MINCE: MINing ChangE History of Android Project

Vibha Singhal Sinha, Senthil Mani, and Monika Gupta
IBM Research - New Delhi, India
{*vibha.sinha,sentmani,monikgup*}@*in.ibm.com*

***Abstract*—An analysis of commit history of Android reveals that Android has a code base of 550K files, where on an average each file has been modified 8.7 times. 41% of files have been modified at-least once. In terms of contributors, it has an overall contributor community of 1563, with 58.5% of them having made $>$ 5 commits. Moreover, the contributor community shows high churn levels, with only 13 of contributors continuing from 2005 to 2011. In terms of industry participation, Google & Android account for 22% of developers. Intel and RedHat account for 2% of contributors each and IBM, Oracle, TI, SGI account for another 1% each. Android code can be classified into 5 sub-projects: *kernel, platform, device, tools* and *toolchain*. In this paper, we profile each of these sub-projects in terms of change volumes, contributor and industry participation. We further picked specific framework topics such as UI, security, whose understanding is required from perspective of developing apps over Android, and present some insights on community participation around the same.**

I. INTRODUCTION

The Android Core code base (AOSP) was released to Open Source to foster collaboration and participation from the community outside of Android and Google organizations. Being open source, the restriction on the demography and skill-set of users participating in the project is not governed[1]. In this study, we attempt to get insights on the development of AOSP code. We start by investigating the change volumes in various sub-projects (Section III). Then we analyze the contributor community in AOSP (Section IV), namely: relative contributor density across sub-projects, contributor overlap across sub-projects and contributor churn across the years. We also profile the contributors on their organization affiliation and attempt to derive information on volume and nature of non-Google, non-Android contributions in the project (Section V). Overall, this analysis is similar to the consolidated data *Eclipse* exposes from their change history through their Dash project[2].

Android App Developers, use the core code base to leverage the basic features provided by the OS. For the benefit of it's app developers, Android has provided documentation on the core framework areas[3] that are typically exercised during app development. In our analysis, we further classified the "java" files in the Android core code to these framework topics and analyzed change activity, contributor density and industry participation around the same (Section VI).

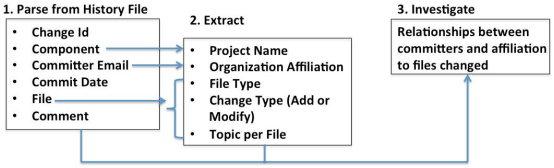
1. Parse from History File
• Change Id
• Component
• Committer Email
• Commit Date
• File
• Comment
2. Extract
• Project Name
• Organization Affiliation
• File Type
• Change Type (Add or Modify)
• Topic per File
3. Investigate
Relationships between committers and affiliation to files changed

Figure 1. Approach for Data Collection and Analysis

II. Data and Approach

Our approach (Figure 1) for data collection and analysis is a three step process. The challenge team had provided a XML file for Android change history log[1] containing change history from 1998 to Apr 2011. In step 1, we parsed it to extract out the following information per change set: component where change set was committed, names of files modified in change set, committer email, commit date and comment. In step 2, we further extracted the following additional information using the data from the previous step:

- **Sub-projects**: Using the first identifier of the component name which was composed of multiple identifiers joined by underscore, we identified 5 unique sub-projects : *device*, *kernel*, *platform*, *toolchain* and *tools*.
- **Organization affiliation**: For each committer, this affiliation was extracted from the email domain. We only resolved *com* domains leaving *org* and *edu* . For example, email id *a@Android.com* was resolved to *Android*. Email domains open for public use such as *gmail, hotmail, yahoo* were ignored.
- **Change and file type**: For each file we identified change type as added/modified and file type as *Java*/*C*/*Others* based on file extension.
- **Topic**: We mapped key class names mentioned for each framework topic from Android development guide to file names (part or whole match), and assigned it to that topic. For example, for topic "Activity", one of the key class is "Loader" and any file whose named contained the word "Loader", was assigned to this topic. However, this mapping was only done for ".java" files. This is referred to as *Topic Mapping Heuristic* in Section VI.

