

3.4 Glocal Dreams

“World domination,” Rodrigo said with a smile, his tone suggesting that the answer was obvious. I had just asked him to clarify what purposes he had in mind when he jokingly raised the question of whether knowledge of Chinese or Japanese would be more important for “his purposes.” It was Wednesday afternoon in May of 2007 and we were at an *a quilo* restaurant in Copacabana, a typical Rio lunch place selling food by weight. A block away was the office high-rise where Rodrigo’s company occupied a few small rooms and where both of us had been spending most of our time for the last few months, sharing a tiny office, the backs of our chairs just about a foot apart. It was perhaps a somewhat odd place to be plotting world domination. We laughed. A few minutes later we got up, paid, and headed back to the office. We did have a world to conquer, after all.

Rodrigo, whom the reader has met on a few occasions in the earlier chapters, is a *carioca* in his late 30s who has dedicated the last decade of his life to building a web development platform based on Lua, the programming language discussed in the previous two chapters. Since 2005, Rodrigo has pursued this goal in the form of an open source software project called “Kepler.” I first met Rodrigo during my time in Rio in 2005, he was in fact my first technical interviewee that year. While finding the project intriguing, I dismissed it as an outlier, choosing to dedicate my time to interviews with developers working in more typical companies. I stayed in touch with Rodrigo, however, and as I learned more about Kepler over the course of the following year, I gradually came to the conclusion that observing a project aiming to develop a global product based on local innovation would provide a useful contrast to a study of a “typical” company, focused on bringing foreign technology to Brazil. (See chapter 3.1.) When I returned to Brazil in 2007, I decided to accept Rodrigo’s invitation to study his project and his offer of a desk in his office, using it as a base for my investigation of Lua (see chapters 3.2 and 3.3) as well as for looking at Rodrigo’s own project—the subject of this chapter.

Kepler and the company that has sponsored it stand in important contrast with both Alta and Lua. In contrast to Alta’s founders, Kepler represents an attempt to create a global platform—a software product developed for the world rather than for the needs of specific local clients and meant as a foundation for other software development rather than a product in itself. While Rodrigo’s talk about “world domination” is a joke (as are his colleagues’ references to him as “the Brain” of “Pinky and the Brain”), the project does have global aims, positioning itself in competition to globally popular web development technologies.³⁴⁰ Its audience was also primarily outside Brazil.

Unlike Lua, however, whose success can be attributed to a combination of pre-existing foreign ties and successful disengagement from the local context, Kepler had to create crucial

340 To fully appreciate the challenge, one must note the massive role of network effects for software platforms. In other words, the value of a software platforms depends on the number of people using it. See Katz & Shapiro (1986) for the general discussion of network effects.

local alliances, on which it came to rely. One of those alliances involved betting on Lua itself—a programming language developed just a few kilometers away from Rodrigo’s current office. Rodrigo also decided to work together with a local company oriented towards Brazilian clients, and to rely on funding provided by the Brazilian government.³⁴¹

Paradoxically, local use of local innovation is itself a starkly non-local move. Using local innovation is something software companies in Silicon Valley do all the time. It is not commonly done in Brazil. Rodrigo’s project is thus simultaneously global and local in important ways —“glocal” to use Wellman and Hampton’s (1999) term. This chapter shows some of the challenges involved in pursuit of such “glocal” dreams.

Attempting to bring together the resource of local universities, the local industry and the national government in pursuit of innovation that aims to be global in its significance, Kepler represents the kind of innovation that may be crucial for development. Understanding its challenges may thus provide valuable insights into the dynamics of technological development at the periphery.

Making Waves

Rodrigo Miranda grew up in Rio, as a son of an engineer and a journalist. As a child, he wanted to become an architect, but an encounter with a computer at the age twelve led him to a new passion. (See chapter 2.1 for a discussion of Rodrigo’s early days as a “nerd.”) By the time he was choosing a college program, Rodrigo knew he wanted to study computer science. He was planning to apply to the public Federal University of Rio de Janeiro (UFRJ), but PUC-Rio opened an undergraduate program in computational engineering at the last moment.³⁴² Since his mother was teaching at PUC at the time, Rodrigo managed to get a scholarship to study there.

Rodrigo sometimes describes Kepler as the third of the three “crazy ideas” that he had pursued over the years. The first, an idea for a video game, had to be put on hold back in college. Rodrigo did not believe he could do it all by himself and could not convince his friends to join forces.³⁴³ The second vision, a hypertext database system, was abandoned in 1994 when Rodrigo saw the Mosaic web browser. The web, however, gave Rodrigo his third idea, one he had has

341 Lua of course was also to some extent funded by the Brazilian government and PUC, itself an actor tied closely to the local context. In Lua’s case, however, the funding system appears to have stabilized a while back, now forming a part of the infrastructure that is almost invisible to the actors. By contrast, Kepler’s alliances with the funding agencies are new and highly problematic, requiring re-thinking and re-negotiation.

342 PUC earlier had an undergraduate program in “informática” but Rodrigo found it outdated. According to a PUC faculty member, department’s “informática” program is generally more popular among lower middle class students, perhaps because it is taught at night. The newer computational engineering program is considered substantially more prestigious.

343 After reading a draft of this chapter, Rodrigo commented that while the project had been on hold for years, he had never fully given up on it. He could not find money at the time and his friends would not work for free. “But now there is the ‘open source way,’” Rodrigo commented.

pursued for over a decade since: a web development platform based on Lua.

Lua, whose story was presented in the two previous chapters, is a programming language developed in Rio de Janeiro by a team of three people, headed by Roberto Ierusalimschy, who advised Rodrigo during his undergraduate and master's studies, as well as through Rodrigo's two-year stint as a Ph.D. student at PUC. As described in chapter 3.2, Lua is today used primarily in computer games, a domain for which it is often seen as being particularly fit because of its performance and ease of integration with C. Few software companies in Rio de Janeiro, however, build products of that kind. Instead, most provide software services, which typically involve development of web applications. Lua is rarely seen today as a good foundation for web development.

Yet web development in Lua has a long history. The first attempts were made in 1995, by a group of PUC students. According to "Cássio," one of the first students to apply Lua to web development in a project called "LuaForm," the effort started as "a game among friends," driven primarily by a desire to understand the emerging web technologies. Cássio was pursuing his Master's degree at the time, while working at Tecgraf, together with a number of friends. He and others working at Tecgraf at the time describe the laboratory as an intellectually stimulating place where work projects, academic research and things done just for fun transitioned one into another easily. "So, in this environment we produced this toy [*brincadeira*]. It was called LuaForm at the time. Just because we wanted to have a way of handing HTML forms," he explained to me. "Just simply to learn, to discover how things worked then."

Little by little the project grew features. "And from there, one thing leads to another, right?" Cássio explained in our interview.

Cássio: Given that you already have the data in the Lua environment, with pretty parameters, why don't I... I could just do a *print*, right? And that would produce a response to this form. So, it would write the page based on what I would put in the "print" below. So, from here it could be made even easier, because instead of writing "print '<H1>'," it would be easier to have something generate those *strings* for me.³⁴⁴

The project was then picked up by another student, who developed it into a Master's thesis (completed in 1998) and published several papers about the project (Hester et al. 1997, 1998).

CGILua, as the extended system came to be called, had a number of advantages over the better known alternatives, and in particular over writing web applications directly in C, a common practice at the time. While the first iterations of many currently popular methods for building web applications were being developed around 1996, none were really well known. For a short time, CGILua was about the best approach one could use for building web applications.³⁴⁵

344 See appendix C, "Original Interview Quotes," (Cássio, April 2007, "Uma brincadeira entre amigos"). Cássio's quotes from the previous paragraph are included in the same original quote.

