

**使ってみよう Open usp Tukubai ! !**

**初心者のためのチュートリアル**

2012年8月24日

ユニバーサル・シェル・プログラミング研究所

## Open usp Tukubai ってなに？

Open usp Tukubai は、売上データの処理や勘定系システムの構築に利用されているエンタープライズ向けの高性能コマンド群「**usp Tukubai**」のオープンソース版です。業務で特によく使われるコマンドを選びすぐって Python で再実装したものです。多くの方々に usp Tukubai の便利さを知ってもらいたいという願いから公開をはじめました。

usp Tukubai は短期間低コストで企業システムを構築することができる**ユニケー****ジ開発手法**で使われています。ユニケージ開発手法は東急ハンズさまや成城石井さま、ローソンさまなど数々の企業さまでシステム構築の方法として採用されています。

ユニケージ開発手法はデータの整理方法からシステムのデプロイ、人材教育まで含めた包括的な開発手法です。学習が容易で、現場の担当者をシステム開発者まで引き上げることができます。

**Tukubai on FreeBSD** はこうした開発手法を知っていただきたいと、すぐに Open usp Tukubai を利用できるようにセットアップされたプラットフォームです。VirtualBox の仮想環境として提供されています。FreeBSD は Open usp Tukubai を簡単に試すことができるプラットフォームです。

Open usp Tukubai に関する情報は **UEC - usp engineers' community site** にまとまっています。



**UEC - UNIX を深く理解し、シェルスクリプティングを極めるコミュニティサイト**  
**<https://uec.usp-lab.com/>**

**本チュートリアル**を実践しながら、ぜひ UEC のサイトを訪れてみてください。ユニケーエンジニア手法や Open usp Tukubai に関するもっと深い情報がまとまっています。オンラインのコマンドマニュアルが提供されているほか、プログラミングのサンプルや面白い話が満載です。

