Documentation at Scale

Kenneth Reitz

Hi.

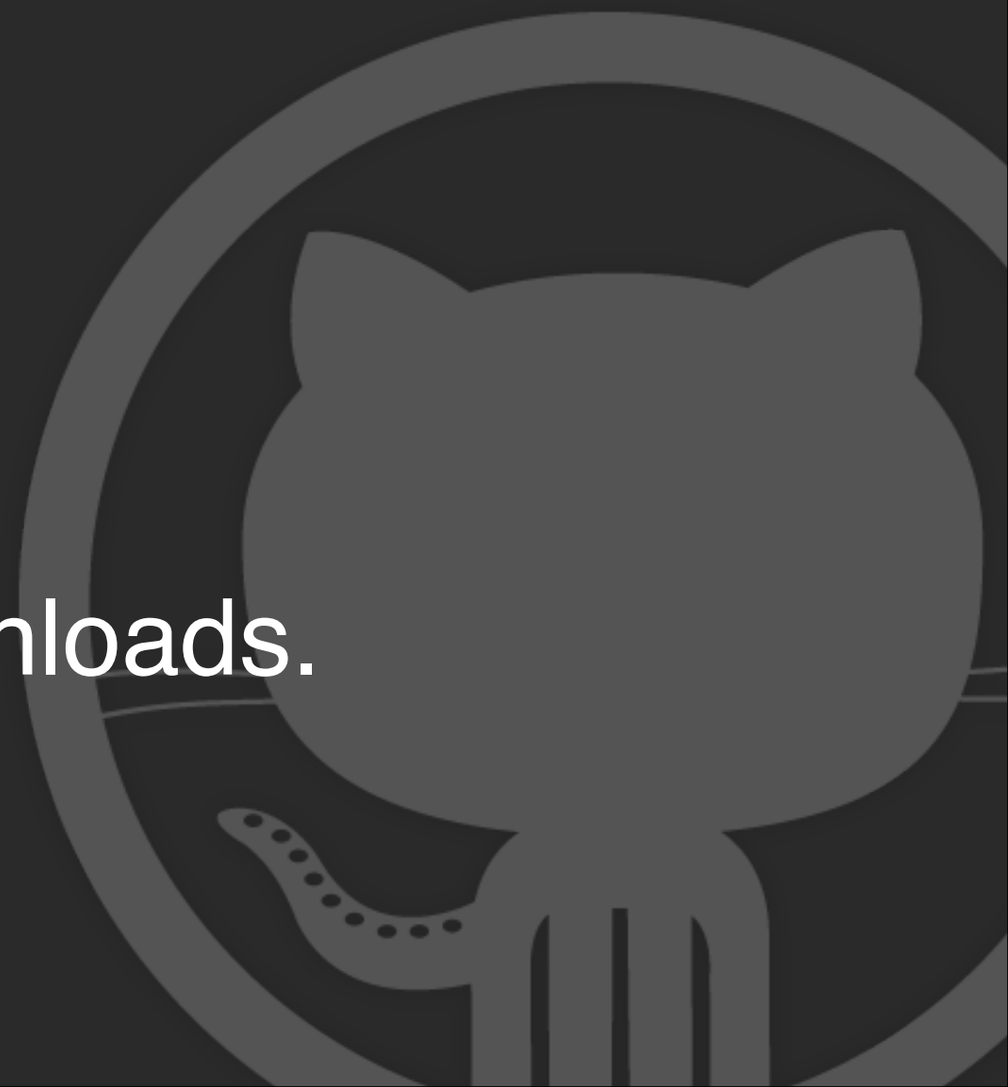
@kennethreitz

heroku

Python Software

github.com/kennethreitz

- ~18 serious projects.
- 100+ experiments.
- OSX-GCC-Installer: 56TB of downloads.
- Requests: 13.8+ million downloads.



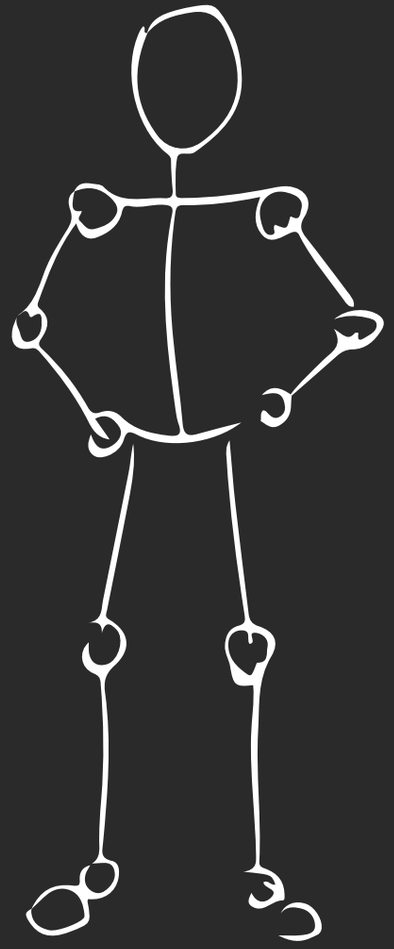
Other Interests...

