软件微过程的挖掘和分析
-- Digital Archeology

周明辉
zhmh@pku.edu.cn
https://sei.pku.edu.cn/~zhmh
软件研究所
北京大学

背景

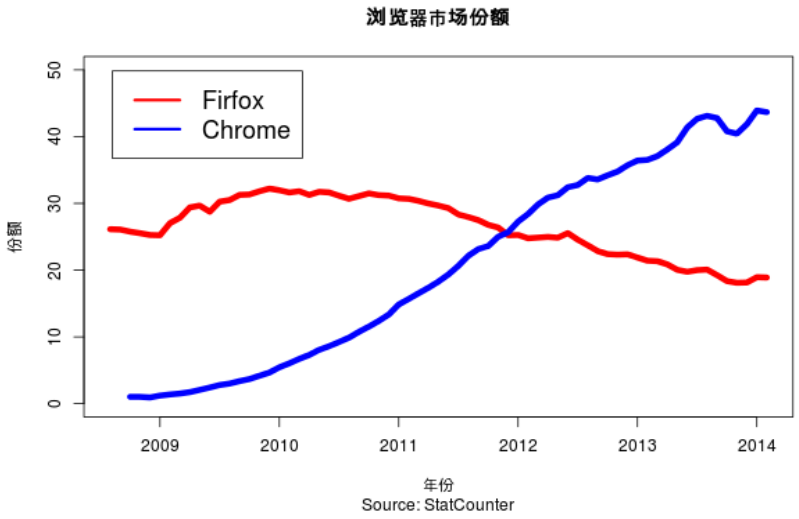
- 动机：理解软件是怎么开发的
- 目标：提高软件生产效率和软件质量
  - 流水线过程：需求-〉设计 -〉实现
  - 构件化组装方法：产品线工程
  - 高级语言
  - 开发工具
  - ……

已有许多方法，存在一些问题

- 过程，描述开发遵循的步骤
  - 程序员会根据自我经验（隐式知识）进行调适
  - 过程是模版，需要根据特定场景实现和定制
- 实验，度量因素影响
  - 能够处理的因素较少，但软件开发的影响因素繁多且复杂交织
  - 实验获得的结果难以应用到不同的场景
  - 企业环境下的受控实验非常昂贵

项目的成功经验难以复制

- Chrome成功于快速发布
- Firefox 学习Chrome，然而严重影响错误处理效率和用户体验

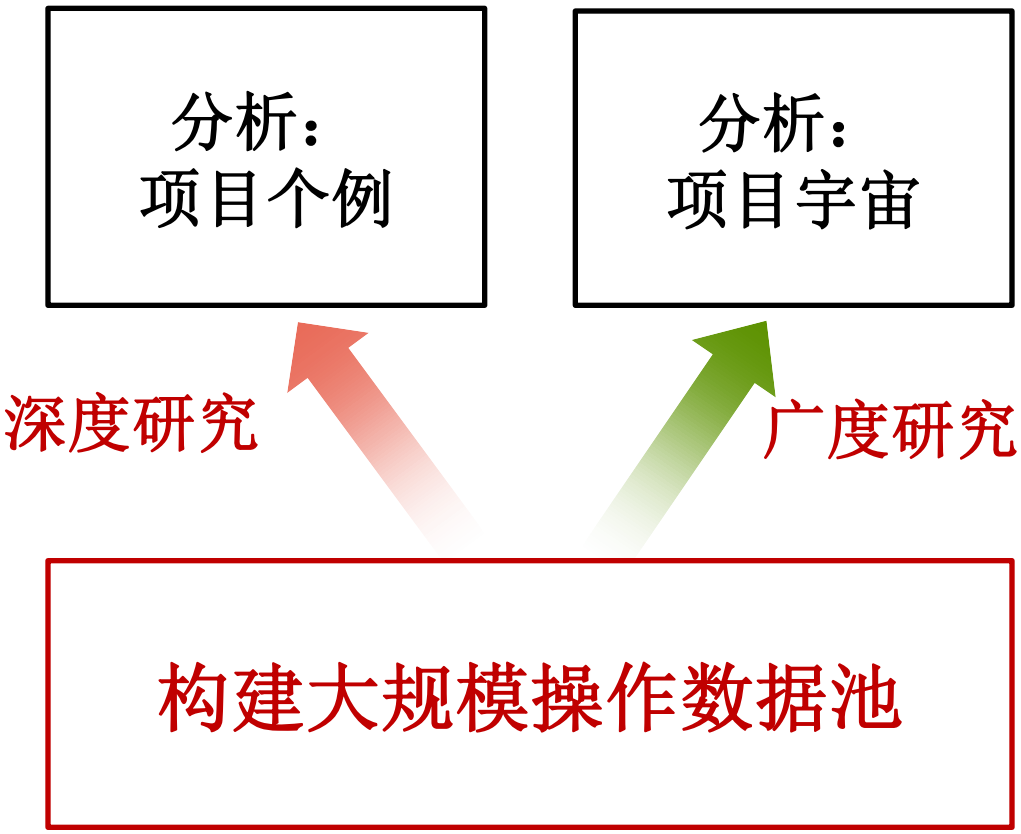


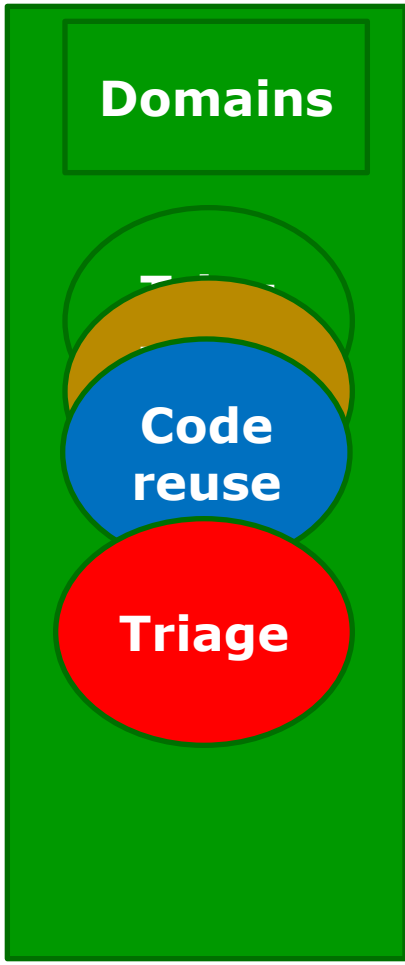
- Linus Law：足够多双眼睛，错误将无处遁形
- OpenSSL 出现重大安全漏洞

软件微过程及其优势

- 什么是微过程？
  - 个体或项目在完成基本开发任务时，所采用的可重复的活动模式（例如解决缺陷，合并代码，沟通需求，指导新手等） -- repeatable activity pattern
- 好处？
  - 反映人们的实际开发行为，
    - 能捕捉隐式知识
    - 能包含所有的上下文变化性
  - 不需要昂贵实验，
    - 使用操作数据（operational data）
  - 在细粒度度量软件开发

路线图： 发现和利用微过程





Minghui Zhou and Audris Mockus. Mining Micro-practices from Operational Data. FSE 2014 Visions and Challenges Track.

What is Operational Data?

- Digital traces produced in the regular course of work or play (i.e., data generated or managed by operational support tools)
  - No carefully designed measurement system
  - Different from traditional data science

Data Science and Operational Data

- Data science: The study of the generalizable extraction of knowledge from data
  - Goal: Laws of extracting knowledge from data?
  - ➢ What properties of data make it Data Science?
    - science extracts knowledge from experiment data
- Operational Data (OD): data generated or managed by operational support tools
  - ➢ no carefully designed measurement system

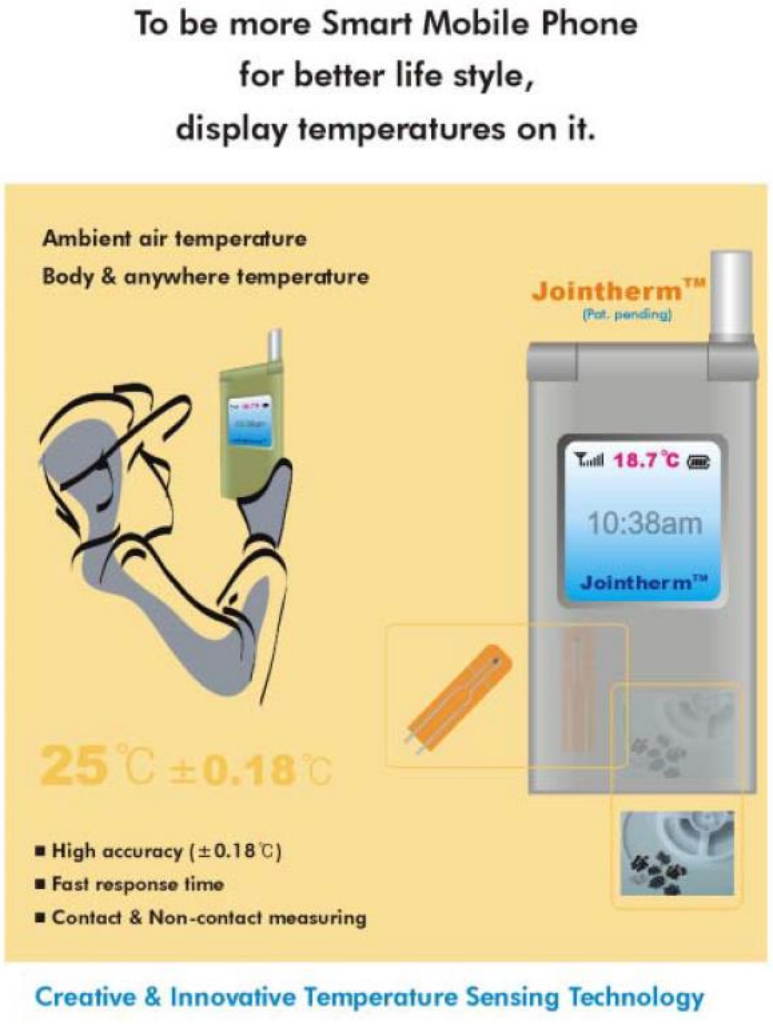
Audris Mockus: Engineering "Big Data" Solutions. ICSE'2014

Experimental data, 实验数据

- 例如，气温度量
  - 天气预报中所说的气温，指在野外空气流通、不受太阳直射下测得的空气温度
    - 一般在地面1.5米高度百叶箱内测定
    - 中国用摄氏温标（℃摄氏度）
    - 一般一天观测4次，分别为02、08、14、20四个时次
  - 标准的度量设计
    - 标准温度计，标准部署，固定气象站，标准时间点观测

Operational data (OD), 操作数据

- 利用移动手机度量气温
  - 没有上下文：室内/室外/汽车里/是否开空调
  - 并非所有位置、所有时间都被覆盖
- 需要发现Data Law,例如,
  - 温度是如何影响传感器的？
  - 如何识别室内/室外等？



Software Tools that Generate OD

- Version control systems (VCS)
  - SCCS, CVS, ClearCase, SVN, Bazaar, Mercurial, Git
- Issue tracking and customer relationship management
  - Bugzilla, JIRA, ClearQuest, Siebel
- Code editing (Eclipse), communication (Twitter), documentation (StackOverow), . . .

