

Rubyで ソフトウェアづくり をアジャイルにする ということについて

Making Software Development Agile w/ Ruby

角谷 信太郎

日本Rubyの会
(株)永和システムマネジメント
shintaro@kakutani.com

KAKUTANI Shintaro; Nihon Ruby-no-kai; Eiwa System Management, Inc.

関西Ruby会議02/関西オープンソース2009; 2009-11-06(Fri)

角谷信太郎

kakutani.com

KAKUTANI Shintaro

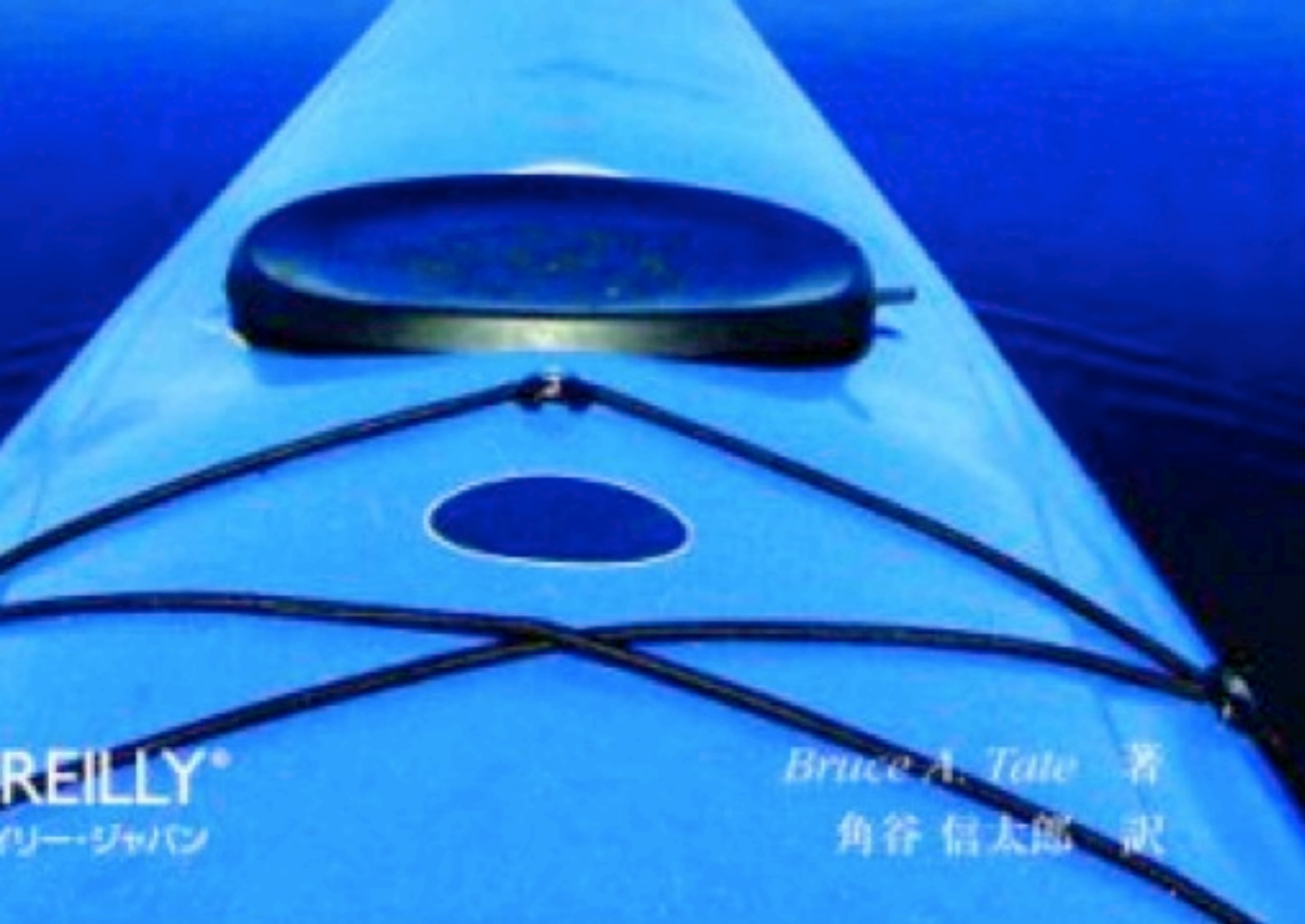
角谷 信太郎

✓ 日本Rubyの会理事

✓ RubyKaigi

✓ Regional RubyKaigi

✓ アジャイル関連技術書翻訳



インターフェイス 指向設計

アジャイル手法による
オブジェクト指向設計の実践



The Pragmatic Programmers
Practices of an Agile Developer
Working in the Real World

アジャイル プラクティス

達人プログラマに学ぶ現場開発者の習慣

Venkat Subramaniam and Andy Hunt 著 / 角谷信太郎・木下史彦 監訳



アジャイルな 見積りと計画づくり

価値あるソフトウェアを育てる概念と技法

MIKE COHN [著]
安井 力、角谷 信太郎 [監訳]

