

pygount

#112 Add chart for git tags and figures

Thomas Aglassinger, Python user group Graz, 2026-04-07

Pygount

- Counts source lines of code (SLOC), excluding comments and empty lines
- Supports 300+ programming languages and domain specific languages (DSL)
- Implemented in Python
- Based on pygments: <https://pygments.org/>
- Available from <https://pypi.org/project/pygount/>
- Open source (BSD license)

Example

```
uvx pygount --format=summary https://github.com/roskakori/pygount.git
```

Language	Files	%	Code	%	Comment	%
Python	20	48.8	2507	63.7	410	10.4
TOML	1	2.4	43	21.8	38	19.3
YAML	1	2.4	28	84.8	0	0.0
Bash	4	9.8	19	55.9	15	44.1
Markdown	12	29.3	0	0.0	385	40.7
Text only	1	2.4	0	0.0	22	81.5
__generated__	2	4.9	0	0.0	0	0.0
Sum	41	100.0	2597	50.2	870	16.8

Other features

- Additional options for `--format` to integrate with common CI/CD tools
 - CSV and JSON: for post-processing
 - sloccount: <https://dwheeler.com/sloccount/>
 - cloc-xml: <https://github.com/AIDanial/cloc>

- Run on local files

```
pygount --format=summary ~/projects/example
```

- Run with git tags

```
uvx pygount --format=summary https://github.com/roskakori/pygount.git/v1.5.1
```