Example OD: Version Control Data

Developers use VCS to make changes to code (in parallel)

Traces Left by VCS

Code Before

```
int i = n;
while (i--)
    printf (" %d", i);
```

Code After

```
//print n integers iff n≥ 0
int i = n;
while (--i > 0)
    printf (" %d", i);
```

one line deleted two lines added two lines unchanged

- date: 2014-10-16 01:25:30,
- developer id: minghui,
- branch: master, Comment: \Fix bug 3987 - infinite loop if n<=0"

Example OD: issue tracking data

- GNOME issue tracking

Who	When	What	Removed	Added
chpe@gnome.org	2009-11-16 12:55:22 UTC	Blocks		138020
mclasen@redhat.com	2009-11-29 23:11:41 UTC	CC		jhs@gnome.org, mclasen@redhat.com
jhs@gnome.org	2009-12-02 09:20:36 UTC	Status	UNCONFIRMED	NEW
		Ever confirmed	0	1
jhs@gnome.org	2009-12-02 10:53:07 UTC	Attachment #148892 Attachment is obsolete	0	1
mclasen@redhat.com	2010-01-18 06:24:40 UTC	Status	NEW	RESOLVED
		Resolution		FIXED

总之，软件操作数据 (OD)

- 记录着开发的数字轨迹
  - 每一次代码提交、每一个缺陷报告、每一个邮件等，都被保存在软件支持工具中，
    - 例如，问题追踪系统，版本控制系统
  - 记录了软件开发过程和代码的演变、
  - 以及开发者个体及其交互的行为

软件操作数据

Issue Data
- Gnome (517793), mozilla (620479), jboss (93149), apache (228870)

Email Archives
- Gnome (1343372), jboss(1052291), apache (6323416)

Version control data
- gnome, mozilla, jboss, apache, netbeans, openjdk, opensolaris, symbian, andriod, debian, github, gitkernel, gitorious, savannash, repo.or.cz

Forge	Type	Files	File/Ver.	Unique File/Ver.	From
Large cmpny.	Var.	3,272K	12,585K	4,293K	1988
SourceForge	CVS	**26,095K**	81,239K	**39,550K**	1998
code.google	SVN	5,675K	14,368K	8,584K	1996
repo.or.cz	Git	2,519K	11,068K	5,115K	1986
Savannah	CVS	852K	3,623K	2,345K	**1985**
git.kernel.org	Git	12,974K	**97,585K**	856K	1988
OpenSolaris	Hg	77K	1,108K	91K	2003
FreeBSD	CVS	196K	360K	75K	1993
Kde	SVN	2,645K	10,162K	527K	1997
gnome.org	SVN	1,284K	3,981K	1,412K	1997
Gcc	SVN	3,758K	4,803K	395K	1989
Eclipse	CVS	729K	2,127K	575K	2001
OpenJDK	Hg	32K	747K	60K	2008

理解过去
预测未来
推荐最佳实践

软件操作数据 - 规模

Gnome有70多万个缺陷数据

每天有上百个新缺陷被提出

Apache有60多万封邮件

每天有1千多封新邮件产生

Mozilla有2亿多条代码提交

每天有2万多条新的代码提交

- Github.com拥有超过1.2千万的项目
- SourceForge.net和GoogleCode分别有超过30万项目
- Internet上所有开源项目的数据总量估计在300T

构建大规模数据池

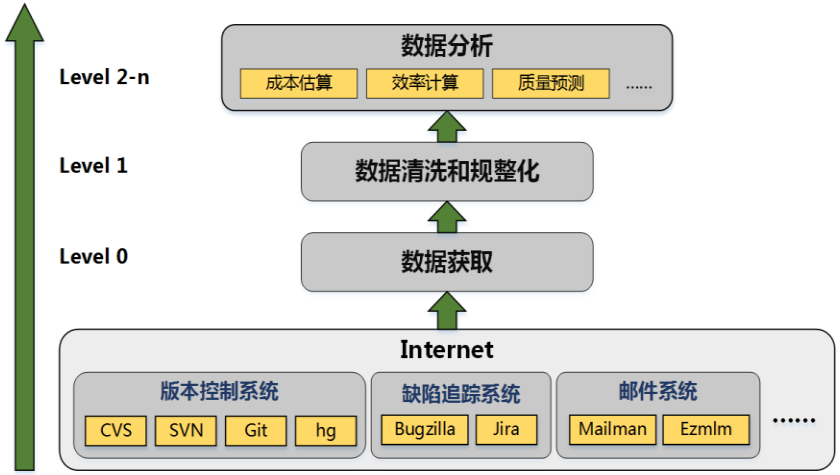
- Retrieve data from Internet
  - The various types of repositories
    - Cvs/svn/git/hg for VCS; jira/bugzilla for ITS
  - the project administrator's policies
    - banning the IP addresses that do the data retrieval
  - the network bandwidth
  - the huge amount of changes, issues in a project
    - GitHub has more than 12 million repositories
- Standardize data
  - It is a lot of work to extract the raw data from the operational support tools and to standardize into formats convenient for analysis

人们的尝试

- FOSSMole, Sonar, …
- 我们的尝试：
  - 收集，规整化和开源数据

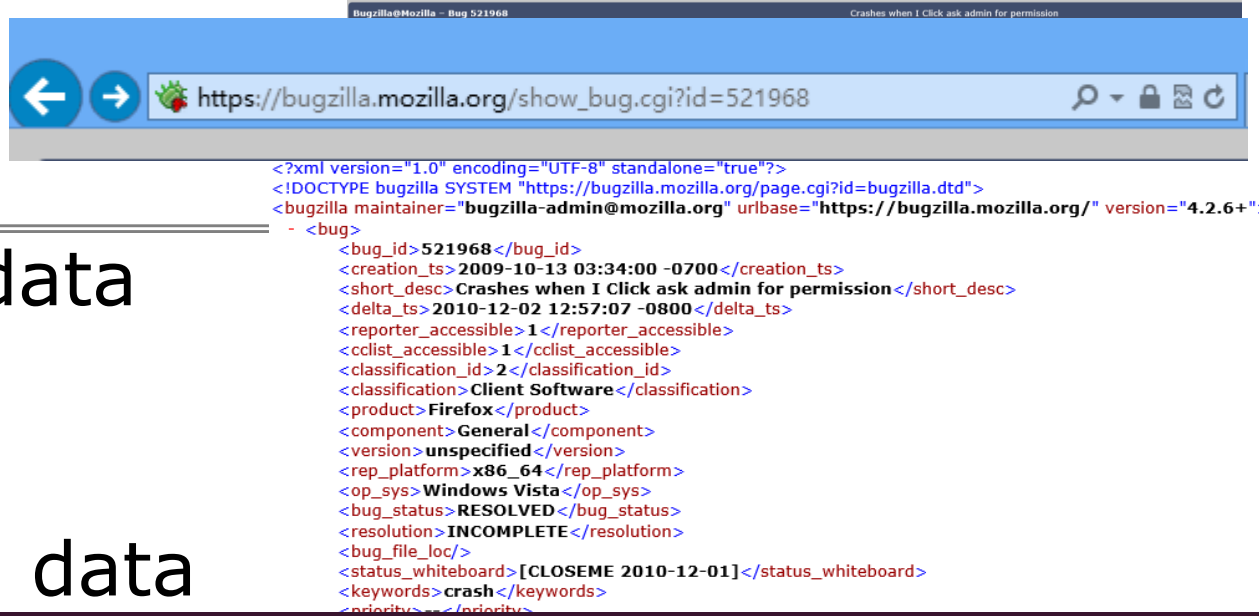
    https://passion-lab.org
  - 可视化项目基本信息和基础量度

Passion-Lab
Software Enginnering Institute
Peking University
Data Center



- DELL R910(4U), 16-cores X7550, 64GbRAM, 500G*12 SAS
- DELL MD3200, 12*2TB SAS
- DELL MD3600, 12*4TB SAS
- DELL R710 * 4, E5506(2.13G)*2, 64GbRAM, 2T SAS (+8T)
- DELL R720XD, 8-cores Xeon E5-2670(2.6G)*2, 384GbRAM, 2*300G+10*3T

