

ISIS-NBP developer's view

Rodrigo Dias Arruda Senra

My background

- BIREME's collaborator since 2006
- Software Architect and Developer
GPr Sistemas since 1996
- Computer Engineer and PhD student
IC/Unicamp
- **newbie/novato in ISIS world**



Who is in the audience ?

Who is in the audience ?

- ✓ I saw ISIS-NBP
yesterday's
presentation?



Who is in the audience ?

- ✓ I saw ISIS-NBP
yesterday's
presentation?
- ✓ ISIS Developers ?



Who is in the audience ?

- ✓ I saw ISIS-NBP yesterday's presentation?
- ✓ ISIS Developers ?
- ✓ Software Developers ?



Who is in the audience ?

- ✓ I saw ISIS-NBP
yesterday's
presentation?
- ✓ ISIS Developers ?
- ✓ Software Developers ?
- ✓ Software developers ?



Who is in the audience ?

- ✓ I saw ISIS-NBP
yesterday's
presentation?
- ✓ ISIS Developers ?
- ✓ Software Developers ?
- ✓ Software developers ?
- ✓ Screamers ! (Wenke?)



Who is in the audience ?

- ✓ I saw ISIS-NBP yesterday's presentation?
- ✓ ISIS Developers ?
- ✓ Software Developers ?
- ✓ Software developers ?
- ✓ Screamers ! (Wenke?)
- Not covered above!



Presentation Outline

Presentation Outline

- Defining ISIS-NBP

Presentation Outline

- Defining ISIS-NBP
- Developing **client applications** for ISIS-NBP

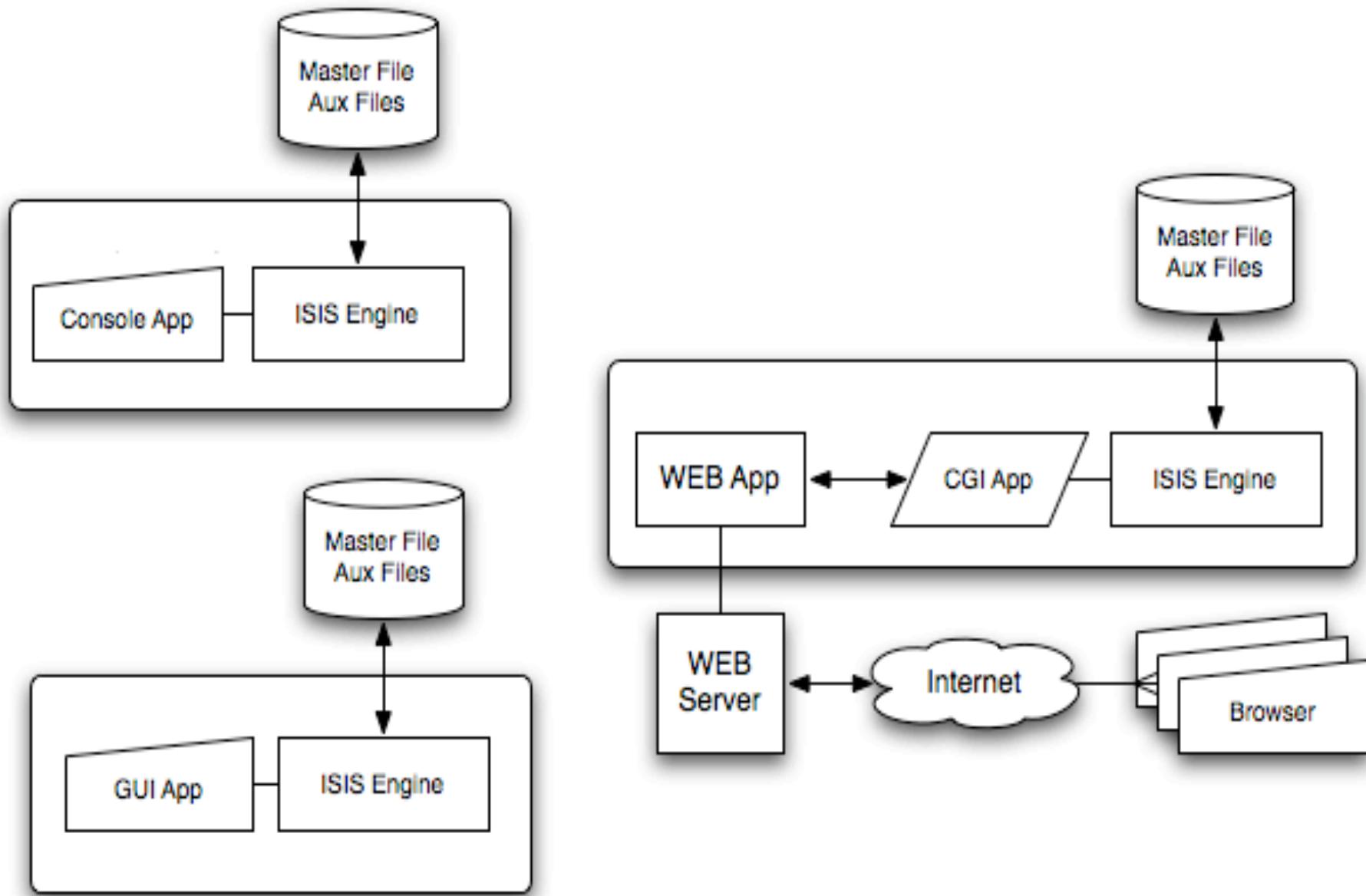
Presentation Outline

- Defining ISIS-NBP
- Developing **client applications** for ISIS-NBP
- Developing ISIS-NBP **itself**

