REDBACK BIM

Developing 'De-Localised' Open-Source Architecture-Centric Tools

EMILY LEUNG[1], ROB ASHER[2], ANDREW BUTLER[3],
BEN DOHERTY[4], ALESSANDRA FABBRI[5], NICOLE GARDNER[6]
and M. HANK HAEUSLER[7]
[1,2,3] *Cox Architecture Sydney*
[1,2,3] *{emily.leung|Rob.Asher|Andrew.Butler}@cox.com.au*
[4,5,6,7] *UNSW / Computational Design*
[4] *Ben_Doherty@bvn.com.au* [5,6,7] *{a.fabbri|n.gardner|m.haeusler}@unsw.edu.au*

Abstract. Emerging technologies that use data have contributed to the success of communication all over the world. Social media and gaming industries have already taken advantage of the web to provide synchronous communication and updated information. Conversely, existing methods of communication within the AEC industry require multiple platforms, such as emails and file sharing services in conjunction with 3D Modelling software, to inform changes made by stakeholders, resulting in file duplication and limited accessibility to the latest version, while augmenting existing practice's inefficiency. As communication is critical to the success of a project and should be enhanced, Redback BIM promises to establish a workflow for a dynamic platform, while achieving similar results to that of a 3D modelling program hosted on the web. Using existing open-source web development software, multiple users will be able to collaboratively organise and synchronise changes made to the design scheme in real-time. Features such as this would enable more fluid communication between multiple stakeholders within the life of a project.

Keywords. De-localised Workspaces; Web-based Software Platforms; Data; Open-source; Collaboration.

1. Research Motivation

Architectural practice should and can take greater advantage of data driven emerging technologies, whether through data analytics, artificial intelligence, robotics, digital supply chains and advanced construction (Raisbeck, 2016). Critical to the architecture, engineering and construction (AEC) industry is the use of 3D modelling operating systems, such as Rhinoceros, Revit or AutoCAD, which allow for accurate design and documentation of spaces and buildings. Although the augmentation of the foregoing programs' has led to enhancements in the form of visual scripting platforms, such as Grasshopper and

T. Fukuda, W. Huang, P. Janssen, K. Crolla, S. Alhadidi (eds.), *Learning, Adapting and Prototyping, Proceedings of the 23rd International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2018*, Volume 2, 21-30.

Dynamo, the communication of information between multiple stakeholders hasn't experience a similar evolution and still remains a key challenge. Conversely, since the release of Web 2.0 and the most recent advent of cloud computing, there has been a significant growth in synchronous collaborative web-based software platforms, such as Google's G Suite (Google Docs, Google Slides, etc), Microsoft's Office 365, Trello and Slack. Allowing multiple users to access and edit live documents without requiring software licensing or hosting on localised private servers or devices, many of these existing platforms, here referred to as 'de-localised' systems, have already been implemented within the AEC industry for text editing and visualisation purposes that typically require the input of various shareholders. Nevertheless, formal evaluations of design aspects still occur through ordinary means of correspondence, such as email and telephone. As effective communication is reflective of its expression medium, the current research argues that existing consultation's practices within the AEC industry can be additionally enhanced by creating a 'de-localised' 3D modelling interface. In an era of globalised relationships, designers, contractors and consultants collaborate more and more across vast geographic distances and time zones. Consequently, the reliance on outdated localised design platforms results in practice's inefficiency and underdevelopment. Storing files on private servers leads to redundancy of inaccuracies and obsolescence of information. Existing web-based software programs seeking to enhance collaboration while addressing issues such as version control, file storage, downloading, model viewing and commenting, include cloud-based applications, such as AutoDesk 360 BIM, Dropbox and Aconex. However, they are not only costly alternatives requiring additional licensing but also limited in functionality and not implemented to enable real-time updates. Therefore, the combination of synchronous collaborative spaces and 3D modelling programs could be achieved through the repositioning of the design process on a web-based platform. Current multiplayer online games provide key examples of the potential of concomitant coaction within a common virtual environment: for instance, the popular game Minecraft allows users to work or converse with anyone in the world at the same time within the same "seed" (equivalent to an ID). In particular, the significance of synchronous participation is represented by the users' ability to elaborate, evolve or modify each other's work in real-time. Therefore, synchronous collaborative environments show promise in delivering improved design efficiency and quality from unified workspaces (French et al, 2016). The contemporary architectural practice is increasingly characterised as a collaborative environment that challenges the traditional idea of curatorship from a single keeper and organiser of information, to one where authorship is the result of teamwork-oriented practice (Aguirre Leon and Molina 2009). As a result, the comparison between available tools and current practice's methods has proved the web to be the common denominator within a collaborative scenario. The management of project information using de-localised platforms combined with 3D modelling capabilities stands to offer significant benefits in terms of workflow and project collaboration.

