

# An expertise communication study, relevant aspects and facts

Jhonny Cerezo

Department of Computer Science (DCC) University of Chile, Santiago, Chile

*jcerezo@dcc.uchile.cl*

## I. ABSTRACT

Communication has always played an important role in Software development, since the current grow of globally distributed software development, communication confronts more challenges, making it an important problem in software development. Then, different communication practices are adopted by companies with help of new technology, replacing face to face communication, although developers claim that it cannot replace live and personal communication. Let's see what happens in an open source community, since we know that development in open source communities are a special case. We conducted an exploratory study on active developers in Pharo's community, an open-source project, and contrast how software development communication takes place in open source community, focusing on expertise and direct communication. We conclude with future work in this area, with the objective of nurture communication.

## II. INTRODUCTION

Communication plays an important role in the software development process in terms of coordination among the different teams involved [12]. Now developers work from different continents, in globally distributed software development teams. So, communication quality must be nurtured, taking account the different factors presented in this technical report.

Communication can be split in two different categories, expertise communication and coordination-communication [11], in this technical report we will focus only on expertise communication, since coordination is aimed to same project developers, usually in private company software, we wait to cover the wider and little studied expertise communication, focusing in a free open source project as Pharo. Pharo is interesting, because they maintain direct and semi-synchronized communication, using chat channels, with any developer who would like to participate, and it is open to join.

When open source community or private software developers struggle with source code, they post a question in the community seeking for help, seeking for an expert, also software managers search for the most appropriate developer for a software task. Even though users in these online communities usually do not know each other and are identified by using pseudonyms, they are willing to help each other for various reasons, such as altruism reputation-enhancement

benefits, expected reciprocity, and direct learning benefits [7]. We know that experts are willing to help others, let's see if they are difficult to find. Even inside software industry companies they are difficult to locate [6]. This technical report aims to clarify how expertise communication takes place in an open source community.

The study pays attention to expertise communication, and the different aspects of it. To achieve this study's goal, we addressed the following Research Questions.

- **RQ1:** What is an expert to a developer?. This question aims to reveal the developers perception of an expert in free software communities.
- **RQ2:** Why software people aim to answer questions in a forum community?. Experts are not easily located, and if they do they might not have the time to answer questions, this research question will reveal why experts answer questions in forums or any other place.
- **RQ3:** Why developers ask questions maintaining private communication with an expert?. Assuming developers maintain private communication, we examine why leads to developers to choose private channels, we describe private communication as a peer-to-peer chat of two developers.
- **RQ4:** What leads developers, when they look for help, to ask questions instead of going to other knowledge source?. Expertise-communication occurs when a developer might have found an error or problem while working in some part of a particular project. Regardless these situations, expert developers use different strategies to solve problem. So, we explore what triggers expertise communication.
- **RQ5:** How difficult it is to find an expert to a developer?. Developers might find difficult to find experts in globally distributed software development [6].

## III. BACKGROUND

There is a long history of software development communication studies in globally distributed teams. It is known that communication inside a collaborative environments influences positively on productivity inside software development projects [6, 16], and such communication must be nurtured increasing the communication quality with technology that helps developers to communicate effectively, in this way

online communication channels play an important role in how expertise is shared manipulated and captured [13, 16].

1) *Communication in software development*: James D. Herbsleb and Audris Mockus presented their work on communication in globally distributed software development [6] (2003), although there is not work recent, it is relevant and well know in the field of communication in software development. They found that developers "find it much more difficult to identify distant colleagues with needed expertise and to communicate effectively with them", this problem has been intensified and also mitigated by different solutions online.

Nakakoji et al. [11] in 2010 defined communication in software development as coordination communication and expertise communication. In coordination-communication a developer tries to coordinate his or her task with dependent peers in order to avoid and/or solve emerging or potential conflicts. In expertise communication, a developer seeks information to solve his or her task at hand and asks peers for help. They also propose different aspects necessary to take into account in designing mechanisms to support coordination communication and expertise communication. During this technical report we will focus on expertise communication.

2) *Expertise*: Before the work of Nakakoji et al. [11], different studies have been carried out to study expertise in software development. Since expertise is a critical resource in a team, where the effective distribution of expertise can have a big influence in the success of a project [4]. In communication, particularly in expertise communication, the success of the communication itself cannot be effective unless you know the right person to contact with. For this particular problem many solutions proposed mined expertise from different sources and provided recommendation systems to find experts.

Jun Zhang et al. [17], investigated the expertise network in forums, setting the idea that people use forum platforms to share and collect expertise. They use social media algorithms to determine how expertise is distributed in a forum community.

