

Ryan: Okay, so why don't you state your name and just generally tell us what you do here at Mozilla.

Parmenter: My name is Stuart Parmenter. I work on our graphics, back end of the structure pieces here at Mozilla.

Ryan: And how did you come to work at Mozilla?

Parmenter: I guess it starts a long time ago. I guess back in '98, started working on the Mozilla project back in high school in Georgia and in '99 I got hired by Netscape. And did that for about four years and then went off to another company called Open Source Applications Foundation. Which is Mitch Kapor's sort of personal information manager client thing.

Did that for a year. Saw that wasn't the right thing for me. Then I went with a group of guys who are now here at Mozilla to Oracle. We were doing some Mozilla calendaring work. So that sort of brought me back in, after a year away and we decided that wasn't the right place either.

So then we sort of came back and during that period, I guess we had been working out of the Mozilla offices and, you know, because we were really working that project, it wasn't really—we were paid by Oracle, but it wasn't really doing much related to the company itself. Aside from trying to help them get their stuff out. And then started at Mozilla here in July of last year, I guess.

Ryan: And when did you begin working with computers?

Parmenter: Let's see. In fourth grade, fifth grade. I don't remember when that was.

Ryan: Do you remember the first programming project you worked on?

Parmenter: I did, like, some random, like, little basic things. Like, I remember trying to do something for, like, to help do my spelling homework or something. To try and, you know, help me remember stuff or ask questions and sort of how to get them right and randomize that sort of thing. Or math problems. I think I did some little things like that, a long time ago.

But what happened was—so in, I guess—I guess right when I—I don't remember how—I guess fourteen years, I think. I had gone to work for my dad who does insurance agent—he's an insurance agent along with his brother. And I did just some random data entry stuff for them, which was pretty boring. So that didn't last very long.

So I guess in '95, I don't know, it was something like that. I got a job at an Internet search provider that had just started and they were looking for just

sort of—there were, like, two guys there and they didn't really know what they were doing. So they were looking for somebody to sort of help out.

So I started doing some web page stuff originally, some support on the phone with customers and stuff. And that slowly grew into sort of—started doing some, you know, get our servers running and mail maintenance and sort of turned that into going more to the, like, network administration side of things.

We also started to do a system where we were outsource—sort of outsourcing—we were allowing other sort of Internet service providers to outsource their stuff to us. So we would do their dial-up and system email and web hosting and everything. And so for that, I ended up building a pretty comprehensive website to allow them to manage all of that.

So I did—that was some programming there, I guess. Sort of also in my free time, I guess. I don't know when I had free time. But I started working on some of the—some open source projects, probably in '96 or so. I got involved with the GTK and GNOME projects really early on. Did some work with them. Fixed just random little things here and there. And I guess through that, I met a lot of people in sort of the open source community and started to get to know people.

And then in—at some point later, I decided that there was sort of a need for a mail client on Linux. So I wrote this program called Balsa, which was a GUI mail client at a time that really the only other available GUI mail client on Linux was Netscape Communicator Suite stuff and nobody really liked that because it was big and slow and ugly.

So I did that and that went well for a little while. Continued doing random GNOME pieces. And I guess in October of '98, I went to the [inaudible-audio skip] showcase where I met Mike Shaver, who was working at Netscape at the time and I guess I maybe met him through email or something before. But we got to talking and this was—I guess this was after they had announced—Netscape had announced they were releasing the source to Mozilla in March of '98.

And so I think the conference was in October of '98. And we met up. There was sort of—this is right at the point where they were deciding they were sort of going to change the whole layout engine that was in the product. It was sort of a big turning point for the, sort of the open source Mozilla. Like, the very sort of first big, I guess, turning point.

Because when they originally released the code, it was all—it was basically Netscape 4.5 plus. So it was sort of really targeted to be Netscape 5.0, which never actually got released. But in October, so they

decided that—these guys that were working for Netscape down in San Diego, had come up with this great new technology, this new layout engine that fit on a floppy and it was fast and standards compliant and great and wonderful. I think there were a lot of arguments at Netscape at the time about whether or not that was the right decision. Those continued for as long as I was there. But for whatever reason, they chose that. And so at that point, they decided that on Linux and Unix, they didn't really want to use Motif anymore. It was sort of the toolkit. Because that wasn't really open source. It was sort of ugly. It was not really staying with the times.

So they were looking for some people who knew GTK because that's sort of the toolkit they decided they wanted to use. And so I said, "Sure, you know, I know a little bit of GTK. Done some mail client stuff. I—you know, worked on GNOME." You know, thinking, sure, I'd just sort of help out. And what turned into me thinking that, sort of turned into sort of a, "Well, let's see if we can get it up and running."

And so I spent a lot of time working on that throughout probably the first, I don't know, pretty much immediately right off. So I think probably within, like, two weeks, I had something up and running. It was sort of—Mike Shaver was doing a little bit of stuff and another guy at Red Hat, Owen Taylor and Matt Wilson, who was also at Red Hat, were helping out a bit as well.

