

# MSR 2012 @ ICSE



---

## 目次

---

### Contents

- Mining Software Repository 2012 @ ICSE
  - MSR(MicroSoft Research) talk @  
MSR(Mining Software Repositories)
  - Towards Improving BTS with Game  
Mechanisms
  - GHTorrent

- The evolution of software
- Do Faster Releases Improve Software Quality?
- Security vs Performance Bugs in Firefox
- 思いつき
  - topicに基づいてcommitの分析と分割
  - 今回に皆使っているslideのシステム
  - マイクロソフトは腹黒っ子!

# Mining Software Repository 2012 @ ICSE

今年のMSRに参加しました、会場はチューリッヒ大学にあります。朝早く大学に着いて、登録するときちょっと事情がありました。スイス人は明らかに中国人の名前をわからないから、3つの中国からの楊（Yang）の名札を間違えた。そして堀田先輩の名札に"Japan, Japan"になって、日本代表になった。

# MSR(MicroSoft Research) talk @ MSR(Mining Software Repositories)

まず一番目のKeynoteはマイクロソフトアジア研究院(MicroSoft Research @ Asia ,MSR Asia)のZhang氏が発表する、こうしてMSRがMSRに発表するになった。

Zhangの発表はSoftware AnalysisとXIAOの2つの紹介です。XIAOはマイクロソフトが開発したCode Clone Detector、ある会社が私達に任せるのもこのようなシステムです。もっと詳しく知りたいが、実装に関わるものは言ってなかった。

## Towards Improving BTS with Game Mechanisms

この内容は基本的にこのブログに書いています：

<http://www.joelonsoftware.com/items/2008/09/15.html>

同じ理論をIssue Trackingとかに応用できるかを言いました。個人的にこれは 意味ない気がします。  
stackoverflowの成功はOpen Software Communityに

もと もとある名誉システムを具現化したですから、それを会社の中に応用するのは 難しい気がする。

# GHTorrent

---

この研究のスライドはこちら

に：<http://www.slideshare.net/gousiosg/ghtorrent-githubs-data-from-a-firehose-13184524>

Data exporter for github. Githubの主なデータはコード、それは既にgitから アクセスできます、wikiはgitとして保存しているからそれも含まれている。ですからこのプロジェクトの目的は他のデータを表せる、つまりissues, commit commentsなど。このプロジェクトはgithub apiを通じて、分布システムとして apiの制限を超える、そしてtorrentの形で歴史をdownloadできます。元のデータ はbsonとしてMongoDBの保存して、Schemaを追加したデータはMySQLに保存する。

わたしの意見では、データをgitのrepoの形で保存する方がいいかもしれない。今のwikiのように、そしてgitoliteも全てのデータをgit自身の中に保存している。

# The evolution of software

---

二日目のkeynotes, social mediaをソフトウェア開発に巻き込めるについて話しました。もしかしてこれはGithubの成功の理論かもしれない。IDEの中に social mediaのアクセスを欲しいと言いました。

## Do Faster Releases Improve Software Quality?

---

Firefoxを例として研究しました。

結論としては、早い発行はbugを多く持たされ、crashがもっと頻繁になるが、bugの修復も早くなって、そしてユーザー側はもっと早く新しい発行に移動することをわかりました。

## Security vs Performance Bugs in Firefox

---

性能に関するbugはregression テストが要る、そして発行を阻止する。

---

# 思いつき

---

## topicに基づいてcommitの分析と分割

---

よく使うツール（例えばgit）のユーザーはツールの設計者の意図を従って ツールを使うことはない、設計者が思った用途以外にも使っていることが多い、それはMiningに対しては色々困難を持たされています。例えばgitには完璧な branch機能がある、通常gitのユーザーが一つのcommitに一つの機能を実現してほしい、例としてはbugの修復とか、機能の追加とか。それは難しいなら branchを使って、一連のcommitを一つのbranchになって、一つのbranchに一つの機能を実現してほしい。それなのに、現状では、沢山の編集を一つのcommitに含まれていて、後の管理とか情報の収集とかが困難になってしまう。

それはユーザーの悪いと思わない、ツールの方がもっと頑張らないとユーザーは正しく使えない。もしcommitの時、自動的にcommitの内容を分析して、その中にtopicによって分けて、ユーザーに推薦するのをてきたらいいなあ、と思っています。このように一つのcommitを多くに分割したら、commitの履歴をもっと見やすくなって、続いて分析とかも便利になるはず です。

## 今回に皆使っているslideのシステム

---

タイトルは

Incorporating Version Histories in Information Retrieval  
Based Bug Localization

の人が使っているのはbeamerです。数式が多くて、overlayも多くて、iterateも多い、図だけ少ない、典型的にbeamerに作れそうなスライドです。mindmapの使いもうまい。今日の一日に少なくとも3個のslideはbeamerで作られています。

タイトルは

Towards Improving Bug Tracking Systems with Game Mechanisms

の人はpreziを使いました、図が多くて、transitionも多い。但しスライドとしては必要なページ数とかがなくて、このような国際会議の場合にはもっと工夫をした方がいいかもしれな。

少なくとも六人以上はAppleのKeynoteをつかてい  
まう。Keynoteによる作ったスライドはPowerpointの  
になかなか区別しがたいですが、その中に二人は  
defaultのthemeを使ったからわかります、他の人は  
PPTに決してありえないアニメーションを使っています  
から、多分keynote。

残りは勿論Powerpointです。MSRAの張さんが作ったのはpowerpointなんですけど、すごくbeamerの感じがします、例えばheaderとfooterの使い方とか、overlay 見たいのものでページのitemを一つずつ展開するとか。それらを全部powerpointで作るのは相当手間がかかりそうです。

ちなみに言いたいのは一つタイトルは

Green Mining: A Methodology of  
Relating Software Change to Power Consumption

のスライドは全部

下手

な手描きの漫画で表せている、火狐のアイコンさえ手描きする、効果は意外に評判がいい。省エネでグリーンで環境にいいで可愛らしい。具体的な効果は下のリンクから見えます、現場で見たのは別のバージョンなんですけど：

<http://softwareprocess.es/a/greenmining-presentation-at-queens-20120522.ogv>

マイクロソフトは腹黒っ子!



まあ大したニュースではないですけど、MSR2012のMining Challengeのスポンサー はマイクロソフトで、商品はXboxとKinectですけど、今年のチャレンジのテーマは：

Mining Android Bug

マイクロソフトの殺意を感じました。