数据层次



- Level0: raw data

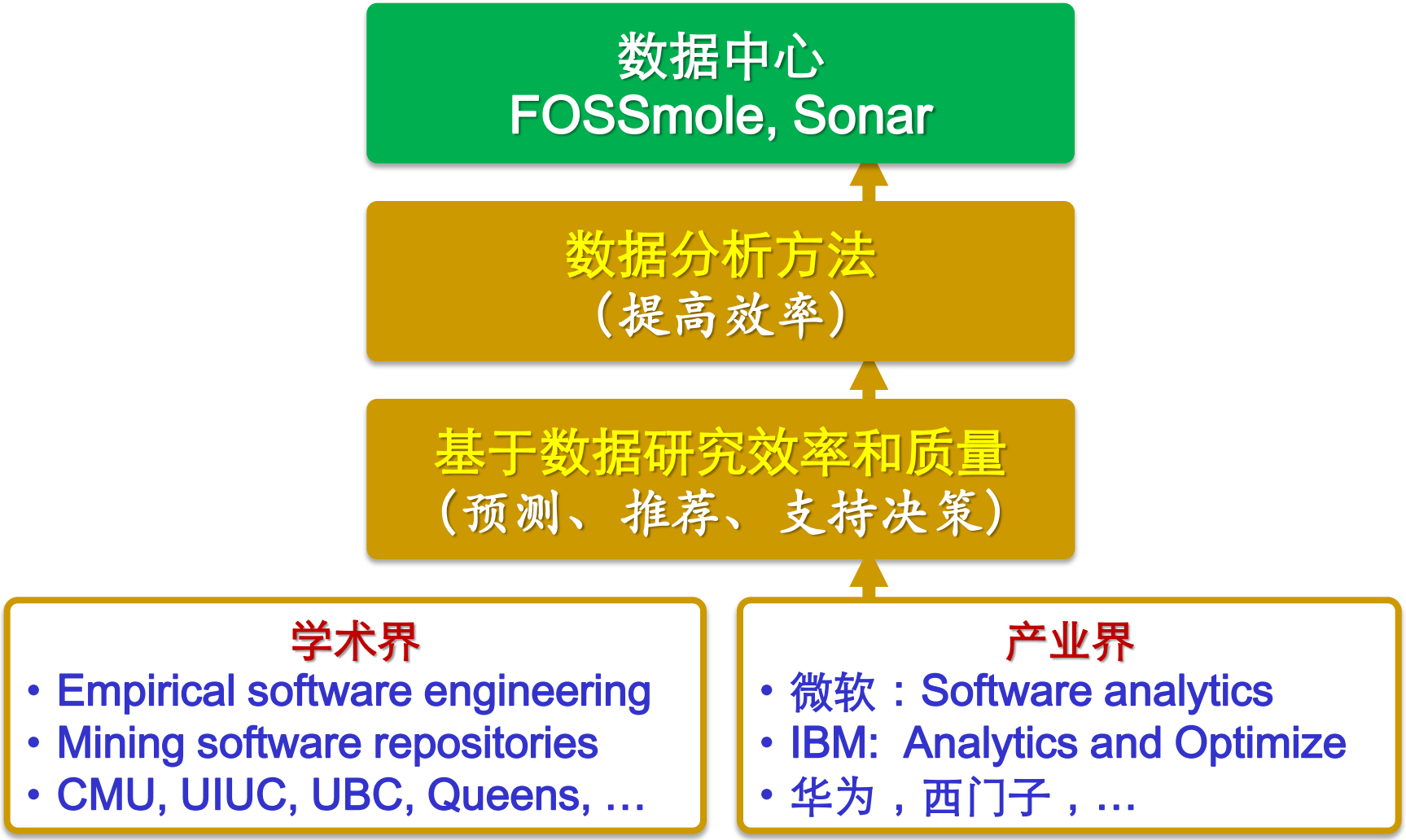
- Level1:
  - normalized data

```
pae:/store/bug/mozilla/20110328>head -1 info_level1
1272;Bug#=35;assigned_to=mcafee@mocha.com;assigned_to_name=Chris McAfee;bug_severity=minor;bug_status=VERIFIED;cc=tymerkaev@gmail.com:wlevin
tion_id=6;component=XFE;creation_ts=891887820;delta_ts=1291783737;everconfirmed=1;long:1:bug_when=1998-04-07 01:37:03 -0700;long:1:commentid=
ril 6, 1998 6:37:03 PM PDTAdditional Details :PREF_Cleanup is not called when Navigator exits(File-&gt Exit), causing memory leaks.Updated b
8:16 PM PDTUpdated by Sarah Wilson (swilson@netscape.com) on Tuesday, April 7, 1998 6:49:27 PM PDT;long:1:who=weitsang@cs.cornell.edu;long:1
id=2;long:2:text=leaks on exit are low priority  yes we should fix this.;long:2:who=mcafee@mocha.com;long:2:who_name=Chris McAfee;long:3:bug
in new world.;long:3:who=mcafee@mocha.com;long:3:who_name=Chris McAfee;long:4:bug_when=1999-02-26 12:55:59 -0800;long:4:commentid=4;long:4:t
ape.com.tld;long:4:who_name=;long:5:bug_when=2002-06-08 13:22:21 -0700;long:5:commentid=1388133;long:5:text=Not at all related to this bug.W
g no 1 ?.Is there a bug earlier than this one ?.Just for Trivia sake;long:5:who=Lee.Jnk@gmail.com;long:5:who_name=Lee;long:6:bug_when=2004-0
to comment #4)&gt  Not at all related to this bug.&gt  &gt  Which is the first ever bug to be submited at bugzilla ?.&gt  Where is bug no 1
sakeI think that this is the first ever bug here.respect!Dan-Shk;long:6:who=dan-shk@bezeqint.net;long:6:who_name=daniel;op_sys=Solaris;prior
.cornell.edu;reporter_accessible=1;resolution=WONTFIX;short_desc=Navigator does not free preference hash table when exit.;target_milestone=-
pae:/store/bug/mozilla/20110328>
```

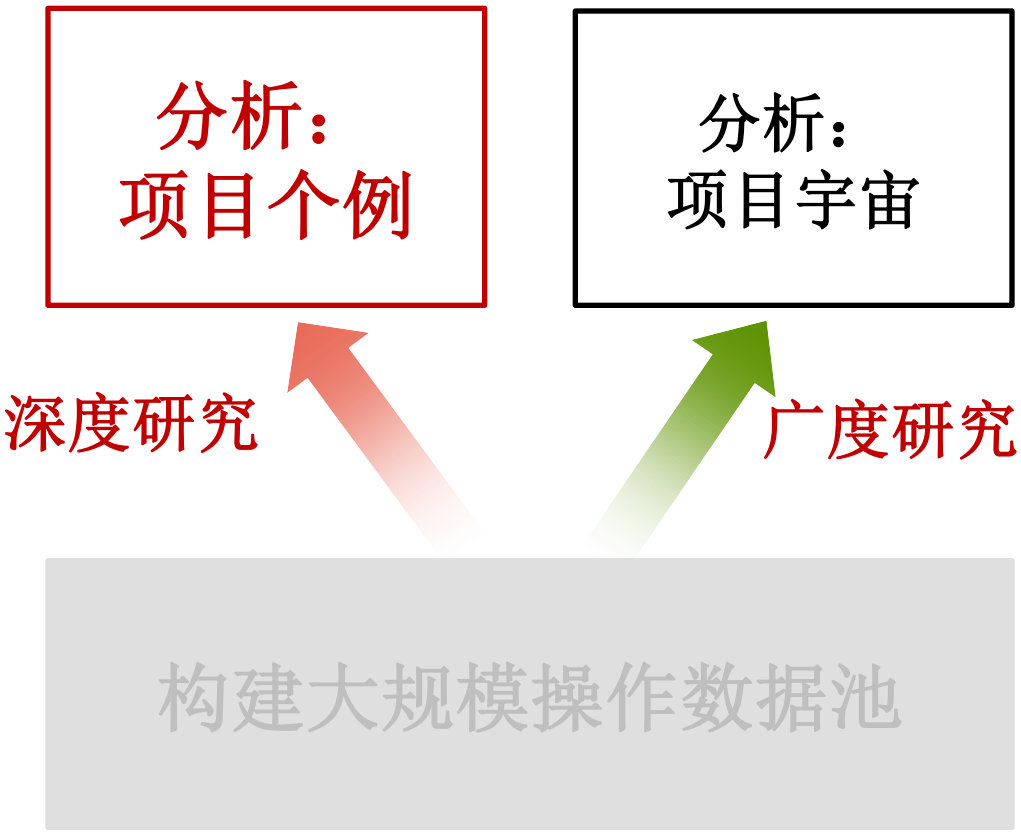
- Level2-n:
  - standardized data

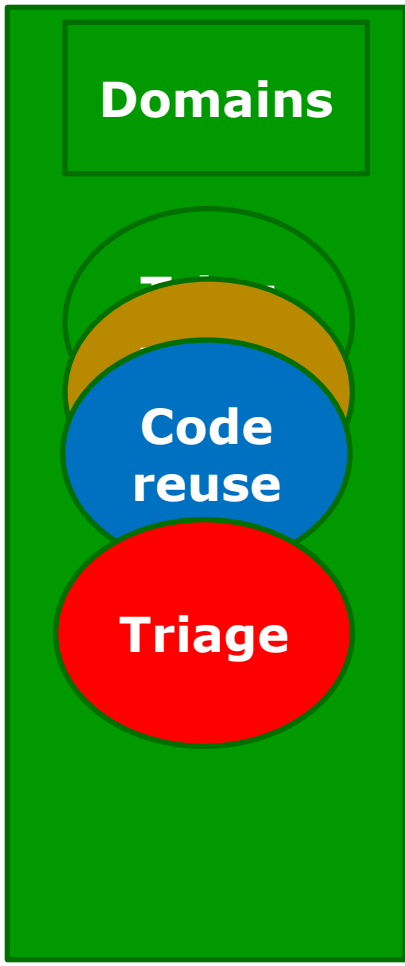
```
pae:/store/bug/mozilla/20110328>
pae:/store/bug/mozilla/20110328>
pae:/store/bug/mozilla/20110328>head -1 info_level2
35;-1;weitsang@cs.cornell.edu;891887820;reporter;VERIFIED;WONTFIX
pae:/store/bug/mozilla/20110328>
pae:/store/bug/mozilla/20110328>
```

