

Person1: Uh, from your point of view, what is sustainability, in terms of software?

Person2: Um, so, I guess there's two aspects to it. One would be, uh, that the code, uh, should run on, um, on hardware in the next, for the next, for the foreseeable future. Um, so that means, that it's sufficiently portable, that it can, uh, run pretty much, uh, any hardware of interest, uh, in our case it's mostly uh, Unixes, uh, whether laptops or super computers. Um, and, uh, had a second aspect of [inaudible 00:03:23] now. Uh, in, oh, yeah, it means also, that, um, adding, so most of the codes we work with are, are always in active development, um, so the, sustainability, I think is also about how expensive it is to add a new feature in terms of human resources.

Person1: Mhmm (affirmative).

Person2: Um, and, and yes, so that would be the second handle to it, uh, that, uh, the cost of adding new features should not increase exponentially, as it does in some codes.

Person1: Mhmm (affirmative).

Person2: Um, yeah.

Person1: Um, what are the attributes or features this software, uh, has that led you believe that it is sustainable.

Person2: Could, could you repeat that?

Person1: What are the attributes of, or features, of the software that led you believe to believe that it is sustainable?

Person2: Um, so, generally, I, uh, again, so, so this if from point of view of a programmer, so I have access to the source code. Um, by looking at the source code, it's generally fairly easy to look, to see whether it will be, uh, sustainable in that, uh, it will have tests that will make sure that it's fairly easy to be portable. Uh, we can look at the dependencies, uh, so if it depends on, uh, on proprietary, uh, libraries and such, that generally makes it less, uh, less sustainable because uh, because you have to have those libraries, and if something happens to the provider, or if the cost increase too far or simply there's no money for it, uh, then the, the software goes pear shaped.

Um, there's how it's coded, whether it's, it's, it's sufficiently easy to get into, and to understand what the code is doing. Um, so that adding new features is relatively easy. So, um, yeah. I, I guess that's it.

Person1: Uh. Regarding this software that you have developed, was sustainability a consideration?

Person2: It wasn't a direct, um, I mean, how do I put it? I, so, I mean, today it is, up till, up until I joined this group, I wouldn't, I, it wasn't explicitly on my mind. Um, I would try and think on how, of how people would try and interact with the software and how easy it is for them to understand what I'm thinking when, when I programmed something. Uh, but I didn't think explicitly in terms of, of writing code that would

be, uh, sustainable that is, that, um, that can be used in, in, uh, a portable way or that, uh, uh, that other coders might get into. Um, so I'm sorry, what was the question?

Person1: Uh, regarding the software that you've developed, what sustainability or consideration?

Person2: So, now yes, so, now I try to write a software in, uh, since basically we're writing for, for customers, if you will. I'm trying to write a code that will be understandable, by, um, by our colleagues who don't have necessarily expert programming experience. So, it's, it's really about clarity and, uh, um, and, uh, ease of use. Uh, so, uh, so, where ease of use is, is both for having features for coders, or, uh, for the users of that, that, uh, software.

Person1: Mhmm (affirmative).

Person2: Um, it's also about, if, if the, some software design practices aren't used in that group, it's about recommending that they use stuff like, um, uh, version control testing, um, that kind of stuff.

Person1: Mhmm (affirmative). Um, if you asked at what point in time did it become a consideration? Sustainability.

Person2: Um, so it was an explicit consideration, um, when I, when I joined this group, because, uh, our, our, uh, group manager, group team leader, uh, Person1, is, is big on this kind of thing. He's a fellow with this, uh, [inaudible 00:07:47]. Um, so, uh, so he talked about it, and I can see that it makes sense.

Person1: Mhmm (affirmative).

Person2: Ah, that basically it was a concept I didn't have a name for, but I, I could, you know, relate to, um, so, uh, so that would be, you know, I can put it a date on it, January 2013, I guess.

Person1: So, it was from the beginning.

Person2: When, when I joined this group?

Person1: Yeah.

Person2: Yes. It was definitely, uh, a part of, it is part of, of, you know, uh, um, founding document for this, for this team, so. Yeah.

Person1: Mhmm (affirmative). Have you done any projects that were not sustainable?

Person2: Um, depends on how harsh you are with these projects.

Person1: Mhmm (affirmative).

Person2: Uh, there's a, I've, I've worked with a number of projects. Um, for which, uh, which is, you know, despite what, what we think, still exists, and probably will still, uh, exist, but the cost of keeping these, uh, projects working on your hardware, of

adding stuff to this, to these projects, is, is, uh, very high, in terms of the number of people that work on it. Uh, this is mostly, um, codes from, um, that, I guess, were started in the 80's, uh, in Fortran, and have grown to be huge buh- blobs um, and so, uh, they still exist, they're still very much and very widely in use, but, uh, uh, they're, they're definitely not sustainable most definitions, like, unless you pour sufficient amount of energy and money in it.

Person1: And the, were there any consequences of it not being sustainable?

Person2: Um, yes. Our research has slowed down. Uh, what should be easy isn't.

Person1: Mhmm (affirmative).

Person2: What should take, you know, a few minutes, can take a few months, uh, to modify, so, so, uh, yeah, the cost is, is, uh, relative slow down of, of features that can be added to these codes.

Person1: Mhmm (affirmative).

Person2: Um, so, a slow down of the science done, there, with those codes.

Person1: Yes. Yeah.