Mining repositories to reveal the community structures of Open Source projects
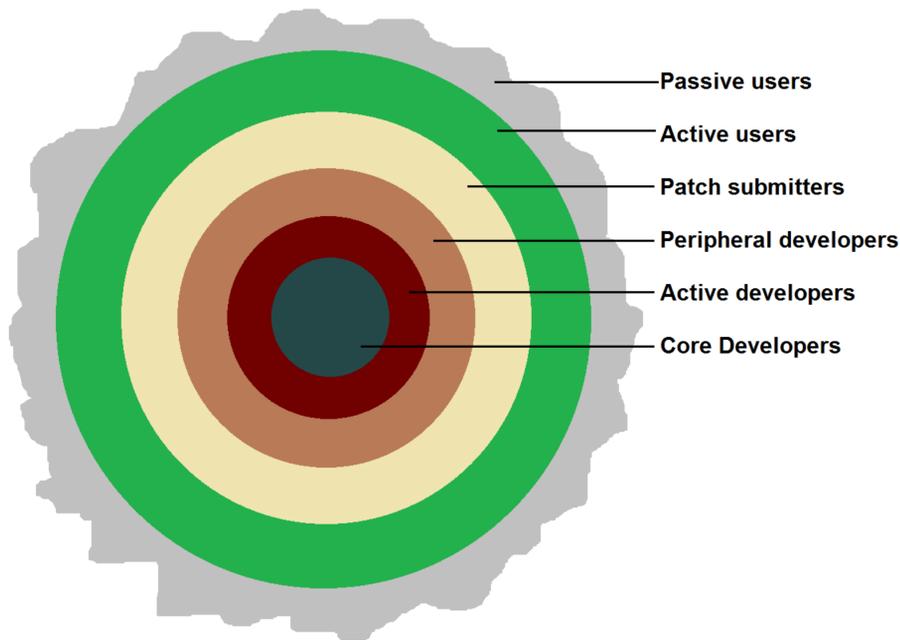
Amiangshu Bosu

Department of Computer Science, University of Alabama, Tuscaloosa, Alabama
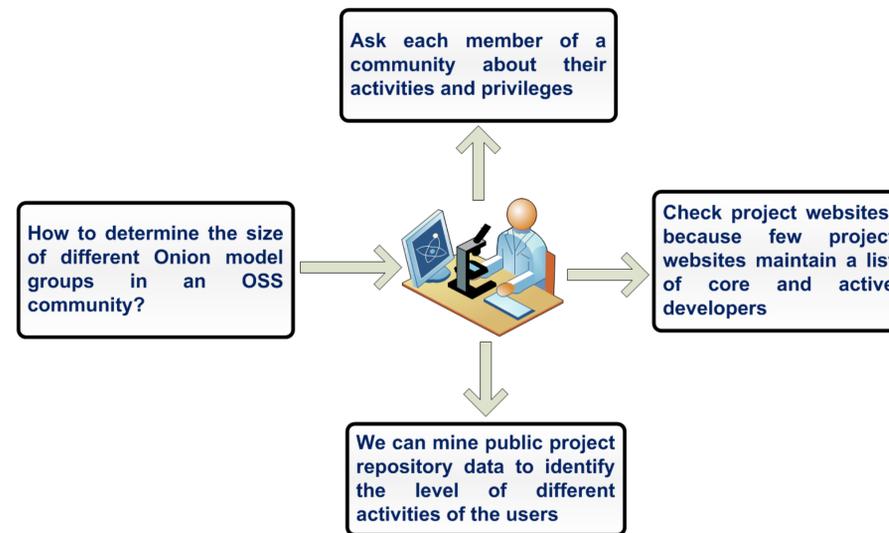
Introduction

The number of members within an Open Source Software (OSS) community is a good indicator of the success of an OSS project. Moreover, the size of an OSS community and the social structure of that community help to understand the social aspects (e.g., socialization, collaboration between members, mutual trust and reputation) of that community. In this poster, I describe a research plan to mine publicly available OSS repositories to reveal the social structures of OSS communities.