数据驱动的软件工程：现状



路线图： 发现和利用微过程



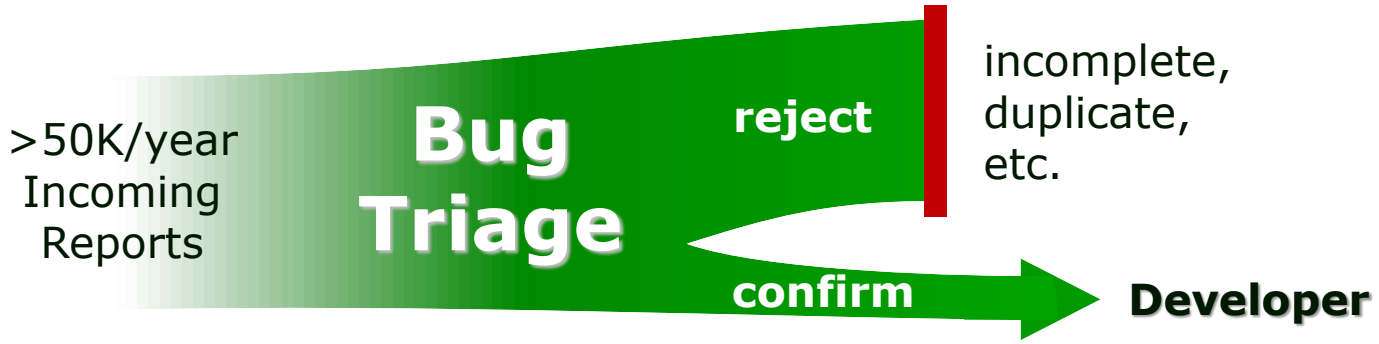


从一个特定问题开始:
什么是缺陷分类(BUG TRIAGE)的微过程?

Mozilla和Gnome的triage微过程

- 方法
  - 对Mozilla和Gnome做深度分析
  - 从Bugzilla数据来度量issue workflow

- 初始发现
  - Developer teams are overwhelmed by the massive inflow of low quality issues
  - Many more non-developers than developers

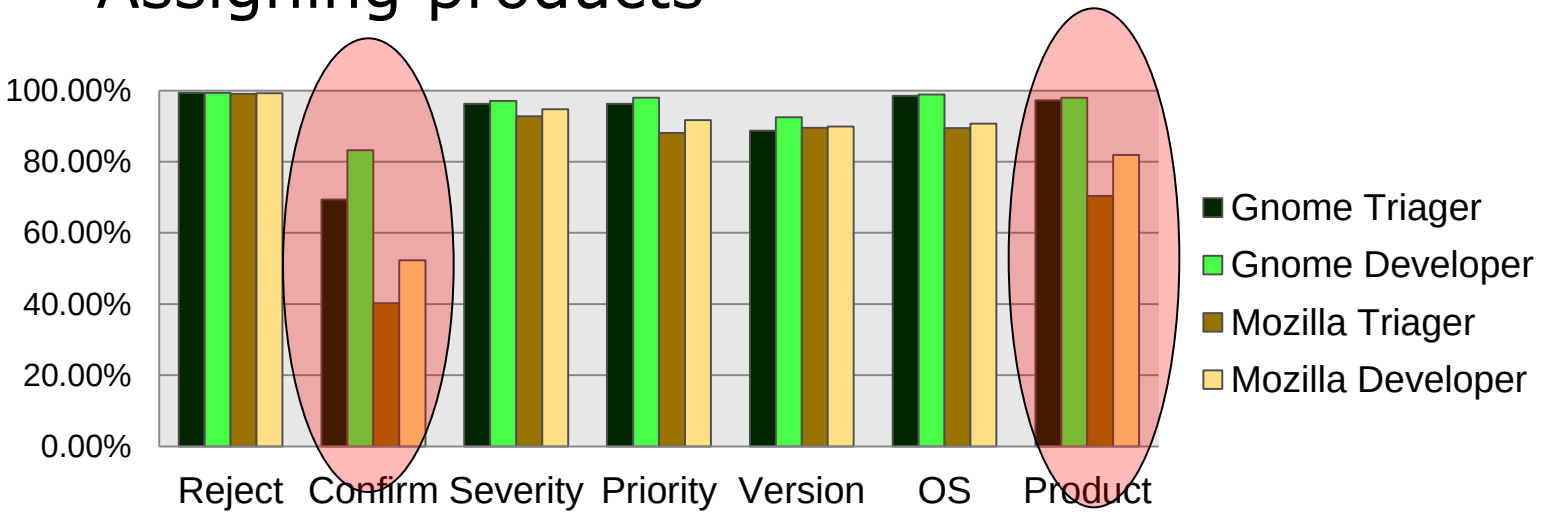


Mozilla和Gnome的triage微过程

- Triage任务: filter, complete, assign
- 挖掘微过程的方法
  - Data vs interviews
  - Official workflow vs observed transitions
  - Unusual time trends
- 微过程的例子
  - Pick recent issues, pick issues on specific topic
  - Correct OS version, Product, other fields
  - Developer opens issue as NEW (not UNCONFIRMED)
  - Auto-close massive number of issues
  - Correctness of a value is confirmed by a subsequent activity

使用发现的微过程度量triage

- Measure accuracy of triage tasks
  - Non-developers are slightly less accurate at
    - Confirming issues
    - Assigning products



Accuracy of Issues by Triage Task and Role

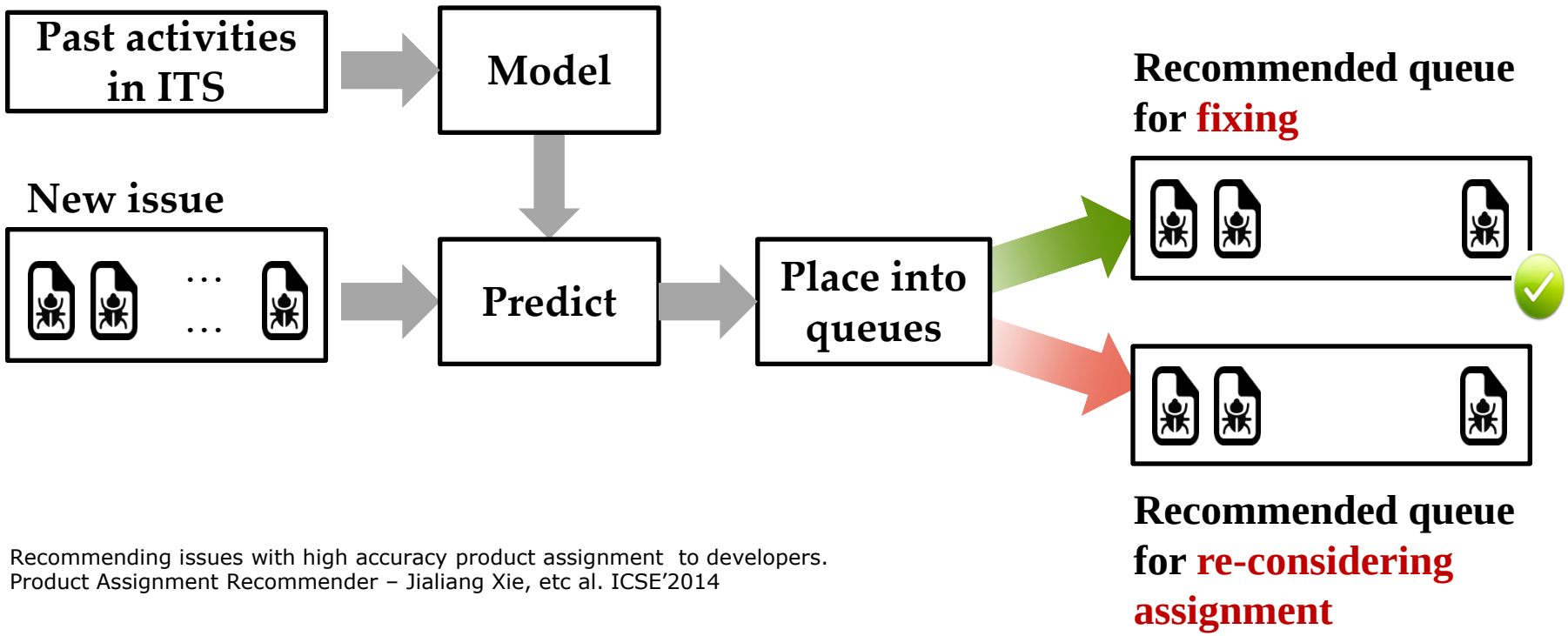
Understanding the impact of triage in open source software projects.
Impact of Triage: a Study of Mozilla and Gnome – Jialiang Xie, etc al. ESEM'2013