- Street Photography & Photojournalism
- Synthesizers & Music Production
- World Travel (~140,000 miles last year)
- Public Speaker (29 events last year)

Language

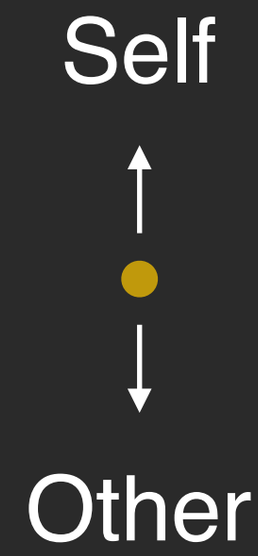
Early Human:

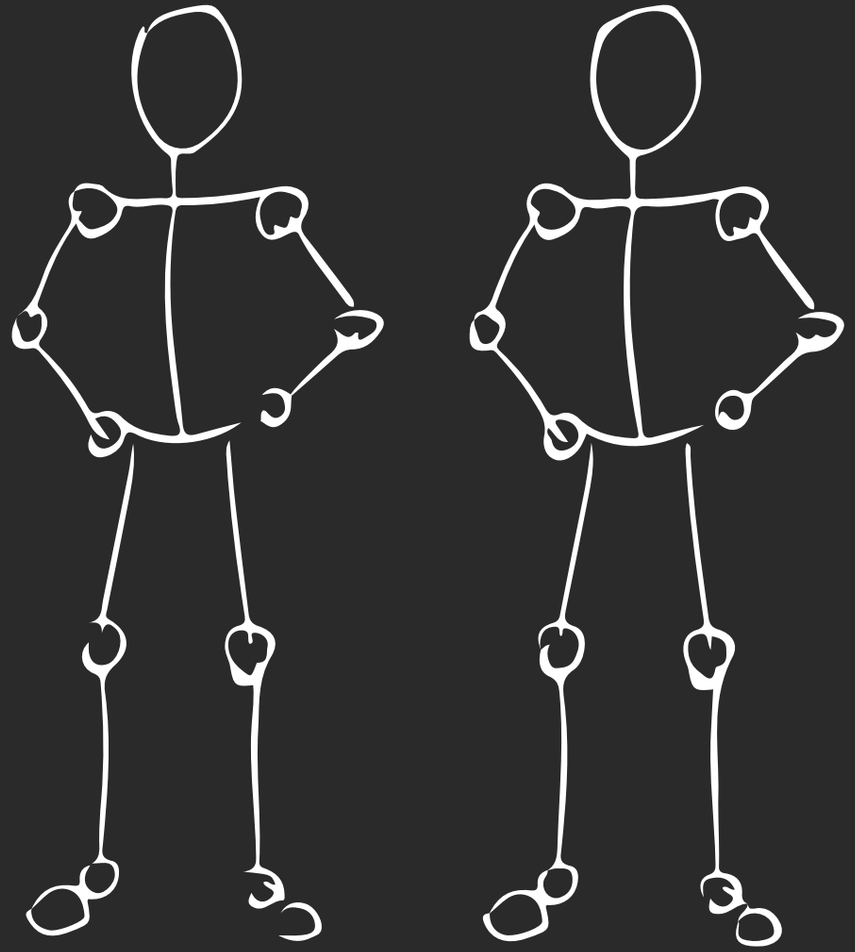
Alone with self & ideas.



Spoken Language:

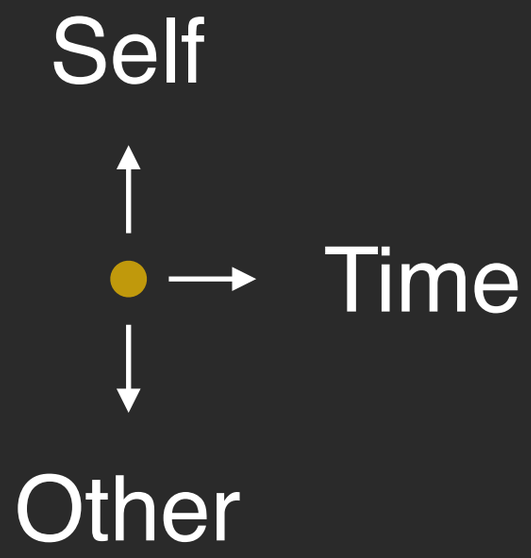
Express ideas to others.