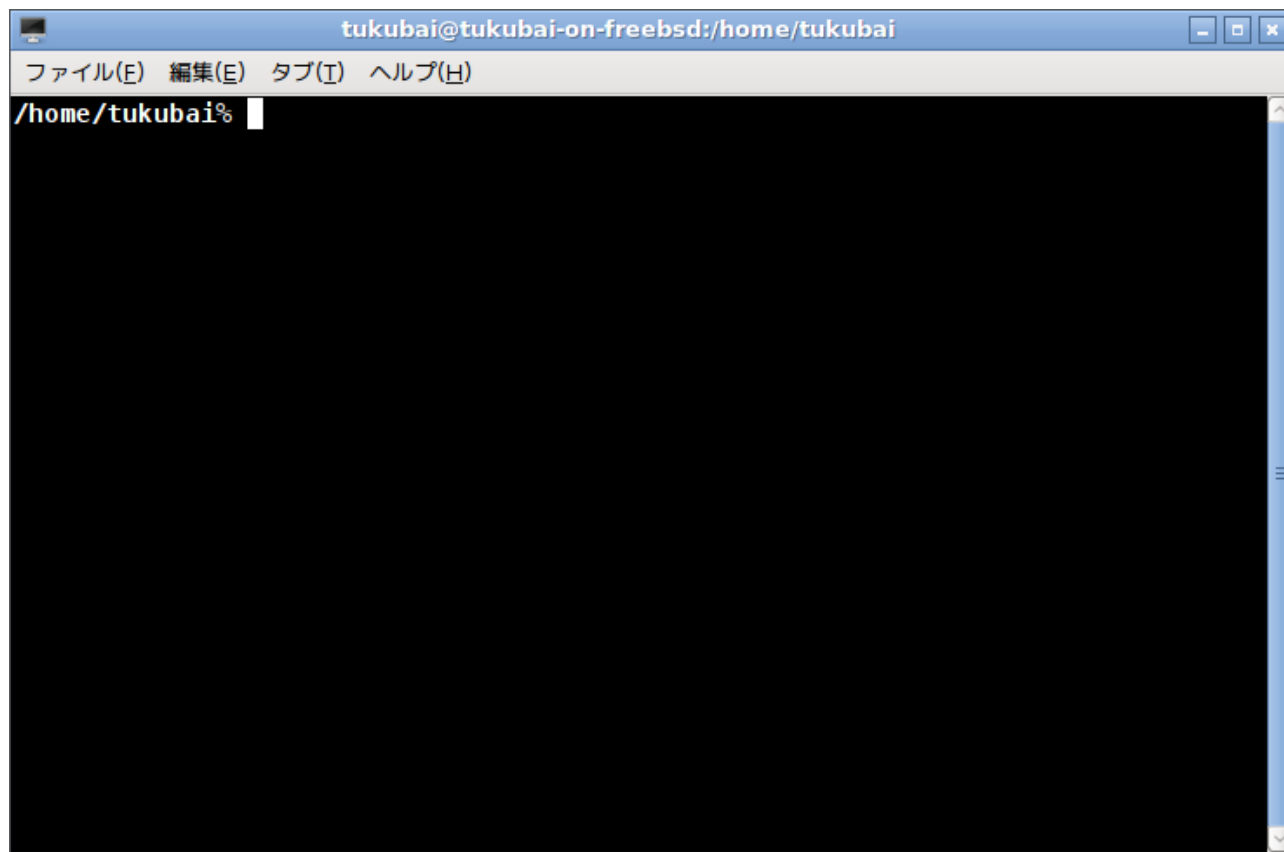
Contact us: uecinfo@usp-lab.com

Copyright (c) 2012 Universal Shell Programming Laboratory All Rights Reserved.