既然non-developer经常将issue分配到错误的product，因此，

PRODUCT ASSIGNMENT RECOMMENDER

Product Assignment Recommender

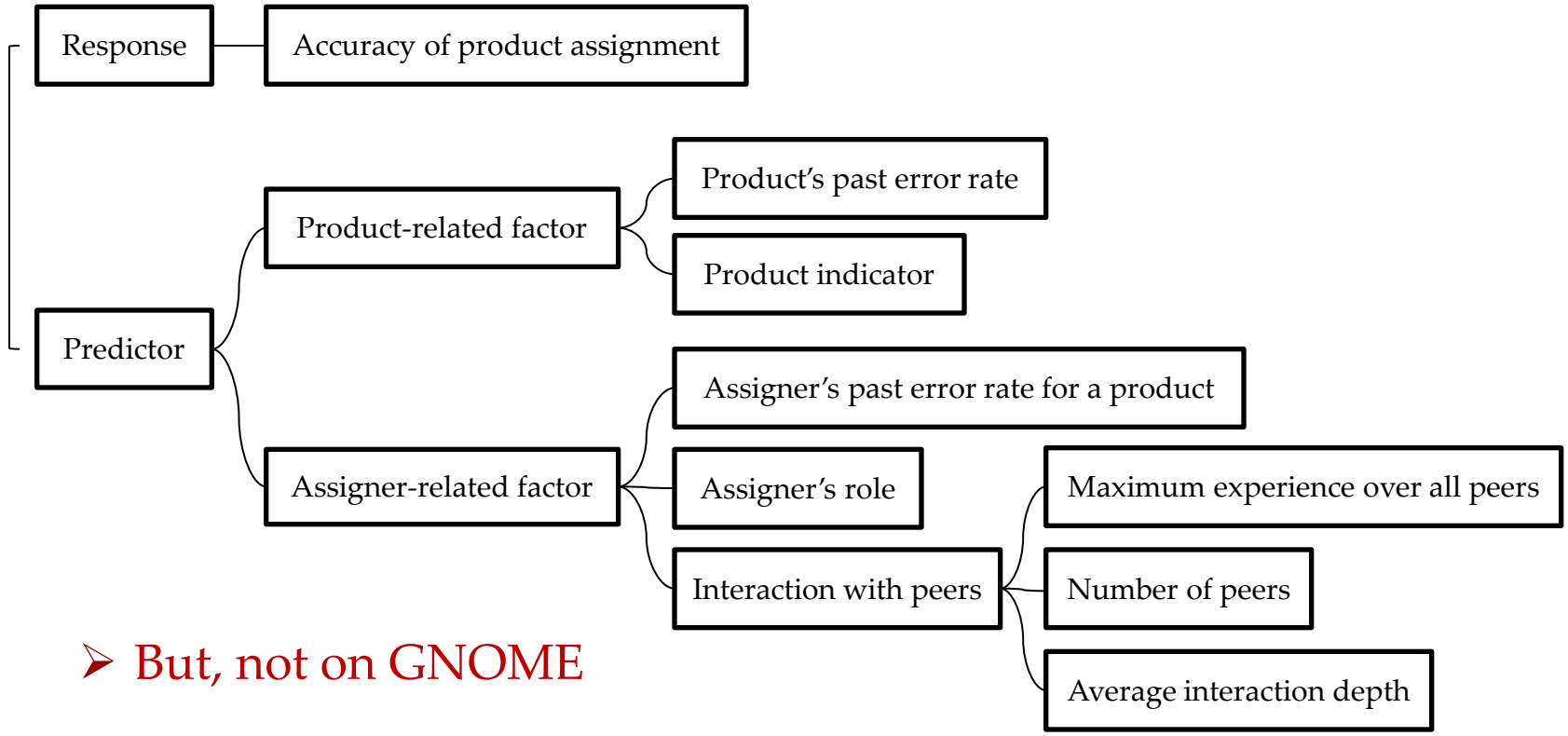
- Model the accuracy of product assignment
- Predict the accuracy of product assignment for the new issue
- Place the issue into queues according to its accuracy



Recommending issues with high accuracy product assignment to developers.
Product Assignment Recommender – Jialiang Xie, etc al. ICSE'2014

Approach

- Model the accuracy of product assignment
  - Build linear regression model



  - But, not on GNOME

Data Source: Mozilla Bugzilla Time Span: 2001-2011

PAR

Product Assignment Recommender

BugID Predict Set Threshold

Recommending Issues

BugID	Product
627020	Calendar
627426	Calendar

Warning Issues

BugID	Product
644300	Mozilla Labs
664115	Websites

Data Source: Mozilla Bugzilla | Time Span: 2001-2011

BugID: 644440

Product: nss

Actor: alvolkov.bgs@gmail.com

Metrics Calculated

Prediction:

Product

Product's Error Rate	0.07

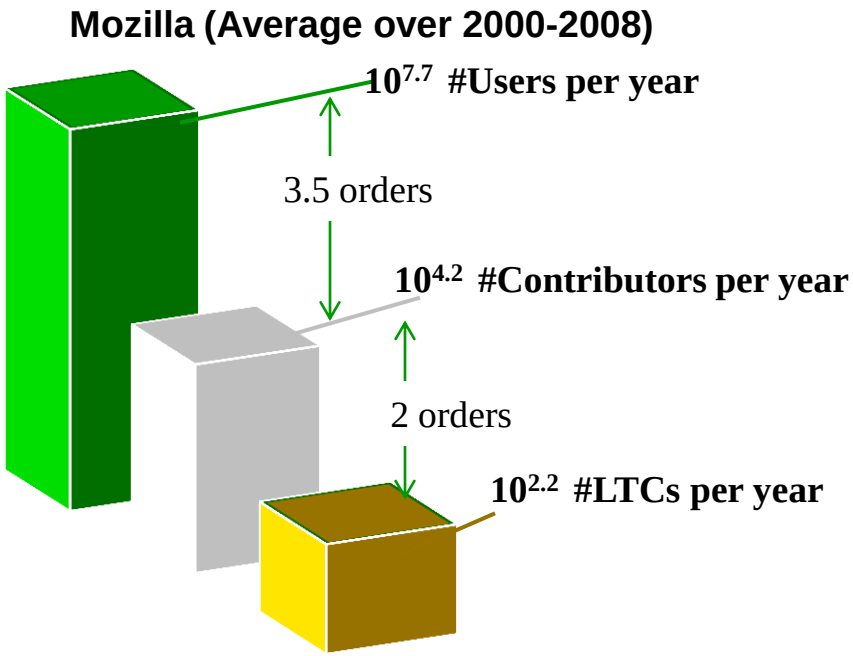
Actor

Actor's Error Rate for The Product	0.00
Maximum Experience over All Peers	1
Number of Actor's Peers	3
Average Social Depth	452.00
Actor's Role	triager

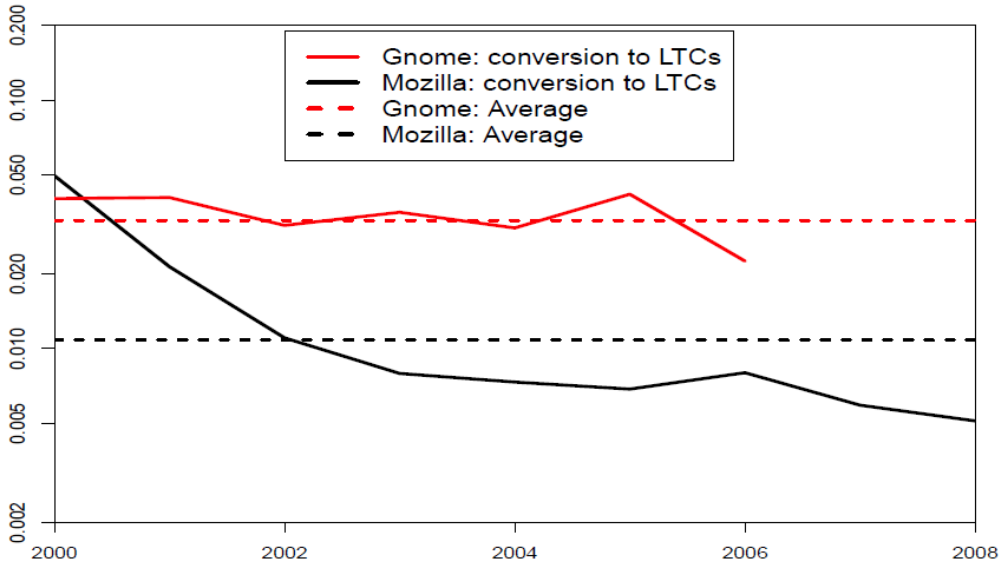
另一个例子：
什么是项目新进人员的微过程?
如何影响其成为长期贡献者?
LONG TERM CONTRIBUTOR

为什么研究这个问题?

