Open Source beyond software: An empirical investigation of the open design phenomenon

R&D Management Conference 2009

Kerstin Balka, Christina Raasch, Cornelius Herstatt

30. April 2009

Abstract

This paper contributes to the literature on open source by providing a quantitative study ($N = 76$) of open design projects. Extensive research has been done to analyze the phenomenon of open source software from different perspectives, however very little research has been done with reference to open source development of tangible objects, so-called open design, so far. Until recently, limitations to the availability of successful empirical examples of this 'new innovation model' outside software may have been a key reason for this gap. Our goal is to develop an empirically founded perspective on those projects and to understand their contextual background.

Along a framework describing open source innovation we first show univariate statistical analyses to explore the variety of projects. We find that actors stem from different backgrounds and that projects vary considerably in terms of community size, complexity of the objects being developed, and innovativeness of the intended outcomes. We also observe different approaches towards the protection of intellectual property and the responsibility for development and production processes.

Bivariate analysis can partly explain the relations between project advancement and project characteristics. Open source development seems feasible for tangible products under a variety of different circumstances. However we identify some attributes favoring high stages of advancement among projects. Our results are partly in line with large-scale empirical studies on open source software, but we also point to potential discrepancies between open development of software and physical products.

1 Introduction

A striking phenomenon in recent years has been the rise of open source software (OSS). Source codes are freely revealed via the Internet, allowing geographically distributed programmers to download and utilize the software, to suggest improvements to the community, or to make modifications themselves and to redistribute their modified code. A large number of successful examples of open source software programs have been developed over the last years and extensive research has been done to analyze this phenomenon from different perspectives (for an overview see von Krogh and von Hippel, 2006).

In view of the success of OSS, experts from science and practice suggest a broader applicability of the 'open source model' of product development outside the software industry. To date, however very few studies have been conducted in fields other than software. While research on OSS may provide some clues for industries beyond software, Dahlander and Magnusson (2008) point out that it also faces clear limitations. Some recent research investigates open content, but there are hardly any studies on the open development of physical products, so-called open design.

Open design is being characterized by free revealing of information on a new design with the intention of collaborative development of a single design or a limited number of related designs for market exploitation (cf. Raasch, Herstatt, and Balka, 2009). Our present work concentrates on the enablers and inhibitors of open design and shows how projects are performing in practical applications. Our goal is to develop an empirically founded perspective on those projects and to understand their contextual background.

With this paper we want to provide a first statistical analysis of this largely uninvestigated phenomenon and disclose the variety of open design projects. Among others we address questions on who is launching the projects and who are the participants, how IP issues are being solved in practice and whether limitations in terms of the complexity or the innovativeness of products are present. The considered projects stem from very different industries and show diverse contextual and organizational backgrounds.

The paper is structured as follows: In the next section, we discuss the open source innovation framework and derive hypotheses on the relation between project advancement and certain project characteristics from the literature on OSS for application on open design. The third section describes our research design and data acquisition. The fourth section presents the findings organized along the open source innovation framework, and the fifth section discusses those findings and their implications in relation to the hypotheses. Finally, we conclude the paper and outline implications for future research.

2 Theoretical background

2.1 Open source beyond software

Open source (OS) is a development methodology (Stallman, 2007). The term originates from the software industry and denotes the revelation of the source code. Open source development is an example of the private collective model (Hippel and Krogh, 2003) and a form of open technology (Nuvolari and Rullani, 2007). In the scholarly literature, open source software (OSS) has been described as "new innovation model" (Osterloh and Rota, 2007), it has been referred to as the "community-based model" (Shah, 2005), as "opensourcing" (Agerfalk and Fitzgerald, 2008), or as the "open source method" of product development (Osterloh and Rota, 2007).

Shah (2005) considers OSS development as perhaps the most prominent example of the "community-based model", which extends well beyond the domain of software. In view of the success of this model in the software industry, an increasing number of experts from science and practice (e.g., Raymond, 1999a; Lerner and Tirole, 2004) suggest a broader applicability outside the software industry. As confirmation they refer to a small number of existing projects, e.g. biotechnology projects or the OScar project (Hope, 2007; Mueller-Seitz and Reger, 2008).

To generalize the 'OS model' to a non-industry specific level, we use the term "Open Source Innovation", which we define as follows: Open source innovation is characterized by free revealing of information on a new design with the intention of collaborative development of a single design or a limited number of related designs for market or non-market exploitation (cf. Raasch, Herstatt, and Balka, 2009). According to this definition open source innovation is characterized by a non-market, non-contractual transfer of knowledge among actors, sharing relevant information with a non-definite set of other actors without any immediate recompense. Actors share their ideas with the clear purpose of contributing to a joint development. Revealing of design information in order to gain reputation, build social status or other motives, without the understanding that this design is part of a larger design task, shall not be considered as open source innovation. The outcome is exploited, either for-profit or not-for-profit or both, in the sense that the design is produced and sold on a market, integrated into other products that are marketed, or deployed during the development of such products.

Looking at the breadth of open source innovation, a first distinction has to be drawn between intangible and tangible objects of development. In the non-physical world, beyond software, so-called open content is currently attracting considerable attention. Examples as the entire family of wikis, open science, educational materials, cultural goods such as music or films, geographic maps (the OpenStreetMap project), and other applications suggest that open content is a viable model and offers sustainable business opportunities to companies (e.g., Parker and Van Alstyne, 2005; OECD, 2007). In the realm of tangible goods, open design (OD) refers to the open development of physical objects. While much of the development work in this group can be accomplished virtually, the ultimate purpose is the design and production of a physical artefact. Examples range from beer to cars and from manufacturing equipment to IT hardware. Most of these projects are still in development, but nevertheless many have already marketed semi-final products or fully functional intermediate versions for several years.

As "hardware is becoming much more like software" (von Hippel in Thompson, 2008), it is important to point out that the border between open development of tangible and intangible objects is not always as clear-cut as it might appear. OSS is essentially based on sharing instructions that will be interpreted by a computer. Frequently also open design is based on developing and sharing designs and instructions to create physical objects (Smith, 2008). According to Shirky (2007), "An increasing number of physical activities are becoming so data-centric that the physical aspects are simply executional steps at the end of a chain of digital manipulation".

This paper focuses on open design rather than open content, and draws comparisons between open source development of tangible objects and software. For the empirical exploration we will use a conceptual framework proposed in Raasch, Herstatt, Balka, and Abdelkafi (2009). With regards to content we keep the 6 constituents, however we adapt the graphical representation to visualize the relations among the constituents.