AGILE
ESTIMATING AND
PLANNING



このトークの後
17:00よりジュー
ンク堂書店Kの
F店(6F)にてサ
イン会があり☑

提 供

情報化技術を通じて社会と共生する



永和システムマネジメント

✓ **受託開発**の会社です

✓ 本社は**福井**なので**関西**のお仕事も歓迎です :)

✓ **Ruby**のお仕事承ります

✓ **JRuby**での納品実績もあります

✓ **アジャイル**な開発のご支援

✓ 認定**Scrum**マスターが4人います

今日、伝えたいこと



Beautiful Code

ビューティフルコード

Andy Oram, Greg Wilson 編

Brian Kernighan, Jon Bentley, 著

まつもと ゆきひろ 他

久野 禎子、久野 靖 訳

O'REILLY®

オライリー・ジャパン

“
プログラムを書いたことのない
システムエンジニアが威張って
いるような会社は早晚亡びる。

竹内郁雄 「推薦のことば」 (『ビューティフル・コード』)



**“With great power comes
great responsibility.”**



GOT TO GET A GRIP!

今日、お話しすること

Head First

ソフトウェア開発

頭とからだで覚えるソフトウェア開発の基本



Learn the real user story of how Mary satisfied her customers



Score big by using velocity to figure out how fast your team can produce

BANG!

Use test-driven development to avoid unsightly software



ソフトウェアづくり

✓ 実行**主体**

✓ ひと

✓ **プロセス**

✓ 「実行すること」の連続

✓ **アクティビティ**

✓ 話す、聞く、立つ、座る、歩く、考える、書く、
描く、キーボードを打つ...

プロセス、すなわち**実行**することと、その**実行主体**（つまり**人**）は**既に遍在**し実践され**続**けて**いる**。

つまり“プロセス”とはソフトウェアをつくっている活動そのものの、すなわちソフトウェアづくりである。

adaptability

transparency

simplicity

unity

VALUES

Agility is...

Strategy

charter

funding

Release

estimation

Iteration

retrospective

Daily

acceptance testing

Continuous

goals

vision

backlog

release plan

review

iteration plan

standup

Being Agile

Ruby



よるしく

お願いします

パソコン、 Wiki、XII

時を超えた創造の原則

Eto Kouichirou

江渡浩一郎

[著]



Java Apps Log Analysis

J2EE and Java Logs Mining Software Web Apps analysis, Download Now!

www.log-viewer.com/LogAnalysis

C-Max at Digikey

Search Our Huge Selection of Quality Electronic Components Here!

jp.digikey.com

SCRUM Project Management

Free 5-users pack painless agile project management

www.targetprocess.com

Ads by Google

The way to the timeless way of programming

Share Favorite Get File More...

時を超えたプログラミングの道への道



The way to the timeless way of programming

角谷 信太郎

日本Rubyの会 (株)永和システムマネジメント s-kakutani@esm.co.jp

KAKUTANI Shintaro; Nihon Ruby-no-kai; Eiwa System Management, Inc. Developers Summit 2009 [12-A-3]; 2009-02-12(Thu)



Shintaro Kakutani, 8 months ago

custom

Embed <div style="width:425px;text-a

4119 views, 10 favs, 7 embeds more stats

Related Presentations

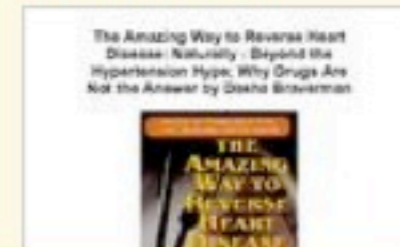
More by user



Intranet Pattern



Intranet Pattern



oDesk 22,000+ coders ready to start!



the 2001
leadership
awards

christopher alexander
grabs the
hall of fame
award

Christopher Alexander



The Nature of Order

価値宇宙の森羅万象
即ち生命にあらわれ
るいきいきとした秩
序の幾何学的特性と
それを生み出すプロ
セスについて

The Nature of Order

✓ Life is degree

✓ 構造:15の幾何学的特性

✓ プロセス:構造保存変換

✓ Structure Preserving Transformation

✓ Wholeness Extending Transformation



M83: The Thousand Ruby Galaxy

<http://apod.nasa.gov/apod/ap080927.html>

構造保存變換

- ✓ **Step-by-Step Adaptation**
- ✓ **Feedback & Correction**
- ✓ **Unpredictability**
- ✓ **Awareness of the Whole**

adaptability

transparency

simplicity

unity

VALUES

Agility is...

Strategy

charter

funding

Release

estimation

Iteration

retrospective

Daily

acceptance testing

Continuous

goals

vision

backlog

release plan

review

iteration plan

standup

Being Agile

パソコン、 Wiki、XII

時を超えた創造の原則

Eto Kouichirou

江渡浩一郎

[著]

The **Nature** of **Software**



Beautiful Code

ビューティフルコード

Andy Oram, Greg Wilson 編

Brian Kernighan, Jon Bentley, 著

まつもと ゆきひろ 他

久野 禎子、久野 靖 訳

O'REILLY®

オライリー・ジャパン

Nature of Software

1. 人とソフトウェアの
あいだに価値がある

2. “システム”全体を構成する

3. 変更に対応できることが
求められている

Nature of Software

1. **人**が**ソフトウェア**を
使ってみないとわからない
2. **ハード**, **ソフト**, **文書**, **運用**
3. 育てることと**技術的負債**

