

# Seleniumium



# Jenkins

ソフトウェアテストシンポジウム 2013 東北

今日のおはなし

**Selenium**×**Jenkins**  
を使ってWebアプリケーションの受入テスト  
を自動化した事例紹介  
をします

# Seleniumとは

Webアプリケーションのテストを自動化する**ツール群**

**Webブラウザの操作**を自動化することで、Webアプリケーションのテストを行う



Webブラウザの操作を記録してテストケースを作成する

# Selenium IDE

テストケースを実行する

# Selenium RC/WebDriver

複数のPCを使ってテストを行う

# Selenium Grid



# SeleniumIDE



ブラウザの操作をSeleniumの  
テストケースとして記録するツ  
ール

Firefoxのアドオンとして提供

# Jenkinsとは

オープンソースのCIサーバ  
(Java製)

今やCIサーバの代名詞的存在

開発者は日本人の川口耕介さん

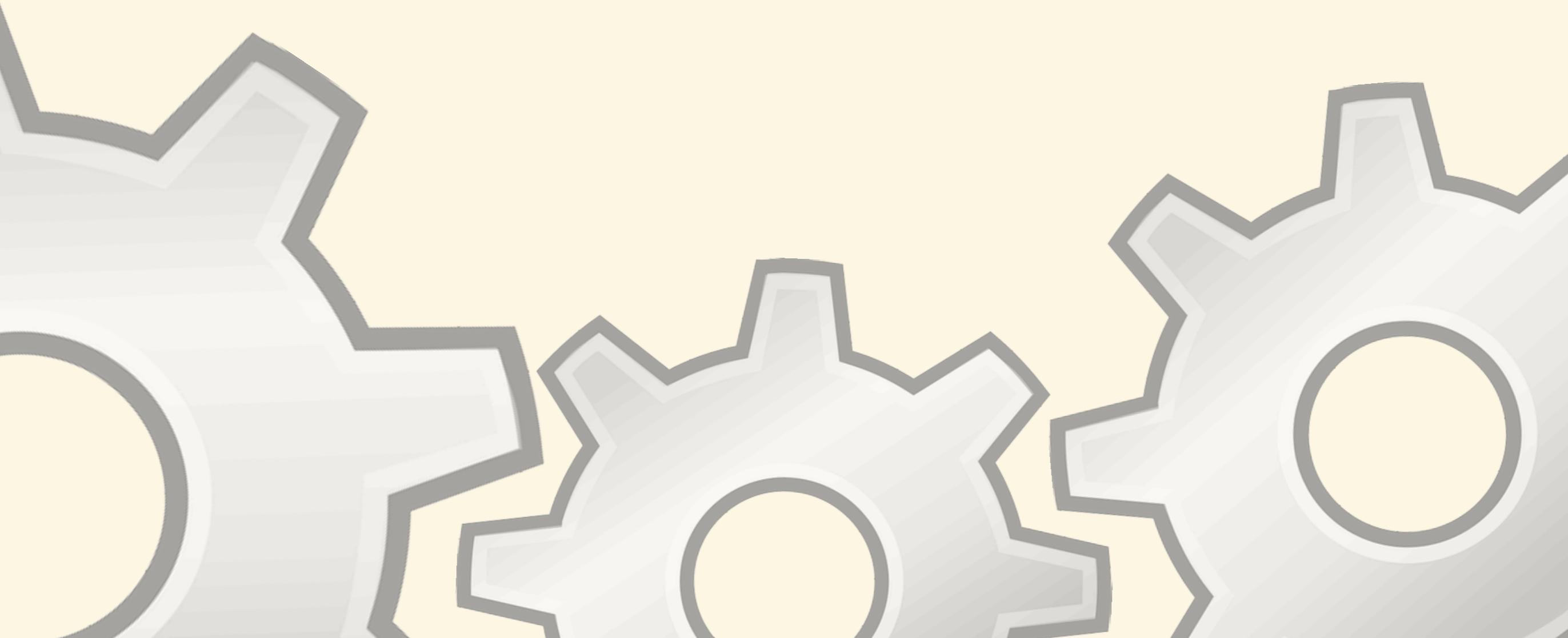
CI : Continuous Integration  
継続的インテグレーション

簡単に言うと

超高性能

cron

# 5秒でわかった気になる 継続的インテグレーション



常に

ゼルネト

世よ!!