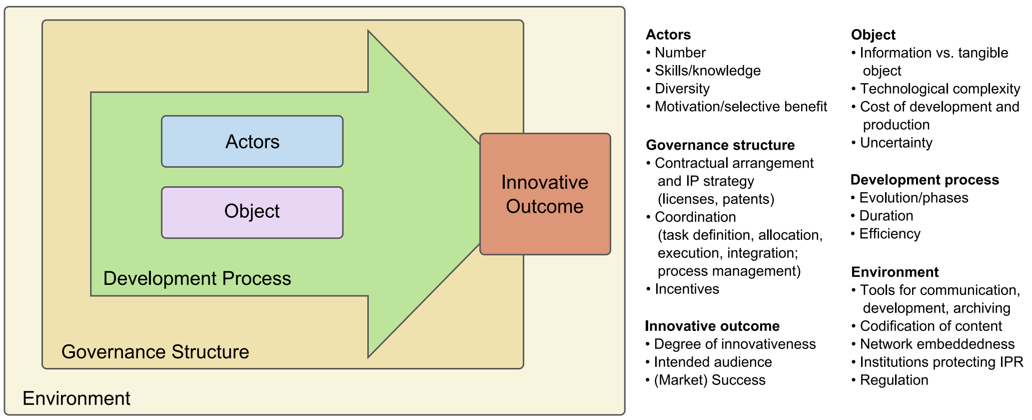

Figure 1: The open source innovation framework

According to this framework (Figure 1), actors, in a broader sense, including investors, manufacturers, etc., collaborate within a development process to work towards a design of an object and finally come up with an innovative solution for market exploitation. A suitable governance structure framing the development is devised and possibly adapted across the evolution of the development to organize the collaboration and provide required contractual arrangements. The object, given its inherent characteristics, can pose requirements to actors, development process and governance structure, whereas the latter are in turn actively shaped by the actors. The constitution of the governance structure can back-propagate to the group of actors, causing self-selection and affecting their effort invested, and to the development process, influencing its evolution and efficiency. Finally each project operates within a technical, social, economical and legal setting surrounding the development.

2.2 Empirical studies of open source software

A large number of successful examples of open source software programs have been developed over the last years and extensive research has been done to analyze this phenomenon from different perspectives (for an overview see von Krogh and von Hippel, 2006). Large-scale descriptive statistics visualizing the variety of projects are provided by Weiss (2005), who used "relatively easy accessibility of high volumes of information about open source software" from Sourceforge, a well-known repository currently hosting about 150.000 projects. The same data source is used by Comino, Manenti, and Parisi (2007) who analyze the relationships between the various characteristics of OSS projects and model their influence on project success. Based on similar data sets obtained from Sourceforge, among others, Crowston and Scozzi (2002) identify and test factors important for the success of OSS projects. Healy and Schussman (2003) analyze patterns in the overall structure of OSS development communities by comparing projects using different activity measures.

Numbering the 6 constituents of the open source innovation framework from (1) to (6) we will now compile 5 hypotheses derived from empirical studies on open source software. The hypotheses will be used to compare previous findings from OSS to our own empirical findings in the area of open design. Obviously we cannot cover all aspects in detail and need to focus on certain characteristics, which seem particularly interesting and appropriate for comparisons between OSS and open design.

(1) Actors: For OSS, Raymond (1999b) predicted and Comino et al. (2007) proved empirically that the size of the community has a positive im-

pact on project development status. Concerning the type of actors, Healy and Schussman (2003) observe that "successful OSS projects are most often staffed by professional software developers" and "are (more often than not) run by professionals"

For industries other than software a sufficient number of contributing developers (Shah, 2005) with the required skills (Lerner and Tirole, 2004) are mentioned in the literature as a precondition for open development to be feasible. For our analysis we arrive at two hypotheses:

H1: The size of the community is positively correlated with project advancement.

H2: The participation of commercial contributors is positively correlated with project advancement.

(2) Object: Many researchers agree that open development today is more easily applied to information goods rather than tangible objects (e.g., Baldwin and Clark, 2006). Due to the dissimilarity of the objects we refrain from explicitly comparing open source software characteristics to characteristics of tangible objects.

(3) Governance structure: Demil and Lecocq (2005) describe 'the OS model' as "a generic structure regulating transactions which could be employed in different industries" and frame governance in the dimensions of control, incentives and a contractual framework. As a fundamental attribute of open source we focus on the contractual framework, i.e. the license regulating the revelation and use or re-use of developed knowledge. OS licenses are usually classified according to the restrictions they impose on derivative works (Bonaccorsi and Rossi, 2003). Investigating the impact of different licensing models Comino et al. (2007) showed that projects distributed under highly restrictive regulations are less likely to reach an advanced stage of development. This relationship shall also be examined in open design projects:

H3: Highly restrictive licenses are negatively correlated with project advancement.

(4) Development process: Due to the evolving nature of both open design and open source software over time it is difficult to reason about their evolution using data only from a given moment in time (c.f. Comino et al., 2007). Research on the development process, its stages and efficiency is still in its infancy. Hence we restrain our hypothesis in this category to the intensity of the co-operation. Crowston and Scozzi (2002) find that OSS projects with higher activity are in a more advanced stage of development. Therefore, we propose for open design:

H4: Activity is positively correlated with project advancement.

(5) Innovative outcome: Different studies have shown that relevant rea-

sons for contributing to an OS project are related to the desire of learning and performance improvement (Rossi, 2004). Sophisticated applications targeted towards an advanced audience may on average offer more learning opportunities to the individual contributor than projects addressing end users directly. Accordingly they may have a higher likelihood of receiving substantial contributions from developers. In the field of OSS Comino et al. (2007) show that those applications are indeed more likely to reach an advanced stage of development. Therefore, we will study for open design whether:

H5: Addressing an advanced audience is positively correlated with project advancement.

(6) Environment: Several environmental factors have been found to support or hinder open development. A precondition for the feasibility of OSS development is the existence of strong supporting tools, in particular the Internet, that are easily accessible to the developer community. In our analysis we study the use of different tools for communication in the different communities. However due to the wide scope of environmental factors and the lack of suitable systematic empirical studies, we will not phrase an explicit hypothesis relating environmental factors to project success.

3 Approach and methodology of empirical research

To collect data concerning practical applications we launched a community-based directory of open design projects on August 26, 2008. Registration is for free and participants are encouraged to contribute by entering new projects as well as by maintaining all the information up-to-date. A discussion forum completes our attempt to create a platform allowing better connection between projects and simplifying information exchange.

