

Java, Python, Zope and Indexing

Having Your Cake and Eating It

Chris Withers

chrisw@npltd.com

<http://www.zope.org/Members/chrisw>



Overview

- Java and Python Integration
- Indexing
 - ZCatalog
 - Lucene



New Information Paradigms (NIP)

- In Business 12 years
- Specialise in Knowledge & Content Management
- Customers include:
 - Most large Pharmaceutical companies
 - London Stock Exchange
 - Readers Digest



NIP's Technologies

- Wide range of skills including:
 - Zope Consulting & Hosting
 - J2EE and Oracle
 - Lotus Notes
- Operating Systems:
 - Windows
 - Solaris
 - Linux



Contacting NIP

- <http://www.nipltd.com>
 - For an overview
- <http://zope.nipltd.com>
 - For Zope specific stuff
- info@zope.nipltd.com
 - To contact by email



Java and Python

- Why use Java?
 - It's overly verbose
 - Not very dynamic
 - Painful Exception Handling
 - “Too” object oriented
- But...



Java and Python

- Why use Java?
 - Quicker Execution
 - Very Popular Language
 - More libraries
 - More robust
 - more testers
 - Better documentation
 - more authors around
 - Politically acceptable



But I want to use Python!

...so find a way to use Java and Python in the same environment.

- So what are the options?
 - Jython / JPython
 - Web Services
 - ...and other loose couplings
 - Java Python Environment



Jython / JPython

- Python implemented in Java instead of C
 - + Very politically acceptable
 - Can't use C extensions to Python
 - Not the “main branch” of Python development
- Status?



Loose Couplings

- Web Services
 - Shared Files
 - Low-level socket protocols
-
- + No restrictions on versions of languages or extensions to languages used.
 - + Easy to distribute applications over several machines
 - A lot more work for the developer
 - Inefficient communication between virtual machines

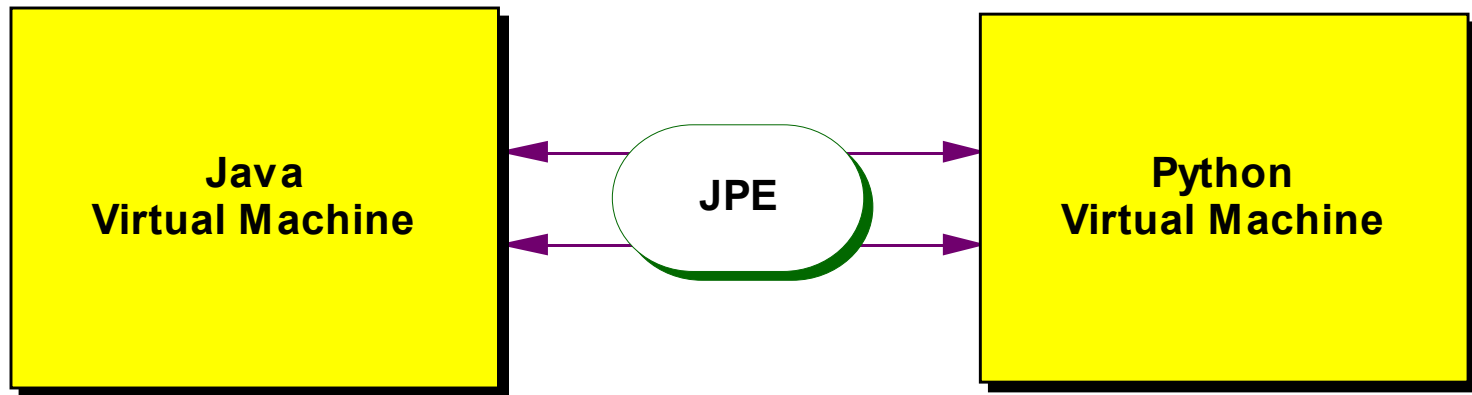


Java Python Environment (JPE)

- Low-level bridge between a Java virtual machine and a Python virtual machine
 - + Use almost any Java library from Python
 - + Use almost any Python library from Java
 - + Very Transparent
 - Difficult to Build, Install and find out about



How does JPE work?



- Bridge written mainly in Python and Java
- C extension to Python (wrapped in Python package)
- C extension to Java (wrapped in Java package)

So lets see it in action...

- Using Java from Python
- Using Python from Java



Using Java from Python

```
import java
if not java.isInitialized():
    java.initialize()

out = java.importClass( 'java.lang.System' ).out
out.println('Hello Python World from Java')
```

- How about a demo?