**60秒**で補足する

継続的インテグレーション



# 名言あらかる



Naoya Ito

@naoya\_ito



フォロー

継続的インテグレーションがなぜ重要かって  
のをお堅い人たちに話すときに、テストの自  
動化とか言うんじゃないくて 「高速に何度も石  
橋を叩いて渡る方法だからです」といえばい  
いって教えてもらって目から鱗だった



返信



リツイートを取り消し



お気に入り



その他

282

件のリツイート

282

お気に入り



2013年6月10日 - 10:11

[https://twitter.com/naoya\\_ito/status/332664039563993089](https://twitter.com/naoya_ito/status/332664039563993089)

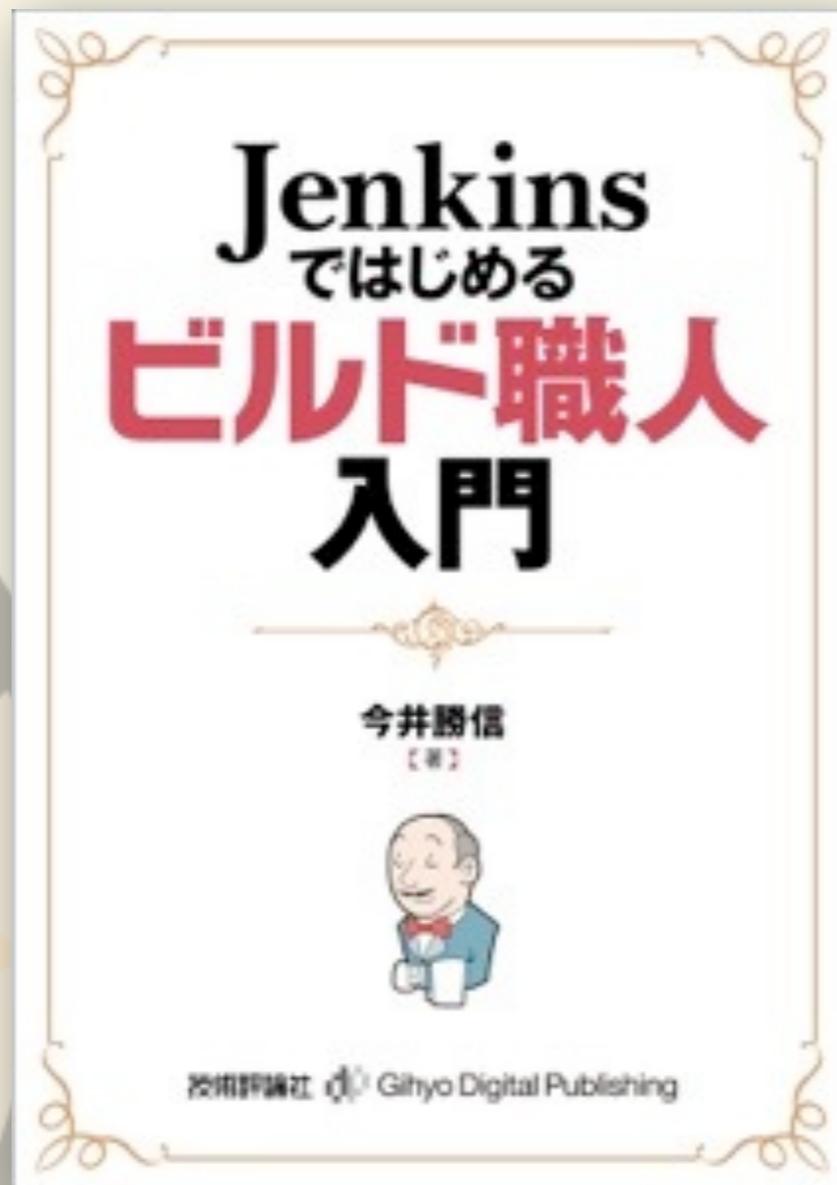
# 『現代ソフトウェア開発 の三本柱

1. バージョン管理
2. テスティング
3. 自動化』

なんでも

自動化しよう

# くわしくは書籍で...



**Selenium**

**x**

**Jenkins**

**による化学反応**

とこるで...

