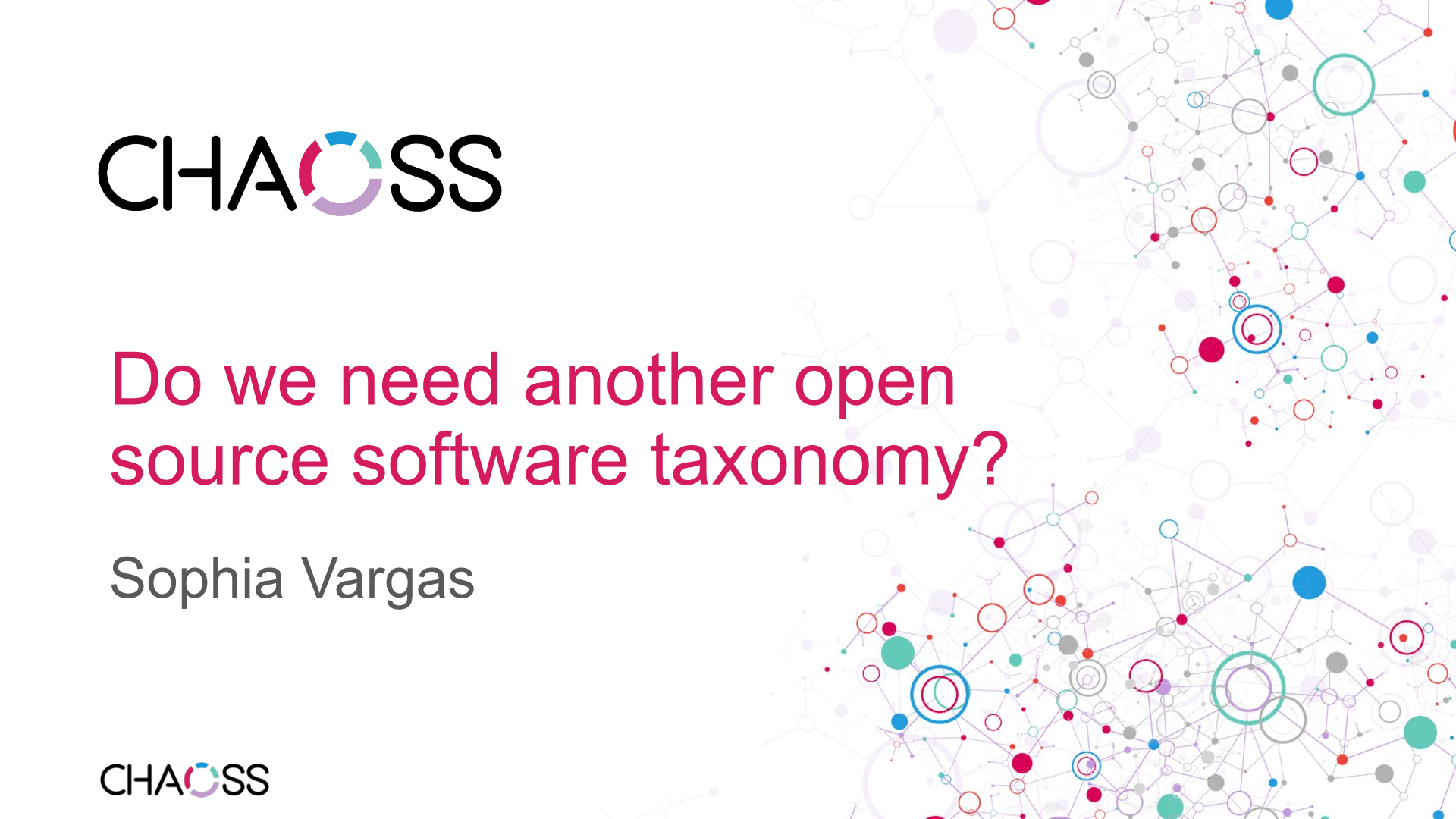


Do we need another open source software taxonomy?

Sophia Vargas

CHAOSS

About me

Market researcher

Consultant

Analyst

Program Manager

CHAOSS

How do OSS project characteristics influence [hypothesis]?

CHAOSS

What taxonomies already exist?

Standard Industrial Classification (SIC) (est. 1937)

Computer & office Equipment
Electronic Computers
Computer Storage Devices
Computer Terminals
Computer Communications Equipment
Computer Peripheral Equipment, NEC

…Hardware

Source: en.wikipedia.org/wiki/Standard_Industrial_Classification

What taxonomies already exist?

Standard Industrial Classification (SIC) (est. 1937)

Services-Computer Programming, Data Processing, Etc.
Services-Computer Programming Services
Services-Prepackaged Software
Services-Computer Integrated Systems Design
Services-Computer Processing & Data Preparation
Services-Computer Rental & Leasing

…Software!