Using Python from Java

```
import python.PyModule;
import python.PyObject;
class HelloWorld
{
    static void main( String args[])
    {
        PyModule sys = new PyModule( "sys");
        PyObject stdout = (PyObject)sys.getattr(
            "stdout");
        stdout.callmethod( "write", new PyObject[]
        { PyObject.asPython( "Hello Java world from
        Python\n") });
    }
}
```

- How about a demo?



What are the problems?

- Needs the environment correctly set up
 - Python & Java versions important
 - PATH, CLASSPATH & PYTHONPATH important
- Difficult to build
 - See How-To
 - DON'T use *nmake install*!
- Performance
 - But only in recent versions!



Questions ?

<http://zope.nipltd.com/>



Indexing

- What do we mean by indexing?
 - Numbers
 - Dates
 - Text
 - Sorting in Relevance Ranking
- It's a HARD problem!
 - Don't let Google fool you...



What are the options?

- Commercial Solutions
 - Verity
 - Google boxes
 - \$\$\$ ☹
- ZCatalog
- Lucene



ZCatalog

- Solves generic indexing problem for Zope
- Stores information in ZODB
 - Participates in transaction framework ☺
 - Stores all old revisions ☹
- TextIndex has very limited functionality



Lucene

- Written originally by Doug Cutting
 - Xerox's Palo Alto Research Center (PARC)
 - Apple
 - Excite@Home
- Now part of the Apache Jakarta project
- Only tackles text indexing
- High Performance
- Fully Featured
 - Phrase matching
- Written in Java ☹️



Let's see some code...

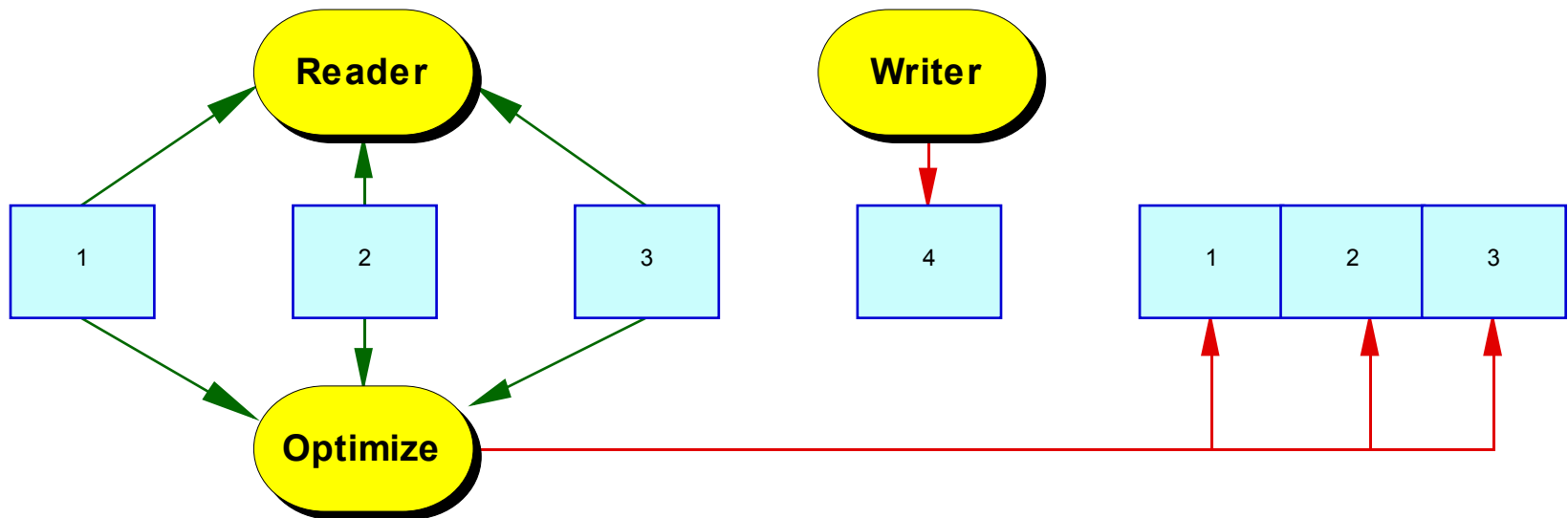
...written in Python!

- Indexing Files
- Searching Indexed Files



How does Lucene handle concurrency?