どうやって  
**Selenium**を現場適  
用したのか？

どうやって**Jenkins**  
を現場適用したの  
か？

気になりますよね...

ウ・(・)チラッ

残念ながら

わたしの一存で決め  
ました



# 主な原動力

# 面倒はキライ

• う • ぐ

でも、すべてがそう  
上手く行くことは無  
いですよね...

（泣）

能書きはここでおしまいです

次から事例紹介へ

「(3)」(4)

実現したかったこと

Seleniumのテスト  
の**起動**を自動化したい

できれば**複数**のWeb  
ブラウザでの動作テス  
トも行いたい



ちなみに...

(これまた自分の好みで)

テストケースのフォーマットは  
**HTML** (Selenese) を利用

**seleniumhq** プラグイン

<https://wiki.jenkins-ci.org/display/JENKINS/Seleniumhq+Plugin>

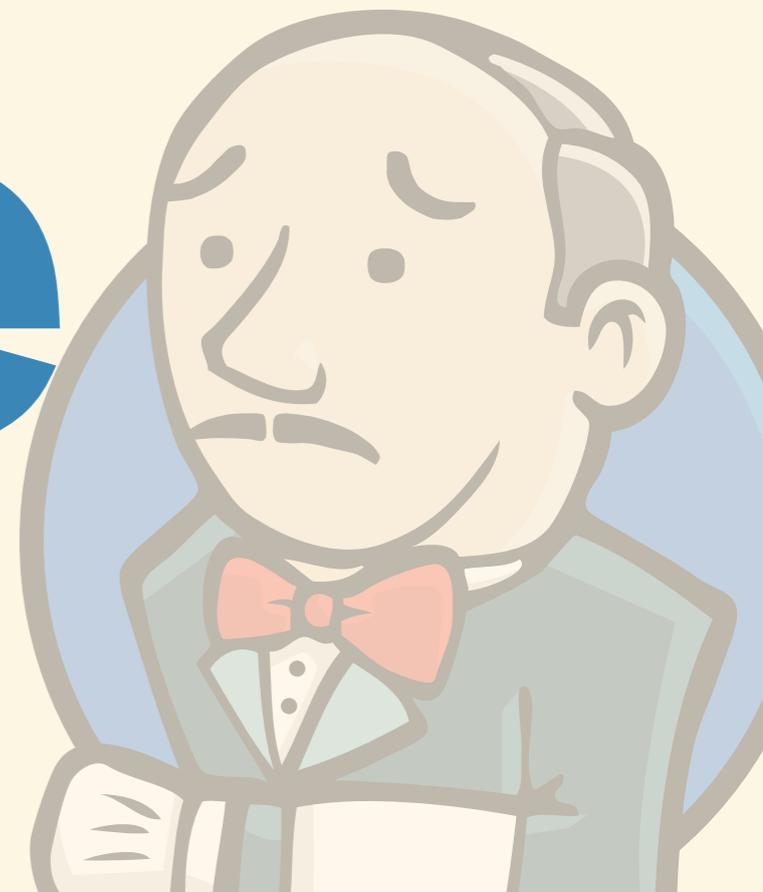


(ご主人さま)