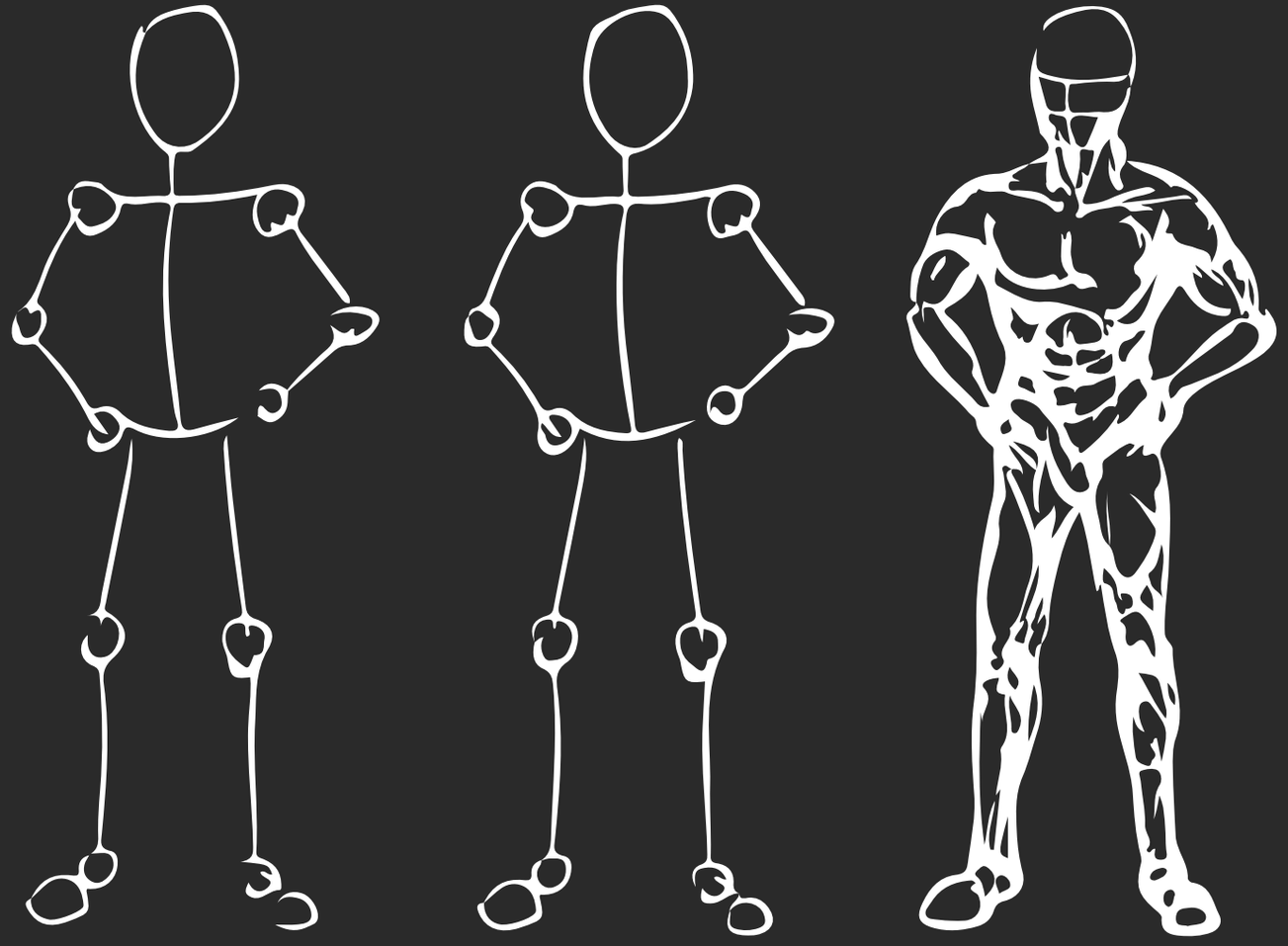




Written Language:

Persists ideas over time.





Human hardware hasn't changed —

The software has been upgraded.

Communication: One-to-One.

At first, language was mostly used for a single person to communicate to another single person, or a small group of people.

This is no longer the case.

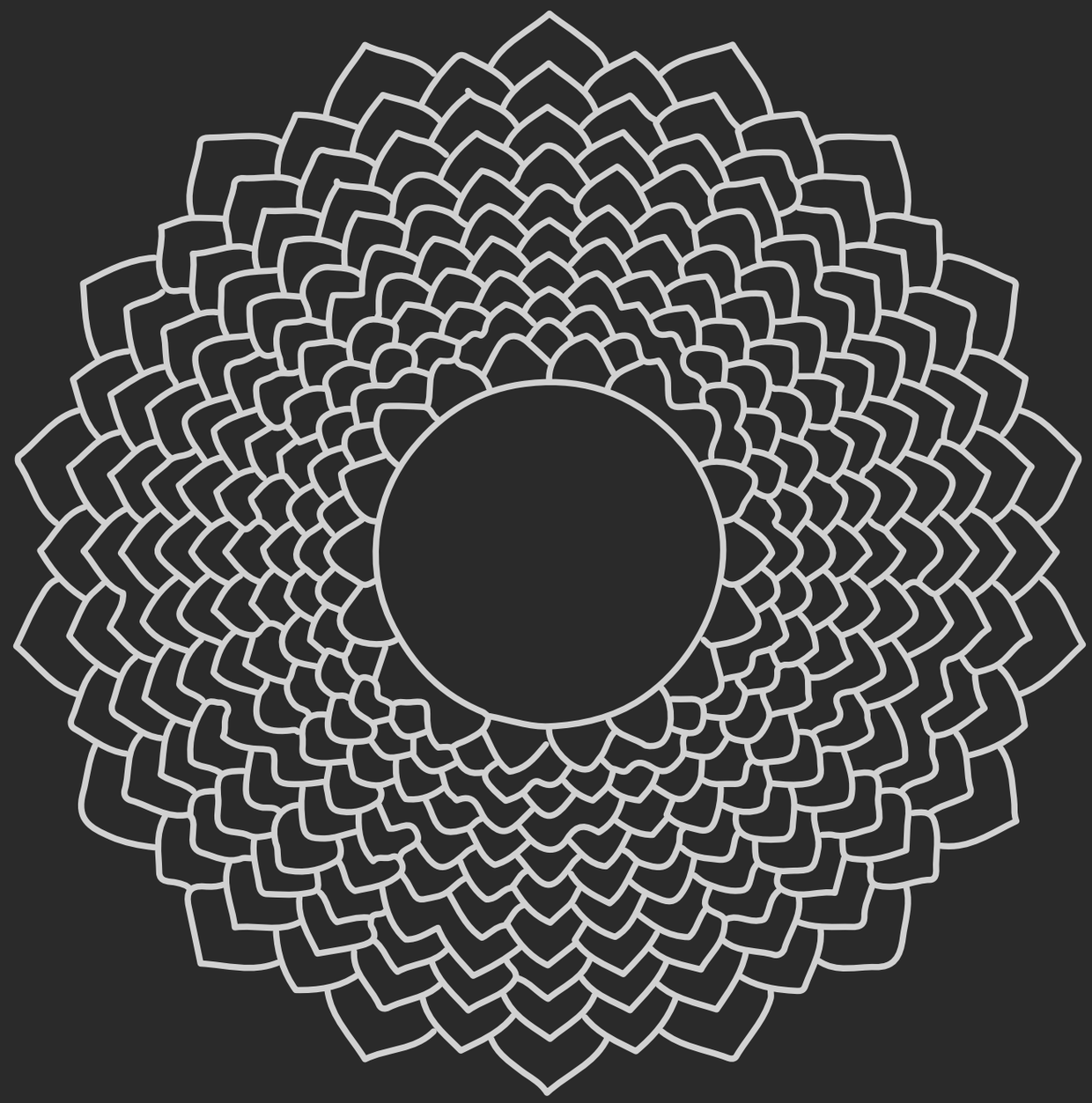
Communication: One-to-Many.

