

## Association for Information Systems AIS Electronic Library (AISeL)

---

MWAIS 2019 Proceedings

Midwest (MWAIS)

---

5-21-2019

# “Something to talk about” Exploring open source design spaces

Kevin Lombard

*University of Nebraska at Omaha*, [klumbard@unomaha.edu](mailto:klumbard@unomaha.edu)

Vinod Kumar Ahuja

*University of Nebraska at Omaha*, [vahuja@unomaha.edu](mailto:vahuja@unomaha.edu)

Matt Germonprez

*University of Nebraska at Omaha*, [germonprez@gmail.com](mailto:germonprez@gmail.com)

Sean Goggins

*University of Nebraska at Omaha*, [sgoggins@gmail.com](mailto:sgoggins@gmail.com)

Follow this and additional works at: <https://aisel.aisnet.org/mwais2019>

---

### Recommended Citation

Lombard, Kevin; Ahuja, Vinod Kumar; Germonprez, Matt; and Goggins, Sean, "“Something to talk about” Exploring open source design spaces" (2019). *MWAIS 2019 Proceedings*. 18.

<https://aisel.aisnet.org/mwais2019/18>

This material is brought to you by the Midwest (MWAIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in MWAIS 2019 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# “Something to talk about”

## Exploring open source design spaces

**Kevin Lumbard**

University of Nebraska at Omaha  
klumbard@unomaha.edu

**Vinod Ahuja**

University of Nebraska at Omaha  
vahuja@unomaha.edu

**Matt Germonprez**

University of Nebraska at Omaha  
germonprez@gmail.com

**Sean Goggins**

University of Missouri  
sgoggins@gmail.com

### ABSTRACT

Open source projects are dynamic environments where individuals and organizations collaborate to accomplish mutually beneficial design tasks. Open source designers utilize different design spaces, including discussion spaces (where conversation happens) and implementation spaces (where work happens) to complete design tasks. To investigate the relationship between discussion spaces, implementation spaces, and the completion of design tasks in open source projects, this research explores design as it occurs in organizational-communal open source projects under the umbrella of the Linux Foundation and focuses on design processes of three projects – Kubernetes, GRPC, and Zephyr. Design tasks - that include discussion spaces linked to implementation spaces - are successfully completed at a higher rate, than those with no link. This research demonstrates that the discussion space - implementation space relationship plays a key role in open source design processes.

### Keywords

Open Source, Design Spaces, Discussion Spaces

### INTRODUCTION

Design is a negotiation and takes place directly and indirectly among the designer and the community of users; design is shared and grounded in human experiences; design is a process and not necessarily a resultant artifact or easily identifiable framework (Buchanan, 2007). The role of designers in complex, interdisciplinary, collaborative, and distributed design projects is evolving, as they take on additional roles, such as mediators and facilitators (Howard & Melles, 2011). Designers advance varied and sometimes conflicting perspectives that are often further complicated due to geographic distance (Olson & Olson, 2000), sporadic availability (Howison & Crowston, 2014), many trajectories (Germonprez et al., 2017), complexity of language (Krippendorff, 1998), and cultural differences of designers (Gonzalez-Barahona, Robles, Andradas-Izquierdo, & Ghosh, 2008).

This research explores how design occurs in large collaborative geographically distributed projects - specifically open source projects. In open source, successful completion of a design task results in a change to the project (Howison & Crowston, 2014). Open source designers utilize different design spaces, including discussion spaces (where conversation happens) and implementation spaces (where work happens) to complete such design tasks (Sack et al., 2006). In implementation spaces, designers work with, store, and record changes. In discussion spaces, designers propose changes, communicate design processes, and identify flaws in the current design. Stemming from activity in discussion spaces, development and influence of open source design is evident in the resultant changes in implementation spaces.

