

We Don't Need Another Hero?

The Impact of “Heroes” on Software Development

Amritanshu Agrawal, Akond Rahman, Rahul Krishna, Alexander Sobran and Tim Menzies

ICSE-SEIP'18 Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice

Pages 245-253

背景

- 活発に活動を行う "Hero" と呼ばれる開発者が存在
- 開発者の20%が貢献の80%を占めるものを
"Hero Project" と呼ぶ
- こういったHeroはプロジェクトに対して
良い影響を与えないと言われることが多い^[1]

[1]Norman Bier, Marsha Lovett, and Robert Seacord. 2011. An online learning approach to information systems security education. In Proceedings of the 15th Colloquium for Information Systems Security Education.

- RQ1

How common are heroes?

- RQ2

How does team size affect the prevalence of hero projects?

- RQ3

Are hero projects associated with better software quality?

データセット

- GitHubのプロジェクト+フィルタリング

Sanity check	Discarded project count	
	Enterprise	Public
No. of Commits > 20	68	96
No. of Issues > 10	60	89
Personal purpose (# programmers > 8)	47	67
SW development only	9	51
Duration > 50 weeks	12	46
No. of Releases > 0	136	44
Collaboration (# Pull requests > 0)	35	54
Projects left after filtering	171	661

- Public 1,108 Enterprise 538



- Public 661 Enterprise 171

メトリクス

- プロジェクトを開発者数により3つに分類

small 8~15

medium 15~30

Large >30

- 6つのメトリクスを使用

$$I_r/I_t = \frac{\text{Total number of issues closed}}{\text{Total number of issues created}}$$

$$B_r/B_t = \frac{\text{Total number of Bug tagged issues closed}}{\text{Total number of Bug tagged issues created}}$$

$$E_r/E_t = \frac{\text{Total number of Enhancement tagged issues closed}}{\text{Total number of Enhancement tagged issues created}}$$

$$IR_t = \text{Median time taken to resolve issues}$$

$$BR_t = \text{Median time taken to resolve Bug tagged issues}$$

$$ER_t = \text{Median time taken to resolve Enhanced tagged issues}$$

RQ1 How common are heroes?