345 See Hester et al. (1997) for a discussion of CGILua in comparison to a number of the better known alternatives. The paper does not, however, compare CGILua to PHP—a system that had been available since 1995 but was relatively unknown at the time. PHP was very similar to CGILua in a number of ways and came to dominate

As with Lua, CGI Lua used English keywords and the papers about it were written in English. Other documentation, however, was written in Portuguese and the software was never widely advertised outside PUC. It was not announced on the Lua list until 1999, when the documentation was finally translated into English.³⁴⁶ At the time when CGI Lua started getting attention on the list, the student who was the primary author of CGI Lua had moved to United States and web platforms built in other languages were much ahead. “Lua lost the web wagon,” says Roberto Ierusalimsky.

Meanwhile, Rodrigo had quit his Ph.D. program at PUC in 1996 and went to work as a manager in a web development company, which at the time built web applications in C and had a strategy not very different from Alta’s. Being acquainted with the students who were working on CGI Lua, Rodrigo understood the time savings that CGI Lua offered over building applications in C and proposed using it for the client’s projects. Rodrigo says that the manager refused to even consider the idea, finding Lua to be a toy language. Rodrigo proceeded to introduce Lua in the company surreptitiously, while also starting to think about the possibility of building a complete web development platform based on Lua. A year later he left the company. Another year later, Rodrigo found a new opportunity to pursue his dream of a web development platform based on Lua.

While Rodrigo was looking for opportunities to pursue his technological dream, his brother “João” was pursuing his own. Unlike Rodrigo, João saw himself as an entrepreneur. His dreams, therefore, focused on starting and running a successful business. In 1997, having earned a small amount of money in a venture that provided software localization services to foreign software companies, João wanted to move to a new level: launch a *product* company based on proper development methodologies. I refer to this company here as “Nas Nuvens,” a self-deprecating pseudonym suggested by Rodrigo. “Nas Nuvens” means “in the clouds” in Portuguese, a name that suggests at the same time a degree of disconnection from reality (much like its English equivalent) and an association with Lua through an allusion to the phrase “no mundo da Lua” (lit. “in the Moon world”), a more idiomatic translation for the English “head in the clouds.”³⁴⁷

João wanted to start a US-style company, hoping to distinguish himself from the local competition by a strict adherence to foreign software development methodologies and, as far as possible, foreign business methods. In a particularly stark attempt at following foreign methods, João decided to build his company around local research—a practice standard in California but rarely pursued in Rio. Having heard Rodrigo’s praises of Lua, João decided that Lua could prove to be a key strategic advantage for a software firm based in Rio: close access to the Lua team would help the company stay abreast of any changes and perhaps influence Lua’s development to its own advantage, while PUC would provide a steady stream of interns and employees skilled in the use of Lua. João also decided to start with a focus on the needs of the domestic market, perhaps expanding internationally in the longer term.

web development a few years later.

346 Two messages mentioned it in passing in 1997 and 1998, the first of which was written in Portuguese.

347 The name captures Rodrigo’s doubts about whether the venture was ever realistic, but should not be interpreted as signifying my own position as to whether the efforts were worthwhile.

As João saw it, the local market needed an easy way to build websites with dynamic content. Rodrigo's experience with CGI Lua had proven that this could be done with Lua. Using his ties to PUC, João acquired a web publishing system based on CGI Lua that was built by another PUC student, using it as a starting point for Nas Nuvens' future product. He also managed to find start-up capital from among relatives, friends and a PUC professor. Soon after, Rodrigo joined the company as a chief software architect, seeing the venture as a chance to take CGI Lua to the next level and build web development platform based on Lua in pursuit of his own technological vision. The two brothers were pursuing two different dreams—one looking to create a global platform with relatively little interest in the financial aspects of the venture, the other looking build a high tech company using a foreign model, less interested in the details of technology to be built. Their parallel pursuit of their professional dreams, attempting to reproduce the practice centered in the same place, allowed for an alliance, much like many of the alliances described in chapter 2.2.

The company's product was innovative for the time and the company successfully obtained small innovation grants from FINEP, an agency that funds industry research. João's superb skills as a salesman then allowed him to quickly attract substantial capital (about a million reals³⁴⁸), hire development and actually build a product, something few entrepreneurs could do in Brazil even at the height of the dot-com boom. As the product was nearing completion, however, João faced a new challenge: finding additional money to actually launch the product, another two million reals in João's estimate. In 2000, as the clouds were starting to gather over the Internet industry, attracting additional money turned out to be a challenge even for João. Banks he turned to asked for no less than 50% annual interest, João told me, while some wanted considerably more. A large foreign company (a household name) offered to invest as much as two million reals, but did not want to do so alone. João eventually found what appeared to be a solution: the Brazilian Development Bank (BNDES) offered to fund the project at the interest rate of 30% a year—an exceptionally low interest rate for Brazil. João knew that the bank would take months to make a final decision, but decided to take the risk, seeing few alternatives. The bank eventually offered João the money, but demanded that he would personally guarantee the loan, while giving the bank an option to convert it into equity if the venture was to succeed. Such terms left João with a risk of finishing with millions reals of personal debt³⁴⁹ while giving the bank most of the profit if the venture were to succeed. In João's view, this perverted the meaning of "risk capital." He decided to not take the loan and accept that the key ingredient of the California startup recipe was missing in Rio de Janeiro and the plan had to be rethought.

Nas Nuvens was soon out of money and had to lay off many of the developers, including nearly all of the most skilled ones, who were also the most expensive. João ended the lease on the two floors that Nas Nuvens used to occupy, limiting the company to the few rooms it has occupied since. The company shifted its focus to providing services to the customers it already had—the approach that many consider the only realistic route for a software company in Rio de Janeiro anyway. Even this option, however, was turning out to be problematic. Unlike its

348 A US dollar was worth around \$1.80 reals in 1999 and 2000. So, a million reals was a little over half a million US dollars.

349 Note that Brazil has much stricter personal bankruptcy laws than the United States, so an individual with a large personal debt cannot easily start his life over again.

competitors, Nas Nuvens had build up expertise in building web solutions using Lua and thus had to rely on CGILua, a rapidly aging platform, and could not provide much of what the clients were starting to expect. The author of CGILua had by then departed for the United States and the CGILua was largely abandoned by the PUC Lua community, which was looking to solidify Lua's success in other domains (see chapter 3.2). (João's expectations of steering Lua's development had proved unrealistic—a tiny local venture was too insignificant, considering that Lua was getting adopted by major players such as Adobe and Microsoft, whose interests were quite different from those of Nas Nuvens.) Shifting towards services and a closer relationship with local clients was also making the job less and less attractive to Rodrigo, who had wisely abstained from financial involvement with João's venture and could find better opportunities elsewhere.

The solution came in the form of government support. By that point the company had already received small grants from FINEP—a government agency tasked with supporting industry research.³⁵⁰ By 2001, FINEP showed less interest in the company's product, no longer considering it particularly innovative. In 2002, however, FINEP started providing additional funding earmarked for open source projects.³⁵¹ It was thus decided that the company would apply for FINEP funding for an improved web platform based on Lua to be released as open source. The project would make it possible to hire skilled developers from among PUC students and alumni and would satisfy Rodrigo's desire to pursue his grand vision. It would also give Nas Nuvens a better Lua platform, which it could use to provide more competitive services. The project was called "Kepler." As the project's website explained, the name alluded to Johannes Kepler's discovery that tides on Earth were caused by the Moon, and was to suggest that Lua ("moon") was about to cause some tides. For the six years since, Rodrigo has been seeing himself as primarily the leader of this project which he runs in something of a partnership with Nas Nuvens.

