Concept Paper: Network of Stewards for Free and Open Source Software projects

Catharina Maracke* and Axel Metzger**

Version 1.0 October 2013

* Dr. iur., Fellow, Shuttleworth Foundation.

* * Dr. iur. (Munich and Paris), Dr. iur. habil. (Hamburg), LL.M. (Harvard), Professor of Law at Leibniz Universität Hannover, Germany.

Executive Summary

Despite the success of free and open source software, the friction and transaction costs involved in moving rights between developers and free and open source software projects are still high. Even if standardization efforts for public licenses and contributor agreements can serve as a starting point to create a framework for open collaborative production models, developers, projects and companies are often confronted with the hurdle of reviewing, understanding and negotiating different copyright and patent terms without having the capacity to deal with those issues.

The present concept paper investigates different scenarios and possibilities to set up a network of intermediaries that would mediate between developers and free and open source projects to make rights move efficiently and across jurisdictions. In addition to questions around copyright and patents, such intermediaries would also play a role in managing the risk inherent in the development and use of free and open source software. For example, the intermediaries could provide support for audits and a new Intellectual Property insurance yet to be developed and limited to covering legal fees without covering damage claims. By delegating legal and risk management to organizations that are competent and designed for that purpose, the envisaged network would build the much-needed infrastructure to enable a smooth flow of rights across jurisdictions.

A. Problem Statement

The success of free and open source software and especially projects like GNU/Linux or Apache provide an example of how innovation can be enabled by open collaborative processes outside the entrenched attitudes of exclusive control and "closed doors".[1] The concept of open source is based on the understanding that programmers rarely develop anything from scratch. When working on new ideas and writing code, they most likely use a prewritten component and build new elements as they see fit.[2] Since software does not sit alone on a plinth like a sculpture but interacts with other software, it is always necessary to take existing software into account and therefore more efficient if programmers can modify existing code.

Over time and in response to the constant growth of free and open source software, programmers have developed orderly and formal ways to build upon and improve already existing products and to efficiently collaborate on projects. This way of development through collaboration by many has resulted in a debate about whether the collaborative production model or the traditional proprietary production model is the better business model - and which model will promote further investment and innovation. Since the viability of the open source model has been demonstrated, the goal of this paper is not to compare and evaluate the contrasting production models but to provide additional thoughts on potential frameworks that can support and improve the collaborative production model.

Open source production takes place in a world already shaped by historical assumptions about creativity and incentives, one result of which is copyright law. And since software

1 See server market share survey by W3Techs - http://w3techs.com/technologies/overview/operating_system/all http://w3techs.com/technologies/details/os-unix/all/all; Apache and webserver market shares by Netcraft: http://news.netcraft.com/archives/2013/08/09/august-2013-web-server-survey.html.

2 See Heather Meeker, The Open Source Alternative, page 7 seq.

is covered by copyright law, which along with contract law provides a legal basis for its owner to establish exclusive rights, these rights can be exercised to allow or prohibit specific use of software. Consequently, any collaborative production model in the area of free and open source software requires legal tools to grant permission and modify code. As an immediate solution for such tools, a variety of standardized licenses have been developed to give participants the necessary rights to use and reproduce, modify and distribute software and related documentation. Depending on motivations and political views, these rights sometimes come with fewer or greater restrictions. [3]

While direct licensing of all software by individual programmers and developers is one possible way to ensure collaboration,[4] we can see more and more projects (and quite often legal entities such as companies and trusts) acting as gatekeepers and managers of software development, accepting contributions from various different programmers and (sub)licensing the results of the project to distributors and other users. In such cases, licenses can be divided into the so-called "inbound" and "outbound" licenses: Inbound licenses refer to the structure of rights between each individual developer and the organizing project; outbound licenses cover the rights to be licensed by the project to the public. Inbound licenses are also known as "contributor agreements", whereas outbound licenses are usually referred to as "public licenses".