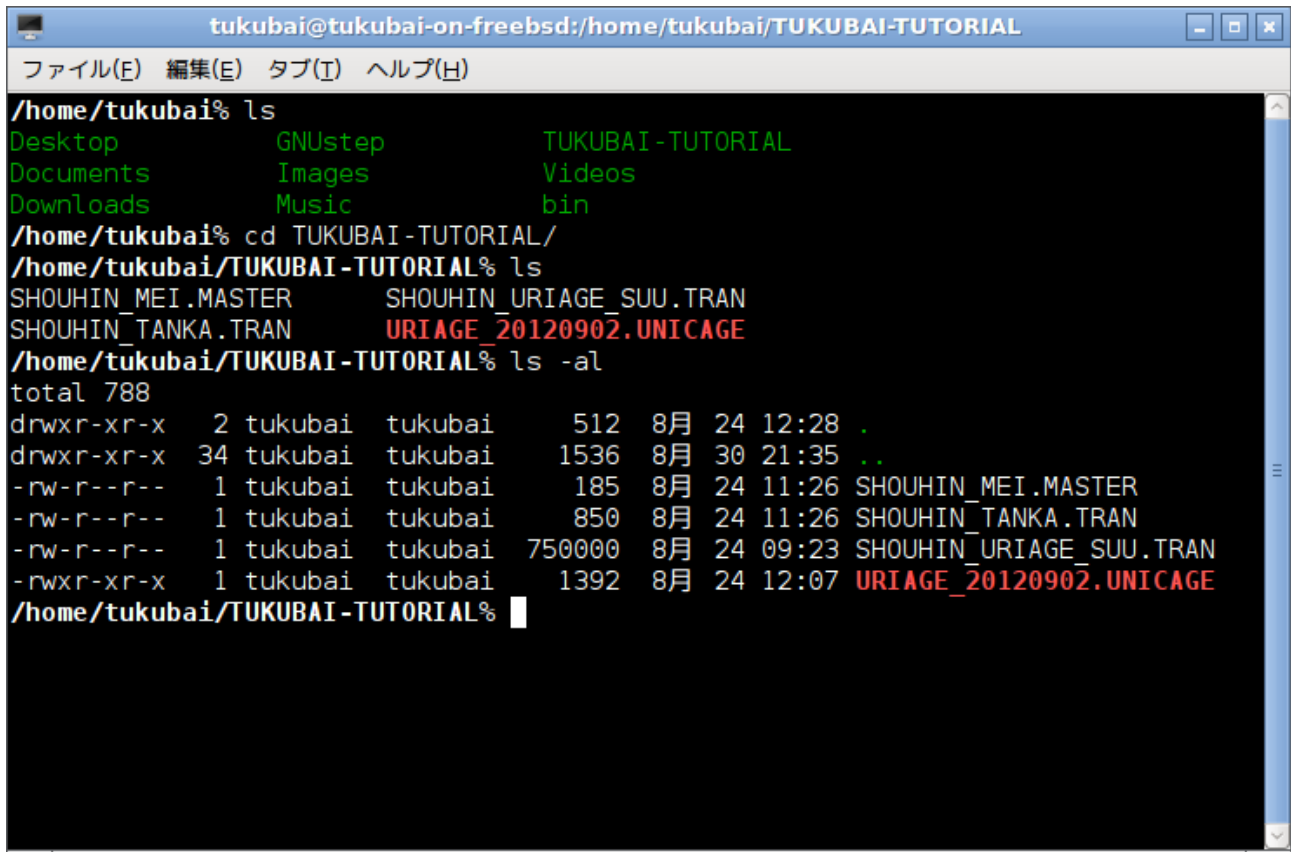
## ターミナルでの操作

Open usp Tukubai は業務データを処理するためのコマンドです。操作はターミナルからコマンドを実行したり、作成したシェルスクリプトを実行することでおこないます。

Tukubai on FreeBSD を起動して立ち上がってくる黒いウィンドウがターミナルです。ここにコマンドを入力して操作します。



ために **ls** と入力してエンターキーを押してみましょう。ホームディレクトリにあるファイルやディレクトリの一覧が表示されます。



```
tukubai@tukubai-on-freebsd:/home/tukubai/TUKUBAI-TUTORIAL
ファイル(E) 編集(E) タブ(T) ヘルプ(H)
/home/tukubai% ls
Desktop          GNUstep          TUKUBAI-TUTORIAL
Documents        Images           Videos
Downloads        Music            bin
/home/tukubai% cd TUKUBAI-TUTORIAL/
/home/tukubai/TUKUBAI-TUTORIAL% ls
SHOUHIN_MEI.MASTER  SHOUHIN_URIAGE_SUU.TRAN
SHOUHIN_TANKA.TRAN  URIAGE_20120902.UNICAGE
/home/tukubai/TUKUBAI-TUTORIAL% ls -al
total 788
drwxr-xr-x  2 tukubai  tukubai    512  8月 24 12:28 .
drwxr-xr-x 34 tukubai  tukubai   1536  8月 30 21:35 ..
-rw-r--r--  1 tukubai  tukubai    185  8月 24 11:26 SHOUHIN_MEI.MASTER
-rw-r--r--  1 tukubai  tukubai    850  8月 24 11:26 SHOUHIN_TANKA.TRAN
-rw-r--r--  1 tukubai  tukubai  750000  8月 24 09:23 SHOUHIN_URIAGE_SUU.TRAN
-rwxr-xr-x  1 tukubai  tukubai   1392  8月 24 12:07 URIAGE_20120902.UNICAGE
/home/tukubai/TUKUBAI-TUTORIAL% █
```

ホームディレクトリにある **TUKUBAI-TUTORIAL** というディレクトリに、サンプルのスクリプトやデータが格納されています。

ファイル名やディレクトリ名が大文字なのは、**ユニケース開発手法**でこうした名前規則を採用しているためです。別に強制するようなものではありませんので、気になる場合には小文字を使ってもかまいません。