Finally in step 3, this augmented data was stored in *mysql* database and sliced/diced in different ways to derive the insights discussed in the paper.

[1] http://source.Android.com/source/roles.html
[2] http://dash.eclipse.org/dash/commits/web-app/
[3] http://developer.Android.com/guide/index

978-1-4673-1761-0/12/$31.00 © 2012 IEEE

Table I
SUMMARY OF CHANGE ACTIVITY INFORMATION

Project	File Counts			Change Sets	File level Change Activity		C & Java Source Modifications			
	Total	Java	C		2005-11	2011	No	Low	Medium	High
device	1147(0.2%)	42(3.7%)	348(30.3%)	968(0.3%)	890(77.5%)	43	19.5%	75%	6%	0
kernel	5904(10.4%)	0	49,549(84%)	2,41,560(78.8%)	57,320(97.8%)	14,509	13.2%	40%	46%	1.4%
platform	1,99,759(35.1%)	38,039(19%)	74,953(37.5%)	60,115(19.6%)	1,62,239(81.2%)	6490	15%	59%	25%	0.1%
toolchain	30,231(53.2%)	2486 (8.2%)	1,16,785 (38.6%)	84 (0.02%)	7110(2.35%)	249	98%	0.04%	0	0
tools	5385(0.9%)	2950(54.7%)	157(3%)	3666(1.2%)	3434(63.7%)	457	15%	55.13%	30%	0
Total	5,67,645	65,894	2,41,792	3,06,393	2,30,993	21,748	1,62,883	91,673	52,350	15

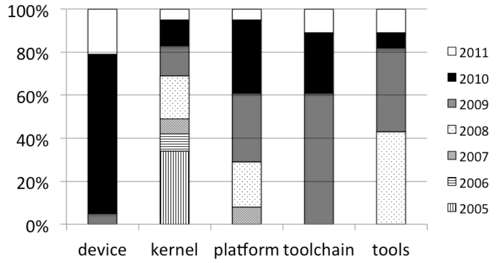

Figure 2. Distribution of new files added per Project per Year

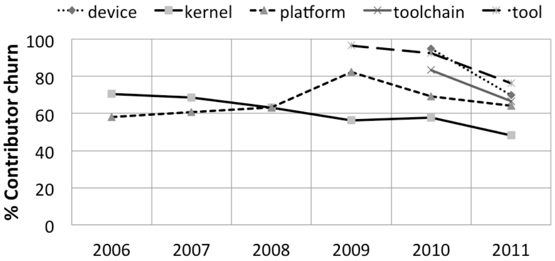

Figure 3. Contributor Churn

III. CHANGE VOLUME PROFILING

Table I provides a summary of change activity for sub-projects in Android. Column 2 lists the total file count per sub-project. Column 3 and 4 lists the count of Java and C source files respectively. With regards to project compositions based on their file types, *kernel* is completely C based (84%) whereas *tools* is mostly Java based (54%).

Column 5 gives the change sets submitted per sub-project. *kernel* accounts for 10.4% of overall files, but witnessed maximum amount of change activity (78.8%). *platform* comes next, with 35.1% of overall files and 19.6% commit activity. *toolchain* contains 53.2% of overall files but hardly had any commit activity (only 84 change set commits). Each change set can contain multiple files, with each files being modified or added. Overall there were 4963798 file modifications with an average file change ratio of 8.7.

Column 6, show the files modified at-least once. In *kernel*, 97% of files were modified at least once after being added. In *device, platform, tools*, files changed were 77%, 81% and 64% respectively. However, for *toolchain*, only 2.3% of files have been modified. Column 7 lists the number of files modified in 2011 across the sub-projects. Additionally, Figure 2 shows for each project, the percentage of new files added from 2005-2011. 20% of new files for *kernel, tools* and 40% of new files for *platform, toolchain* and 90% of files for *device* was added in 2010-11.