In 2002 Audris Mockus and James D. Herbsleb presented in their work an expertise browser [10], that might be the first one in this field. That gives developers the capability of contacting with experts inside the same project in a distributed development private company. They collected expertise from source code. In a similar way, different other work have followed this work [5, 8, 9], where all of them were implemented in globally distributed private companies.

3) *Channels of communication*: Lucy Berlin et al. [3] in 1993 declared, experts serve as mentors in an informal apprenticeship, that interactions are crucial, and not only in helping the learners past immediate obstacles, but in transmitting a variety of useful, hard-to-find information. No research has found that remote communication replace live and personal communication. Through time, different technologies appeared to mitigate the long distance work, and the difficulty of coordination and expertise communication.

Margaret-Anne Storey et al. [13] created a survey on how social and communication media support software develop-

ment activities, in collaboration development environments. They showed that development tools are often integrated with or supplemented by communication channels and social media, made to support developers' collaboration and communication. The discussed on how social and communication media can be a distraction or can negatively impact their productivity through interruptions. And also provided a list of recommendations while using different communication channels.

Muhammad O. Ahmad et al. [2] investigated the Communication channels and practices adopted in Agile Software Development, they conducted their work using Systematic Mapping Study, in their work they reviewed relevant research papers before July 2018. Under their results they made a compilation of communication channels and their function in software development, and so the practices adopted on them by software developers. An interesting remark they made is that "chat is deemed to be more effective and useful for daily, informal information exchange or asking question from an expert about software functionality", this sentence gives relevance in this work at the time of focusing on synchronized and expertise communication.

4) *Open source community communication*: We know that communication plays an important role in open source communities, since there is no restrictions as in a private software company, people contribute freely and for their own benefit, and people join by maintaining communication with the core developers of such project [15]. We can know that the same rules of expertise communication, explained by Nakakoji et al. [11], plays the same role in opens source communities, we will make certain of it in this technical report.

#### IV. METHODOLOGY

The participants of this exploratory study are part of the Pharo's community, Pharo is an open source project which maintains different channels of communication: a chat server on Discord, a mailing list, Bugzilla, and Git Hub. This study is focused on Discord chat service which holds the biggest group of Pharo's contributors from different projects in the community. The information was collected from four interviews made to community members of Pharo's community. The interview was divided in two parts, a conversational interview and an activity part.

1) *Participants*: The participants were all active contributors of the Pharo project, one undergraduate and two currently in a postgraduate program, and one a full time developer and researcher. All of them claimed to be "fluent" in Pharo programming language where two of them had Pharo as their main language; the participants had from 3 to 6 years of experience working on Pharo; and the four participants were all males. However, note that the different participants experiences are of great value for this study.

2) *Interview*: The interview was guided with a questionnaire first, and later the participant was asked to tell their perspective of how communication was perceive in Pharo's community, and the experiences the participant had. The interview data was processed using the Grounded Theory, a

qualitative data processing technique, which aims to describe a phenom from qualitative and unstructured data. Every interview was processed, defined later as concepts and then as categories or factors.

3) *Activity*: The activity part was also split in two sections. First, the participants were given the task to find experts while interacting with Pharo's integrated development environment. And as a second activity, every participant was given a piece of paper to fill it with a list of experts they knew by first hand, or experience, in the community, for different popular packages in Pharo.

## V. EXPLORATORY STUDY

We aim at identifying factors that influence developers to maintain expertise communication through an interview and an activity study. In this section we will answer the research questions proposed in this technical report, we made use of other researchers previous work to answer them.

**RQ1. What is an expert to a developer?.** The Merriam Webster defines expert, as to have, involving or displaying special skill or knowledge derived from training or experience, other sources do not differ from this definition. Expert is not an adjective used to define people with experience with skill in a particular topic that cross a defined threshold, experts are recognized in a sort of popularity, people describe experts from its subjective perspective [12].

In software development, some work tried to extract expertise from source code, analyzing analysis (usage expertise, 10 line rule, etc), and community analysis (bug reports, community reputation).

While processing the information, many participants related or recognized an expert by community participation. Even though two of them relied on code-ownership, they also agreed that code-ownership might provide wrong feedback to recognize actual experts.

We propose the next definition in terms of sync-communication: "An expert is a person that can help effectively with a problem the developer is facing at a certain time".

Some participants' declarations were:

- "I ask a question in the community and If someone response we know that this person is an Expert".
- "You can say that a person is someone who has collaborated with the code, but you have to corroborate this by asking him, do you feel an expert in THIS?".

**RQ2. Why software people aim to answer questions in a forum community?.** All participants agreed that they aim to answer questions of people that made contact with them, three have played the role of experts responding some questions and even offering help in such interaction with developers.