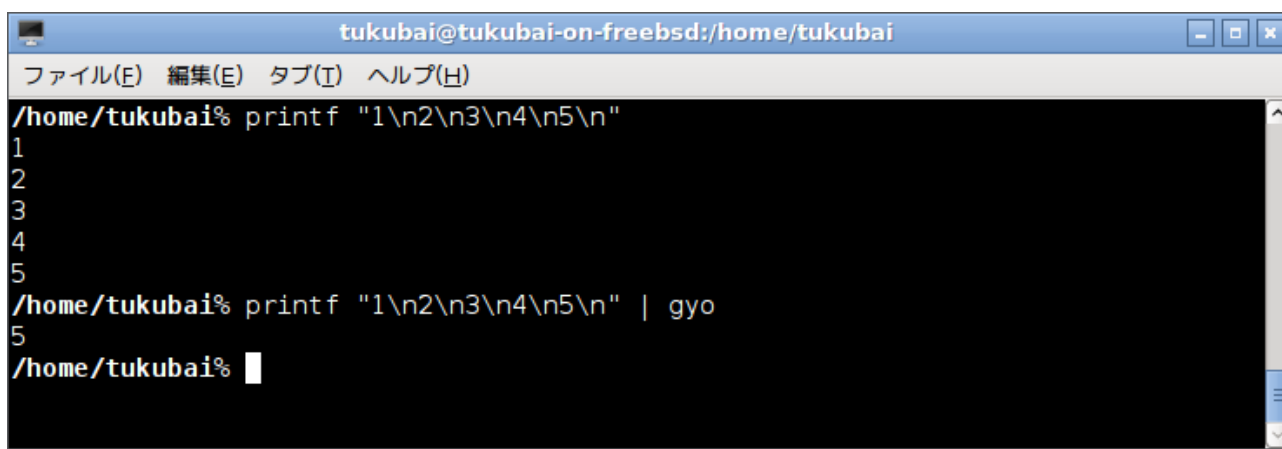
ターミナルに表示されている `/home/tukubai%` や `/home/tukubai/TUKUBAI-TUTORIAL%` はプロンプトと呼ばれています。ここにコマンドを入力してエンターキーを押します。

プロンプトにはカレントディレクトリが表示されています。カレントディレクトリは、ユーザが操作の対象とするディレクトリのことです。cd コマンドで移動することで変更できます。

Tukubai on FreeBSD でいろいろ試す前に、Tukubai on FreeBSD から抜ける方法を覚えておきましょう。Windows では **右 Ctrl キー** で、Mac OS X では **左 Command キー** を押すことで Tukubai on FreeBSD を抜けることができます。操作がよくわからなくなったら、このキーを押して Tukubai on FreeBSD を抜けてみてください。

## Open usp Tukubai 初仕事

ターミナルで次のようにコマンドを入力してみましょう。最初のコマンド `printf "1\n2\n3\n4\n5\n"` は **5行分** のデータを出力するコマンドです。`\n` が改行を意味しています。日本語キーボードでは `¥` と印字されているかもしれませんが、`¥` と `\` は同じものだと思っておいてください。



```
tukubai@tukubai-on-freebsd:/home/tukubai
ファイル(E) 編集(E) タブ(T) ヘルプ(H)
/home/tukubai% printf "1\n2\n3\n4\n5\n"
1
2
3
4
5
/home/tukubai% printf "1\n2\n3\n4\n5\n" | gyo
5
/home/tukubai% █
```

2つ目のコマンドは、1つ目のコマンドの出力を **gyo(1)** という Open usp Tukubai のコマンドに流し込んでいます。**gyo(1)** は行数をカウントするコマンドです。5行なので5と出力されていることがわかります。Open usp Tukubai は基本的にこのようなコマンドを組み合わせてデータを加工していきます。簡単ですね。

それでは、以降、サンプルスクリプトを読みながら、Open usp Tukubai でどういった処理をするのか追っていきましょう。

## チュートリアルデータの説明

チュートリアル向けのデータとして「野菜の売上データから、特定の日付の野菜別の売上合計を求める処理」を想定して、サンプルを用意しました。ホームディレクトリにある **TUKUBAI-TUTORIAL** ディレクトリ以下にまとめてあります。

```
$ cd
$ tree TUKUBAI-TUTORIAL/
TUKUBAI-TUTORIAL/
├── SHOUHIN_MEI.MASTER          商品番号と商品名
├── SHOUHIN_TANKA.TRAN         商品と単価データ
├── SHOUHIN_URIAGE_SUU.TRAN    商品と売上数データ
└── URIAGE_20120902.UNICAGE    集計プログラム

0 directories, 4 files
$
```

データ処理は **URIAGE\_20120902.UNICAGE** ファイルにまとまっています。特に使われることが多い `sm2(1)`、`join1(2)`、`self(1)`、`delf(1)`、`comma(1)`、`keta(1)` といった Open usp Tukubai コマンドを使用しています。