MASTER

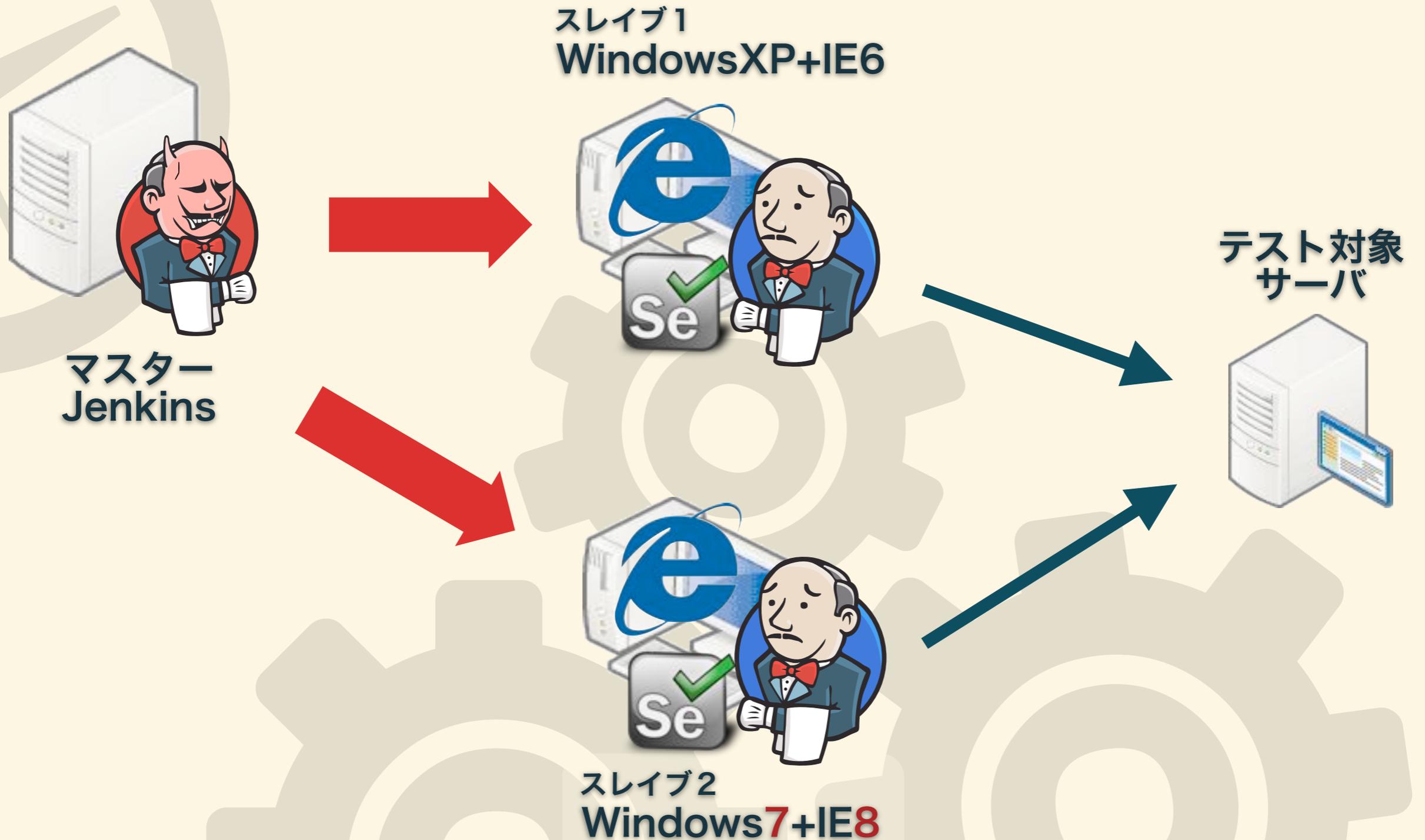
(奴隸)

slave



Jenkinsは**おど**  
**る**く**ほ**ど簡単に、  
**マ**スタ/**ス**レイブ  
構成を構築**で**きる

# 構築したシステムの構成



まとめ

良い点：

- ▶ **Selenium** & **Selenese** は、Webアプリケーションのテストを自動化するお手軽な方法
- ▶ Jenkins と組み合わせることで、テストを**ほぼ完全に自動化**できる
- ▶ **構築 / 運用コスト 約0円**  
(電気代は除く)

# 悪い点：

あえて言えば...

## ▶ Seleniumのクセが強い

- ▶ ツールの性格上、各WebブラウザのJavaScript実装の影響を受ける
- ▶ Selenium自体が万能ではない

## 今後の課題

- ▶ 受入テストの**範囲** / **定義**
- ▶ テストケースが**不良債権化**するのを如何に防ぐか
- ▶ Selenium / Jenkinsの普及  
(CI環境の**属人化の防止**)

それでも、自動化して損すること  
とは少ない

大切なのは**バランス**

(割り切りとも言う)

なんでも

自動化しよう

**ご静聴**  
**ありがとう**  
**ございました**