"Open source community often display a meritocratic hierarchy", experts are recognized in the community by reputation. Which will be built by participation and collaboration [12]. Although the participants did not recognize that they support this affirmation as experts, they acknowledge this sentence being a participant in the community, since experts are usually the people who has gain reputation through collaboration.

A contradiction was also revealed by the participants, not all experts aim to answer questions. All participants have experienced that some experts are rude, or they do not even answer questions, they claim that this behavior is common with some experts in the community. Also 2 participants mentioned that experts can get overwhelmed with questions, so they just cannot answer every question made by the users. This is clearly seen when a project has few core developers with multiple users, at some point they will not be able to handle every question made by the users. This problem will bring distrust in the user, and eventually the user will abandon the project.

**RQ3. Why developers ask questions maintaining private communication with an expert?.** Two participants made clear that maintaining private communication is a bad practice for the community, which also is against the philosophy of open source communities, the information that occurs inside the community must be open. The remaining two participants either affirmed or rejected this claim. All of them said that they rather ask to the public forum than to a specific person. Anyway, all of them have contacted a person (expert) directly before.

All the participants have maintained private communication with experts before, we'll see how true is that all of them rather ask questions in the public forum. This might seem confusing for two things. First, the concept of expert might have a higher expectation or even worshiped (which is different from the definition given before). Second, the question is not intended to an expert.

To answer the question it is necessary to reference work related in recommendation systems that enables engineers to easily discover and communicate with the people who have contributed to the source code throughout large parts of the project [8]–[10], remarking the work by Audris Mockus and James D. Herbsleb, with Recommendation System, the work in expertise recommendation systems were evaluated in a private software company with geographically distributed developers. This indicates us that direct communication is not only used inside Software Industry (Enterprises, open-source projects), at least in open source community, where everyone can collaborate with source code [12]. Open source projects start with one or a group of people, and if it the project is popular enough more contributors will join.

We conclude as follows, Direct/private communication is directly related with project collaboration, and split up in coordination and expertise communication [11]. We will support this affirmation describing answers from three of our participants, they accepted that they have maintain communication with an person, asking questions, where:

- Participant 1, maintained direct communication because, the work in which the participant was doing was relating to fix a bug in Pharo's bug-list.
- Participant 2, maintained direct communication with the main and maybe the only one contributor of a project, to ask something, and give a feedback about the tool.

- Participant 3, maintained direct communication to an expert because the work in question was related to private-industry project.

Also, all of them said that although they do contact experts, they might be careful and respectful with the time the expert spends helping the person.

**RQ4. What leads developers, when they look for help, to ask questions instead of going to other knowledge source?** Participants in this exploratory agreed that time is big factor for someone to ask questions in a forum or community, when some particular problem is too costly in time to solve. We know that from different questions posted in Stack Overflow, many of them are repeated [14], from this we conclude that many basic, or naive, questions might be answered already in the forum, but different other problems asked in the forum that might be more complex and need more time to solve need the help of an expert.

#### A. Study activities

For these study activities, the participants were asked to search for experts by their knowledge, and use any resource they have at hand.

**Task 1.** The participants were asked to find experts using the Pharo's integrated development environment (IDE), the most experienced ones found many developers in a short period of time, in the figure 1 you can find how many developers were found by each one. In this task all the participants coincided to look for experts in the source code repository to find the right expert. Even the most skilled developer has to struggle looking in each commit, looking in frequency and relevance of the commit. Discriminating between the lines of code changed, and the meaning of the commit comment. In every new task the participant processed, they claimed that the task was tiring.

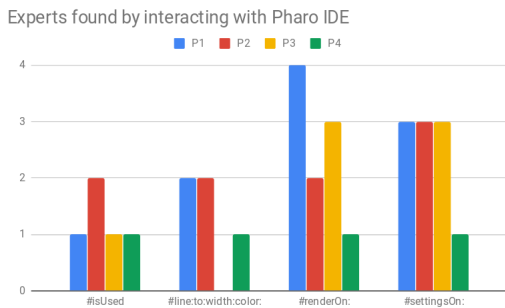


Figure 1. Results of experts found by each participant using Pharo IDE (task 1).

**Task2.** Clearly developers might already know one person that is an expert in Pharo, a mentor, by conferences or they might have collaborated together in some project. So the participants were asked to name different experts for different popular packages Pharo. Although the results shows that the participants respond experts by knowledge, they claim that even though the people they chose were not experts in that topic, they might provide a clue of where to start if they face

a problem, or they might give them the name of the expert that can help them; the majority of experts answered in this task were closer to the participants.