The funding provided by FINEP was expected to be rather modest, and the company could not use it to hire as many qualified programmers as it expected. Rodrigo's connections to PUC networks, however, helped him find students and recent graduates already interested in Lua and willing to dedicate part-time efforts for less than the market wage. One of the new project members was "Márcio"—a recent PUC graduate working as a programmer, on a PUC web application based on CGILua. A dedicated fan of CGILua, Márcio had worked privately to improve the platform, without advertising his efforts. He had little interest in Rodrigo's plans for "world domination." Rodrigo's offer of paying him to work on Kepler, however, provided a good

350 Observing the lack of collaborative relationships between industry and local research and unwilling to wait for companies to start building such relationships on their own initiative, the Brazilian government requires companies to contribute money to "sectoral funds" which are then used to fund collaborative projects such as the one described in this chapter. (Though typically the funding occurs on a much larger scale, according to my conversation with a FINEP grant officer.) The distribution of those grants is managed by FINEP. In addition to FINEP's sectoral funds, Kepler has relied on money from other agencies, such as CNPq (an agency responsible for funding academic research) and funding agencies of the State of Rio de Janeiro. I avoid a discussion of the complex web of funding relationship, as such a description would require a tome of its own, focusing instead on FINEP, the major source of funding.

351 See Schoonmaker (forthcoming) for a discussion of the Brazilian governments rationale for supporting open source.

opportunity to dedicate more time to improving CGI Lua by not having to take on other consulting projects. Others joined the project partly out of interest in Lua and partly as a favor to Rodrigo. (See chapter 3.1 for Alta's early participation in Kepler.)

Rodrigo also returned to the Lua list and resumed his efforts to transform the Lua community to fit his needs. Lua is typically used as a programming language embedded inside a larger C application, taking advantage of Lua's small size, efficiency and ease of integration. Lua's community had consequently emphasized minimalism and use of highly customized solutions. The members of the list had thus often preferred to share ideas rather than actual code, stressing that one project's code would rarely provide a perfect fit for another project's needs. Development of web applications, on the other hand, required a large collection of software modules that could be readily used. Realizing that he could no longer hope to fund the development of all the modules he needed, Rodrigo attempted once again to convince the Lua community of the importance of sharing code and assembling a larger collection of modules. He refers to his efforts as "culture farming." Since Kepler's own software was now going to be released under an open source license, Rodrigo used this as an opportunity to jump start the sharing culture, actively announcing the release of each Kepler module on the mailing list and setting up a website called LuaForge for sharing modules.

When I first met Rodrigo in 2005, Kepler was relatively well-funded and Rodrigo had a number of skilled developers working on the project. A later gap in funding decimated the team, as most Kepler developers felt that working for free was a luxury they could not afford. (Márcio took a somewhat different approach, continuing his involvement in the project but refusing to accept money for it. This gave him the freedom to pursue other sources of money without constraining himself with a commitment to Kepler, which was too unreliable as a source of income. It also started, however, his gradual disengagement from Kepler.)

While the development had slowed down substantially, the work done in 2005 was starting to pay off by early 2007 as the project was slowly attracting some users. In Spring 2007, the project's mailing list (run in English) had around one hundred people, who were increasingly involved in the project, at least to the point of asking questions. Additionally, the Lua community was gradually warming up to the idea of sharing modules. LuaForge included around two hundred projects, with several new projects getting registered on a typical week. This was giving Rodrigo hope that the project was going to succeed. "My friends used to call me crazy, now they just say I am insane," said Rodrigo.

Opening Kepler

On one of my first days in Rio in March of 2007, Rodrigo told me he had a habit of going to PUC roughly once a week, to talk to the two Kepler contributors who worked there—Márcio, the PUC graduate who was working in PUC's IT department, maintaining an administrative web application based on CGI Lua, and "Tiago," a Ph.D. student of Roberto Ierusalimschy. (A third major contributor, "Alan," used to be there as well but had just recently moved to Porto Alegre,

about a thousand kilometers south-west of Rio de Janeiro.) Occasionally, he also met with Roberto, Chico and others members of the local Lua community.

A week later, I was in a taxi with Rodrigo, going to PUC where I expected to sit through his typical meetings, observing the routine. As soon as I got into the taxi, however, Rodrigo handed me a printout of a web article, in English, entitled “Financing Volunteer Free Software Projects” (see Hill 2005). I scanned it briefly. The article argued that paying open source developers is a dangerous practice, since paid labor would tend to “crowd out” volunteer contributions. In other words, paying some people would make others less willing to work for free, and even those that are paid might work less, since they would now see their work as an economic transaction. This article, explained Rodrigo, had helped him identify the problem that had plagued his project for a long time. He was releasing his source code under an open source license, but was not running Kepler as a “real” open source project. He was hiring developers with the money provided by FINEP, and they worked while they were paid. When the money dried up (as was the case for most of 2006), the work stopped. Plus, he simply could not hire all the developers he needed: FINEP funding was limited and could only be used to pay people in Brazil. He had hoped to get people from outside to participate, but this had not happened. He had to get them involved. Perhaps this would require that he stopped paying his current developers, even though the project depended on them.

We arrived to PUC and went to meet Tiago, a Ph.D. student of Roberto Ierusalimsky, whom Rodrigo paid to work on Kepler a few hours a week. We found Tiago in a small room that he shared with three other Ph.D. students. The three of us went into a conference room across the hallway and sat down under a white board covered in set-theoretic formulas, half in English and half in Portuguese. Rodrigo told Tiago that he wanted to start running Kepler as a “real” open source. They were a project with open-sourced code, he said, but a closed-source development method. He wanted to change this. Perhaps the fact that Alan had just moved to Porto Alegre would make this easier: Alan’s departure already made it impossible to resolve all the issue in face-to-face meetings at PUC, and they had to discuss most of the decisions by email and IM. Now they just had to take it a step further. He wanted to start discussing more things on the mailing list, openly, explained Rodrigo. Of course they would use English for those discussions. Kepler already had a mailing list operating in English, used primarily for announcements. This list could become the center of the Kepler project, the new forum for the discussions that up until now occurred inside PUC walls. Tiago listened to Rodrigo speak, occasionally asking clarifying questions. He looked concerned, though it was hard for me to tell whether he was unhappy with Rodrigo’s plans or with my presence there, after all, we had just met a few minutes ago. (As I learned from my later interview with Tiago, he supported Rodrigo’s idea to “open” the project, but was concerned that Rodrigo was spending too much time focusing on the process.)

“This finishes the Weird Ideas of the Week part,” finally said Rodrigo, “Now the practical part.” Rodrigo and Tiago spent the next half hour talking about specific problems with the launch of Kepler 1.1. Even so, Rodrigo’s suggestion of changing the model repeatedly seeped back into the discussion. They needed people to test, Rodrigo said. Perhaps people at Nas Nuvens could help, but this was also a sort of thing that list members could assist with. Perhaps there should be a list of tasks on the website where people could go and see what needs to be done. Rodrigo gave an example: a new person had appeared on the list, asking what he could help with, but Rodrigo

did not know what to tell him. While yearning for outside participation, the project was not organized so as to take advantage of it. There should be a page with tasks, said Rodrigo, some of them in red. In this case, they could tell new people: go to that page, pick a task.

We walked Tiago back to his room, chatted briefly with other Ph.D. students there (as I later found out, Rodrigo was working on recruiting one of them to work on Kepler), then headed to meet Márcio, the other contributor. We were going to meet Márcio outside, near a kiosk that served coffee. Márcio liked short meetings, Rodrigo explained, and thus preferred to meet on a bench rather than in a comfort of a conference room. Márcio had recently asked to not be paid for his work on Kepler, to have fewer obligations to the project. The meetings, therefore, had to be done in the way he wanted to do them.