Our effort has been quickly embraced by the larger open source community. As of February, 2009, we count 44 registered members and observe fairly satisfactory numbers of visitors, exhibited in the left part of Figure 2. The chart shows the number of unique human visitors per month and visualizes a strong increase of attention over the last 6 month.

To our knowledge, our site (open-innovation-projects.org) contains the largest existing online directory collecting and providing information about open design projects. During the first 6 months, 121 project entries have been created, of which we need to exclude 14 due to complete inappropriateness. Accordingly we arrive at 107 relevant projects. Every entry is carefully checked and only approved information is visible on-line and integrated into

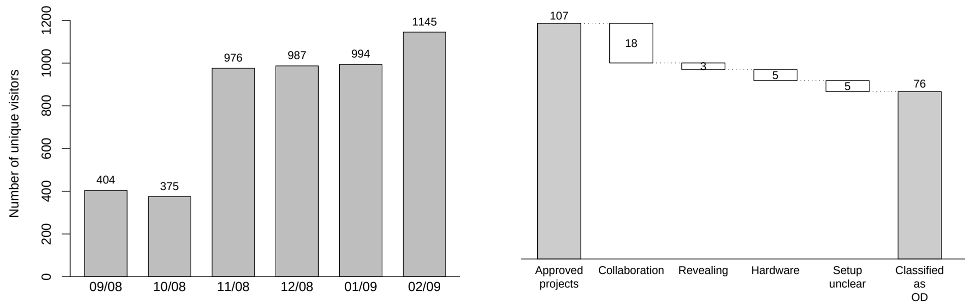

Figure 2: Upgrowth of unique human visitors from September, 2008 to February, 2009 (left) and break down of approved projects on reasons for exclusion and selected cases (right)

the database. This quality check not only eliminates spam entries; it primarily avoids that information is purely based on declarations of the project administrators rather than on objective measures. Missing data is filled as far as the respective information on projects is available for us. In some cases projects have been contacted and asked to provide specific details.

In a second step, we analyzed each entry to check conformity with the definition of open design. As shown in the right section of Figure 2, 76 projects have currently been identified as such and 31 projects have been excluded due to the following reasons:

- In 18 approved projects we observe pure revealing of information without the intention for collective development required by the definition of open source innovation. Examples are communities of hobbyists exchanging their ideas and instructions.
- 3 projects fail for the second part of the definition, because they do not freely reveal relevant information. Those projects could be tagged "Open Innovation" according to Chesbrough, Vanhaverbeeke, and West (2006).
- Another 5 cases have been excluded due to our focus on physical goods. Those cases include projects as, for example OpenStreetMap and Open Source Cookbook, which should rather be considered open content than open design.
- 5 entries could not yet be classified, because their approaches towards collaboration and free-revealing became not yet clear.

Validity and limitations of the data base

With 76 open design projects we established a reasonably large directory, but still 76 is a small number compared to about 150.000 OSS projects hosted at Sourceforge. In the current situation we do not have a clear view of the size of the total population of open design projects; accordingly we have to be careful when generalizing the results.

Concerning the actual data, we have to remark that our data base is not fully filled and that data is missing, which could not have been obtained on an outside-in basis. Also our quality checks are conducted outside-in, which allows for adjustments from an independent perspective. However potentially wrong inputs can not be corrected if the accurate data is not available from outside.

Since our directory is fairly young, we only started to get input from the broader open source community. However with our effort we already reveal a striking variety of projects and gain fairly deep knowledge about project and product details. In the following chapter we present a number of statistics demonstrating the diversity of projects and providing insights into their characteristics.

The data presented has been obtained as of February 28, 2009. For the purpose of statistical analysis we chose the software package 'R'. For detailed explanations of variables please refer to appendix A.

4 The variety of open design

In accordance with the selection criteria discussed in the previous section, we examine 76 open design projects along the framework presented in section 2.1, linking (1) actors, (2) artifacts, (3) governance structure, (4) development process, (5) innovative outcome, and (6) project environment. Each of these encompasses multi-faceted issues reaching beyond the scope of this paper. In order to give an exploratory overview, we will analyze all six parts, but focus on some aspects of particular relevance. We present univariate statistics using barcharts which show the number of projects per respective category. Due to data availability constraints not all data fields could be filled for some projects, which is why most plots do not sum to 76. Subsequently we focus on multivariate statistics in the last part of this section.

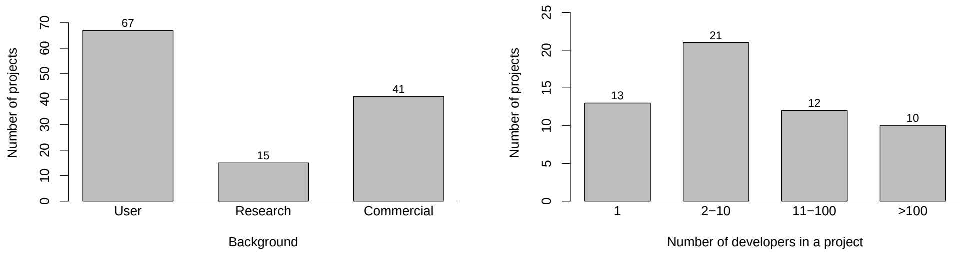

Figure 3: Background of contributing actors (left, multiple answers possible) and distribution of number of developers in projects (right)

(1) Actors: Stem contributors from private or commercial background and how many participate?

Contributing actors stem from different backgrounds as shown on the left side of Figure 3. Different types of actors can contribute to the same project, hence the categories are not disjoint. The data contain 8 projects receiving substantial contributions from all three types, 36 projects from a combination of user and commercial actors, 11 projects from user and research and 10 projects from commercial and research.

Open design projects vary considerably in terms of community size and, more particularly, the number of developers. The right part of Figure 3 shows how the number of projects is distributed on the number of contributors. While obviously a number of project leaders did not manage to attract any further contributors, other projects have to deal with the coordination of hundreds of, in some cases even more than 1.000, active developers. As in similar analyses in the field of OSS, the chart suggests a left-skewed distribution.

(2) Object: What types of products are developed?