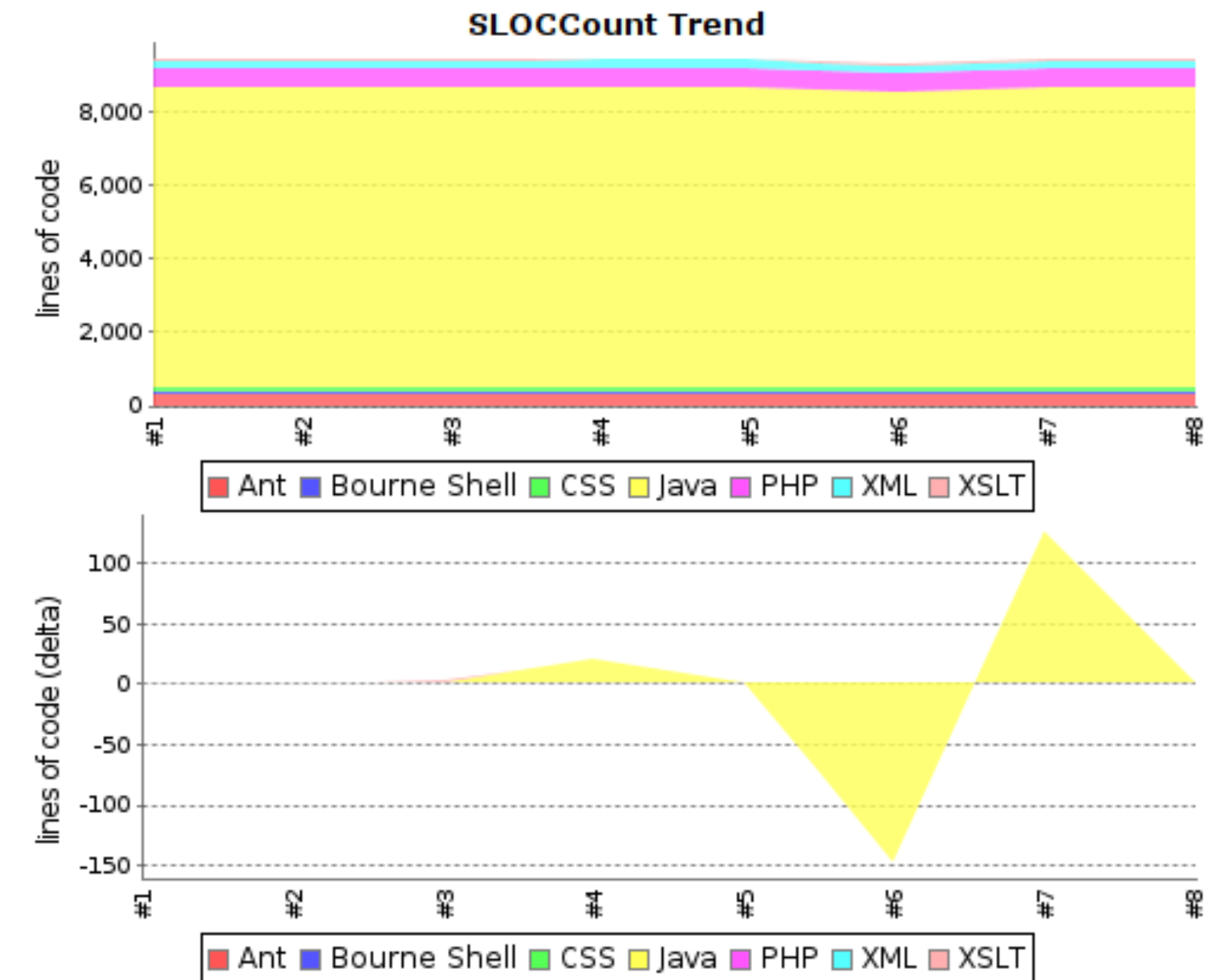
Value of SLOC

- Inane marketing: "more SLOC = more productivity"
- Cost: more SLOC = more dev time = more cost → COCOMO
<https://en.wikipedia.org/wiki/COCOMO>
(not supported by pygount but sloccount)
- Complexity: more SLOC = more complexity? → very uneven
 - Boiler plate code: many SLOC, low complexity
 - Dense algorithms: few SLOC, high complexity
- **Risk management:**
 - "every line of code is a liability"
 - Number of bugs per SLOC is pretty constant → more SLOC = more bugs
- Architectural impact: Can show impact of architectural decisions

#112 Add chart for git tags

SLOC chart

- Inspired by Jenkins sloccount plugin: <https://github.com/jenkinsci/sloccount-plugin>
- Shows how SLOC changed over time



git tags of pygount

```
% git tag --list
```

```
v0.1
```

```
v0.2
```

```
v0.3
```

```
...
```

```
v0.8
```

```
v0.9
```

```
v1.0.0
```

```
v1.1.0
```

```
v1.2.0
```

```
v1.2.1
```

```
v1.2.2
```

```
v1.2.3
```

```
...
```

```
v1.8.0
```

```
v2.0.0
```

```
v3.0.0
```

```
v3.1.0
```

```
v3.1.1
```

GitHub issue: #112 Add chart for git tags

<https://github.com/roskakori/pygount/issues/112>

Add chart for git tags #112

Edit

New issue



Open



roskakori opened on Jun 14, 2023 · edited by roskakori

Edits

Owner

Goals

- When a user specifies the command lines option `--format=graph`, the output is an image file showing how the SLOC changed over time for each git tag. ...
- If the command line options for the source path are anything but a single git URL, pygount fails with the error: `--graph requires a single git URL as source` ...
- The command line option `--colors` allows to specify the colors for the most popular languages as comma separated list of hex codes with white space and # being ignored. **i** For example: `#123456, 789abc` ...
 - The default is **? TBD**: Use up to 6 colors that look "nice" but also alternate between bright and dark to make them easy to distinguish even for people with color deficiencies. **i** For inspiration, take a look at [Atlassian's categorical colors](#).
 - If less than 2 colors are specified, the option is rejected with the message: `At least two colors must be specified.`
 - If there are more languages then colors, the colors rotate and are re-used.
 - If there are more colors than languages, only the required number of colors shows in the graph.
- The chart output image format is **? TBD**. HTML? SVG? ...
- In the graph, the languages are sorted in **? TBD**: Sort how? ...
- The command line option `--width` and `--height` specify the size of the chart in **? TBD**: pixels? dip? pt? ...
 - The defaults are 1024 resp. 768.
 - If the value is less than than 600 resp. 400 or more than 32,767, it is rejected with the message: `Value for option --xxx must be between XXX and 32767.` **i** PNG could use sizes up to $2^{31} - 1$.
- ? TBD**: General chart layout, percentage used, maximum length of language before cut off, font to use, ...