This research investigates the link between discussion spaces and implementation spaces in three open source projects to understand how designers can influence successful completion of design tasks through activity in design spaces. In doing so, this research addresses the following research question:

*RQ1: What is the relationship between discussion and implementation spaces as evident in the design of open source software?*

### LITERATURE REVIEW

#### Open Source Design

Open source projects are dynamic environments where individuals and organizations collaborate to accomplish mutually beneficial design tasks through the creation of shared systems, architectures, and platforms (Chesbrough, 2006; Germonprez et al., 2017). Open source projects are the foundation of the modern digital infrastructure (Eghbal, 2016). Thus, organizations

are investing resources in open source projects as evidenced by research that suggests approximately half of open source designers are now acting as agents on behalf of organizations (Riehle, Riemer, Kolassa, & Schmidt, 2014). As organizations engaging in open source design find creative ways to balance internal and external design while working with open source projects (West & Gallagher, 2006), project dynamics are altered through new governance and coordinating mechanisms (Feller, Finnegan, Fitzgerald, & Hayes, 2008). In this context, open source designers coordinate their design in design spaces, including discussion and implementation spaces (Sack et al., 2006).

### Open Source Discussion Spaces

Open source discussion spaces include email lists, chats, video conferencing, web platforms (e.g., GitHub and Stack Overflow), private messaging modes for sensitive communication, and in-person conferences and hackathons. Discussion spaces provide support for formal and informal design communication in open source projects (Boden, Rosswog, Stevens, & Wulf, 2014). Open source discussion spaces have evolved from their initial form where they were primarily used to track and archive identified design flaws into a platform that allows designers to discuss new design concepts and coordinate large interdependent design tasks (Bertram, Volda, Greenberg, & Walker, 2010). Discussion spaces are beneficial - enabling trust to be built between designers, clarifying problems with design, and providing feedback, which may lead to design improvements (Haraty, McGrenere, & Bunt, 2017). Timely and visible feedback allow open source designers to reap the advantages of temporal and geographic distance (Dabbish, Stuart, Tsay, & Herbsleb, 2012; Lundell, Lings, Ågerfalk, & Fitzgerald, 2006) rather than the associated problems due to locality and spatiality in distributed engagements (Olson & Olson, 2000). As discussion spaces have evolved to become more central to the design process, they have begun to service many of the conversational, archival, and organizational needs of projects (Bertram et al., 2010). From their central role in open source design, discussion spaces are expected to shape open source design tasks.

### Open Source Implementation Spaces

Open source projects complete design tasks within design spaces through the implementation of proposed design changes into an expanding codebase (Howison & Crowston, 2014). Proposed changes may result from implicit coordination in the implementation space or explicit coordination in discussion spaces (Bolici, Howison, & Crowston, 2016). To manage interdependencies in design tasks, implementation spaces often incorporate or link to discussion spaces to share knowledge (Lindberg, Berente, Gaskin, & Lyytinen, 2016). Implementation spaces further utilize discussion spaces for identification of design flaws by capturing bug reports from users and designers (Fogel, 2005). This shows that implementation spaces are often dependent upon discussion spaces for guidance on open source design tasks.

### Relationship Between Discussion and Implementation Spaces

Well defined design processes create constraints that promote the layering of design tasks into small manageable units leading to code that is more reusable, higher quality, and easier to understand (Howison & Crowston, 2014). As projects self-organize, implicit design processes--trust, mental maps, and shared frames--and explicit design processes--written rules, plans, and feedback --are defined (Bird, Pattison, D'Souza, Filkov, & Devanbu, 2008; Bolici et al., 2016) Open source projects often create documents for how to contribute, joining scripts, and project roadmaps to reduce the barriers of entry for new designers and increase success outcomes (Steinmacher, Conte, Gerosa, & Redmiles, 2015). Further, many open source projects define explicit connections between discussion spaces and implementation spaces. For example, the published design processes of many open source projects recommend starting design tasks within discussion spaces prior to moving the task to the implementation space to increase the likelihood of success (<https://kubernetes.io/docs/contribute/>). This indicates that linking discussion spaces to implementation spaces can play an important role in completing design tasks.

## METHOD

To investigate *the relationship between discussion spaces, implementation spaces, and the completion of design tasks in open source projects*, this research explores design in organizational-communal open source projects under the umbrella of the Linux Foundation and focuses on design processes of three projects – Kubernetes<sup>1</sup>, GRPC<sup>2</sup>, and Zephyr<sup>3</sup>. Kubernetes is an open source system for managing of containerized applications. GRPC is an open source remote procedure call framework used to connect devices across different platforms. Zephyr is an open source operating system for “Internet of Things” devices. These projects were selected because they are active Linux Foundation projects and share similar levels of organizational interest. Additionally, project work for all three occurs on the GitHub<sup>4</sup> platform making trace data collection repeatable for each project.