In all considered cases, the artifact being developed is a tangible object, but information on the artifact is frequently digitalized for exchange during product development. Many projects incorporate written software code as a substantial part. We therefore categorize projects according to the type and amount of code that has to be developed (Figure 4, left). The first category 'Extensible trough software' includes all products whose functionality can be extended trough software applications, e.g., laptops, mobile phones, and programmable robots. The second category 'Includes programmed components' contains products which require software to fulfill their intended function-

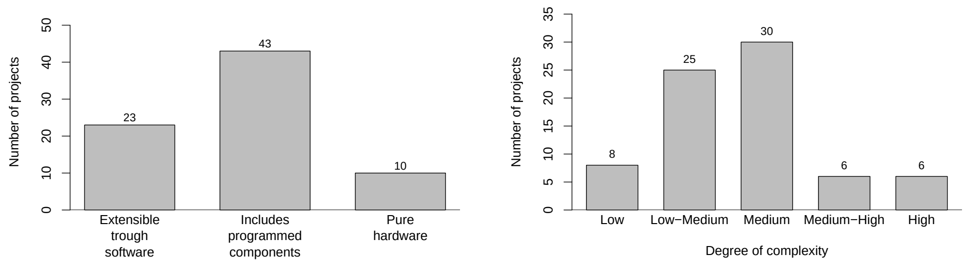

Figure 4: Importance of source code across projects and distribution of complexity of the final product

ality[1], but whose functionality cannot be extended trough software applications, e.g., printers, cars, and medical prosthetics. A laptop, for example, can gain complete new functionalities by installing new software programs, whereas a printer, for example, will only be able to print and its functionality cannot be extended through adding software applications. 'Pure hardware', the third category, includes all products which do not need a single line of code, e.g., beverages and clothes.

With 23 projects in the first category, about 1/3 of the examined projects has strong analogies to OSS. The product might supply, for example, a software development kit (SDK) allowing developers to create suitable applications. However we observe important differences to pure software development, as, for example, contributors need access to detailed hardware information in order to extend a product's functionality.

More than half of the cases belong to the second category, where hardware development plays the major role and software development fades into the background, but remains important to control functionality. Both information on hardware, e.g. in the form of descriptions, specifications, schematics, etc., and software code are exchanged.

In the third category the digitalization of information happens, for example, in terms of construction manuals, sewing instructions or recipes.

The right part of Figure 4 shows the distribution of different complexity levels ranging from as simple as beer and lights to as complex as cars and intricate machines, visualizing that all levels of complexity are being tackled by open design projects.

[1]covering microprocessors, digital signal processors (DSPs), and programmable logical devices, like field-programmable gate arrays (FPGAs), requiring written source code in a programming or description language to fulfill their intended functionality

(3) Governance structure: How do projects protect intellectual property?

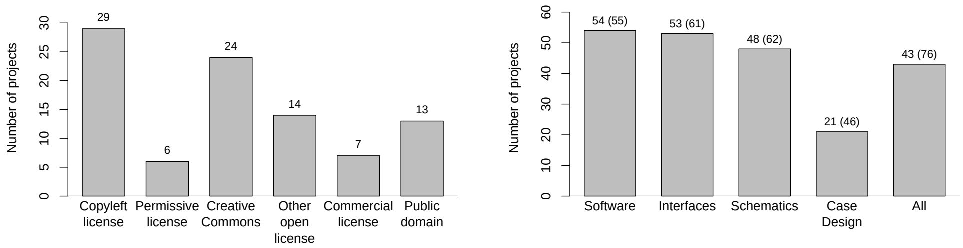

Figure 5: Type of selected license (left,multiple answers possible) and degree of openness (right)

As Figure 5 (left) points out open design communities frequently make use of an open license. The two major categories of free software licenses are copyleft and permissive or non-copyleft. Where copyleft licenses such as the GNU General Public License (GPL) are highly restrictive and insist that modified versions of the program must be free software as well, permissive licenses as the Berkeley Software Distribution (BSD) license are less restrictive and allow modifications to remain closed-source. Beyond software Creative Commons provides several free licenses that copyright owners can use when releasing their work. These licenses delimit certain rights to the use of the work and differ in their restrictiveness, depending on the chosen type of license.

We observe that many projects make use of software licenses for software components. Some use Creative Commons licenses for hardware parts, but in a number of cases intellectual property is released into the public domain without protection. In addition 24 cases ($\sim$ 30%) decided to protect their name by registering a trademark.

We further observe that many projects carefully select which knowledge to keep secret and which parts to reveal freely. The right chart of Figure 5 shows the number of projects revealing information across four categories - software, interfaces, schematics, and case design, and summarizes these categories in the last stack, which presents projects revealing all relevant information across all four categories. The numbers in brackets indicate the number of projects relevant for this category. By relevant we mean that the respective artifact includes parts of this category and development work on those parts has already started and the question of revelation thus arises.

This analysis reveals that it is common across all examined projects which include software development to open at least parts of their software. By contrast, about three quarters publish schematics or similar information and only less then half of the relevant projects decided to release their case design. Assuming that the revelation of knowledge in general is based on conscious decisions, one may conclude that projects typically gain more advantages from publishing software source code than from publishing any other parts of their work.

(4) Development process: Who is responsible for development and production processes?

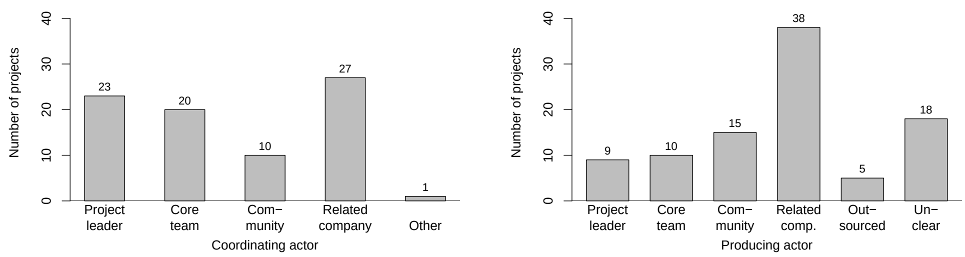

Figure 6: Distribution of actors promoting and coordinating the development (left) and of producing actors (right), multiple answers possible

At this early stage it is hard to capture process evolution and efficiency on a quantitative basis. With Figure 6, illustrating which group of actors is mainly driving a project and which takes the role of the producer, we gain first insights on how development and production processes may be characterized. Projects which have not yet started the production of prototypes or final products are tagged 'Unclear' in the right chart.

The chart visualizes that already for a large portion of examined projects the development is driven by a company, and even more products are marketed and produced by a related company. Correspondingly the plot shows a fairly high number of projects whose product development is driven by private contributors, i.e. by the project leader, the core team or even the larger community without a dedicated authority. In about half of the respective cases this group also acts as producer, the second half interacts with a company supporting production and marketing. Furthermore we observe 5 projects outsourcing the production of the entire product.

More details on the development process, in particular on the intensity of co-operation within projects, will be discussed in section 4 using multivariate statistics.

(5) Innovative outcome: How advanced are the observed projects, who are their target customers and how innovative are the intended outcomes?