Assignees

No one - [Assign yourself](#)

Labels

enhancement

Projects

Open source projects

Status Ready

Milestone

No milestone

Relationships

None yet

Development

Code with agent mode

#112 add graph for git tags

roskakori/pygount

Notifications

Customize

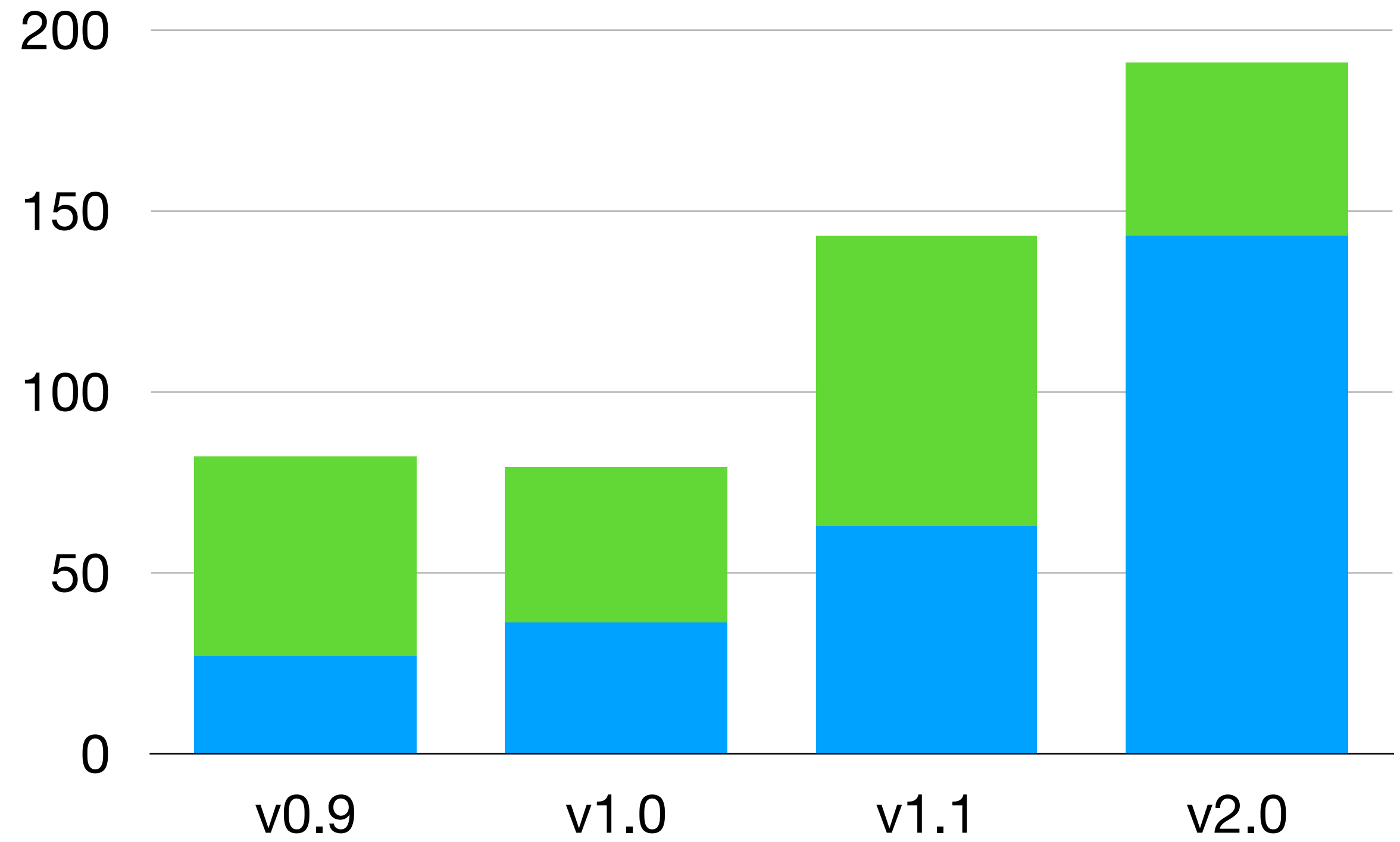
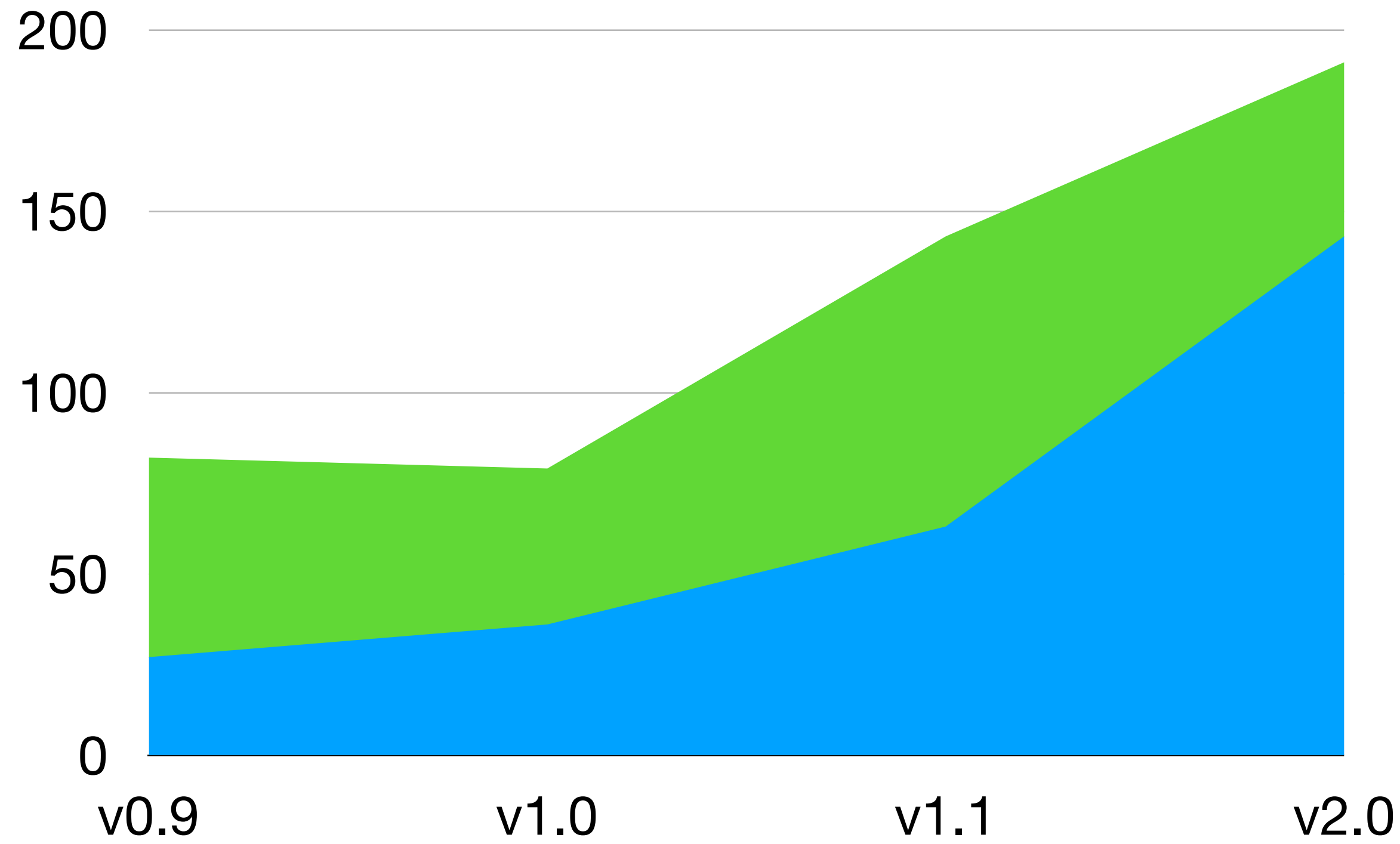
Unsubscribe

You're receiving notifications because you're subscribed to this thread.

Participants



Chart styles



**Implementation:
The agentic way**