- After the printing press, a single privileged entity could communicate to the masses.
- Newspapers, Books, Television, Radio, etc.
- This formed the narrative of "the public".

Communication: Many-to-Many.

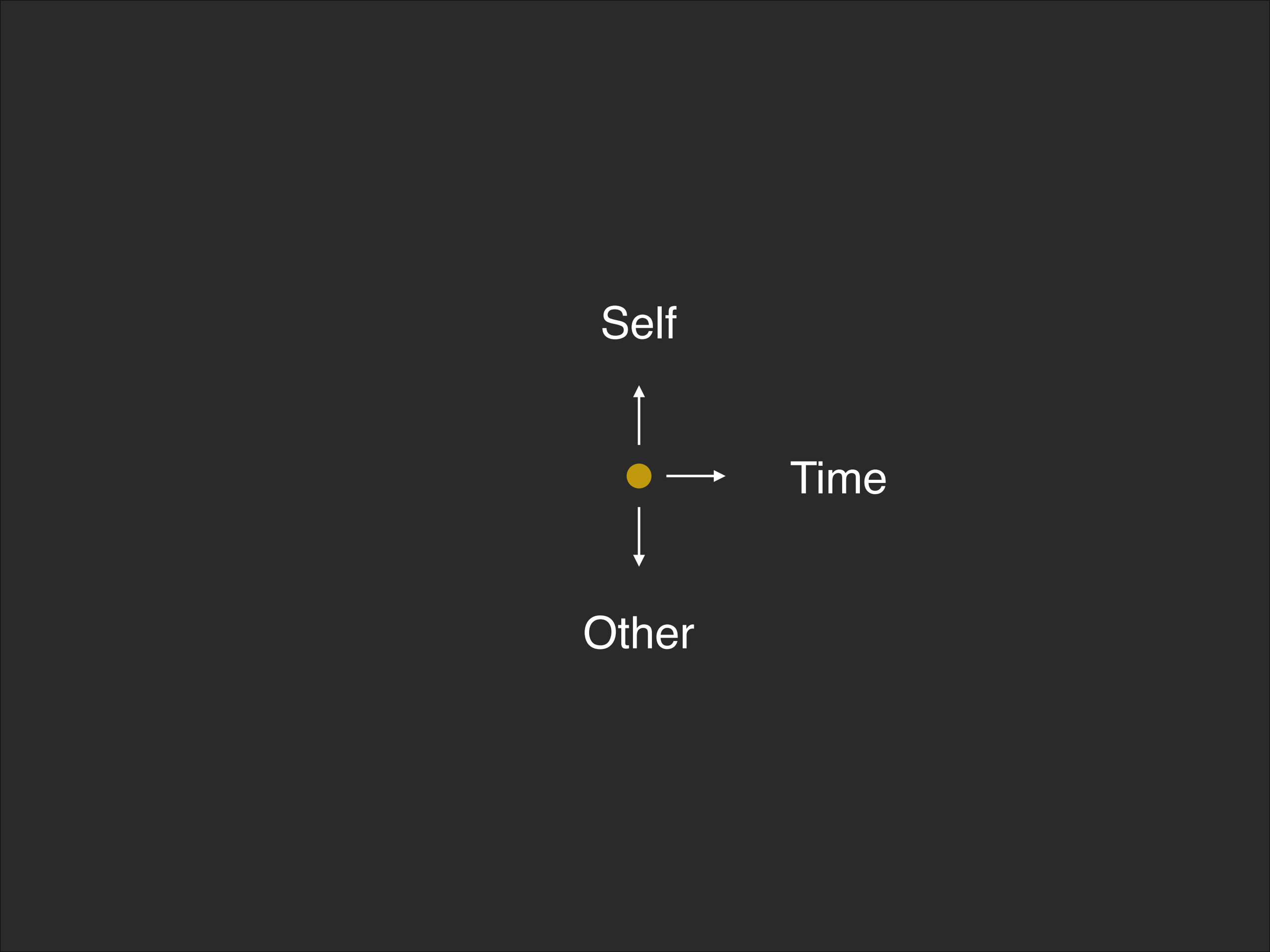
?