To reduce the friction and transaction costs of reviewing, understanding and negotiating different copyright terms, standardization efforts for both licensing models have been established. Such standardized licensing terms help to reduce the overhead of scrutinizing each new license or license element, but, especially for the inbound licensing model, they are far from being widely accepted as industry standards.[5] In fact, most projects still tailor various different legal agreements relating to copyright and patents in order to structure and clarify rights in contributions. The result is a complex amalgam of different copyright and patent terms to be reviewed and negotiated, which represents a weakness and source of friction in the ecosystem of free and open software: Each time, developers and authors decide to collaborate and contribute to a specific project, all parties involved, be it contributors and their employees or the projects and companies, experience friction and transaction costs in reviewing and negotiating legal terms to define mutual relationships.

To address this problem, it is not only important to further improve standardized agreements, which can be used as templates to manage and move rights between different parties involved. It is also critical to think about the underlying infrastructure: What kind of framework can help to reduce friction and increase efficiency in the collaborative model of free and open source software development? One possible answer is the idea of a system whereby third-party organizations, trusted by developers and competent to represent them, would mediate between the different parties involved and make rights move smoothly and across jurisdiction. Reflection on the exact concept of such a system raises the following questions:

---

3 See overview of various different license models at http://www.ifross.org/en/license-center

4 This model has been described as "Inbound = Outbound", see Richard Fontana, "The trouble with Harmony" (2011), available at http://opensource.com/law/11/7/trouble-harmony-part-1.

5 For a critical stand on standard inbound licenses see Fontana, supra note 3; Bradley Kuhn, "Project Harmony Considered Harmful" (2011), available at http://ebb.org/bkuhn/blog/2011/07/07/harmony-harmful.html, Dave Neary, "Harmony Agreements reach 1.0" (2011), available at http://blogs.gnome.org/bolsh/2011/07/06/harmony-agreements-reach-1-0.

- Prospects of a stewardship model: What are the requirements and potential pitfalls for a third-party model, which can facilitate an efficient rights management system to mediate and move rights between developers and other authors, projects and companies, and potential end-users? And what would be the added value for developers of a stewardship model?
- Business model: What kind of business model can ensure long term sustainability for a stewardship model?
- Legal issues and licensing model: What are the key legal issues and how can they be addressed? What kind of licensing model is needed to support a third-party rights management system?

Addressing these questions requires investigating different settings and possibilities for the potential stewardship model. As a starting point for further discussion, it is envisaged that such a framework would consist of a network of "agencies" or "stewards", each chosen and trusted by the developers they serve, with formal relationships between the different agencies and also standardized agreements between the respective developers (and projects if they act as rightholders) and agencies to make rights move smoothly. This way, developers would be relieved of friction by shifting the burden of complex legal negotiations to organizations that are built for that purpose, and projects would be relieved of transaction costs because they only need to deal with a limited number of agencies able to act as conduits and move rights between jurisdictions.

B. The Stewardship Model

Key benefits of the proposed stewardship model include the possibility to replace the complexity of legal permissions with a fiduciary system of trust and simplification. Thus, success or failure of a network of stewardship organization depends on the participation of free and open source developers and especially on the trust that defines the scope of rights the developer is willing to grant a particular project or company. Without each developer's trust and willingness to participate, the envisaged third party model of stewardship organizations cannot be established. Experience has shown that the question of trust is closely related to the disposition of the parties involved and the definition of relationship between these different parties. Hence, one of the key questions to be investigated is the role of different parties involved in the intermediary model (I.) and their relationship to one another (II.). Another important aspect is the envisaged added value for developers: While it is obvious that clarification and simplification of the legal relationships is advantageous for professional users of free and open software, including large companies or governments, it is also clear that the intermediary model can not be further investigated without strong and well established advantages for developers (III.).

I. Parties involved

Developers typically contribute to specific open source projects and therefore transfer or license rights to the organization (legal entity) conducting the respective project. Projects without a stable organizational backbone may also chose to transfer or license rights to political or service oriented organizations supporting the development of free and open source software, e.g. one of the free software foundations, conservancies etc. In both cases, the transfer or license is based upon a special relationship of trust between developers and projects or developers and supporting organizations. And in both cases,

it is crucial to understand the importance of trust - developers will only hand over their intellectual property rights to organizations that follow the same values as their own. One developer may trust a non-profit community foundation but decline to collaborate with a for-profit company, whereas a different developer may have good and sometimes personal links with a particular for-profit company but less familiarity in working with community driven projects or foundations.