- 77(78)%のPublic(Enterprise)がHeroProject
- プロジェクト黎明期の貢献が原因

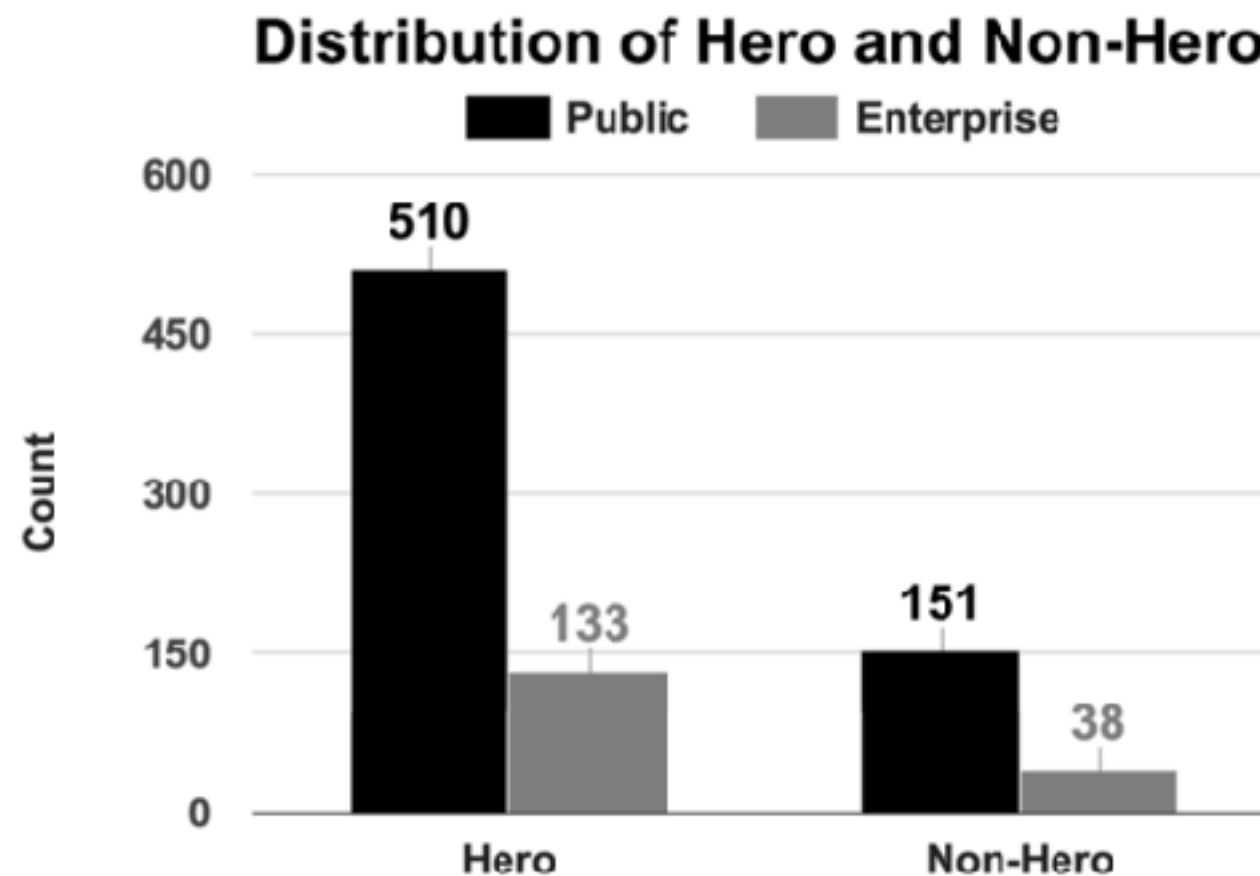
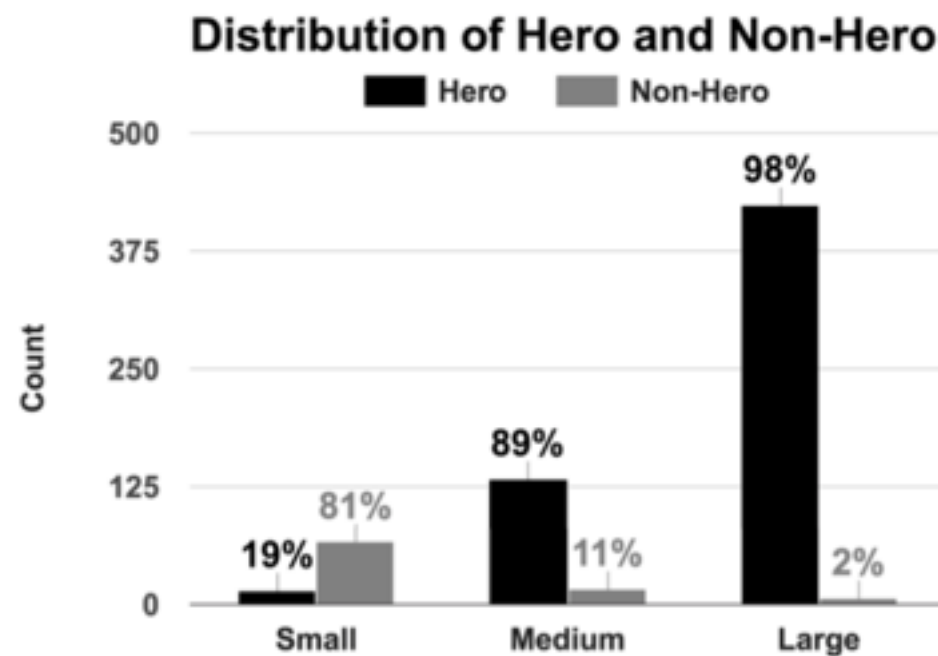


