Manage presentations like code with Pandoc and Github

Jim Baker

jim.baker@{python.org, rackspace.com}
@jimbaker on Twitter

Overview

This presentation uses a variety of tools:

- Github
- Markdown, including code fragments
- Pandoc for translating to Beamer presentations
- Can use LaTeXbut just for bits like math
- Beamer templates and themes

Markdown

Mark up of your presentation with Markdown is easy to use, as seen in the source for the previous slide:

```
Overview
========

This presentation uses a variety of tools:

* Github
* Markdown, including code fragments
* Pandoc for translating to Beamer presentations
* Can use \LaTeX but just for bits like math
* Beamer templates and themes
```

Including code fragments

Pygments is built in, as seen in this code snippet from the `itertools` recipes:

```
def random_permutation(iterable, r=None):
    "Random selection from itertools.permutations"
    pool = tuple(iterable)
    r = len(pool) if r is None else r
    return tuple(random.sample(pool, r))
```

- You do need to figure out how to make it fit!
- Good practice to use defaults in terms of sizing, vs trying to figure out how to squeeze in more text

Including code fragments

Source for the earlier fragment:

Pygments is built in, as seen in this code snippet from the ‘itertools‘ recipes:

````
````python
def random_permutation(iterable, r=None):
    "Random selection from itertools.permutations"
    pool = tuple(iterable)
    r = len(pool) if r is None else r
    return tuple(random.sample(pool, r))
````

...
````

Github

The usual invocations:

```
$ git add presentation-pipeline.md
$ git commit -m "Initial version"
$ git push
```

Pandoc

```
$ pandoc -f markdown -V theme:PaloAlto \
--write beamer --template lecture.beamer \
-o presentation-pipeline.pdf \
presentation-pipeline.md
```

- Or wrap in the script of your choice
- Add `-i` option for incremental bullets
- Read the Pandoc docs for more options on producing presentations

LaTeX

- Use arbitrary math
- Such as getting $O(n^2)$ just by writing `$O(n^2)$`
- Or use macro packages from stylesheets

Example: math

All even natural numbers E can be defined inductively using judgment forms, which connect a premise to a conclusion:

- example **axiom** - no premise is of course used

$$\frac{}{0 \in E}$$

- example **induction rule**, such that the inductively-defined set E is the **least set** that is **closed under** such induction rules:

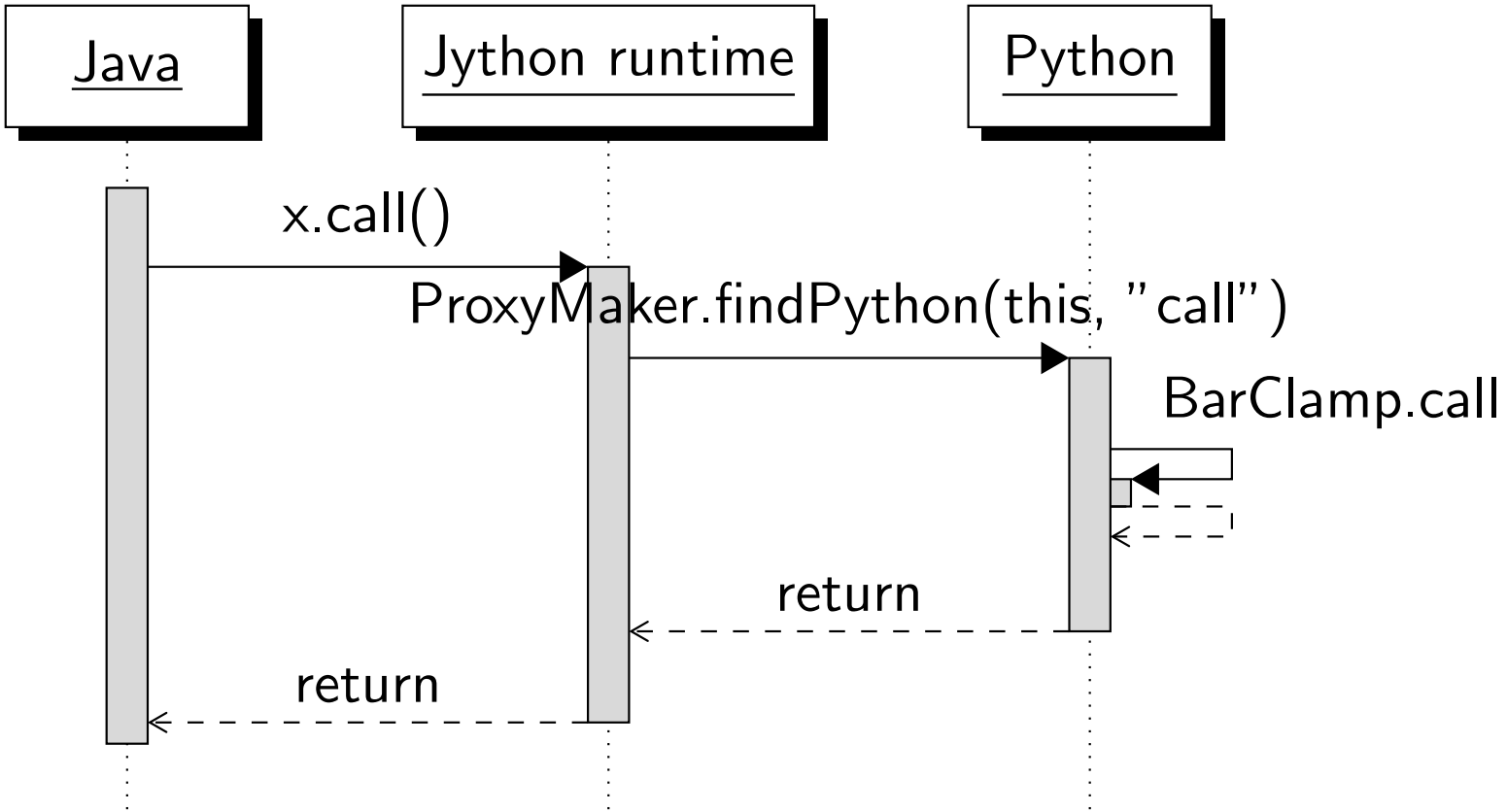
$$\frac{n \in E}{n + 2 \in E}$$

Example: math

Can use LaTeX macros:

```
\begin{mathpar}
\inferrule{\hspace{5pt}}
          {0 \in E}
\end{mathpar}
```

Example: sequence diagrams



Example: sequence diagrams

Produced by this source text in LaTeX:

```
\begin{sequencediagram}
\newthread{J}{Java}
\newinst[1]{R}{Jython runtime}
\newinst[1]{P}{Python}
\begin{call}{J}{x.call()}{R}{return}
\begin{call}{R}{ProxyMaker.findPython(this, "call")}{
\begin{callself}{P}{BarClamp.call}{}
\end{callself}
\end{call}
\end{call}
\end{sequencediagram}
```

Beamer templates

- Support for sequence diagrams, etc., requires importing the supporting Latex stylesheets
- Straightforward to import:

```
\usepackage{amssymb,amsmath,mathpartir}
\usepackage{longtable}
\usepackage{ifxetex,ifluatex}
\usepackage{fixltx2e} % provides \textsubscript
\usepackage{tikz}
\usepackage{tikz-qtree}
\usetikzlibrary{arrows,shadows}
\usepackage{pgf-umlsd}
\usepackage{smartdiagram}
\usesmartdiagramlibrary{additions}
```

Beamer themes

- `PaloAlto`
- `Berlin`
- and other city names

Fixing bugs

- Latex does produce somewhat inscrutable errors
- Especially when combined with another tool like Pandoc

Diagnosing Latex errors

- Generate Latex source
- Go to the produced error
- Maybe that works
- Or comment out

Markdown comments

Markdown simply uses standard HTML comments:

```
<!--- put your comments here
Such as commenting out this block
--->
```

You can include other HTML markup as well

Questions

Any questions?

Now or for later:

`jim.baker@{python.org, rackspace.com}`
`@jimbaker`

Talk available at github.com/jimbaker/talks