In summary, this very unique requirement of trust is often defined by the individual developers' goal and ideological mission and it can easily limit developers' willingness to contribute to important projects. In a scenario, where neither the project nor the supporting organization seems trustworthy, developers may decline to contribute. To address this potential weakness in the free and open source ecosystem, additional intermediaries should be identified or set up, chosen and trusted by the developers whom they serve. The proposed system of a network of stewards should therefore include different types of potential stewards: Open source projects with organizational mainstay, external organizations built for the purpose of rights management and support, and other organizations on the meta level, which are competent and willing to become potential partners for cooperation.

II. Relationships between parties

1. Developer - Steward

In order to represent and manage rights on behalf of developers, stewards will have to enter a formal relationship with each developer determining the exact purpose and scope of that relationship and the transfer or grant of rights, if any. The current landscape of existing contributor agreements offers a variety of different approaches: Some contributor agreements require a transfer of copyright whereas others follow a different approach and provide for an exclusive or even non-exclusive license.[6] Quite often, contributor agreements also differ with regard to the legal obligations of the recipient organization. Some major projects work with mere declarations of origin and do not ask for any transfer or license, but require that the contribution must be placed under the requisite license by a contributor before acceptance.[7]

These conceptual differences are one of the main challenges for the establishment of a network of stewards, because different recipients (which are supposed to act as different stewards) do not all have the same level of rights to bring into a network. Differing abilities to represent developers are a serious impediment for a third party rights management system, since network organizations cannot rely on a standard set of rights transferred to the steward but have to check the scope of rights for each bundle of rights. This problem could be solved if stewardship organizations use standardized contributor agreements, ideally drafted and developed in close collaboration with developers themselves.

An alternative design for the proposed intermediary would be to organize stewardship organizations based on *agency* relationship. If the intermediaries act as agents rather than

6 As examples see the FSF Copyright Assignment at http://ftp.xemacs.org/old-beta/FSF/assign.changes and the Apache Individual Contributor License Agreement available at http://www.apache.org/licenses/icla.txt. The variety of different license models is also reflected in the Harmony Agreements, which provide for different options when it comes to the scope of the transferred/licenses rights.

7 See Linux Developer Certificate Of Origin at http://ltsi.linuxfoundation.org/developers/signed-process.

licensees (or even transferees of copyright), much of the friction and complexity that can be found in already existing third party models,[8] will be avoided. Such a system of agents could ease the general scheme of rights management. Developers would not have to transfer copyrights or grant licenses. The agency model could also be based on mere declarations of ownership as they are used by many important projects.

2. Steward - Steward

In an ideal scenario, different stewards would be organized in a decentralized network, in which each steward represents – for a well defined number of issues – all developers committed to one of the other stewards, whether they are situated in the same jurisdiction or abroad. As a result, copyright clearance will be effected in a one-stop-shop: Potential free and open source software users or clients can work with only one specific stewardship organization to clarify copyright and ownership questions for a given program, even if other relevant parts of the program are managed by another steward and participating developers are represented by that other steward. As long as membership of the stewards in the proposed network requires adequate reciprocal cooperation between all stewardship organizations involved, free and open source software users will benefit from the proposed decentralized and international network structure.

However, up to date, most relevant potential stewardship organizations operate on a national, regional or even local level. Cross-border licensing efforts and related activities outside their home region are costly and burdensome and often not as effective as necessary. Even those organizations, which seek to cooperate on an international level through regional or national branches, have not yet established a stable framework for coordinated management of the transferred or licensed rights. There are multiple reasons for this lack of coordination, especially the lack of confidence in other organizations, the fear of loosing power and status if other organizations enter into one of the core activities of the stewardship organization, and uncertainty about the legal mandate to cooperate with other organizations in the management of rights of contributors. Consequently the stewards-as-agent model, which suggests that stewardship organizations are based on agency relationship, may evolve as a straightforward solution favored by many potential stewardship organizations and easier to realize compared to the license or transfer of rights model suggested by most contributor agreements.