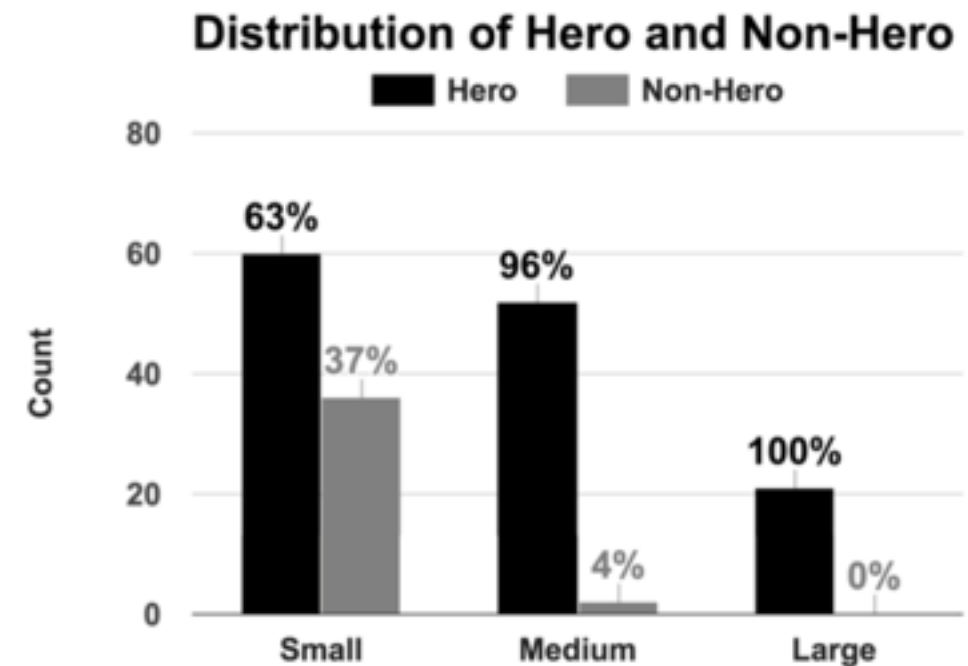
Figure 1: Distribution of Hero and Non Hero projects in Public and Enterprise projects. Note that these hero projects are very common.

RQ2 How does team size affect the prevalence of hero projects?

- 大きくなるほどHeroに依存する
- 大規模プロジェクトだとほぼ100%



Public



Enterprise

RQ3 Are hero projects associated with better software quality?

- RQ3a Does having a hero programmer improves the number of issues, bugs and enhancements being resolved?
- RQ3b Does having a hero programmer improves the time to resolve issues, bugs and enhancements?