Green

グリーンコード

Robert C. Martin 著
花井志生 訳

Code

プログラミング

✓ **スキル**を備えた**人**の営み

✓ **世界観**を構築し、
世界をつくりだす

✓ **文字列**を並べるだけの
簡単な**お仕事**ではない

アジャイル ソフトウェア開発

Agile Software Development



Alistair Cockburn
The Agile Software Development Series

アリスター・コーバーン 著
株式会社テクノロジックアート 訳
長瀬 嘉秀 + 今野 睦 監訳

付録B:

Naur, Ehn, 武蔵

**“Programming as
Theory Building”**

-- Peter Naur

プログラミングとは、プログラマが直面している問題に対するある種の見方、つまり理論を形成または構築するようなアクティビティであると適切に見なされるべきである、と提案する。この提案は、プログラミングとはプログラムなどのテキストを産み出すことである、という一般的な考え方とは対照的である。

Agile Software Development

SECOND
EDITION

The Cooperative Game

Alistair Cockburn



コードにしただも
のとコードにし
なかつたところが
プログラミング

私は**受託開発**、即ち**経済的**
環境圧による**人工的**なソフト
ウェアづくりを**生業**にし
ているのだけれど、ソフト
ウェアを含む**価値宇宙**にお
いて**“自然”**にソフトウェア
は**存在**するのだろうか？

“自然な”ソフトウェア開発

✓ **Free/Open Source**
Software

✓ **Bazaar** Style Dev.

✓ **Hackers**

✓ **Culture & Inheritance**

Ruby



入門

Git

Git is a free & open source, **distributed version control system**, designed to handle everything from small to very large projects with **speed** and efficiency. Every Git clone is a full-fledged repository with complete **history** and full **revision tracking** capabilities, not dependent on

The Nature of Software Development

The Nature of Order

✓ Life is degree

✓ 構造:15の幾何学的特性

✓ プロセス:構造保存変換

✓ Structure Preserving Transformation

✓ Wholeness Extending Transformation

構造保存變換

- ✓ **Step-by-Step Adaptation**
- ✓ **Feedback & Correction**
- ✓ **Unpredictability**
- ✓ **Awareness of the Whole**

adaptability

transparency

simplicity

unity

VALUES

Agility is...

Strategy

charter

funding

Release

estimation

Iteration

retrospective

Daily

acceptance testing

Continuous

goals

vision

backlog

release plan

review

iteration plan

standup

Being Agile

開発が**デジタル**で
あるということは**自
然**なソフトウェアづ
くりにとれだけ近付
いているかの**度合い**

ag·ile /ædʒəl | -ail/

形容詞

1 敏捷 (びんしょう) な, 機敏な, すばしこい, はしこい (⇔awkward)

be (as) agile as a squirrel | リスのようにす早い.

2 活発な, いきいきとした (active) (⇔sluggish) ; 頭の切れる, 頭の回転が早い

an agile mind | 明敏な知力.

[ラテン語agilis (agereする + -ilis形容詞語尾=力をもってする→機敏な) . ΔAGENT]

ag·ile·ly

副詞

The Pragmatic Programmers
Practices of an Agile Developer
Working in the Real World

アジャイル プラクティス

達人プログラマに学ぶ現場開発者の習慣 

Venkat Subramaniam and Andy Hunt 著 / 角谷信太郎・木下史彦 監訳



才一公社

アジャイルな開発を実現するためには何よりもまず、ひとりの開発者としてアジャイル開発者としての心構えと習慣を身につけること。そのためには日々の現場で自分が感じる気持ちを大切にすること。これが本書のメッセージです。