And so we did a lot of the early, sort of foundation pieces. Got things up on running on Linux and so that was a pretty big, you know, accomplishment. I think at that point, there really weren't very many sort of other contributors to the project. It was sort of, you know, it was big—when they released it, nobody really knew what to do with it, because it didn't build, it wasn't usable, it, like, it didn't really have anything.

So this was sort of the first point where they had kind of gotten past some of those pieces. They had it building and they had figured out, you know, some of how to make it work. But it was still really early on and so we got it up and running and that was a pretty big accomplishment, I guess and I don't think any of the Netscape guys really knew what to do. They were, like, "Wow, we have contributors now."

You know, so this was sort of—it was an interesting period where, you know, they would call me into meetings and stuff like that at Netscape, I'm sitting there at home and I'm on the east coast, they're on the west coast, you know, I'm getting, like, this seven pm call and my mom's, like, "Why aren't you doing your homework?"

You know, so that was always pretty good fun. And so we got a lot of stuff done and I mean, it was sort of—I've looked back a couple of times since to sort of see what, you know, actual work I did and committed to the tree. And some of the stupid things I did, some of the dumb bug comments I've made.

It's always kind of amusing to see and sort of look and see, like, how much I actually did and sort of try and figure out, you know, if I knew what I know now, how long would it have taken me or had the time? Things like that. But—so it's very interesting.

So in—when was it? February maybe, of '99, got this call I think, I think it was a call, from this producer basically from—he was doing a PBS documentary on Netscape, doing this documentary called Code Rush and he—he was, I think, Dave Winton and he was, like, "Yeah, we're doing this documentary on Netscape. We've been following them for the last, like, two or three years, two years, maybe a year, I don't know. And, you know, we sort of—we followed them sort of before they open sourced everything and that's a pretty interesting plotline. So we'll follow that."

And they followed that pretty well, and they were, like, "So these guys at Netscape are telling us that you're doing lots of cool stuff." So they were, like, "We'd like to come out and film you for a while and do an interview and stuff." And I'm, like, "Mom, these guys want to come out." She's, like, "What are you talking about? You're crazy."

So I had her call—or had them call her or something. And she was, like, "Huh, interesting. So that's what you've been doing all this time." So they came out and followed me around the school and just sort of some funny interview stuff and followed me to work. It was pretty amusing. Sort of, I don't know, I was kind of—I got tired of them by, like, the second hour or so. It was, like, "Can you guys stop following me?"

So that was good and then in, I guess June of '99 or—yeah, June of '99, I got hired by Netscape. So I flew out here and I guess I haven't really looked back. So I started at Netscape. I was basically doing the same thing I had been doing before, continuing with that work on that code. Trying to speed it up, trying to—in Code Rush, one of my infamous statements I guess is, I made this claim that I was going to make Linux as fast as Windows and nobody lets me forget about it because it still hasn't happened.

And, you know, but—so yeah, so I came out, I did a lot of work on that. Did a lot of other things at Netscape. I worked on some of the mail codes, some of the security code. When we were first trying to get the security code out into, all these sort of SSL pieces and everything. The

government, at that point, was sort of, like, "Yeah, we can't really let you release that, because, you know, it would give other countries and stuff problems and—"

So I was, you know, trying to push to help get that stuff done and help blend it and get it all—get those pieces integrated into the tree. Because at that point, it was basically, Mozilla couldn't do SSL. They were sort of this add on, it was a binary thing, but none of it was open sourced. And so we wanted to, obviously, integrate that in.

So I did some work on that. What else did I do? Oh, yeah, at some point, I guess, 2000 or something, I got sort of—or volunteered maybe to do—to re-write our image library. So basically everything from loading images from the network, to caching them, to displaying them, to de-coding them and sort of that whole work—that whole flow of data. From through the network to the screen, I guess.

So I did that for probably six, seven months and that was a pretty—I worked way too hard on that. I kind of burned myself out at the end. Just too many hours, too many days straight. So I kind of went into a lull briefly after that. Which, looking for something else to do that was kind of interesting. But I sort of finished that stuff up and so that was all good.

Then there was sort of a period of figuring out what was next. So I put together a proposal for doing sort of a browser on Play Station 2, trying to figure out if that market made sense. And so went and did a lot of sort of development work, sort of trying to get a prototype up and running, something like that.

Got a lot of pieces together. Got a lot of cool stuff working. But in the end, sort of the browser just didn't seem like—at least our browser, at that point, didn't seem like it was really the right fit. So we sort of chunked that project.

At that point, sort of Netscape and AOL were sort of getting kind of—well, sort of—I guess, AOL bought Netscape in '99, April of '99 I think. So sort of—they were sort of becoming sort of more—AOL was not sure what to do with these guys, basically.

So at some point we got re-assigned to a different group that was working with a group in Southern California, in Irvine I guess. Doing sort of another similar browser thing. And so—but right before that, I basically started another project where—we were trying to do mobile—sort of stay on the small browser idea, see what we can do.

So for a while, Doug Turner and some other people were sort of trying to just generally reduce the size of Mozilla. I said, "Well, let's try something completely different. Let's make Mozilla do server side rendering and we'll spit that down a client that can be really, really small."

