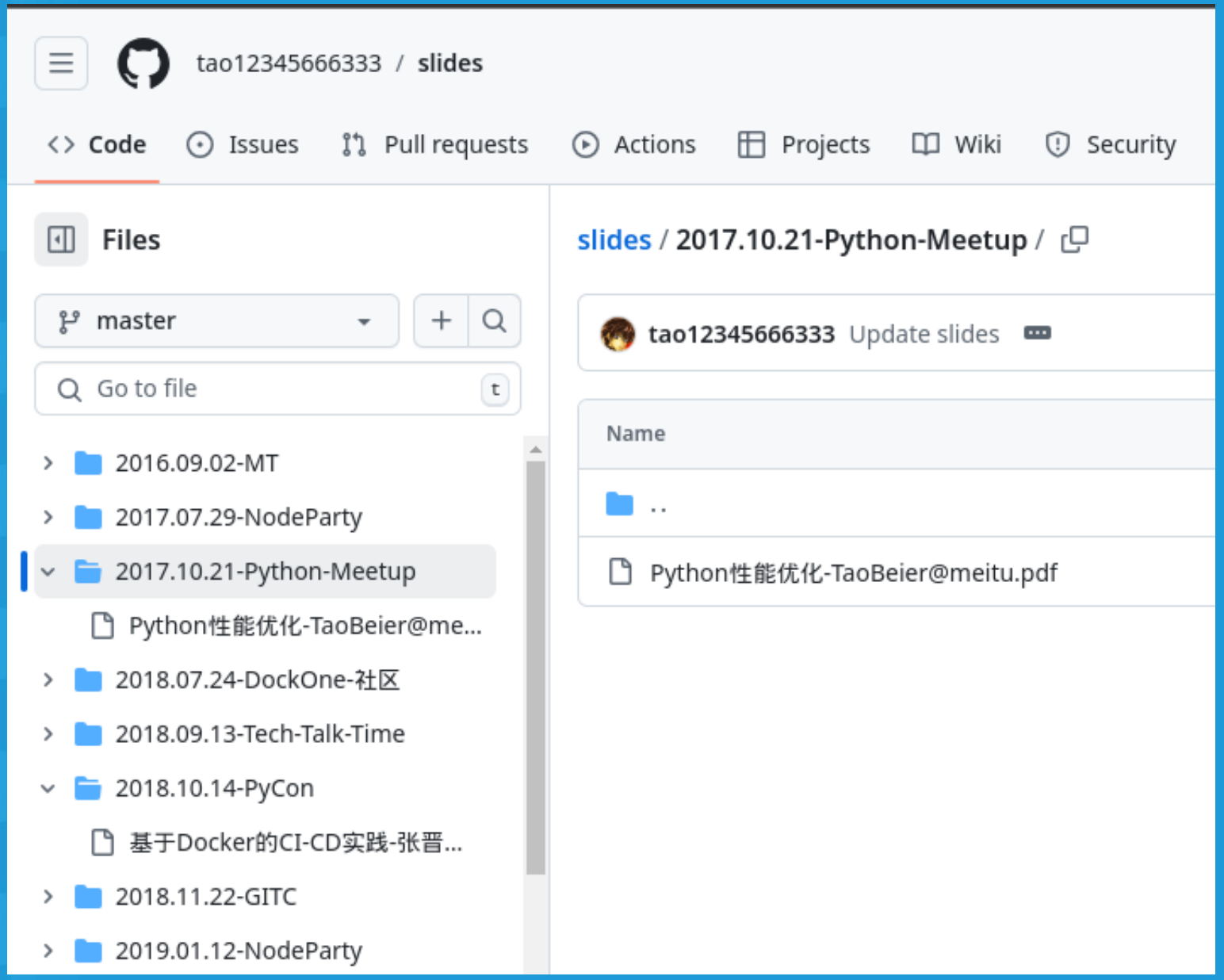
PyCon China 2024

For Good . For fun.

2024/11/23 中国 上海

深入Python debugger及其演进



https://github.com/tao12345666333/slides/

About Me

- 张晋涛 @ Kong Inc.
- CNCF Ambassador
- Kubernetes Ingress-NGINX maintainer
- Microsoft MVP
- 公众号：MoeLove

什么是Python debugger

- pdb - The Python Debugger
- 断点/执行
- 运行时信息
- IDE/Editor with debugger
- ...