III. Added value for developers

Success or failure of the proposed network depends on developers and their willingness to entrust the respective stewardship organization with the right to act as an agent and to cooperate with other network organizations. In other words, the network of stewards will have to offer clear advantages to developers. Depending on design and exact determination of stewardship organization, different services could offer added value for developer's to contribute, namely references to competent and specialized legal support, insurance, visibility, reputation and revenue:

1. Simplified communication and support

One obvious added value for developers could be the opportunity to stop struggling with legal documents and delegate all licensing issues to their stewardship organizations. Instead of having to deal with the legal complexity inherent in contributing to different projects, developers would only have to clarify once what freedoms and restrictions they

8 See the partially dysfunctional system of collecting societies in the music industry.

want their contributions to carry. Since the stewardship organizations would be chosen by the developers and therefore considered trustworthy, communication and clarification of rights with the stewards directly would be considered less burdensome compared to the current inevitability to negotiate with different projects and companies. In addition, the stewardship organizations could connect developers with competent partners outside the network to provide relevant legal support when needed. Whether and to what extend the stewardship organizations should be set up to provide legal support themselves, depends on the exact structure of the network.

2. Insurance

Insurance services could offer a major benefit for developers and projects to cooperate with stewardship organizations. The question of effective and affordable IP insurance has been discussed repeatedly over the past years but has not yet led to satisfying results. One obvious reason for this market failure is the ongoing debate around software patents. Whether or not software may be patented differs from jurisdiction to jurisdiction, which brings unpredictable risks and uncertainties to the calculation of respective insurance products. As long as the question of patentability, and especially the scope of potential software patents, is far from being answered, the risk of being sued for patent infringement is perceived as high. Consequently, any global IP insurance covering not only copyright but also patents would likely be exceptionally expensive to compensate for this unpredictable risk. However, there seems to be growing interest to identify affordable solutions. As a starting point for further discussion, a potential IP insurance could be tailored to cover narrowly defined risks. In the context of free and open source software, risk evaluation and risk management includes careful maintenance of technical specifications as well as legal and contractual obligations.

Against this backdrop, a potential IP insurance scheme could be set up to cover only legal fees without covering damage claims. Costly legal fees often seem to be the first barrier when considering defending an infringement claim. In fact, even if a defense has good chances to be successful, the initial costs to "buy" the necessary expertise to navigate through the procedural steps of a defense can become an insurmountable hurdle. This is especially true in the context of the US legal system, where a lawsuit over patent infringement usually involves different levels of complex arguments (e.g. that the respective product does not infringe the patent and that the patent itself is invalid), which can require disproportional effort and time. In addition, lawyers' fees are comparatively high, which is usually justified by the argument that very detailed and exceptional expertise is needed. If legal fees were covered by insurance, at least part of the pressure and fear of being targeted by law suits or even patent trolls can be taken away from developers and free and open source software projects.

Offering limited IP insurance products to cover legal fees also comes with the side effect that it can help developers and other inventors to defend themselves but does not encourage or provoke additional (patent) litigation. If damages are excluded, established "rich" companies and patent trolls may still face the risk of being left "empty-handed" and carefully weigh advantages of filing a lawsuit. Instead, chances for settlement and fairly negotiated license agreements will increase. In other words, wide availability of legal defense would raise the costs of frivolous or speculative patent and even copyright litigation, undercutting business models based on speculative litigation.

Providing insurance services for free and open source developers will have to be financed by the stewardship network. Therefore, the implementation of such a service will depend on the business model of the network.