While waiting for Márcio, I asked Rodrigo about the move to English. It was simply pragmatic, he explained. He had to draw on the developers outside Brazil, so he needed a mailing list in English. He did not have the time to maintain two lists. Brazilians interested in Kepler would know how to read English and usually would know how to write it as well. Plus, he continued, there simply was little benefit to discussing things in Portuguese. Kepler had almost no documentation in Portuguese, so there was no meaningful way of engaging with the developers who did not read English. He illustrated with an example. *Let's say someone would write to the Portuguese list and ask, in Portuguese, "Where is the documentation?"* What would Rodrigo tell them? He could only point them to the documentation in English. Requiring Kepler contributors to discuss all the project plans in English on the mailing list would perhaps inconvenience them somewhat. Rodrigo was confident, however, that they were capable of doing this and would agree to do it if asked. When Márcio came down, Rodrigo presented him with a much condensed version of what he said to Tiago. Márcio nodded, seemingly finding no problems with this. In somewhat of a contradiction to the new plan of the moving decision making from face-to-face discussions to the mailing list, Rodrigo and Márcio then proceeded to quickly discuss the state of specific sub-projects.

A week later I had a chance to talk to Márcio about the proposed changes. He appeared divided. He did not share Rodrigo's desire to conquer the world, he explained. He did not care if Kepler was popular, he said. He wanted it to be *good*. That was the difference, he repeated: "Rodrigo wants Kepler to be popular, I want it to be good." When Márcio started working on Kepler, he did not expect it to be successful—it was a fun project that also paid some money. Now it had succeeded and Márcio hoped that it would succeed more. It was good for the world to know, he continued, that here in Rio de Janeiro they were doing something interesting. For this reason, he supported Rodrigo's plan, even as he did not see Kepler's future success important to him personally. He agreed that further success would require opening the project to outside participation and moving all decision making to the English mailing list. Doing so meant more work, but the effort was worth it for Kepler's success. But sometimes, he then added, he found himself just deciding not to write.

A week later, Rodrigo and I were again at PUC, sitting at a picnic table not far from the Informatics Department, together with two other people. One was a potential future contributor whom Rodrigo was courting at the time. Another one was Renato, a close friend of Rodrigo always willing to lend Rodrigo an ear when Rodrigo needed a sympathetic listener. Rodrigo

talked again about the idea of moving on an open source model and getting people on the list to contribute more. He did not really know how to do that, he said, since he had never run this kind of project before and his knowledge was only theoretical. (“I have read books, articles on the web, talked to Yuri,” he said, naming three foreign sources of information.³⁵²) But it had to be done. Kepler could not just rely on the efforts of local developers funded by FINEP. He had to look for people outside. This “outside” (*lá fora*) was potentially ambiguous, as it could mean either outside the project, or abroad. In this case, however, the two largely coincided. Looking for help outside the project meant looking for it abroad, through Lua’s mailing list. As Rodrigo talked, the rest of us just sat listening and asking clarifying questions. Our job was to help Rodrigo feel that his plan was not altogether crazy.

When we got up, it was clear that the decision had been made. Next day, an email to the Kepler list informed the subscribers that the list was now going to function as a “real” open source project. The lengthy message, entitled “Opening Kepler,” started as follows:

Hi,

This mail got a lot bigger than I imagined at first... :o)

As you have noticed, the conversion of Kepler to Lua 5.1 is taking a lot longer than expected.

Not only we have found that this involves more work than we assumed, but also that the development model being used until here is not working as well as we would like.

The Kepler team is currently using a model that involves too much communication outside the public channels (this list and the site for example). We are trying to educate ourselves in order to change that, but this is not exactly easy for a team used to rely on interpersonal communications.

After laying out a detailed plan for specific Kepler components, Rodrigo concluded by saying:

As I hope you can notice by this mail, we are trying hard to move to a more open development model. That includes using this list in a different way and opening the site wiki for others to contribute.

Before going on, we would like to know what you think about the general idea and what other ideas could be added to this "new vision".

352 In this particular case, Rodrigo was referring to our conversations in the day between our first visit to PUC and this meeting, in which he had asked me for my opinions about what he was planning to do. More generally, throughout my time in Rio, Rodrigo and I had numerous conversations about his project and he occasionally asked for my opinion. In cases when my advice was requested, I provided it without reservations. I also allowed myself to express unsolicited opinions to the same extent as I saw other contributors doing it, but trying to limit such suggestions to specific technical matters rather than the larger strategic concerns. According to Rodrigo, however, the *questions* I asked may have been the most important source of influence on the project, as thinking about the answers had led him to occasionally rethink his approach.

Thanks in advance for any suggestions and thanks for reading that much...

Rodrigo

The three subscribers who responded to Rodrigo's message expressed support for the plan. For some of them, Rodrigo's decision to "open" Kepler may have seemed obvious and natural.

Working the List

As we came back from lunch on Wednesday, we turned our attention to the task at hand. Rodrigo and I were working on a chapter for an upcoming book about Lua programming. The chapter presented an opportunity to promote Kepler, putting Lua's use for web development side by side with such accepted uses of Lua as embedding it in games or running it on micro-controllers. Our chapter focused on implementing in Kepler a "Model-View-Controller" (or MVC) application. MVC was a popular approach to organizing software applications and had been growing in popularity in web development for the last few years. Allowing MVC development in Kepler was one of Rodrigo's projects for this year. This goal, which seemed natural to me, was not shared by all of Kepler's contributors, Rodrigo complained. Márcio, in particular, felt no need for it, preferring the methods that Rodrigo considered outdated. Rodrigo talked about having to explain to Márcio that he had to implement MVC "to please his friends." Unsure whether the phrase referred to the foreign users of Kepler, on the mailing list, or to people in Rio, I decided to ask. Being in perhaps too jovial of a mood, I phrased the question in a somewhat unfortunate way: "Are you talking about your real friends or your imaginary friends?" Rodrigo was taken aback by the question and asked me if I knew what "imaginary friends" means in Brazil. I told him it meant the same in the United States and apologized.

I do have a real friend, said Rodrigo. He mentioned Renato—a close friend who was with us at PUC when Rodrigo was finalizing his decision to open up the project. Renato was always willing to listen and be supportive. Perhaps too much so. The problem was that Renato would say "This is very interesting" to almost anything. Renato was too busy with his own job to follow Rodrigo's interest in depth. He was willing to express support, but had no ability to accompany Rodrigo in the journey.

The conversation moved back to Kepler contributors—not exactly friends, but perhaps colleagues, Rodrigo pointed out. For some of them, such as Tiago, Kepler only made sense with MVC. Others, like Márcio, saw no value in this approach. Rodrigo had to find a way to navigate between them.

We returned to work. I read a section Rodrigo had drafted, then proceeded to work on the next section, which we had agreed I would write. Rodrigo, meanwhile, dedicated himself to implementing a change to the Kepler code that we had discovered we needed to present a more elegant example in our chapter. (The chapter was describing a version of Kepler that was yet to be released, so we had the freedom to change Kepler to fit our description of it.)

By 4:30 p.m. Rodrigo had completed the change and sent out a lengthy email to the Kepler mailing list, describing the problem we had encountered and the proposed solution. It concludes with a request for comments. “What do you think about the change?” As he finished the message, he shouted out, as if talking to the people on the list: “Answer! Because I am here lonely, sitting in this room with a crazy Russian guy!” After reading his message, I posted a brief response, pointing out an additional benefit of the proposal and raising the possibility of an additional change. Rodrigo saw the new message in his inbox. “Oh, someone answered!” he exclaimed. “Oh, my friend Yuri!”

As I read Rodrigo’s message, however, I noticed a problematic ramification of the change we had proposed, realizing that it would break something else that our examples depended on. I turned around and mentioned this to Rodrigo. We went to Nas Nuvens’ lobby and planted ourselves in the two bean bags. We proceeded to discuss the problem for a long time, eventually coming up with a better solution. Rodrigo returned to his seat and wrote another email, describing the problem we had found and the new solution. “Again, if someone is still with me, comments are welcome,” concluded his message.

Another hour later, while Rodrigo and I were having a late dinner around the block (a different place from where we had lunch, since we were celebrating the day’s success), a list member “John” responded to Rodrigo’s second message with a short follow up question. His message ended up being the only one in this thread written by someone outside Rio de Janeiro. (Of course, we did not know where John was, but it was safe to assume he lived abroad.) Rodrigo responded to John’s question after getting home; I replied to Rodrigo from home in the middle of the night, this time disagreeing with his response. Another message from Rodrigo, just a few minutes after mine was the last one in the thread. Next morning Rodrigo committed the change to the code repository.