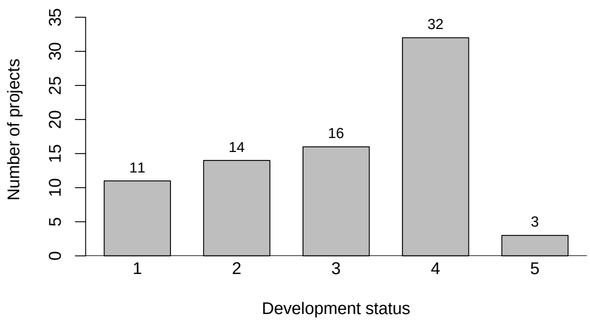

Figure 7: Distribution of stages of advancement in projects: 1 - Planning/Virtual development, 2 - Prototyping started, 3 - First working prototypes, 4 - Production stable, 5 - Mature

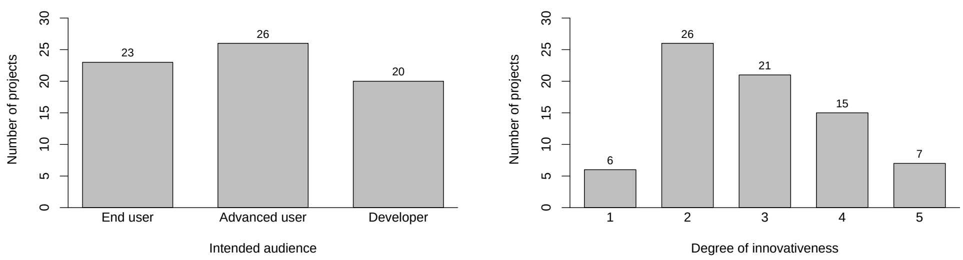

Figure 8: Distribution of intended audience (left) and degree of innovativeness: 1 - Imitative innovation, 2 - Incremental innovation, 3 - Discontinuous innovation, 4 - Really new innovation, 5 - Radical innovation (right)

Figure 7 shows that most cases have not entirely completed development, but many have reached a stable production stage and are marketing their products. One could argue that the distribution is skewed due to the fact that our database is fairly new and we started by entering projects known to us or other experts we talked to, which naturally are larger and more advanced projects. However one could also argue the other way around as new and small projects profit more from our directory and have accordingly a larger incentive to participate. In conjunction we assume that the database is fairly representative.

As development success is hard to evaluate across projects in different stages of advancement, the rest of this section remains limited to some preliminary findings on the intended audience and the degree of innovativeness (Figure 8). The distribution of projects across target customers is fairly uniform, which leads to the assumption that open design products are generally suitable for end users, advanced users, and developers. For the categorization of innovativeness we follow Garcia and Calantone (2002) and evaluate the degree of innovativeness in five categories. Our cases reveal that open design is applied to the whole spectrum from the generation of incremental to radical innovations with a higher portion tackling incremental and discontinuous innovations.

(6) Environment: From which industries and countries do projects originate and which communication tools enable open design?

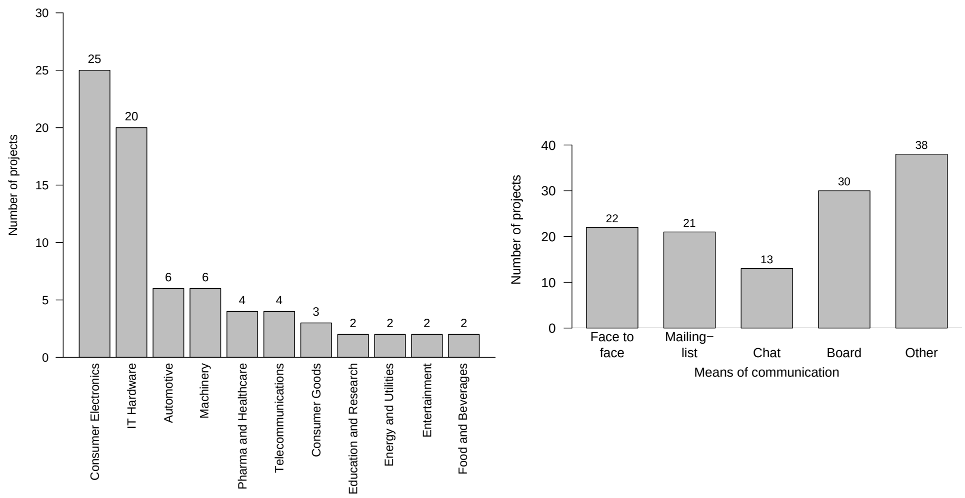

Figure 9: Distribution of industries (left) and means of communication (right, multiple answers possible)

The project environment spans a wide range of topics, from technologies and tools to the legal and social framing. The left side of Figure 9 illustrates the variety of industry backgrounds of our projects, visualizing that open design is already being applied in a number of different industries ranging from very traditional industries as 'Food and Beverages' or 'Energy and Utilities' to newer industries as 'Consumer Electronics' and 'Telecommunications'. Not graphically shown are the projects' home countries: about 50% of our collected examples stem from the United States, approximately $10-15\%$ each from Germany and the United Kingdom; in total about 12 countries have been named.

Another point worth investigating is the use of various tools for communication, development and archiving (Figure 9, right) allowing participants to overcome constraints on communication and to access shared digital resources. As different actors seem to prefer different communication tools, most projects use more than one tool. Our data set contains 2 projects using tools from all 5 categories, 5 projects from 4 categories, 9 projects from 3 categories and 17 projects from 2 categories. The last stack 'Other' includes mostly newer communication technologies. Included are wikis and shared workspaces facilitating the cocreation of content across large, distributed sets of participants, blogs and podcasts offering individuals a way of sharing information with a broad set of other individuals, and social networks or hosting providers as Sourceforge, offering contact management and access to hosting services.

Multivariate analysis

To gain first insights on potential relationships between variables we examine correlation coefficients and multivariate comparisons using relative measures as in Figure 10, where the diagrams are scaled to 100% to show trends in the data set. Correlations (ρ) are calculated as Pearson product-moment correlation coefficients giving a value between -1 and $+1$ inclusive. Significance levels of estimated correlations are obtained via analysis of p-values of two-sided tests for association between paired variables in order to confirm or reject the hypothesis that the underlying correlation is not equal to zero. In case of missing data, the respective pairs are excluded from the calculation.

The left chart of Figure 10 visualizes the positive correlation between the number of active developers and stages of advancement of $\rho = 0.4$ confirmed by a p-value below 1%. One could conclude that larger projects reach higher stages of advancement or, vice versa, that projects grow while maturing. The correlation is even more pronounced when contrasting the total community size, including passive consumers and lurkers, and the development status ($\rho = 0.6$ with p-value $< 1\%$).

