

CIC GOCDB-PI

Transfer from SQL to XPath queries

Sinikka Loikkanen

15.4.2009

Content

- Actual situation
 - gocdb.class
 - SQL queries
- PHP & XML
- CIC-GOC
 - cicgoc.class.php
 - cicgoc functions

Class GOC db (actual)

- SQL queries
- ~ 70 functions

/www/htdocs/lib/gocdb/gocdb.php :

```
class GOCdb
{
...
    function getSites()
    {
        $request = "SELECT SV.* FROM ".$this->username.".SITES_VIEW SV where
SV.SITENAME NOT LIKE 'GRIDOPS-%' order by SITENAME" ;
        $result = $this->gocdb->Execute($request);
        return $result;
    }
...
}
```

Use

- Using in the HTML forms

/www/htdocs/section_cod/page_dashboard/othertools.php :

```
echo("<form name='form_ticket' method='post' action='index.php?
section=$section&page=$page&subpage=othertools&action=sitedetails'>");

echo("<select name='\"siteid\"'>");
$i=0;

$sites=$gocdb->getSites();

for ($i=0;$i<$sites->RecordCount();$i++)
{

    $sitename=$sites->fields["SITENAME"];
    $siteid=$sites->fields["SITE_ID"];
    echo("<option value='\"$siteid\"'>$sitename</option>");
    $sites->FetchRow();
}
echo("</select>");
echo("<input type='hidden' Name='from_scratch' value='1'>");
echo("<input type='submit' value='Go'>");
echo("</form>");
```

Data manipulation

- Data management in a separate file

/www/htdocs/section_cod/page_dashboard/view_site_details.php :

```
$siteid=$_REQUEST["siteid"];

$sites=$gocdb->getSiteFromSiteid($siteid);

$sitename=$sites->fields["SITENAME"];

$sitelink="<a href=\"".$external_link_goc_portal_site_id."$siteid\" title=\"view site info for $site
in the GOC-DB\" target=_blank>".$external_link_goc_portal_site_id."$siteid</a> ";

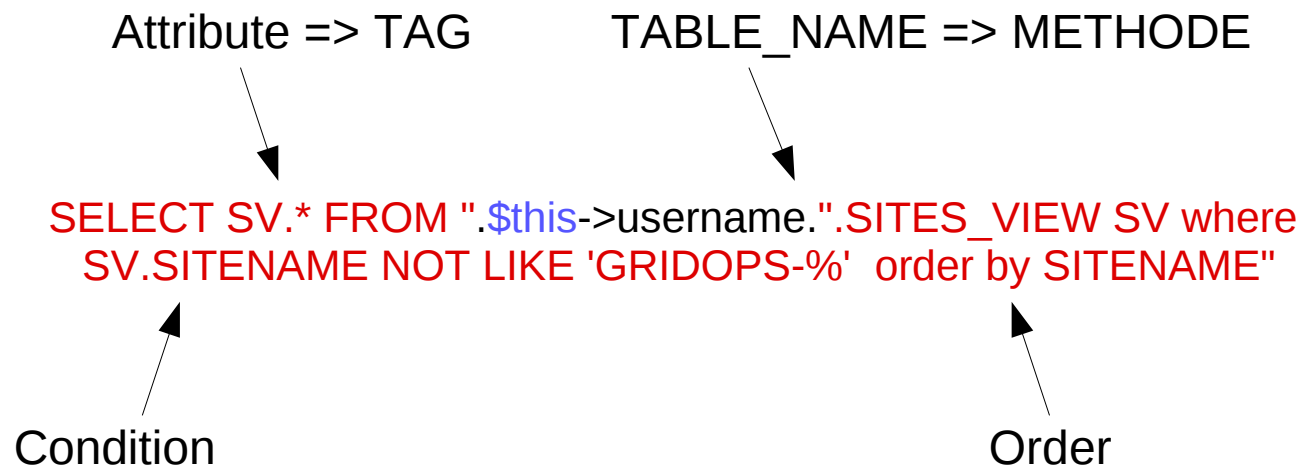
$region_info=get_site_roc_contacts($siteid);

$site_region=$region_info["rocname"];

echo("<H2>Details for $sitename</H2>");
echo("<b>Region:</b> $site_region<br>");
echo("<b>Site ID in GOC DB:</b> $siteid<br>");
echo("<b>Site page on GOC portal:</b> $sitelink<br>");
```

From SQL to Xpath-queries

Until now, normal SQL queries are used



How to translate this request to XPath queries ?