For source files (c, java, h, cc), columns 8 through 11, present the % of files that were never modified even once after being added (column 8), modified <5 times (low code churn - column 9), modified between 5 to 100 times (medium level code churn - column 10), and modified >100 times (high level code churn - column 11). For *toolchain* 98% of source code has not been modified even once after the initial check in. For all other sub-projects, <20% of source files have not been modified after initial commit, hence, indicating active development. In *kernel* 46% of source files have seen medium churn with 12 files being modified >500 times. For *device*, *platform* and *tools*, most files have undergone low churn (modified <5 times).

To summarize, Android is seeing a large amount of change activity (average 8.7 modifications per file) with 41% of code based modified at-least once over the 2005-11 time-period and 12% of the overall files were added or modified in 2011.*kernel* sub-project has the maximum volume of changes both in terms of commits (78.8%) and files modified (97.8%), followed by *platform*. *toolchain* has seen the least amount of change.

IV. CONTRIBUTOR PROFILING

The analyzed change history had a total of 1563 commiters[4]. Table II shows per sub-project, number of committers who contributed to the project (column 2), who significantly contributed (>5 commits) to that project (column 3) and active committers (column 4 - >5 commits in 2011). The number of contributors is highest in *platform* (993), while it is lowest for *toolchain* (9). 83.6% of committer population in *kernel* has made significant contributions, while it is lower in other sub-projects. Also, the current active committers population is a small subset of the overall committer population. For e.g, only 8% of total committers in *device* are currently active, while in *kernel* it is 34.2%.

Comparing this data with project size in Table I, for *toolchain* only 0.5% of overall committers contributing to Android are responsible for producing 53.2% of the overall Android code base. However, 63.5% of contributors are responsible for 35.1% of code belonging to *platform* project, which accounts for 19% of overall changes in Android. We

[4] AOSP is the core OS code for Android. There is a much larger community of developers contributing Android apps. An analysis of Android app developer community is beyond the scope of this paper

Table II
SUMMARY OF CONTRIBUTORS INFORMATION

Projects	Summary of Contributors			Contributor overlap across sub-projects					Distribution of file by number of contributors		
	Total	Significant	Active	K	P	D	T	TC	Low	Medium	High
device (D)	64 (4%)	28 (43.7%)	5 (8%)	14	59	3	7	2	1121	20	6
kernel (K)	537 (34.3%)	449 (83.6%)	184 (34.2%)	516	20	14	2	0	49142	8458	1443
platform (P)	993 (63.5%)	507 (51%)	74 (7.4%)	20	912	59	16	8	189462	9856	441
toolchain (TC)	9 (0.5%)	3 (33.3%)	2 (22.2%)	0	8	2	1	1	302308	3	0
tools (T)	64 (4%)	22 (34.4%)	7(11%)	2	16	7	48	1	5366	17	2

Table III
TOPICS & COMPANIES CONTRIBUTING

Topic	Companies
Sensors	Invensense
Data Storage	Sony Ericsson
Multimedia & Camera	Samsung, MM-Sol, Sony Ericsson, TI
Bluetooth	StEricsson SonyEricsson
Intent Filters	T-mobile, HTC

further investigated the committers in terms of their overlap across sub-projects, their churn and their commit density.

A. Contributor Overlap across Projects

Only 86 out of total 1563 contributors, commit to more than one sub-project. Table II (columns 5 - 9) provides a sub-project level overlap of contributors. *kernel* has 516 unique contributors and 20 of it's contributors are shared with *platform*, 14 with *device* and 2 with *tools*. *device, toolchain* have maximum contributor overlap with other sub-projects, 96% and 89% respectively. It is interesting to note that in-spite of the smallest project-size (0.9%) (Table I) and relatively smaller committer coverage (4%) (Table II), 75% (48) of contributors in *tools* are unique to that project.