Presentation Outline

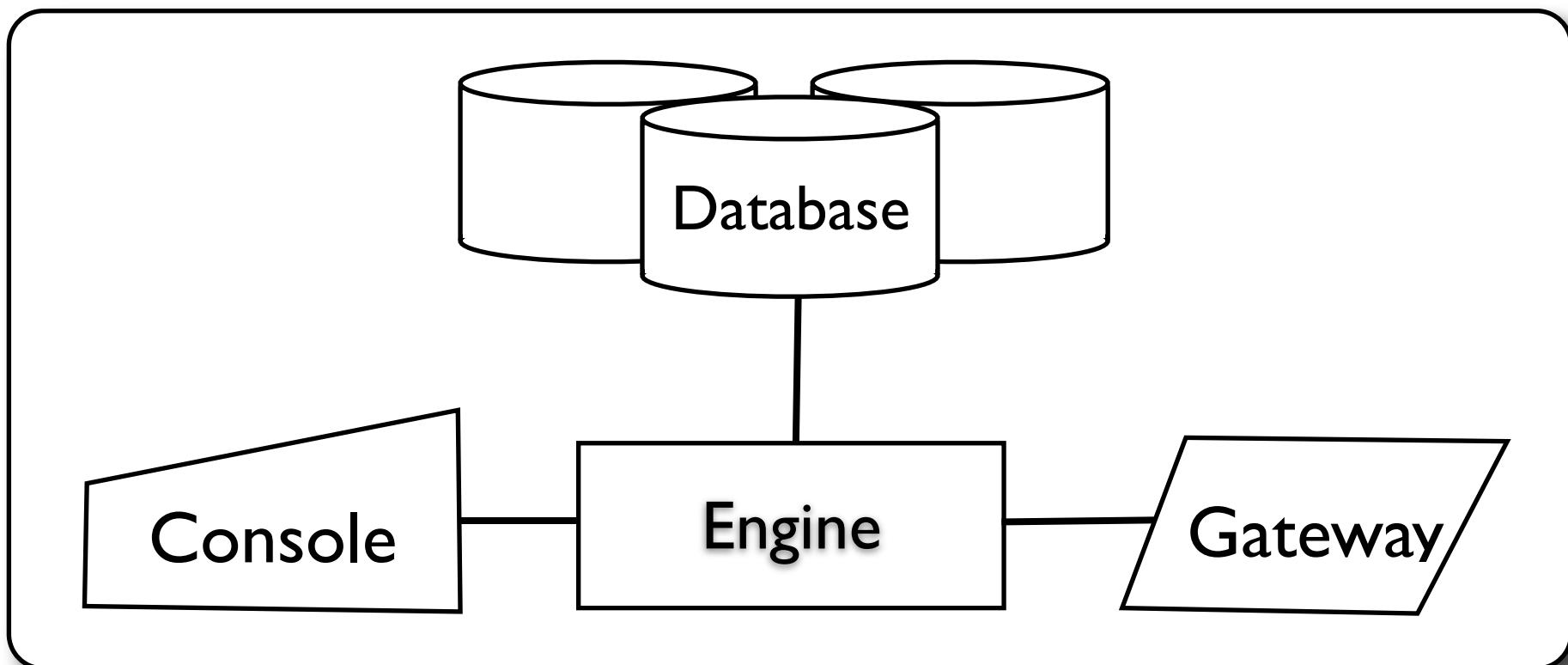
- Defining ISIS-NBP
- Developing **client applications** for ISIS-NBP
- Developing ISIS-NBP **itself**
- F.A.Q.: Why Python ?