Source: en.wikipedia.org/wiki/Standard_Industrial_Classification

What taxonomies already exist?

The North American Industry Classification System (NAICS) (est. 1997)

…Software?

Professional, Scientific, and Technical Services

Source: en.wikipedia.org/wiki/Standard_Industrial_Classification

CHAOSS

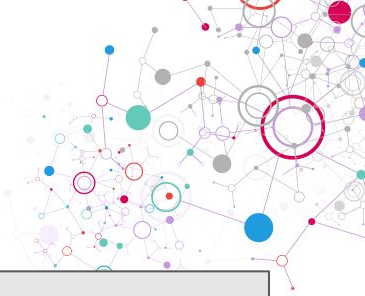
Where to start? What taxonomies already exist?

the **United Nations Standard Products and Services Code** (**UNSPSC**) (last release 2023)

- Business, Communication & Technology Equipment & Supplies
  - Information Technology Broadcasting and Telecommunications
  - Office Equipment and Accessories and Supplies
  - Printing and Photographic and Audio and Visual Equipment and Supplies
  - Published Products

Source: en.wikipedia.org/wiki/UNSPSC

What does the internet say?



Hardware

Software

- Application
- System

Other: Protocols, frameworks, standards, templates, etc

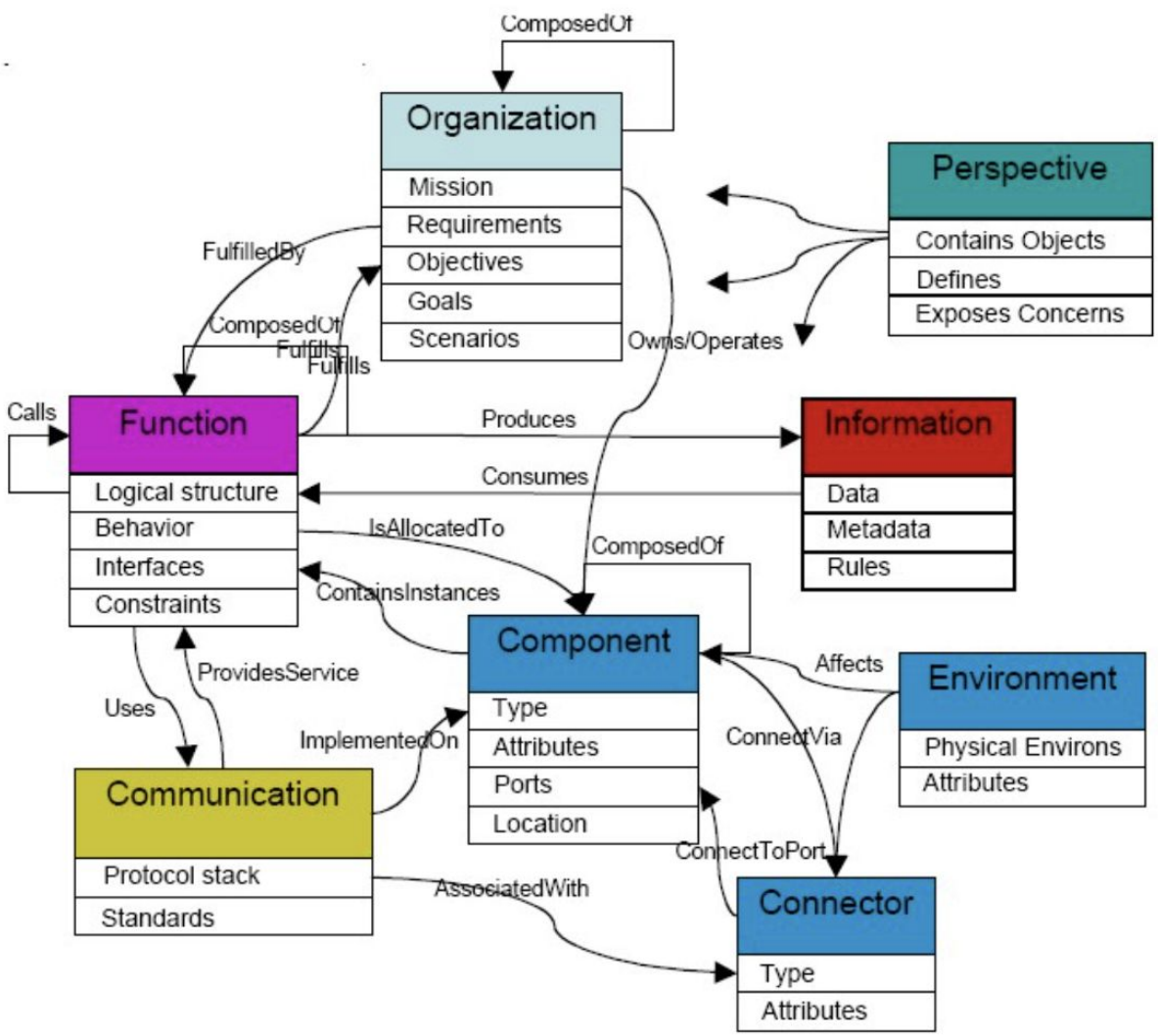
End user, developer?