の2つの観点で調査する

RQ3a the ratio of issues/bugs/enhancements successfully closed

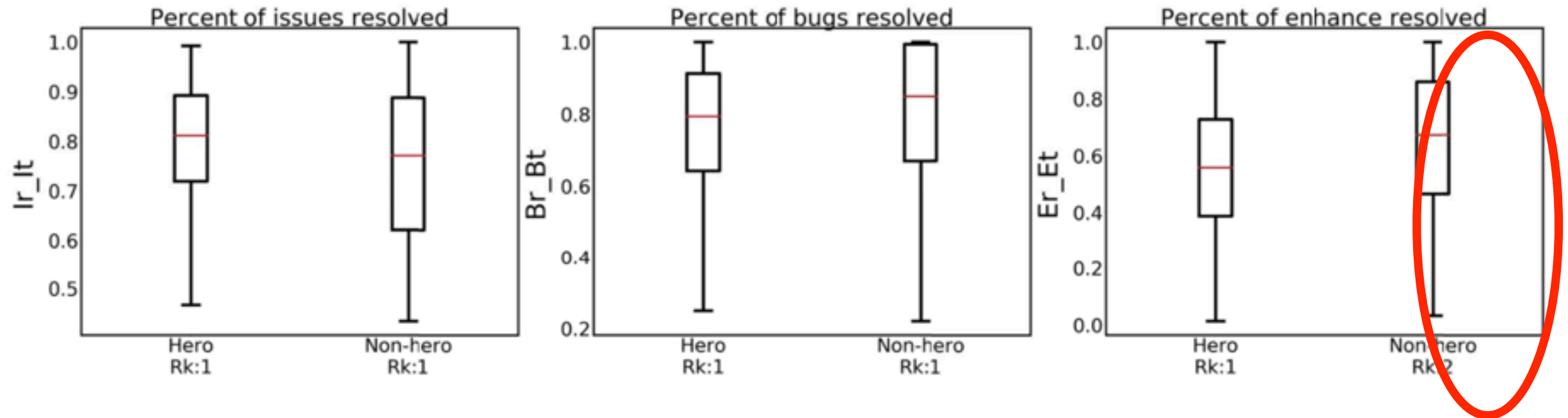


Figure 4: Public projects: Hero and Non-hero values of $I_r_{I_t}$, $B_r_{B_t}$ and $E_r_{E_t}$ (which is the ratio of issue, bug, enhancement reports being closed over the total issue, bug, enhancement reports created, respectively). Of these distributions, only the enhancement rates are different between hero and non-hero projects.

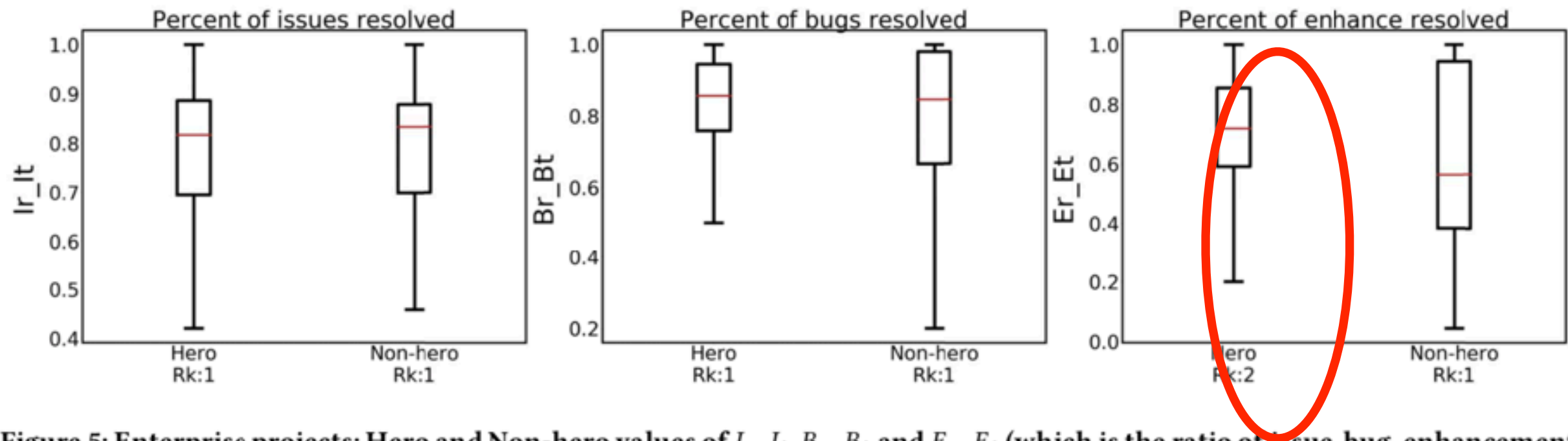


Figure 5: Enterprise projects: Hero and Non-hero values of $I_r_{I_t}$, $B_r_{B_t}$ and $E_r_{E_t}$ (which is the ratio of issue, bug, enhancement reports being closed over the total issue, bug, enhancement reports created, respectively). As before, only the enhancement rates are different between hero and non-hero projects.

RQ3b the time required to close those issues.