ISIS Classic App Architecture

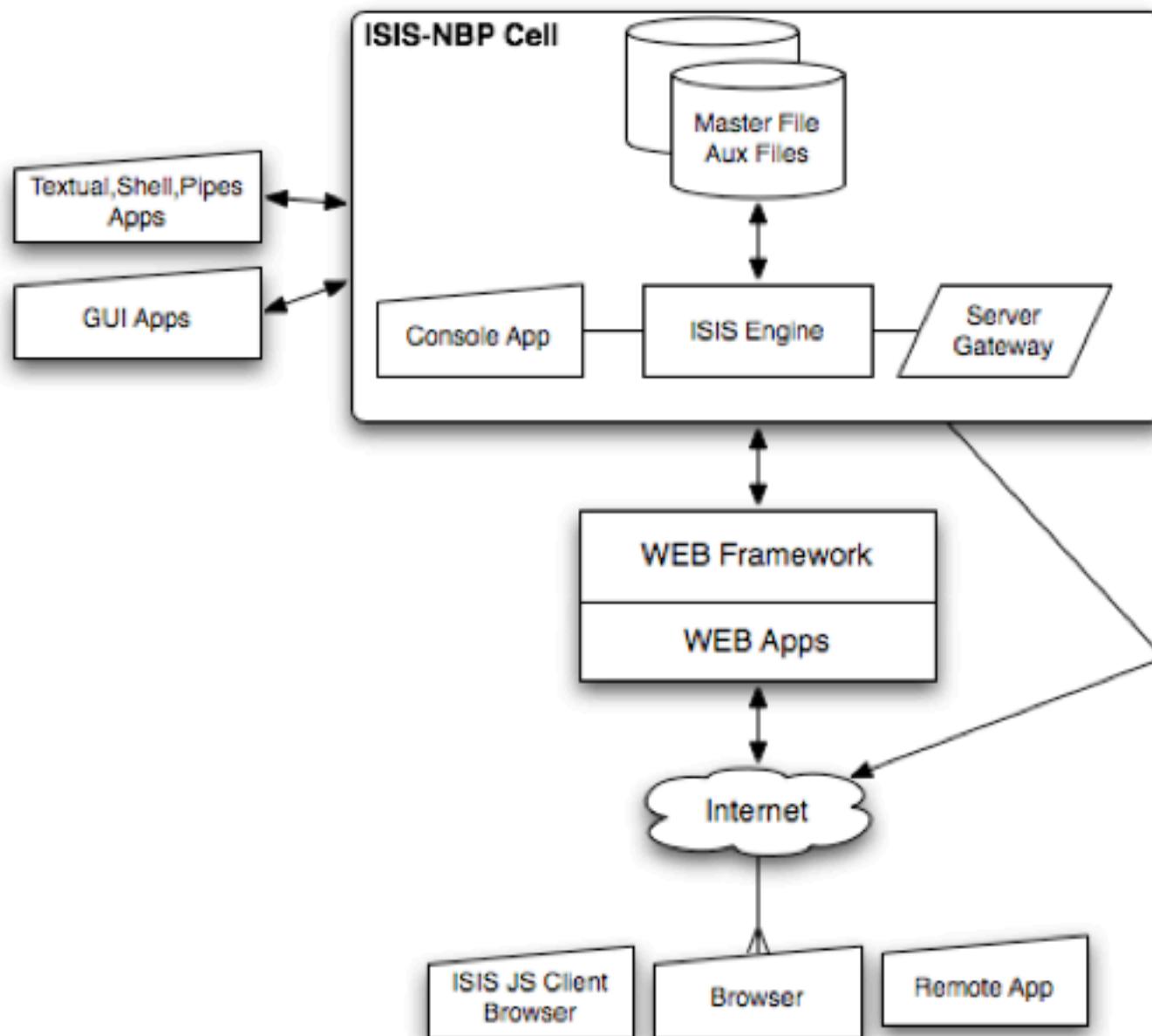


ISIS-NBP is...

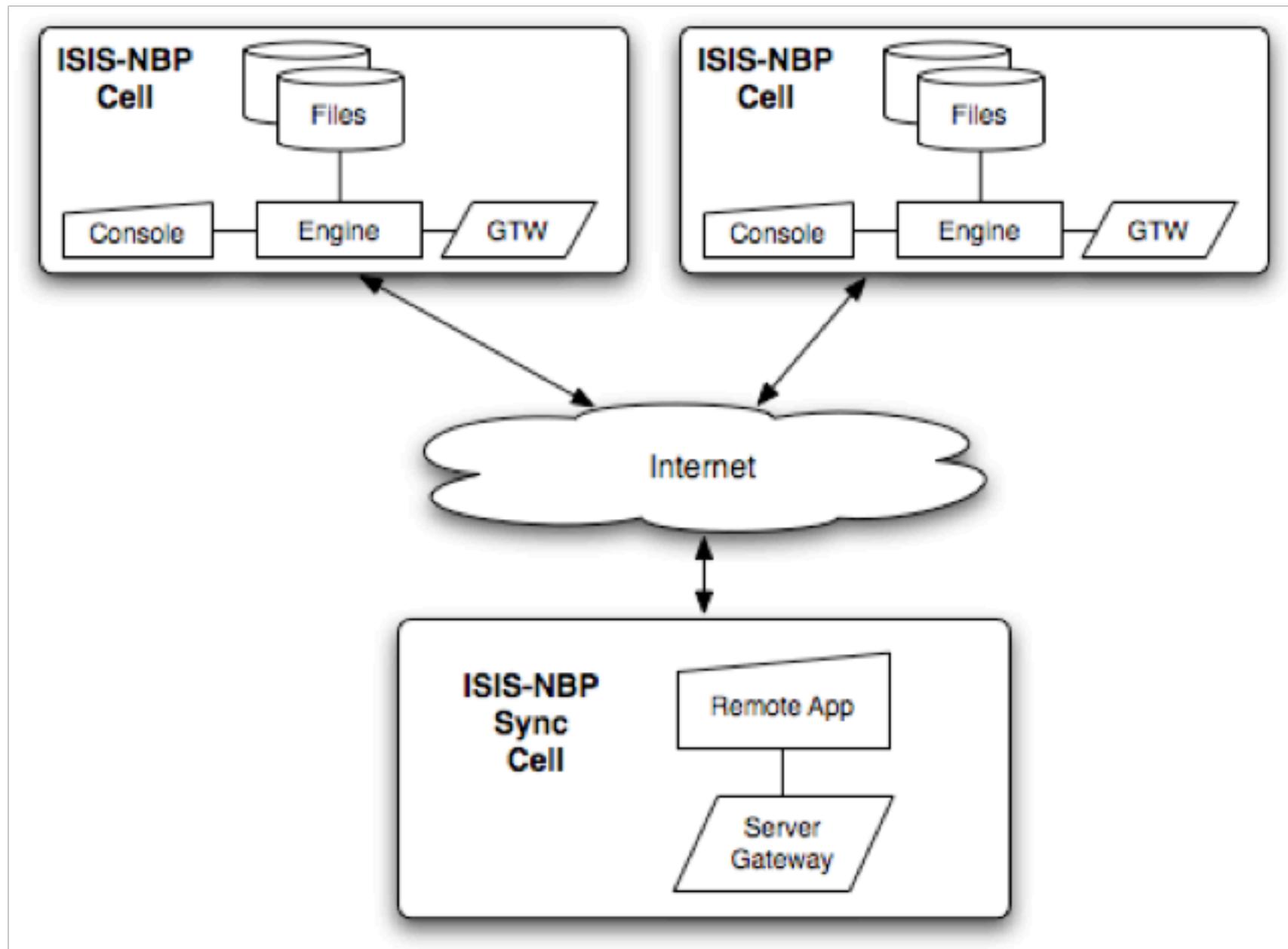
... a **platform** targeting the development of
interoperable, flexible, and cross-network
ISIS-based applications.



