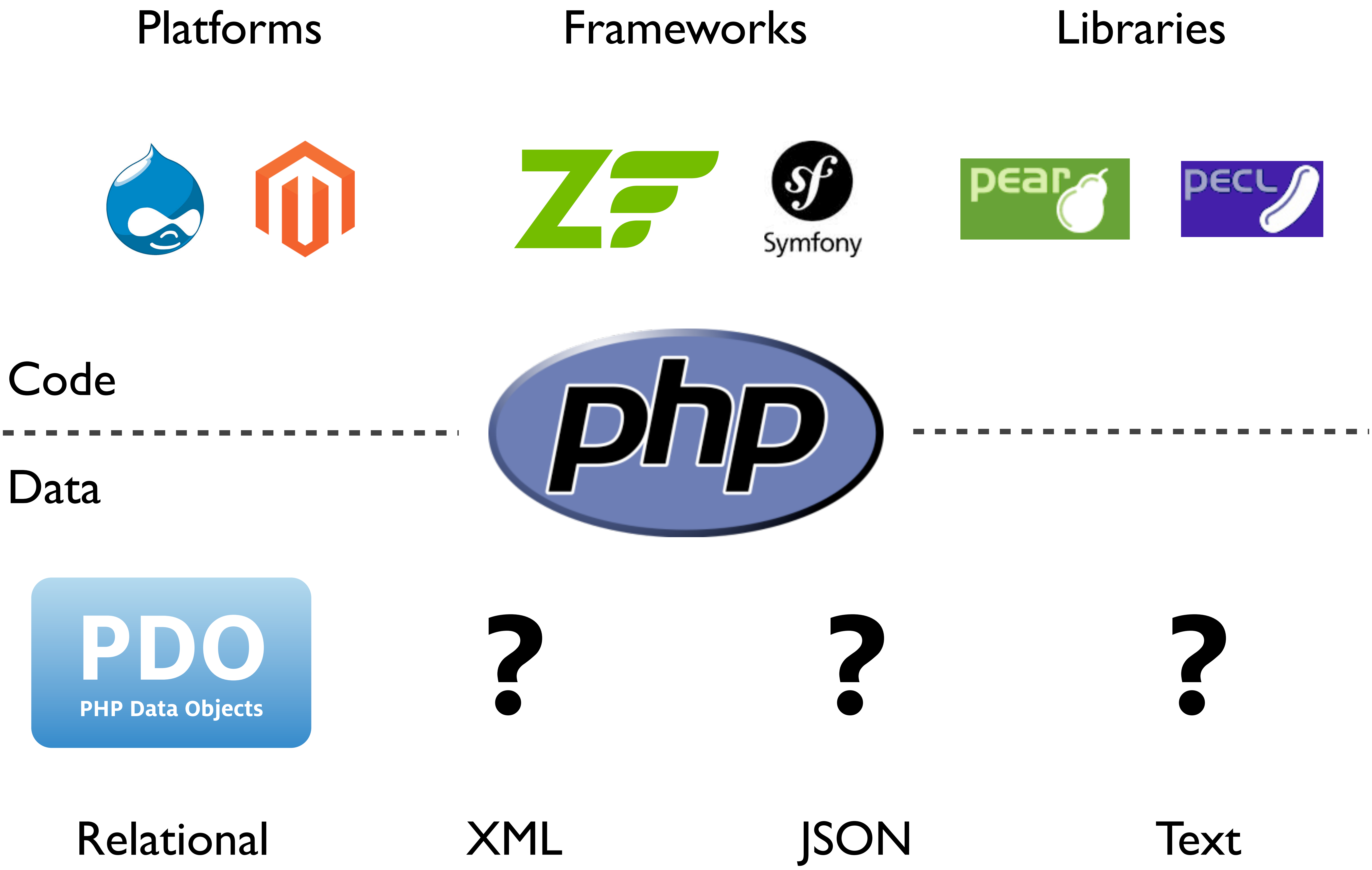
Cutting Edge Data Processing with PHP & XQuery

William Candillon {candillon@28msec.com}

PHP Tour Lille 2011

A Rich Ecosystem



Platforms

Frameworks

Libraries

ZF

Symfony

pear

pecl

Code

php

Data

PDO

PHP Data Objects

?

?

?