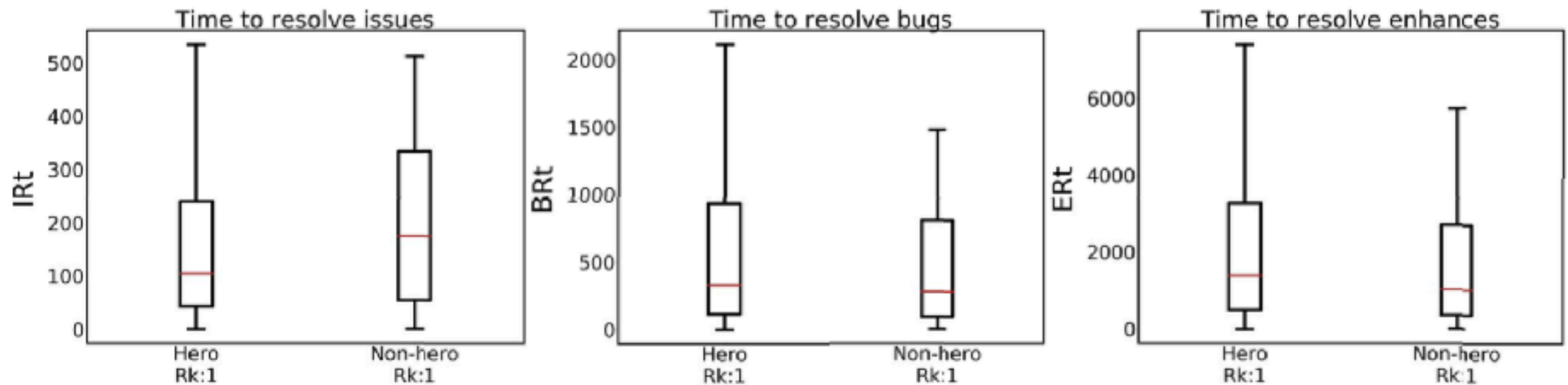


Figure 6: Public projects: Hero and Non-hero values of IR_t , BR_t and ER_t (which is the median time taken to resolve issue, bugs, enhancement reports respectively). Y-axis shown is in hours.

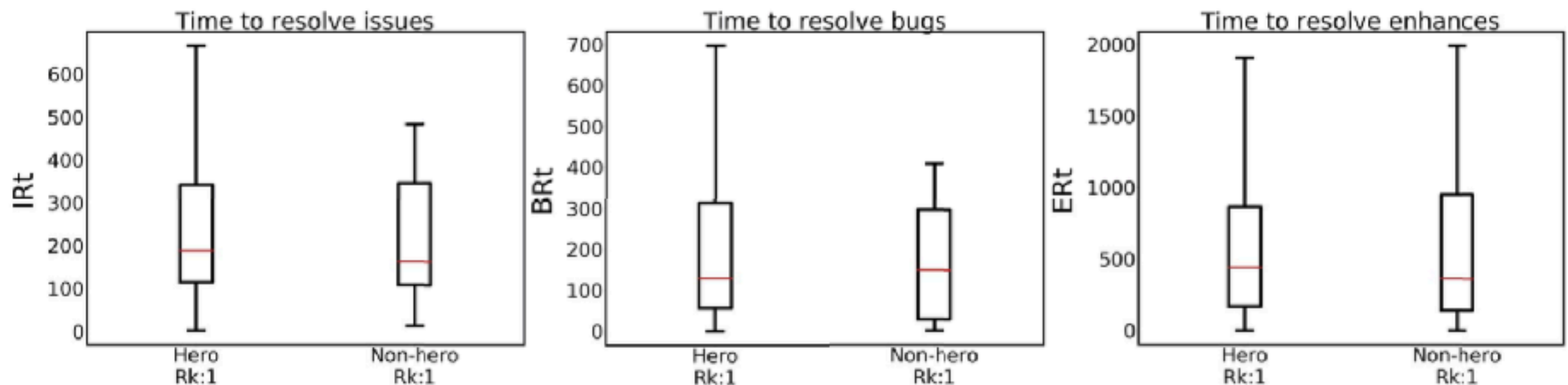


Figure 7: Enterprise projects: Hero and Non-hero values of IR_t , BR_t and ER_t (which is the median time taken to resolve issue, bugs, enhancement reports respectively). Y-axis shown is in hours.

まとめ

- ほぼ大多数のプロジェクトはHeroProject
- Heroが多くともバグなどの時間/率に影響はない
- これらのHeroを多く保持していく必要がある

所感

- 選択プロジェクトが多く
フィルタリングもしており良さげ
- Heroを多く保持するという結論があまり納得できない
 - enterpriseのenhancementに有意差があったが . . .