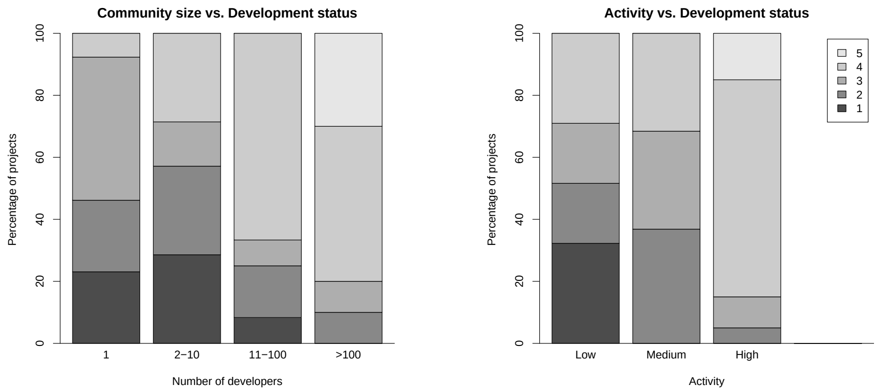

Figure 10: Comparison of developer community size (left) and activity (right) against development status; 1 - Planning/Virtual development, 2 - Prototyping started, 3 - First working prototypes, 4 - Production stable, 5 - Mature; in percent

The right chart visualizes the positive correlation between developer activity, measured by the frequency of communication and distinguishing between low, medium and high, and stages of advancement of $\rho = 0.5$ with a p-value below 1%. High activity appears to have a positive impact on development status. Not surprisingly we also find a positive correlation between the number of contributors and their activity ($\rho = 0.5$ with p-value $< 1\%$). It seems reasonable that more developers generate more activity.

More details on correlations between selected attributes are summarized in Table 1. Exhibited are pairs of variables with absolute correlation values above 0.25 and high significance, i.e. p-values below 5%, which point to relations between different constituents of our underlying framework. Strong correlations between different variables of the same aspect, as it would be the case for open interfaces and open schematics, for example, shall not be the focus of this analysis. Due to space constraints in the table they are only partly exhibited.

Remarkable is the strong positive relationship between development status, the presence of commercial contributors and a registered trademark. Commercial actors seem to favor to protect their work by registering the projects name, which is less often the case for private or research actors. Furthermore commercial contributions seem to have a positive impact on stages of advancement, a correlation which has to be further examined as the relationship might also be due to companies joining projects in later stages or projects founding companies in order to market their products. The cor-

	Comm.	C.S.	Comp.	P.d.	R.Tr.	E.o.	Act.	D.st.	In.
(1) Commercial	1								
(1) Com. Size	0.5**	1							
(2) Complexity	0.1	0.3*	1						
(3) Public domain	-0.1	-0.2	0	1					
(3) Reg. trademark	0.5**	0.5**	0	-0.2*	1				
(3) Entirely open	-0.3**	-0.4**	0	0.2	-0.4**	1			
(4) Activity	0.3*	0.4**	0	0	0.4**	-0.1	1		
(5) Dev. Status	0.5**	0.5**	-0.3*	-0.3*	0.4**	-0.2	0.5**	1	
(5) Innovativeness	0	0.2	0.4**	0.2	-0.1	0.2*	0.1	0	1

** 1% and * 5% refer to the significance levels for the estimates

Table 1: Estimated correlations between selected variables

relation between late stages of advancement and registered trademarks may be easily explained as it might make more sense to protect a project's name if success is augured.

Consequently, one might expect a negative correlation between high stages of advancement and release into the public domain without licenses, which is also confirmed by our analysis. In addition, we observe a negative correlation between entirely open projects and the presence of commercial contributors as well as registered trademarks. Both the absence of licenses and complete openness might reflect a general absence of IP strategies which seems to have a decelerating effect on project advancement and is rarely observed if commercial actors are involved.

We also observe a positive correlation between the degrees of complexity and innovativeness. As both categories might be influenced from the subjective perspective of the person entering the data, one could assume that the relation arises from biased entries. However to avoid this type of bias we carefully checked every data entry from outside and only approved entries are being considered in our analysis. Accordingly we arrive at the observation that projects developing highly complex products seem to or at least plan to achieve more innovative outcomes.

Furthermore we observe high positive correlations between the size of the developer community and commercial actors and registered trademarks, as well as between activity and registered trademarks, and a high negative correlation between community size and complete openness. These correlations are closely connected to the relations discussed earlier in this section and hint to high interrelations between the different variables. Closer investigation of dependencies among variables is required in order to arrive at secure evidence about actual relationships among the constituents of the open source innovation framework in Figure 1.

5 Discussion of results and comparison to open source software

Open source development seems feasible for tangible products. More and more physical products are being designed collaboratively via the Internet, but it is mostly early days to evaluate success. There are several promising precedents which have entered the market, though. In this section we discuss first implications of our findings for the hypotheses from section 2.2 and point out ancors for further research.

Compared to OSS repositories such as Sourceforge, our sample does not contain the large portion of projects with one developer, no discussion and no interest from the larger community. Healy and Schussman (2003), for example, phrase the conclusion that for every successful OSS project there are thousands of unsuccessful ones. We are not claiming that this phenomenon might be different for open design; however, we have so far not observed as many of those projects. This might partly have its seed in the definition of open source innovation, requiring the intention of collaborative development for market exploitation, and partly arise from the fact that the open design phenomenon and more particularly our directory is fairly new and accordingly the number of failed and inactive projects is low compared to large OSS repositories.

(1) Actors: Quite a few open design projects manage to attract a sufficiently high number of active contributors, both from private and commercial backgrounds, to build a developer community and to achieve progress in terms of project advancement. Our multivariate analysis confirms *H1* for tangible products, showing a clearly positive correlation between the size of the developer community and project advancement. As discussed in the previous section this might arise from different factors. On the one hand one might guess that projects tend to grow over time and it could be easier for successful projects to attract considerable attention from the broader developer community; on the other hand a larger community might push ahead development. In the software realm, Krishnamurthy (2002) found no relationship between the release date and the number of developers associated. If this finding holds for open design, one may arrive at the conclusion that indeed a larger community positively influences project advancement.

Also *H2* gets confirmed by our analysis revealing a positive correlation between commercial actors and project advancement. Taking both observations together, we find that projects with a large community, which includes commercial actors or is even organized by professionals, have a high likelihood of reaching advanced stages of development.