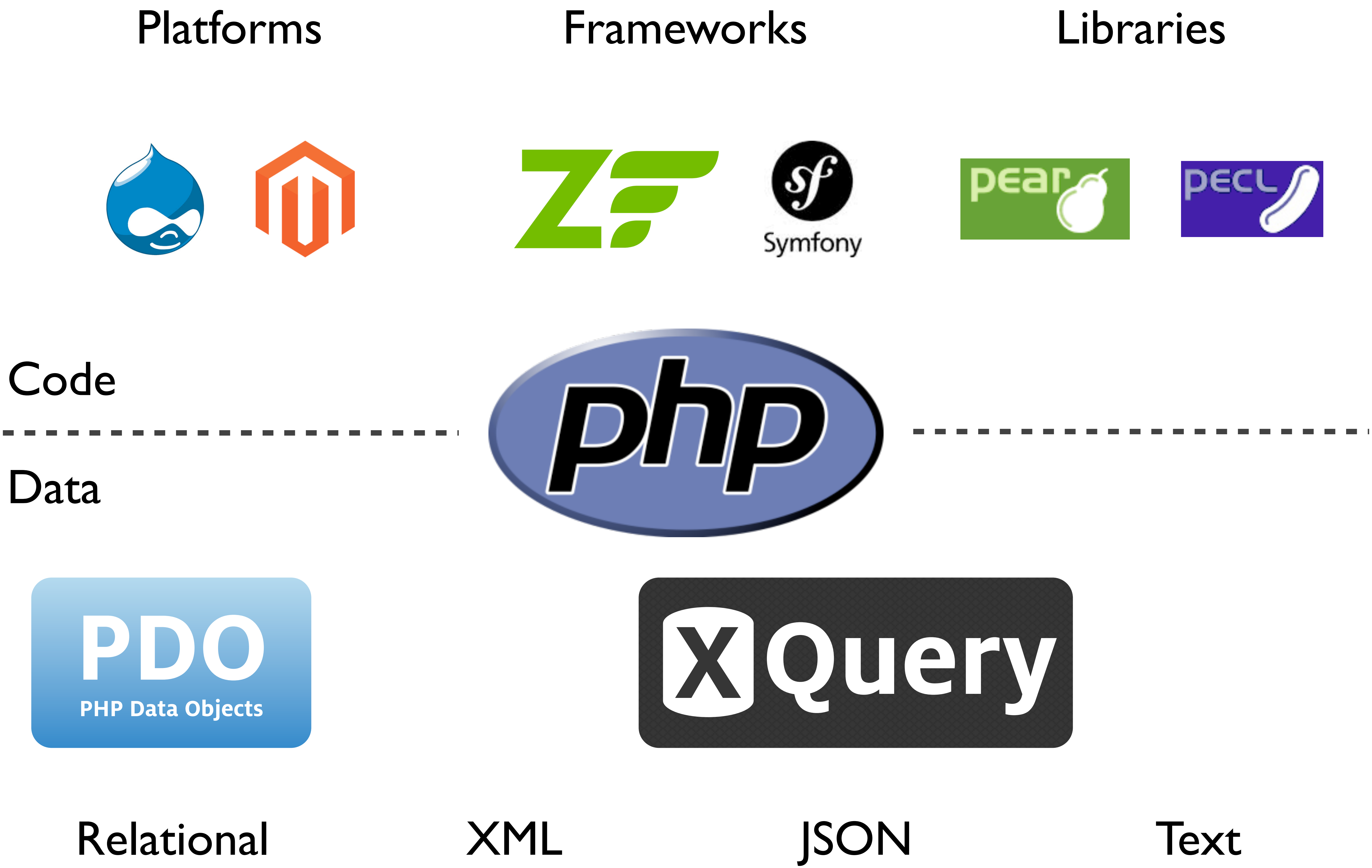
Relational

XML

JSON

Text

A Rich Ecosystem



Platforms

Frameworks

Libraries

Code

Data

Relational

XML

JSON

Text

Our Goal

Contribute the features from relational APIs to unstructured data

Data APIs

	PDO	SimpleXML	Zend JSON	XQuery
Data	Relational	XML	JSON	XML/JSON
Queries	✓	XPath	✗	✓
Updates	✓	✗	✗	✓
Streaming	✓	✗	✗	✓
Full-Text	✓	✗	✗	✓

Meet XQuery

- Family of specifications from W3C
    - XQuery 3.0
    - Update Facility
    - Scripting Extension
    - Full-Text

- EXPath: de-facto standard libraries

Meet Zorba

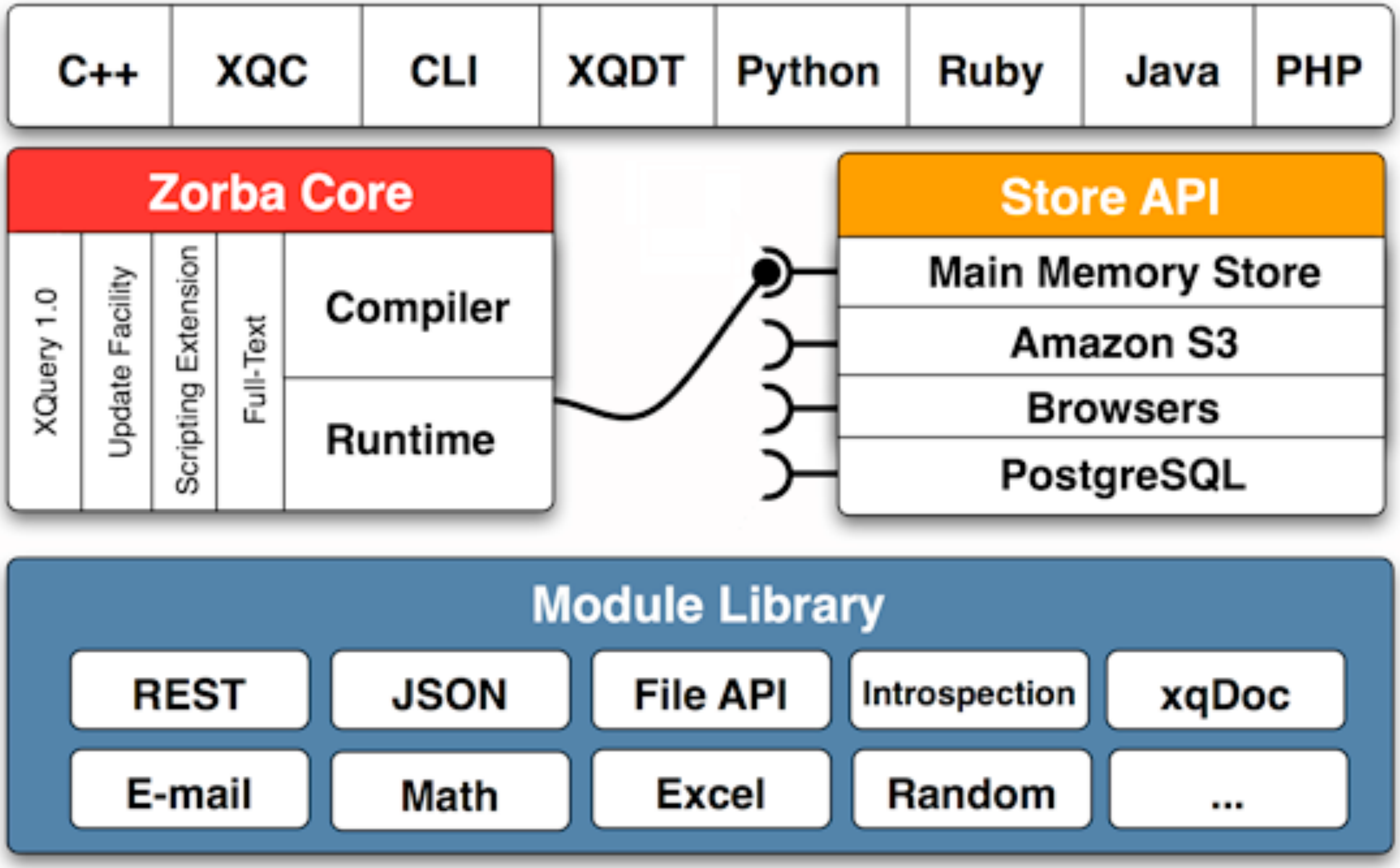
OPEN SOURCE 100% FREE FREE OF CHARGE

Open Source XQuery Processor

Contributors: Oracle, 28msec, FLWOR

- All Flavors Available
- Rich Module Libraries
- Pluggable Store
- Runs Everywhere
- Developer Tools
- Fun & Productive

Zorba


C++
XQC
CLI
XQDT
Python
Ruby
Java
PHP
Zorba Core
XQuery 1.0
Update Facility
Scripting Extension
Full-Text
Compiler
Runtime
Store API
Main Memory Store
Amazon S3
Browsers
PostgreSQL
Module Library
REST
JSON
File API
Introspection
xqDoc
E-mail
Math
Excel
Random
...

```php
<?php
require_once ‘ZorbaXQueryProcessor.php’;

$xquery = new XQueryProcessor();

$xquery->importQuery(‘1+1’);

echo $xquery->execute();
?>
```

```
<?php
require_once ‘ZorbaXQueryProcessor.php’;

$xquery = new XQueryProcessor();

$query = <<<‘XQ’
  let $world := ‘World’
  return <h1>Hello {$world}</h1>
XQ;

$xquery->importQuery($query);

echo $xquery->execute();
?>
```

```
<?php
require_once ‘ZorbaXQueryProcessor.php’;

$xquery = new XQueryProcessor();

$xquery->importQueryFromURI(‘hello.xq’);

echo $xquery->execute();
?>
```

```php
<?php
require_once ‘ZorbaXQueryProcessor.php’;

$xquery = new XQueryProcessor();

$query = <<<‘XQ’
  declare variable $world external;

  <h1>Hello {$world}</h1>
XQ;

$xquery->importQuery($query);

$xquery->setVariable(‘world’, ‘World!’);

echo $xquery->execute();
?>
```

```
$xquery = new XQueryProcessor();

$query = <<<‘XQ’
declare variable $foo as xs:string external;
declare variable $bar as xs:integer external;
declare variable $doc1 as document-node() external;
declare variable $doc2 as document-node() external;

$foo, $bar, $doc1, $doc2
XQ;

$xquery->importQuery($query);

$xquery->setVariable(‘foo’, ‘bar’);
$xquery->setVariable(‘bar’, 3);

$doc = simplexml_load_file ('data/sessions.xml');
$xquery->setVariable("doc1", $doc);

$doc = $xquery->parseXML ("<root />");
$xquery->setVariable("doc2", $doc);

echo $xquery->execute();
```