# SHOUHIN\_MEI.MASTER

商品番号と商品名を格納したマスタファイルです。次のようなデータが入っています。

001 カイワレダイコン

002 タマネギ

003 ホウレンソウ

004 チンゲンサイ

005 キャベツ

006 レタス

007 トマト

008 ニンジン

009 ダイコン

010 ゴボウ

マスタファイルは第1フィールドがキーフィールドとして `sort(1)` で整列されている必要があります。キーの重複は許可されません。

# SHOUHIN\_TANKA.TRAN

商品番号と日付、その日付の商品の単価が格納されたファイルです。次のようなデータが格納されています。

001	20120901	120
002	20120901	110
003	20120901	150
004	20120901	120
005	20120901	180
006	20120901	110
007	20120901	130
008	20120901	170
009	20120901	180
010	20120901	130
001	20120902	160
002	20120902	120
003	20120902	180
004	20120902	110
005	20120902	180
006	20120902	130
007	20120902	130
008	20120902	160
009	20120902	180
010	20120902	120

第1フィールドが商品番号、第2フィールドが日付、第3フィールドが商品単価です。

## SHOUHIN\_URIAGE\_SUU.TRAN

商品番号、日付、商品の売上数が格納されたファイルです。レジで販売した時点でデータが随時追加されていくファイルのようなものと想像してください。たとえば玉ねぎを2つ購入すると、「002 日付 2」というレコードがこのファイルに追加されます。

```
007 20120901 3
001 20120901 3
005 20120901 2
009 20120901 2
001 20120901 2
007 20120901 5
007 20120901 4
003 20120901 5
009 20120901 5
006 20120901 4
001 20120901 3
010 20120901 5
000 20120901 5
006 20120901 5
001 20120901 2
010 20120901 5
010 20120901 3
000 20120901 5
005 20120901 2
001 20120901 4
```

これらデータのファイルを `Open usp Tukubai` のコマンドを組み合わせることで、特定の日付の売上金額を求めるといったことが簡単に実現できます。

## URIAGE\_20120902.UNICAGE

2段階の処理を経てデータを処理しています。まず、個々の売上数データを加算して、9月2日における商品ごとの総売上数を計算します。

スクリプトの最初で定義しているのは、一時ファイル用の変数です。ユニケース開発手法ではこのように変数を設定し、一時ファイルの出力先として利用します。

まず処理の対象となるレコードを SHOUHIN\_URIAGE\_SUU.TRAN から日付で `grep(1)` することで取り出します。`cd ~/TUKUBAI-TUTORIAL;`  
`cat URIAGE_20120902.UNICAGE` とコマンドを実行してスクリプトの中身を表示させてみましょう。

```
#!/bin/sh

tmp=/tmp/$$

# 2012年9月2日の売上データを抽出
# 1:商品番号 2:日付 3:個別売上数
grep 20120902 SHOUHIN_URIAGE_SUU.TRAN      |
# 商品番号で整列
sort -k1                                     |
# 商品番号をキーにして第3フィールドの売上数を加算
sm2 1 1 3                                   |
# 1:商品番号 2:総売上数
# 商品番号をキーにして商品名をマージ
join1 key=1 SHOUHIN_MEI.MASTER -           > $tmp-1
# 1:商品番号 2:商品名 3:総売上数
```

取り出したデータは `sort(1)` で整列をかけたあとで、`sm2(1)` コマンドで商品ごとに加算しています。`usp Tukubai` のコマンドはデータが整列されていることを前提にしたものが多く、そうすることで実行速度の高速化を実現しています。

マスタファイルとトランザクションファイルを `join1(1)` を使って結合します。この処理で商品番号のあとに商品名が挿入されることとなります。この段階で出力される一時ファイルには次のようなデータが書き込まれています。

```
001 カイワレダイコン 2778
002 タマネギ 2697
003 ホウレンソウ 2763
004 チンゲンサイ 2830
005 キャベツ 2685
006 レタス 2911
007 トマト 2679
008 ニンジン 2763
009 ダイコン 2668
010 ゴボウ 2879
```

ユニケーシ開発手法では、途中経過のデータをファイルへ出力します。問題が発生したときに問題をトレースしやすくする効果があるほか、データが重複するため、万が一、操作を誤ってファイルを削除した場合などでも、データの復旧がやりやすくなります。