B. Contributor Churn

We define *Contributor Churn* (C_C) for a project, as the ratio of number of changes in contributors (δ_C) over total number of contributors (C_T) in the project. The changes in contributors (δ_C) include new contributors who started committing in a particular year (C_A) and existing contributors from previous year who did not commit in that year (C_O).

$$\delta_C = C_A + C_O, C_C = \frac{\delta_C}{C_T} \tag{1}$$

Figure 3 plots the *% Contributor Churn* calculated yearly for each project as a line graph over the time period from 2006 to 2011. All sub-projects have significant contributor churn >50% across all years. In 2011, 60% of *platform, device, tool, toolchain* contributors were either added or removed and only 40% continued from previous year. For *kernel* the constant developer population is around 48%. Only 13 contributors continued from 2005 to 2011.

C. Contributor Density Per File

Table II (columns 10 - 12) provides the distribution on number of committers per sub-project. Columns 10, 11 and 12 list the count of files that has been modified by <5 contributors (low contributor density), 5 to 10 contributors (medium contributor density) and >10 contributors (high contributor density) respectively.

Overall across projects, 96.4% of files have contributor density <5. *kernel* has the maximum files (1443) with high contributor density (>10), and *platform* has the maximum files (9856) in medium contributor density (5 - 10). However, within the projects, *kernel* again has 14.33% of its files with medium contributor density when compared to 4.9% of *platform*. Even though *platform* has the highest number of committers (993), only 0.2% of its files have contributor density greater than 10.*kernel* which stands second with respect to number of committers (537) and third with respect to number of files (59043), has 16.7% of files with contributor density greater than 5, maximum among all the sub-projects, followed by *platform* (5.1%) and *device* (2.2%).

V. INDUSTRY ACTIVITY IN ANDROID

We were able to assign an organization affiliation to 858 out of 1563 contributors and thus identified 171 companies. About half of the companies mentioned in Open handset alliance[5] showed up in the commit history, for example, IBM, TAT, NXP, ST, TI. This mapping between companies and contributors and further details on commit activity can be navigated at[2]. We refer to 10 (out of 171) companies as top players who had >= 10 contributors. Table IV lists these companies along with information on their commit volume in each sub-project. Column 2 gives the contributor count. Columns 3–7 list the number of change sets committed. The value in brackets indicates the % of unique files committed (added and/or modified) across sub-projects.

From Column 2 it is evident that, Google and Android constitute only 22% of the overall contributor population, while the top players account for another 34% of contributor population and the remaining 44% are individual contributors. This indicates a healthy open source and institutional participation in Android.

Based on change set count (columns 3–7), Android and Google account for bulk of the delivered commits (>90% in *tools, toolchain and device* and 68% in *platform*). However, in *kernel* there is hardly any contribution (1%) from them with the highest change set contributors being redhat and suse. Further we saw a lot of contributors with ".org" as domain name committing to *kernel* especially gnome.org. This is in accordance with the known fact that Android kernel is essentially linux based and a lot of linux fixes make it to Android code. Also, companies like IBM, Oracle, TI, Intel, Samsung have also contributed to *kernel*.

[5] http://www.openhandsetalliance.com/oha_members.html

Table IV
TOP COMPANIES CONTRIBUTING TO ANDROID PROJECTS

		Change Set Count (% File Coverage)				
Project	#	kernel	platform	device	tools	toolchain
oracle	13	4090 (2.3)				
suse	14	25181 (20.8)				
sgi	15	1585 (.5)				
samsung	16	563 (.9)	17(.01)	2 (1.6)		
ibm	19	3860 (2.6)				
ti	19	764 (.7)	15 (.01)			
redhat	32	18069 (11.5)	1063 (.42)			
intel	34	9055 (6.6)	161(1.2)			3
Android	63	1932 (2.3)	15278 (53.7)	141 (26)	34 (23.1)	6
google	285	889 (1.3)	25867 (69)	822 (78.4)	3355 (77)	69 (100)

We were surprised to see SAP contributions in AOSP. Investigation revealed that SAP contributions to Android happened recently (2010-11), probably owing to their commitment towards *Enterprise Mobility*[6]. Also majority of *device* code has been contributed by Google in 2010 -11 and this might have been towards their preparation for Motorola acquisition in August 2011.