Queries

Updates

Full-Text

Streaming

Queries

Updates

Full-Text

Streaming

```
import module namespace functx = "http://www.functx.com/";
import module namespace html  = "http://example.com/html";

declare variable $sessions  external;

<html lang="en">
  <body>
  {$html:header}

  <div id="main">{
    html:sessions($sessions/*)
  }</div>

  {$html:footer}
</body>
</html>
```

```
<div id=”sessions”>{
  for $session in $sessions
  let $title := string($session/@title)
  where $session/@conf = “PHP Tour Lille”
  order by $session/@starts ascending
  return <div>
    <h1>{$title}</h1>
  </div>
}</div>
```

```
<div id=”sessions”>{
for $day-sessions in $sessions
let $starts := dateTime($day-sessions/@starts)
let $day    := xs:day($starts)
group by $day
order by $starts ascending
return <div>
  <h1>{$day}</h1>
  {
    for $session in $day-sessions
    return <h2>{string($session/@title)}</h2>
  }
  </div>
}</div>
```

Queries

Updates

Full-Text

Streaming

```
insert node $bios[@id=”wcandilllon”]
       into $sessions[@id=”xquery”]
```

```
replace value of node $session/@starts
        with “2011–11–24T10:15:00”

replace value of node $session/@ends
        with “2011–11–24T10:15:00”
```

```
let $session := $sessions[id="XQuery"]
return
  if ($session/rating) then
    replace value of node $session/rating
                     with "B"
    else
      insert node <rating>B</rating>
             into $session
```

Queries

Updates

Full-Text

Streaming

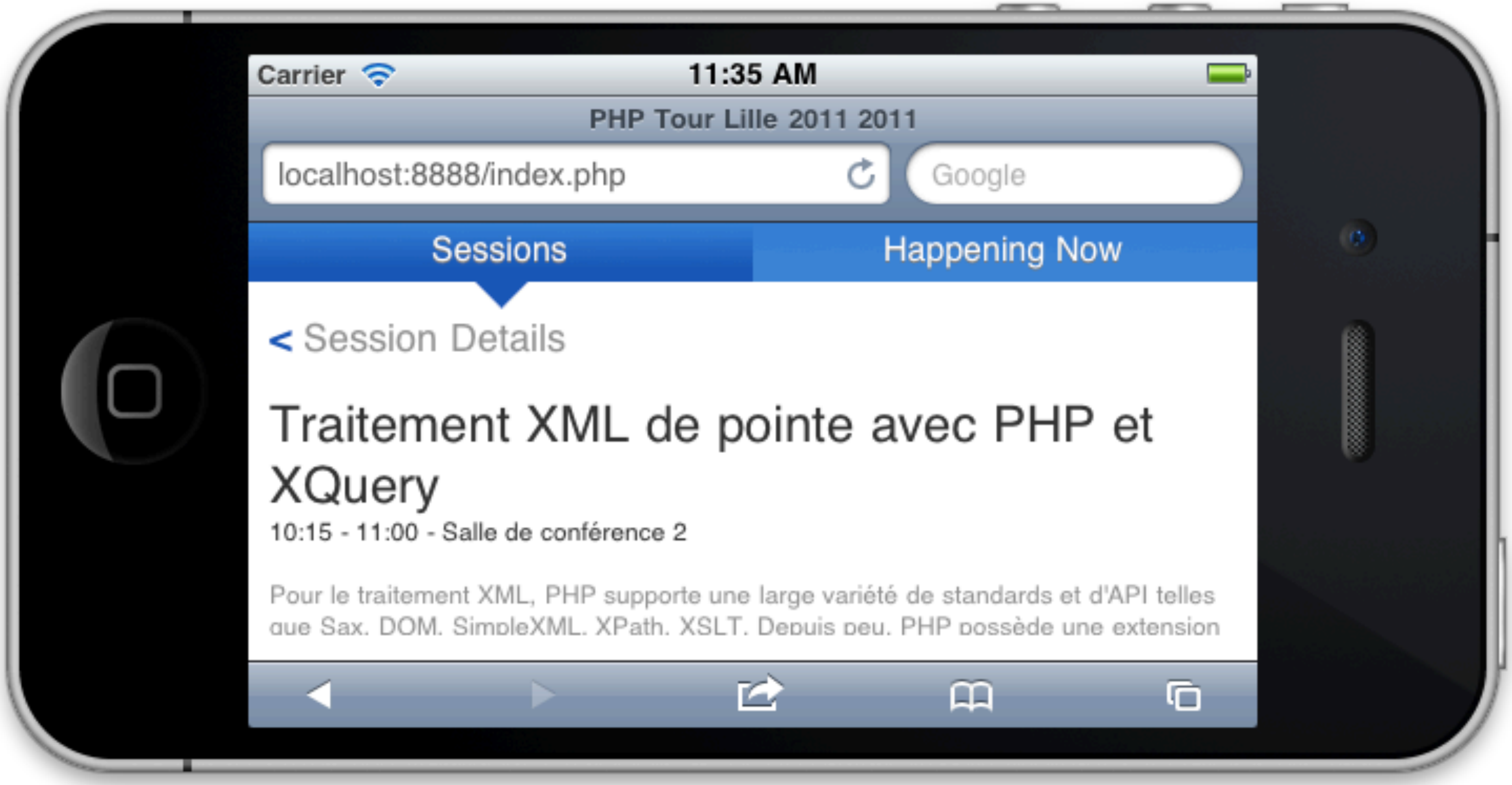
```
let $sessions := $sessions/session[
                   . contains text {$search-term}
                   all words
                   distance at most 8 words]
return
  if(empty($sessions)) then
    <h1>No Results found</h1>
  else html:sessions($sessions)
```

```
let $x := <msg>breakfast of champions</msg>
return $x contains text "meal"
```

```
let $x := <msg>breakfast of champions</msg>
return $x contains text "meal"
  using thesaurus at "http://wordnet.princeton.edu"
  relationship "narrower term"
```

```
let $doc := doc(“doc.xml”)
for $token in ft:tokenize($doc)
return
  concat($token/@value, “ at ”,
         $token/@paragraph,
         $token/@sentence
  )
```