(2) Object: All levels of complexity can be observed across the examined artifacts and no strong evidence could be found pointing out that low- or high-complexity products are more suitable for open source development. We observe a slight negative correlation between the level of complexity and project advancement. First indications point to the assumption that this relation is driven by some highly complex products, which have not yet taken off and are still in the planning stage. However closer analysis of the individual cases is required to examine the impact of object complexity.

Although the comparison of object characteristics between open design and OSS is only meaningful to a certain extent, we want to highlight some particular points. Similar to OSS, we observe that, complex objects get modularized into manageable pieces and developed separately. Participants make a large effort to enable digital design and development as far as possible. In addition the development of 3D printers, CNC cutters and similar tools for home use increasingly enables people to produce their own designs independently of a central production. With the emergence of communities around the necessary equipment to share expenses and ease access, the production of open design products based on a set of instructions shows a strong parallel to OSS (c.f. Smith, 2008). Further examination of this phenomenon is clearly required in order to assess its potential to reach a larger audience and estimate its impact on open source development beyond software.

(3) Governance structure: Open design projects generally tend to make use of an open license, but licensing is less straightforward than for OSS. *H3*, proposing that projects distributed under highly restrictive terms are less likely to reach an advanced stage of development, is not reflected in our data set. We rather observe a correlation between trademark protection and late stages of advancement, and an interrelation between free release into the public domain and early development stages. As discussed this might reflect the general existence or absence of IP strategies and their influence on project success. However, we could not determine a clear relationship between sophisticated strategies towards revealing of certain components under certain rights and development status.

(4) Development process: Across our case database we observe different groups of actors being responsible for the creation of a product concept, the actual development work, and the final production, but find no formally distinguishable patterns. We do find a strong correlation between the intensity of developer activity and development stages, confirming *H4*. As a result we conclude that both OSS and open design projects with more activity tend to be in more advanced stages of development. However, in both fields further research is required in order to arrive at conclusions about the impact of process design on project success.

A major difference between software and tangible products obviously lies in the necessity of actual production prior to marketing a product. Our observation that many projects buy parts in addition and some even decide to outsource production of the entire product requires projects to thoroughly consider strategies towards market transactions. The tradeoff between increased efficiency through adding readily available components and increased freedom of design, in particular towards openness as external parts are frequently non-open source, seems to be an important topic within many communities.

(5) Innovative outcome: Open design projects tackle both incremental improvements and radically new designs and for both cases our data provide examples with fairly high development status. Similarly we observe projects, which are in late stages of advancement, both addressing developers or advanced users and developing products suitable for end users. *H5* proposing that applications for sophisticated users are more likely than others to reach an advanced stage of development is not supported by our data. Given that Comino et al. (2007) derived their hypothesis from findings on the motivation of participants, our observation could hint to differences in the reasons for contributing between open source software and tangible goods. Clearly further investigation is required prior to deriving conclusions on this topic.

(6) Environment: Our examples stem from a variety of different industries with a large subset in consumer electronics and IT Hardware. The development of these products does not only typically involve a large amount of software development, but many of these products are also shapeable through software. This characteristic makes them very attractive for open source software enthusiasts. For instance, by changing a mobile phone's software users can add new functionalities and customize their device to a high degree; and through access to hardware interfaces developers can modify the purpose of a chip. Both examples illustrate potential reasons for the desire to gain access to relevant information. This may explain the rise of open development activity in those industries.

A precondition for the feasibility of the OSS phenomenon is the existence of strong supporting tools, in particular the Internet, that are easily accessible to the developer community. For open design, especially in the technical domain, new suitable tools seem to be of particular importance to enable digital development of tangible artifacts. Our study leaves the assumption that the availability of such tools is rising, which has to be further examined. We encountered, for example, free or cheap design software, platforms for the exchange of designs, and open source equipment for prototype production.

6 Summary and conclusions

In this paper, we investigated a comprehensive data base of open design projects, with the aim of showing and comparing selected project details. Cases have been identified according to the definitions of open source innovation and open design and carefully maintained during data collection. Along the open source innovation framework we presented a number of statistics describing common patterns among our investigated examples. A principal finding was that open design is already being implemented in a substantial variety of projects.

Open source development of tangible goods is not a new, but a rapidly evolving field and more and more physical products start being designed collaboratively via the Internet. Contributors from private and commercial backgrounds form communities with sizes varying from 1 to several hundred developers and tackle the development of products across all degrees of complexity and innovativeness. The considered projects stem from various industries and show diverse contextual and organizational backgrounds. Their current development stages range from the evolvement of first rough ideas to mature and successfully marketed products. The largest proportion of projects already has fully functional products permanently available on the market, but is still working on further development.

Hypotheses derived from the literature on open source software were tested in the context of open design using our data set. We found strong relationships between the stage of advancement of the development and the size of the developer community, the presence of commercial contributors, and the intensity of co-operation, respectively.

We indicated that, in open design communities, tangible objects can be developed in very similar fashion to software; one could even say that people treat a design as source code to a physical object and change the object via changing the source. This suggests the practical applicability of transferring the 'Open Source model' to different industries beyond software. The success of OSS warrants a closer investigation of its potential to generate innovative and commercially viable products. With our contribution we hope to highlight the demand and smooth the way for further research.

References

Agerfalk, P. J., and Fitzgerald, B. (2008). Opensourcing to an unknown workforce: Exploring opensourcing as a global sourcing strategy. *MIS Quarterly*(forthcoming).

Baldwin, C. Y., and Clark, K. B. (2006). The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science*, *52*(7), 1116-1127.

Bonaccorsi, A., and Rossi, C. (2003). Licensing schemes in the production and distribution of open source software: An empirical investigation. *SSRN eLibrary*.

Chesbrough, H., Vanhaverbeeke, W., and West, J. (2006). *Open innovation - researching a new paradigm.* Oxford: Oxford University Press.

Comino, S., Manenti, F. M., and Parisi, M. L. (2007). From planning to mature: On the success of open source projects. *Research Policy*, *36*(10), 1575-1586.

Crowston, K., and Scozzi, B. (2002). Open source software projects as virtual organisations: compentency rallying for software development. *IEE Proceedings Software*, *149*(1).

Dahlander, L., and Magnusson, M. (2008). How do firms make use of open source communities. *Long Range Planning*, *41*, 629-649.

Demil, B., and Lecocq, X. (2005). *Neither market nor hierarchy or network: the emerging bazaar governance.*

Garcia, R., and Calantone, R. (2002). A critical look at technological innovation typology and innovativeness terminology: a literature review. *Journal of Product Innovation Management*, *19*, 110-132.