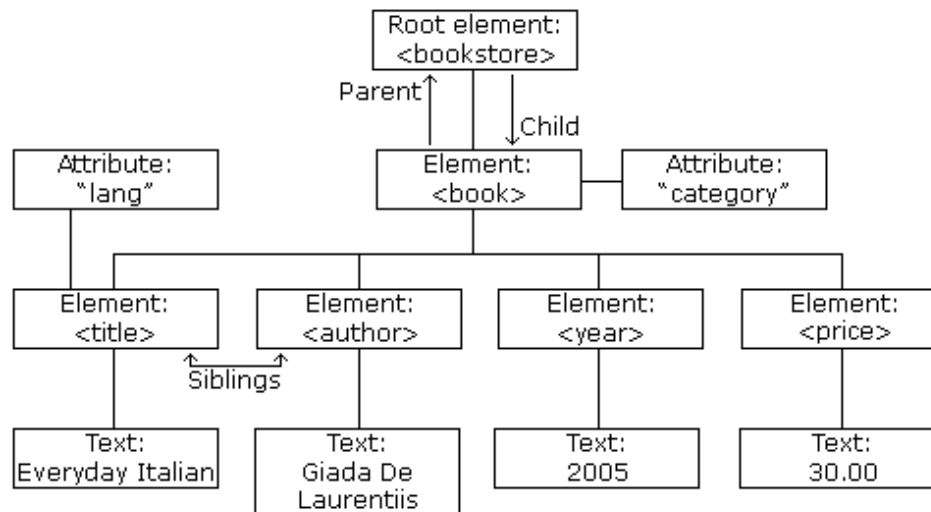
PHP & XML

- PHP can be used to parse, process, and validate XML documents
- With Simple API for XML (SAX) or with Document Object Model (DOM)
- Server-side XSL Transformations
- ...
- There is more than one way to manipulate XML data

Here, the DOM based approach is chosen

XML DOM

- XML DOM defines a standard way for accessing and manipulating XML documents.
- The DOM presents an XML document as a tree structure, with elements, attributes, and text as nodes:



<http://www.w3schools.com/dom/default.asp>

PHP XML DOM

- PHP extensions
- Object-oriented approaches
- Multiple methods:
 - *SimpleXML* extension provides a simple tool set to convert XML to an object that can be processed with normal property selectors and array iterators.
 - *DomDocument* allows to navigate and manipulate XML data
 - *XPath* allows to address parts of an XML document

SimpleXML example

XML file:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

```
<?php
$xml = simplexml_load_file("test.xml");
echo $xml->getName() . "<br />";
foreach($xml->children() as $child) {
    echo $child->getName() . ": " . $child . "<br />";
}
?>
```

Output:

```
note
to: Tove
from: Jani
heading: Reminder
body: Don't forget me this weekend!
```

See XML data as a list of elements using SimpleXML (case cicgoc)

```
if (file_exists("xml_files/$method.xml")) {
    $xml = simplexml_load_file("xml_files/$method.xml");
} else {
    exit ('Failed to open '.$method.'.xml.');
}
$n = 0;
$sites = array();

echo "<div id=\"methodsData\">";

foreach ($xml->children() as $child) {
    $n ++;
    echo "<b>".$child->getName()."_$n : ".$child."</b><br/>";
    $sites [$n] = $child;

    foreach ($child->attributes() as $attr) {
        echo "<span class=\"attrib\">+</span>".$attr->getName()." = ".$attr."<br/>";
        $sites [$attr->getName()] = $attr;
    }

    foreach ($child->children() as $subchild) {
        echo "--".$subchild->getName()." = ".$subchild."<br/>";
        $sites [$subchild->getName()] = $subchild;
    }
    echo "<hr/>";
}
echo "</div>";
}
```

The previous gives the following

```
SITE_1 :  
+ID = 11  
+NAME = CERN-CIC  
--OFFICIAL_NAME = European Organization for Nuclear Research, Geneva, Switzerland  
--CONTACT_EMAIL = grid-cern-cic-admins@cern.ch  
--CONTACT_TEL = +41227675610  
--GIIS_URL = ldap://  
--TIER = 1  
--COUNTRY = Switzerland  
--ROC = CERN  
--CERTIFICATION_STATUS = Closed  
--PRODUCTION_INFRASTRUCTURE = Production  
--TIMEZONE = Europe/Zurich  
--LATITUDE = 0  
--LONGITUDE = 0  


---

SITE_2 :  
+ID = 56  
+NAME = TORONTO-LCG2  
--OFFICIAL_NAME = University of Toronto, Ontario, Canada  
--HOME_URL = http://bigmac-www.physics.utoronto.ca/  
--CONTACT_EMAIL = lcg-admin@listserv.physics.utoronto.ca  
--CONTACT_TEL = 1-416-978-2959  
--ALARM_EMAIL = groer@physics.utoronto.ca  
--GIIS_URL = ldap://bigmac-lcg-ce2.physics.utoronto.ca:2170/mds-vo-name=toronto-lcg2,o=grid  
--TIER = 2  
--COUNTRY = Canada  
--ROC = CERN  
--CERTIFICATION_STATUS = Certified  
--PRODUCTION_INFRASTRUCTURE = Production  
--TIMEZONE = US/Eastern  
--LATITUDE = 43  
--LONGITUDE = -79  


---

SITE_3 :  
+ID = 57
```