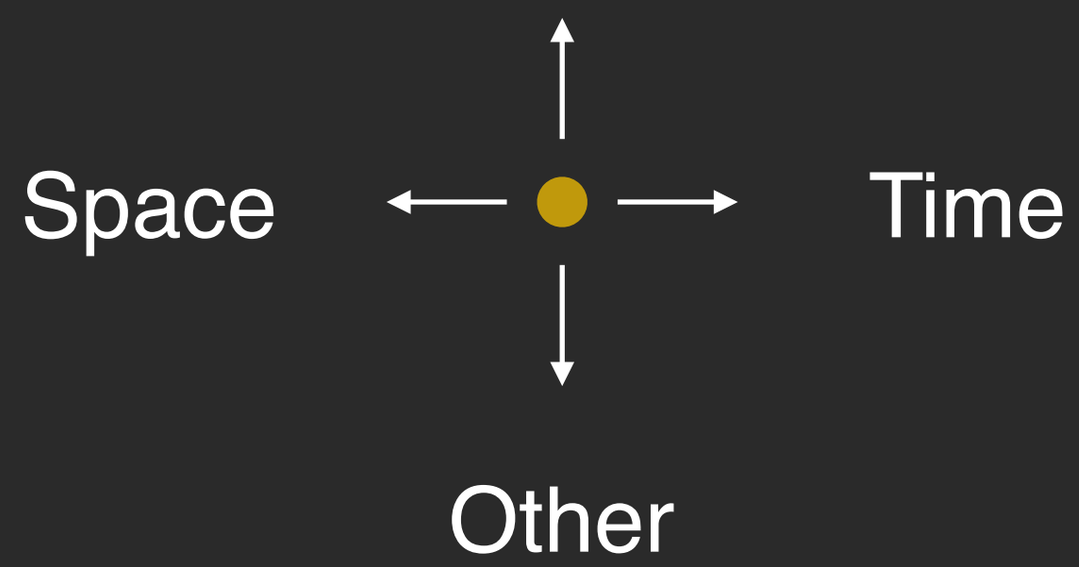
The Internet!