This email conversation on an English mailing list, in which all but two sentences were written by the two of us, who spent most of our day sitting in the same small room, may seem strange and almost farcical. We were resolved, however, to continue developing Kepler in an “open” way, hoping that someone will join us eventually. Turning Kepler into a global project, a trans-local place where space would seem to no longer matter, would require substantial work on the ground. We did not expect it to be easy, and even small successes counted.

As we were having dinner, I asked Rodrigo if he really felt that he was alone, talking to himself, as some of his earlier comments suggested. He disagreed. He had a hundred people on that mailing list, after all. He did not expect them to write code, it was enough that they read what he wrote and comment, if only occasionally. This, pointed out Rodrigo, was *much* more than what they did up until March!

So, they are becoming a good audience, I thought to myself. They were not good collaborators, at least not yet, but they were now behaving as a good audience. Before that Rodrigo was performing without audience, now he had one. Instead of thinking of myself as his only interlocutor, I concluded, I should perhaps think of myself as a side-kick in his performance.

Noting that Rodrigo had said that he did not expect the list members to contribute code, I asked him if he thought that opening Kepler had the effect he hoped for and whether the list members were as involved as he wanted them to be. (The project had been “open” for over a

month, and it was perhaps too early to expect great results, but it was worth a discussion.) He did not expect them to do much, said Rodrigo. But maybe they will, he then added. The distinction between expectations and what *might* happen reminded me of another conversation I had that day.

In the early afternoon, I stepped outside of Rodrigo's office to get some water. There I ran into Pedro, one of the Nas Nuvens' employees who worked in customer support as well as some software development. While he was still working on his undergraduate degree—and was doing it through a night program at a university that can hardly be compared to PUC—Pedro was considered by Rodrigo to be one of the most promising people at Nas Nuvens. (All PUC graduates had long left for better salaries. A few months later Rodrigo helped Pedro get into PUC's prestigious Master's program in Computer Science. Another year later Pedro left the company, having become overqualified for the salary Nas Nuvens could pay him.)

Pedro was dressed up to a surprising degree: black pants, white shirt, a tie. He had sunk himself into a bean bag, which made his attire seem even more out of place. It turned out that he was presenting the final project for his undergraduate degree later that day—a web application written in Ruby-on-Rails, a recently popular web framework for Ruby, a programming language sufficiently similar to Lua to be seen by some as one of its main competitors. (Ruby was developed in Japan, was little known for years and then suddenly exploded in popularity. As another *peripheral* language and a former underdog that suddenly made it, Ruby has a special place in the imagination of the Lua community.) We talked about the report Pedro had to turn in for the writing project. I asked in which language he wrote it. “In Portuguese,” he replied, “It *has* to be in Portuguese.” Intrigued by his “has to be,” I asked in what language he had written the code. The code was all in English, explained Pedro. Unlike the report, the code could be in English, since only the advisor had to look at it, and the advisor knew English. So it was all in English, he continued: the names of functions and variables, as well as the comments—all but the report. “Why?” I asked. Pedro laughed before answering. “Out of megalomania,” he said then. He explained that he and his partner wanted to think that one day people abroad would be using their code, perhaps even contributing. Writing the code in Portuguese would exclude all of those people.

Of course, making their code in English excluded some Brazilians too, continued Pedro. But those people were already excluded. The code was based on Ruby-on-Rails, which was documented only in English. Ruby-on-Rails function names were in English too. One could not work with it without knowing English. If Pedro were to start a company around his final project, he would not consider hiring anyone who did not know English. But again, he summed it up, it was also about megalomania. What if he decided to hire a developer in India? If their code were in Portuguese, they would not be able to do this.

Was Pedro joking? On the one hand, he had to be. Someone in California talking about “hiring a developer in India” has most likely worked closely with developers from India (perhaps having found himself on more than a few occasions as the only non-Indian in a conference room) and may know people who have traveled to India's outsourcing capitals. The decision to hire a developer in India would thus be a practical question, a matter of cost-and-benefit analysis. Things could not be more different for Pedro, who had never met anyone from India or anyone

who had traveled there.³⁵³ Pedro's wages (and those of most programmers in Rio) were hardly high enough to justify hiring programmers in India as a matter of cost savings. The talk of outsourcing was thus not a matter of planning for cost savings, but a matter of dreaming about one day entering the same field with the big global players.

This dream was so far-fetched that Pedro himself called it "megalomania." Having learned those global dreams from the global technical culture in which he engages virtually, Pedro maintained an ambiguous attitude towards them. This was not something he was ready to defend publicly as a plan for action and he was ready to mock his own global imagination calling it "megalomania." Yet, he did write all his software in a foreign language, entertaining the possibility that those dreams *just might* come true.

Pedro's story makes explicit the ambiguity of global imagination that came up in more subtle ways in numerous conversations with Brazilian software practitioners. This ambiguity could be understood in two ways. One approach is to liken it to what Favret-Saada calls "I know, but still..." in her discussion of witchcraft in France (Favret-Saada 1980). Favret-Saada describes the ambiguous approach to witchcraft, which combines the public acceptance of the rational view (*I know* there are no witches) and suppressed belief in witchcraft, rarely verbalized and quickly withdrawn upon questioning, yet strongly affecting the practices. Global dreams show a similar duality, being powerful enough to actually affect practice, yet not defensible in public and eagerly labeled "megalomania" upon interrogation.

A somewhat different, but closely related, approach is to look at such global dreams as a game, a make-belief, a simulation of a global practice. Regardless of whether Pedro's project has a global future, writing the code in English and imagining future plans for hiring programmers in Bangalore could similarly be *entertaining*, a way of doing in imagination what could not be done in reality.

Like Pedro, Rodrigo had a "megalomaniac" dream of running from Rio a major international open source project, a web development platform that could compete with the popular frameworks for Ruby and Python. Like Pedro, he repeatedly described the dream as a "crazy," a part of his half-humorous plan towards "world domination." The word "crazy" (*maluco*), was in fact one of the most common words that he applied to himself, his work and the people he respected. Unlike Pedro, Rodrigo had actually invested a number of years of his life into his "crazy" dream *publicly*. He thus could not as easily dismiss it as a joke. Instead, he embraced the image of a crazy dreamer. Even so, many of his plans fit within the "I know, but still" space.

One way to deal with this gap was to maintain a healthy distinction between things that were to be discussed and things that were actually to be done. As I started getting involved with Rodrigo's project hands-on, I was quickly overwhelmed by the flood of ideas Rodrigo had for how my part of the project could be improved. As new ideas were coming before old ones could be implemented, I told Rodrigo that we had to slow down—I only had so many hours in the day and had to reserve time for writing field notes and conducting interviews. In response, Rodrigo pointed out, that the problem was arising simply because I was taking his suggestions as things to

353 My own planned trip to Bangalore later that year provoked active interest from many developers, as did my stories from India when I returned to Brazil later.

be *done*. “You have to learn to talk about things but not work on them,” he explained. He did see the need for action, but the action was to inevitably fall behind the big dreams. Comparing them too closely would lead to nothing but frustration. We could talk about what Kepler could become one day. Meanwhile, we had to take the small steps: carrying out discussions on the mailing list, appreciative of the few short comments we occasionally got in response.