Demo



Carrier 11:35 AM

PHP Tour Lille 2011 2011

localhost:8888/index.php

< Session Details

Traitement XML de pointe avec PHP et XQuery

10:15 - 11:00 - Salle de conférence 2

Pour le traitement XML, PHP supporte une large variété de standards et d'API telles que Sax, DOM, SimpleXML, XPath, XSLT. Depuis peu, PHP possède une extension

Queries

Updates

Full-Text

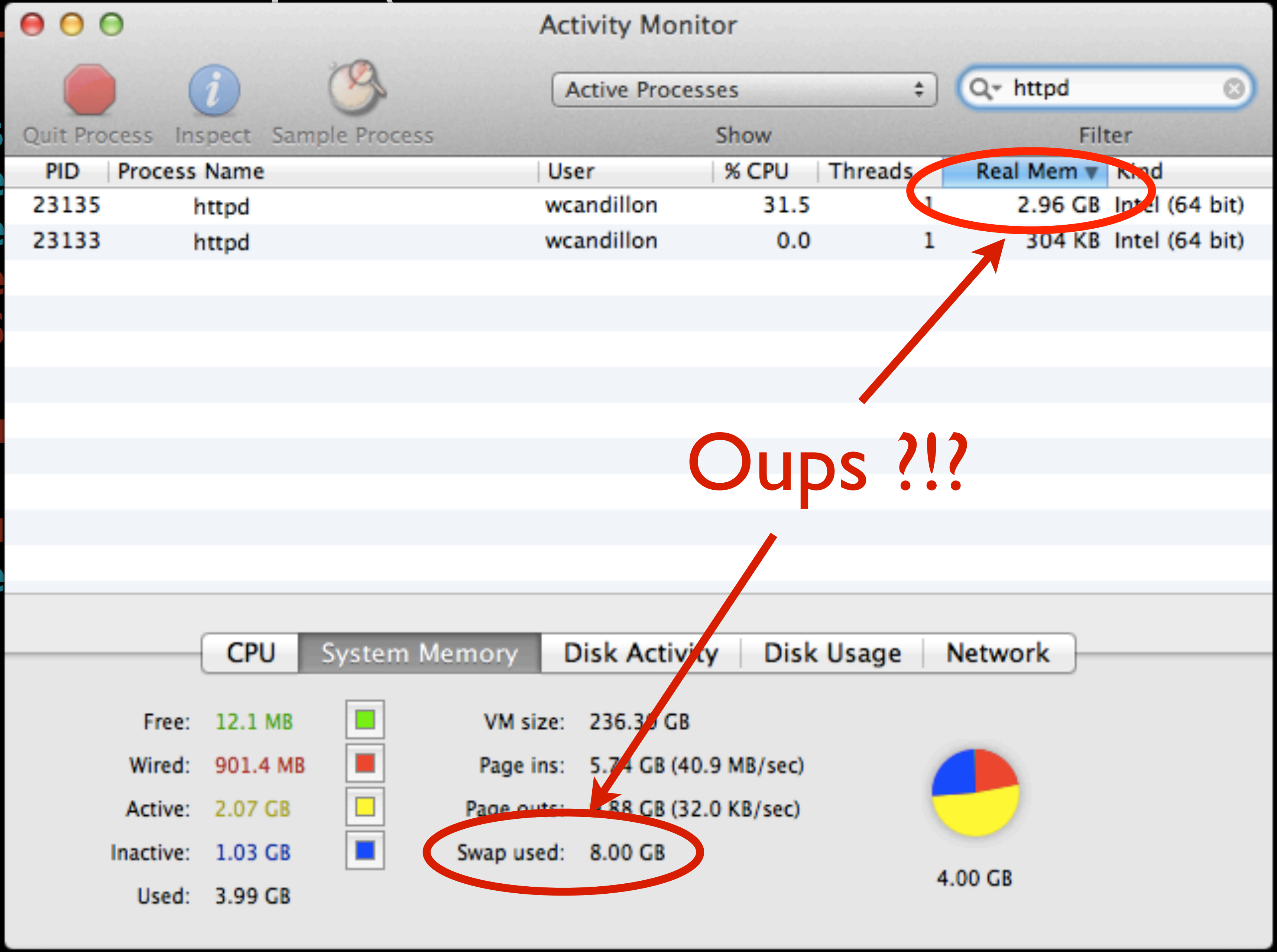
Streaming

Forecast Data

- 1 Day of Forecast data: 727MB
- Get data for a specific site
- Send selected temperatures to clients
- Display chart

```php
<?php
$siteId = 3;
$forecasts = simplexml_load_file('forecasts.xml');
$forecasts = $forecasts->xpath(
     "/forecast-list/forecast[@site-id='$siteId']");

foreach($forecasts as $forecast) {
  $time  = $forecast->xpath("@time-step");
  $value = $forecast->xpath(
    "//weather-elements/weather-element"
   ."[@name = 'ScreenTemperature']/text()");

   echo "<temperature time='"
         .$time[0]
         ."'value='"
         .$value[0]."' />\n";
}
?>
```

```php
<?php
$siteId = 3;
$forecasts = simplexml_load_file('forecasts.xml');
$forecasts = $forecasts->xpath(
     "/forecast-

foreach($forecas
  $time  = $fore
  $value = $fore
    "//weather-e
   ."[@name = 'S

   echo "<tempe
         .$time[
         ."'valu
         .$value
}
?>
```



Activity Monitor

Active Processes

httpd

PID	Process Name	User	% CPU	Threads	Real Mem ▾	Kind
23135	httpd	wcandillon	31.5	1	2.96 GB	Intel (64 bit)
23133	httpd	wcandillon	0.0	1	304 KB	Intel (64 bit)

Free: 12.1 MB 🟩
Wired: 901.4 MB 🟥
Active: 2.07 GB 🟨
Inactive: 1.03 GB 🟦
Used: 3.99 GB

VM size: 236.30 GB
Page ins: 5.74 GB (40.9 MB/sec)
Page outs: 88 GB (32.0 KB/sec)
Swap used: 8.00 GB

Oups ?!!

