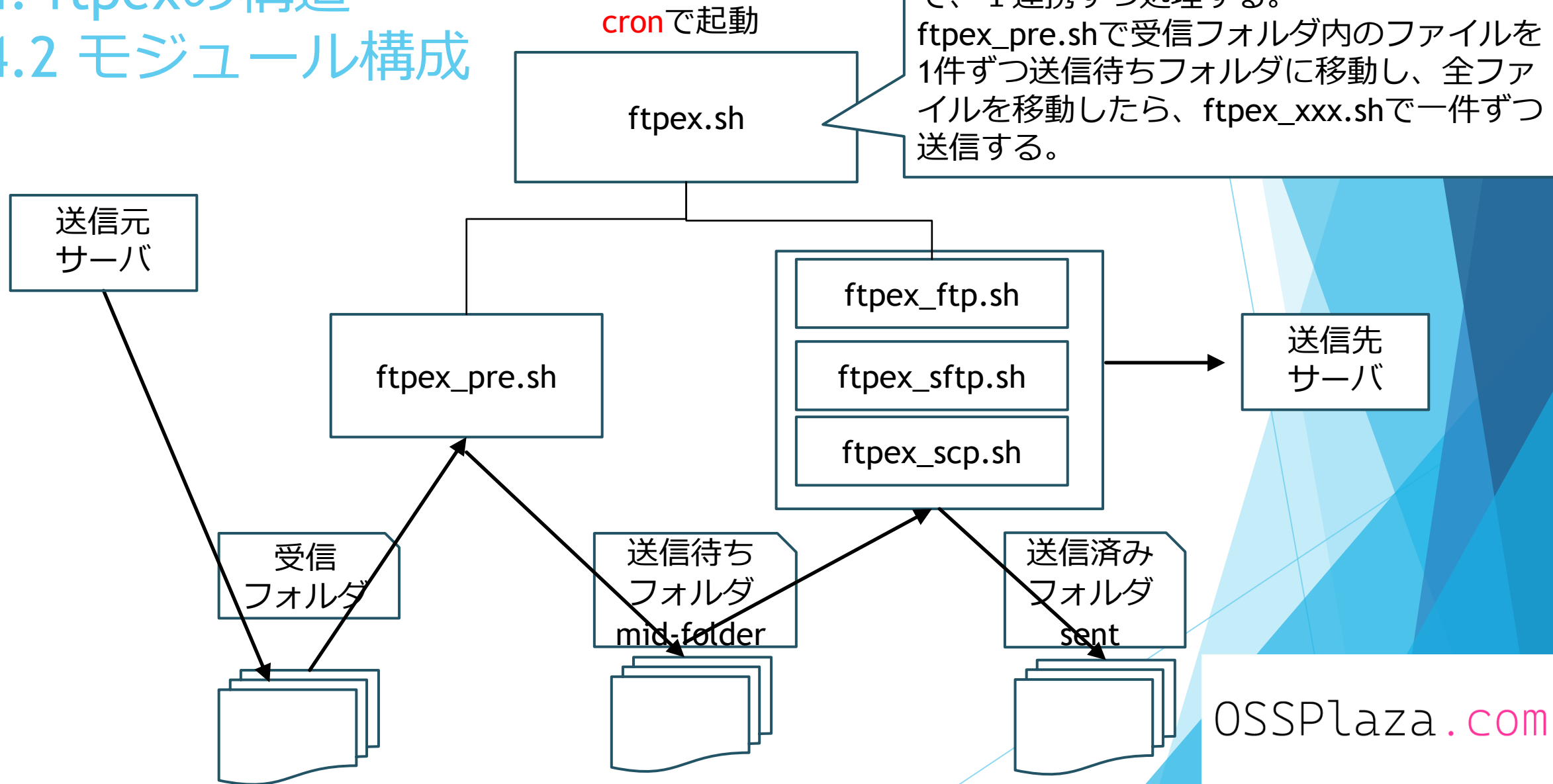
- ▶ OS: Ubuntu 24.04 Serverにて確認
 - ▶ (依存はないと思うので、他のLinux(RHELなど)でも大丈夫なはず)
 - ▶ ー> Ubuntu Server はGUIがないので、容量を食わない、という理由で選定
- ▶ 言語 : bash, expect
- ▶ 一定時間ごとにcron等で起動する

▶ 重要な留意点

- ▶ 受信はftpexではなにもせず、サーバの他の機能(ftp, sftp, scp, 他)で受信。ftpexはそれを取得する。(こちらから取りに行くタイプの受信をする機能はない)
- ▶ ディレクトリそのまま転送はできない。転送は特定ディレクトリにあるファイル(複数可)で、1階層の想定。
- ▶ ファイル名はユニークである必要がある。前に受信したファイルと同名ファイルを受信した場合、転送待ちや転送先にある同名ファイルを上書きする。