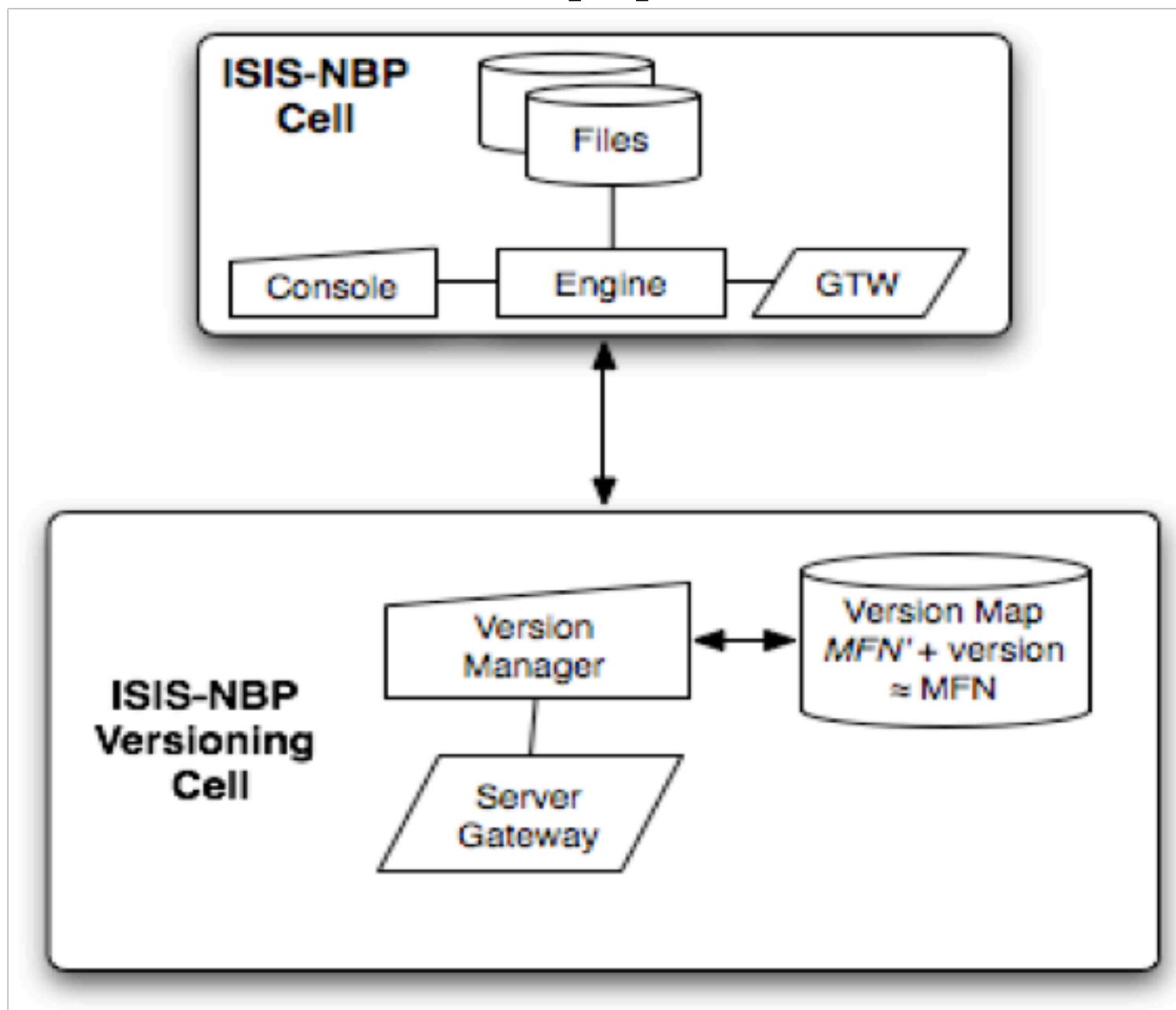
ISIS-NBP App Scenario



ISIS-NBP App Scenario



ISIS-NBP App Scenario



ISIS-NBP Client Applications

1. You do **not** need to know **any** Python
2. ...OTOH, **Python** is **easy to learn**, fun to use.
3. Two API options: **Restful HTTP** or **XML-RPC**

Restful HTTP

REST means...

REST is important because:

- Web friendly, scalable, flexible, standard
- transparent access through firewalls (uses HTTP)
- content: bookmarked, redirected, proxied, cached
- MIME types → binary ISIS, ISO, XML,...
- programming language agnostic
- authenticated and encrypted access (for free)
- consolidated standard != SOAP/WSDL

Rest Example: PROTOCOL

POST /web/restapi/sample/

User-Agent: MyApplication ID

Accept: application/xml, text/xml

Accept-Language: en-us,pt-br

Accept-Encoding: gzip,deflate

Accept-Charset: ISO-8859-1,utf-8

Connection: keep-alive

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

Content-Length: 90

Pragma: no-cache

Cache-Control: no-cache

```
<?xml version="1.0" encoding="utf-8">
```

```
  <parameters>
```

```
    <database>mybiblio</database>
```

```
  </parameters>
```

Rest Example: Application

```
CREATE_DB_TEMPLATE =""  
<?xml version="1.0" encoding="utf-8"?>  
<parameters>  
    <database>mybiblio</database>  
</parameters>  
"  
  
import httplib  
con = httplib.HTTPConnection('127.0.0.1', 8000)  
conn.request('POST','/web/restapi/sample/', CREATE_DB_TEMPLATE)  
result = conn.getresponse()  
print result.read()  
con.close()
```

Rest Example: Application

```
READ_DB_TEMPLATE ="  
<?xml version="1.0" encoding="utf-8"?>  
<parameters>  
    <gizmo></gizmo>  
    <from>1</from>  
    <to>20</to>  
</parameters>  
"  
  
import httplib  
con = httplib.HTTPConnection('127.0.0.1', 8000)  
conn.request('POST','/web/restapi/sample/mybiblio', READ_DB_TEMPLATE)  
result = conn.getresponse()  
print result.read()  
con.close()
```

XML-RPC API

<http://localhost:8080/web/xmlrpc/>

This is an XML-RPC Service.

You need to invoke it using an XML-RPC Client!

