

THE LINGUISTIC RELATIVITY OF PROGRAMMING LANGUAGES

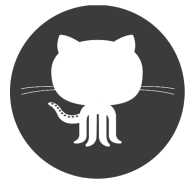
Jenna Zeigen
Frontend Camp 2015

jenna.is/fecamp15.pdf

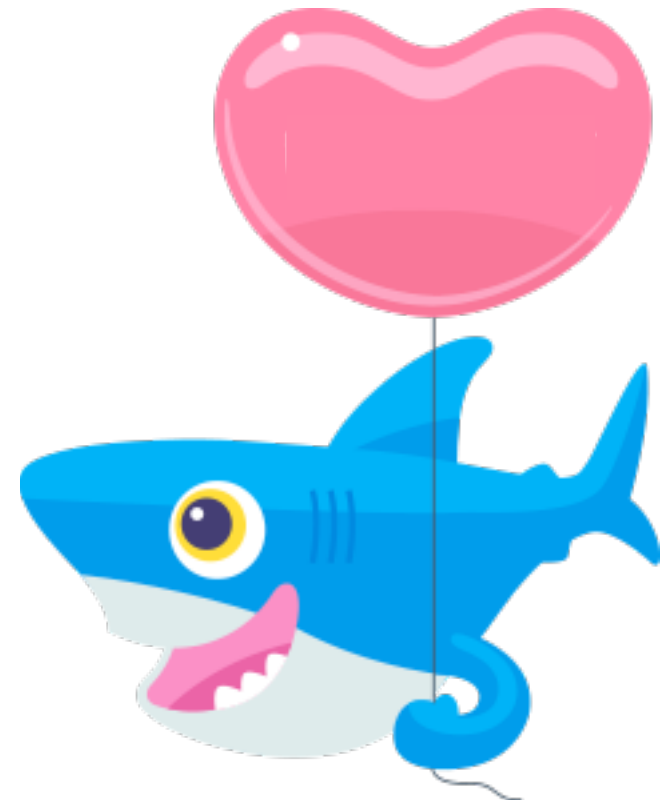
User Controls Team Lead @ DigitalOcean



zeigenvector



jennazee

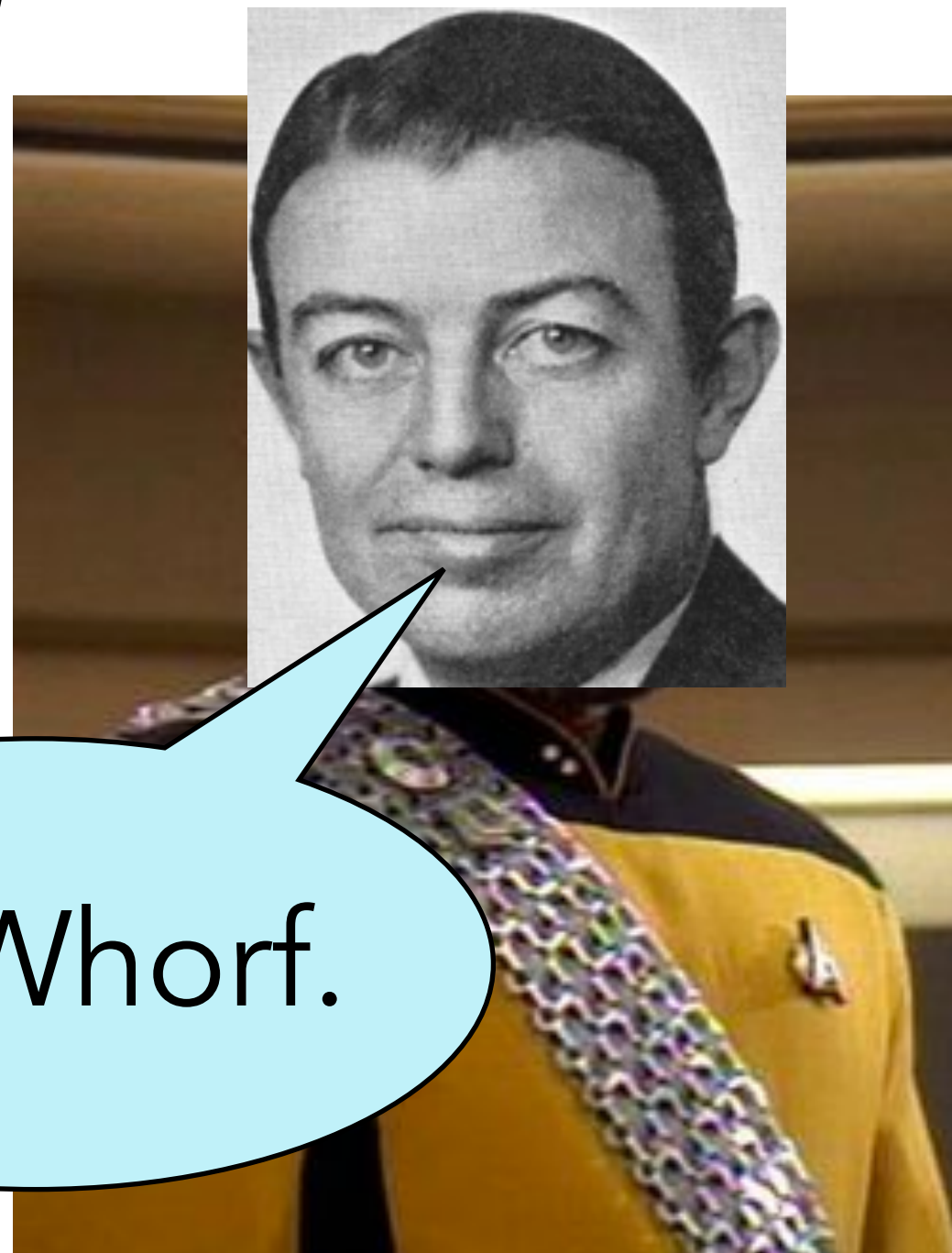


1. Linguistic Relativity
2. ... and Programming Languages?
3. Previous allusions
4. How it applies
5. Implications

I'm Sapir.



I'm Whorf.



SAPIR-WHORF HYPOTHESIS

The languages you speak
{determine | influence} the way
you think.

SAPIR-WHORF HYPOTHESIS

The languages you speak
~~{determine | influence}~~ the way
you think.

SAPIR-WHORF HYPOTHESIS



Green

Blue

SAPIR-WHORF HYPOTHESIS



Grue

But what about the

JavaScripts

????????????????

Proposal: The programming languages we know strongly influence the way we think about programming.

Programming languages create
and manipulate the space,
rather than just describe it.

“Programming languages,