The Windows Build

As we celebrated the day’s success on Wednesday, Rodrigo was in a positive mood, focusing on what Kepler had achieved rather than on the project’s troubles. This contrasted sharply with many of the days from the previous month, during which the themes of “being wrong,” being “alone” and just “giving up” had come up again and again in our conversations. (The difference, I suspected, had much to do with the fact that the two of us were now actively working together.) Two weeks earlier Rodrigo had shown me a checklist page he had created on a wiki. The problem, he said, was that nobody seemed to want to follow it. Tiago and others did not even think Kepler should be making releases. Perhaps they were right and he was wrong. He had fought with the Lua community for a very long time, trying to make Lua into something that the majority of the users seemed to have little interest in. Perhaps he was just wrong. He was all by himself—even those who were working on Kepler with him were not in agreement. (He was not *planning* to give up, Rodrigo explained on a number of occasions, but the possibility was always there on the table. After all, while Nas Nuvens perhaps could not give up on Kepler, *he* could do just that, perhaps joining Renato as a manager of Java developers.)

One particular source of contention was the installation process. Like other software systems with a part written in C, Kepler’s code had to be compiled or *built* before it could be used. This process could in theory be left to the users, and the Lua community had historically stressed this approach, since most users of Lua were themselves skilled developers working on software products written in C and were assumed to be capable of handling the compilation step. It meant, however, that compared to other languages used for web development, Lua was quite difficult to install.³⁵⁴ This created a problem for Nas Nuvens, since most customers did not have the expertise to compile software on their own. Simplifying the installation procedure was therefore one of Rodrigo’s main projects in 2007.

The problem could be broken in two: one approach for users of Unix-based systems such as Linux and another for users of Windows. The first problem was easier: users of Unix could be expected to have the software tools needed for software compilation and to be accustomed to compiling source code into executable software on their own machines, at least as long as the

354 While Rodrigo’s work described in this section helped this problem somewhat, installing Lua and Kepler continued to be much harder until 2008 (about a year after the events described in this chapter), when a different group made an effort to make Lua easy to install on Windows. The existence of this project could be attributed in part to Rodrigo’s “culture farming” efforts (for example, the project was hosted on Rodrigo’s LuaForge website) and drew heavily on Kepler’s modules, though it arose without direct participation of the Kepler developers.

process was automated and did not require too much tweaking. Rodrigo also had access to Unix enthusiasts among PUC students. In 2006, he used FINEP money to hire “Alan,” a PUC Masters student. By April 2007, Alan had written a build script—a program that could be used to compile and install Kepler. With the new script, a user could download, build and install Kepler in less than a minute, making the process of installing Kepler on Unix computers quite trivial. As almost all Kepler contributors used Linux or other variations of Unix for their own server needs, the general opinion seemed to be that Kepler 1.1 was done. As a Linux user myself, I was of the same opinion.

Rodrigo, however, wanted to have a version of Kepler that could be used on Windows—partly due to the fact that he believed this would open a wider “market,” but mainly because nearly all of Nas Nuvens’ costumers ran Windows. (Rodrigo and many others sometimes attributed this to their being in Brazil, not yet on the Linux bandwagon.) Making Kepler work on Windows was a challenging task, however, both because of the quirks in the Windows build system and because of the Windows users’ higher expectations for the ease of the installation process.

A further challenge lay in finding people to do the work. The developers to whom Rodrigo had access appeared to be divided into those who did not have the skills for the task (for example the developers employed by Nas Nuvens) and those who had the skills but lacked interest. Tiago, in particular, had the knowledge necessary to solve this problem (and had done this for the earlier release), but was more interested in other aspects of the project. Following the new policy of relying on developers’ intrinsic motivation rather than simply just hiring them to do tasks, Rodrigo decided to not press, hoping instead find volunteers among the list subscribers. A few people expressed interest in helping (a big improvement from February, noted Rodrigo), but none were willing to lead the task. By early May Rodrigo had to face the fact that, if there were a Windows build, it would have to be done by *him*.

Rodrigo’s attempt to do the Windows build, however, made it clear that despite good high-level understanding of the technology behind Kepler, he was unable to make sense of its internals and occasionally had to rely on *me* to help him out. Being “out of shape” in this way concerned him greatly. While working on an example for our chapter, for example, I at one point turned to Rodrigo to ask for a suggestion on how to do in Lua a particular thing which I expected to be simple. A quick discussion showed this problem to be non-trivial, though perhaps doable. Rather than looking for a solution, which could take some time, I told Rodrigo I would avoid the problem by taking a different approach. Some time later, however, I saw that Rodrigo was still thinking about the problem. I asked him why he was wasting his time. “This problem shows,” Rodrigo said, “how out of practice I am.” He wanted, he then explained, to use opportunities like this to get his programming skills back.

Frustrating as it was, the Windows build proved a blessing in disguise. As Rodrigo was starting to understand, getting his programming skills back was not just a matter of solving the specific practical task of completing the Windows build. It was also a matter of adjusting to the change in the rules of the game brought about by “opening” Kepler, following through on the course he had already chosen.

As many people pointed out to me, in Brazil, where less educated technical workers are

abundant and cheap and where many needs of the domestic market do not require high skill, highly educated people like Rodrigo get quickly promoted out of programming jobs into management. Those who are thus promoted often lose their hands-on programming skill quickly, in part due to the fast pace of change in the software technology. They often lament the loss, but stay in management. Rodrigo's friend Renato, working as a manager in a local software company, often talked about his desire to get back to programming, carrying a copy of *Programming in Lua* on in the back seat of his car. (He had also at one point volunteered to translate the book into Portuguese, hoping that this would help him get back into the actual practice of software development.) Renato's management job, however, kept him quite busy and left him exhausted at the end of the day. Renato still wanted to think of himself as a "computer scientist," but appeared to be "stuck" in management work indefinitely.

Rodrigo, whose first job after leaving the university similarly involved managing software developers rather than writing code, had stayed more closely involved with technology, but had similarly become more skilled at bringing together people, ideas and resources than at getting code to compile. Rodrigo's desire to develop (or, rather, lead the development of) something more than a customized system for a local client, required software developers with higher level of skill than the financial resources of the project could afford. In a recorded interview around the same time, Rodrigo explained:

Rodrigo: I wanted to build a web platform for Lua. But I didn't have the technical capacity for this. So, I had to think like this: "How could I structure things so as to bring together a bunch of people and this way build a web platform?" The first attempt was Nas Nuvens. When Nas Nuvens appeared, I said: "Look, this is a place where I can make a web platform—because they want to use Lua." That's good, but it didn't work. [...] Because the level of the developers wasn't the level of the developers for a *platform*. Those are different levels. For developers of applications on top of a platform there are certain expectations. If you want to do a platform itself, you have to know a lot more about the operating system, a lot more of C, and about the integration with other things to do the connectors. This is a lot of work.³⁵⁵

If Nas Nuvens had been successful in attracting the additional money, the problem would perhaps been solved: Rodrigo would have the money to hire the right developers and could focus on directing the work. Without money, however, he faced a daunting task. "As far as C programmers go, I know a few," said Rodrigo, "As far as C programmers who want to work for free on a Lua platform—I don't know any."

Rodrigo thus decided that he had to attract foreign open source developers. He discovered, however, to get their help he needed an entirely different currency, one he had in short supply: respect earned through demonstrated technical competence.

Rodrigo: I noticed that on a free [software] project, there are two currencies of exchange: credit and respect. And there is a tangible good, which is the *source*

355 See appendix C, "Original Interview Quotes," (Rodrigo, May 2007, "Querida fazer uma plataforma web").

[i.e., source code, says in English]. So, you have two abstract currencies, and one concrete. I can provide *source* to the community and with this earn respect. Or someone can provide me *source* and I could pay with credit. [...] If I give someone credit, they get respect. And I am trying now is to earn respect.³⁵⁶

His skills as a coordinator who had managed to bring together people and money to lay the foundation of web development in Lua did not easily convert into respect. While Rodrigo's quote above suggests the possibility of a simple conversion of code into respect into more code, Rodrigo understood that the free software community assigned respect primarily to the people who wrote the code, not to those who organized and funded its production.

Rodrigo thus found himself in a paradoxical situation:

Rodrigo: I was in a very delicate situation, because I had a platform that I had come up with—the idea of the platform was mine—but that I didn't know how to use, because I didn't know all the...

