Speeding up Python with Rust

Pablo Klijnjan
Some issues with Python

• Core Python library
  • Used by one of our web services and directly by other teams
  • Not as fast as required
  • Scientists consuming it are used to Python

• Decided to re-write in Rust and consume it as a Python extension
What’s a Rust?

- System level programming language developed at Mozilla
  - Fast
  - Secure
  - Productive
  - Flexible
  - Empowering
DUDE

JUST SHOW ME SOME CODE
An example

def is_blank_py_naive(filepath):
    with open(filepath, "r") as f:
        for c in f.read():
            if c != " ":
                return False
    return True
Input: 100MB file full of whitespaces

------- benchmark: 1 tests -------

<table>
<thead>
<tr>
<th>Name</th>
<th>Mean (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>test_is_blank_PY_naive</td>
<td>2,572.614</td>
</tr>
</tbody>
</table>
BLANK_RE = re.compile("^\s*$")

def is_blank_py(filepath):
    with open(filepath, "r") as f:
        return BLANK_RE.match(f.read()) is not None
<table>
<thead>
<tr>
<th>Name</th>
<th>Mean (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>test_is_blank_PY</td>
<td>550.4425</td>
</tr>
<tr>
<td>test_is_blank_PY_naive</td>
<td>2,597.0330</td>
</tr>
</tbody>
</table>
fn `is_blank`(filepath: String) -> bool {
    let mut file = File::open(filepath)?;
    let mut content = String::new();
    file.read_to_string(&mut content)?;

    content.chars().all(|c| c.is_whitespace())
}

Rust implementation
---------- benchmark: 3 tests ----------

<table>
<thead>
<tr>
<th>Name</th>
<th>Mean (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>test_is_blank_RUST_naive</td>
<td>116.2861</td>
</tr>
<tr>
<td>test_is_blank_PY</td>
<td>565.9652</td>
</tr>
<tr>
<td>test_is_blank_PY_naive</td>
<td>2,618.9210</td>
</tr>
</tbody>
</table>

Calling Rust from Python

• Define a function to expose in Rust
• Configure the Rust toolchain to build a dynamic lib
• Load the dynamic lib from Python
```rust
fn init_mod(py: Python, m: &PyModule) -> PyResult<()> {
    fn is_blank(filepath: String) -> PyResult<bool> {
        Ok(content.chars().all(|c| c.is_whitespace()))
    }
    Ok(())
}
```
[dependencies]
pyo3 = "0.2"

[lib]
name = "fast_is_blank"
crate-type = ["cdylib"]
from setuptools import setup
from setuptools_rust import RustExtension

setup(
    name='is_blank',
    version='0.1.0',
    packages=['is_blank'],
    rust_extensions=[RustExtension('is_blank._fast_is_blank', 'Cargo.toml')],
    install_requires=[],
    setup_requires=['setuptools-rust>=0.9.2'],
    include_package_data=True,
    zip_safe=False,
)
Links

• https://github.com/PyO3/pyo3
• https://github.com/PyO3/setuptools-rust
• https://github.com/Twistbioscience/pycon-2018-rust-example
Questions?
Thank you!