

Cython Status Report

Robert Bradshaw

SAGE Days 5: Oct 2, 2007



What is ~~Pyrex~~ ~~SageX~~ Cython?

Pyrex lets you write code that mixes Python and C data types any way you want, and compiles it into a C extension for Python.
— Greg Ewing (Author)

What is it?

- ▶ Pseudo-Python to C compiler
- ▶ Language extensions for statically declaring types
 - ▶ Potentially massive speedups
 - ▶ Integration with external libraries
- ▶ Python memory management and Python object \leftrightarrow c data type coercions done automatically.

Use in SAGE

- ▶ Compiled language necessary for a serious CAS
- ▶ Pyrex provides a consistent interface
- ▶ Pyrex provides an easy migration path
 - ▶ Python code easily becomes Pyrex code (even incrementally)
 - ▶ Python developers easily become Pyrex developers
- ▶ Pyrex provides direct and natural access to both the Python and C environments.

Why Cython?

- ▶ Hard to get Pyrex patches upstream

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...
- ▶ Release early, release often

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...
- ▶ Release early, release often
 - ▶ Since its inception, SAGE has had 153 official releases.

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...
- ▶ Release early, release often
 - ▶ Since its inception, SAGE has had 153 official releases.
 - ▶ In that time, Pyrex has had 6 (4 were bug-only releases)

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...
- ▶ Release early, release often
 - ▶ Since its inception, SAGE has had 153 official releases.
 - ▶ In that time, Pyrex has had 6 (4 were bug-only releases)
- ▶ We want more out of Pyrex than Greg has time to provide.

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...
- ▶ Release early, release often
 - ▶ Since its inception, SAGE has had 153 official releases.
 - ▶ In that time, Pyrex has had 6 (4 were bug-only releases)
- ▶ We want more out of Pyrex than Greg has time to provide.
 - ▶ Features: List comprehension, inplace operators, ...

Why Cython?

- ▶ Hard to get Pyrex patches upstream
 - ▶ Cross-directory imports
 - ▶ Introspection patches
 - ▶ Anything but bugfixes...
- ▶ Release early, release often
 - ▶ Since its inception, SAGE has had 153 official releases.
 - ▶ In that time, Pyrex has had 6 (4 were bug-only releases)
- ▶ We want more out of Pyrex than Greg has time to provide.
 - ▶ Features: List comprehension, inplace operators, ...
 - ▶ Optimization: Indexing, calling conventions, ...

Cython

- ▶ After 1 year, SageX has nothing specific to SAGE.
- ▶ Increasing interest by other people.

Cython



Web page: <http://www.cython.org>

Mercurial: <http://hg.cython.org>

Wiki: <http://wiki.cython.org>

Bugtracker: <https://launchpad.net/cython>

Mailing list: cython-dev@lists.berlios.de

Issues with Forking

- ▶ Politics

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package.

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package. *without any history.*

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package. *without any history.*
 - ▶ 0.9.4.1a → 0.9.5.1a was a huge pain!

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package. *without any history.*
 - ▶ 0.9.4.1a → 0.9.5.1a was a huge pain!
 - ▶ And that was being intimately familiar with most of the changes.

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package. *without any history.*
 - ▶ 0.9.4.1a → 0.9.5.1a was a huge pain!
 - ▶ And that was being intimately familiar with most of the changes.
- ▶ Division of Resources

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package. *without any history.*
 - ▶ 0.9.4.1a → 0.9.5.1a was a huge pain!
 - ▶ And that was being intimately familiar with most of the changes.
- ▶ Division of Resources
 - ▶ Essentially Greg works on Pyrex, so we're OK here.

Issues with Forking

- ▶ Politics
 - ▶ Greg Ewing mostly silent, but so far seems supportive of a parallel project.
- ▶ Merging
 - ▶ Pyrex released as a megolithic package. *without any history.*
 - ▶ 0.9.4.1a → 0.9.5.1a was a huge pain!
 - ▶ And that was being intimately familiar with most of the changes.
- ▶ Division of Resources
 - ▶ Essentially Greg works on Pyrex, so we're OK here.

Not so much of a new fork, as a merging of all other forks.

Benefits of Forking

New Developers

Benefits of Forking

New Developers

- ▶ Stefan Behnel

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ `lxml`

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ lxml
 - ▶ C API (cimport cdef functions)

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ lxml
 - ▶ C API (cimport cdef functions)
- ▶ Thomas Hunger

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ `lxml`
 - ▶ C API (`cimport cdef` functions)
- ▶ Thomas Hunger
 - ▶ `cdef` class-level bodies (including work on `classmethod`)

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ `lxml`
 - ▶ C API (`cimport cdef` functions)
- ▶ Thomas Hunger
 - ▶ `cdef` class-level bodies (including work on `classmethod`)
 - ▶ Other misc. fixes

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ `lxml`
 - ▶ C API (`cimport cdef` functions)
- ▶ Thomas Hunger
 - ▶ `cdef` class-level bodies (including work on `classmethod`)
 - ▶ Other misc. fixes
- ▶ Ulisses Furquim

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ `lxml`
 - ▶ C API (`cimport cdef` functions)
- ▶ Thomas Hunger
 - ▶ `cdef` class-level bodies (including work on `classmethod`)
 - ▶ Other misc. fixes
- ▶ Ulisses Furquim
 - ▶ Callback and the GIL

Benefits of Forking

New Developers

- ▶ Stefan Behnel
 - ▶ `lxml`
 - ▶ C API (`cimport cdef` functions)
- ▶ Thomas Hunger
 - ▶ `cdef` class-level bodies (including work on `classmethod`)
 - ▶ Other misc. fixes
- ▶ Ulisses Furquim
 - ▶ Callback and the GIL
- ▶ Huge influx of interest...

More New Stuff

- ▶ Better support for other Python versions

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling
 - ▶ METH_O and METH_NOARGS

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling
 - ▶ METH_O and METH_NOARGS
- ▶ Docstrings for special methods

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling
 - ▶ METH_O and METH_NOARGS
- ▶ Docstrings for special methods
- ▶ Fast tuple unpacking

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling
 - ▶ METH_O and METH_NOARGS
- ▶ Docstrings for special methods
- ▶ Fast tuple unpacking
- ▶ `cdef overrideable` (as of last night)

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling
 - ▶ `METH_O` and `METH_NOARGS`
- ▶ Docstrings for special methods
- ▶ Fast tuple unpacking
- ▶ `cdef` overrideable (as of last night)
 - ▶ No more `_add_c`, `_add_`, `_add_c_impl`, etc.

More New Stuff

- ▶ Better support for other Python versions
- ▶ Better support for non GCC
- ▶ Optimized method calling
 - ▶ `METH_O` and `METH_NOARGS`
- ▶ Docstrings for special methods
- ▶ Fast tuple unpacking
- ▶ `cdef overrideable` (as of last night)
 - ▶ No more `_add_c`, `_add_`, `_add_c_impl`, etc.
- ▶ Generators (work in progress)

Questions?