The following methods are available:

- * **createDB**: Create new database
- * **createRecord**: Create new record
- * **deleteDB**: Delete database
- * **deleteRecord**: Delete record
- * **readRecord**: Read record
- * **updateRecord**: Update record

ISIS-NBP Server

XML-RPC Example: Python

```
import sys
import xmlrpclib
target_url = 'http://%s:%d/web/xmlrpc/' % ('localhost', 8000)
rpc_srv = xmlrpclib.ServerProxy(target_url, verbose=False)
params = {'collection':'sample', 'database': 'mybiblio'}
params['24'] = 'Techniques for the measurement of transpiration of individual plants'
params['26'] = '^aParis^bUnesco^c-1965'
params['30'] = '^ap. 211-224^billus.'
params['44'] = 'Methodology of plant eco-physiology: proceedings of the Montpellier Symposium'
params['50'] = 'Incl. bibl.'
params['69'] = 'Paper on: <plant physiology><plant transpiration><measurement and instruments>'
params['70'] = ('Magalhaes, A.C.', 'Franco, C.M.')
rpc_srv.createRecord(params)
```

XML-RPC Example: PHP

```
<?php
include("xmlrpc.inc");
$c=new xmlrpc_client("/web/xmlrpc/", "localhost", 8000)
$database = new xmlrpcval( array (
    'collection' => new xmlrpcval('sample','string'),
    'database' => new xmlrpcval('mybiblio','string'),
    '24' => new xmlrpcval('Techniques for the measurement of transpiration of individual plants','string'),
    '26' => new xmlrpcval('^aParis^bUnesco^c-1965','string'),
    '30' => new xmlrpcval('^ap. 211-224^billus.','string'),
    '44' => new xmlrpcval('Methodology of plant eco-physiology: proceedings of the Montpellier Symposium','string'),
    '50' => new xmlrpcval('Incl. bibl.','string'),
    '69' => new xmlrpcval('Paper on: <plant physiology><plant transpiration><measurement and instruments>','string'),
    '70' => new xmlrpcval(
        array (
            new xmlrpcval('Magalhaes,A.C','string'),
            new xmlrpcval('Franco, C.M.','string')
        ), 'array'),
    ),'struct');

$f=new xmlrpcmsg('createRecord',array/php_xmlrpc_encode($database)));
prtSend($f);
$r=&$c->send($f);
print $r;
?>
```

