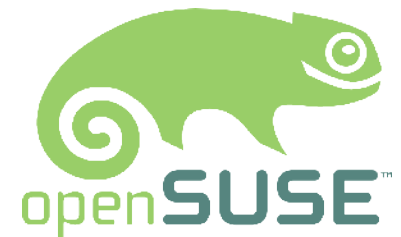


# 覚えておきたい! zypper コマンドの使い方

by @ftake  
日本 openSUSE ユーザ会 6月勉強会



# あらすじ

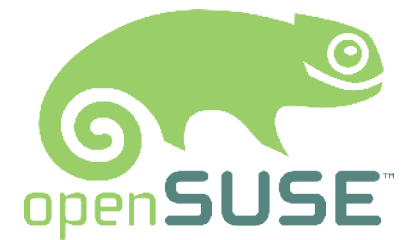
① zypper コマンドの使い方を復習する

② ツッコミ歓迎!

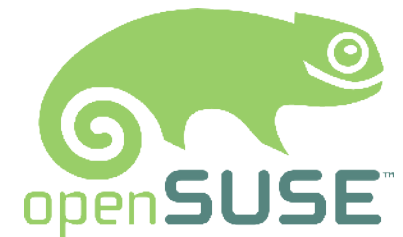
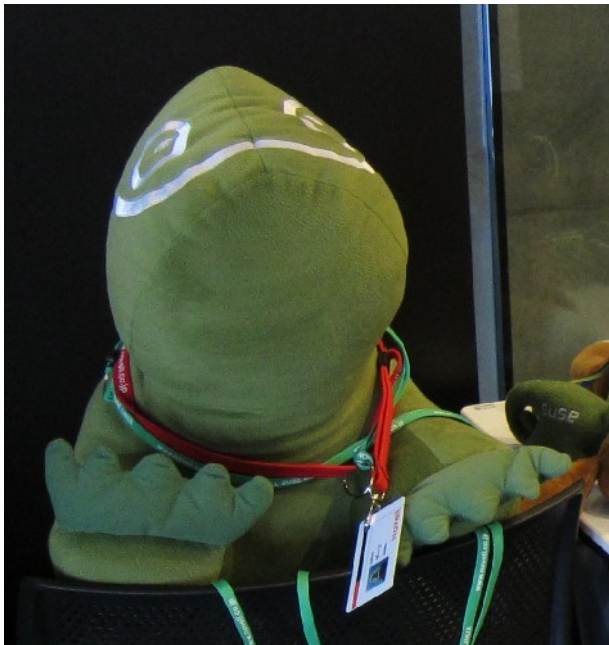
- 調べたけど、まだ怪しいところが
- 改良して OSC のセミナーなどのネタになればいいな



時はXXXX年

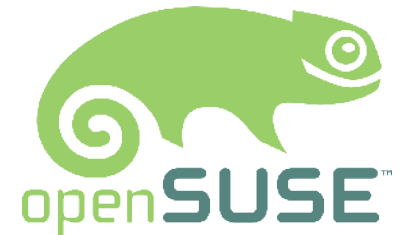
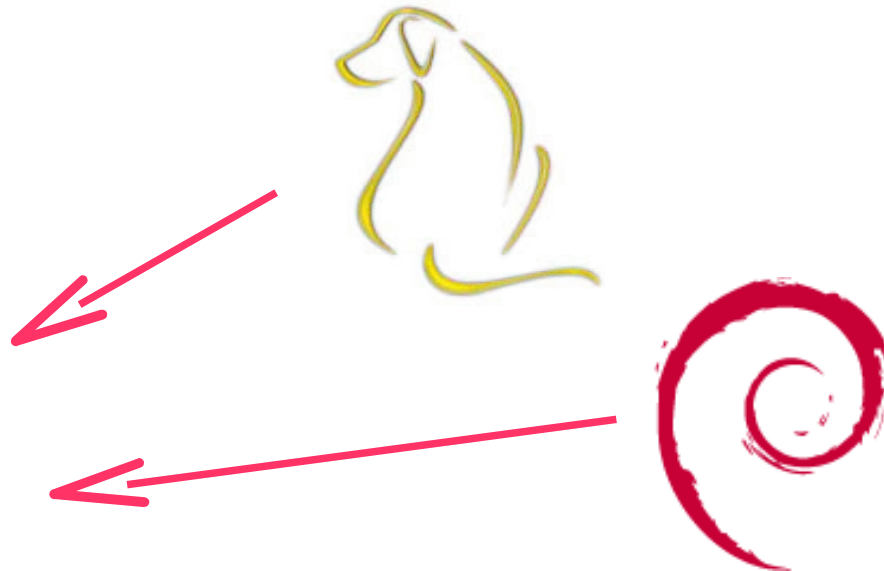