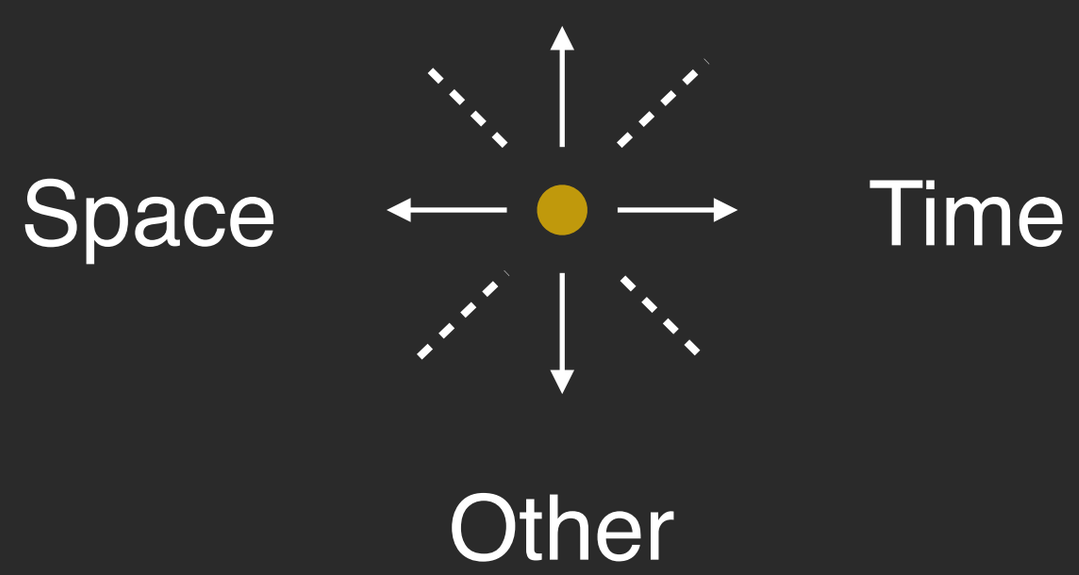


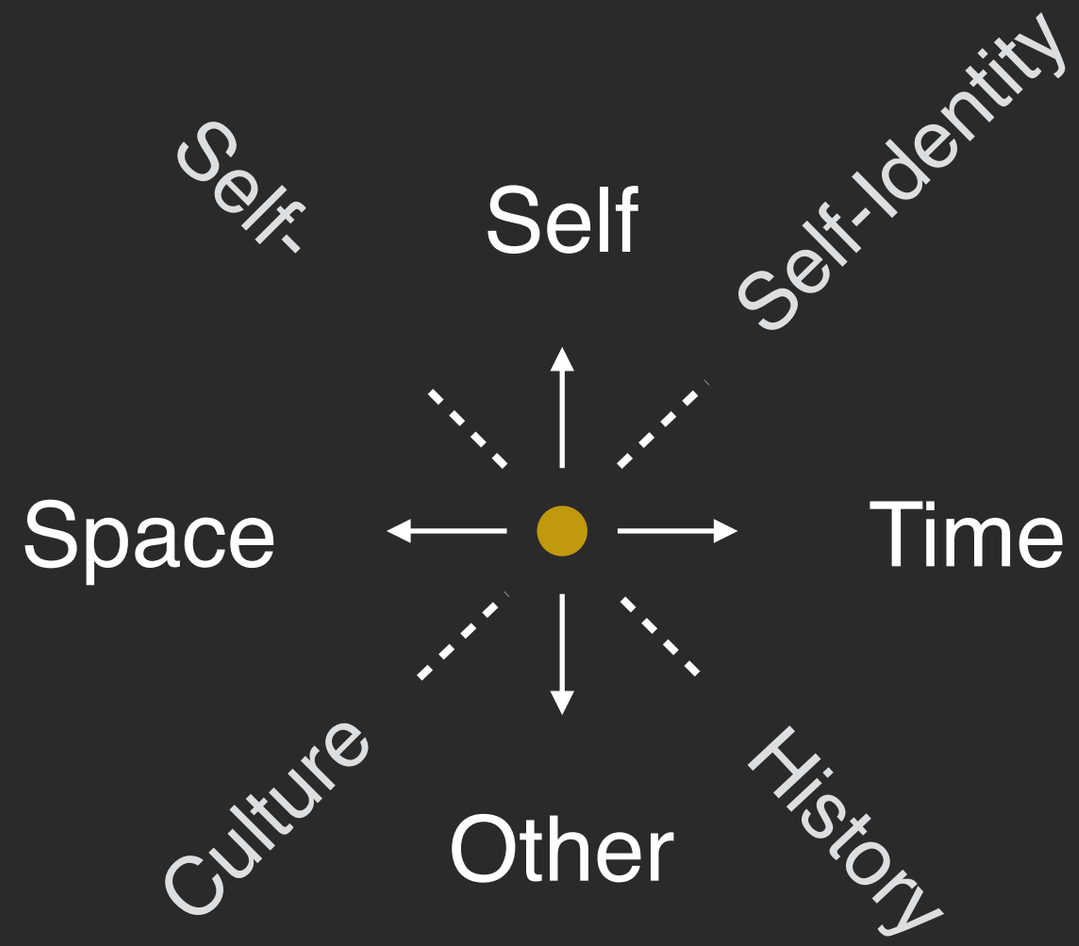
Communication: Many-to-Many.

- If you have access to the internet, you have access to a universe of information and ideas.
- Anyone can publish anything to any number of people, large or small. A level playing field.
- The implications of this are huge.