Operating systems, drivers, firmware, language processors, utilities…

CHAOSS

Ontology engineering

Source: en.wikipedia.org/wiki/Ontology_engineering#



CHAOSS

Taxonomy for taxonomies?!

Functional

Organizational

Context driven

CHAOSS

Functional?

- Framework
- Language
- Library
- Database
- Utility
- Operating system

[missing: Infrastructure?]

Source: Internal project categorization attempt (2000)

Organizational?

System

- Environment
  - Server
  - Runtime
  - Operating system
  - Version control
- Network / Communication
  - API
  - Traffic controller
  - Transport
- Utility / function
- Language processing

Application

- Data / Information
  - Syntax transformation
  - Filesystem
  - File manipulation
  - Data structure
  - Data (content, metadata, rules, etc)
- Observability
  - Log management
  - Monitoring
  - Prober
  - Syntax analysis
  - Telemetry /Tracing
- Security
  - Identity and access management
  - Key management
- Q/A
  - Testing
  - Readability
  - Compliance

Source: Internal project categorization attempt (2024)

Mixture - (data) context?

1. **Science and medicine** (healthcare, biotech, life sciences, academic research, etc)
2. **Programming languages and supporting tools** (libraries, package managers, testing tools, etc)
3. **Development tools** (version control systems, CICD tools, text editors, issue managers, q/a tools, etc)
4. **Web** (tools and frameworks)
5. **Infrastructure and cloud** (hardware, software defined infrastructure, cloud native tooling, orchestration and automation, etc)
6. **Operating systems**
7. **Media** (graphics, video, audio, VR, streaming, gaming, content management, etc)
8. **End user applications**
9. **Data** (databases, analytics, visualization, AI/ML, etc)
10. **Security** (tools and frameworks)
11. **Social and communications** (Blog, chat, forums, wikis, etc)
12. **Other - please specify**

CHAOSS

Source: Survey of OSS projects and organizations (2022)

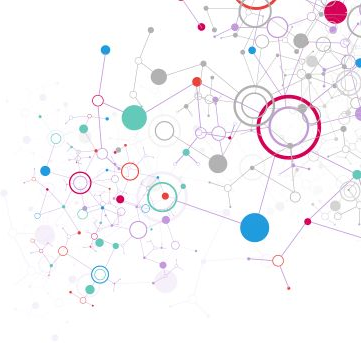
In practice: person context

- User / use case
  - Infrastructure and ops (System)
  - Developer (backend, front end, full stack)
  - End user (Application)
- Contributor
  - Participation style
  - Level of contribution

Who engages with the project? How do they engage?

CHAOSS

Stackoverflow survey (2024)



Developers (context) engaging with tools (functional)

- Programming, scripting, and markup languages (JS, HTML/CSS, PY...)
- Databases (PostgreSQL, MySQL, SQlite...)
- Cloud platforms (AWS, Azure, GCP...)
- Web frameworks and technologies (Node.js, React, jQuery...)
- Embedded technologies (Raspberry Pi, Arduino, GNU, CMake...)
- Other frameworks and libraries (.NET, NumPy, Pandas...)
- AI tools

Source: survey.stackoverflow.co/2024/

How can we find records of non-code contribution? [Young, et al]

- **Platform-attributed contributors**: visible on platforms like GitHub
- **Contributors identified by automated tools**: identifiable via platform records, events, APIs
- **Contributors identified with taxonomies**: credited through formatted files following some prescribed “standard.”
- **Contributors identified by ad hoc methods**: identified by parsing non-standardized data sources- websites (e.g., a board of directors), in text files, documentation or the license files of projects, etc.