- 开源社区中，LTC是项目生存的关键要素
- 然而：



每年用户：千万级别
每年新的参与人数：万
每年成为长期贡献者的人数: 百



Gnome和Mozilla中
Newcomer成为LTC的比率逐年降低

方法：基于Bugzilla Issue Workflow

Brant@gurganus.name

2002/04/02

Report

Wolruf@gmail.com

2002/04/02…

Modify OS

bugzilla@iwaruna.com

…

Comment

Mozilla-06@oliverklee.de

…

Change Status

bugzilla@iwaruna.com

…

Change Status

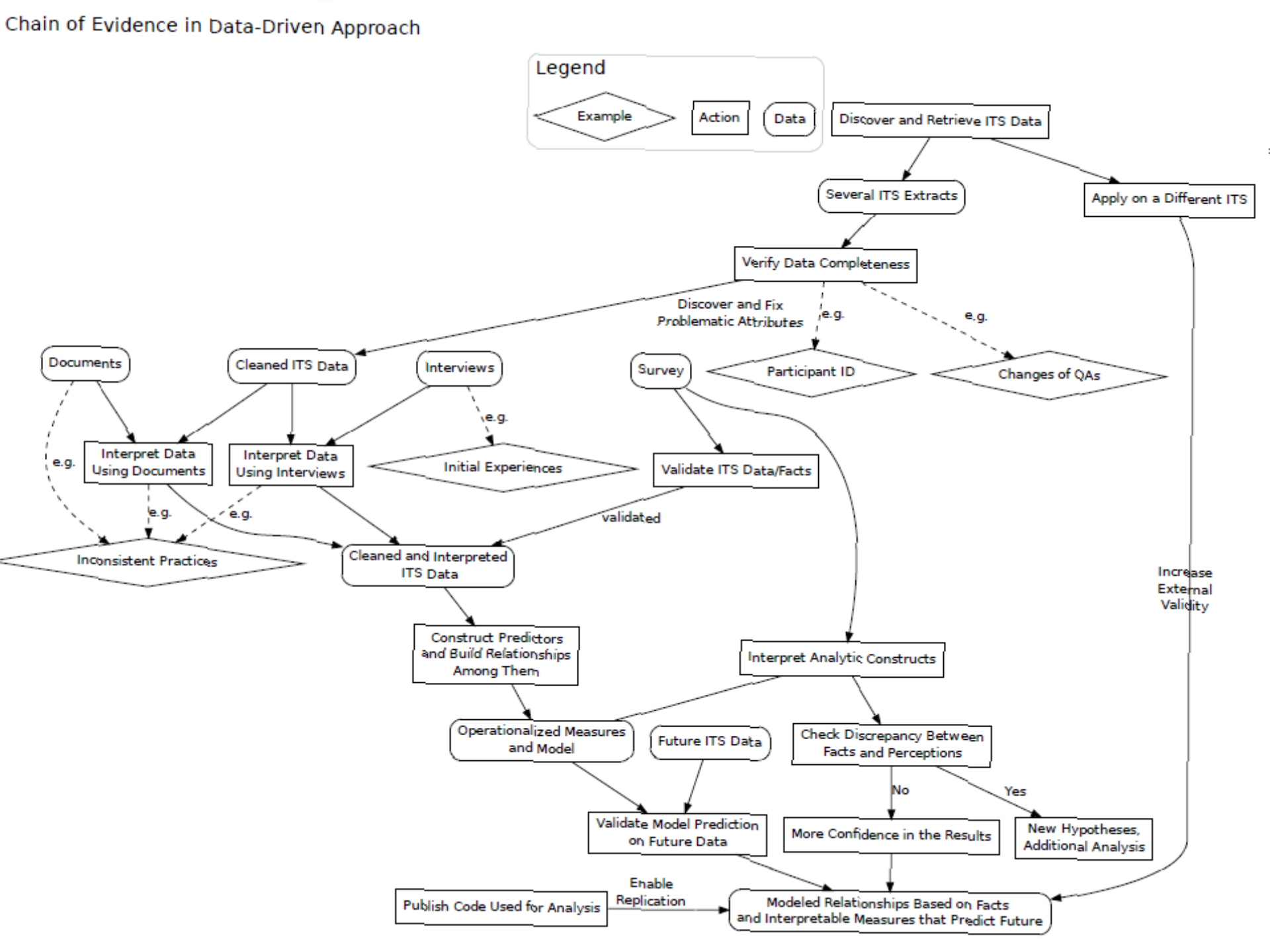
myk@mozilla.org

2004/11/22

Modify Product

方法：基于六个Bugzilla snapshots

- Learn what was going on
  - Read issues of 40 contributors
  - Survey 56 (36 non-LTCs and 20 LTCs)
  - Extract practices published on project web sites
  - Review research literature
- Measure discovered factors via activity in Bugzilla
- Fit models of future LTCs
- Validate
  - Predict future LTCs
  - Conduct Survey on 240 Mozilla/Gnome participants
- Interpret, consider practical implications


Chain of Evidence in Data-Driven Approach
Legend
Example
Action
Data
Discover and Retrieve ITS Data
Several ITS Extracts
Apply on a Different ITS
Verify Data Completeness
Discover and Fix Problematic Attributes
e.g.
e.g.
Documents
Cleaned ITS Data
Interviews
Survey
Participant ID
Changes of QAs
e.g.
e.g.
Interpret Data Using Documents
Interpret Data Using Interviews
Initial Experiences
Validate ITS Data/Facts
e.g.
e.g.
e.g.
validated
Inconsistent Practices
Cleaned and Interpreted ITS Data
Increase External Validity
Construct Predictors and Build Relationships Among Them
Interpret Analytic Constructs
Operationalized Measures and Model
Future ITS Data
Check Discrepancy Between Facts and Perceptions
No
Yes
Validate Model Prediction on Future Data
More Confidence in the Results
New Hypotheses, Additional Analysis
Publish Code Used for Analysis
Enable Replication
Modeled Relationships Based on Facts and Interpretable Measures that Predict Future

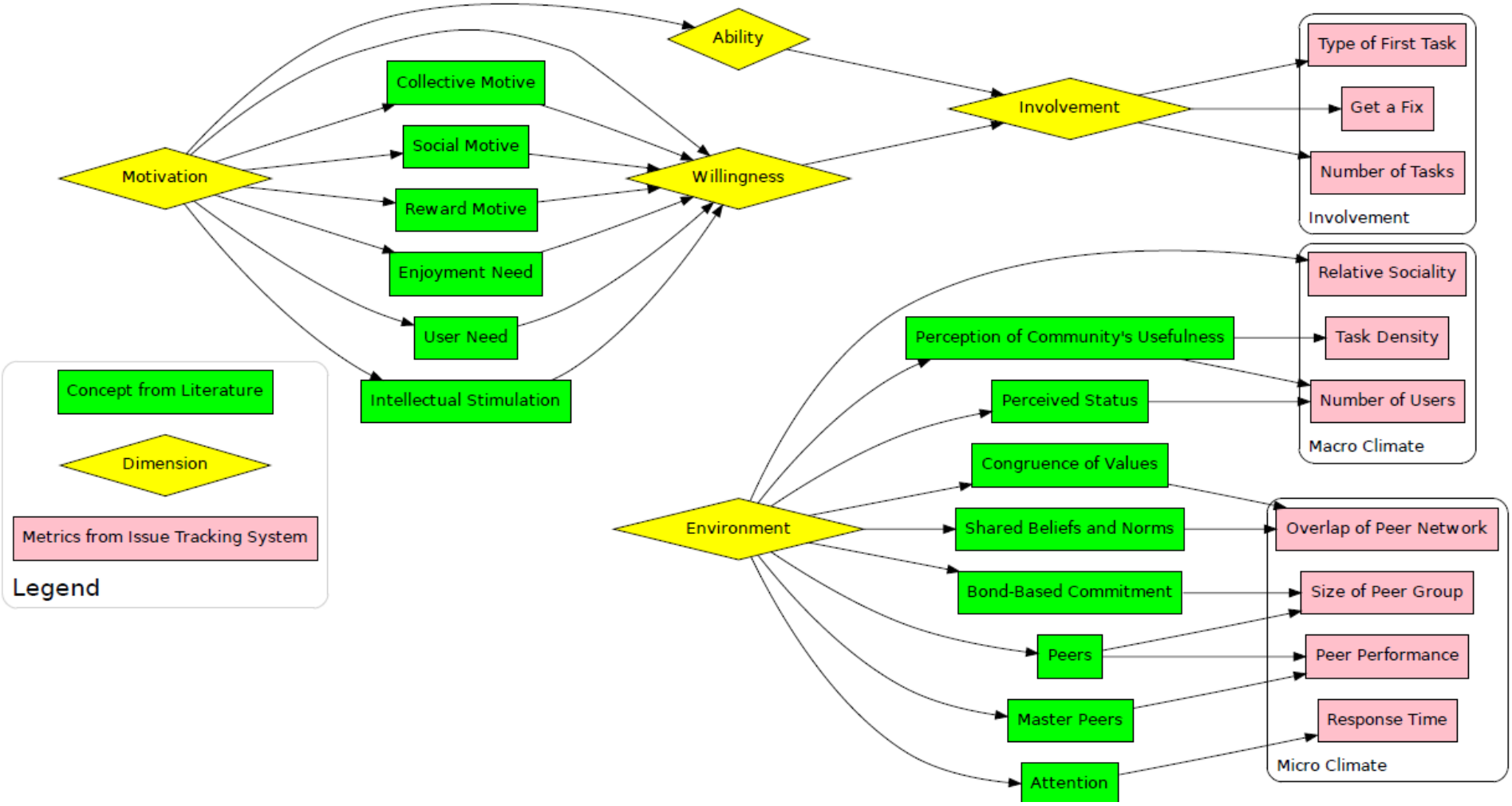
Ability/Willingness distinguishes LTCs

- Numbers and types of tasks
  - Non-LTC: ”I don’t have enough time/knowledge to resolve issues by myself”, provide minimum information necessary to report, don’t respond to requests for information
  - LTC: “Patch to get access attributes for nested class/struct/union”
  - LTCs had higher response rate (Fisher’s-test p-value=0.07)
- Willing to spend more effort on tasks
  - “If you have faced a bug, you need to spend effort to describe it... to check for duplicates... to create report... to wait until response.”
  - “All time you are waiting you must keep an issue in mind.”
  - “After [the] initial response there is [a] good possibility that devs can’t or don’t want to reproduce the issue and you must know how to [do] diagnostics and how to prove that issue really exists.”

Environment determines people's fate

- Macro-climate: same for everybody
  - Popularity: "GNOME is something which you can show to your friends and family members"
- Micro-climate: specific to each individual
  - Attention, Number of peers, Performance of peers
    - "With bugzilla, ... the feedback from the developers shows that they care, and appreciate the effort I made, and actively work to solve the bug in a way that I can see progress."
    - "As I met a lot of nice people at GUADECs who became friends there was also a personal component involved in the motivation."
    - "I learned a lot from this leading open source project while working with other contributors"