Self
Time
Other

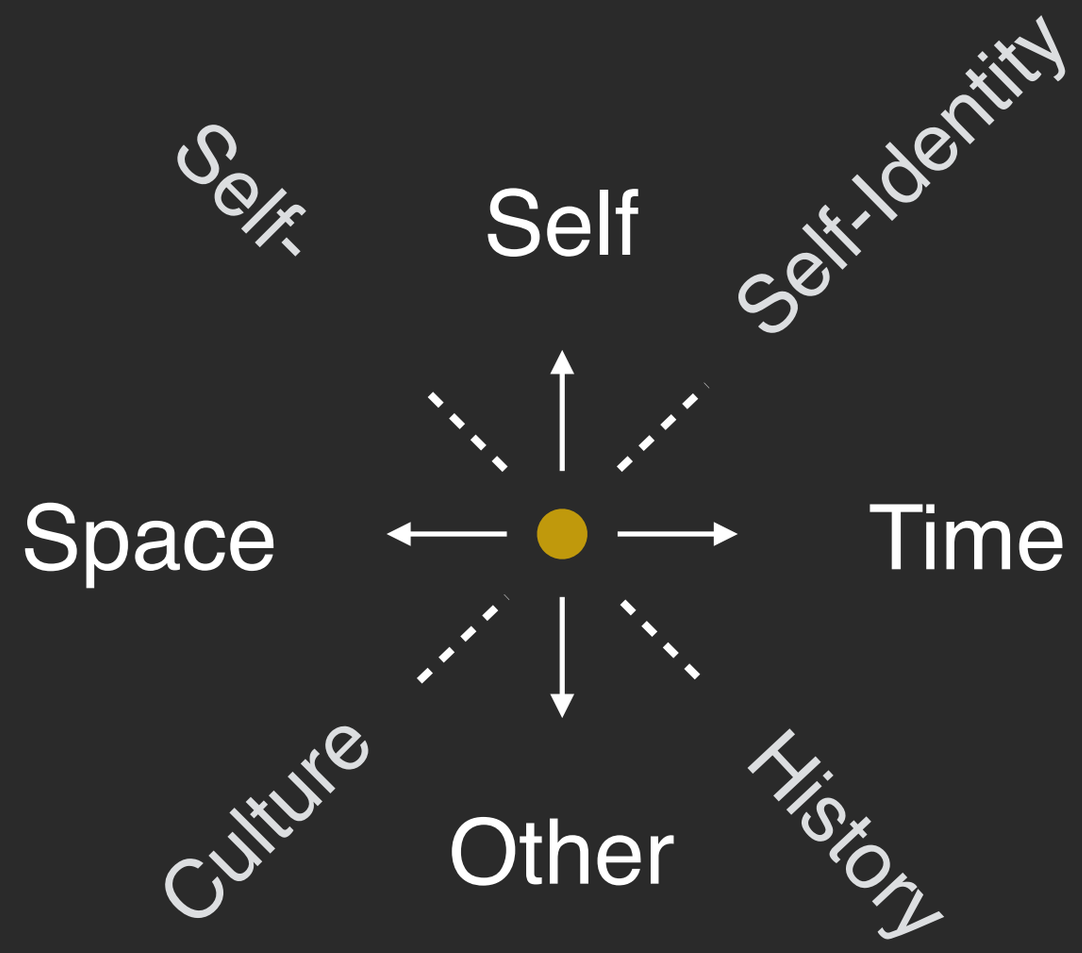

Self
Space
Time
Other


Self
Space
Time
Other


Self-
Self
Self-Identity
Space
Time
Culture
Other
History

Creation &
Publishing

Consumption &
Discovery



Social Media

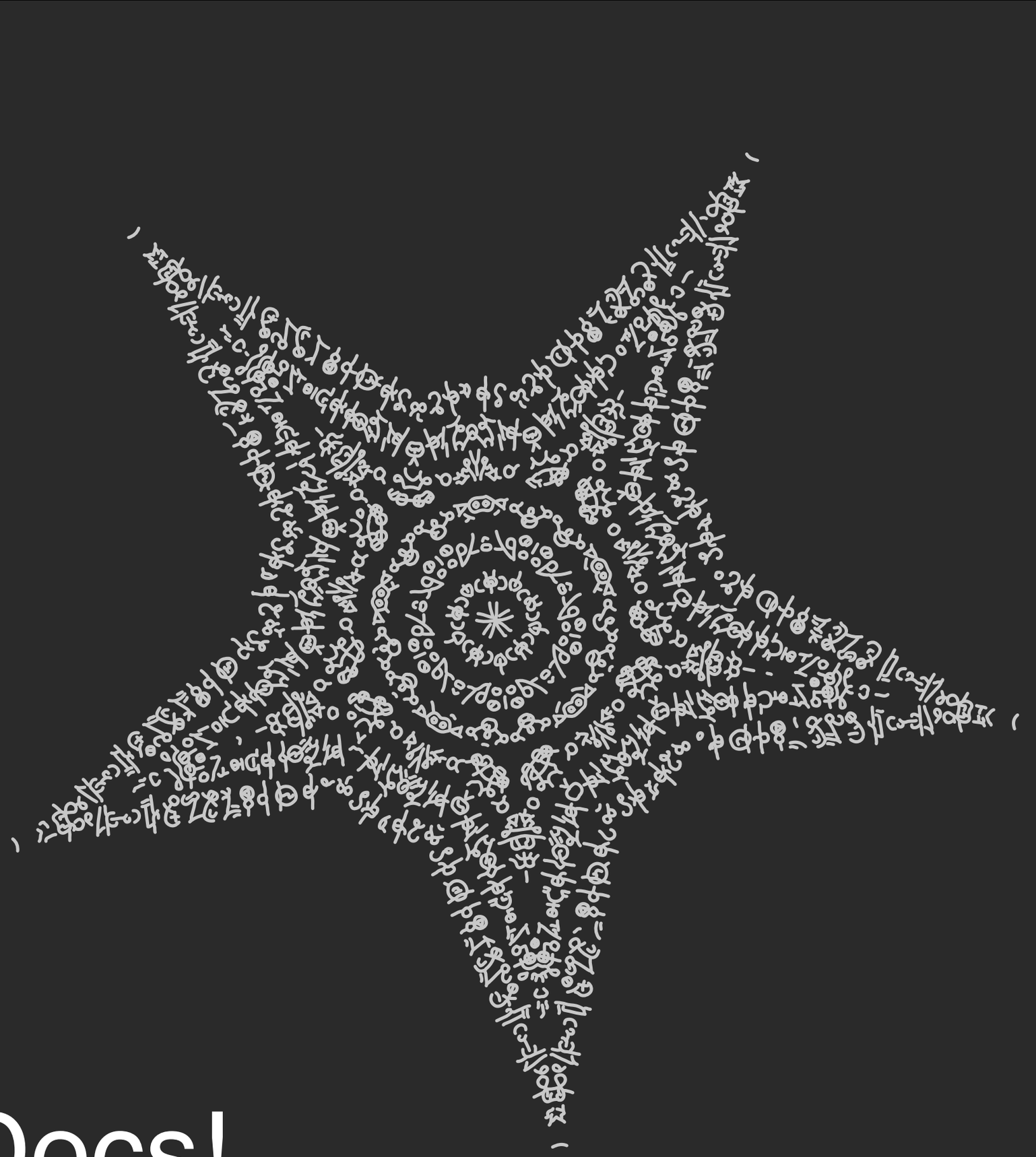
Research &
Information

Self

Other

?

Wikipedia



Write the Docs!