So we worked on that. I got two other guys to also come and help work on that and we put together a pretty compelling prototype. We had, like, a seventeen kilobyte executable on Windows CE device and we had it running on Palm and Play Station 2 and some other stuff. And it was really, really small. It was really fast. There were some issues, sort of with server scalability. How do you, you know, handle a million concurrent users, that sort of thing.

Got some good answers for it. So I pitched that to a lot of AOL guys and they said, "We don't really know what to do with it." So, you know, we said, okay. And around that time, basically we—I got permission to open source all that stuff that I had been doing. So we kind of released that, sort of quietly. I mean, we weren't really sure what to do with it. It wasn't really clear how much, you know, what people were going to use it for and it was still sort of prototype stage. So it wasn't the easiest thing to do or build or anything like that.

And then I guess in July of 2003 or so, AOL laid off pretty much everyone who had ever been associated with the browser project. They kind of picked people from these random other places and they were, like, "You got to go." So went through that. It was good though, because things really were kind of falling apart at that point. It wasn't really a fun place to work anymore. I think a lot of people at that point were looking for other places to do different work and kind of saw it coming, I think.

You know, at that point, I was kind of looking for something else that would be the next big thing. Didn't really find it. But I found a couple of things that could have been fun. But, you know, so that was all right. I took a few months off, three or four months because I needed a break. And at that point, I ended up deciding to go to Open Source Applications Foundation up in San Francisco. After about a year, I decided the commute was too much, 'cause I lived down in Santa Clara, California and commuting to San Francisco every day was a pain.

So I didn't want to do that anymore and the job just really wasn't working out for me. So then, as I said earlier, we did the—went to Oracle after, like, I don't know, I think I took, like, two weeks off, three weeks off or something. Went to Oracle, did that. We got a lot of stuff done there and a lot of that stuff's still alive and kicking. Another guy that was on our team is still there and he's working on it. That community's growing up now and getting more people. And so I'm hoping, at some point, they'll,

you know, maybe invest a bit more into the calendaring side of things. I think there's just a lot of interest there and I think there's a lot of room for a product or a project or something there.

But we basically, myself and Mike Shaver and Dan Mosedale and Vladimir Vukicevic we basically ripped out the entire—so if you look back at sort of the original Mozilla calendaring project, it was sort of a—it was based on this company OE1, who had done this sort of set top box, I guess it was a set top box. It was really more like a just—I don't know, an Apple IIE or—like, a big just—Imax sort of, you know, hard drive in a thing with a screen.

And it had sort of your calendaring stuff in it. It had a web browser. It had all these little applications that you sort of could use. And they were all built on top of Mozilla. So it was really sort of the first big external project that really built everything on top of Mozilla.

So it was pretty cool and—but they—and so they did this calendar stuff. And so at some point, they kind of went in a different direction and sort of—this stuff was open sourced. And so the Sunbird project came out of that. Which was sort of an open sourced calendaring thing, built on Mozilla.

And we were trying to—we sort of looked at that and we said, a lot of this stuff is, you know, works, but it's kind of crufty. It doesn't really—it was written, like, in the early days of the Mozilla code base. So it was, like, didn't really follow most of the newer sort of ideas or rules or components or—just kind of out of date really.

So we chunked a lot of it. Took the ideas that were good and tried to, you know, keep those around. So we basically did a new back in for calendaring. So we made everything AC current as we added plugable providers. So you could do a local provider or a, you know, an ICS provider or [inaudible] provider or all these different things and we made it all very flexible and good.

And then we started working on the UI side of things. We were kind of working on this side project which was called "widening" which we were trying to integrate calendaring into Thunderbird, because it seems like everybody in the enterprise at least, uses Exchange and, you know, Outlook. So we figured there's something to this, you know. If we can get good integration, and I think that was really a key, is getting good integration, I think.

The stuff we got in wasn't really good. It was—you know, and it's still an evolving project. I mean, this was only, like, nine, ten months ago. So,

you know, we got some basic integration in. We started doing new, you know, calendar views and drag and drop and made it all look nice and work well.

So we did a lot of good stuff there and so that stuff's still going. So that's good. And then I guess sort of, back at Netscape, one of the pieces that sort of—off of the original stuff I did—I was very interested in sort of the graphic side of things. I did a lot of work early on on the GTK stuff. I did a little bit of Windows stuff here and there and as I did, some of the PlayStation 2 and small device stuff. I was sort of looking more at the graphic side, low level pieces.

I tried it a couple of times at Netscape, to sort of say, "Our current graphic stuff is bad. We need to fix it. We need to rev it. We need to do something good," and I would always get back sort of a, "Yeah, we agree but who's going to do it? Who's got the time?" And so finally, sort of there was an opportunity here at Mozilla, that where they said, you know, "We now see that graphics is a big part of our future. We need to invest there."

So, you know, I'm, like, "I'm here, I'm here! Let's—you know, let's do it." So myself and Vladimir, are here working on the graphics, writing our graphics back end, trying to provide all the sorts of new capabilities. We want to be able to do rotations and scaling and fancy text rotation and paths and all sorts of good stuff like that.