4. ftpexの構造

4.2 モジュール構成

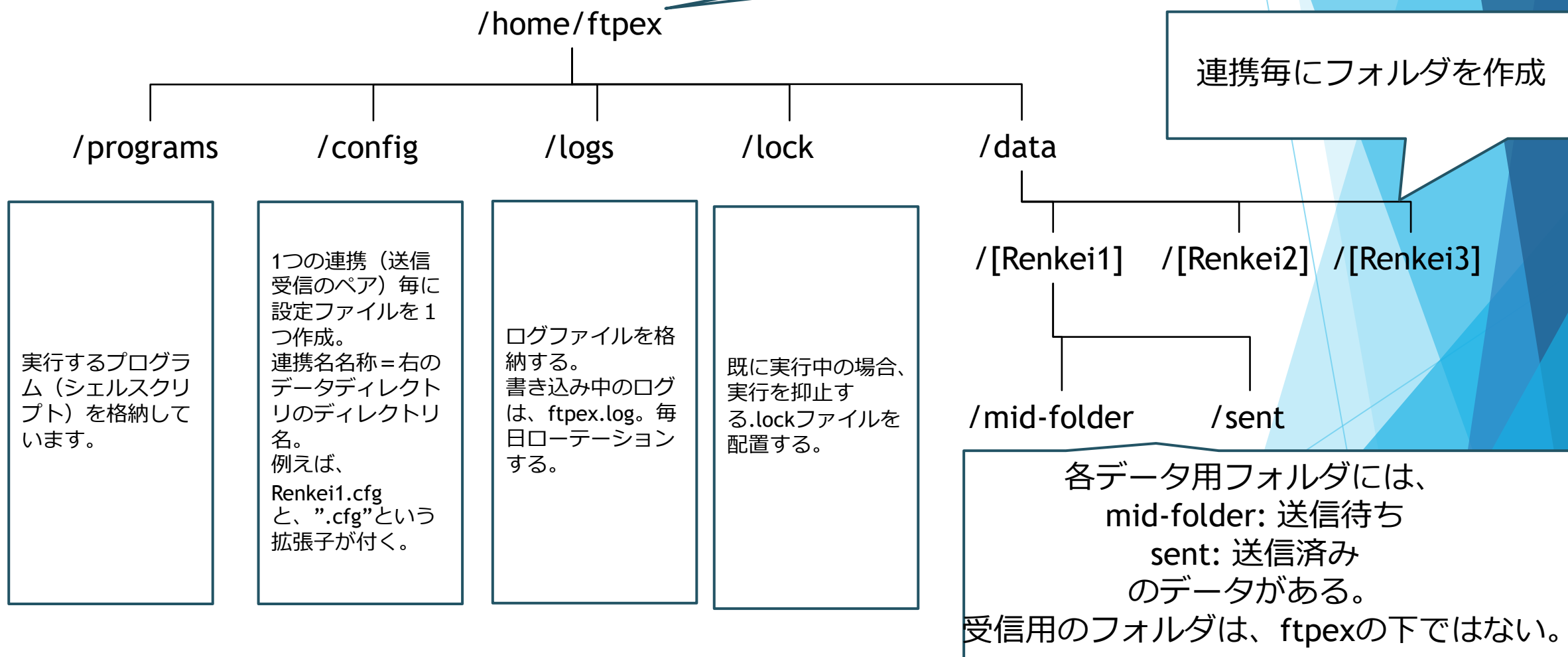


4. ftpexの構造

4.3 ディレクトリ構成

ftpexというユーザを作る
そのホームディレクトリ
\$HOME で変更可

連携毎にフォルダを作成



5. bashのコマンド内コマンド対話処理

5.1 コマンド内コマンドの対話処理

- ▶ 一般のコマンドと異なり、ftpは「コマンドの中でコマンド(サブコマンド) を実行」する。
ftpの中でopenやcd, putを命令する
- ▶ そのため、一般的なbashの対話処理では対応できず「一工夫」必要。

```
$ftp
```

```
ftp> open 192.168.0.3
```

```
ftp> user user01
```

```
ftp> password [password]
```

```
ftp> binary
```

```
ftp> put xxxx.file
```

```
ftp> close
```

```
ftp> quit
```

```
$
```

ftpコマンドを打った後は、quitで終了するまでは”ftpの中”。この中のコマンド内コマンド（サブコマンド）はbashで直接は扱えない。

5. bashのコマンド内コマンド対話処理

5.2 ヒアドキュメント

対策としてまず「ヒアドキュメント」が考えられる。

=> 比較的単純なフローの場合には使えるが、今回は厳しいと判断

```
ftp -v -p -n <<- EOF > $TMP 2>&1
```

```
open $DEST
```

```
user $USER $PSWD
```

```
binary
```

```
cd $SUBDIR
```

```
put $line
```

```
quit
```

```
EOF
```

