

Adder

John Stracke
Lightning talk at ILC 2009

Adder: Python with a Lisp

Adder : Python :: Clojure : Java

What kind of Lisp?

- Lisp-1
- Lexical scope
 - Not just Python's two-level scope
- Function names similar to Scheme
 - e.g., set!, begin, is-list?
- defmacro

Goal: integrate smoothly

- No impedance mismatch.
- An Adder function is a Python function, and vice versa.
- An Adder list is a Python list.
 - Not cons-based.
- `foo.bar.baz` works
 - `.bar.baz` is a function: `(.bar.baz foo)` is `foo.bar.baz`

Yes, it's been tried before

- There have been Lisps in/on Python before.
- Adder's advantages:
 - Technical background: Python has better metaprogramming than it used to.
 - Social background: Clojure has prepared the ground.
 - Performance: Adder JITs to Python bytecode.

Sample code

```
(import 'http.client)

(begin
  (let ((c (http.client.HTTPConnection)))
    (c.connect "www.itasoftware.com" 80)
  )
  (catch http.client.HTTPException e
    (print e)
  )
)
```

Status

- Code generation is stable.
 - Could do with some optimization.
- Supports functions, macros, generators, object access.
- Can't define classes yet.
- Can't use keyword arguments.

Releasing the code

- The code will be available this weekend at <http://www.thibault.org/adder/>
- GPL.

Questions?

Adder: John Stracke, ILC 2009