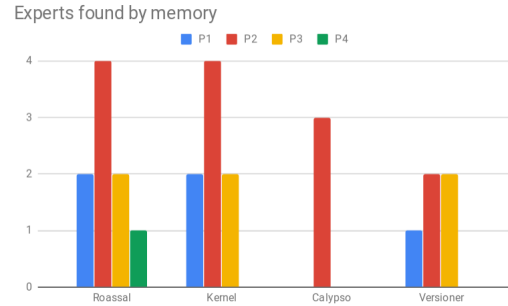


Figure 2. Results of experts found by each participant, using their experience in the community, or memory (task 2).

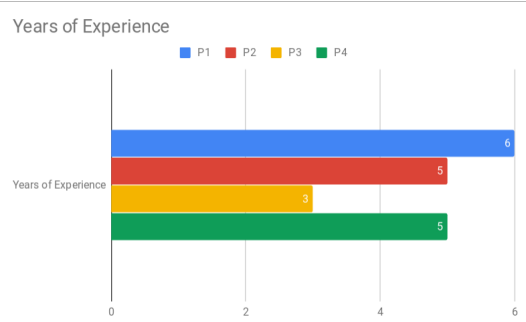


Figure 3. Experience years of each participant.

**RQ5. How difficult it is to find an expert to a developer?** The results of James D. Herbsleb and Audris Mockus [6] in their work on communication in globally distributed software development in 2003, showed that developers find difficult to find experts, they got their result from a study in a international company, and its developers. The conclusion this exploratory study show is that even with the big quantity of solutions to contact people remotely, developers still find difficult to find experts.

## VI. DISCUSSION

This technical report revealed the different factors to take into account at the time of studying expertise communication in open source communities. It has been shown that developers still don't know how to find an expert in the community, for this problem we propose as a solution a chat bot in the main communication channel of Pharo, which will contain the characteristic of giving developers a experts list for a particular piece of code (method, class or package) [1].

1) *Threads of validity:* Because it was an exploratory study, the limited number of participants and their experience limit this study's generalizability. Developers with more experience in the community may be different because of their work context and depth of experience. We plan to do an extensive study

with more interviews, literature review, and questionnaires in the open source community to validate the findings presented.

This exploratory study only considered one open source community as target, some of the findings would obviously be different if other communities and environments are targeted.

## VII. CONCLUSIONS

”Nurturing communication in software development is not about increasing the amount of communication but about increasing the quality of the communication experience in the context of software development” [11]. In this technical review we carried out a practical study to explore more about expertise communication in Open source community.

This work has given a glance of what an expert is, in developers perspective, and why do developers contact them, defined in terms of expertise communication, under which circumstances, and how do they do to contact them, and what factors trigger this communication. We conclude that developers use public and private communication for different purposes, specifically private expertise communication occurs when developers and experts work in a same project.

We noticed that although new systems provide as many information as possible, it seems that is still difficult for developers to find the expert they need [6]. To continue with this study, we anticipate that future interviews with developers would shed more light on the problems developers face on their daily working.

## VIII. FUTURE WORK

We created a chat bot to address the problem of expertise finding. As a future work we will give this bot the capability to manage different documentation of a Software project, answering different questions regarding documentation, source code and recorded questions in the forum or chat.

1) *Chat bot as experts recommendation system* : In order to address the problem of expert finding, a chat bot has been created as a recommendation system for experts in the Pharo project [1], you can find an example in the figure 4. This particular chat bot is different to other solutions because:

- This chat bot is implemented to find experts in an Open source project as Pharo.
- The interface presented to the users its more friendly because it uses natural language to receive questions. It answers questions as ”Who can help me with Roassal?”, and the bot responds a experts list of Roassal, where Roassal is a popular Pharo project. The chat bot can also respond to names of classes or methods.
- The chat bot is attached to the Server used as main channel of communication in Pharo’s community. It means that the bot is accessible from everyone in the community without the need of an installation.

