Reaction of Guy Steele Tech Talk

I really like the style of the talk. Guy Steele starts the talk by start defining words in English. Set up rules like we can use one-syllable words as granted and define multiple-syllable words. In the beginning, I was really confused why he was giving the talk in such style. However, after keep watching and after he start taking about the growing of different programming languages, I got the point of such style. He wanted to use the talk to show us the growth of a language what we are all familiar with, English.

One philosophy that I really like is the part that Guy was explaining the Worse is better. It is such an interesting human nature that when we are doing something, we are always trying to do the “right” thing. However, is it always “right” to do the right thing? Doing the right thing may end up with a big cost, like Guy mentioned in the talk, it may take a lot of time or a lot of manpower. So it may not be the right choice to do the “right” thing. Combining the later topic about “pattern” and “meta-pattern”, I think that when we are trying to design a programming language, it would be better that we start off by designing a small language, a pattern of a language and give the user enough space to contribute and after certain amount time, the original language that we developer will become a “meta language”. His analysis of weather a language should be small or large reinforced this philosophy.

When he was talking about weather we should use a small language or large language, he mention the draw back of both of languages. However, when I was taking java as an example, I was wondering why Java, as a big language is easy to learn. Should a language be large or small? After listen to more of the talk, I found that the reason of the success of Java is not that it is either a large language, or a small language, but it is designed for growing. Java starts off as a small language, which is easy to learn. On top of this, Java allows the user to help develop the language with defining their own types and creating library.

The way Java works highly reflects the “high point” of the talk, we need to design a way of doing rather than design a thing. The original built in Java Jre gives a pattern, which is a way of doing. So the users can use the pattern as a meta-pattern to develop their libraries, which helps the languages grow.

Even though, the talk was given more than 15 years ago, a lot of the principle still holds. If we take look at some current popular languages, for example javascript, we will find that the reason of its popularity largely comes from it is simple to learn and is designed to grow. When I started learning javasript, people always tells me, if you don't know how to do something in javascript, there is probably some open sourced library online that takes care of what you need. Now a days, with the fast grown of the open source community, the growth of a language has become way faster than 15 years before, the language that are not design for growth will loose its popularity much faster. There is one thing that may not hold any more now a day. He mentioned that we need to design a way of doing rather than design a thing in the “high point”. I think the “design of things” has become more and more important. Languages are growing so fast that if we do not have a good idea of what direction the language is growing, the language may end up not able to fulfill user’s needs and fade out from the market.

All in all, I learned a lot from this tech talk. It changes the way that I look at a programming language. In the past, when I heard people’s comments about a programming language, I never understand why a programming language is good. But after watching this video, I found that since a good programming language is that language design for growth, we also need to look at a programming language in a growing way, we need to have a prediction about how the language is going to look like after 5 year, 10 years, even longer. I found it is interesting that he mentioned that there may be a language eventually that may fulfill all of our purpose which we do not need any other languages anymore. I know it is a ideal condition, but I am interested in seeing what language it is going to be or at least what language is coming close to this ideal.

I think this is definitely a very useful video to watch and I would highly recommend keep requiring the video. I may watch it for couple of more times myself.