because they were designed for
the purpose of directing
computers, offer important
advantages as tools of thought.”

Kenneth Iverson, “Notation as a Tool of Thought” (1979)

(bit.ly/Iverson-NotationAsToolOfThought)

BLUB PARADOX



Some programming languages are more powerful than others.

Paul Graham, "Beating the Averages" (2003)

(bit.ly/blub-paradox)

BLUB PARADOX

“Some programming languages are more powerful than others.”

Weak
language

Blub (average)



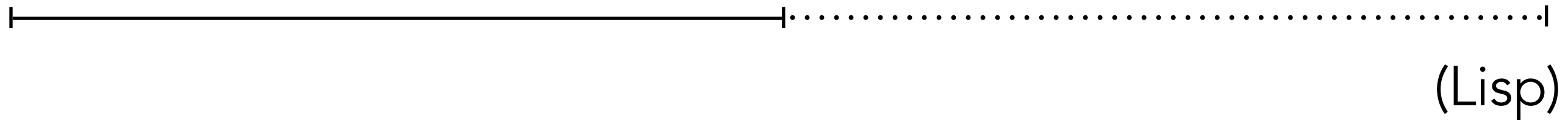
BLUB PARADOX

“Some programming languages are more powerful than others.”

Weak
language

Blub (average)

Super strong
language!



BLUB PARADOX

"I look at [Python, Java, C, and Perl].
How can you get anything done in
them, I think, without macros?"

Paul Graham, "Beating the Averages" (2003)

(bit.ly/blub-paradox)

BLUB PARADOX

“They're satisfied with whatever language they happen to use, because it dictates the way they think about programs.”