2. Research Aims and Objectives

The overarching aim of this project is to generate a workflow for a dynamic platform capable to achieve similar results to that of a 3D modelling program hosted on the web. The key focus here will be on the dynamic update of geometry data to a synchronous collaborative web-based platform which will be made accessible on any mobile device. This new digital interface has the potential to create an impact on existing communication related to a consolidated 3D architectural model in design and architecture, bridging the gap between multiple stakeholders.

3. Research Question

In what ways can participating in developing open-source and dynamic web applications be implemented to enable synchronous relationships between data and 3D geometry beneficial to the optimisation of communication practices in the AEC industry?

4. Methodology

4.1. ACTION RESEARCH METHODOLOGY

Recent achievements in design, delivery and research of Information and Communication Technology (ICT) systems and new media applications have established action research as the methodology to thoroughly understanding a problem while galvanising a change in the process (Dick et al, 1997). Action research defines how stages of planning, acting, observing and reflecting are cycled iteratively throughout the project life. This reflective process offers empirical justification and substantiation of theoretical ideas through deriving relevant technological artefacts. The overarching methodology driving this research aligns with Stephen Kemmis's Simple Action Research Model (O'Brien, 1998) as it seeks a change through cycled design iterations.

4.2. PROJECT METHOD IN PRACTICE

The following method has been applied in practice:

- From Flux's online database, create a web interface
- On the dynamic application, a project can be selected with a click. Project's keys are displayed and accessible for selection, to obtain embedded data
- A key housing the geometry shows in the viewport. The meshes in the viewport can be selected. When a specific mesh is selected, a table of attributes is displayed
- Changing the attributes in the table results in real-time geometrical alterations of the project's configuration and its embedded data

As a proof of concept, the research adopted Google Doc's systematic relationship by implementing the open-source web platform Flux.io in order to build the virtual application. This is based on the hypothesis that:

- It resolves existing communication limitations of 3D models through

web-based technology
- CAD and 3D modelling data can be analysed, edited, and documented online
- It produces similar functionalities to that of a localised 3D modelling software.

5. Background Research

5.1. WEB-BASED APPLICATION PRECEDENTS

An extensive comparative analysis was conducted on dynamic platforms currently employed in the AEC Industry (See Table 1). 'Documentation' refers to operative systems' ability to store and produce 2D architectural drawings for practice. '3D / Interactive' represents platforms' flexibility in allowing users to build and interact with a 3D model. 'Three.js', an open-source JavaScript library framework, refers to systems' versatility for users customisation of web three-dimensional scenes. Finally, platforms that allow collaborative live documents to be hosted on the web exist under the heading 'Free Collaboration'.

Table 1. Web-based application precedents.

Precedent	Documentation	3D / Interactive	Three.js	Free Collaboration
Autodesk 360 BIM	✓	✓	✓	
Aconex	✓	✓		
Clara.io		✓	✓	
Dropbox	✓			
Flux.io		✓	✓	✓
OnShape	✓	✓		
Playpus		✓	✓	✓
ShapeDiver		✓	✓	