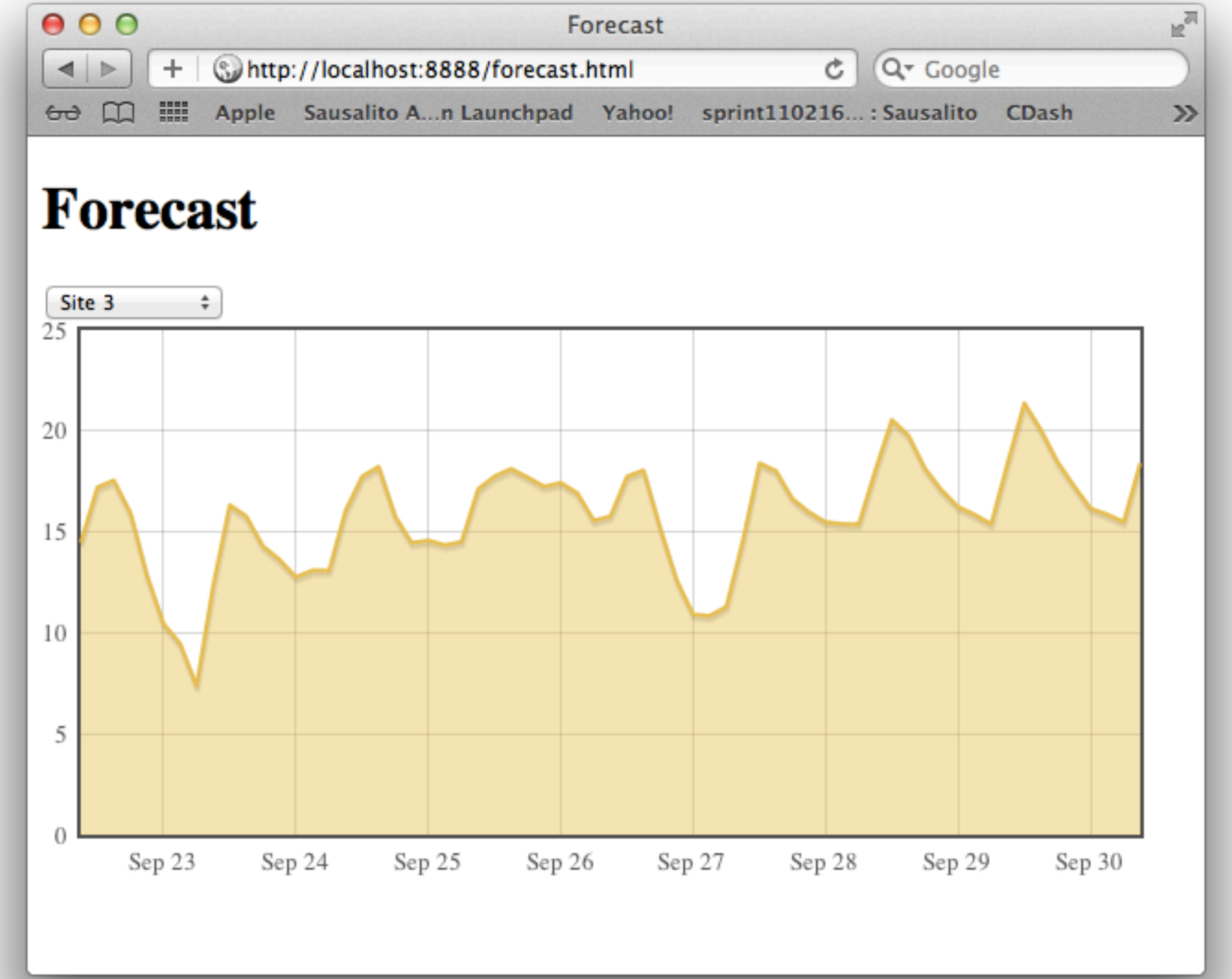
```php
<?php
$siteId = 3;
$forecasts = simplexml_load_file('forecasts.xml');
$forecasts = $forecasts->xpath(
     "/forecast-list/forecast[@site-id='$siteId']");

foreach($forecasts as $forecast) {
  $time  = $forecast->xpath("@time-step");
  $value = $forecast->xpath(
    "//weather-elements/weather-element"
   ."[@name = 'ScreenTemperature']/text()");

   echo "<temperature time='"
         .$time[0]
         ."'value='"
         .$value[0]."' />\n";
}
?>
```

Memory Footprint: 9GB

```
for $forecast in z:parse-xml(file:read-text("forecasts.xml"),
                             <opt:options>
                               <opt:parseExternalParsedEntity
                                 opt:skipRootNodes="1"/>
                             </opt:options>)
where $forecast/@site-id = "3"
let $time  := string($forecast/@time-step)
let $value := $forecast/weather-elements/weather-element
                         [@name = 'ScreenTemperature']/text()
return
  <temperature time="{$time}" value="{$value}" />
```

Memory Footprint: 19MB

Demo



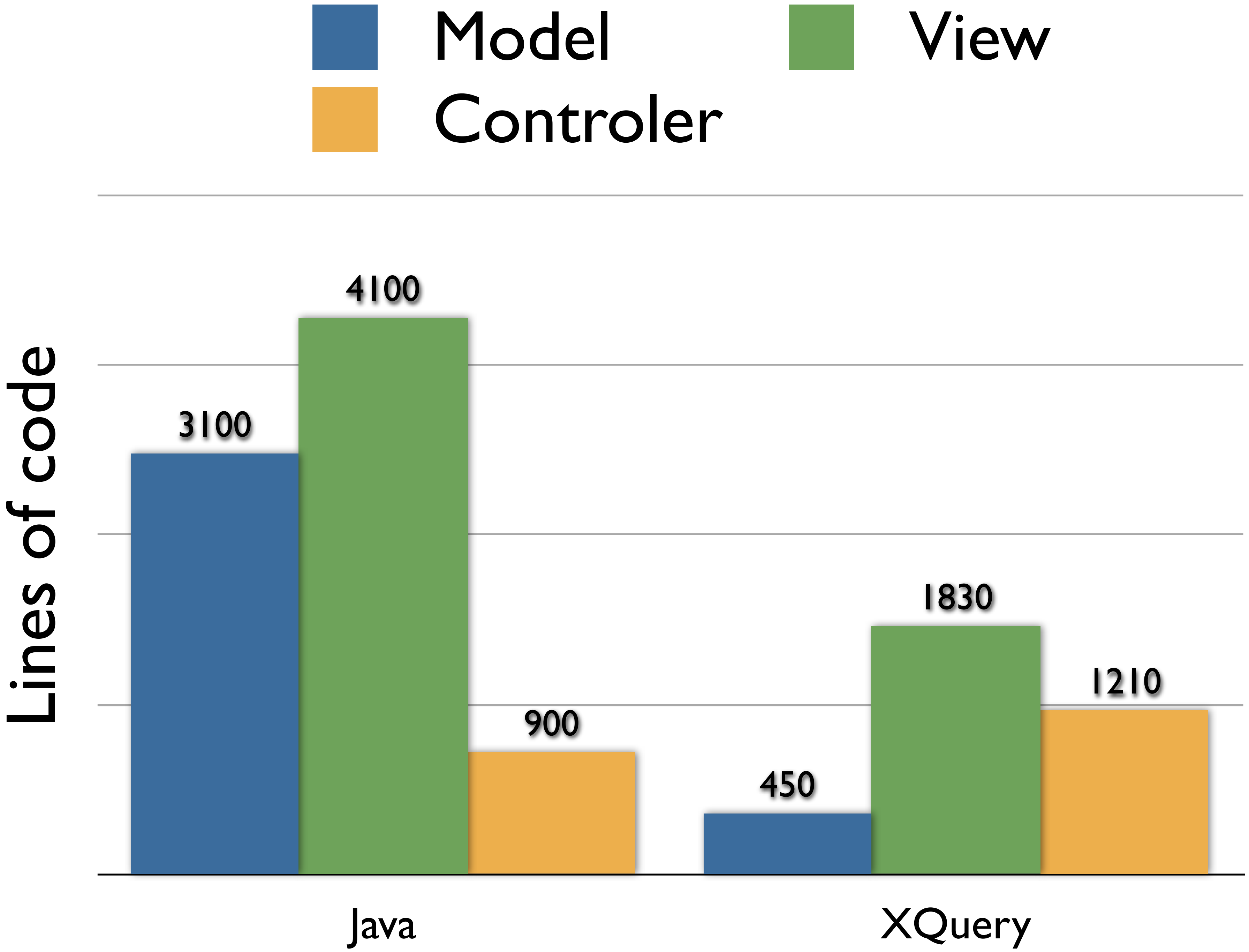
Forecast
http://localhost:8888/forecast.html
Google
Apple
Sausalito A...n Launchpad
Yahoo!
sprint110216... : Sausalito
CDash