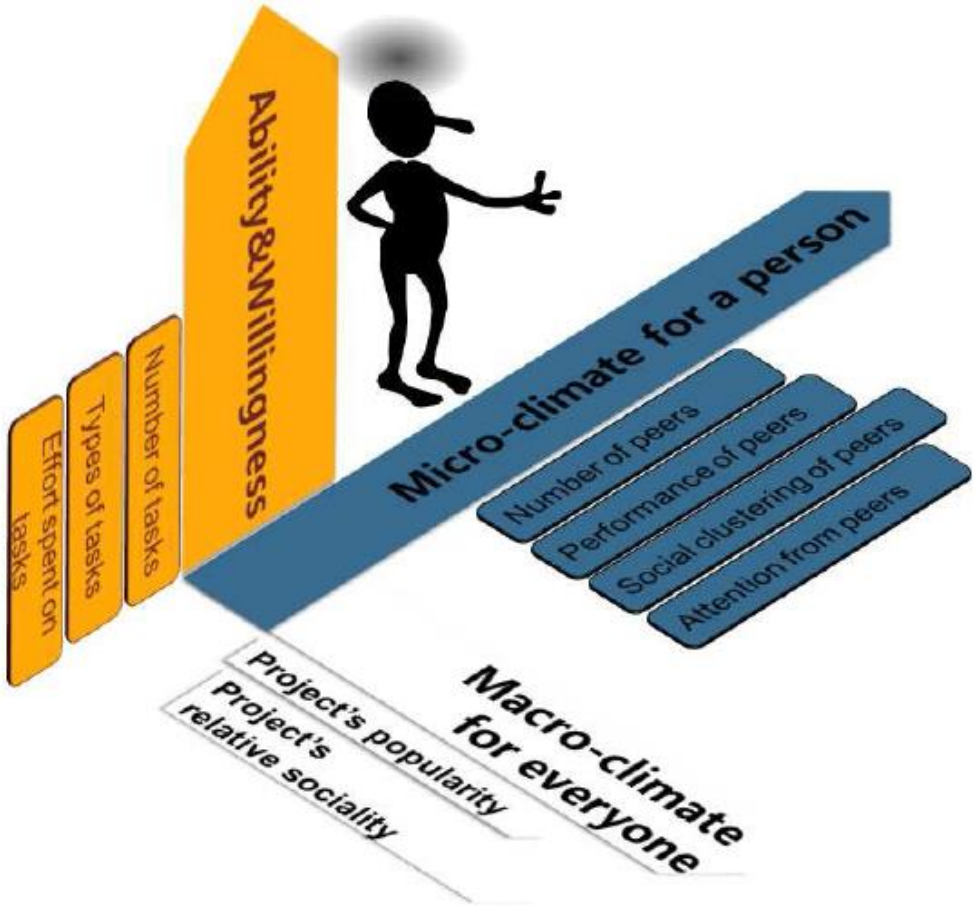
Metrics Derived from Bugzilla



Measures of Ability/Willingness and Environment

- Ability/Willingness can be measured via
  - The volume and the type of tasks
  - The effort spent on tasks
- Environment can be measured via
  - Macro-climate (shared among participants)
    - Project's popularity
    - Project's relative sociality
  - Micro-climate (unique for each person)
    - Number of peers
    - Peers' productivity
    - Peers' social clustering
    - The attention received from peers

Three Dimensions



Logistic Regression Model for LTCs

$$\begin{aligned} isLTC \sim\ & nUsr + RS + GotFix + BtE \\ & + nCmt + nPeer + pShared \\ & + LckAttn + PeerPerf + prj \end{aligned}$$

Logistic regression model for LTCs

Measure	Predictor	Odds Ratio		Direction
		Mozilla	Gnome	
Ability & Willingness	got at least one fix	2	2	⇑
	comment/not BB	1.5	3	⇑
	number of comments	2	1.5	⇑
Micro env	lack of attention	$\frac{2}{3}$	$\frac{2}{3}$	⇓
	peers' productivity	1.2	2	⇑
	peers' soc. clust.	1.5	1.2	⇑
	number of peers	1.14	0.94	⇕
Macro env.	number of users	0.85	$\frac{1}{2}$	⇓
	relative sociality	1.07	0.73	⇕

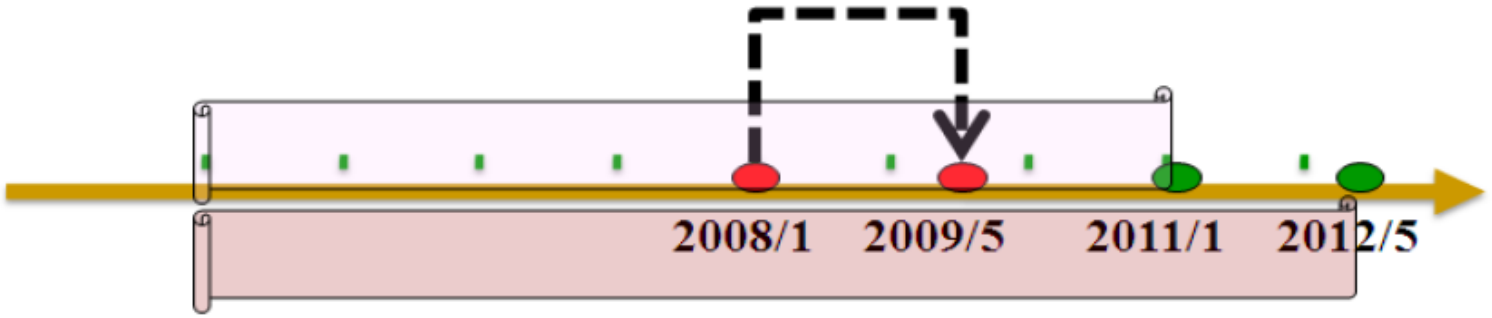
Response: {not-LTC, LTC} for Mozilla/Gnome (130,472/125,665 observations)

Who will become an LTC?

- Actions in the first month predict LTCs
  - Pro-community attitude has the greatest positive effect
    - The choice to start by a comment for an existing issue
    - Effort spent to improve the quality of issue reporting
  - Bad environment deters via
    - Macro-climate of high project popularity
    - Micro-climate of low attention
  - Good environment attracts via
    - Micro-climate of peer performance and
    - Micro-climate of peer social clustering

Can we predict future LTCs?

- Predict future using a new Moz extract



- Created prediction using 2011 snapshot:
  - 25,406 joiners during 2008.01-2009.05
- Determine LTCs from a new Mozilla snapshot on 2012.05
- Prediction performance
  - 24% recall (32 out of 131 LTCs were predicted)
  - 37% precision (32 of 86 predictions were LTCs)
  - 72 times higher than a random choice

Can we reproduce the model?

- Reproduce the model using Mozilla dump

Comparing Models for Mozilla 2011 and 2013
(170,237 Observations)

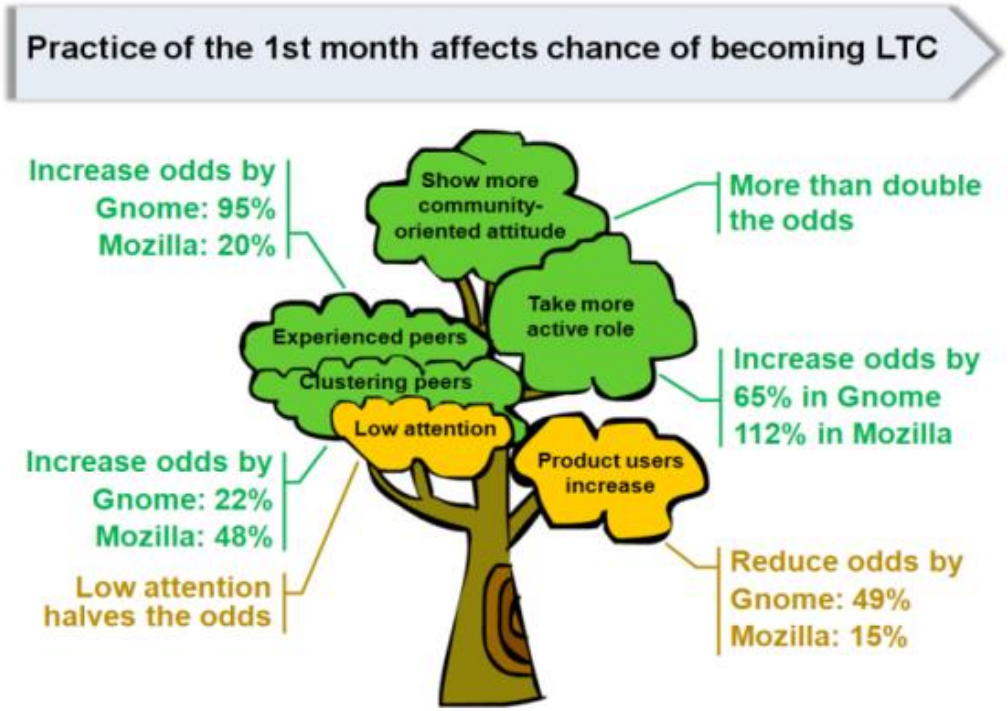
Coeff	Est'11	Est'13	z-val'11	z-val'14	change
(Intcpt)	−7.49	−7.18	−17.87	−23.031	+
nUsr	−0.601	−1.09	−4.00	−8.238	+
RS	0.701	0.19	2.39	0.684	
GotFix	0.74	0.84	8.90	11.138	+
FNotRep	0.507	0.40	6.17	5.577	−
nCmt	0.819	0.73	20.02	20.857	+
nPeer	0.142	0.14	6.92	7.970	+
pShared	2.35	2.55	17.40	21.035	+
LckAttn	−0.325	−0.42	−2.62	−4.548	+
PeerPerf	0.0649	0.07	4.95	6.473	+

Conduct a survey to validate analytic constructs

- Sampled 240 participants
- Carefully designed questions
- Customized emails for every individual
- 71 could not be delivered, and 29 responses were usable for our analysis
  - One unusable response: “Given that you call GNOME an OSS project, i don’t think I want to participate. GNOME is a free software project.”

