

PHP Conference

SimpleXML: Integrando PHP e XML de maneira simples

Anderson Casimiro

Quem é esse cara?

- Coordenador de TI
- Desenvolvedor há mais de 4 anos
- Entusiasta PHP
- Áreas: Saúde, RH e Comercial
- Master PHP5 - Brainbench
- PHP, ColdFusion, Oracle, Mysql...

XML

- eXtensible Markup Language
- Compatível com SGML
- Sintaxe Simples
- Regras idem
- HTML não é XML
 - XHTML é XML

XML

- Criado para “guardar” informação
- Formato uniforme
- Independente de plataforma
- Principais usos:
 - Configuração
 - Armazenamento de dados
 - Transferência de informação uniforme
 - Arquivos de troca
 - webservices

XML: agenda.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<agenda>
  <contato id="1">
    <nome>Anderson Casimiro</nome>
    <email>anderson@duodraco.com</email>
  </contato>
  <contato id="2">
    <nome>Fulano de Tal</nome>
    <email>fulando@de.tal</email>
  </contato>
</agenda>
```

PHP+XML: Como lidar?

- Parsing (Classes e Funções)
- P.E.A.R.
 - PHP Extension and Application Repository
 - pear.php.net
- SOX (Simple API for XML)
- DOM (Document Object Model)
- SimpleXML

SimpleXML

- Nativa do PHP5
- Uso fácil e descomplicado
- Estrutura de objetos
- XPATH
- DOM
- Regras simples

4 Regras

- Propriedades: iterador de elementos
- Índices numéricos: elementos
- Índices não-numéricos: atributos
- Conversão de Strings: dados de texto

Mãos à obra

SimpleXMLElement simplexml_load_file(string arquivo[,...])

SimpleXMLElement simplexml_load_string(string dados[,...])

***retornam FALSE se falhar**

```
<?php
$SXO = simplexml_load_file("teste.xml");
print_r($SXO);
// Onde $SXO recebe uma instância da
// Classe SimpleXMLElement, referente ao
// nó raiz (agenda) do arquivo agenda.xml
?>
```

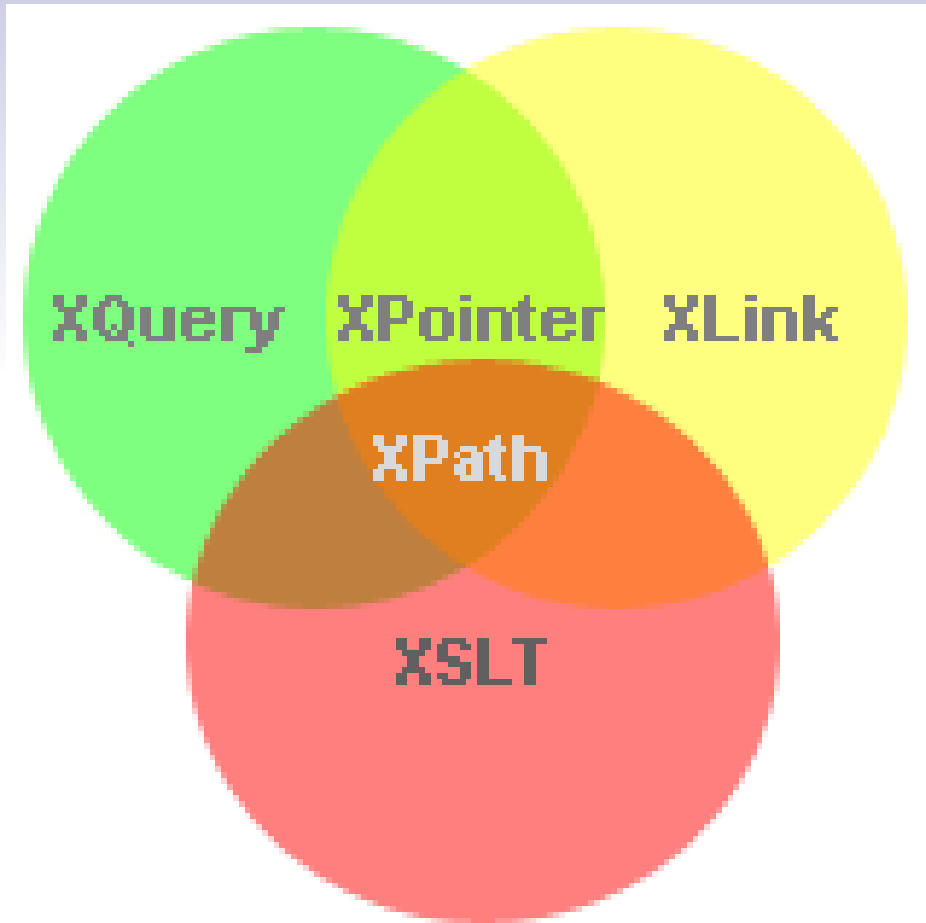
Anatomia...