Yuri: Didn't know how to develop?

Rodrigo: Not even how to use! I didn't know the internal details. "How does LuaExpat work?" I don't know! I never stopped to look. I said: "I need a LuaExpat!" And it appeared. Ah, good, now I have this piece. "Now I need a LuaZip!" It appeared. "Now I need MD5." It appeared. So, now I have all the squares. Good.

Rodrigo was familiar with the theory of open source and the idea that such projects are supposed to proceed differently, originating in an individual programmer's desire to "scratch an itch"—solve a specific problem through actual programming work.³⁵⁷ They are not supposed to proceed top down from, as a government-funded pursuit of a grand technological vision. (Some projects do start this way in Rio—many if not most of my interviewees have pursued one at some point. Such projects rarely go very far, however, for lack of time and inability to find others willing to participate.) Rodrigo's project, as he himself saw it, was proceeding "backwards."

Running the project by the local rules did not require Rodrigo to understand the details of the platform that was being built under his leadership. Local participants had accepted Rodrigo's role as the provider of an engaging vision and the resources they themselves could not obtain. Such resources, however, had no power in an international free software community to be supported by volunteer efforts. Rodrigo had to account the new currency:

Rodrigo: Because this is something I already noticed on the Lua list. [...] If I ask a question, my question shows a great deal of ignorance—because I am

356 See appendix C, "Original Interview Quotes," (Rodrigo, May 2007, "Duas moedas").

357 This idea was codified by Raymond (1999), who wrote: "Every good work of software starts by scratching a developer's personal itch." (Rodrigo had read Raymond's "The Cathedral and the Bazaar" when it was still an online article, and then again when it came out as a book, Raymond 1999.)

ignorant. Because there is a huge quantity of things I don't know. [...] And this shows in the question. And people's reply tries to be polite, but is... kind of *patronizing* [says in English]. Kind of like: "Look, eh... Try this here. Go see how this is done." And when I send a more technical response, I feel that this produces a connection with the person on the other side. They say: "Oh, I am not talking to a stupid manager. I am talking to someone who will at least understand what I am thinking." So, you feel in the reply a different vibe.³⁵⁸

Rodrigo thus had to learn to engage in his own project in a new way: as a developer capable of discussing the minute details of the code with the members of the list and making changes to this code when necessary. The Windows build of Kepler was forcing him to do this work. In our interview a few months later, Rodrigo described the time as a difficult but important experience. The difficulty, he explained to me, corresponded to a "deconstruction of a mental model" of how the project worked and of what his place was in it, a time when he had already realized that his old approach was not working yet was not ready to accept the need to make a transition.

Ups and Downs

I left Rio in early August, but continued to follow the project remotely, partly out of researcher's desire to know what happened later, but also as a simple matter of commitment. In April 2007 I had volunteered to write a wiki based on Kepler. The wiki became the first public application built on top of Kepler, and quickly came to be seen as a demo of Kepler's capabilities and a proof that Kepler could actually be used to build real web applications.³⁵⁹ I now had a role in the project that could not be easily dropped.

By September, I had started to notice a substantial growth in list traffic, finding myself barely capable to keep up with it. I called Rodrigo to catch up on what had happened since my departure. Rodrigo started the conversation by communicating his excitement over the recent changes. The opening of Kepler was finally bringing fruits. The activity on the list was an important part of that. There was a clear spike in traffic. The increased list activities were matched by the growth in interest among local companies. In particular, João was talking to local firms and FINEP about the possibility of bringing Kepler into the digital TV projects. (See previous chapter for the discussion of Lua's role in digital TV.) This was made possible, Rodrigo was sure, in part because Kepler was finally easy to use on Windows.

Rodrigo offered an example of collaboration that illustrated the list's new dynamic. Two weeks before our interview, while working on one of Nas Nuvens' projects (which were starting

358 See appendix C, "Original Interview Quotes," (Rodrigo, May 2007, "Meio patronizing").

359 My wiki application ("Sputnik") was not the first web application built on top of Kepler—the administrative application maintained at PUC by Márcio had been using Kepler since 2005 and many list members reported using Kepler for their own applications. However, Sputnik was the first application that a member of the general public could see live and obtain the code for.

to use Kepler), Pedro discovered a particular problem specific to Windows. He sent a message to the Kepler list, which went unanswered. After thirty to forty hours of work, by Rodrigo's estimates, Pedro managed to identify two potential causes of the problem. (This work included talking to Tiago off the list.) He sent a new message to the list, now much more specific, and two foreign members of the list joined in the discussion—one of them from Uruguay and another from Southern California. In the course of a lengthy discussion that involved Rodrigo, Pedro, Alan and the two foreign members, the California engineer proposed using a library developed by Luiz Henrique, one of the authors of Lua. The library did not work with Windows, which led to an additional conversation, off the list, with Luiz Henrique. In the end, the California engineer adapted the library for Windows. Rodrigo and Tiago then jointly made the necessary changes to Kepler. This highly collaborative process, which brought together a Nas Nuvens employee, FINEP-funded Kepler contributors, a member of the Lua team and two external participants served as an example of the project's new dynamic, helped Nas Nuvens solve a specific problem it faced, and resolved a serious underlying problem in Kepler, improving the quality of the platform on Windows.

External participation was also affecting Kepler in more fundamental ways. In April, Rodrigo talked about two new ideas: a radical re-design of the installation method that would make it easier for the user to install Lua modules and a re-design of Kepler's method for connecting to the web server. As the first major changes to Kepler after the decision to open it, Rodrigo sought to discuss the ideas at length on the mailing list, and both ideas have in fact led to lengthy discussions, with active participation of a few list members outside Rio. While this early success had gotten obscured by the troubles with the Windows release, both proposals started attracting interest over the following months. Rodrigo was starting to receive more and more comments and code submissions. As both subprojects were nearing completion, Rodrigo found an opportunity to bring back some of the members of the Lua list who had interest in the project in 2005 but appeared to have lost it later.

The project was also helped by arrival of Jason, a Rio developer whom we met in chapter 2.1. In Spring of 2007 I noticed Jason's name on the Lua mailing list, as one of a small number that sounded Brazilian yet had not come up in my map of the local Lua community built around PUC. I emailed Jason, discovered that he was working for a local company in Rio. I decided to use the opportunity to talk to him in person. (I contacted two other people in a similar manner. One of them turned out to be a Brazilian graduate student in New Zealand, while another one was a software developer working in São Paulo. In both cases, we exchanged some messages but did not do an oral interview.)

Jason was working for a company that was using Lua to develop applications to run on old, underpowered computers. Jason originally discovered Lua several years earlier, while working on a computer game as a hobby project together with a friend, discovering at one point that Lua was developed in Rio de Janeiro. The project was abandoned, however, when Jason's friend left Brazil. Several years later, while looking for a programming language to use for his company's applications, Jason ran into the developers of SuperWaba, a Java-like development platform maintained by a software developer on Rio. SuperWaba reminded Jason of Lua, which

became his platform of choice.³⁶⁰ After introducing Lua at work Jason became a fan of the language. Having used some of the modules maintained by Kepler, Jason also talked with much interest about this project and his own desire to participate in something like this. Noticing that Jason seemed to have the exact combination of expertise and interests that Rodrigo was desperately seeking, I asked him if he thought practically about participating in Kepler and suggested that he should meet with Rodrigo.

Despite his excitement about the fact that Lua was developed locally, Jason had never approached any members of the local Lua community in person, considering himself too far removed from that community socially:

Jason: Look, I think it would even be interesting to see that they are real people, right? They are people whom I really came to admire, seeing the amazing job those guys did. But I... It's like this story with you meeting a great musician, who plays amazing music. You go and meet him personally. Damn... [Laughs]. He can play for you, that's cool, you know. But man, to go there and talk to him is different. To exchange ideas about music, etc... You have to understand music as well as he does to be able to go and take advantage of this.³⁶¹

Jason did hope to meet the members of the Lua community eventually, but expected to only get there slowly, step by step.