<sup>1</sup> <https://kubernetes.io/>

<sup>2</sup> <https://grpc.io/>

<sup>3</sup> <https://www.zephyrproject.org/>

<sup>4</sup> <https://github.com/>

GitHub is an established source of data for research (Cosentino, Luis, & Cabot, 2016). The GitHub discussion space includes social features and a transparent workflow that make information about users and their work accessible. Designers can make issue comments and code comments in online forums built around the workflow. The GitHub implementation space provides a mechanism for designers to share and merge code bases via pull requests. Pull requests allow designers to create a thread for proposing code changes and discussing it with others. Within a pull request, proposed design changes can be viewed, commented, edited, voted on, and finally accepted or rejected.

Throughout a seven-year research project, the research team has become active participants in numerous open source projects. This gives us a unique position to understand and report on open source project research broadly. From this broad understanding, time was spent specifically exploring the three aforementioned projects on GitHub to understand specific components of their design process. The three projects selected show different levels of activity providing some variation in the contexts within which the research question is addressed. (see Table 1).

**Table 1: Project Activity**

Project	Pull Requests	Pull Request Files changed	Code & Comment Contributors	Comments
Kubernetes	42,485	727,218	13,423	464,834
GRPC	10,749	223,699	3,613	59,109
Zephyr	7,078	87,858	932	22,474

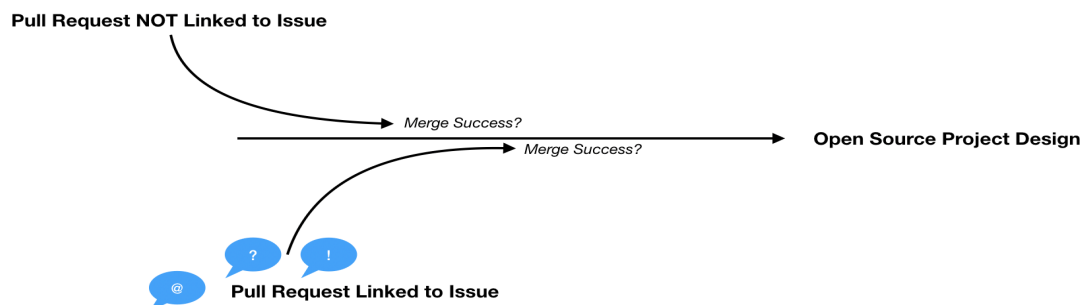
### Research Approach

Digital ethnography was used to extract and examine trace data and project documentation from Kubernetes, GRPC, and Zephyr. Digital ethnography is a participant-observation research approach adjusted to an online world (Kozinets, 2015), in this case the lens aimed explicitly at both the online work space and the online talk space (Goggins, Mascaro, & Valetto, 2013). Online data sources are used to arrive at understandings and representations of online social experiences, through observation, exploration of archival data, and quantitative analysis. Human communications were interpreted – not merely words but also digital artifacts - “under realistic contexts, *in situ* in native conditions of interaction” (Kozinets, 2015, p. 5).

### Data Extraction and Analysis

Data was extracted from GitHub using its GraphQL interface on January 15th, 2019. RStudio was used to clean the data by removing data that did not fit the described schema, outlier data that were the result of automatic processes malfunctioning (i.e., bots), and pull requests that did not include changes to the repository. To operationalize the link between discussion and implementation spaces, R and regular expressions were used to extract whether a pull request was explicitly linked to an issue from the body text of the pull request.

To understand the relationship between discussion spaces and implementation spaces, this research explored how posted issues (discussion spaces) are related to pull requests (implementation spaces) that have been merged (success) or closed (non-success). Within pull requests, data was extracted about, whether the pull request was explicitly linked to an issue and merge success (see Figure 1).