```
SimpleXMLElement Object (
  [contato] => Array (
    [0] => SimpleXMLElement Object (
      [@attributes] => Array ([id] => 1)
      [nome] => Anderson Casimiro
      [email] => anderson@duodraco.com
    )
    [1] => SimpleXMLElement Object(
      [@attributes] => Array([id] => 2)
      [nome] => Fulano de Tal
      [email] => fulando@de.tal
    )
  )
)
```

Na prática...

```
<?php
$sxo = simplexml_load_file("agenda.xml");
echo $sxo->contato[0]["id"]; //1
echo $sxo->contato[0]->nome; //Anderson
Casimiro
echo $sxo->contato[1]->email;
//fulando@de.tal
?>
```

XPath



- Linguagem base p/ busca de informação
- Topo da XSLT
- XQuery e XPointer: construídos sob o XPath

XPath

```
<?php
/*...*/
//Email contato com id=2
$sxo->xpath(' //contato[@id=2]/email ');
//Email contato com nome='Anderson Casimiro'
$sxo->xpath("/agenda/contato[nome='Anderson
Casimiro']/email");
//nome do ultimo contato
$sxo->xpath("/agenda/contato[last()]/nome");
//email do contato 1
$sxo->xpath("/agenda/contato[1]/email");
?>
```

SimpleXML + DOM

SimpleXMLElement simplexml_import_dom (DOMNode nó[,...])

```
<?php
$dom = new domDocument;
$xml = <<<XML
<agenda>
    <contato>
        <nome>Anderson</nome>
    </contato>
</agenda>
XML;
$dom->loadXML();
$s = simplexml_import_dom($dom);
echo $s->contato[0]->nome; // Anderson
?>
```

Editando o XML

```
<?php
header( "Content-Type:text/plain" );

$sxo = simplexml_load_file("example.xml");
$sxo->addChild( "contato" );
$sxo->contato[2]->addChild( "name", "Livea" );
$sxo->contato[2]->addChild( "email", "--" );
$sxo->contato[2]->addAttribute( "id", "3" );

$x = $sxo->xpath( '/agenda/contato[last()]' );

print_r($x);
?>
```

Voltando ao XML

```
<?php
```

```
header("Content-Type:text/plain");
```

```
$sxo = simplexml_load_file("agenda.xml");
```

```
$sxo->contato[1]->email = "beltrano@de.tal";
```

```
$sxo->contato[1]["id"] = "fulano";
```

```
echo $sxo->contato[1]->asXML();
```

```
/*imprime:
```

```
<contato id="fulano">
```

```
    <nome>Fulano de Tal</nome>
```

```
    <email>beltrano@de.tal</email>
```

```
</contato>
```

```
*/
```

```
?>
```

O novo agenda.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<agenda>
  <contato id="1">
    <nome>Anderson Casimiro</nome>
    <email>anderson@duodraco.com</email>
  </contato>
  <contato id="fulano">
    <nome>Fulano de Tal</nome>
    <email>fulando@de.tal</email>
  </contato>
  <contato id="3">
    <nome>Livea</nome>
    <email>- -</email>
  </contato>
</agenda>
```

Nem tudo é perfeito

- SimpleXML não é uma boa saída para arquivos muito grandes
 - Peca principalmente em performance
 - Para esses casos utilize preferencialmente XMLReader e XMLWriter
- Não tem métodos nativos para remoção de elementos

Aplicação - SimpleXML

- Agenda [CRUD]
 - Inserir [Create]
 - Listar contatos [Retrieve]
 - Editar [Update]
 - Excluir [Delete]
- Contato
 - nome
 - endereço
 - telefone

Até que enfim...

Duvidas?

Referência

www.php.net
www.wikipedia.com
www.xmlbr.com.br
www.w3schools.com

PHP5 – Programação Poderosa
Gutmans-Bakken-Rethans

Autor

Anderson de Arruda Casimiro

www.duodraco.com

anderson@duodraco.com

+55 11 9465.3590