Jason: I'll go there [to take PUC courses] and will be there, transfixed, wanting to see what they are doing and how it works. Maybe something I could help with. So, when I discovered that the guys were at PUC, that I started using something and the guys were from PUC, I was like: "Damn. This is there. I think I'll go there and see what it's like." But I wouldn't even know were to meet those guys. I think if there was a course they were offering, I would go immediately, running. But I wouldn't go and knock on the door to learn where the guy is.³⁶²

360 Jason's discussion of his choice to use Lua shows a complex mix of pragmatism and interest in local technology:

Jason: Those Waba guys [actually, SuperWaba]... Waba is famous. I discovered that they are used there abroad, right? [...] So, I took Waba, looked and said: "Wow, that's cool." And having gotten something done by a guy in Rio de Janeiro, the bell went off in my head. I was like: "Damn, there are the people at PUC!" I have already heard of Lua before, right. I was researching something and found that there was a game that used the Lua language and discovered that Lua was done in Rio de Janeiro, was done by PUC-Rio. And I was like: "Man, why would I use something super-sophisticated like Java, when there is this thing, which is Lua, which is made for the core of game mechanisms and must be very efficient. Java is much more versatile, sophisticated. So it's slower, it has a lot of requirements. So, man: Lua is very lean, must be perfect for my needs. And it even is a national product, is something... It's like this: it's not enough... It's not just because it's a national product. It's a good product. It's something good, done in Brazil. So, look, for this reason I have to at least look at it and understand it better.

See appendix C, "Original Interview Quotes" (Jason, June 2007, "Negocio muito bom, feito no Brasil").

361 See appendix C, "Original Interview Quotes" (Jason, June 2007, "Ver que eles são gente").

362 See appendix C, "Original Interview Quotes" (Jason, June 2007, "Vou ficar amarradão").

After returning from our interview, I sent Jason an email encouraging him to talk to Rodrigo. A day later they met for lunch. A few weeks later Jason was actively involved in Kepler. For Jason, meeting Rodrigo was a transforming experience. Rodrigo showed him, he said, that one could live in Brazil doing something outside Java and .Net:

Jason: The fact that I am now working with Lua, and seeing this potential, it makes me want to find a way a little into this intersection that I find amazingly cool. Now, there is little motivation, if you look at this from the economic point of view, right? So... I mean, we don't just live by the economy, but speaking like modern thinkers, it's a factor that cannot be ignored. So, I have a lot of interest in becoming a researcher. I think if there is a calling that I haven't yet pursued, it's this. And I think participating in this project might count, in some way, for this. If only for networking, to meet the people and to be contributing, entering a community that I always wanted to enter, and now a door opened. [...]³⁶³

Jason: With the PUC people, I mean. Like how I met Rodrigo, for example, who is a person who showed me that it's possible to work like this. Because this is a question that has always been with me, you know. Me thinking: "Well, great, I've always wanted to do this, but how would I make a living? If I were to be doing those highly experimental and advanced things, which do not have... are not commercial in a mainstream way? How would I live outside Java and .Net?"³⁶⁴

Rodrigo was a living proof that this was possible: "Well, he is there and living, right?"³⁶⁵

Jason's entry into the project, however, also proved a life-saver for Rodrigo. While a number of new sub-projects were generating a lot of interest, several others were stagnating, in part due to the fact that Alta's Fabio and Rogerio were increasingly busy with Alta's expanding projects and had less and less time for Kepler (see chapter 3.1). Rodrigo's reliance on FINEP funding, however, meant that projects could not be simply dropped for lack of interest. Jason's adoption of one of such lagging project helped Rodrigo focus on the other ones, the ones he felt were most promising. Another month later, Jason left his job and came to Nas Nuvens, taking over Rodrigo's *de facto* role as the company's technical director. This further freed Rodrigo to focus on Kepler: he was now able to spend as much as 80% of his time working "as a developer." Another month later Rodrigo managed to dedicate some of this time to do a side project, for which he wrote half of the code.

The new dynamic continued for several months. After the end of the calendar year, however, the project found itself without money again. A grant had been awarded for the new year, but no money had arrived, having disappeared somewhere in the complicated network of funding transactions. (The work, of course, was expected to proceed on schedule.) Over the next

363 See appendix C, "Original Interview Quotes" (Jason, July 2007, "Me tornar pesquisador").

364 See appendix C, "Original Interview Quotes" (Jason, July 2007, "Está lá e vive").

365 See appendix C, "Original Interview Quotes" (Jason, July 2007, "Está lá e vive").

many months Rodrigo had to suspend his newly acquired career as a developer, dedicating much of his time to the work that he thought he was starting to put aside: finding out what had happened to the money the project was owed and what would need to be done to get it back. The team's morale was also seriously hurt, as some of the contributors had to go for months without getting paid for their work. The team managed to release the next version of Kepler in June 2008, but after that the project was effectively suspended. When I returned to Brazil in December of 2008, Rodrigo had managed to finally get access to the money awarded a year earlier and was starting to bring the project back to life, but a lot of momentum had been lost and a lot of work had to be re-done.

Expanding the Practice

After reading a partial draft of this dissertation, Rodrigo Miranda commented that my exposition had shown to him that his project had been futile, what he had tried to do over the last ten years was simply too hard to do, at least in Brazil. Indeed, this chapter attempted to show the many difficulties involved in such a project, arising from the seeming improbability of the alliances that must be made and maintained for such a project to succeed. One may wonder if I have led Rodrigo into diabolical experiment, encouraging him through my display of interest in his project, only to use it in the end as demonstration of why a project like this cannot succeed.³⁶⁶ Difficult, however, is not the same as impossible. While the structural difficulties faced by a project like Kepler may seem daunting, I believe it is important to look at the project historically, as incremental efforts to adapt the local context for the increasingly more central forms of the global software practice, looking at the local practice that the project has enabled, rather than just its outcome. Jason's quote presented above is key for this view of Kepler. "I met Rodrigo," Jason said, "who is a person who showed me that it's possible to work like this. [...] He is there and living, right?"

By trying to make Lua into a practical option for web development, Kepler had sought to change Lua (understood in the broader sense, as a techno-social system³⁶⁷) from a highly specialized niche programming language, strong in very specific areas of applications but

366 In a different conversation Rodrigo suggested that our early conversation in 2005 had influenced him by convincing him that what the project was doable, that he was "just crazy, not maniac." "Because you came from the Valley," he wrote, "and you looked at Kepler as it was viable, even if remotely. Nobody had looked at it this way before. Maybe not even me." When I pointed out that my willingness to listen could hardly be interpreted as a sign that I truly believed in his project, Rodrigo explained out that such a belief was not expected from me. "You just took it seriously enough," he said.

367 In other words, Kepler did not bring many changes to Lua *itself*—that it is, to Lua understood in the narrow sense of the programming language specification and implementation. (It did help bring one important change, though—Lua's packaging system, which made it easier to write additional modules for Lua. The new system came from a combination of efforts, which included, in particular, Rodrigo's lobbying for its need and Márcio's early implementation.) Kepler's efforts, however, have made substantial changes to Lua understood in a broader sense, as a techno-social system of elements that enable different uses of Lua, comprising the language itself, a collection of modules, a community, the documentation, etc.

assumed impractical for everything else, into a mainstream language that could in principle be used for anything. While early specialization was undoubtedly key to Lua's success, it also has put a limit on the cultural significance of Lua, suggesting too simple of a story: a software project done in Brazil can succeed, but only through a combination of extreme specialization and luck. Kepler has attempted to broaden Lua's success by turning it into a general purpose platform. While Lua's future in web development is still uncertain, the larger effort to widen Lua's use by building a community of people willing to share reusable code and gradually undoing working to undo the network effects that make Lua successful, was perhaps starting to succeed.