3. Visibility and Reputation

Another way that stewardship network could add value for developers, is by assisting the individual developer to increase her visibility and foster her reputation in the community and in the user space.[9] One of the basic services of the steward network offered to free and open source software users could be clearing of ownership issues. Users should have the possibility to contact one of the stewardship organizations and to get information about the developer or company who has contributed a specific patch to a program. Such a service will require transparency between the different stewards about the identity of contributors. This transparency will allow the stewardship network to collect information about the contributions of specific contributors and produce – at their request – a private or public programmer's CV, which may be useful in different respects.

4. Revenues

Finally, the stewardship network could try to yield revenues and to distribute the revenues amongst the developers. Although it is not the primary interest of free and open source software developers to earn money with their contributions, it could still be a benefit for programmers if the network offers a substantial participation in the gains from the business use of their programs. Whether this is a realistic scenario, depends on the business model of the network. Collecting and distributing revenues would lead to a number of consequential questions, e.g. how to determine the precise share for each developer, how to review the accurate administration and distribution of shares, issues of tax law etc.[10]

C. Business Models for the Stewardship Network

Another critical question for the success of a network of stewards is the potential business model that would allow the different stewards to operate on a long term and sustainable basis. While philanthropic and corporate funders may be willing to finance the start-up of a network of stewards, once the network has been set up and demonstrated its utilities and benefits, different business models should emerge. Potential business models include:

I. Rights and risk consultancy services

Stewardship organizations could operate, on a basic service level, as advisors or consultants to offer free and open source software users specific services at pre-defined and fixed charges. This potential business model becomes promising in a scenario where stewards are not only representing individual developers but also developers, who are employed by companies or well-funded projects (and explicitly hired to contribute to free and open source software projects). In this case, it is assumed that companies and projects hiring free and open source software developers as well as potential end-users have a strong interest and budget for consulting services offered by the stewards, including primarily guidance on:

9 Signaling is often described as one of the extrinsic motivations of FOSS developers beside their intrinsic motivation, see e.g. Jürgen Bitzer et al., Intrinsic Motivation versus Signaling in Open. Source Software Development, http://pure.au.dk/portal/files/4921/WP_06-7.

10 Free and Open Source developers and their stewards could profit from the experience of collecting societies and design smarter and more equitable solutions.

- Right clearance (stewardship organizations could offer clarification of authorship and copyright ownership for all parts of free and open source programs where network organization hold contributor agreements or other certificates of origin)
- Questions on license compliance and related audits (stewardship organizations would be in a privileged position to track code and licenses, review legal and technical changes, and provide different auditing services)
- Questions on liability and insurance (up stream and down stream)

Under this model, stewardship organizations would not only take over the legal management for developers, but also support free and open source software users with risk evaluation and risk management. Compared to other license compliance services[11], the suggested clarification of rights service can offer a clear added value as users will know the precise contributors and their declarations of ownership or relevant contributor agreements. This basic model would not even require developers to give away or license their copyrights to stewardship organization, because a simple declaration of origin can suffice as legal basis.

With such an advanced level of transparency, the risk of third-party IP violation will be significantly reduced. Clarification of ownership would also facilitate the establishment of insurance coverage for developers and projects. As such, the network could be set up as a decentralized system that represents developers, their work, their values, and their rights, and which could serve as a repository of code, reputation and rights. In an ideal scenario, the revenues from the consultancy services could be used to pay for the insurance of developer and projects. Stewardship organizations could either offer such services directly or entrust lawyers or audit service providers with experience in the field, e.g. the Software Freedom Law Center, the OSADL Compliance Audit etc.

II. License enforcement

License enforcement could be a second level of services provided by the stewardship network. Experience with enforcement activities in Europe and the US has shown that individual developers are often overburdened with the discovery, documentation and legal assessment of license enforcement cases. Stewardship organizations could offer support and license enforcement on behalf of the developers they represent. However, details and possibilities of license enforcement depend on the legal relationship between developers and the respective stewardship organizations.