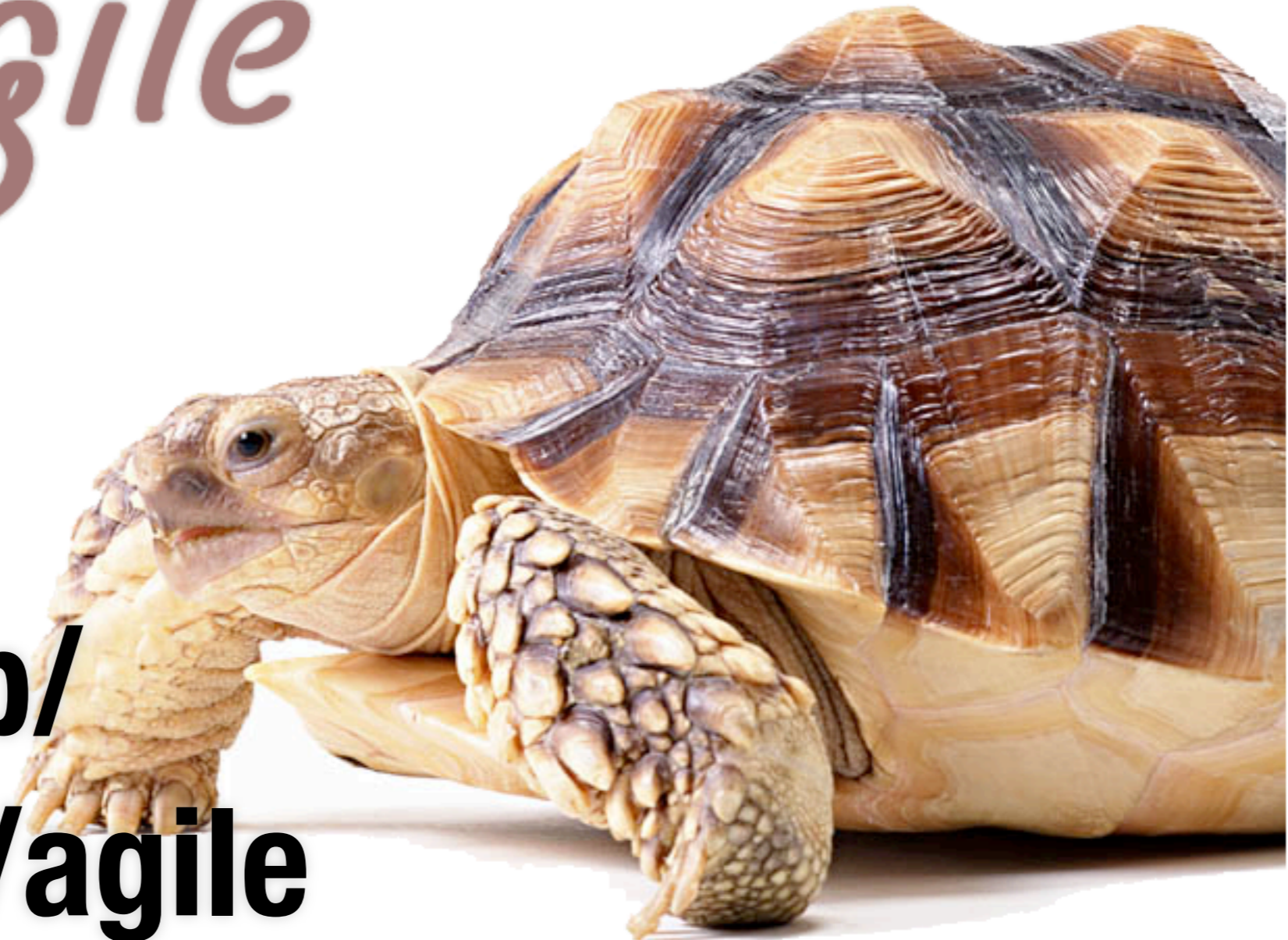
アジャイルな開発者たちが、アジャイルにチームで開発するからアジャイル開発

こうした本書の立場は「アジャイル (agile)」という単語が形容詞であることを考えれば納得できます。形容詞は単独では成立しません。後ろに修飾される言葉が続かねばなりません（例えば、アジャイル開発やアジャイルプロセス）。「アジャイルな開発」とは、開発の現場におけるさまざまな活動をアジャイルにしていく——変化に適応することを継続させていく、不断の営みです（単なる開発速度や納期の早さはアジャイル開発と関係ありません）。つまり「開発がアジャイルである」という状態はあっても「アジャイル開発」というものはありません。

アジャイル開発とは、アジャイルな開発者たちが、アジャイルにチームで開発するからアジャイル開発と呼べるのです。アジャイル開発の中心にいるのは人です。開発にたずさわる人が「プロフェッショナルとしての意識を持ち、自らの持てる力を最大限に発揮したいという意欲に満ち」ることなしに、アジャイルな開発の実現はありえません。

acts_as

agile



**[http://gihyo.jp/
dev/serial/01/agile](http://gihyo.jp/dev/serial/01/agile)**

アジャイルなソフトウェア開発

まず、連載の根幹に関わる「アジャイル」という（2007年現在では若干色褪（いろあ）せてしまった）バズワードを私なりに説明することから始めます。

アジャイルは形容詞

「アジャイル」という言葉は形容詞です。ですから、アジャイルソフトウェア開発とは「ソフトウェア開発がアジャイルである」という状態のことです。Chet HendricksonとRon Jeffriesは、この状態を「Early and Often（はやめに・こまめに）」という言葉で表現しています（注5）。

注5)

2004年9月に行った講演タイトルが「Early and Often」でした。残念ながら執筆時点ではWeb上にこの資料はありません。

参考：<http://capsctrl.que.jp/kdmsnr/diary/20040930.html>

はやめに・こまめに

「はやめに・こまめに」が実現されている状態とは、ソフトウェア開発

→ 充実の福利厚

Panasonic Excel S

技術者派遣のパナソニック

tech.excelstaff.jp

その他の連載

● 目指せ！ iPhone



2008年7月、
のユーザを獲
の1つは、オ

う。本連載では、iPhone
から公開の手順まで、

● 組込み教育委員会



組込みシステム
技術者の人材
す。本連載で

（ETSS）、トロン技
み技術者試験制度（
者資格試験プログラ
スキル向上に役立つ

● FLARToolKitを使



FLARToolKit
て、Flash上で
説します。

● Python 3.0 Ho

“Agile” is degree.