And so we've been working on that for, I guess, the last seven or eight months now and it's starting to come together. We've got—it's sort of our—on nightly builds on the trunk. Using our stuff now and Linux should be following in the next—any day now, basically. And Macs should be following shortly after that. So I guess that brings us to now.

Scheinfeldt: You mentioned that you work in groups and you mentioned that you kind of have traveled around with some of the same people. When you kind of—when you work in groups with people, how do you decide who's going to do what and how is sort of division of labor worked out?

Parmenter: Well, I think it depends a lot on the group and sort of how the project is. At Netscape, when I was working on this mobile thing, it was called Onya, I sort of—it was me originally. I sort of was doing it all. I had written up basically what we were—what the plan was and everything.

So when I ended up getting—I basically said, "I need help." And so the manager—our group was technically working on something else. But I was on that group, for whatever reason. So our manager managed to get two more guys and so at that point, I was kind of trying to help them



figure out what they, you know, where they could go, while still sort of trying to lead it and get development done.

And I found that to actually be really hard, because it was sort of like I was halfway trying to sort of split up the task and everything. While also spending, like, you know, as much time as I could getting things done, because that's really what I wanted to be doing.

So I think there, it was sort of like, "Here's the things we need to do, who wants to do what?" You know, I didn't really care. There were areas, certainly I knew how to do better and so I think, in a lot of cases, on that project at least, the other guys sort of, you know, ended up with some of the areas that, like, just hadn't been looked at yet. Tried to figure out, what do we do here?

But I think more recently—and so with the imagelive stuff, that was mostly just me. There was another guy who was doing some work on some the de-coders, sort of looking over my shoulder to make sure, sanity checks, basically. But a lot of that was me.

And then more recently, with the graphic stuff, I think Vlad and I—we seem to be—we work really well together, I think. And so I think things just sort of fall out, I think. Since we're working on Mac, Windows and Linux, Vlad does a lot more stuff on Linux. So at some point, I switched to Windows from Linux and everybody kind of gives me crap about that too.

So I've been doing a lot more on Windows now. So I basically took over a lot of the Windows problems. He took over a lot of the Linux problems. Mac, we just sort of said, eh, we'll get to it. And then some of the bigger pieces, we just sort of, you know, I took over fonts, he took over some of the rendering pieces. So we just sort of—we sort of, you know, you either arm wrestle for it or just, you know, kind of see where things fit naturally.

So he's done some of the Mac—certain pieces of the Mac stuff and I'm working on other pieces. It just seems to all sort of fall together. We sort of just—you know, pick and choose, see what makes sense.

Scheinfeldt: Is that pretty common? Where people have other people that they have worked with for a long time and they kind of stick together as a group, working on stuff? Or do people kind of—do groups come together and then break apart?

Parmenter: I think it's probably more common that groups come together and break apart. You know, I think with the Mozilla project, I think it's—I think

things are a little different than normal though. I think—it seems, at least, that a lot of times people get in their little area and they stay there sort of forever. They don't really break out and do something different.

So, I think you do see a lot of those groups sticking together for a long time. Those guys work together. You know, our layout guys, David Barron and Boris Zbarsky and—I'm forgetting some other people. Robert O'Callaghan and some of the guys, they've been working on that stuff for a long time. They sort of all know how each other work. They have their own sort of areas.

Generally, like, when people are working on stuff, the components are big enough that, you know, you—there's not often a lot of overlap in sort of what people's areas are. There can be and people, you know, depending on really what's needed, I guess. People also like to jump around a lot. People, you know, aren't necessarily always on that same thing. They're sort of, you know, if we keep the components big enough and that's what their interest lies in, they can sort of jump all around and do that.

Scheinfeldt: How does it—you know, how does—if somebody's going to move from working on one aspect of the code, to working on another aspect of the code, how do you—if there's already kind of an established group of people, how do you kind of break into that group?

Parmenter: Well, I think there's two different answers there. I think if you look at it from, sort of the pure Mozilla community's perspective, I think often somebody will basically just show up and they'll start submitting patches. They'll talk to people about their ideas. They sort of break in slowly into that group until, you know, eventually maybe they're a peer of that group.

And I think that's sort of a process that, it can take a while. But I don't know that it always does. I think it kind of depends on the activity level of that person. If they're really motivated and really into it, you know, I think they can sort of break in really quickly and, you know, they're putting up as much stuff as the other guys, right? You know, I mean, there's sort of the learning curve but, you know, I think if they can get there, then they break in really fast.

But I think sort of the other side of that is, sort of people who just kind of are, like, in this one area one day, and then like, all of a sudden, they're sort of in a different area the next. And I don't think we've ever figured out really how to do that well. I know, you know, sort of when I went from doing a lot of the GTK widget and graphics pieces to sort of doing image library, like, I just didn't have time for, you know, the other stuff anymore.

And so some of that sort of fell, I think, you know, fell into a hole. Some of it was picked up by some of the other guys who were in that area. But, you know, I think, it never quite picks up. I'm sure there's half, you know, probably a hundred bugs still assigned to me that are from back then that I've just never really had the time to look at.