**Figure 1. Comparing merge success rates of pull requests that originate in issue discussions to pull requests that do not.**

Based on open source literature (McDonald, Blincoe, Petakovic, & Goggins, 2014), it is expected that discussion in issues will influence pull request merge success. To determine the correlation between issue discussion and merge success, descriptive statistics were used to explore the relationship between the variables. To determine if the relationship is statistically significant and to calculate odds ratios, a logistic regression was performed. Logistic regression was chosen because the two variables in the analysis, pull request merge and pull request issue link are both binary.

## RESULTS

### Discussion Spaces are viewed as essential parts of the design process

All of the projects explored referenced discussion spaces in their documented design processes and encourage planning and discussion first in the discussion space and then continuing into the implementation space when working on design tasks. The following are examples from each project documentation:

*“Even for small changes, it is often a good idea to gather feedback on an issue you filed, or even simply ask in the appropriate SIG’s Slack channel to invite discussion and feedback from code owners.” -- Kubernetes<sup>5</sup>*

*“For speculative changes, consider opening an issue and discussing it first. If you are suggesting a behavioral or API change, consider starting with a gRFC proposal. Provide a good PR description as a record of what change is being made and why it was made. Link to a GitHub issue if it exists.” -- GRPC<sup>6</sup>*

*“Before starting on a patch, first check in our issues Zephyr Project Issues system to see what’s been reported on the issue you’d like to address. Have a conversation on the Zephyr developer mailing list (or the #zephyrproject IRC channel on freenode.net) to see what others think of your issue (and proposed solution). Send a message to the Zephyr devel mailing list to introduce and discuss your idea with the development community.” -- Zephyr<sup>7</sup>*

### Linking discussion spaces to implementation spaces results in design tasks being completed at a higher rate

Design tasks - that include discussion spaces **linked** to implementation spaces - are **successfully completed** at a higher rate, than those with **no link**. For example, in Kubernetes, 51.50% of successfully merged pull requests are linked to an issue whereas 43.15% of successfully merged pull requests are not linked to an issue (See table 2).

**Table 2: Design Tasks Link to Discussion Spaces**

Project	Linked to Discussion Space (Issue)	Successfully Completed (Merged Pull Request)
Kubernetes	Yes	51.50%
	No	43.15%
GRPC	Yes	33.86%
	No	28.21%
Zephyr	Yes	19.98%
	No	18.69%

The calculated odds ratios from the logistic regression show that design tasks where discussion spaces are linked to implementation spaces are more likely to be successfully completed for all three projects. Pull requests linked to issues are 1.39 times more likely to be merged on Kubernetes, 1.29 more likely on GRPC and 1.08 more likely on Zephyr. For Kubernetes and GRPC the results are statistically significant. While this positive relationship is observable in Zephyr, the results are not significant, suggesting that there may be other factors that influence successful task completion, or, possibly a different discussion space such as an email list or routine meetings where pull requests are discussed (See table 3).

**Table 3: Logistic Regression**

Project	Coefficients	Estimate	Std. Error	Z Value	Odds Ratio	Pr (> z )
Kubernetes	(Intercept)	1.24433	0.01644	75.69	3.4706	<2e-16 ***
	Link to an Issue Success	0.3357	0.02457	13.66	1.3989	<2e-16 ***
	Null deviance: 42201 on 42484 degrees of freedom					
GRPC	(Intercept)	1.56431	0.01362	50.134	4.7794	<2e-16 ***
	Link to an Issue Success	0.25611	0.05771	4.438	1.2919	9.09e-06 ***
	Null deviance: 9490.2 on 10714 degrees of freedom					
Zephyr	(Intercept)	1.95298	0.04024	48.528	7.04965	<2e-16 ***
	Link to an Issue Success	0.0831	0.09273	0.896	1.08665	0.37
	Null deviance: 57237 on 60277 degrees of freedom					
<b>Significance codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1</b>						

## DISCUSSION

This research shows that the discussion space - implementation space relationship plays a key role in open source design processes for three specific projects on GitHub, supporting and elaborating more specifically on prior research (Bertram et al., 2010). The role of discussion spaces supporting implementation spaces to share knowledge, manage interdependencies, and

<sup>5</sup> <https://kubernetes.io/docs/contribute/>

<sup>6</sup> <https://github.com/grpc/grpc/blob/master/CONTRIBUTING.md>

<sup>7</sup> <https://docs.zephyrproject.org/latest/contribute/index.html#pull-requests-and-issues>

implement design changes is surfaced in prior work (Bertram et al., 2010; Howison & Crowston, 2014; Lindberg et al., 2016). This research contribution shows the specific influence of discussion spaces on design task completion success.