SUSE 帝国は皇帝 YaST によって支配されていた



• • • •

しかし、APT が率いるうずまき帝国など  
隣国の勢力が強まり



# SUSE 帝国は危機にさらされていた



そんなとき、ある勇者が現れた  
その名も



# Zypper

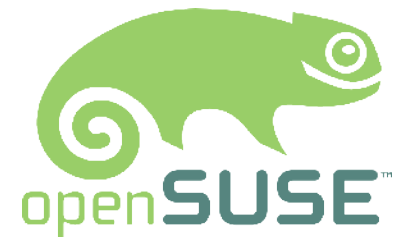




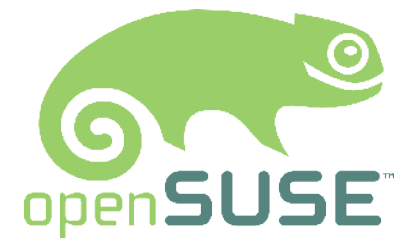
# Zypper は YaST の元で大活躍し、 危機を救ったのであった



←大活躍中



という話はさておき  
ここまでの話はフィクションです



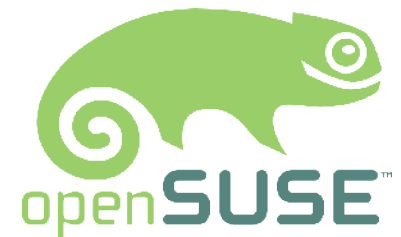
# zypper/Libzypp

## ④ パッケージ管理コマンド

- 10.2 から導入
- libzypp のフロントエンド
  - 現在の「YaST - ソフトウェアの管理」も libzypp のフロントエンド
- 他のディストロだと: aptitude, yum, (port)

## ④ GUI よりコマンドのほうが便利なおことがある！

- YaST もコンソールから使えるが Curses なので…GUI に近い (TUI)



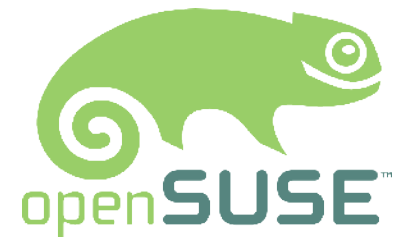
# 基本中の基本 1/2

## ④zypper コマンドの構文

- \$ zypper グローバルオプション サブコマンド オプション

## ④ヘルプを見るとき

- \$ zypper help
  - グローバルオプションとサブコマンドの一覧がみられる
- \$ zypper help サブコマンド
  - サブコマンドのオプションなどがみられる
- \$ man zypper



# 基本中の基本 2/2

```
$ zypper help
```

使用方法:

```
zypper [グローバルオプション] <コマンド> [コマンドオプション] [パラメータ]
```

グローバルオプション:

```
--help, -h
```

ヘルプ

```
--version, -V
```

バージョン情報を表示

**\*\*中略\*\***

コマンド:

```
help, ?
```

ヘルプを表示

```
shell, sh
```

一度で複数のコマンドを入力する

リポジトリの操作:

```
repos, lr
```

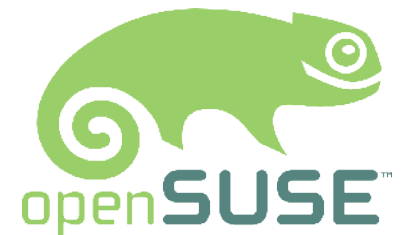
設定済みのリポジトリを表示する

```
addrepo, ar
```

新しいリポジトリを追加する

```
removerepo, rr
```

指定したリポジトリを削除する



# ポイント: 長い名前と短い名前

④ サブコマンドやオプションに長い名前と短い名前がある

- ex) install → in
- ex) repos → lr (list repositories)
- ex) --detail → -s (??)