XML-RPC Example: Java

```
package br.com.gpr.javaclient;

import java.util.HashMap;
import java.net.URL;
import org.apache.xmlrpc.client.XmlRpcClient;
import org.apache.xmlrpc.client.XmlRpcClientConfigImpl;
import org.apache.xmlrpc.*;

public class JavaClient {

    private final static String server_url =
        "http://127.0.0.1:8000/web/xmlrpc/";

    static String result;

    public static void main (String [] args) {
        try {
            XmlRpcClientConfigImpl config = new XmlRpcClientConfigImpl();
            config.setServerURL(new URL(server_url));
            XmlRpcClient client = new XmlRpcClient();
            client.setConfig(config);
            HashMap<String, String> params = new HashMap<String, String>();
            params.put("collection", "sample");
            params.put("database", "dbtest");
            createRecord(client);
        } catch (XmlRpcException exception) {
            System.err.println("JavaClient: XML-RPC Fault #" +
                Integer.toString(exception.code) + ":" +
                exception.toString());
        } catch (Exception exception) {
            System.err.println("JavaClient: " + exception.toString());
        }
    }

    public static void createRecord(XmlRpcClient client){
        try{
            HashMap<String, String> params = new HashMap<String, String>();
            params.put("collection", "sample");
            params.put("database", "dbtest");
            individual plants);
            params.put("24","Techniques for the measurement of transpiration of
            params.put("26","^aParis^bUnesco^c-1965");
            params.put("30","^ap. 211-224^billus.");
            params.put("44",
            "Methodology of plant eco-physiology: proceedings of the Montpellier Symposium");
            params.put("50","Incl. bibl.");
            params.put("69",
            "Paper on: <plant physiology><plant transpiration><measurement and instruments>");
            params.put("70","Magalhaes, A.C. Franco, C.M.");
            Object[] params1 = new Object[]{params};
            result = (String) client.execute("createRecord", params1);
            System.out.println("createRecord >>: " + result);
        } catch (XmlRpcException exception) {
            System.err.println("JavaClient: XML-RPC Fault #" +
                Integer.toString(exception.code) + ":" +
                exception.toString());
        } catch (Exception exception) {
            System.err.println("JavaClient: " + exception.toString());
        }
    }
}
```

Developing ISIS-NBP

Developing ISIS-NBP

- Sorry, probably not enough time for that!

Developing ISIS-NBP

- Sorry, probably not enough time for that!
- Go to our Open Source Panel tomorrow.

Developing ISIS-NBP

- Sorry, probably not enough time for that!
- Go to our Open Source Panel tomorrow.
- Go to my Presentation on PyCon2008 on Thursday (here in Rio).

Developing ISIS-NBP

- Sorry, probably not enough time for that!
- Go to our Open Source Panel tomorrow.
- Go to my Presentation on PyCon2008 on Thursday (here in Rio).
- Talk to me afterwards, and I show it to you...

Developing ISIS-NBP

- Sorry, probably not enough time for that!
- Go to our Open Source Panel tomorrow.
- Go to my Presentation on PyCon2008 on Thursday (here in Rio).
- Talk to me afterwards, and I show it to you...
...I mean the source code!!!

Why Python ?

FAQ: Why Python?

FAQ: Why Python?

- **general purpose** Programming Language

FAQ: Why Python?

- **general purpose** Programming Language
- focused on **developer productivity**

FAQ: Why Python?

- **general purpose** Programming Language
- focused on **developer productivity**
- **multi-platform** and **multi-paradigm**

FAQ: Why Python?

- **general purpose** Programming Language
- focused on **developer productivity**
- **multi-platform** and **multi-paradigm**
- **rich**: batteries included + 3rd party extensions

FAQ: Why Python?

- **general purpose** Programming Language
- focused on **developer productivity**
- **multi-platform** and **multi-paradigm**
- **rich**: batteries included + 3rd party extensions
- truly **floss** software