In organizational-communal projects (Germonprez et al., 2017), discussion spaces are used for planning and coordination of design tasks. Well defined project design processes let contributors know that changes to the code base should be proposed and discussed in discussion spaces prior to making a change. Activity within discussion spaces often leads to work within implementation spaces and when it does, it has a higher completion success rate than work that does not originate in discussion spaces. Analysis of Kubernetes, GRPC, and Zephyr indicate that it is more likely that the work of a designer will result in successful completion of a design task if the task stems from a combination of work in *both* discussion and implementation spaces.

Further, the occurrence rate and the completion success rates of linked discussion and implementation spaces may vary by project activity. For example, in projects with large amounts of activity, like Kubernetes, more pull requests are linked to issues and there is a strong positive relationship between issue linking and pull request merge success, whereas the two projects showing less activity also have fewer occurrences and a weaker positive relationship. In this context, project activity appears to be a moderating variable that affects the relationship between design spaces. Future research could explore the impact of project activity and project size on design spaces.

### LIMITATIONS

This research focused on strategically important open source projects at the Linux Foundation. This research does not explore open source projects outside of the Linux Foundation or volunteer-driven projects, nor is it meant to negate prior research on volunteer-driven communities. Further, this research it is not intended to generalize design for all open source projects but merely explore design through the study of organizational-communal open source projects. Additionally, bots were identified in Kubernetes (4 bots), GRPC (2 bots) and Zephyr (1 bot) but they were not removed from the research. Bots were included as participants and may add comments to pull requests and participate in discussion spaces. This research sees bots as agents acting on behalf of project leadership. In this, they communicate design guidelines, enforce rules, and review pull requests for compatibility with interdependent files. As such, they are participants in design. Removal of the bots from this research is unlikely to change the results drastically but their existence should be noted.

### CONCLUSION

This research explored design in three open source software projects. This work contributes to design literature and open source literature by exploring *the relationship between discussion spaces and implementation spaces as evident in the design of open source software*. Discussion spaces are an increasingly important part of design in organizational-communal open source projects. Understanding how discussion spaces and implementation spaces affect task completion can help open source designers create well defined design processes that promote success outcomes and improve designer engagement for all.

### ACKNOWLEDGEMENTS

This project received funding from the Alfred P. Sloan Foundation Digital Technology grant on Open Source Health and Sustainability, Num 8434 (<https://sloan.org/grant-detail/8434>).

### REFERENCES

1. Bertram, D., Volda, A., Greenberg, S., & Walker, R. (2010). Communication, collaboration, and bugs: the social nature of issue tracking in small, collocated teams. *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, 291–300. ACM.
2. Bird, C., Pattison, D., D’Souza, R., Filkov, V., & Devanbu, P. (2008). Latent social structure in open source projects. *Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, 24–35. <https://doi.org/10.1145/1453101.1453107>
3. Boden, A., Rosswog, F., Stevens, G., & Wulf, V. (2014). *Articulation spaces: bridging the gap between formal and informal coordination*. 1120–1130. <https://doi.org/10.1145/2531602.2531621>
4. Bolici, F., Howison, J., & Crowston, K. (2016). Stigmergic coordination in FLOSS development teams: Integrating explicit and implicit mechanisms. *Cognitive Systems Research*, 38, 14–22.
5. Buchanan, R. (2007). Strategies of design research: Productive science and rhetorical inquiry. *Design Research Now*, 55–66.
6. Chesbrough, H. W. (2006). *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business Press.