- File locks
- Never add to an existing index



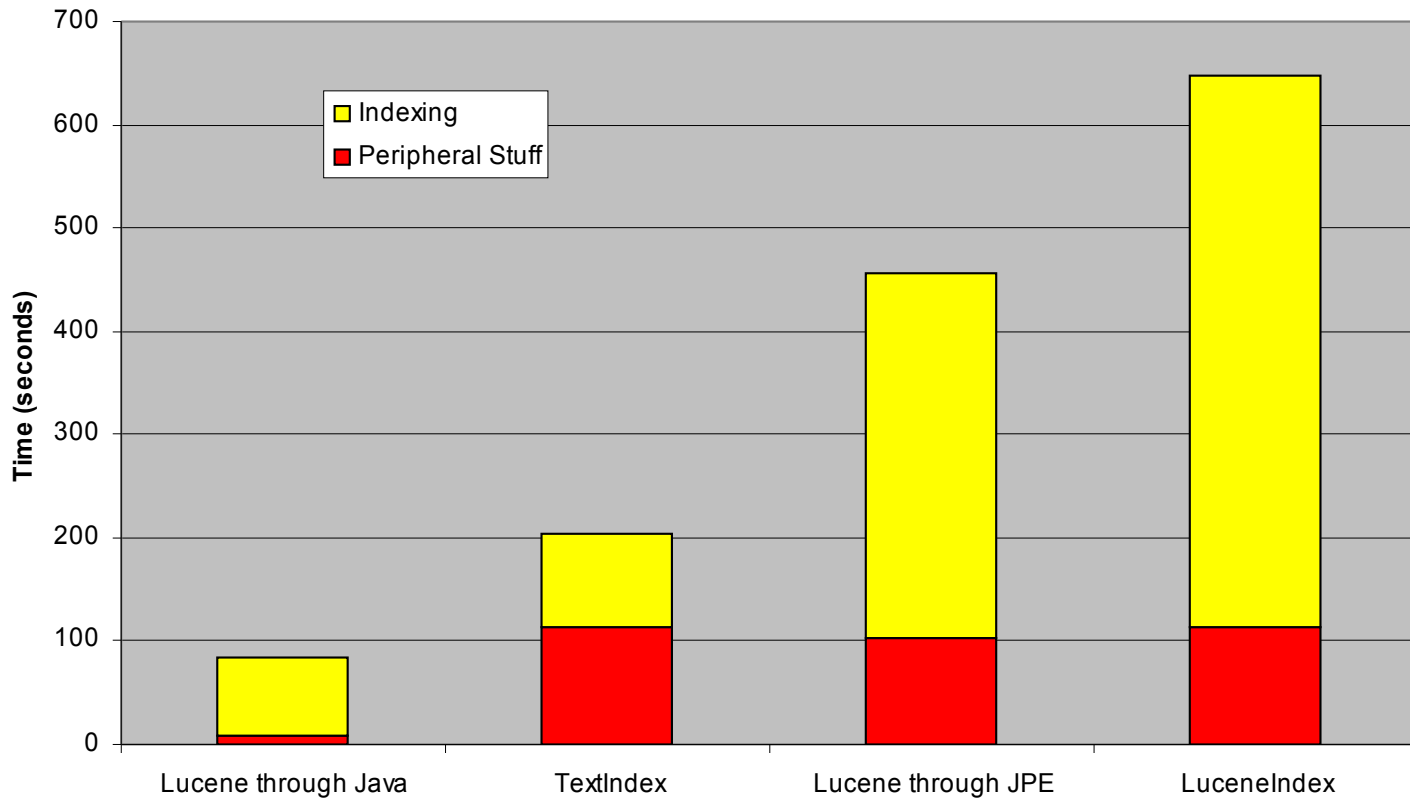
LuceneIndex

- A PluggableIndex for Zope 2's ZCatalog
- Really painful to implement ☹
 - PluggableIndexes are Clunky
 - Undocumented reliance on *id* attribute
 - Really hoping it'll be better in Zope 3...
- Lets have a look...



A Comparison

- 1000 Documents, Average length 5781 Bytes



Was that fair?

- Performance for BIG numbers of long documents
- TextIndex doesn't do phrase matching
 - Something which did took MUCH longer
- Lucene doesn't support undo
 - Do we care?
- LuceneIndex proved that the Lucene architecture and ZCatalog's architecture aren't very compatible ☹



Conclusions

You can have your cake and eat it...

...just slowly ☹️

...for now 😊



Where from here?

- Optimise JPE?
- CORBA?
- Re-implement Lucene in Python?
- TextIndexNG?



Questions ?

<http://zope.nipltd.com/>



Thankyou!

(PS: Swishdot is still on the way ;-)

(PPS: It was Steve A's birthday yesterday!)