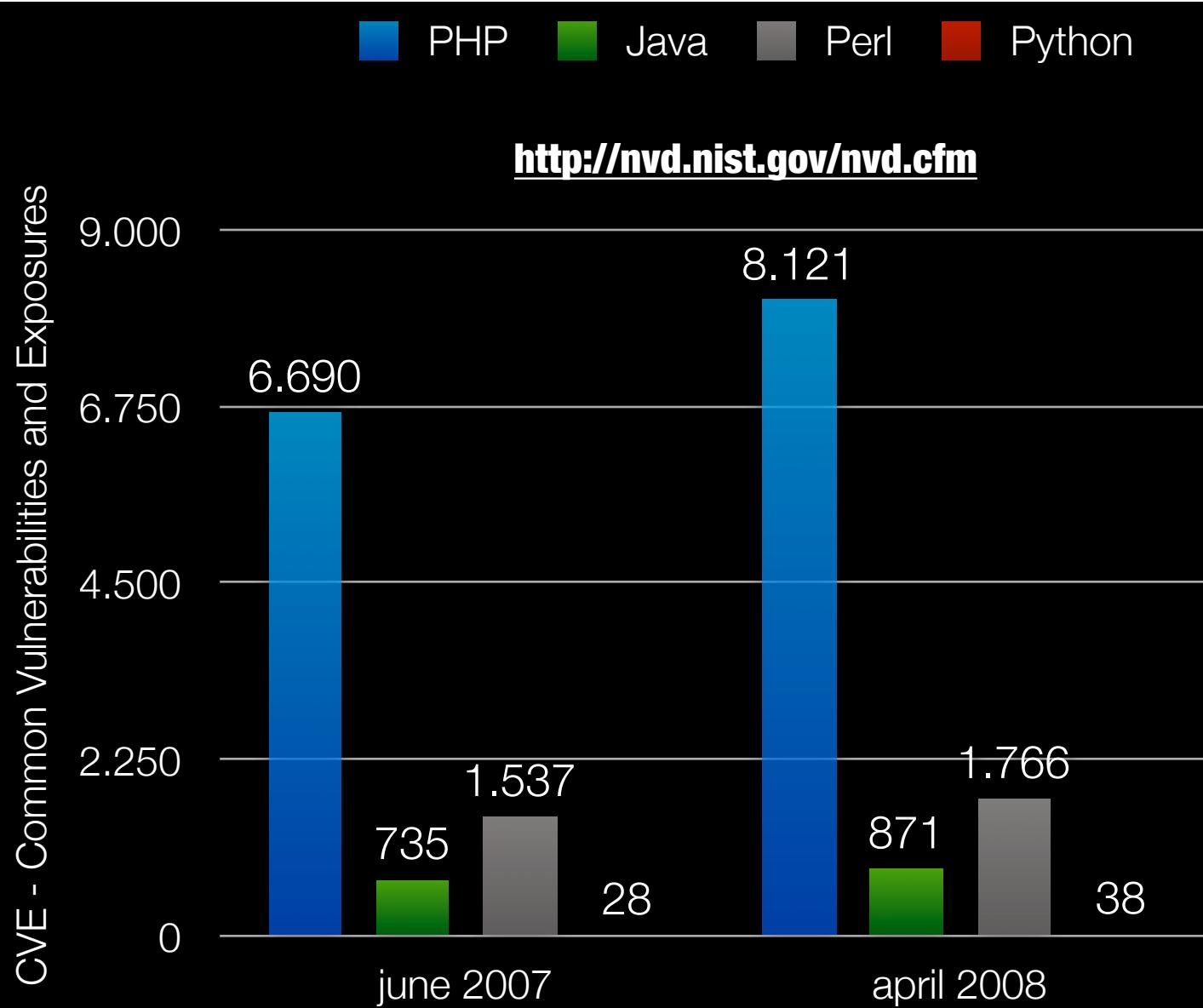
FAQ: Why Python?

- **general purpose** Programming Language
- focused on **developer productivity**
- **multi-platform** and **multi-paradigm**
- **rich**: batteries included + 3rd party extensions
- truly **floss** software
- **easy** to extend and embed (C,C++,Fortran,Delphi, Java)

FAQ: Why Python?

- **general purpose** Programming Language
- focused on **developer productivity**
- **multi-platform** and **multi-paradigm**
- **rich**: batteries included + 3rd party extensions
- truly **floss** software
- **easy** to extend and embed (C,C++,Fortran,Delphi, Java)
- **mature** (created in 1989, released 1991)

Security



Who is using Python ?



YAHOO!