7. Cosentino, V., Luis, J., & Cabot, J. (2016). Findings from GitHub: Methods, datasets and limitations. *Proceedings of the 13th International Conference on Mining Software Repositories*, 137–141. <https://doi.org/10.1145/2901739.2901776>
8. Dabbish, L., Stuart, C., Tsay, J., & Herbsleb, J. (2012). Social coding in GitHub: Transparency and collaboration in an open software repository. *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, 1277–1286. <https://doi.org/10.1145/2145204.2145396>
9. Eghbal, N. (2016). *Roads and bridges: The unseen labor behind our digital infrastructure*. Ford Foundation.
10. Feller, J., Finnegan, P., Fitzgerald, B., & Hayes, J. (2008). From peer production to productization: A study of socially enabled business exchanges in open source service networks. *Information Systems Research*, 19(4), 475–493.
11. Fogel, K. (2005). *Producing open source software: How to run a successful free software project*. O’Reilly Media, Inc.
12. Germonprez, M., Kendall, J. E., Kendall, K. E., Mathiassen, L., Young, B., & Warner, B. (2017). A theory of responsive design: A field study of corporate engagement with open source communities. *Information Systems Research*, 28(1), 64–83. <https://doi.org/10.1287/isre.2016.0662>
13. Goggins, S. P., Mascaro, C., & Valetto, G. (2013). Group informatics: A methodological approach and ontology for sociotechnical group research. *Journal of the Association for Information Science and Technology*, 64(3), 516–539.
14. Gonzalez-Barahona, J. M., Robles, G., Andradas-Izquierdo, R., & Ghosh, R. A. (2008). Geographic origin of libre software developers. *Information Economics and Policy*, 20(4), 356–363.
15. Haraty, M., McGrenere, J., & Bunt, A. (2017). Online Customization Sharing Ecosystems: Components, Roles, and Motivations. *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, 2359–2371. <https://doi.org/10.1145/2998181.2998289>
16. Howard, Z., & Melles, G. (2011). Beyond designing: Roles of the designer in complex design projects. *Proceedings of the 23rd Australian Computer-Human Interaction Conference*, 152–155. <https://doi.org/10.1145/2071536.2071560>
17. Howison, J., & Crowston, K. (2014). Collaboration through open superposition. *Mis Quarterly*, 38(1), 29–50.
18. Kozinets, R. V. (2015). *Netnography: redefined*. SAGE Publications.
19. Krippendorff, K. (1998). *Design discourse: A way to redesign design*. Keynote presented at the Society for Science of Design Studies, Tokyo, Japan.
20. Lindberg, A., Berente, N., Gaskin, J., & Lyytinen, K. (2016). Coordinating interdependencies in online communities: A study of an open source software project. *Information Systems Research*, 27(4), 751–772.
21. Lundell, B., Lings, B., Ågerfalk, P. J., & Fitzgerald, B. (2006). The distributed open source software development model: Observations on communication, coordination and control. *ResearchGate*, 683–694.
22. McDonald, N., Blincoe, K., Petakovic, E., & Goggins, S. (2014). Modeling Distributed Collaboration on GitHub. *Advances in Complex Systems*, 17(7–8), 1–24.
23. Olson, G. M., & Olson, J. S. (2000). Distance matters. *Human-Computer Interaction*, 15(2), 139–178.
24. Riehle, D., Riemer, P., Kolassa, C., & Schmidt, M. (2014). Paid vs. volunteer work in open source. *System Sciences (HICSS), 2014 47th Hawaii International Conference On*, 3286–3295. <https://doi.org/10.1109/HICSS.2014.407>
25. Sack, W., Détienne, F., Ducheneaut, N., Burkhardt, J.-M., Mahendran, D., & Barcellini, F. (2006). A methodological framework for socio-cognitive analyses of collaborative design of open source software. *Computer Supported Cooperative Work (CSCW)*, 15(2–3), 229–250. <https://doi.org/10.1007/s10606-006-9020-5>
26. Steinmacher, I., Conte, T., Gerosa, M. A., & Redmiles, D. (2015). Social barriers faced by newcomers placing their first contribution in open source software projects. *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, 1379–1392. <https://doi.org/10.1145/2675133.2675215>
27. West, J., & Gallagher, S. (2006). Challenges of open innovation: the paradox of firm investment in open-source software. *R&D Management*, 36(3), 319–331. <https://doi.org/10.1111/j.1467-9310.2006.00436.x>