III. Relicensing (stewards acting as free and open source software banks)

As a third level of services, the stewardship organizations could have the right to relicense contributions from developers represented by the network organizations. Whether stewardship organizations should have the right to license free and open source software under different free and open source licenses (to solve compatibility issues) or under dual licensing models as proprietary software is one of the most controversial issues regarding contributor license agreements.[12] The right to grant proprietary licenses would allow stewardship organizations to raise license fees and to distribute revenues to developers, e.g. to keep 10% of the license fees for the administrative costs, 10% for social purposes and to transmit the remaining 80% to the developers which have transferred/licensed rights. It would be interesting to discuss with developers whether

11 See Blackduck or Palamida.

12 See e.g. Fontana, supra note 4.

the chance to earn revenues from the software would motivate them to enter into contributor agreements of this type or whether there would be a chilling effect on the developer's motivation to cooperate with stewardship organizations. In light of the controversial nature of such a service, it should be very clear from the outset that relicensing would only be an optional second level service offered to interested contributors.

IV. Funding options

In addition to the consultancy and bank models, and especially for the "startup" period of the network, donations and grants from a variety of different funders could be explored. Potential financial supporters include foundations and IT industries with an interest in legal certainty in free and open source software development. However, funding from one or more of the key players in this field comes with the risk of undermining the trustworthiness of the stewardship network. Thus, one of the most important issues when looking into options for the funding model (whether as an initial support or a long terms strategy) is to guarantee diversity in funding sources and to implement a clear and transparent allocation process.

D. Legal issues

Relevant legal issues will depend on the concrete structure of the network of stewards and the offered services.

I. Consultancy services, esp. clearing of copyright

The stewardship network could provide services based on public information, auditing services, where programmers and lawyers would scan the customers products and determine the applicable licenses and compliance issues and – as a new feature not available on the market yet – right clearance, where the steward would provide the customer with detailed information about the copyright status of the different modules and parts of the free and open source software product at hand. The first two elements do not raise specific legal questions, as long as only publicly available information or data of the respective customer is used. Clarification of the copyright status by contrast has legal implications. Although all major free and open source software licenses require the licensee to leave copyright notices untouched, there is no obligation under any renowned outbound license for developers to disclose their real name or even any name. Hence, clarification of ownership and the implied disclosure of the programmer's name to third parties depends – at least in many jurisdictions - on her authorization as required by applicable copyright and privacy or data protection law. In many jurisdictions, the moral rights of authors encompasses the right of an author to not disclose her name and to publish under a pseudonym or even anonymously, see e.g. § 12 German Copyright Act, section 77(8) UK Copyright, Designs and Patent Act, or the New York Artists' Authorship Rights Act. Jurisdictions without specific provisions may conceptualize the protection of the author's anonymity as a privacy issue. Whatever the legal basis is, the stewardship network will have to rely on developers' willingness to disclose their names and explicitly authorize the disclosure.

In practice, the required authorization could be communicated through different channels: Programmers could interact directly with *all* stewardship network organization and sign a certificate of ownership which allows all network organization to make

internal or even external use of relevant information.[13] This would allow the network organization to clarify the copyright status of the respective programs or modules and provide status information or even the programmer's name to the customer. A second possible solution could be for the developer to interact with the specific stewardship organization she trusts and allow that organization to hand over the information to other network organization or to communicate at least a status information within the network. Whatever the exact solution will be, most of the currently used certificates of origin or respective declarations in contributor agreements would have to be amended to provide a clear legal basis for the transfer of authorship information. Potential provisions to be implemented in future versions will have to clarify if authorship information can be disclosed to other stewardship organizations and under what conditions, e.g. no disclosure to third parties or disclosure with prior consent only, no use in public etc.

II. Relicensing as (sub-)licensee or agent

Relicensing of software under different free and open source licenses or under proprietary license terms could be offered under different legal conceptions: The network organization could either act with the authority to sublicense rights in its own name or as agent. The first option requires a transfer or grant of rights between the developer and her stewardship organization including the right to transfer or sublicense these rights. Thus, mere declarations of origin will not suffice as a legal basis. In addition, for most of the contributor agreements used today it is obvious that proprietary sublicensing by third parties is not covered by the scope of the license grant. Even relicensing software under free and open source software licenses may be bound to certain requirements, e.g. only to use licenses that oblige the licensee to provide machine-readable source codes[14] or to use only specific outbound licenses.[15] Therefore, relicensing by network organizations would require additional or different contributor agreements. It seems not very likely that organizations and developers will be inclined to accept such a change in their legal practice. Moreover, developers opposed to any contributor agreement will not cooperate with the stewardship organization for such services.