And they get fixed. You know, people find them and they're, like, "Hey, what about this?" And, you know, I'll try and look and help out. But figuring out sort of that whole, how those pieces all fit together—because a lot of this project is about these guys showing up and they're, like, "We're interested in this thing." You know, it's hard—especially as a community to say, "You go do this," or, "You go do that."

You know, if you're sort of—if it's more corporate structured or somebody's paying you, then sure, you'll see that happen more often where they're, like, you know, "We're writing your checks, you go work on this." So I don't know if that really answers your question but—

Scheinfeldt: It does. How do you communicate with—you know, like, what do you use to communicate with people? Both remotely and if you're working locally with them.

Parmenter: Right. Well me personally, I've always found IRC to be pretty convenient. Sort of, you have that instantaneous sort of chatter, right? You kind of go back and forth with things. Whereas sort of with the emails, you kind of need to get your thoughts straight first, you know, so you make sense when you send it and people read it. You know, because it's not—you don't have sort of instant feedback.

So I've always sort of preferred that. You know, certainly in the office, I sit next to Vlad where you throw things at each other or yell across, you know, at each other or whatever. Whatever works. Email works for a lot of things. Newsgroups are fine for some things.

I think we, as a community, try and sort of embrace everything, which I'm not sure is really the best approach. But there's been a recent push to go back to newsgroups, after four or five years of sort of not using them. Some of us are reluctant to do that. We think there's sort of better technologies out there at this point, than stuff that was invented in, like, the '70s.

Ryan: Why do you think there's the push to go back to newsgroups? What would be the reasons and what's the reasons for the resistance?

Parmenter: Well, I think there's just different groups. I think there's just people who are used to using newsgroups. I think there's a set of very vocal people

who have sort of continued to use the newsgroups, for very sort of specific pieces. And I think that that group's very—was very vocal that, "Hey guys, no, we want to continue to use the newsgroups because that's, you know, what we're used to." And I think there was recently just this big re-org where there were a couple of people who were very interested in sort of getting all the newsgroups back up and running, getting new groups and everything organized well.

And pretty much any time they would send out mail, "Is anybody having problems with this?" Or—nobody—just nobody replied, nobody cared. Everyone was used to doing things some other way, at this point and nobody—and then they sort of appeared and then they were, like, you know, trying to—then they were sort of poking at people. People have done a good job of trying to get sort of mailing lists and newsgroup posting things working, so that, you know, if you don't really want to use the newsgroups, you could do mailing lists or something like that. But there's inherent problems with sort of that, that people are hitting.

Scheinfeldt: When you're using multiple modes of communication, do things sometimes get lost? Like, is it difficult to--?

Parmenter: Yeah, for sure. I mean, especially as we've recently been changing those modes of communication more. It's sort of, you know, nobody's really sure where to necessarily post things. Just, like, the other day, we were trying to figure out how to announce that we were going to turn our stuff on, on Linux and we got, like, five different answers. We were, like—some guys were like, "Post in the newsgroups." Some guys were "Post on the website, over here," you know, "Announce it here." And it's like—and then when we did that, we ended up getting responses from the newsgroups, responses from the mailing list and none of them were kind of cross linked.

So you end up with, like, these separate threads, where people are just missing the other responses. So I think it's something we got to figure out and do a better job of. I think it's a—especially with the newsgroups, I think it's a recent enough problem that we're trying really hard right now to figure out how to best use it and to deal with that. But yeah, I think things definitely get lost.

You know, and IRC's the same way, right? People—it's not logged as well as it could be or indexed as well as it could be. Discussions are—you know, it's hard to necessarily find discussions inside the chatter too, because people are always talking about different things and you kind of lose those key things. But at the same time, you know, it's, like, you've made this decision now. So you know, we could sort of formalize a process of saying, okay, "A decision's been reached, somebody should

email that out,” and maybe that's a good idea. But at the same time, it's sort of—it's something else, you know, that's time consuming. Developers don't really want to do it, that sort of thing.

So I think there's some things that we could definitely get a lot better at there.

Scheinfeldt: How about the comments in the code as a way of communicating and do you ever use that? And how important are kind of comments to the smooth development of Mozilla products?

Parmenter: Comments in the code can be good. But, you know, in a way, I don't know that they're super useful. I think every now and then, you'll write some really ugly, scary code that you really just want to write, like, this page long comment. And you'll do that and it'll be great.

But you also find these little, like, one line comments or, like, "I'll fix this later." And I know of one in particular that's been there for the last eight years and it's, like, that guy hasn't worked on the project in six years. You know, it's, like, what does that comment mean? What does that do for you?

So I think it depends a lot. I tend to comment in some of the more scary areas of my code and not really worry about the other ones that maybe I understand. And it's a hard problem. Because I think, as a developer, especially as you're working on that stuff, it's something that you understand at that point in time. So it's not ever clear really to you, what needs to be commented. What do other people who were looking at this code at that point, understand or not understand?

It all makes sense—you know, it all makes sense to me at that point. So I don't really know—

Ryan: How often are you looking at somebody else's code and wishing that they would have made a comment?