④ 無理に短い方を覚えられない方がいいかもしれない



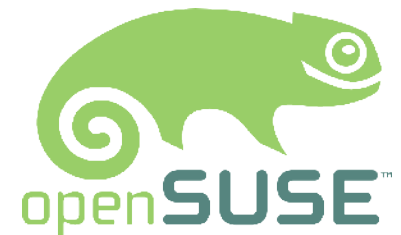
# パッケージの検索: search

⑥登録されているリポジトリにあるパッケージを検索する

- \$ zypper search キーワード
- キーワードが名前の一部に含まれているパッケージを検索

```
fuminobu@tpx41l:~> zypper search dolphin
リポジトリのデータを読み込んでいます...
インストール済みのパッケージを読み込んでいます...
```

S	名前	概要	種類
i	dolphin	KDE File Manager	パッケージ
	dolphin-plugins	version control plugins for dolphin	パッケージ



# パッケージの検索: search

## ⑥ パッケージのバージョンや提供元も表示する

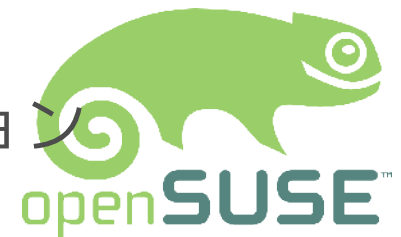
- `$zypper search --detail`

```
fuminobu@tpx41l:~> zypper search --detail dolphin
リポジトリのデータを読み込んでいます...
インストール済みのパッケージを読み込んでいます...
```

S	名前	種類	バージョン	アーキテクチャ	リポジトリ
i	dolphin	パッケージ	4.6.0-6.6.2	i586	openSUSE-11.4-0ss
	dolphin-plugins	パッケージ	4.6.0-4.7.1	i586	openSUSE-11.4-0ss

## ⑥ S 列の読み方

- i: インストール済み
- v: インストールされている以外のほかのバージョン





# パッケージの検索: search

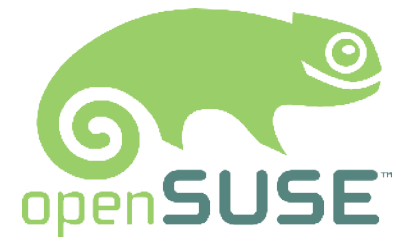
④ `--match-exact`: 完全一致

- 部分検索だと `--detail` で結果が多すぎることもある

④ `-i`: インストール済みのパッケージを検索

④ キーワードを2つ並べると…

- `search foo bar` は `search foo` と `search bar` の結果をあわせたもの (`--match-any` ??)
- `--match-all` `--match-any` でなぜが違いがでない



# パッケージのインストール: install

④ \$ zypper install パッケージ

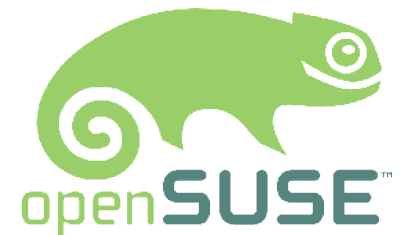
④ パッケージ名/能力名:

NAME[.ARCH][OP<VERSION>]

- ex) \$ zypper install "firefox>=4.0"  
- クオートするかエスケープする

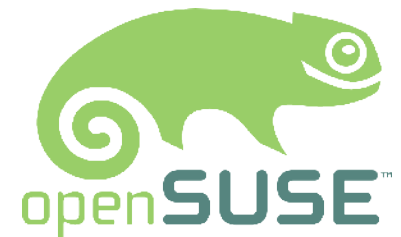
④ URI

- ex) \$ zypper install http://example.com/foo.rpm
- ex) \$ zypper install ~/Download/bar.rpm



# ポイント: インストール元リポジトリ

- ◎ どのリポジトリのパッケージが選択されたかわからない
  - バージョンが新しいものが選ばれる
- ◎ グローバルオプション `-v` を指定
  - `$ zypper -v install foo`
  - リポジトリの情報が表示される



# パッケージの削除: remove

④ \$ zypper remove パッケージ

- パッケージの選択方法は install と同じ

④ install / remove どちらでもインストール・削除可能

- ex) \$ zypper install -- foo -bar
- ex) \$ zypper remove bar +foo
- どちらも foo をインストール、bar を削除



# パッチ: patch, list-patches (lp)

## ④未適用のパッチを表示

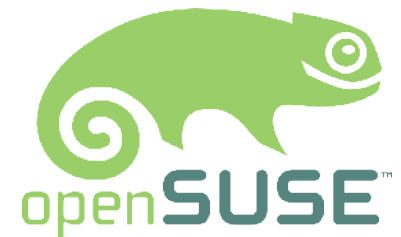
- \$ zypper lp
- パッチ数を表示するだけなら  
\$ zypper patch-check