Forecast

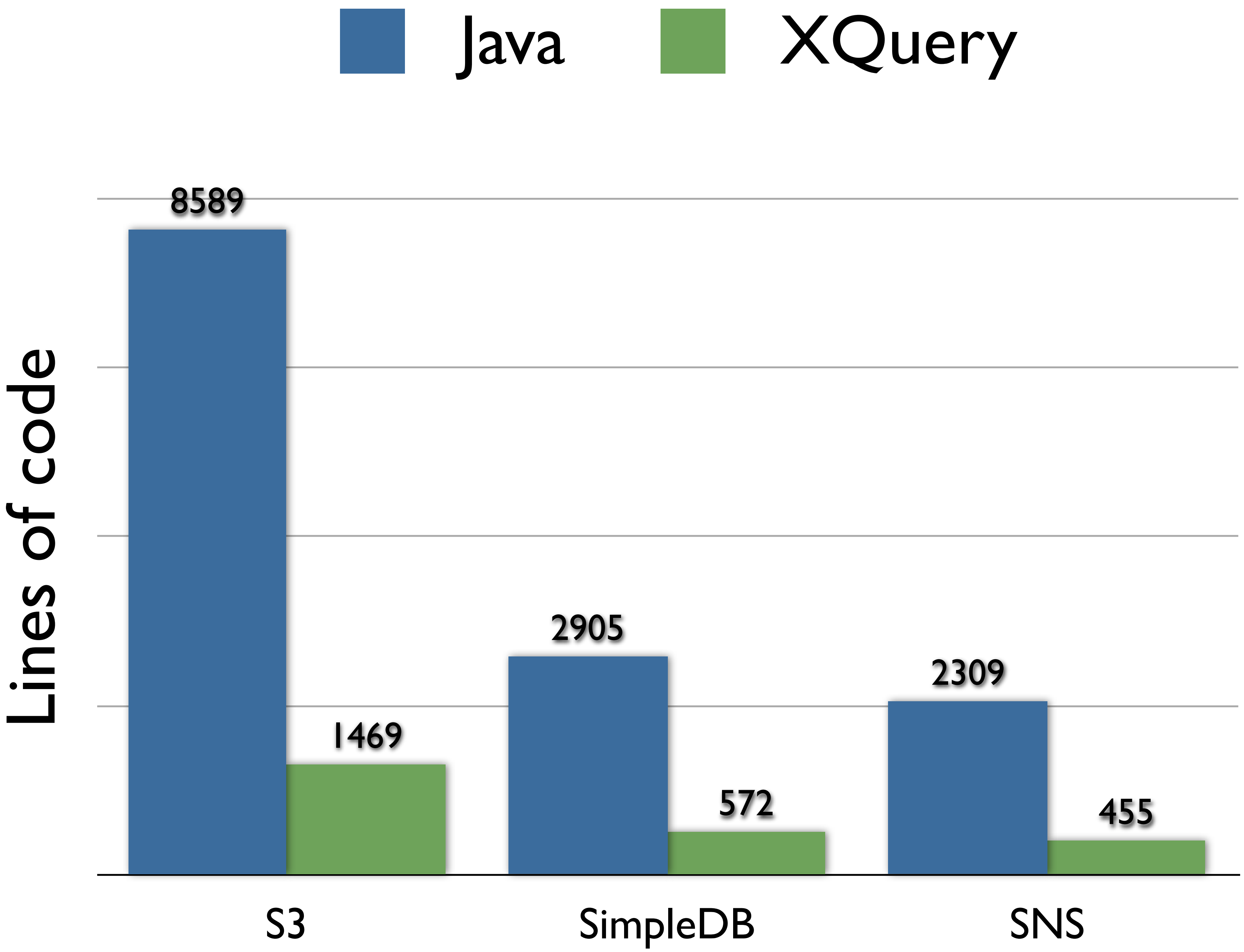
Results

	SimpleXML	XMLReader	XQuery
Streaming	✗	✓	✓
Productivity	✓	✗	✓

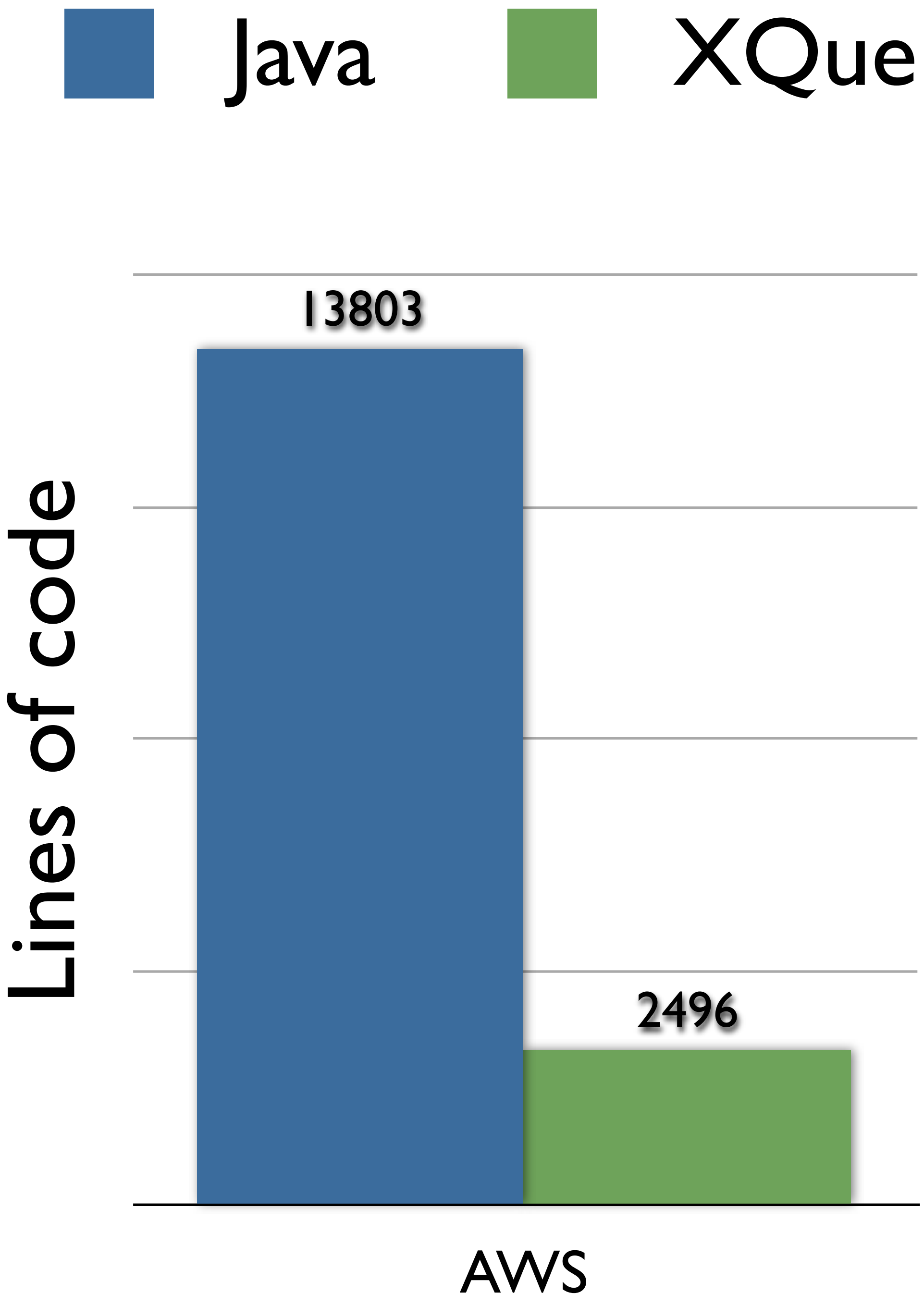
Pubzone (2009)