5.2. LITERATURE REVIEW

Web-based collaborative applications are supported by several scholars in the fields of architecture and computer science (Raymond 1999, Graham 2009, León and Molina 2009, Counsell 2012 and Marble 2012). Since the advent of digital modelling for the construction industry, the majority of the models have been stored and modified on remote desktop computer, and occasionally shared on local servers, failing to provide real-time integration of information from multiple stakeholders (León and Molina, 2009). Conversely, the present research project proposes a "federated single point of truth" concept, to emphasise the need for hosting the model on a common virtual environment (Counsell, 2012, p.510). While succeeding in generating an efficient data storage process, BIM hasn't achieved similar results in web-based 3D models visualisation. BIM underlying information has been defined as data like any other, where it can be handled in large database management systems (DBMS), which justifies the accessible nature of data to be one that can be transformed for the user's needs (Counsell, 2012). Flux Flow, for instance, is a visual scripting web application that allows users to extract and transform data from properties within a building information model, to be streamlined into a digital spreadsheet readily available for contractors. While the majority of studies provide valuable information in relation to the concepts drawn from accessing data on the internet, caution needs to be taken to compare the tools that existing papers have created, such as Vroom (León and Molina, 2009). Therefore, focusing on visualisation and interaction with 3D models, these studies fail in managing the bi-directional data flow between multiple users in real-time. Considering the foregoing arguments, it can be concluded that existing investigations do not primarily see the web as a space for documentation and collaboration on architectural models. On the other hand, the current research focuses on such applications, aiming to detach digital architectural modelling systems, such as BIM and CAD, from localised servers. It is therefore advantageous to break down the process into universally applicable processes and logic, to provide a methodology which has the potential to be reapplied using alternative open-source software and programming functions. Current practices within architecture limit users' flexibility with available digital tools (Marble, 2012). In particular, being designed by computer scientists, architectural operative systems are not easily customisable. Consequently the "sheer fact of using architectural software means already to operate like an engineer" (Marble, 2012, p.18). Yet, if architects possess the ability to understand and even author the tools that influence their process, potential efficiencies could emerge. Scott Marble derives three themes that define the integration of digital technologies in practice: Designing Design, Designing Assembly and Designing Industry. Understanding and questioning the need for synchronous collaborative workspaces fits into Marble's category: Designing Industry. More specifically, computer science's scholars have idealised the need for open-source methodologies in creating successful technologies when no one owns the software. 'Hackers and Painters' research (Graham, 2003) draws upon the significant connection that creatives such as architects have very similar processes to hackers, such as making and managing. Therefore, pulling apart (hacking) into software can significantly impact unprecedented discoveries (Ibid). By removing *The Cathedral Model* (commercialised) to the *OpenSource Development Model* - The Bazaar (Raymond, 1999) - a full collaborative effect is attained through being susceptible of public input. In conclusion, free-from-owner technologies would promote the success of collaboration in today's digital age.

6. Case Study (REDBACK BIM)

When developing a 3D modelling web application, the main aim is the optimisation of sending and receiving data between the frontend (user interface) and backend (database). The app was named Redback BIM because of its correlation of using the web as a space to accommodate BIM model data. In relation to the current research, Redback BIM, the user must access and update data from a web-based interface, which is then synchronised in Flux.io. After synchronising the data, Flux.io can interpret and update the data, which is then sent back to the interface for the user to see. This paper looks specifically into synchronising edited data back to the web server. There are several steps in this process, and the focus draws upon constructing a scenario for an editable JSON string in association with its geometry.

6.1. COMPARATIVE STUDY BETWEEN DEVELOPING APPLICATIONS USING A-FRAME AND FLUX SDK DOCUMENTATION

Building an A-Frame Scene A-Frame (open-source WebVR JavaScript library) was employed to elaborate an existing workflow towards setting up a 3D architectural model on the web from a localised software. Being the first prototype in development, a default sample Revit model was used as a test subject to be exported into a 3D DWG file. Additionally, as A-Frame requires specific file types to be imported, the DWG files were transformed into OBJ files through Rhinoceros. The steps are as follows: (1) Export the Revit model as a DWG file; (2) Modify the settings to export the file as ACIS solids; (3) Choose to export it as an AutoCAD 2007 DWG file; (4) Open the exported file in Rhinoceros to be transformed into an .obj files, or any file compatible with A-Frame; and (5) Store exported files into a folder to reference into A-Frame.A-Frame requires model files (.obj) to be manually referenced into the HTML code with an assigned ID, which becomes a unique property of each <a-asset-item> tag. After completing this step, additional functionality was added using JavaScript. More specifically, a JavaScript variable that holds an array of JSON objects and each object's data was referenced and aligned to each object's tag by associating the data with the object's ID. When an object within the scene is selected, the associated ID of that object is called, displaying its data within the HTML element situated on the left sidebar.