In the case of an agency model, many of the legal frictions connected with the different sublicensing options could be avoided. Agency as a legal concept is known to all modern jurisdictions, though with differences in the legal technicalities. Agency can be defined as the authority of a person ("the agent") to affect the legal relations of another person ("the principal") by or with respect to a contract with a third party.[16] In the typical scenario agency is disclosed to the contracting party. Where an agent acts within the scope of his authority and the third party knows or could have known that the agent was acting as an agent, the acts of the agent shall directly affect the legal relations between the principal and the third party and no legal relation is created between the agent and the

13 The Linux Developer Certificate Of Origin at http://ltsi.linuxfoundation.org/developers/signed-process provides in paragraph 4: "I understand and agree that this project and the contribution are public and that a record of the contribution (including all personal information I submit with it, including my sign-off) is maintained indefinitely and may be redistributed consistent with this project or the open source license(s) involved." Right clearance in the interest of users of the project's software could be understood as "redistribution consistent with the project". However, not all CAs/CLAs and DCOS contain such a provision.

14 See FSF Copyright Assignment at http://ftp.xemacs.org/old-beta/FSF/assign.changes.

15 This is the case under some Harmony Agreements, see e.g. Harmony Individual Contributor License Agreement, 2.3 ("Outbound License") Option One and Two, at http://harmonyagreements.org/docs/ha-cla-i-v1.pdf.

16 See Art. 2.2.1 et seq. UNIDROIT Principles of International Commercial Contracts 2010.

third party.[17] Hence, relicensing by stewardship organizations could also be effected by an agency relationship, under which the licensing steward acts on behalf of the developer and grants licenses to third parties. Such a construction can avoid any immediate transfer or license of rights when cooperating with the stewardship network. Consequently, the network will not be burdened with the bias that many developers have against copyright assignments. However, agency would also presuppose a relationship of trust between the developer and the agent, even if the agreement between principal and agent is drafted to define the mandate conferred on the agent in restrictive terms.

III. License enforcement

Legal actions against copyright violation may only be taken by a person who has the copyrights, thus if someone other than a contributor whishes to take legal action they can only do so on the basis of copyright transfer or exclusive license agreements.[18] Owners of mere non-exclusive licenses typically have no legal standing (ius standi in iudicio) in case of copyright violation. In some jurisdictions, e.g. Germany, the owner of a non-exclusive license may bring suit she has an explicit authorization from the rightholder, but such a solution is not available in all jurisdictions, e.g. not in the U.S. Hence, many contributor agreements (in so far as they are drafted as non-exclusive license agreements) will not put the stewardship organization in a position to enforce license violations.

Conclusion

The proposed network of stewardship organizations can build the infrastructure to reduce friction and transaction costs in free and open source development. Within a system of trusted intermediaries, developers will benefit from new services including legal support, insurance, visibility, reputation and revenue. Projects and end-users will benefit from increased transparency and legal certainty.

Practical implementation and success of the network depend on the ability to build trust between free and open source developers and potential stewardship organizations - and on funding possibilities. The ability to build trust will partly depend on the question which organizations are willing and competent to serve as stewards of the network and partly depend on political views. The question about funding and long-term sustainability will depend on the need and possible market for the services that can be offered by the stewardship organizations. As an immediate next step, it is suggested to discuss the network idea with selected partners and present the different settings in public conferences. Feedback, questions and comments from potential participants, be it individual developers, free and open source software projects or established companies, will help to further develop details concerning the concrete implementation.

17 See Art. 2.2.3 UNIDROIT Principles.

18 For the details see Tim Engelhardt, Drafting Options: Assignment, (Non)Exclusive License and Legal Consequences: The Unnecessary Gravity of the Soul, forthcoming SCRIPTed 2013.