AWS Libraries



AWS Libraries



Lines of Codes

- 80%

```
<html>
  <head>
    <script type='text/javascript'>
    function buy(e) {
      newElement = document.createElement("p");
      elementText = document.createTextNode
                  (e.target.getAttribute(id));
      newElement.appendChild(elementText);
      var res = document.evaluate(
        "//div[@id='shoppingcart']",
        document, null,
        XPathResult.UNORDERED_NODE_SNAPSHOT_TYPE,
        null);
      res.snapshotItem(0).appendChild("newElement");}
    </script>
  </head>
  <body>
    <div>Shopping cart</div>
    <div id="shoppingcart"></div>
    <%
      // Code establishing connection
      ResultSet results =
      statement.executeQuery ("SELECT * FROM PRODUCTS");
      while (results.next()) {
        out.println("<div>");
        String prodName = results.getString(1);
        out.println(prodName);
        out.println("<input type='button' value='Buy'");
        out.println("id='"+prodName+"'");
        out.println("onclick='buy(event)'/>").
        out.println("</div>");
      }
      results.close();
      // Code closing connection
    %>
  </body>
</html>
```

HTML

JavaScript

XPath

Java

SQL

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
   <script type='application/xquery'>
declare updating function local:buy($evt, $obj) {
  insert node <p>{$obj/@id}</p> as first
    into //div[@id="shoppingcart"]
};
b:addEventListener(b:dom()//input,
                   "onclick",
                   xs:Qname("local:buy"));
</script>
  </head>
  <body>
    <div>Shopping cart</div>
    <div id="shoppingcart">{
      for $p in doc("products.xml")//*:product
      return
        <div>
          {$p/*:name}
          <input type='button'
                 value='Buy'
                 id='{$p/*:name}'/>
        </div>
    }</div>
  </body>
</html>
```

XQuery Everywhere

http://www.youtube.com/watch?v=ql2TLTsilo8

Dimitri finds out about XQuery

wcandill

Fucking SQL injections

683 views

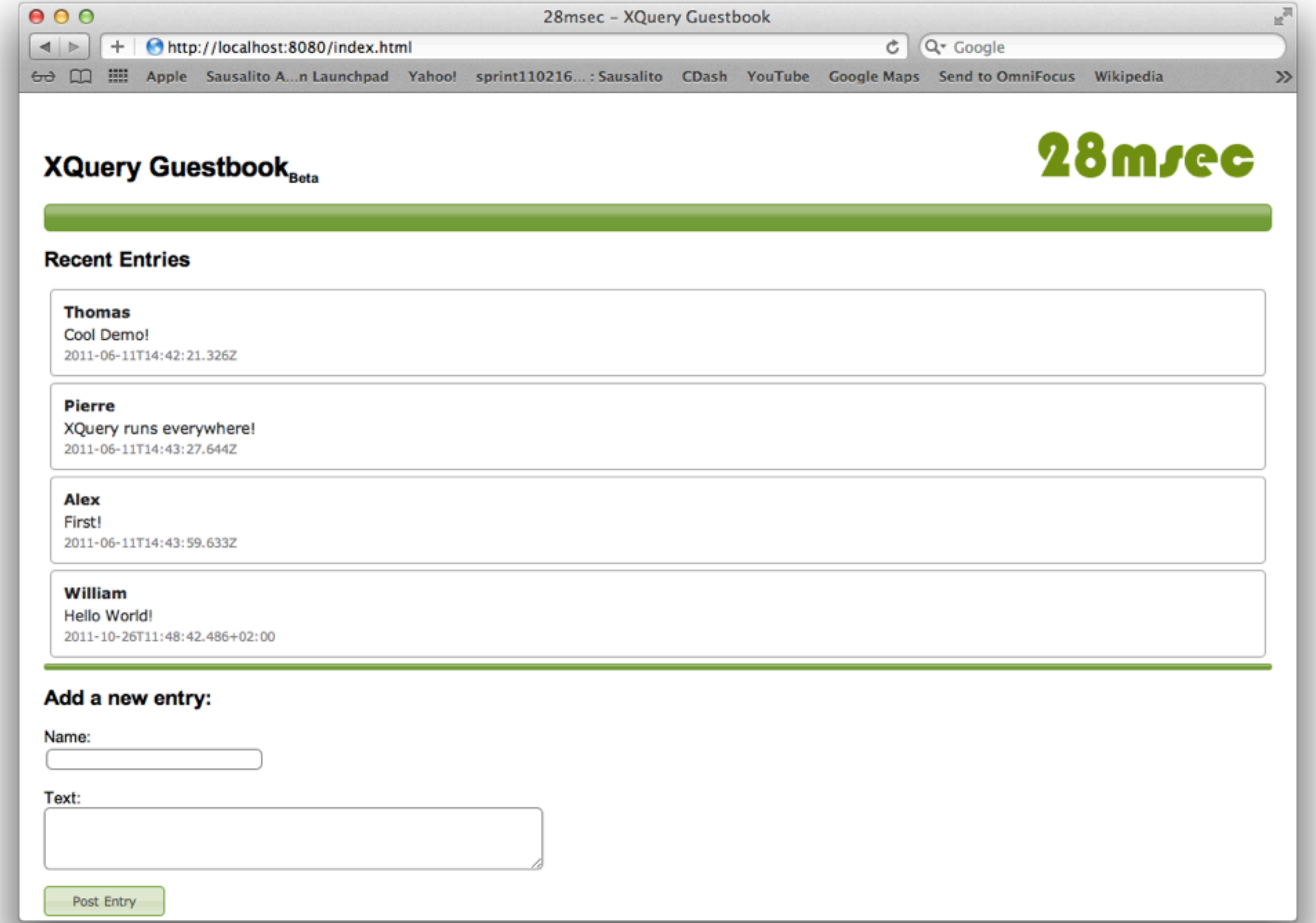
XQuery in the Browser

- Open Source Project from ETH
- http://xqib.org
- XQuery in the browser without a plug-in
- Processor compiled to JavaScript
- DOM as the processor store

