



# POP-Python

## Dynamic Parallel Object Programming

Jonathan Stoppani – 20.10.2010

“Dynamically distributable objects for high performance computing on the grid

sounds cool...

IS IT?

# POP?

POP-CORNS

# The POP Model...

...allows to easily parallelize  
program execution

The POP Model...

...offers an OO API to remotely  
distributed objects

## The POP Model...

...remains transparent to the developer  
by offering a clean definition syntax

## The POP Model...

...takes care to allocate the needed resources,  
on the right host,  
at the right time

BUT . . .

C++





So...



...what's this all about?

“Provide a feasibility study of a POP  
model implementation for the Python  
programming language

# Why Python?



Never heard about decorators, significant whitespace, readability, keyword arguments, `import this`, duck typing, native lists, native tuples, native dictionaries, native sets, generators, closures, slice operator, `from __future__ import braces`, `for...else`, in-place value swapping, contexts, modules, packages, object descriptors, try...except...else...finally, operator overloading, properties, negative rounding, boolean and string multiplication, list/dictionary comprehensions, tuples packing/unpacking, metaclasses, (readable) ternary operator, lambda functions, map/reduce/filter builtins, integer base guessing, ...

**NO?**

This slide took 20 seconds of  
my 7 minutes only to display!

Python...

...provides a clear  
and readable syntax

Python...

...allows for rapid development  
and prototyping

Python...

...is highly dynamic

(interpreted, but can still be optimized with C/C++)

What do I have to do?

Tasks...



Analyze the application protocol

Tasks...



Analyze the application protocol

PAINT SOME "DIAGRAMME EN FLECHE"

Tasks...



Pick a tool for the job

# Tasks...



Define a syntax which suits the environment philosophy

Tasks...



Play the software architect

Tasks...



Play the software architect

UML, YEAH!

and

# Tasks...



Let someone else deal with it

Tasks...



Let someone else deal with it

THIS IS THE BEST PART... ;-)



“Dynamically distributable objects for high performance computing on the grid

sounds cool...

IT IS!

# Questions?