Building a web application using Flux SDK documentation To achieve accessibility, synchronous data and collaboration, Redback BIM would require a platform very similar to Flux.io. Specifically, a web platform that allows users to send and receive various types of data (integers, string and geometry) between compatible software via the web. Flux.io has developed plugins which allows the value of the data to be stored within your account. Just like JSON, you are assigned a key and a value {"key": "value"}: expressed in JSON format, the geometry becomes the value, while being organised by the user for others to access. As a result, building a web application using Flux.io can open opportunities for creating live 3D documents through a user interface by editing the geometry's data (JSON string).

6.2. ENABLE COMMUNICATION BETWEEN WEB APPLICATION AND FLUX DATABASE

Defining the Workflow Architects and Stakeholders in possess of an existing site or building digital model could send it to the web for others to access it through Flux.io. Using Flux.io as a database for the application, the steps are as follows: (1) Existing 3D models are sent to Flux.io through its compatible plugins; (2) Flux stores the geometry and its data in the user's projects and keys; and (3) Redback BIM accesses the Flux database through JavaScript which parses the JSON into the Flux Viewport and React components. The following are the relational components to the workflow (See Fig. 1): (A) Database for Flux; (B) JSON data (integers, strings, Flux reads as geometry); (C) Three.js (3D viewport) and (D) React components.

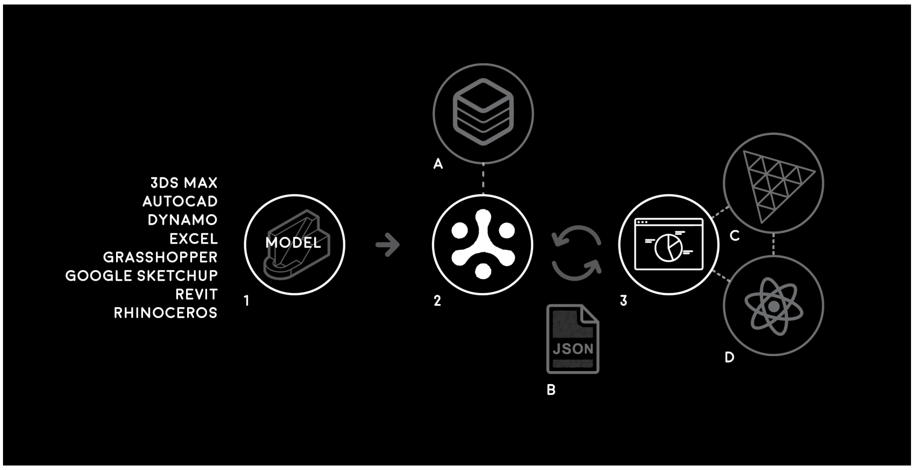

Figure 1. Workflow diagram of Redback BIM.

Updating a key's value To enable a correct data flow between Flux.io and Redback BIM, data, such as the value of a slider, must be presented and edited within Redback's interface. This slider's value is then submitted to Flux.io through implementing their SDK functions, changing the original state of the value in Flux's database to be updated with the user's input, which is then synchronised in the web application's state and viewport.

6.3. OBTAIN GEOMETRY DATA FOR EDITING IN WEB APPLICATION

JSON Data Table via mouse click As geometry in Flux.io is made of a JSON string, the next research step was to generate a data table which would be viewable and editable for stakeholders. Switching from HTML, CSS, and JavaScript to React was pursued due to its ability to dynamically update specific elements on a web page. In a collaborative effort between this research and [NAME WITH HOLD], an Engineer from [NAME WITH HOLD], a sample app was built to facilitate the development of the research outcomes. Following, using pure JavaScript, the JSON string of a selected geometry was obtained and extracted to generate a HTML table. Considering the data table of a selected sphere (see Figure 2), a problem in illustrating nested data emerges: due to the particular structure of the geometry's JSON interpretation in Flux.io, the data table fails to preview that specific set of information within its allocated cell. The areas highlighted in blue emphasise the nested JSON as '[object Object]', while, additionally, limiting the users' capacity to edit the HTML element.