```
<script type="application/xquery">
declare updating function local:onclick(
  $loc, $evtObj
) {
  insert node <hr color="red"/>
  as last into b:dom()//body
 };

b:addEventListener(
  b:dom()//input, "onclick", local:onclick#2
)
</script>
```

```
<script type="text/javascript">
foo = function (arg){
        return 'the text was '+ arg;
};
</script>

<script type="application/xquery">
   let $x := b:js-call('window.foo', “Foo”)
   return
     b:alert($x)
</script>
```

Demo


28msec – XQuery Guestbook
http://localhost:8080/index.html
Google
Apple
Sausalito A...n Launchpad
Yahoo!
sprint110216... : Sausalito
CDash
YouTube
Google Maps
Send to OmniFocus
Wikipedia
28msec

XQuery Guestbook$_{\text{Beta}}$

Recent Entries

Thomas
Cool Demo!
2011-06-11T14:42:21.326Z

Pierre
XQuery runs everywhere!
2011-06-11T14:43:27.644Z

Alex
First!
2011-06-11T14:43:59.633Z

William
Hello World!
2011-10-26T11:48:42.486+02:00

Add a new entry:

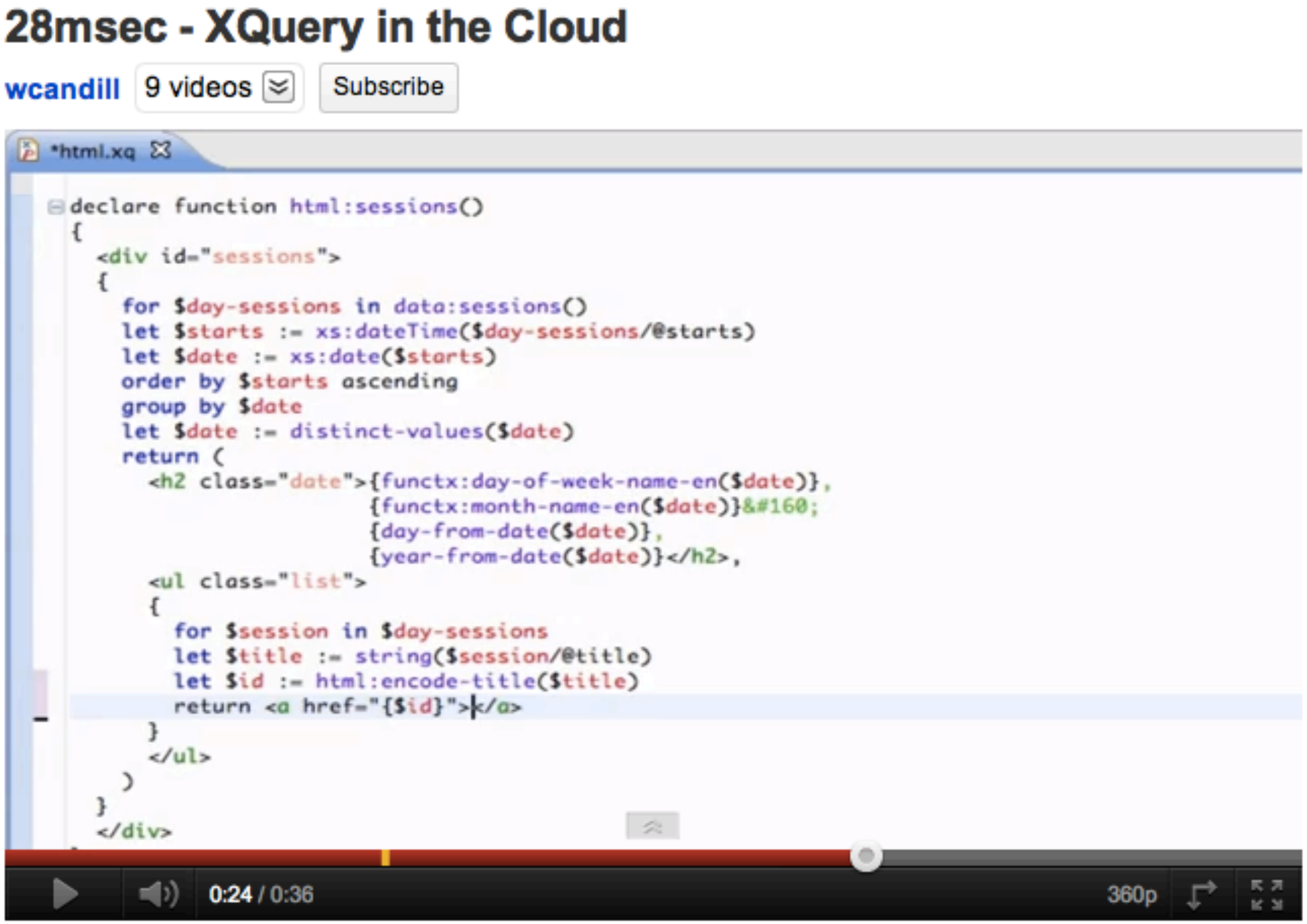
Name:

Text:

Post Entry

Wrap-Up Clip

http://www.youtube.com/watch?v=6oY5ctVHEck



28msec - XQuery in the Cloud

```
declare function html:sessions()
{
  <div id="sessions">
  {
    for $day-sessions in data:sessions()
    let $starts := xs:dateTime($day-sessions/@starts)
    let $date := xs:date($starts)
    order by $starts ascending
    group by $date
    let $date := distinct-values($date)
    return (
      <h2 class="date">{functx:day-of-week-name-en($date)},
                       {functx:month-name-en($date)}&#160;
                       {day-from-date($date)},
                       {year-from-date($date)}</h2>,
      <ul class="list">
      {
        for $session in $day-sessions
        let $title := string($session/@title)
        let $id := html:encode-title($title)
        return <a href="{$id}"></a>
      }
      </ul>
    )
  }
  </div>
```

TillWestmann

DanielTurcanu

DanielaFlorescu

SorinMarianNasoi CezarAndrei 28msec

GabrielPetrovay

MarkosZaharioudakis ChrisHillery

LuisRodriguezGonzalez DiogoSimões CarlosManuelLopez

ETH

Oracle FedericoCavalieri

JuanZacarias FLWORFoundation

MatthiasBrantner DavidGraf PaulJ.Lucas

RodolfoOchoa

NicolaeBrinza

RomanSaratz

BrunoMartins

28 msec

Thank you!