Healy, K., and Schussman, A. (2003). *The ecology of open-source software development* (Tech. Rep.). Department of Sociology. University of Arizona.

Hippel, E. von, and Krogh, G. von. (2003). Open source software and the "private-collective" innovation model: Issues for organization science. *Organization Science*, *14*(2), 208-223.

Hope, J. (2007). Pharmaforschung mit open source-methoden. In B. Lutterbeck, M. Brwolff, and R. A. Gehring (Eds.), *Open source jahrbuch 2007 - zwischen freier software und gesellschaftsmodell.* Berlin: Lehmanns Media.

Krishnamurthy, S. (2002). Cave or community? an empirical examination of 100 mature open source projects.

Lerner, J., and Tirole, J. (2004). The economics of technology sharing: Open source and beyond. *Harvard NOM Working Paper No. 04-35*. Available from `http://ssrn.com/abstract=620904`

Mueller-Seitz, G., and Reger, G. (2008). Open beyond the source code? - an explorative examination of the development of an open source car. *In Proceedings of the EIASM IPDM Conference*.

Nuvolari, A., and Rullani, F. (2007). Curious exceptions? open source software and "open" technology. In K. St.Amant and B. Still (Eds.), *Handbook of research on open source software: Technological, economic, and social perspectives* (p. 227-236). Hershey, New York: Information Science Reference.

OECD. (2007). *Giving knowledge for free: The emergence of open educational resources.*

Osterloh, M., and Rota, S. (2007). Open source software development - just another case of collective invention? *Research Policy*, *36*, 157-171.

Parker, G., and Van Alstyne, M. W. (2005). Two-sided network effects: A theory of information product design. *Management Science*, *51*(10), 1494-1504.

R Development Core Team. (2005). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Available from http://www.R-project.org

Raasch, C., Herstatt, C., and Balka, K. (2009). The open source model beyond software: Comparative case studies on the open design of tangible goods. *R&D Management, forthcoming*.

Raasch, C., Herstatt, C., Balka, K., and Abdelkafi, N. (2009). The open source model - beyond software. *EURAM conference paper*.

Raymond, E. (1999a). Afterword: Beyond software? Available from http://www.catb.org/∼esr/writings/cathedral-bazaar/afterword/ (Version 3.0)

Raymond, E. (1999b). The cathedral and the bazaar. *Knowledge, Technology, and Policy*, *12*(3), 23-49.

Rossi, M. A. (2004). Decoding the f/oss puzzle: a survey of theoretical and empirical contributions. *Department of Economics University of Siena, Working Paper No. 424*.

Shah, S. K. (2005). Open beyond software. In D. Cooper, C. DiBona, and M. Stone (Eds.), *Open sources 2.* Sebastopol, CA: O'Reilly Media.

Shirky, C. (2007). Generalizing peer production into the physical world. Available from http://finance.groups.yahoo.com/group/decentralization/message/6967

Smith, Z. (2008). Objects as software: The coming revolution. *25th Chaos Communication Congress*. Available from http://events.ccc.de/congress/2008/Fahrplan/events/2781.en.html

Stallman, R. (2007). Why open source misses the point of free software. Available from http://www.gnu.org/philosophy/open-source-misses-the-point.html

Thompson, C. (2008). Build it. share it. profit. can open source hardware work? *Wired Magazine*. Available from http://www.wired.com/print/techbiz/startups/magazine/16-11/ff_openmanufacturing

von Krogh, G., and von Hippel, E. (2006). The promise of research on open source software. *Management Science*, *52*(7), 975-983.
Weiss, D. (2005). *A large crawl and quantitative analysis of open source projects hosted on sourceforge* (Research Report RA-001/05). Institute of Computing Science, Poznań University of Technology, Poland.

A Variable explanations

(1) Contributing actors

- User Contribution - Specifies whether private persons or users are actively involved in the development
- Commercial Contribution - Specifies whether commercial companies are actively involved
- Research contribution - Specifies whether research institutions are actively involved

(1) Developers

The approximate number of active developers contributing to this project.

(2) Product complexity

The estimated degree of complexity for the developed product ranging from low complexity, e.g. a simple wooden chair, to high complexity, as for example an aircraft or a nuclear power plant.

(3) License

Indicates if the project is using an open license and specifies the type, including a mark whether they have a registered trademark.

(3) Degree of openness

- Software - Software and other non-physical, content parts are open source.
- Interfaces - Hardware specifications and interfaces are layed open.
- Schematics - Mechanical parts, descriptions, PCBs, etc. are freely available.

- Case design - If applicable the case design is available, e.g. as CAD for download.
- Entirely open - The project is revealing all available information.

(4) Development driver

Specifies the main drivers of the development, i.e. the group(s) of people pushing forward the project. Related company or association refers to a company closely related to the project, for example the investing company.

(4) Production

Specifies the group(s) of people responsible for producing the product. Related company or association refers to a company closely related to the project or a company with an exclusive production mandate. "Outsourced" is checked whenever an external party is paid for taking over the production.

(4) Activity

The activity level in the community or developer group from low (up to one interaction per month on average) to high (daily interaction).

(5) Development status

- 1 - Planning/Virtual development - Ideas and digital development evolving
- 2 - Prototyping started - First physical prototypes assembled, testing phase
- 3 - First working prototypes - Working prototypes available, release to community, further development needed
- 4 - Production stable - Fully functional product permanently available on market, further development possible
- 5 - Mature - Final development stage reached, no further development necessary

(5) Intended audience

- End user - Everybody from school kids to your grandmother.
- Advanced end user - Product will target end users, but usage may require specific knowledge.
- Developer - No intention to reach end users, high specific knowledge necessary.
- Other - All other groups, for example educational purposes

(5) Product innovativeness

- Radical innovation - a new technology that results in a new market infrastructure, e.g., an innovation which does not address a recognized demand but instead creates a demand previously unrecognized by the consumer
- Really new innovation - a really new product results in a market discontinuity or a technological discontinuity but will not incorporate both, e.g., new product lines, product line extensions with new technology, or new markets with existing technology
- Discontinous innovations - new technologies that don't lead to discontinuity in existing markets
- Incremental innovations - products that provide new features, benefits, or improvements to the existing technology in the existing market
- Imitative innovations - imitative products are frequently new to the firm, but not new to the market

(6) Home country

Indicates the main location, i.e. the home country of the project leader, the core team or the office location, as appropriate.

(6) Means of communication

- Face-to-face - Participants frequently interact in person
- Mailinglists - At least one mailinglist is used frequently
- Chat - The community has at least one active chat

- Board - A board/discussion forum is used
- Other - Other communication channels as wikis, blogs, etc. have been established