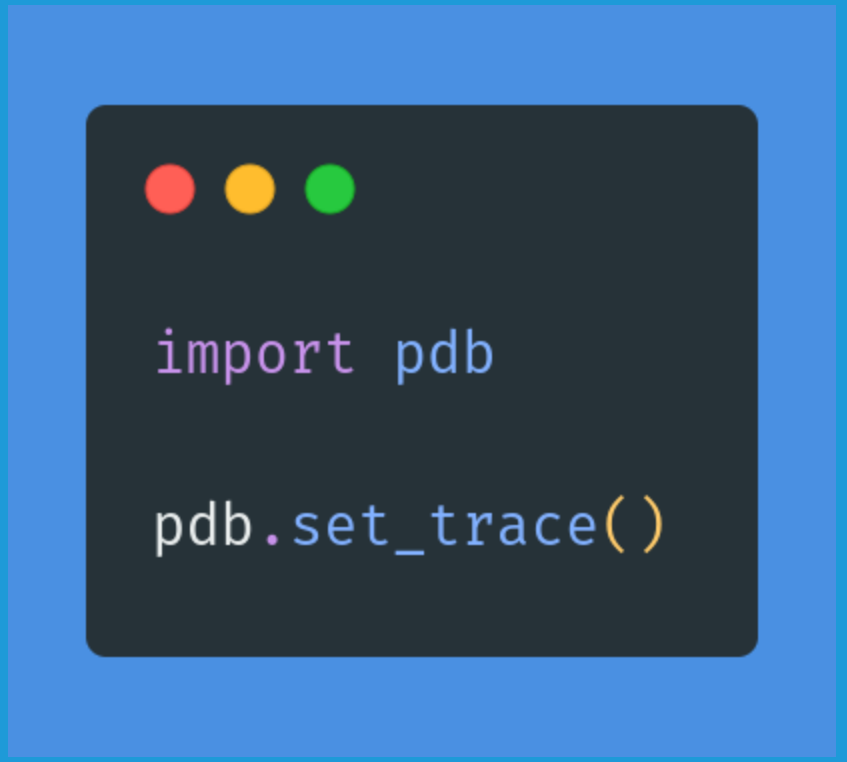
```python
#!/usr/bin/env python3
import os
import kong_pdk.pdk.kong as kong

Schema = (
    {"message": {"type": "string"}},
)

version = '0.1.0'
priority = 0

# This is an example plugin that add a header to the response

class Plugin(object):
    def __init__(self, config):
        self.config = config

    def access(self, kong: kong.kong):
        host, err = kong.request.get_header("host")
        if err:
            pass
        message = "hello"
        if 'message' in self.config:
            message = self.config['message']
        kong.response.set_header("x-hello-from-python", "Python says %s to %s" % (message, host))
        kong.response.set_header("x-python-pid", str(os.getpid()))


# add below section to allow this plugin optionally be running in a dedicated process
if __name__ == "__main__":
    from kong_pdk.cli import start_dedicated_server
    start_dedicated_server("py-hello", Plugin, version, priority, Schema)
```

- Kong Python Plugin
- 需要持续运行
- 不断处理请求

设置 debugger

- pdb.set_trace()

  Python v3.13
  立即执行,不换行



- breakpoint()

  Python v3.7
  PEP-553

深入 debugger

```
    def set_trace(self, frame=None):
        """Start debugging from frame.

        If frame is not specified, debugging starts from caller's frame.
        """
        sys.settrace(None)
        if frame is None:
            frame = sys._getframe().f_back
        self.reset()
        self.enterframe = frame
        while frame:
            frame.f_trace = self.trace_dispatch
            self.botframe = frame
            self.frame_trace_lines_opcodes[frame] = (frame.f_trace_lines, frame.f_trace_opcodes)
            # We need f_trace_lines == True for the debugger to work
            frame.f_trace_lines = True
            frame = frame.f_back
        self.set_stepinstr()
        sys.settrace(self.trace_dispatch)
```

PEP 669 – Low Impact Monitoring for CPython

- 与帧关联 vs 与代码对象关联
- 新增 15+ events
- 新增 sys.monitoring
- 很难兼容 bdb 接口

```
Help on module sys.monitoring in sys:

NAME
    sys.monitoring

FUNCTIONS
    free_tool_id(tool_id, /)

    get_events(tool_id, /)

    get_local_events(tool_id, code, /)

    get_tool(tool_id, /)

    register_callback(tool_id, event, func, /)

    restart_events()

    set_events(tool_id, event_set, /)

    set_local_events(tool_id, code, event_set, /)

    use_tool_id(tool_id, name, /)

DATA
    COVERAGE_ID = 1
    DEBUGGER_ID = 0
    DISABLE = <object object>
    MISSING = <object object>
    OPTIMIZER_ID = 5
    PROFILER_ID = 2
    events = namespace(PY_START=1, PY_RESUME=2, PY_RETURN=4, ...638
```

Thanks!

PYCHINA
Python 中国社区

PyCon China 2024

>> 2024/11/23 上海

>> For good . For fun.