スクリプトのコメントに、どのフィールドが何を意味しているのか書き込まれている点に注目してください。usp Tukubai のコマンドや UNIX コマンドを経由して出力されるフィールドの意味が変わったら、その段階でどのフィールドが何のデータになっているのかをコメントに記載します。

こうすることで、あとから処理を変更する場合などに編集しやすくなります。

次に、9月2日の商品単価を取り出し、先ほど出力した一時ファイルにマージします。先ほどと同じように `grep(1)` で対象となるデータを取り出して、念のため `sort(1)` したあとで `join1(1)` を使ってマージします。

```
# 2012年9月2日の商品単価データを抽出
# 1:商品番号 2:日付 3:商品単価
grep 20120902 SHOUHIN_TANKA.TRAN |
# 商品番号で整列
sort -k1 |
# 商品番号をキーにして売上数と単価をマージ
join1 key=1 $tmp-1 - |
# 1:商品番号 2:商品名 3:総売上数 4:日付 5:商品単価
# 不要なフィールドを削除
delf 1 4 |
# 1:商品名 2:総売上数 3:商品単価
# 売上数と単価の積算から売上額を出力
awk '{print $1, $2*$3}' |
# 1:商品名 2:売上額合計
# 売上額をコンマ区切りへ変更
comma 2 > $tmp-2
```

売上数と単価のデータがマージされたので、`awk(1)` を使って双方の値の積算をとって出力します。これで売上額が求まります。

データとして不要なフィールドは `delf(1)` で取り除きます。売上額は `comma(1)` を使って3桁ごとに区切り文字を追加させます。

この段階で出力される一時ファイルには次のようなデータが書き込まれています。

カイワレダイコン 444,480

タマネギ 323,640

ホウレンソウ 497,340

チンゲンサイ 311,300

キャベツ 483,300

レタス 378,430

トマト 348,270

ニンジン 442,080

ダイコン 480,240

ゴボウ 345,480

この段階で最終的にほしいデータがすでにそろっています。



最後に、体裁を整えてデータを出力します。keta(1)は出力の桁をそろえるコマンドです。

```
# 体裁を整えてデータを表示
# タイトルを表示
echo "2012年9月2日の売上データ"
echo
# ヘッダを表示
echo "商品 売上額(円)" |
# 売上額データを出力
cat - $tmp-2 |
# 桁を揃えて表示
keta --

# 一時ファイルを削除
rm $tmp-*
```

スクリプトの一番最後に一時ファイルを削除して処理完了です。デバッグする場合、一時ファイルの削除をコメントアウトして、随所々々のデータを調べます。

## サンプルスクリプトの実行

サンプルのスクリプトを実行すると、次のような結果が得られます。

```
$ cd ~/TUKUBAI-TUTORIAL
$ ./URIAGE_20120902.UNICAGE
2012年9月2日の売上データ
```

商品	売上額(円)
カイワレダイコン	444,480
タマネギ	323,640
ハウレンソウ	497,340
チンゲンサイ	311,300
キャベツ	483,300
レタス	378,430
トマト	348,270
ニンジン	442,080
ダイコン	480,240
ゴボウ	345,480
\$	

スクリプトを実行するだけでなく、スクリプトの内容を参考にして自分で手動で入力して一行ずつコマンドを実行してみてください。どのようにデータが変換されていくのかがよくわかります。

このサンプルでは50,000レコード処理しているだけですが、実際の業務システムでは数億レコード、店舗商品マスタが10万件といったレベルのデータを処理します。

Open usp Tukubai で提供されているコマンドを組み合わせるだけでも、多種多様なデータを高速に変換し、目的とするデータを得ることができます。

データはL1からL5まで5段階に区別され、それぞれどういった形式にデータを揃えるかが定められています。今回取り上げたデータはL3形式のデータです。

ユニバーサル・シェル・プログラミング研究所ではユニケーゼエンジニアを育成するための教育講座を開講しています。本チュートリアルだけではよくわからない、もっと詳しくユニケーゼ開発手法や usp Tukubai のことを知りたいという場合には、ぜひ一度 [uecinfo@usp-lab.com](mailto:uecinfo@usp-lab.com) またはユニバーサル・シェル・プログラミング研究所 03-3432-1174 までご連絡ください。教育講座は毎月開講しています。