CHAOSS

Source: Which contributions count? Analysis of attribution in open source (2021) doi.org/10.48550/arXiv.2103.11007

How can we find records of non-code contribution? [Young, et al]

TABLE I
COARSE-GRAINING OF THE ALL CONTRIBUTORS TAXONOMY

Coarse contribution	AC contribution[2]
Artifacts	`a11y`, `code`, `data`, `doc`, `design`, `plugin` `tool`, `translation`, `tests`, `userTesting`
Education & Outreach	`audio`, `blog`, `content`, `example` `eventOrganizing`, `mentoring`, `question` `talk`, `tutorial`, `video`
Lead	`business`, `financial`, `fundingFinding` `ideas`, `projectManagement`, `research`
Maintenance	`bugs`, `maintenance`, `review`
Support	`infra`, `platform`, `security`

CHAOSS

Source: Which contributions count? Analysis of attribution in open source (2021) doi.org/10.48550/arXiv.2103.11007

In practice: community context

- Project lifecycle
- Community size
- Processes
- Technology usage

How do we engage with each other?

CHAOSS

Community context: Project status

- **Concept** – Minimal or no implementation has been done yet, or the repository is only intended to be a limited example, demo, or proof-of-concept.
- **WIP** – Initial development is in progress, but there has not yet been a stable, usable release suitable for the public.
- **Suspended** – Initial development has started, but there has not yet been a stable, usable release; work has been stopped for the time being but the author(s) intend on resuming work.
- **Abandoned** – Initial development has started, but there has not yet been a stable, usable release; the project has been abandoned and the author(s) do not intend on continuing development.
- **Active** – The project has reached a stable, usable state and is being actively developed.
- **Inactive** – The project has reached a stable, usable state but is no longer being actively developed; support/maintenance will be provided as time allows.
- **Unsupported** – The project has reached a stable, usable state but the author(s) have ceased all work on it. A new maintainer may be desired.
- **Moved** - The project has been moved to a new location, and the version at that location should be considered authoritative. This status should be accompanied by a new URL.

CHAOSS

Source: www.repostatus.org/

Open source as a socio-technical model [Scacchi]

Socio-technical interaction networks (STINs) are defined as:

"…people (including organizations), equipment, data, diverse resources (money, skill, status), documents and messages, legal arrangements and enforcement mechanisms, and resource flows. **The elements of a STIN are heterogeneous**. The network relationships between these elements include **social, economic, and political interactions**." [Kling 2003]

Applied to OSS: **"**formation and enactment of **complex software development processes performed by loosely coordinated software developers and contributors.**"

Source: Socio-Technical Interaction Networks in Free/Open Source Software Development Processes (2004) ics.uci.edu/~wscacchi/Papers/New/STIN-chapter.pdf; doi.org/10.1007/0-387-24262-7_1

Social model of open source [Ferraioli]

Project classifications by purpose

- Collaboration
- Demonstration
- Education
- Validation
- Facilitation
- Experimentation

Source: Social Model Of Open Source (2022)
juliaferraioli.com/blog/2022/social-model-oss

CHAOSS

Inherent challenges

- Subjective bias
- Built for purpose
- Discrete vs overlapping
- Constant evolution
- “Other”

CHAOSS

Proposal: orthogonal taxonomies

Include all that apply:

- Function
- Organization
- Context

"Organizations design systems that mirror their own communication structure"

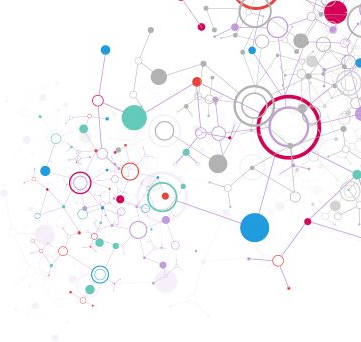
Melvin Conway, 1967
en.wikipedia.org/wiki/Conway%27s_law

CHAOSS

Can we find a better way to build taxonomies?

CHAOSS

What about crowdsourcing?



“**Open Demographics** is a recommended set of questions that anyone can use to ask community members about their demographics.”

Source: github.com/drnikko/open-demographics/

CHAOSS

Community Health Analytics Open Source Software

github.com/chaoss/wg-data-science/

CHAOSS

https://chaoss.community/
https://github.com/chaoss

What if we share our examples?

github.com/chaoss/wg-data-science/tree/main/dataset/taxonomies

Title	
Submitter:	(github handle)
Author(s):	(github handle? email?)
Author affiliation(s):	
Description:	(academic institution(s), company, funder, publisher, unknown, etc)
Data source(s) / data type(s):	(What was this used for? E.g. survey, analysis, publication, etc, How was this taxonomy created? (i.e. human, ML/AI, etc)
Suggested tags for this taxonomy:	(plain text separated by commas)
Link(s) to public resources that use this taxonomy:	(e.g. published articles, webpage, blog, etc)
License(s):	(if applicable)

Taxonomy: (if possible, paste as plain text table or list)

CHAOSS

Thank you!

linkedin.com/in/sophia-vargas-54608220

Github: sophia-IV

CHAOSS