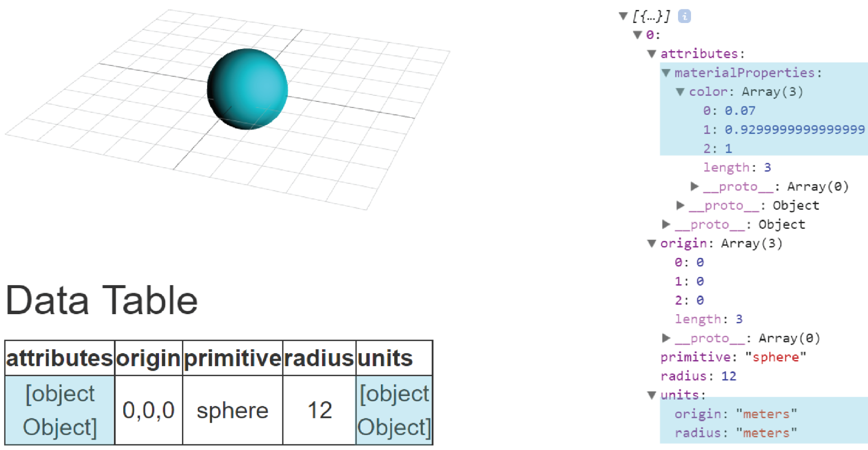

Figure 2. Data table previewed in the application when the geometry is selected. Right is JSON parsed into the console. Below left is the resulting HTML data table.

Open-source React JSON Tree Component An alternative solution was found through open-source components, resolving the inability to edit the data table or view nested JSON information which appears as [Object Object]. A JSON Tree component, created by Mac Gainor was implemented in Redback BIM. By selecting a key, a user automatically pushes its geometry data into the JSON Tree component, and changes made to it are transferred back to Flux, updating the database and the web application interface.

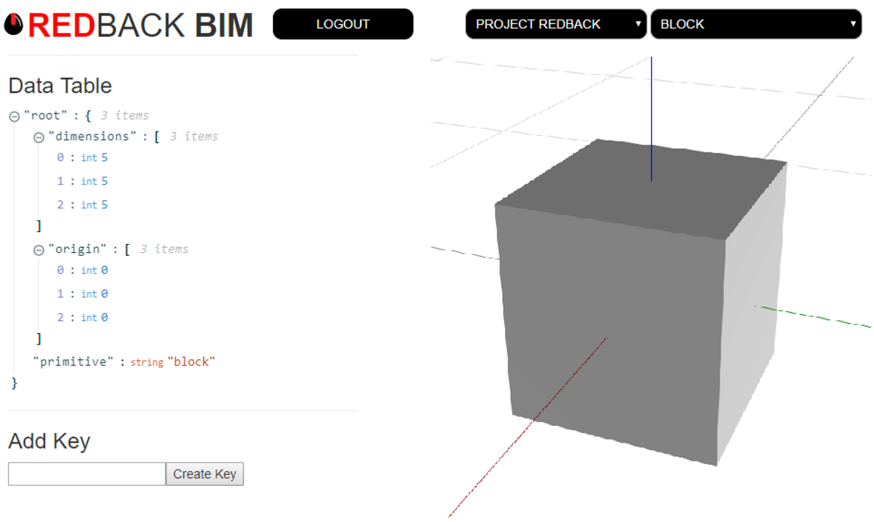

Figure 3. Redback BIM Flux Web Application.

7. Significance of Research

The paper concludes and argues that a direct link between 3D models and a virtual environment would benefit all project's stakeholders. Investigating academic papers and referencing existing web-based synchronous collaborative workspaces has contributed to defining a systematic workflow for successfully removing communication barriers in architectural practice. To do so, accessing, embedding and updating geometry data are required to be performed entirely online. By interconnecting JSON data to client-server interfaces for architects, designers, contractors and consultants, the current paper has demonstrated the ease of access to live information without necessitating additional intermediate tools. By implementing Redback BIM in the industry, multiple collaborators can converse in a live dialogue within their 3D architectural models, while facilitating informed design decisions. In addition, by implementing a combination of currently available open-source libraries and software, architects, computer scientists and creatives will be able to freely benefit of Redback BIM, while contributing to a community of like-minded computational designers, planners and tinkerers of 3D modelling applications. In conclusion, future projects might stem from the current framework, offering scalability as an open-source application.