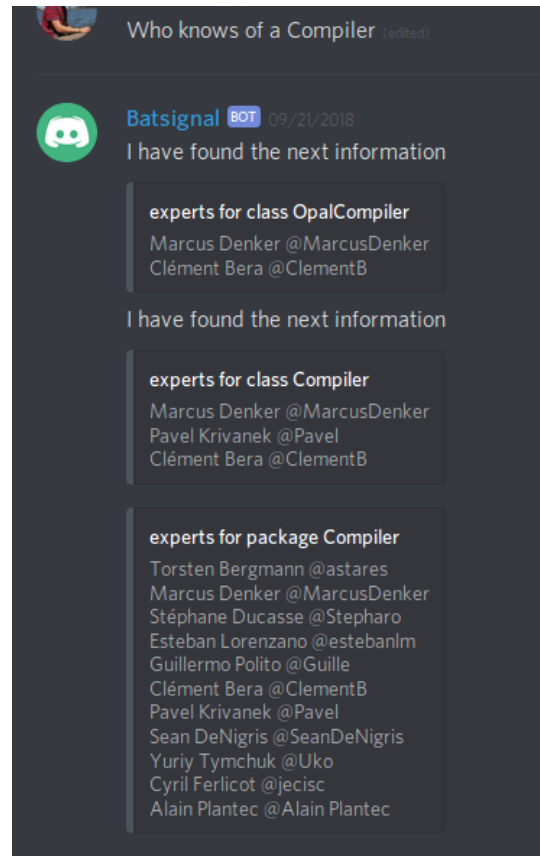


Figure 4. Chat bot answering a question.

2) *Chat bot as a documentation assistant*: We have shown along this project that communication must be nurtured. And in order to achieve this goal, we have to increase the communication quality by means of decrease unnecessary expertise communication. Expertise communication should not be promoted as the first choice; rather, it should be avoided when code, documents, previous communications, and/or other artifacts that satisfy the information needs are available. We aim that our solution will decrease unnecessary expertise communication.

## REFERENCES

- [1] Chat bot BatSignal main page. <https://jhoncc2.github.io/bat-signal/>.
- [2] M. O. Ahmad, V. Lenarduzzi, M. Oivo, and D. Taibi. Lessons learned on communication channels and practices in agile software development. In *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 929–938. IEEE, 2018.
- [3] L. M. Berlin. Beyond program understanding: A look at programming expertise in industry. *ESP*, 93(744):6–25, 1993.
- [4] S. Faraj and L. Sproull. Coordinating expertise in software development teams. *Management science*, 46(12):1554–1568, 2000.
- [5] A. Guzzi and A. Begel. Facilitating communication between engineers with cares. In *Proceedings of the 34th International Conference on Software Engineering*, pages 1367–1370. IEEE Press, 2012.
- [6] J. D. Herbsleb and A. Mockus. An empirical study of speed and communication in globally distributed software development. *IEEE Transactions on software engineering*, 29(6):481–494, 2003.
- [7] K. R. Lakhani and E. Von Hippel. How open source software works: “free” user-to-user assistance. In *Produktentwicklung mit virtuellen Communities*, pages 303–339. Springer, 2004.

- [8] D. Ma, D. Schuler, T. Zimmermann, and J. Sillito. Expert recommendation with usage expertise. In *2009 IEEE International Conference on Software Maintenance*, pages 535–538, Sept 2009.
- [9] D. W. McDonald and M. S. Ackerman. Expertise recommender: A flexible recommendation system and architecture. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, CSCW '00*, pages 231–240, New York, NY, USA, 2000. ACM.
- [10] A. Mockus and J. D. Herbsleb. Expertise browser: a quantitative approach to identifying expertise. In *Proceedings of the 24th International Conference on Software Engineering. ICSE 2002*, pages 503–512, May 2002.
- [11] K. Nakakoji, Y. Ye, and Y. Yamamoto. Comparison of coordination communication and expertise communication in software development: Motives, characteristics, and needs. In *JSAI International Symposium on Artificial Intelligence*, pages 147–155. Springer, 2009.
- [12] C. Shen and P. Monge. Who connects with whom? a social network analysis of an online open source software community. *First Monday*, 16(6), 2011.
- [13] M.-A. Storey, A. Zagalsky, L. Singer, D. German, et al. How social and communication channels shape and challenge a participatory culture in software development. *IEEE Transactions on Software Engineering*, (1):1–1, 2017.
- [14] C. Treude, O. Barzilay, and M.-A. Storey. How do programmers ask and answer questions on the web?: Nier track. In *Software Engineering (ICSE), 2011 33rd International Conference on*, pages 804–807. IEEE, 2011.
- [15] G. Von Krogh, S. Spaeth, and K. R. Lakhani. Community, joining, and specialization in open source software innovation: a case study. *Research policy*, 32(7):1217–1241, 2003.
- [16] S. Wagner and M. Ruhe. A systematic review of productivity factors in software development. *arXiv preprint arXiv:1801.06475*, 2018.
- [17] J. Zhang, M. S. Ackerman, and L. Adamic. Expertise networks in online communities: Structure and algorithms. In *Proceedings of the 16th International Conference on World Wide Web, WWW '07*, pages 221–230, New York, NY, USA, 2007. ACM.