People behave differently when joining!

isLTC ~ nUsr + RS + GotFix + BtE +nCmt + nPeer + pShared +LckAttn + PeerPerf + prj



What Make Long Term Contributors: Willingness and Opportunity in OSS Community. M. Zhou and A. Mockus. ICSE '12, Zurich, Switzerland, 2-9 June 2012. pp518-528

Who Will Stay in the FLOSS Community? Modeling Participant's Initial Behavior. M. Zhou and A. Mockus. Transaction of Software Engineering.2014

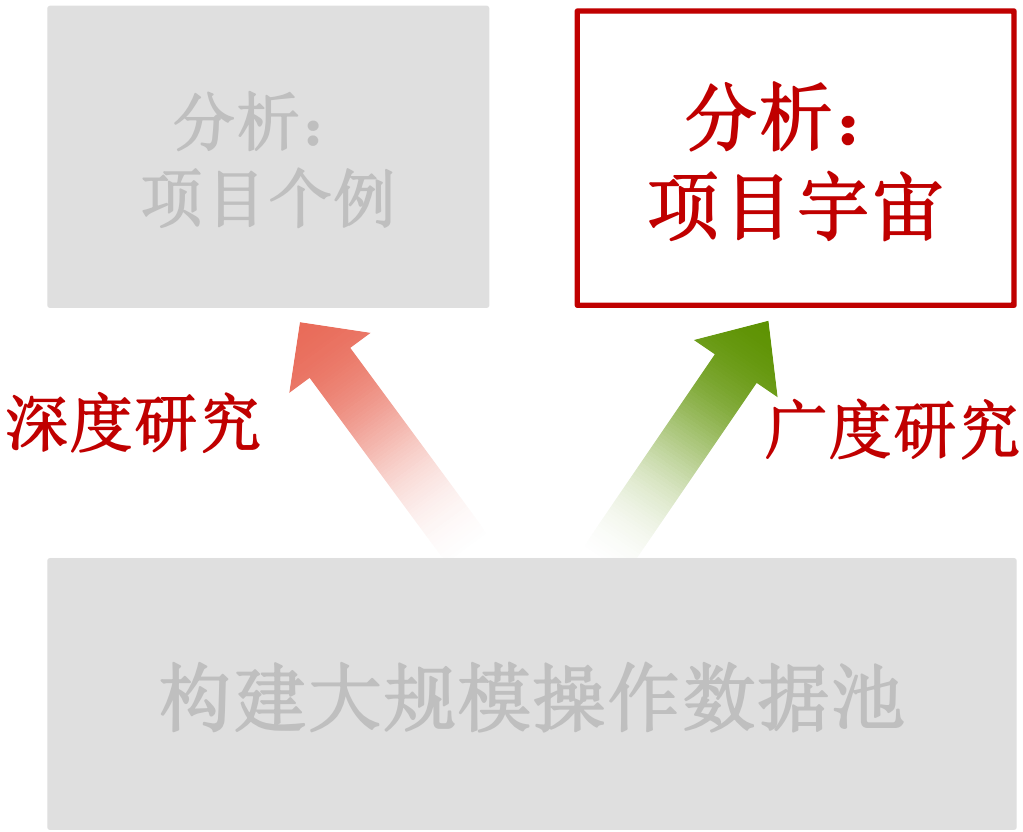
Summary of Contributions

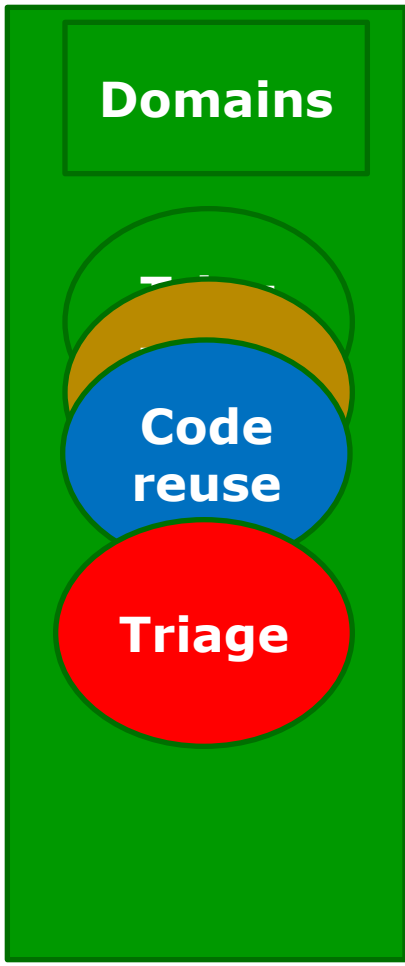
- Methodology
  - Measure individuals' attitudes and emotional dispositions from digital traces of their activity
- Science
  - Models of project success show largest effects brought by soft qualities, such as willingness
- Software practice
  - Projects: particular attention for new contributors
  - Newcomers: deeds matter, not intentions, limit expectations
- Future and Reproducibility
  - Implications for OSS and commercial development practices and non-software domains

Limitations

- Sensitivity analysis using various operationalizations
  - Full email was not available for post-2008 Gnome
  - Person to ID (email) changes over time
- Variation in operationalizations
  - BugBuddy in Gnome vs start from a bug report in Mozilla
- Do measures capture the right concepts e.g., peer clustering
- Should relationships be in the observed direction: e.g. project popularity is bad?
- Are Gnome and Mozilla representative?

路线图： 发现和利用微过程



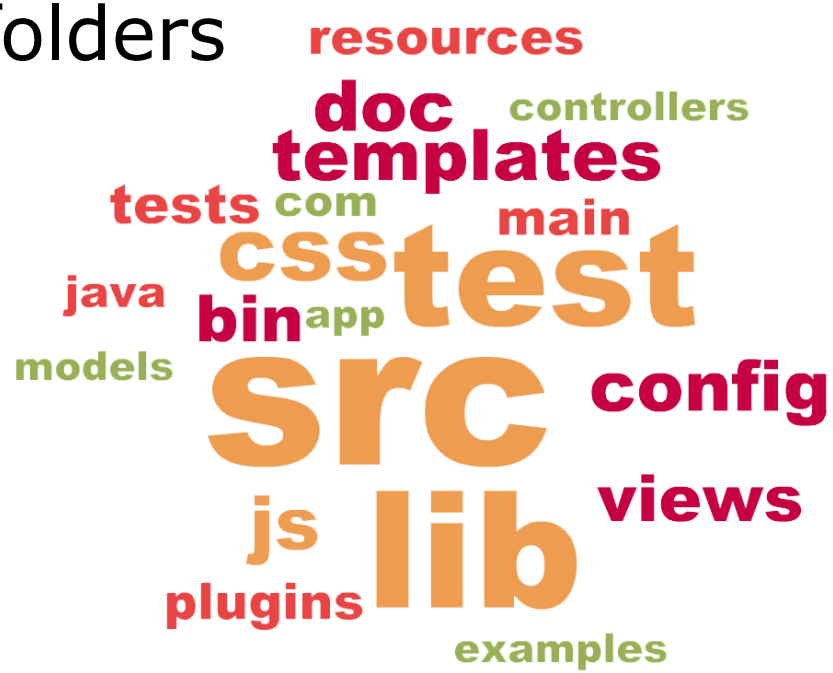


什么是复用代码的微过程?

- Repositories of open source universe
  - The commit history of projects from:
    - Github, googlecode, sourceforge, …
- Start from very basic questions
  - What folder names are common?
  - What folder names mostly co-occur?

MOST OFTEN USED FOLDERS

Top 20 folders



Patterns

- Related to programming languages and application domains, e.g., *com* and *css*
- Involving standard folders, e.g., *test*, *doc* and *examples*

Jiaxin Zhu, et al. 2014. Patterns of folder use and project popularity: a case study of github repositories. ESEM'2014

MOST OFTEN REUSED PROJECTS

Co-occur pair	Project	Application area
lib & mm, include & mm	linux kernel	OS
config & script, config &public	Ruby on Rails	web framework
js & langs, css & langs	TinyMCE	editor component
lib & feature	Cucumber	test framework
lib & spec	Rspec	test framework

如何构建一个健康的开源生态系统?

- Different types of commercial involvement
  - Hosting (Redhat, JBOSS),
  - Supporting (IBM, Geronimo),
  - Collaborating (BULL, JOnAS)

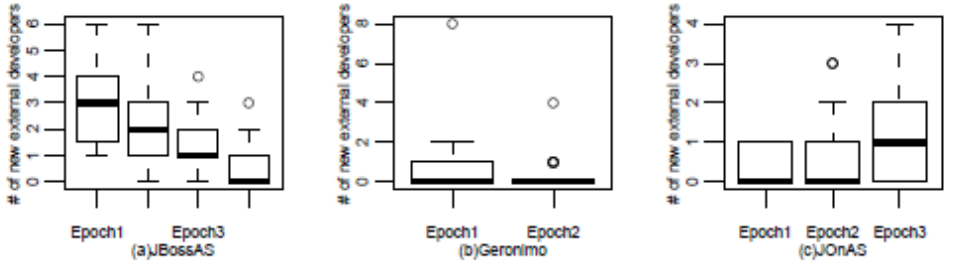


Fig. 1. Inflow of External Developers in JBossAS, Geronimo, and JOnAS

- For example, Hosting mechanism
  - Decrease the number of newcomers, but,
  - Increase their retention

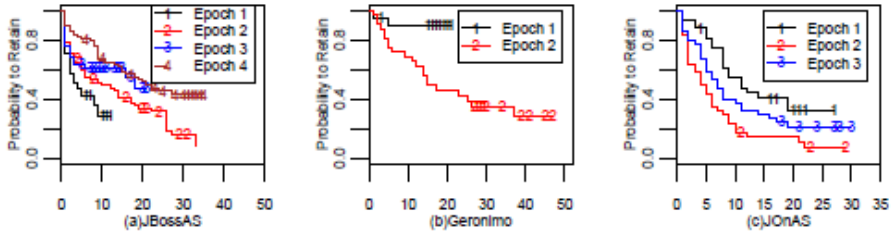


Fig. 2. Survival Curves of New Joiners in JBossAS, Geronimo, JOnAS each epoch

其他一些有趣的问题?

- 如何（基于微过程）评估项目成本?
- 如何（基于项目记忆）更好解决程序bug?
- 怎样的测试覆盖率足够?
- 如何的测试集足够?

- 技术问题vs.社会问题
- 多学科交叉应用：机器学习，高性能计算，...

总而言之，关于基于OD研究微过程

- MP帮助我们，在微过程粒度上，观察和度量影响项目成功的因素，进而预测、推荐和支持决策
- 普适性问题
  - 在一个或几个项目上适用的结果，不同项目、甚至同一个项目的不同数据集下不适用
- 归根结底：
  - 什么是我们所观测MP所产生的上下文？
  - 什么是我们所提出量度的上下文？
  - 什么是一个项目的上下文？

Law extracted knowledge from data?

现代化的社会，它能够将整个的社会以数目字管理

--黄仁宇