## ④パッチの適用

- \$ zypper patch

## ④KpackageKit や YaST より手っ取り早い？

- 一部を選択するなら GUI のほうが楽



# パターン

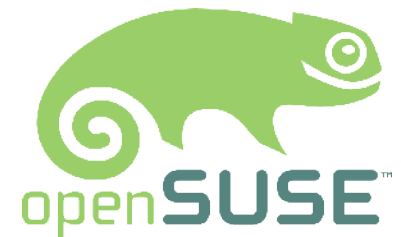
## ◎パッケージをグループ化したもの

- ex) devel\_C\_C++, kde4, lamp\_server, ...
- いくつかのパッケージをまとめて導入するのに便利

## ◎通常のパッケージと同様に検索・インストール可能

## ◎パターンだけを検索するには

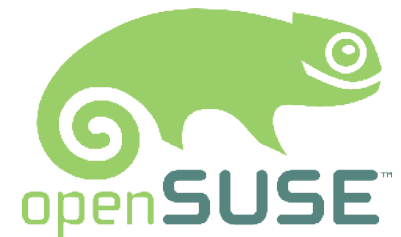
- `$ zypper search -t pattern`
- `$ zypper patterns`
  - 若干フォーマットが異なる



# リポジトリの管理…の前に

## ④zypper でサポートされているリポジトリ

- REPOMOD (Repository MetaData, rpm-md)
  - yum のフォーマット
- Yast2
  - 従来のフォーマット
  - インストール DVD などに使われる



# リポジトリの追加: addrepo

## ⑥ ベース URI で指定する方法

- `$ zypper addrepo --refresh URI alias`
  - 自動更新有効、alias はリポジトリの別名（自由に指定可）

## ⑥ repo ファイルを指定する方法

- `$ zypper addrepo --refresh http://example.com/foo.repo`
- ブラウザで URL をコピーすると楽





# リポジトリの修正: modifyrepo

## ⑥リポジトリのプロパティを設定する

- `$ zypper -p 100 1`
  - リポジトリ番号1 の優先度を 100 に設定
  - 優先度: 優先度が高い (値が小さい) リポジトリからインストール
- リポジトリ番号は `$ zypper repos` で確認できる
- リポジトリ番号の代わりに `alias` を指定しても OK

#	別名	名前	有効	更新
1	M17N_1	M17N	はい (Y)	いいえ (N)
2	My OBS Home 1	My OBS Home	はい (Y)	いいえ (N)
8	openSUSE-11.4-Source	openSUSE-11.4-Source	いいえ (N)	いいえ (N)

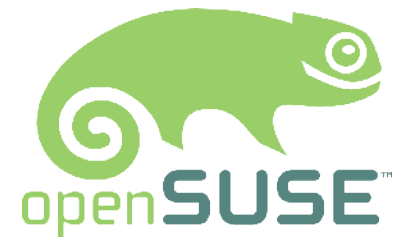
## ⑥URLの修正は YaST から簡単に行える

- `zypper` からの変更は一度リポジトリを削除する必要あり



# ポイント: リポジトリの自動更新

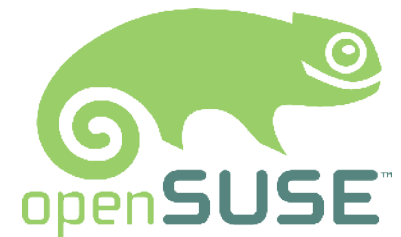
- ④ リポジトリの更新をチェックしてキャッシュを再構築
- ④ 検索のときでも更新チェックすると時間がかかる
  - 更新の多いリポジトリ (OBS, Packman) が含まれるとき
- ④ 自動更新を無効にして手動で更新
  - アップデート以外のリポジトリ
  - 週に1回くらい更新チェックをすればいいかも
  - `$ zypper refresh`



# 対象リポジトリを指定する

◎ほとんどのサブコマンドはリポジトリを `-r` で指定可

- `zypper search -r 1 firefox`
  - リポジトリ番号 1 から `firefox` を検索
- `zypper install -r M17N ibus`
  - M17N (alias) から `ibus` をインストール