— Kakutani Shintaro

“アジャイル”とはプロセスがどれだけ
いきいきしているかを示す度合いである。

このスライドの英語がおかしいのはネタ(しかもAgileというものが“存在”しているかのように扱われていることへの揶揄)だったのだけれど、誤解を招いてしまった事実があるので、ここにつつしんで訂正いたします ×<

“*Agility*” is *degree*.

— Kakutani Shintaro

“アジャイルさ”とはプロセスがどれだけ
いきいきしているかを示す度合いである。

アジャイルなプロセス

- ✓ プロセス、つまり私たちの
“ソフトウェアづくり”の過程
がアジャイルであることの度
合いを言っている (形容詞)
- ✓ “アジャイルプロセス”という
モノは無い (名詞)

アート・オブ・

アジャイル デベロップメント

組織を成功に導く
エクストリームプログラミング

XPのプラクティスの数

- 1.ペアプログラミング
- 2.生き生きとした仕事
- 3.情報満載の仕事場
- 4.根本原因分析
- 5.ふりかえり
- 6.信頼
- 7.全員同席
- 8.真の顧客の参加
- 9.ユビキタス言語
10. スタンドアップミーティング
- 11.コーディング標準
- 12.イテレーションデモ
- 13.報告
- 14.「完全Done」
15. バグなし
16. バージョン管理
17. 10分ビルド
18. 継続的インテグレーション
19. コードの共同所有
20. ドキュメント
21. ビジョン
22. リリース計画
23. 計画ゲーム
24. リスク管理
25. イテレーション管理
26. ゆとり
27. ストーリー
28. 見積り
29. インクリメンタルな要件
30. 顧客テスト
31. テスト駆動開発
32. リファクタリング
33. シンプルな設計
34. インクリメンタルな設計とアーキテクチャ
35. スパイクソリューション
36. パフォーマンスの最適化
37. 探索的テスト

Head First

ソフトウェア開発

頭とからだで覚えるソフトウェア開発の基本



Learn the real user story of how Mary satisfied her customers



Score big by using velocity to figure out how fast your team can produce

BANG!

Use test-driven development to avoid unsightly software



プロセスとは、どのような図、文書、テストを実行すべきかに関する形式的な一連の規則というよりも…実は実行すべきことや実行すべきときを表すものにすぎないのです。また、頭文字も必要ありません…適切に機能すればよいのです。

『Head First ソフトウェア開発』



**自分のチームと自分のプロジェクトに役立つプロセスを選び…
そのプロセスが生み出した成果物を自分の顧客の要望に合うように調整します。**

『Head First ソフトウェア開発』

コードがソフトウェア
の中心なのであれば、**プログラマ**が
コードを書くのに選
ぶ**言語**も重要はず

Ruby



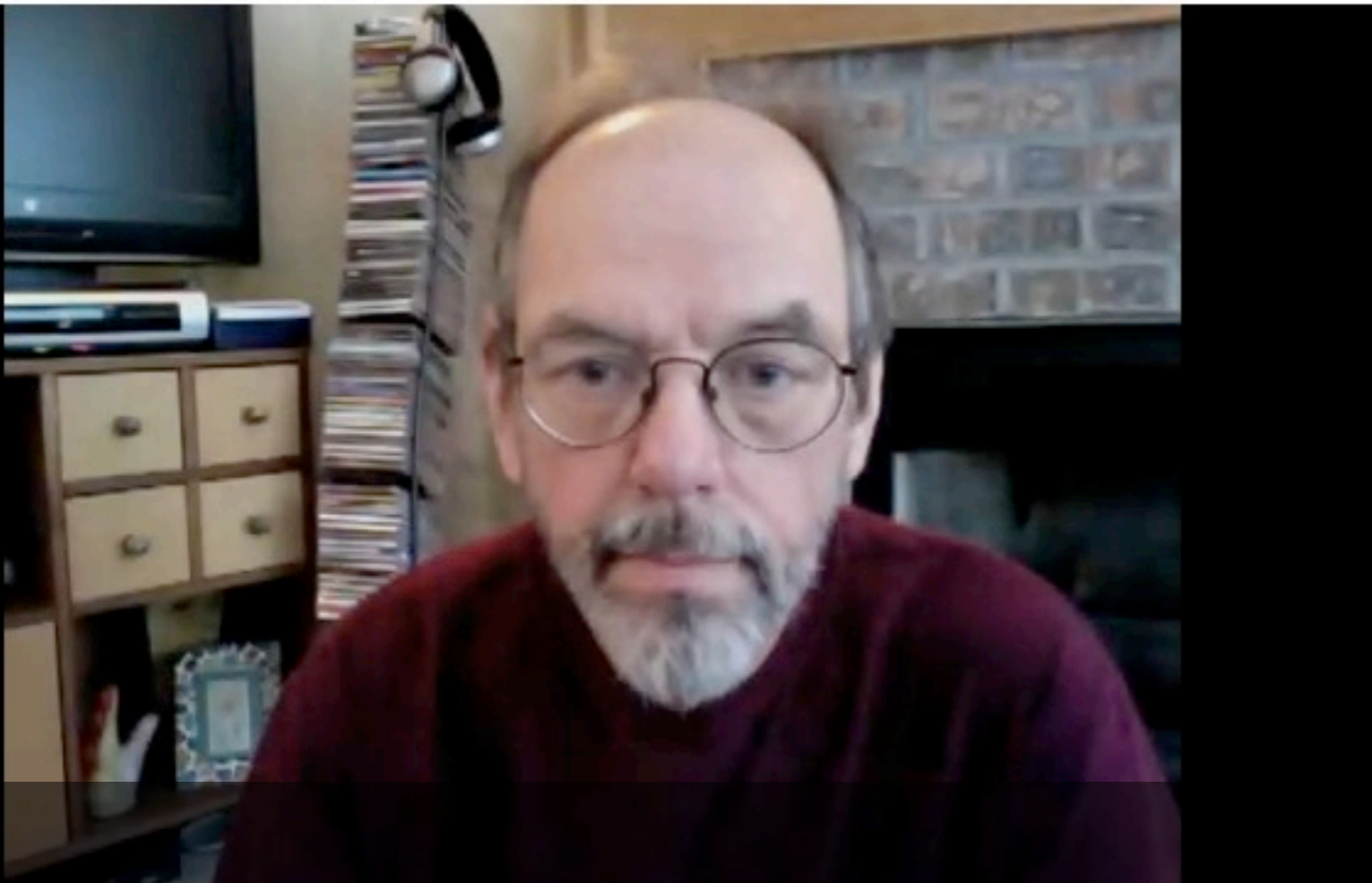
Martin Fowler
says:



Ward(Cunningham)、**Kent(Beck)**、それから、**キレイなコード**、うまく整理された**オブジェクト指向設計**、そして、**テストしやすさ**について提唱してきたすべての人たちの**価値**が…**Ruby**界では…『**正統**』(orthodoxy)になっている。

正統

Debt Metaphor



WardCunningham チャンネル登録
2009年02月14日 (詳細)
Ward Cunningham reflects on the history, motivation and common misunderstanding of the "debt metaphor" as motivation for refactoring.

URL <http://www.youtube.com/watch?v=pqeJFYw>

埋込み `<object width="425" height="344"><param`

▶ WardCunningham の他の動画

▼ 関連動画

- Money As Debt (1 of 5)**
再生回数 284,711 回
compelled2283
おすすめ動画
9:58
- Michael Mehaffy Lightning Interview**
再生回数 274 回
WardCunningham
5:43
- Wiki's Strengths**
再生回数 537 回
WardCunningham
0:47
- Jeffries & Hendrickson Lightning Interview**
再生回数 689 回
WardCunningham
7:53
- Ward Cunningham's Dial-a-Door Demo**
再生回数 370 回
caseorganic
2:21
- Elisabeth Hendrickson Lightning Interview**
再生回数 460 回
WardCunningham
5:21
- Ward Cunningham on Connectipedia**

Ward Cunningham

Agreed! Please watch "money as debt", and "money masters". Wake up America.

Kent Beck



Photo by Improve It:

<http://www.flickr.com/photos/improveit/1574023621/>

アジャイル ソフトウェア 開発の奥義

著●ロバート・C・マーチン
訳●瀬谷 啓介

Agile
Software
Development

2版

Green

グリーンコード

Robert C. Martin 著
花井志生 訳

Code

The RSpec Book

Behaviour Driven Development
with RSpec, Cucumber,
and Friends

```
class Html < Base
  def to_s
    init_header_index
    result = ""
    @language_elements.each_with_index do |language_element, index|
      result << format_header(language_element)
      result << format_body(language_element)
    end
    result
  end
end
```

```
class Html < Base
  def to_s
    init_header_index
    result = ""
    @language_elements.each_with_index do |language_element, index|
      result << format_header(language_element)
      result << format_body(language_element)
    end
    result
  end
end
```

```
class Definition < ActiveRecord::Base
  def to_s
    @language_elements.each_with_index do |language_element, index|
      result << format_header(language_element)
      result << format_body(language_element)
    end
    result
  end
end
```

Ward(Cunningham)、**Kent(Beck)**、それから、**キレイなコード**、うまく整理された**オブジェクト指向設計**、そして、**テストしやすさ**について提唱してきたすべての人たちの**価値**が…**Ruby**界では…『**正統**』(orthodoxy)になっている。



Ruby

A Programmer's Best Friend

Search

[ダウンロード](#) [ドキュメント](#) [ライブラリ](#) [コミュニティ](#) [コア開発](#) [ニュース](#) [セキュリティ](#) [Rubyとは](#)

Rubyとは...

オープンソースの動的なプログラミング言語で、シンプルさと高い生産性を備えています。エレガントな文法を持ち、自然に読み書きができます。

[もっと読む...](#)

```
# Ruby knows what you
# mean, even if you
# want to do math on
# an entire Array
cities = %w[ London
             Oslo
             Paris
             Amsterdam
             Berlin ]

visited = %w[Berlin Oslo]

puts "I still need " +
     "to visit the " +
     "following cities:",
     cities - visited
```



Download Ruby

はじめよう!

[ダウンロード](#)

[インストールガイド](#)

探求しよう!

