

Open Source Software Survivability Analysis Using Communication Pattern Validation

A. Angel Mary, S. K. Prasanth

Dept. Of CSE, Vardhaman College Of Engineering, Shamshabad, A.P, India.

Associate Professor, Dept of CSE, Vardhaman College Of Engineering, Shamshabad, A.P, India.

Abstract: *The Growth Of Open Source Software has created a major change In Software Development. Open Source Binds The Power Of Distributed Peer Review And Transparency To Create High Quality, Secure And Easily Integrated Software At An Accelerated Pace And Lower Cost. Since OSS Has Reached A Mature Stage, Adoption In Business Has No Barrier. Effective Communication Between The OSS Developer Communities Is One Of The Signposts That Contribute The Open Source Success. In This Paper A Communication Validation Tool Is Proposed, That Brings Out Contribution Index, Community Between-Ness Certainty, Community Density Of The Communication Network. It Also Presents The Nativity Of Developer Community, Progress Of Activity And Distribution Overtime.*

Submitted date 21 June 2013

Accepted Date: 26 June 2013

I. Introduction

The Growth Of Open Source Software Has Created A Major Change In Software Development. Open Source Binds The Power Of Distributed Peer Review And Transparency To Create High Quality, Secure And Easily Integrated Software At An Accelerated Pace And Lower Cost. More Than 800 Respondents In 2013 Survey Reported That Better Quality, Freedom From Vendor Lock-In, Flexibility, Elasticity, Superior Security, Pace Of Innovation, Lower Cost And Access To Source Code Are The Main Factors To Adopt OSS In Business.

Facts Of OSS

By Switching To OS We Expect To Cut 20% From IT Budget - CTO Lastminute.Com

We Are Amazed At The Speed And Reliability Of Linux - CTO Unilever

- 50% Of Chief Information Officers Are Committed To OSS
- 70% Of Increase In Investments On Open Source Projects
- 40% Of Companies Are Planning To Migrate Mission Critical Software To OSS In The Coming Year
- 80% Of Internet Infrastructure Is Based On OSS
- Nine Out Of Ten Banks Use OSS

These Are The Current Facts That Fetch The Sight Of Proprietary Software User To Migrate To OSS.

Communication In OSS

Since The Open Source Developers Are Globally Spread, They Don't Have Face To Face Interaction And They Won't Work Under One Roof. They Never Follow Strict Work Hours And Most Of The Developers Are Volunteers Who Coordinate With Others To Bring Up Successful Project. Comfortable Discussions, Efficient, Continuous, Proficient Code Development And Faster Bug Rectification Are Factors That Lead To A Successful, Good Quality Open Source Software. Without Communication And Contribution No Open Source Software Development Is Successful.

Hence Open Source Developers Use Internet Massively For Their Communication Purpose. Social Networks And E-Mail Networks Also Make Their Foot In Their Discussions. Most Of The Open Source Developer Communities Use E-Mail Networks For Discussion And Code Commit. In This Paper We Analyze The Archive Of Mails Send And Received By The Apache Qpid- Open Source AMQP Developer Community. The Extended Information System Model Proposed By Sara Fernandez, Presents That Quality Contribution Without Communication Is Impossible. In This Paper We Analyze The Communication Between The Developers Of Apache Qpid- AMQP Open Source Software To Come Up With Measures Like Contribution Index, Community Between-Ness Certainty, Community Density Of The Communication Network.

II. Background

2.1 Open Source Software

Open Source Is The Term Used To Describe Software, Of Which The High-Level Code Is Supplied Free Of Charge To Whoever Would Like It, By Definition, They Are Permitted To:

- Use The Code For Whatever They Wish
- Edit And Modify The Code To Suit Their Needs
- Redistribute The Code To Whoever They Wish
- Improve The Software, And Redistribute It In Its Modified Form

When All These Conditions Are Met, Software Is Considered Truly *Open Source*.

Open Source Software (OSS) Projects Are Developed And Distributed For Free, With Full Access To The Project Source Code. Recently There Has Been A Significant Increase In The Use Of These Projects. Some OSS Projects Have Earned Themselves A High Reputation And Corporate Sponsorships. Large Corporations (E.G., IBM, SUN Microsystems) Are Becoming Involved With The OSS Movement In Various Capacities.