Open Source Community Structure (Onion Model)

- Passive users
- Active users
- Patch submitters
- Peripheral developers
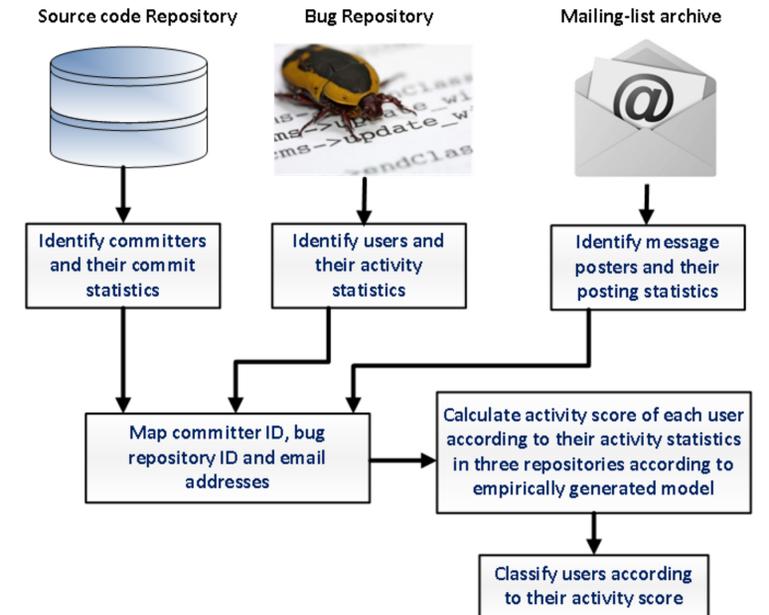- Active developers
- Core Developers

- **Core members** - responsible for guiding and coordinating the development efforts

- **Active developers** - regular contributors to the projects and make up the primary development force

- **Peripheral developers -** typically maintain one or more sub-components (modules) of a project and contribute occasionally

- **Patch submitters** - do not have write access to the core repository. They report bugs and often submit bug fixes to the mailing list

- **Active users** - often report bugs and participate in mailing list discussions to provide important user feedback

- **Passive users -** do not interact with the development community. The passive users constitute the majority of the community members
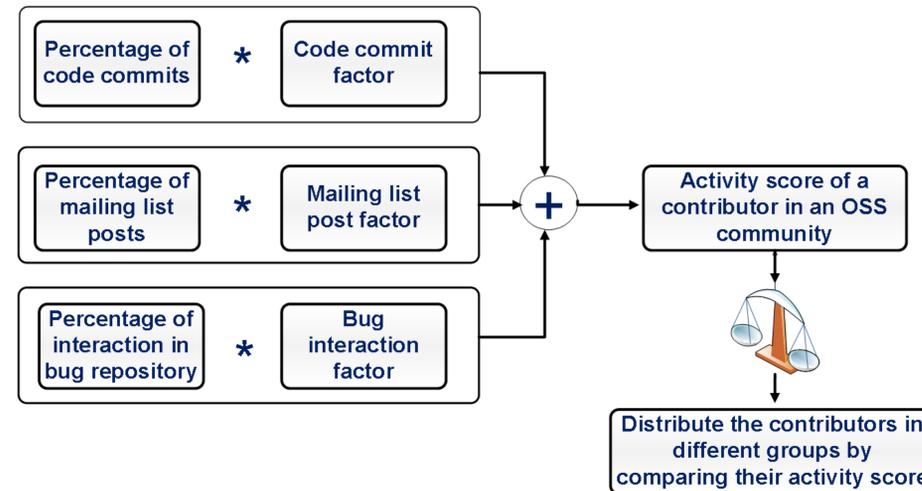
How many members belong to each group?

How to determine the size of different Onion model groups in an OSS community?

Ask each member of a community about their activities and privileges

Check project websites, because few project websites maintain a list of core and active developers

We can mine public project repository data to identify the level of different activities of the users

Our proposed model

Percentage of code commits $*$ Code commit factor

Percentage of mailing list posts $*$ Mailing list post factor

Percentage of interaction in bug repository $*$ Bug interaction factor

$+$ Activity score of a contributor in an OSS community

Distribute the contributors in different groups by comparing their activity score

Proposed workflow

Source code Repository | Bug Repository | Mailing-list archive

Identify committers and their commit statistics

Identify users and their activity statistics

Identify message posters and their posting statistics

Map committer ID, bug repository ID and email addresses

Calculate activity score of each user according to their activity statistics in three repositories according to empirically generated model

Classify users according to their activity score

Challenges

- Need to customize the mining scripts based on the type of repository.

- To determine all the activities of a member it is essential to link all the IDs and email addresses he uses. However, linking those by manual inspection might not be convenient for studies involving a large number of projects.

Available mining tools

- **CVSAnalY:** http://tools.libresoft.es/cvsanaly

- **Bicho:** http://tools.libresoft.es/bicho

- **Mailing list stats:** http://projects.libresoft.es/projects/mlstats

THE UNIVERSITY OF ALABAMA COMPUTER SCIENCE

Big picture

In the future, I plan to use the results to determine how social structures affect the socialization, mutual trust and reputation within OSS communities.