[ドキュメント](#)

[ライブラリ](#)

2009年11月・12月開催の地域Ruby会議(Regional RubyKaigi)のお知らせ

日本Rubyの会が後援する、小さなRubyKaigiをいろんな地域でやっけてしまおうというプロジェクトである[地域Ruby会議\(Regional RubyKaigi\)](#)が下記の場所と日程で開催されます。

- 11/6(金),11/7(土): [関西Ruby会議02](#)
- 11/29(日): [TokyuRuby会議01](#)
- 12/5(土): [札幌Ruby会議02](#)

札幌Ruby会議02はすでに申し込みが始まっております。関西Ruby会議02は受付不要です。TokyuRuby会議01は11月中旬頃に申し込み開始予定です。

2010年以降もいくつかの地域Ruby会議の開催が予定されています。最新情報は[RubyKaigi日記](#)で都度お知らせしています。

Posted by KAKUTANI Shintaro on 03 Nov 2009

コミュニティに参加しよう

[メーリングリスト](#): 世界中のプログラマとRubyについて話しましょう。

[日本Rubyの会](#): Rubyの利用者/開発者の支援を目的としたグループです。

[更新順リンク](#): Ruby関連のサイトのリンクを更新順に並べたものです。

RubyForge トッププロジェクト

- [Ruby Installer for Windows](#)
- [RubyGems](#)
- [Instant Rails](#)
- [SQLite-Ruby](#)
- [typo](#)

[もっと読む...](#)

Syndicate

[最近のニュース \(RSS\)](#)

rubyist.netのサービス停止

<http://www.ruby-lang.org/ja/>

“Rubyをキメると
気持ちいい

まつもとゆきひろ - RubyKaigi2008

たのしい

高橋征義・後藤裕哉 著

Masayoshi Takahashi, Yuusaku Gotou

まつもとゆきひろ 監修

Yukihiro Matsumoto

Ruby

第2版

Rubyではじめる
気軽なプログラミング

入門

Git

Git is a free & open source, **distributed version control system**, designed to handle everything from small to very large projects with **speed** and efficiency. Every Git clone is a full-fledged repository with complete **history** and full **revision tracking** capabilities, not dependent on

“『入門Git』の書名は『The Git Way』であるべき

-- a_matsuda (Asakus.rb)

“The Git Way”の白眉

✓ ワークフローのために

ツールがある

✓ “Linus君の究極のコンテンツトラッキング”

✓ パッチベースのワークフロー

✓ 目にみえる手段の背後にあ

る考えかたの重要性



“
ホモ・サピエンスはパターン認識生物だ、
とパーカーボーイはいう。

それは才能でもあり、罨でもある。

——ウィリアム・ギブスン 『パターン・リコグニション』

入門

git

The Pragmatic Programmers

Pragmatic Version Control
Using Git

The Pragmatic Starter Kit
Volume I

Travis Swicegood 著 / でびあんぐる 監訳

2586



才一ム社

アジャイルな 見積りと計画づくり

価値あるソフトウェアを育てる概念と技法

MIKE COHN [著]

安井 力、角谷 信太郎 [翻訳]

AGILE
ESTIMATING AND
PLANNING

steps to phantasien

2009-10-18

最近もらった本 - アジャイルな見積りと計画づくり

(最近はうそです. もらってから半年くらいたってます....)

プロジェクト管理の本だった. 見積りの話も丁寧に書かれている. プロジェクト管理の二本柱(PMBOK による) "計画" と "監視制御" のうち, "計画" に占めるソフトウェア開発固有な部分は見積りくらいだから, 当然といえば当然かも. 前半 2/3 が計画(=見積り), 後半 1/3 が監視制御の話. 顧客に見積りや計画を求められる管理職には読んでおいて欲しい気がしたけれど, 管理職が読むのを期待するよりは自分で読んで見積りができるようにしておく方がたぶん現実的だし, 本書の趣旨にもあっていると思う.



本書の前半で説明されている見積りについて言えば, 規律にうるさいマコネルの見積りと比べ実施の敷居は低い気がするので, まずはこのコーン式見積りをやってみるのが良さそう. 私のいるチームも planning poker もどきをやってたりしている. 未熟なのでまだ精度は低いけれど, プレッシャーで見積りが縮むことは減ったとおもう. (そういえば "痛ポーカーがあればいいのに" という話を見積りのたびにしている. 本題とは関係ないけど誰かに作ってほしい.)

本書後半の監視と制御の話は, タスクボードやバーンダウンチャートなど Scrum 周辺で見かける話とかぶった. ただ他のアジャイルの本と違ってテストやリファクタリングなどコード寄りの話がなく, 自分の苦手な部分が抜き出されている感じで申しわけなくなりながら読んだ. 監視や制御について自分は素人だと再認識した.

面倒の取り分

世の中には PM という名の管理職制度があるせいで, 管理職でない下っ端が自分は管理をしなくて良いと誤解することがある. 基本的に管理の仕事は面倒だから, アジャイルになろうとすると必ずしも楽しいものではない.