2.2 Measures For Quality And Success In OSS

When The Volunteers Fail To Produce Reliable Contribution For Any Open Source Software, It Meets The Darker Side. Analysis Of Web Hosting For OSS, Such As Github Or Sourceforge Provides Useful Insight About The Success History Of Many OSS. Sourceforge Is One Of The Most Popular Hosting Sites For Free Software With Over 313,000 Projects. It Is Not Only A Good Resource To Find Well Maintained Projects, But Also To Find Out A Large Number Of Abandoned Projects And Low Quality Code. The Most Common Problems Identified Are The Unsupported Code, Configuration Management, Security Updates And Users That Do Not Know How To Report Bugs.

1. Unsupported Code Affects Quality. Note That It Is Impossible To Know How To Handle Code That Has Previously Been Contributed But It Is Now Unmaintained;
2. Configuration Management Affects Quality Depending On The Number Of Contributions. It Is Difficult For The Lead Developer To Test All Contributions;
3. Security Updates Affects Quality Since, Although Some Bugs Are Fixed There Isn't Any Schedule To Make The Fixes Available;
4. Users Who Sometimes Do Not Know How To Report A Bug And Often Fail To Include Enough Information In Bug Reports;

All These Problems Need To Be Faced And It Is Necessary To Come Up With A Range Of Measures That Can Attest The Quality Of An OSS Product. Some Say That The Quality Of OSS Can Be Measured By The Number Of Users And The Code Quality. Actually, In The Book "The Cathedral And The Bazaar", Raymond Suggests That The Level Of Quality Is Partly Due To The High Degree Of Peer Review And User Involvement Often Found In Free Software Projects.

Communication And Contribution In OSS

It Is Clear That The OSS Phenomena Are Mainly Driven By The Developer's Communities. Although OSS Projects Differ Amongst Themselves, For Example, In Their Size, Also The Communities Differ In Levels Participation In Coding And Communication. Actually, Communication And Contribution Also Depend On Having Suitable Strategies. These Strategies Have A Number Of Dependencies That Can Be Measured. Communication In OSS Also Contributes To Knowledge, Sharing And Collective Innovation.

As Far As Communication Is Concerned Stable Releases, Participation, Interactivity And Release Strategy Are Considered Measures.

- **Software Constant Releases** Are Considered A Communication Measure Even If Some Authors Hold That An OSS Project Never Leaves The Beta State. Stable Releases May Be Considered A Measure Of Communication Since New Releases Typically Bring In New Users, I.E., The Number Of Downloads Should Increase Compared To Previous Releases.
- **Also, The Release Notes** Improve Communication Between Developers And Users, Which Helps To Boost The Testing Function Shared By Users;
- **Participation** In Developing, Testing And Use Is Also An Obvious Measure;
- **Communication Interactivity** Is Expected To Have A Positive Impact On OSS Development. It Is Believed That A Good, Interactive Communication Between Developers Leads To Successful And Quality Products;
- **Release Strategy**, Under The Typical OSS Slogan "Release Often And Release Fast". The Advantage Of Using This Mechanism Is To Provide The Necessary Feedback To Understand If The Approach Followed In A Project Is The Best.

III. CPV Tool Architecture

The Architecture Consists Of Two Tier, In Which The First Tier Consist Of A E-Mail Parser Which Decomposes The Sender Id, Recipient Id, Time, Date, Subject And Content Of The Mail And Stored In A Database. The Database Can Be Queried To Produce Varies Results. The Second Tier Contains An Implementation That Produces The Analysis Report Of The Communication Network Of Apache Qpid- AMQP Developers. The Report Will Produce Contribution Index, Community Between-Ness Certainty And Community Density Of The Communication Network.

IV. Methodology

The Individual Developer’s Involvement Or Contribution Can Be Measured Using Contribution Index. The Following Formula Is Used In Measuring The Same.

$$\text{Contribution Index} = \frac{\# \text{ Of Msg. Sent} - \# \text{ Of Msg. Received}}{\# \text{ Of Msg. Sent} + \# \text{ Of Msg. Received}}$$

Community Between-Ness Centrality Measures The Ability Of A Developer In Terms Of Accessing The Communication Network Resources And How The Developer Manages Information Flow With Others. It Also Provides The Level Of Involvement Of A Particular Developer In The Communication Network. Community Between-Ness Centrality Identifies The Core Members Of The OSS Development Process

$$\left. \begin{array}{l} \text{Community Between-Ness} \\ \text{Centrality} \end{array} \right\} = \frac{2 \sum_{k=1}^n G_{ik}(P_j)}{G_{ik}}$$

Where G_{ik} = No. Of Geodesics From Developer P_i To P_k (Geodesics Is The Shortest Path Between 2 Developers In The Communication Network), $G_{ik}(P_j)$ = No. Of Geodesics That Contain Developer P_j As An Intermediary In The Geodesics From P_j To P_k .