To summarize, while there is healthy contribution from non-Google/Android community, these contributions are restricted to pockets of code base and not very wide-spread. Except for *kernel*, Google seems to have high percentage of unique files added/modified in all other sub-projects.

VI. TOPIC ANALYSIS

Based on our topic mapping heuristic, we mapped 18 out of the 20 framework topics to files.Table V lists for some sample topics, the number of files identified (column 2), change-sets which modified these files (column 3), unique contributors who made a change to any of the files in this topic (column 4) and count of companies whose contributors committed to any of these files (column 5). The full data-set is made available here[3]. For e.g, for the topic "Activity", 992 files were identified that have been modified 2843 times by 178 contributors belonging to 12 different companies. This indicates that this topic is of interest to a relatively large developer community. All topics except "Graphics", "Device Administration" and "Session Initiation Protocol" had commits from non-Google, non-Android companies. In Table III, for some topics we show the company diversification ignoring Google and Android. For example, the sensors related code has only been modified by "Invensense", leaders in MEMS gyroscope, and motion processing technologies for consumer electronics.

Table V
FRAMEWORK TOPICS AND RELATED COMMIT HISTORY

Topic	Files	Change Sets	Committers	Company
Activity	992	2843	178	12
Content Provider	603	729	89	7
Copy Paste	533	755	93	7
Intent & Intent Filters	65	533	72	6
Location & Maps	8	132	19	5
Multimedia & Camera	84	801	48	8
User Interface	1849	4354	210	16

Table VI
SECURITY AND NFC TOPIC ANALYSIS

Topic	Companies	Key Class
Security	Finik, SAP, Sony Ericsson	SecuritySupport, Certificate SecuritySettings
NFC	Trusted-Logic , NXP	nfc

For two of the topics Security and Near Field Communication, there were no key classes mentioned. We applied some keywords based on our understanding of the topic to map files to these topics (Table VI). The security area has seen modifications by companies like "Finik", which has an Email app - Libre with a key feature which allows users to decide how to control access to their data. In course of building this app, the company might have required some changes to how Android handles security and hence we see some contributions. Trusted-logic shows up as a contributor under "Near Field Communications" . In 2010, Trusted-Logic released a software platform enabling mobile phone applications to use Near Field Communication (NFC) connectivity[7].

VII. CONCLUSION AND FUTURE WORK

We analyzed the Android change history log to profile the various sub-projects, the developer community and the industry participation. What emerges is that there is a large churn in the developer community. *kernel* and *platform* sub-projects have come out of Google and Android dominance. However, there exists diverse company participation across software, mobile manufactures and telecom service providers. As future work we can use metrics like code churn and developer churn to identify vulnerable code locations [4] and change history information to mine code ownership at a sub-package and file level [5].

REFERENCES

[1] E. Shihab, Y. Kamei, and P. Bhattacharya, "Mining challenge 2012: The android platform," in *The 9th Working Conference on Mining Software Repositories*, 2012, p. to appear.

[2] "Android - company profiling," http://www-958.ibm.com/software/data/cognos/manyeyes/visualizations/android-source-distribution-by-pro.

[3] "Android - topic profiling," http://www-958.ibm.com/software/data/cognos/manyeyes/visualizations/framework-topic-distribution.

[4] Y. Shin, A. Meneely, L. Williams, and J. A. Osborne, "Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities." *IEEE Trans. Software Eng.*, vol. 37, no. 6, pp. 772–787, 2011.

[5] L. Hattori and M. Lanza, "Mining the history of synchronous changes to refine code ownership," in *Proceedings of the 2009 6th IEEE International Working Conference on Mining Software Repositories*, ser. MSR '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 141–150.

[6] http://www.sap.com/solutions/technology/enterprise-mobility/index.epx

[7] http://www.trusted-logic.com/spip.php?article189