**“With great power comes
great responsibility.”**

ハッカーならぬプログラマが身につけるべきスキル



EMERGENT DESIGN

The Evolutionary Nature of

Professional Software Development

創発的設計

テスト駆動開発

パターン駆動開発

ユニットテスト

リファクタリング

パターン

叡知

原則

プラクティス

病理学

コードの質



Beautiful Code

ビューティフルコード

Andy Oram, Greg Wilson 編

Brian Kernighan, Jon Bentley, 著

まつもと ゆきひろ 他

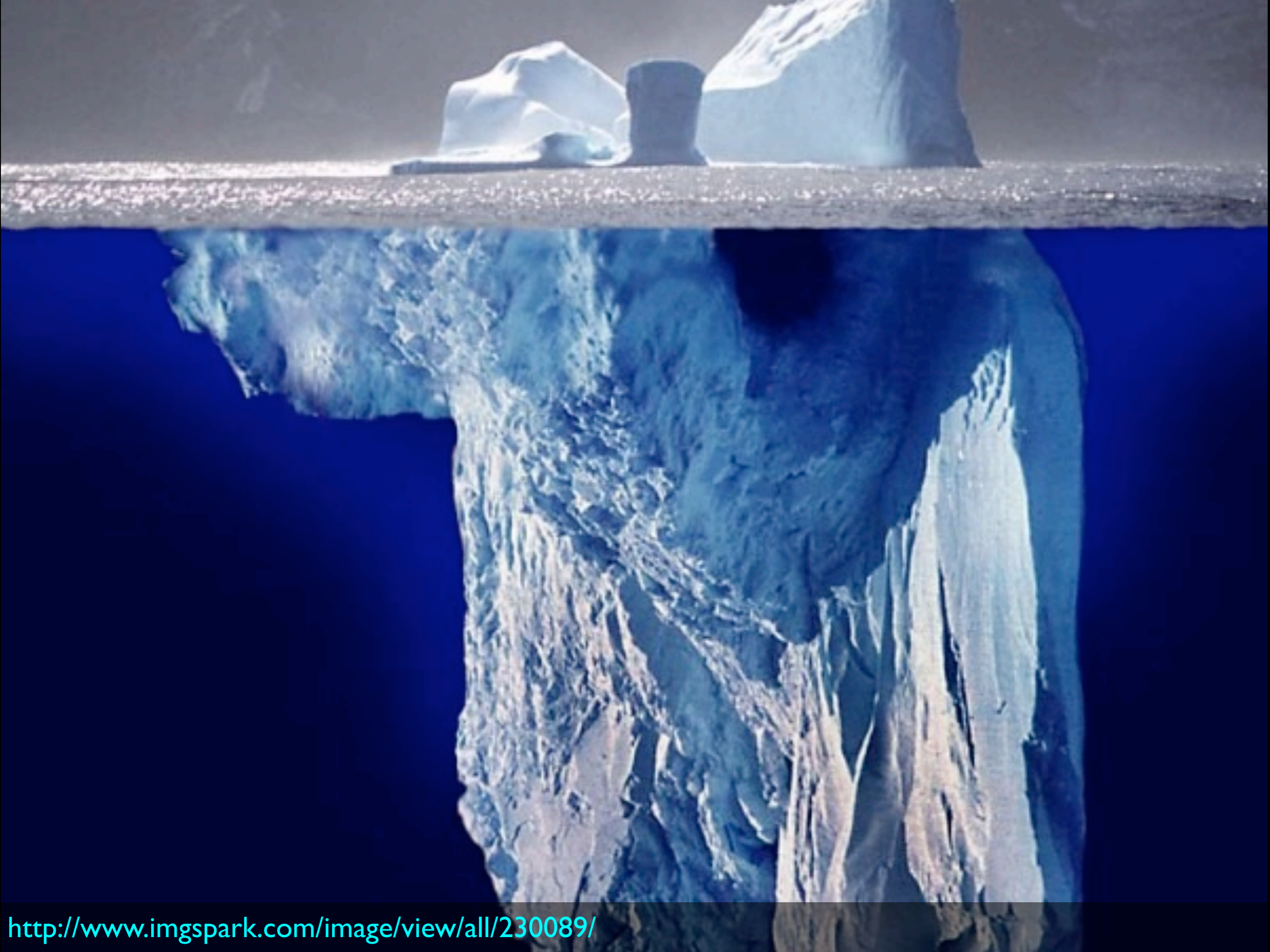
久野 禎子、久野 靖 訳

O'REILLY®

オライリー・ジャパン

“
プログラムを書いたことのない
システムエンジニアが威張って
いるような会社は早晚亡びる。

竹内郁雄 「推薦のことば」 (『ビューティフル・コード』)



“
ホモ・サピエンスはパターン認識生物だ、
とパーカーボーイはいう。

それは才能でもあり、罨でもある。

——ウィリアム・ギブスン 『パターン・リコグニション』



**“With great power comes
great responsibility.”**



GOT TO GET A GRIP!

提 供

情報化技術を通じて社会と共生する



永和システムマネジメント

✓ **受託開発**の会社です

✓ 本社は**福井**なので**関西**のお仕事も歓迎です :)

✓ **Ruby**のお仕事承ります

✓ **JRuby**での納品実績もあります

✓ **アジャイル**な開発のご支援

✓ 認定**Scrum**マスターが4人います

ご清聴ありがとうございました



M83: The Thousand Ruby Galaxy
<http://apod.nasa.gov/apod/ap080927.html>