Community Density Is The Measure That Provides The Ability Of The Group To Respond To Changes, Critical Issues And Unity. It Measures The Withstanding Capacity Of The Group In All Issues.

$$\text{Community Density} = \frac{2L}{G(G-1)}$$

Where L = No. Of Communication Links In The Community Network, G = No. Of Developers In The Community Network.

Three Stage Least Square Estimate

Success Is Measured From Both Demand And Supply Side. It's Known The Need And Supply Would Affect Each Other Concurrently. Therefore, We Apply Three-Period-Least- Square (Three Stage Least Squares Estimate) Method To Assess The Data By Eview. From Outcomes By Three Stage Least Squares Estimate, We Discover The Impact Of Conversation Design On Improvement Action Is Critical. The Product Shows The Critical Impact For Project Density And Both Project Centrality On Improvement Action, Which Support The Following Theory:

Theory 1: Centralized Jobs Could Be Highly Likely To Reach Greater Development Action.

Theory 2: The Project Expansion Activity Will Be Negatively Affected By Project Density.

V. Results And Discussion

The Task Achievement Is Taken From Both Contribute Side - Demand Side, And Advancement Task - Project Reputation. Improvement Exercise Is Assessed By The Amount Of Stopped Bugs In A Fixed Period (I.E. One Month To The Data Compilation Date); Popularity Is Considered By The Number Of Downloads In A Predetermined Period. Both Of Two Measures Are Accounted On Sourceforge.Net.

Data Analysis Reveals The Typical Job Centrality Is Around 54.13% And It Amounts From Zero To 100%;

And The Typical Thickness Is Around 20.63% And It Ranges From 1.84% To 100%. To Immediately Understand The Communication Design Of The Task Group, Currently The Communication Design Graphs Of Two Example Projects By Using Ucinet Draw Tool. Job "Gaim" Has The Best Density In The Test Projects (Centrality: 46.98%; Thickness 1.84%). Job "Filezilla" Has The Greatest Centrality (Centrality 100%; Density: 22.22%).

Communicating Design Of The Task Group Is Taken By Project Density And Project Centrality; While The Project Achievement Is Seen In Relation To Recognition And Improvement Action. Results, Shown In Figure 2, Are Typically Supportive Of The Theories Posted In This Document. In Most Cases, From Our Research, We Realize That Variables Which Lead To Provide And Need Side Success Might Be Distinct. This Means That Various Meanings Of OSS Success Demand Different Group Of Predictors. For Instance The Conversation Patterns Are Somewhat More Related To The Action Level In The Designer Side; The Goal Group Is More Associated With The Recognition Of The Item, Etc.

Outcomes Of Three Stage Least Square Estimate Is Summarized In Figure 2.

DA→Development Activity

Pty→Popularity

PC→Project Centrality

PD→Project Density

TL→Types Of License

PL→Programming Language

FS→Financial Support

P.Comp→Project Complexity

PM→Project Maturity

TA→Target Audience

Value 0 Indicates Not Applicability

Here The Explored Analysis Done By Three-Stage Least Squares Estimation Method With 121 Samples And 103 Observations.

VI. Conclusion

Open Source Represents An Exciting Opportunity For Research In A Wide Variety Of Disciplines. This Project Applies E-Mail Analysis To Understand How Developers Communicate In An Open Source Project. Since The Developers In Open Source Projects Are Geographically Distributed And May Never Meet Face-To-Face, It Is Important To Understand How They Communicate To Organize And Coordinate Their Efforts.

The Analysis Of The OSS Project Suggests An Interesting Pattern Of Communication Where A Developer Having A Large Centrality Signifies A Leading Path Of Information Traffic, The Contribution Index Identifies The Core Developers Of The Project, And Community Density Gives The Readiness Of The Group To Dynamic Changes In Completing The Project. The Analysis Also Reveals That Communications Within The Project Are Driven By A Few Prominent Developers In One Sub-Group Who Are Centrally Located In The Network. These Results Suggest Interesting Opportunities For Future Research. For Example, One Could Examine Whether A Developer Comes Under Core Group Or Not. One Could Also Consider The Influence Of Communication Patterns On Aspects Of Project Routine Or Activity.