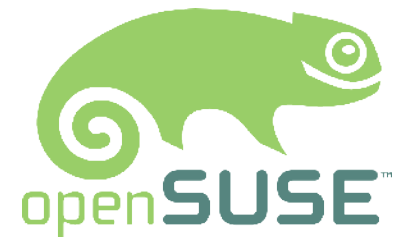
# パッケージの更新: update

## ① パッケージを新しいバージョンに更新する

- 「製造元」が同じで新しいバージョンがある場合
- 製造元:  
openSUSE … 公式リポジトリのもの  
obs://build.opensuse.org/M17N … M17Nプロジェクト
- 非公式リポジトリのパッケージを更新するのに使う
  - 公式はセキュリティパッチとして配布
- `$ zypper update ibus`
  - `ibus` を新しいバージョンに

## ② 更新可能なパッケージの一覧は

- `$ zypper lu`



# ポイント: 製造元を変更するには

## ① install コマンドで更新する

- `zypper install -r` リポジトリ パッケージ

## ② よく次のようなメッセージが出る

- 依存するパッケージの製造元も変更する必要があることがある
- この場合は落ち着いて 1 を選択

問題点: `foo-bar-1.0.0-1.1.i586` は `foo = 1.0.0-1.1` を必要としていますが、この要求を解決する方法がありません

削除したプロバイダ: `foo-1.0.0-1.1.i586`

解決方法 1: 以下の動作を実行します:

`foo-bar-1.0.1-1.2.i586` をインストールする (製造元を変更する)

`openSUSE --> obs://build.opensuse.org/home:ftake`

解決方法 2: `foo-1.0.1-1.2.i586` をインストールしない

解決方法 3: いくつかの依存関係を無視することによって `foo-bar` を壊します

いずれかの数字を入力するか、キャンセル(C) を入力してください [1/2/3/c]

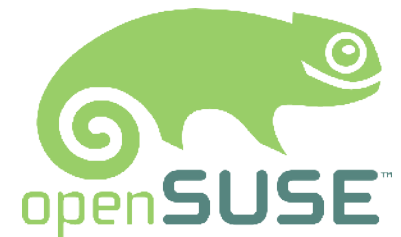


# すべて最新に: dist-upgrade (dup)

④ update と異なり、製造元に関わらず最新に更新

④ 11.3 から 11.4 に更新するときなど

- 11.3 で起動し 11.4 のリポジトリを登録してから  
\$ zypper dup
- インストールイメージ不要



# その他 1/2

④ source-install (si): ソースパッケージのインストール

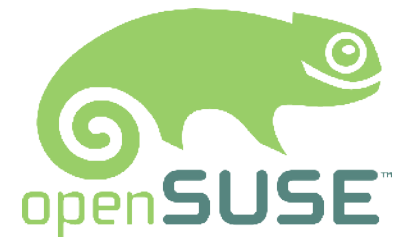
- ビルドに必要なパッケージも同時にインストール

④ ps: 削除したファイルを使用しているプロセス

- パッケージの更新後に \$ zypper ps でチェックして必要であれば終了する

④ install-new-recommends (inr):  
推奨パッケージをインストール

- ロケールを日本語に設定してから inr すると Anthy や日本語フォントが自動的に入る
  - YaST から設定すると勝手に実行される



# その他 2/2

④ versioncmp, vcmp: バージョン番号文字列の比較

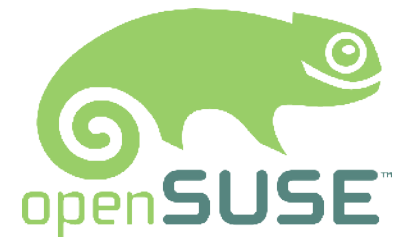
④ targetos, tos:

- \$ zypper tos  
openSUSE-11.4-i586

④ what-provides, wp

- \$ zypper wp java
  - java-1\_6\_0-sun
  - java-1\_6\_0-openjdk
  - java-1\_5\_0-gcj-compat

④ --xmlout: 結果を XML で出力

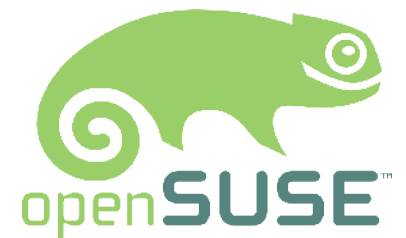




# おまけ

## ④zypper moo

- 試してみてください
- LANG=c と LANG=ja\_JP.utf-8 で結果が変わります



# おしまい

◎ 「これは覚えておきたい」というものが他にあれば