8. Evaluation and Conclusion

This project focused heavily on interdisciplinary learning, incorporating areas of computer science and architecture into one platform. To facilitate both accessing and elaboration of a specific geometry's data through a web application, the geometry's JSON string is implemented on a web-based database (Flux.io). Correspondingly, a new automated means of providing real-time feedback and changes is developed that is accessible anywhere, revolutionising the traditional industry's streamline processes. Whilst the case study produced a successful synchronous collaborative workspace on a web-based platform, few key constraints and limitations should be further entangled. Firstly, many of the React components were not embedded with clear documentation or were deprecated for further use and development, resulting in the impossibility to be implemented within Redback BIM. Further, nowadays applications' success is heavily reliant on basic knowledge and experience with web development languages (HTML, CSS, JavaScript and React), 3D modelling filetypes (obj, json, stl), data structures (JSON) and data management concepts. Seeking to provide a simplified platform for storing, accessing, updating and tracking changes to a 3D model, Redback BIM currently fails to achieve the desired user-friendly interface that a typical localised software would offer and that would be otherwise possible in other web applications, such as Clara.io. Additionally, Redback BIM does not provide direct manipulation of the geometry in the Cartesian space, but only through the editing of its JSON script in the React component. Consequently, 3D model data can be uploaded from external programs to allow for integration or elaboration of their parameters, whereas the application has yet to include the option of creating geometry directly in the platform. With the emphasis on building applications, an area that is yet to be fully implemented is the use justification of Redback BIM in industry practice through user testing scenarios. Given that an objective of this research was to enable dynamic capabilities onto Redback BIM, the application is yet to be quantified for performance issues and qualitatively define the success of removing existing communication barriers in the process. However, these factors can be improved with further development and iterations. This case study should be considered as a first prototype in the development of a complex relationship between web-based tools and data within architectural models. More research and time invested in the progression of Redback BIM would assist in expanding its usability, efficiency and flexibility as an open-source tool between multiple stakeholders of a project.

Scholars within the field of architecture and computer science has defined data as an important asset which should be taken advantage of in the AEC industry. Therefore, the current research explored and developed the use of existing open-source WebVR and database technologies to building synchronous relationships between data and 3D geometry. With the introduction of Redback BIM, the inability to create geometry and need for a user-friendly interface will require additional research to justify its success. However, by bridging the areas of architecture and computer science, the ability to collaboratively organise and synchronise changes made to an architectural model in real-time has proven the success of building architecture-centric applications on the web, contributing to optimising communication practices and focusing our attention into solving real-world problems.

References

Counsell, J.: 2012, Beyond Level 2 BIM, Web Portals and Collaboration Tools, *In 2012 16th International Conference on Information Visualisation*, Montpellier, France, 510-515.

French, D., Stone, B., Nysetvold, T., Hepworth, A. and Edward Red, W.: 2016, Collaborative Design Principles From Minecraft With Applications to Multi-User Computer-Aided Design, *Journal of Computing and Information Science in Engineering*, **16(2)**, 021006 1-9.

Graham, P.: 2009, *Hackers & Painters: Big Ideas from the Computer Age*, O'Reilly Media, Inc..

Hearn, G. and Foth, M.: 2005, *Action Research in the Design of New Media and ICT Systems. In: K. Kwansah-Aidoo, Topical Issues in Communications and Media Research. pp.79-94.*, Nova Science Publishers.

Kim, S., Kuester, F. and Kim, H.K.: 2002, A global timestamp-based scalable framework for Multi-player Online Games, *In Fourth International Symposium on Multimedia Software Engineering*, Newport Beach, CA, USA, 2-10.

Aguirre León, E. A. and Molina, M. R.: 2009, New Interfaces, new scenarios. Vroom n. 0: Vroom n. 0: The emerging potential of collaborative 3D web platforms., *SIGraDi 2009 - Proceedings of the 13th Congress of the Iberoamerican Society of Digital Graphics*, Sao Paulo, Brazil, 227-230.

Marble, S.: 2012, *Digital Workflows in Architecture: Design - Assembly - Industry*, BIRKHÄUSER.

O'Brien, R.: 1998, "An Overview of the Methodological Approach of Action Research" . Available from <http://www.web.net/~robrien/papers/arfinal.html#_edn2> (accessed 01 November 2017).

Raisbeck, P.: 2016, "Knowledge Future: Future Proofing the Architectural Firm" . Available from <http://aca.org.au/article/knowledge-futures-future-proofing-the-architectural-firm> (accessed 16 October 2017).

Raymond, E.S.: 1999, *The Cathedral and the Bazaar*, O'Reilly Media, Inc..