Parmenter: Probably fairly often. You know, I think it depends a lot on the code area. Stuff I know reasonably well or have an idea how it should work, is usually not as bad as things, like, when I'm jumping in to the middle of the code. And the other thing is that, I think more often than not, at least recently, I don't think that comments would help as much as sort of, in some cases, more, like, broader design documents. Or not even necessarily design, but sort of a one page write up on, like, "This is crazy, but this is how things work."

You know, sort of once somebody figures it out and gets it all sort of working. They, you know, say, "You know, the printing code, when you hit "print" you know, it goes through these five steps and then it sets this flag over here and you do that." But at the same time, that's hard, because these things change, and keeping the documentation up to date with the code, is—I don't think anyone's ever figured out how to do that in any really good way.

You know, we try to cross reference all of our check-ins with bugs, so that you can look back and see what was sort of the discussion or reason behind it. But that only works to some extent.

Scheinfeldt: Have you ever clashed with another developer over a specific point of code and how did you kind of work that out?

Parmenter: Yeah, well, there's been a few, for sure. One of the probably biggest sort of conflicts was when—so I actually sort of own the MNG library I guess, and all the de-coders and everything. So we had this module owner system, so I guess I'm the module owner of that. And at some point, we had checked in this de-coder for the MNG format which is a multiple network—or I don't know what it stands for.

But it's basically animated PNGs. So animated GIFs were all the rage with, you know, I don't know, 1998 websites or something. And they—but GIF is basically is indexed formats and it only supports 656 colors and it only works with one bit alpha transparency. So either the pixel is there, it's not. Versus PNG, which has full—it has eight bit alpha support. So you can have 256 levels of alpha so you can—things can be translucent. As well as true color support. So you can see as many colors as you want. So basically, you try to do that.

But then something went wrong and they decided that it also needed to do all these things that, like, Flash can do. So it needed to sort of be a movie format and then it—it kind of went—something went wrong at some point. GIFs are very simple, animated GIFs are easy and they—it was basically a very big and complex thing.

And then after having it in our product for, like, two or three years, we did some searchers and there were, like, fifty of these images on the web. So we decided—you know, we said, this is silly, this is like, four hundred kilobytes, you know, which was, like, three times bigger than all the other de-coders and the image library combined.

So we said, you know, this is—we don't need this anymore. So I removed it and we had a huge uproar. People were, like, "We can't believe you removed this, you know." It's, like—it was one of the, I think, probably

one of the few cases where we've really gone in and said, "We're going to remove a feature that we had before." People apparently don't like that.

So I removed that. We had a lot of uproar and people said, "You know, why did you do this?" We said, "You know, we have this set of reasons. It's too big, it's too slow, it's—nobody's really actively maintaining it." What had happened is, the guy who was sort of working on it also said, "We should remove this." But then sort of the maintainer of sort of one of the MNG libraries and stuff that we were using said, "No, No, this is—you know, we'll get this fixed."

And it kind of got ugly for a while. But effectively what ended up happening was, we sort of have a system that's called a "drivers" which sort of—I guess they're in charge of the product releases and stuff or were. And sort of—it got raised up to there and in the end they sort of said, because I sort of said a thing. I said, "You know if you guys can get it under this size and you can do all these things, we'll take it, whatever."

But they never really did. And so they complained and complained and complained and they're still complaining to this day and this was, like, three years ago. So, you know, but in the end, driver said, "You know, figure out how to meet these requirements. Get an active maintainer, do these things." And nobody really has. They've made some effort but it's not really where we wanted to see it go.

I don't think they were really interested in meeting some of the requirements. I think they said, "You know, this is—." They were kind of just trying to chop random bits out and sort of mess with the functionality as opposed to sort of really reducing the size.

So we got rid of it. A lot of people were mad and, you know, it just sort of—in that case, it just sort of—we just decided that was the right thing to do. But I think in other situations, you know, I think a lot of times it's sort of just very small. Like, I mean, that's kind of a big argument. But you get into a lot of much smaller ones about just kind of minor things or design things, architectural things.

And usually I think, most people are pretty good about just sort of going back and forth. I mean, a lot of this community, right, we know each other. We've worked together for awhile and this sort of—you know, often one guy will push the other guy's buttons because he can. And, you know, you get used to that and I think you just sort of figure out—you push their buttons back or you—you know, you kind of—you work it out I think. I don't know that there's really a scientific process behind it. But I think you take what they said into consideration and they take what you said into consideration.

And, you know, sometimes at the end, it's up to the guy who's doing it, right? You know, if you're running some piece of code, you know, you'll argue back and forth and if you think you're right, then that's probably how you're going to go do it. But it's always good to get that input and figure it out. I certainly—Vlad and I ask each other questions all the time. We're, you know, "What do you think about this and this?" And I'll tell him how I thought he was going to actually do it. He's, like, "Oh, no, that's crazy. What are you talking about?" And it's, like, "Oh, well, yeah, you're probably right." Or part of it and, you know, I'll take it into consideration, try and do the best thing.

Ryan: You mentioned that you got started—you started volunteering and you spent a lot of time volunteering. What motivated you to volunteer?