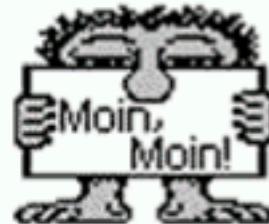
gentoo

NOKIA
CONNECTING PEOPLE



 **Los Alamos**
NATIONAL LABORATORY
EST. 1943

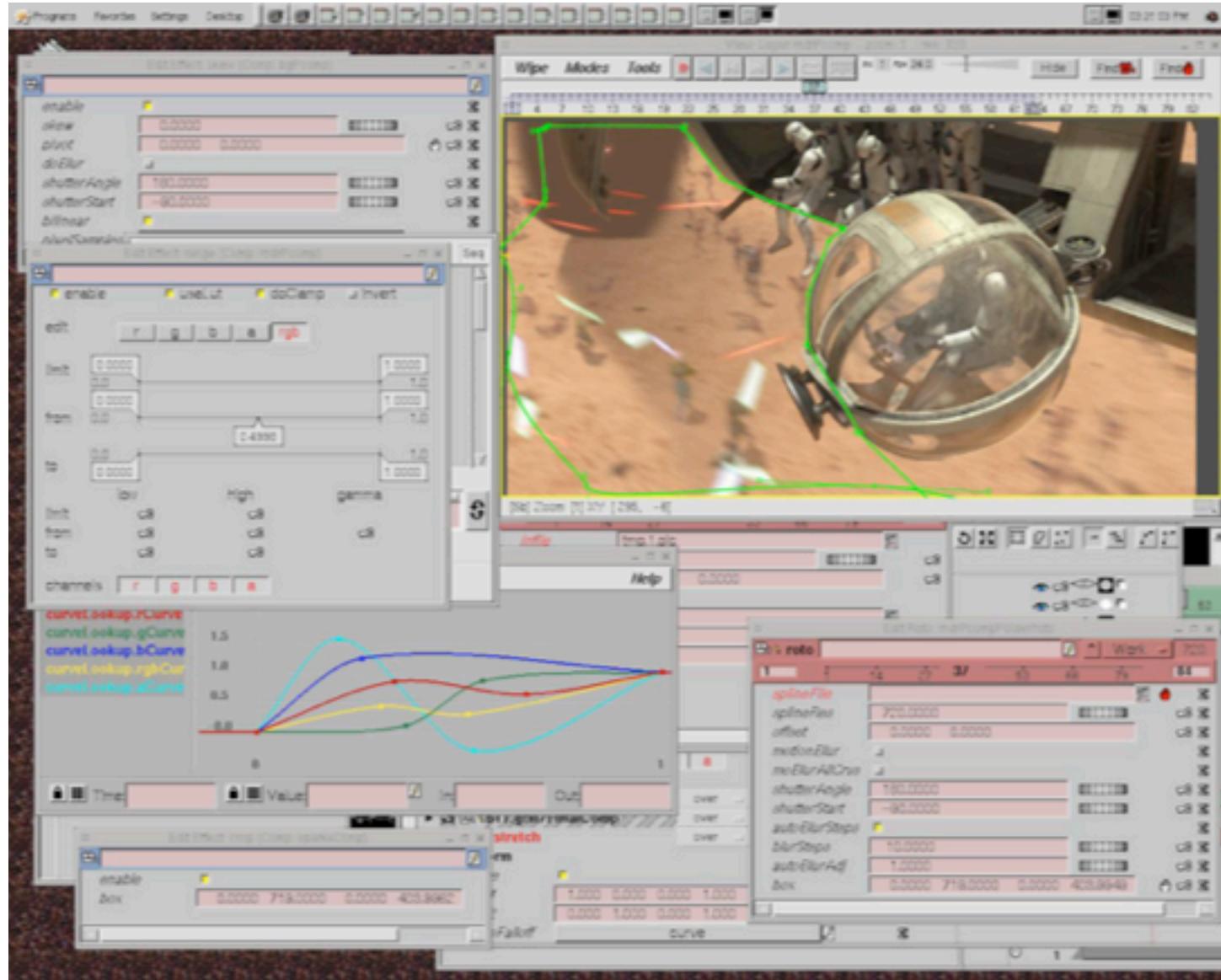
Who is using Python ?



Integrated SCM & Project Management



Who is using Python ?



Acknowledgements

- BIRME/PAHO/WHO
- Unesco
- GPr Sistemas
- ISIS-NBP Team
- External collaborators

Abel L. Packer

João Chaves Jr

Adalberto Tardelli

Renato Murasaki

Ernesto Spinak

Rodrigo Senra

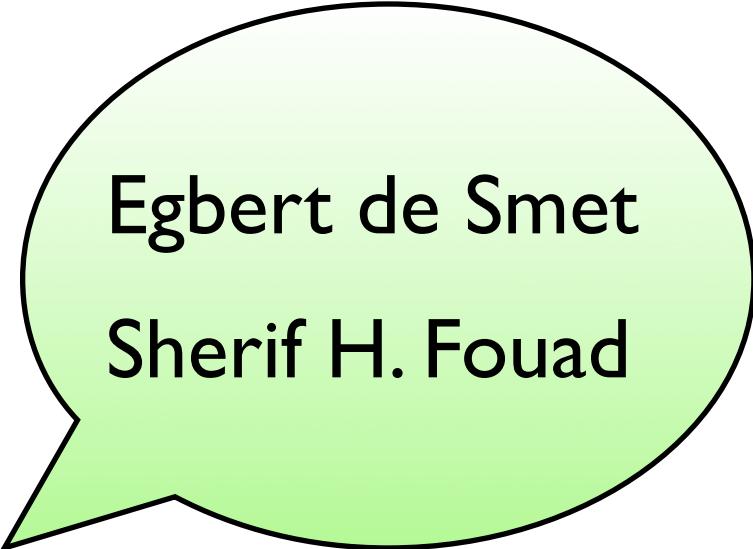
Francisco José Lopes

Heitor Barbieri

- ISIS-NBP Team
- External collaborators

Acknowledgements

- BIRME/PAHO/WHO
- Unesco
- GPr Sistemas
- ISIS-NBP Team
- External collaborators



Egbert de Smet
Sherif H. Fouad

Rodrigo Dias Arruda Senra

<http://rodrigo.senra.nom.br>

rsenra@acm.org

**Obrigado a todos
pela atenção.**

Rodrigo Dias Arruda Senra

<http://rodrigo.senra.nom.br>

rserna@acm.org