Paul Graham, “Beating the Averages” (2003)

(bit.ly/blub-paradox)

BLUB PARADOX

“I know this from my own experience, as a high school kid writing programs in Basic. That language didn't even support recursion... but I didn't miss it at the time. I thought in Basic.”

Paul Graham, “Beating the Averages” (2003)

(bit.ly/blub-paradox)

SCARRED FOR LIFE?

“It is practically impossible to teach good programming to students that have had a prior exposure to BASIC: as potential programmers they are **mentally mutilated beyond hope of regeneration.**”

-Edsger Dijkstra, “How do we tell truths that might hurt” (1975)

(bit.ly/dijkstra-truths)

MENTALLY MUTILATED



BEYOND HOPE?!?!11/?

memegenerator.net

We are influenced by the constructs and idioms of the most powerful programming language we know, not the languages themselves, or the language we are using at the time.



(<http://bit.ly/cutest-red-panda>)

1. We can learn more (powerful) programming languages and how to program in them idiomatically.

HUMANS CAN LEARN

iteration

HUMANS CAN LEARN

iteration

list comprehension

HUMANS CAN LEARN

iteration

list comprehension

map

2. We can implement the constructs of more powerful languages in whatever language we use.

CODE CAN MAKE THINGS

“We should now think of a language design for being a pattern for language designs, a tool for making more tools of the same kind.”

Guy Steele, “Growing a Language”

(bit.ly/growing-a-language)

CODE CAN MAKE THINGS

```
_.map(array, function(el) {  
    console.log(el);  
})
```

```
array.map(function(el) {  
    console.log(el);  
})
```

CODE CAN MAKE THINGS

“Sweet.js brings the hygienic macros of languages like Scheme and Rust to JavaScript. Macros allow you to...craft the language you’ve always wanted.”

The screenshot shows the GitHub interface for the `mozilla / sweet.js` repository. At the top, the GitHub navigation bar includes the logo, a search bar, and links to Explore, Gist, Blog, and Help. The repository name `mozilla / sweet.js` is displayed, along with a 'PUBLIC' label and icons for Watch (131), Star (2,141), and Fork (116). Below the repository name, a description reads 'Sweeten your JavaScript. <http://sweetjs.org>'. A summary bar shows 1,102 commits, 8 branches, 17 releases, and 23 contributors. The main content area features a green 'branch: master' button and a 'sweet.js' link. A recent pull request is highlighted: 'Merge pull request #316 from natefaubion/macroclass-where' by user `disnet`, authored 4 days ago. The latest commit is `bc28a4244b`. A list of recent commits is visible, including '_layouts cleaning up' (8 months ago) and 'bin updating the build' (10 months ago). On the right sidebar, links for Code, Issues (45), Pull Requests (4), Wiki, and Pulse are provided.

GitHub repository page for `mozilla / sweet.js`. The page shows the repository name, a description "Sweeten your JavaScript. <http://sweetjs.org>", and statistics: 1,102 commits, 8 branches, 17 releases, and 23 contributors. The main content area displays a pull request merge for #316 from `natefaubion/macroclass-where` by user `disnet`, authored 4 days ago. The latest commit is `bc28a4244b`. Below the pull request, a list of recent commits is shown, including "_layouts cleaning up" (8 months ago) and "bin updating the build" (10 months ago). The right sidebar contains links for Code, Issues (45), Pull Requests (4), Wiki, and Pulse.

CODE CAN MAKE THINGS

The future can be now
(kinda).

3. Programming languages
are synthetic and can change.

SYNTHETIC LANGUAGES CAN BE CHANGED

iteration

map

SYNTHETIC LANGUAGES CAN BE CHANGED

iteration

map

array comprehension

generators

SYNTHETIC LANGUAGES CAN BE CHANGED

Languages that can't easily
grow will die

Guy Steele, "Growing a Language"

(bit.ly/growing-a-language)

Learn new languages!

Find cool things!

Bring them back!

Share!

Slides: jenna.is/fecamp15.pdf

Blog Post: bit.ly/ling-rel-prog

 zeigenvector

Thanks!



↑
Me