Parmenter: You know, I don't really know. I think sort of the way I learned programming was sort of in this open way—other people looking at other people's code. You know, it was sort of figuring out how things worked. I think sort of taking them apart. I mean, that's basically what you're doing is, you're taking these things apart to figure out how they work. It's sort of like taking a radio apart, I guess, to figure out how it works.

And so, I think it's sort of that exposure, that sort of—not even really looking at open source at that point, right? More just, this is how I—I'm looking at these random bits and saying, "Oh, well, if I copy and paste that here and—these things will eventually start to work."

And I think at some point, I started using Linux, so I'm sure that had something to do with it. But I think always just sort of the idea that, you know, there's this big group of people who you can collaborate with who are smarter than you are. And, you know, there's this vast knowledge that—I was really sort of chasing, to figure out, how can I be better at this—these things? How do I learn all these great things?

There's just so much value—there's so much knowledge out there that—and it's so easy to get to. I mean, it's so—all these communities are very accepting, very sort of, you know, a lot of good people. There's certainly a couple who are, like, "You're stupid. You don't know what you're doing. Get out." But aside from them, there is, you know, so many people willing to help and teach and I think that's probably the main reason I got involved.

You know, at the same time, right, Mozilla's like this big—especially looking at it now. It's this big, giant thing, it's shiny and flashy and it's the cool thing to do because, you know, and you see that your stuff gets out,



you know, millions and millions of people and that's certainly worth something.

You know at the time of Mozilla, I think there was probably some other—you know, Netscape, right? It's, like, oh, that's a big company. Yeah, but they were being, like, destroyed by Microsoft at that point. You know, still it was, like, wow, everybody's heard of Netscape.

But—yeah, I don't really know. I think just sort of learning and I think it's sort of—it's the way it's always been for me. I don't think—I think it's probably—I've worked with enough people, who sort of didn't come in to open source in a similar way and it's always—they have such a different perspective on these things. It's always been kind of interesting.

Scheinfeldt: Why do you think that Mozilla, and I guess particularly Firefox, has been able to attract a large number of users? And sort of, what sets it apart from other open source projects that might be good, but can't attract a lot of people?

Parmenter: Well, I think the number one thing—well, I don't know if this is really the number one thing, but if you look at sort of overall, like if you looked at the number of open source projects out there and looked at what platforms they run on, I would say ninety nine percent of them are for Linux. And—because everybody starts with this one-off little thing and they're the same as the other ten or twenty programs that do the same thing but they have a different name and whatever.

And I think sort of the—understanding the importance of Windows, right, and other platforms, I think was a key early decision. You see, like, Open Office who does something similar. But if you look at, like, you know, our user numbers are, like, ninety percent Windows, right? So I think if you chop that number off, our user base is still relatively small. I mean, we'd be probably more successful than a lot of open source projects. But, you know, still, I think that was a big piece.

I think also, you know, we filled a gap and—with sort of ignoring Firefox for a second and sort of looking at pre-Firefox, I think we were still filling a gap there for sort of this alternate browser market. I think more recently, I think with Firefox, I think a lot of that, I think we got lucky. I mean, I think we'd had something there the whole time really, I think. And I don't know that Firefox—I think Firefox did a better job of being a little more user friendly. I think it, you know, it separated out. A lot of people weren't using the mail client. A lot of people weren't using the editor, things like that and I think that was probably an important piece.

You know, the download became smaller. I think all of these things sort of added up. But I think a lot of it really was just we got lucky. We got, you know, a good break where people started looking at it and they said, "Hey this is, you know, this is good stuff." I think all of the sort of recent security issues with all the browsers, I think that's helped. News media sort of picked up and said, "Hey, look, if you're using IE, you're going to get viruses or whatever. Your computer is going to get destroyed." And I think a lot of that's helped.

But yeah, I don't know that there was really any single thing that really happened there that sort of launched us into popularity. I think we've sort of been around for a while. I think people have been using it and, you know, I think, it just—we finally, I think, maybe it's that we really finally got to the point where it was simple enough that your mom could use it. Or my mom could use it or anything like that, without really having to think about anything else. I think that's been a big part of our focus. Trying to get the auto update stuff working so, you know, you install this, like, 1.5, and it's just going to keep updating itself, whenever we push out a new release and you never have to upgrade it again, things like that.

Sort of I think being a little more user focused. I think before, we always were far more sort of—I don't know, I think it was more developer focused. Everybody sort of put in whatever they wanted and it just sort of appeared and it kind of—it didn't really have a very good sort of person to say, "That's bad, that's bad." And I think that was sort of the development process we'd followed for so long. I think that was part of the problem, when I removed the MNG stuff, was, people were, like, "Wait, somebody said no?" You know, it's never happened before.

So I think we've sort of set that and have tried to really focus it. But yeah, I can't really point to the single thing that says, you know, why we went from four million downloads or ten million, I don't know, whatever we were at before to you know, a hundred and twenty million downloads. So, yeah, I don't know.

Scheinfeldt: How, if at all, has the success of Firefox sort of changed the rest of Mozilla? And what's the relationship now of Firefox to other parts of other projects, Thunderbird and the Calendar project?