IMG.: get_site method

DOMDocument example

XML file:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

```
<?php
$xmlDoc = new DOMDocument();
$xmlDoc->load("note.xml");

$x = $xmlDoc->documentElement;
foreach ($x->childNodes AS $item) {
    print $item->nodeName . " = " . $item->nodeValue . "<br />";
}
?>
```

Output:

```
to = Tove
from = Jani
heading = Reminder
body = Don't forget me this weekend!
```

DOMXPath query example

XML file:

```
<articles>
  <article id="1">
    ...
    <title>PHP XPath Example</title>
  </article>
  <article id="2">
    ...
    <title>DomDocument Tutorial</title>
  </article>
</articles>
```

```
<?php
$doc = new DomDocument;
$doc->load('articles.xml');

$xpath = new DOMXPath($doc);
$sarts = $xpath->query("/articles/article/title");

foreach ($sarts as $art) {
    echo $art->nodeValue."<br />";
}
?>
```

Output:

```
PHP XPath Example
DomDocument Tutorial
```

CICGOC Methods

- Methods for extract information from GOCDB-PI XML database
- Includes
 - `cicgoc.class.php`
 - Functions to
 - create HTML elements
 - make queries
 - present information

CICGOC example

Get site info by selecting a site from a list of sitenames where sitename not like 'GRIDOPS-*

AUVERGRID

AUVERGRID

- ID = 241
- OFFICIAL_NAME = IN2P3-LPC, Clermont-Ferrand, France
- HOME_URL = clrwww.in2p3.fr
- CONTACT_EMAIL = grid-admin@clermont.in2p3.fr
- CONTACT_TEL = +33 4 73 40 72 72
- GIIS_URL = ldap://obssbdii.univ-bpclermont.fr:2170/mds-vo-name=AUVERGRID,o=grid
- TIER = 0
- COUNTRY = France
- ROC = France
- CERTIFICATION_STATUS = Certified
- PRODUCTION_INFRASTRUCTURE = Production
- TIMEZONE = Europe/Paris
- LATITUDE = 0
- LONGITUDE = 0

```
<form name="getSites1" action="<?php echo $url; ?>" method="post">
  <?php
    $attrib = "NAME";
    $sites ->create_dropdown_attrib("get_site","$attrib","-- Select a site by name --",1);
  ?>
  <input type="submit" name="getSites1" value="Go"/>
</form>
<div id="result">
  <?php
    if (isset($_POST['getSites1'])) {
      $selected = $_POST["$attrib"];
      $sites ->getSiteInfo("get_site","$attrib","$selected");
    }
  ?>
```

cicgoc.class.php

```
function create_dropdown_attrib($method,$attrib,$firstentry,$size="") {  
    $xpath = new DOMXPath($this->goc);  
    $nodes = $xpath->query($query);  
    // do dropdown list  
    echo "<select name=\""$attrib\"" size=\""$size\"" >";  
    echo "<option name=\""$firstentry\"">$firstentry</option>";  
    foreach ($nodes as $item) {  
        $attr = $item->getAttribute("$attrib");  
        if ($method == "get_site" && preg_match("/^GRIDOPS-/", $attr)) {  
            continue ;  
        } else {  
            $pairs [$attr] = $attr;  
            asort ($pairs);  
        }  
    }  
    foreach ($pairs AS $value => $name) {  
        echo "<option name=\""$value\"">$name</option>";  
    }  
    echo "</select>";  
}
```

... continued

```
function getSiteInfo($method,$attrib,$selected) {  
    $xpath = new DOMXPath($this->goc);  
  
    $nodes = $xpath->query($this->path."[@.$attrib='".$selected']."'");  
    echo "<span id='rtitle'>$selected</span>";  
  
    foreach ($nodes as $item) {  
        $id = $item->getAttribute("ID");  
        $items = $item->childNodes;  
        echo "&nbsp;- ID = $id<br/>";  
  
        foreach ($items as $it) {  
            echo "&nbsp;- ".$it->nodeName." = ".$it->nodeValue."<br/>";  
        }  
    }  
}
```

CICGOC functions

```
$sites->create_dropdown_attrib("get_site","ID","-- Select a site by id --",1);
```

```
$sites->create_dropdown_element("get_site","COUNTRY","-- Select a site by country --",1);
```

```
$sites->getSitesAsTextList("get_site","$attrib");
```

```
$sites->getSiteInfo("get_site","$attrib","$selected");
```