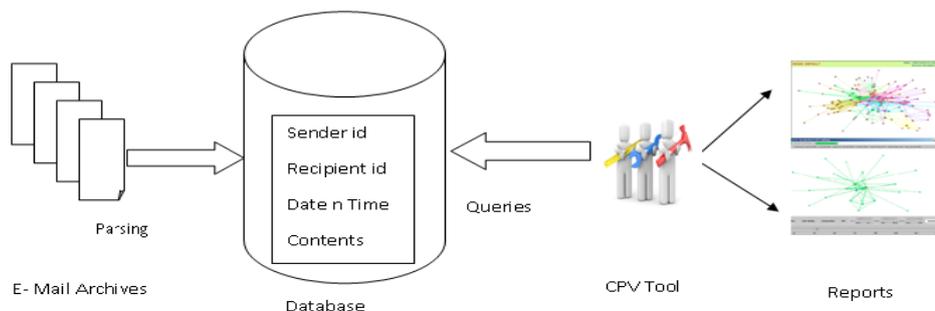


Figure 1: CPV Two Tier Architecture

		DA	Pty	PC	PD	TL	PL	FS	P.Comp	PM	TA
DA	Coeff	0	-0.732	0.446	-1.849	0.259	-0.707	0.648	0.420	0	0
	Std.Err	0	0.296	0.213	0.459	0.333	0.396	0.331	0.236	0	0
Pty	Coeff	1.858	0	0.005	0.285	1.206	-0.755	0	0	0.820	-1.101
	Std.Err	0.409	0	0.452	0.393	0.672	0.608	0	0	0.301	0.678

Figure 2: Results By Three Stage Least Squares Estimate

References

[1] OSI, The Open Source Definition, The Open Source Initiative, (Accessed: May 2001); [Http://opensource.org/docs/definition _ Plain.html](http://opensource.org/docs/definition_plain.html)

[2] R. Abreu And R. Premraj. How Developer Communication Frequency Relates To Bug Introducing Changes. In Proc. Joint Int'l Workshop On Software Evolution (IWPSE-EVOL), Pages 153-157. ACM SIGSOFT, 2009.

[3] E. Raymond, *The Cathedral And The Bazaar: Musings On Linux And Open Source By An Accidental Revolutionary* (O'Reilly, Cambridge, 1999).

[4] K. Lakhani, And E. Von Hippel (2000). How Open Source Software Works: "Free" User-To-User Assistance. MIT Sloan Open Source Project, (Accessed: October 2001); [Http://opensource.mit.edu/papers/lakhanivonhippelusersupport.pdf](http://opensource.mit.edu/papers/lakhanivonhippelusersupport.pdf)

[5] A. Mockus, R. Fielding And J. Herbsleb, A Case Study Of Open Source Software Development: The Apache Server, *Proceedings Of The Proceedings Of The 22nd International Conference On Software Engineering*, Limerick Ireland (2000).

[6] G. Madey, V. Freeh, And R. Tynan. The Open Source Software Development Phenomenon: An Analysis Based On Social Network Theory. In Eight Americas Conf. Information Systems, Pages 1806{1813, 2002.

[7] S. Y. T. Lee, Et Al, "Measuring Open Source Software Success," *Omega-International Journal Of Management Science*, Vol. 37, Pp. 426-438, 2009

[8] J. Wang, "Survival Factors For Free Open Source Software Projects: A Multi-Stage Perspective", *European Management Journal*, Vol.30(4), Pp.352-371, 2012.

[9] W. H. Delone, E. R. Mclean, "The Delone And Mclean Model Of Information Systems Success: A Ten-Year Update," *Journal Of Management Information Systems*, Vol. 19, Pp. 9-30, 2003

[10] V. Midha, P. Palvia, "Factors Affecting The Success Of Open Source Software", *Journal Of Systems And Software*, Vol. 85, Pp. 895-905, 2012

[12] F. Brooks. JR., "The Mythical Man-Month". *Essays On Software Engineering*. Addison-Wesley Publishing Company, 1975.

[13] Borgatti, S.P. & Everett, M.G. (1999). Models Of Core/Periphery Structures. *Social Networks* 21: 375-395

[14] P. Laplante, A. Gold, And T. Costello, "Open Source Software: Is It Worth Converting?" *IT Professional*, Vol. 9, No. 4, Pp. 28-33, July/ Aug. 2007.

[15] R. Pressman, *Software Engineering: A Practitioner's Approach With Bonus Chapter On Agile Development*, Sixth Ed. Mcgraw-Hill, 2004.

[16] D. Baccarini, "The Logical Framework Method For Defining Project Success," *Project Management J.*, Vol. 30, Pp. 25-32, 1999.

[17] OSI, The Open Source Definition, The Open Source Initiative, (Accessed: May 2001); [Http://opensource.org/docs/definitionj3lain.html](http://opensource.org/docs/definitionj3lain.html)

[18] E. Raymond, *The Cathedral And The Bazaar: Musings On Linux And Open Source By An Accidental Revolutionary* (O'Reilly, Cambridge, 1999).