二つのEOFの間に実行したい
コマンドを書いていきます。

ftpコマンドの中の処理を2行目以降に記述
結果を、\$TMPで定義したファイルに出力

ftpコマンドのレスポンスが中間ファイルに
あるので、それを拾ってbashで処理できる

5. bashのコマンド内コマンド対話処理

5.3 expectとは

- ▶ bashから直接操作しにくい「コマンドの中で」対話処理を行いたい時に使う
 - ▶ 例えば、sshのパスフレーズ入力とか
 - ▶ 返ってくるメッセージを元に処理を分けられる
- ▶ 別途インストールが必要
 - ▶ `#sudo apt install expect` (ubuntuなので・・・、RH系だとdnf)
- ▶ 文法的にはTcl言語 (あまり馴染みがない・・・)

5. bashのコマンド内コマンド対話処理

5.4 expectのカンタンな記述例

```
expect “
```

```
spawn ssh user@192.168.0.44
```

```
expect “password”
```

```
send “hogehogeYr”
```

```
exit 0
```

```
“
```

expect “～この間～”で、この間が
expectの対話処理

spawnで、コマンドを呼び出す

expectで”password”という文字列を
待ち受ける

文字列が来たらsendでメッセージや
コマンドを送る

見にくいので、エスケープのバックスラッシュ省略

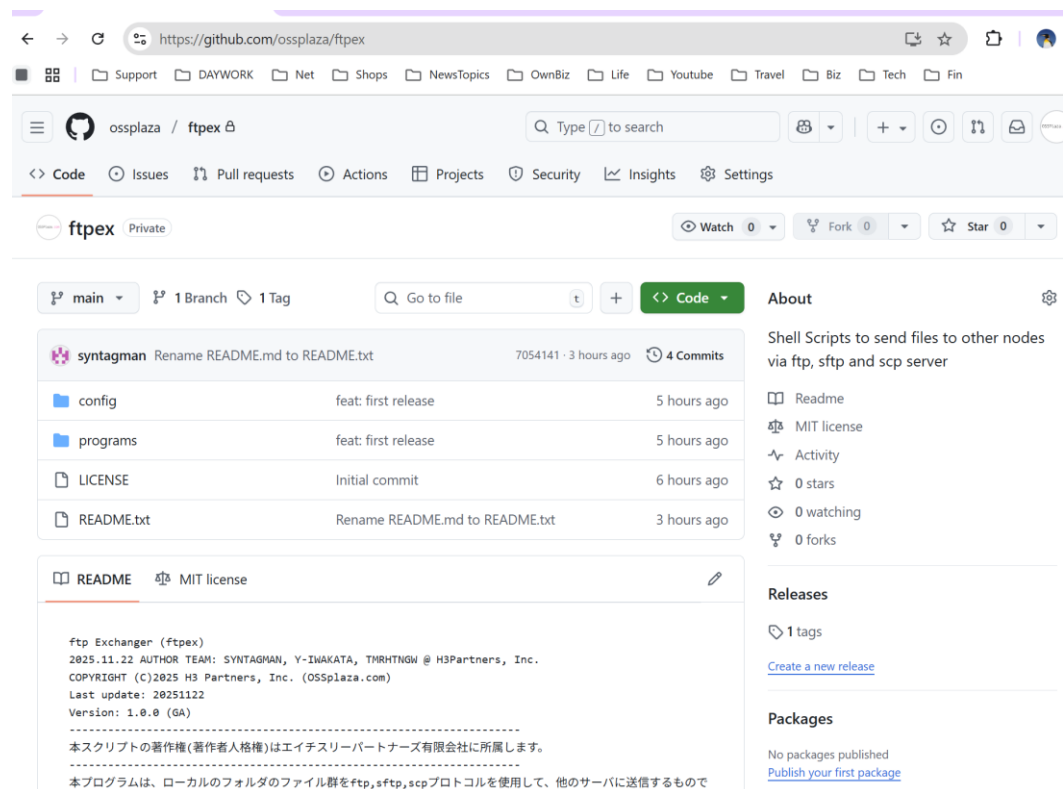
とはいっても、Tclの挙動をイマイチ把握できず、期待した動作をさせるのは
ちょっと大変でした。

6. ftpexの入手先

- ▶ ftpexを、github上で公開しました！（公開ホヤホヤです）

<https://github.com/ossplaza/ftpex>

- ▶ MITライセンスを設定しましたので、自由にイジって使っていただけます
よろしければどうぞ



7. クローリング

7.1 一応、バイト先から

- ▶ 会社等でご利用で、「サポートを受けたい」「構築して欲しい」「改造して欲しい」など、ございましたら社までお問い合わせください。



7. クロージング

7.2 北海道にぜひ遊びに来てください！

- ▶ これからの季節は「超」冬です
(流氷と時計台)



- ▶ OSC北海道は6月か7月
(富良野のラベンダーと小樽)



7. クロージング

7.3 九州に負けないくらい美味しいものもあります

▶ 海鮮丼



▶ 札幌らーめん



▶ スーパークレー



▶ ジンギスカン



▶ ご清聴ありがとうございました！！

OSSPlaza.com