Information is Powerful

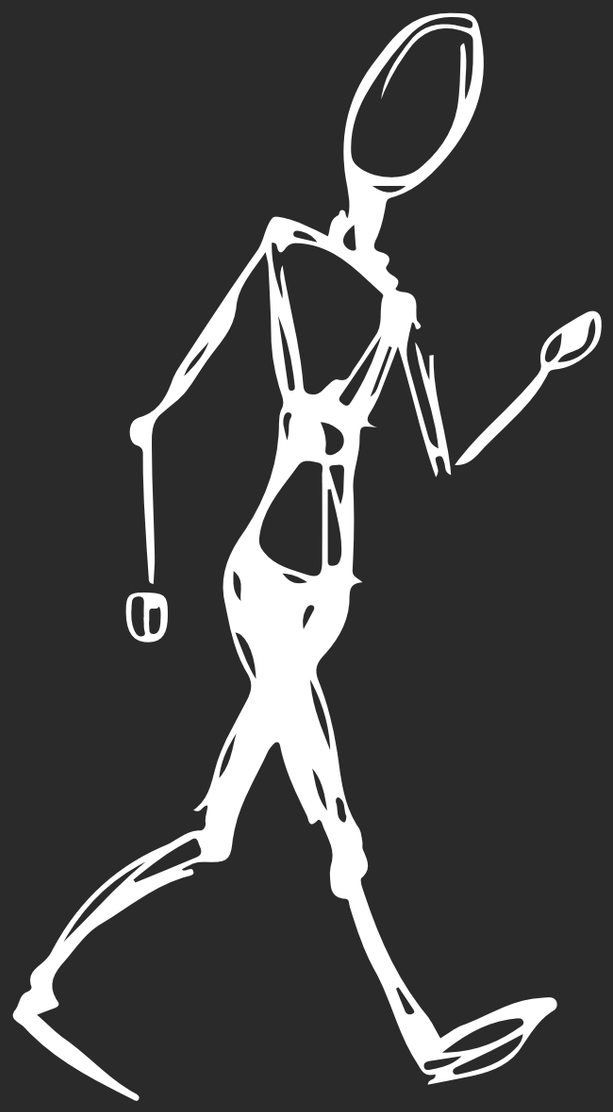
- Information is powerful — every day we see it transform the world around us.
- Documentation doesn't always have to be about a software workflow or open source project — it can be used to develop and convey ideas much larger than yourself.

Documentation at Scale

- Information architecture is a powerful tool for developing ideas over time and space.
- It enables us to evolve and distill information at a much larger scale than a single person or team could ever achieve on their own.

Documentation at Scale

- The Open Source Software style of development works because it allows people to create something larger than themselves.
- Why are we only doing this with tech?
- We should be doing this with everything!



For the first time in Human History!

We have the technology.

The Tools

The Tools Are Ready!

- Read the Docs + GitHub Pull Requests
- Instant version-controlled documentation that anyone in the world can contribute to.
- The software development community seems to be a testbed for technology that will soon be mainstream. This has to be one of them.

Python-Guide.org

- Python-Guide.org uses this workflow.
- Continually updated by 156 contributors.
- 50,000+ people view it every month.
- Originally outlined and continually curated by myself. Just like an open source project.

Bigger Than It Seems

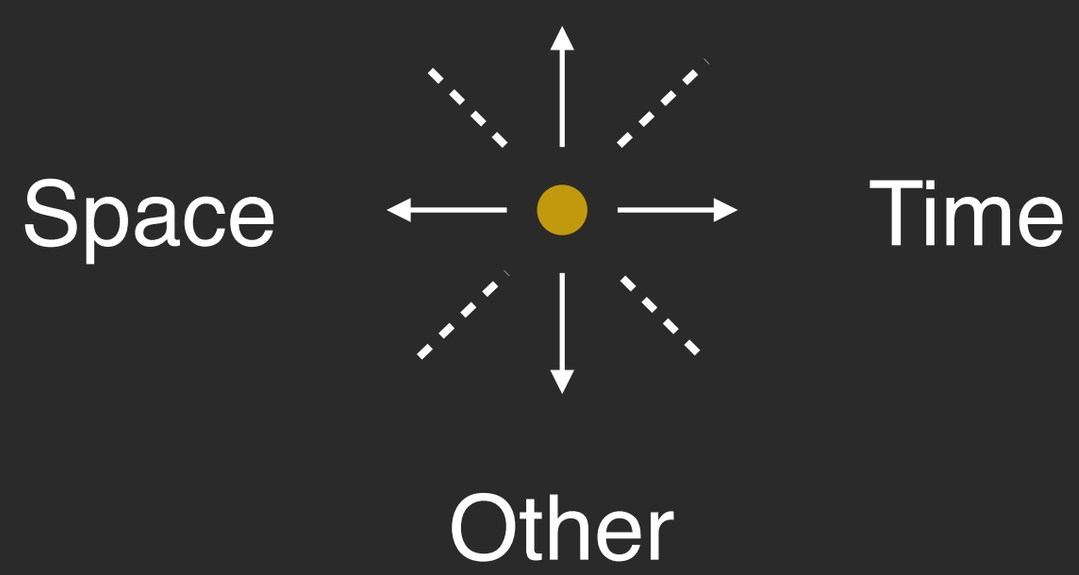
- A book, written by 156 people from all over the world, updated daily, transparently curated by an author, requiring zero time to publish, and available globally within seconds?
- This sounds simple to us. It's revolutionary.

Go, Write Docs!

- Go and make some repos that contain only prose, no code. Start building some ideas with others — or just yourself.
- Topics need not be technical — build docs on philosophy, humanities, literature, arts, sciences, poetry — everything!

Go, Build Tools!

- There needs to be more competition!
- ReStructuredText is amazing, but quite unapproachable for most people.
- The same applies to Git and GitHub.
- There's much work to be done!


Self
Space
Time
Other

Other Considerations

Internationalization

"Be Cordial" Policy

Contribution Guide

Persistent Archiving

Reducing Bus-Factor

Fast Revision Merging

The world is made language.

Become the author.

Write the Docs.