Parmenter: I suspect you'd get pretty different answers from different people. But to me, it doesn't really seem like a lot has changed. I think all along there's been way more people using the browser than have mail or calendar or anything else. I think calendar—calendar's relatively new, it's been around for a little while now, but it's relatively new. You know, we've—when I started working on the project, we—there was no mail news. The mail code was all dependent on some proprietary database, I believe, that

they couldn't, you know, open source and so they had to hold a lot of that code.

So yeah, but mail's been around for a long enough time. I mean, it probably came out in early '99 or something in this source code. But yeah, and I don't think it's ever had the user base that the browser piece has. There's been enough other mail clients out there that other people were using.

So I don't—I think it's always had a small community, I think. If you look at Netscape, it had a bigger—it had a much bigger team than it probably does now. But I think that was just the way that Netscape knew to do things. If you look at the relative sizes, I don't know that they are significantly different. I think the browser had a much bigger team than the mail did.

Calendar, you know, I think there's always been sort of these side projects and I think a lot of them have sort of continued in the way they have. I think we're starting to look, now that we've had a lot of success with Firefox and everything, I think we're looking at these other projects a little more carefully now to see, do they make sense? Or how do they fit in? Can we help them? Can we make them better?

But at the same time, I think keeping Firefox has always been at the front, you know, in the front. And we're still—I mean, we've gained a lot of people in the last year. But you know, for the stuff we're trying to do, it still doesn't feel like, you know, we're all swamped. So it's very hard, I think, to balance getting enough people on all these projects. And then at the same time, right, like, you want to focus on the things that are really going to be popular and going to be good for everyone.

But how do you ever know what those are? It's, like, if we added a hundred people to Thunderbird, does that make Thunderbird more popular? Or does that just add a hundred people to working on Thunderbird?

So I think we've had our feet wet with Thunderbird for a while and I think it hasn't—it has got its user base and they're happy. You know, I use Thunderbird every day. But there's certainly—that community doesn't seem to grow like we've seen in Firefox.

So I think we haven't put a huge number of new resources into that. And I think with things like calendar and stuff, I think we're finally getting to the point where—those have always been community based, right? They've never really been driven by sort of the core Mozilla group. They've been sort of on the outskirts and I think we're probably starting to look at those

a little more carefully and see, you know, can we—are there things we can do to help? You know, I don't know that we're—we've started to commit or anything to saying, we're going to make that a Mozilla.com product. But I think, looking at the community and seeing how we can help them and how we can grow that community better.

Ryan: Do you consider working at open source software projects performing a public service? Have you ever thought of it like that before?

Parmenter: No, I haven't really thought of it like that. Performing a public service. No, I don't really think I do. I don't really—it doesn't feel like a public service really. I mean, it just seems like a different way of developing things. I think it's more of a mind set and a sort of community driven project, more than sort of a public service type thing.

Ryan: How would you define, just sort of generally, a successful open source project?

Parmenter: I'm not really sure. I think there's a lot of different metrics for success. I think one of the key pieces for open source is that—or for really any product, I think, is finding something that people want to use. You can create open source projects all day long that are—nobody cares about except for you, or maybe you don't even care. I would say those are certainly unsuccessful.

You know, you can build up a giant community around something that still the number of—you know, we had a—our community size, I don't know that it's grown. Like, as far as developers and—maybe our tester size has grown a bit. But I don't think, like, our overall developer community has grown a huge amount since Firefox became big, giant success.

So I think we were a fairly successful community prior to any of that. So I think, you know, I think if you get a good community set-up, I think that's probably a good metric for success.

You could look at how many people are using that product or project and that's probably also a good metric for success. But I think it sort of depends on what you're looking for.

Ryan: Do you think that the popularity of Firefox will—or what do you think it might do for sort of the movement, this open source software movement, as a whole?

Parmenter: Moving—open source? I don't know. I mean, I would hope that more and more projects see sort of things that we've done and learn from them. I

think, you know, we've made as many mistakes, maybe more, than any of the other projects out there. But, you know, certainly we've done some things right. So I think there's a lot of things to learn from us.

You know, figure out something that makes a difference and really just sort of be passionate about it and be—try to make a difference. I think there's a lot of good stuff out there that's open source and, you know, I think there's a huge market for it. I think there's a lot of room for these products and projects and people just have to figure out how to make that work. I don't think anybody's really sort of found the magic, you know, how do you make billions of dollars off an open source project yet?

People have tried, you know, doing support for things and all these different things and they sort of—some of them work, some of them don't. And I think you just sort of figure out what makes sense and, you know, get it out to the users. Do something well. Try—I mean, polish it, think about the users. I think that's the biggest mistake really that a lot of open source projects have, is they end up with either engineers designing the UI for things, which, not always, but generally, turns out bad.

Or, you know, they're just focusing on what's interesting to them or what they want, as opposed to sort of, what does everyone else want? And, you know, I think figuring that out, figuring out that a lot of people run Windows. Figure out that, you know, that's a good market for you guys. Figure out how to move that and I think that'll help them a lot.

Scheinfeldt: Great, well, I think that's all we got. So thanks very much and we'll let you know when this goes online.

Parmenter: Sure.