

eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Gilles Guirand – Technical Director at Kaliop



KALIOP Interactive Media
Arche Jacques Coeur
222, place Ernest Granier
34000 Montpellier
France

Phone:
+33 4 99 13 68 60



KALIOP Interactive Media, France

Platinum Partner



Kaliop is a company of expert data-processing engineering in online solutions. Its team of fifteen collaborators is made up engineers specialists in technologies of the Internet. Specialized in the environment, sustainable development and the new solidarity economy, she conceives and implements complete solutions for sites, extranet, Intranet and application one line.

References: *Week end Esprit de Picardie, BrittanyTourism.com, CIRAD - Agricultural Research for Development, Institut de Recherche pour le développement, Aix en bus*

Platinum Partner

eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Summary :

Chapter 1 : Introduction about eZ Find

Chapter 2 : Speed-up eZ Find development tasks

Chapter 3 : Fields & Datatypes in Solr and eZ Find

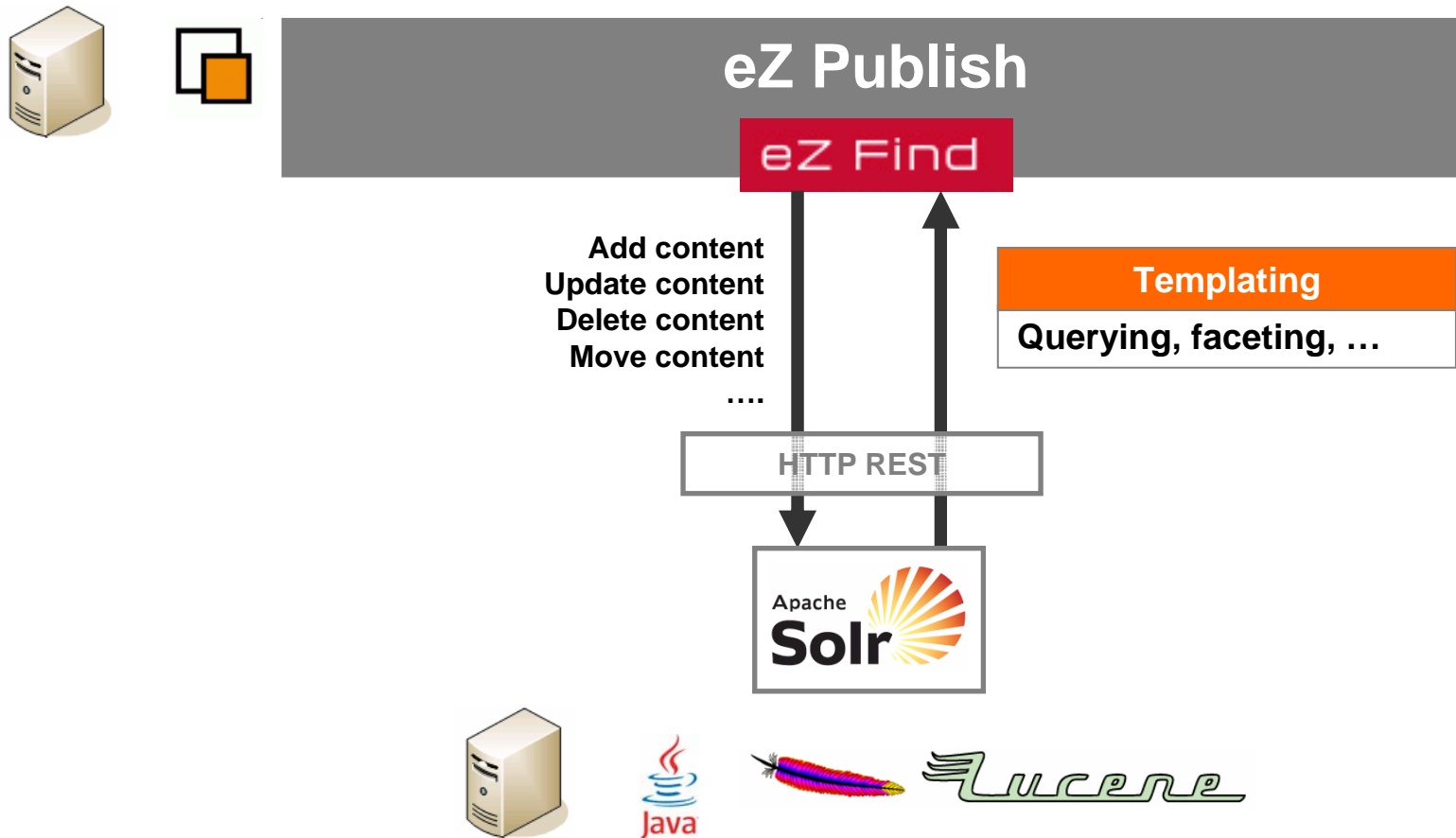
Chapter 4 : Indexing additional fields in Solr

Chapter 5 : Enhance eZ Find using the Solr syntax

Chapter 6 : Conclusion

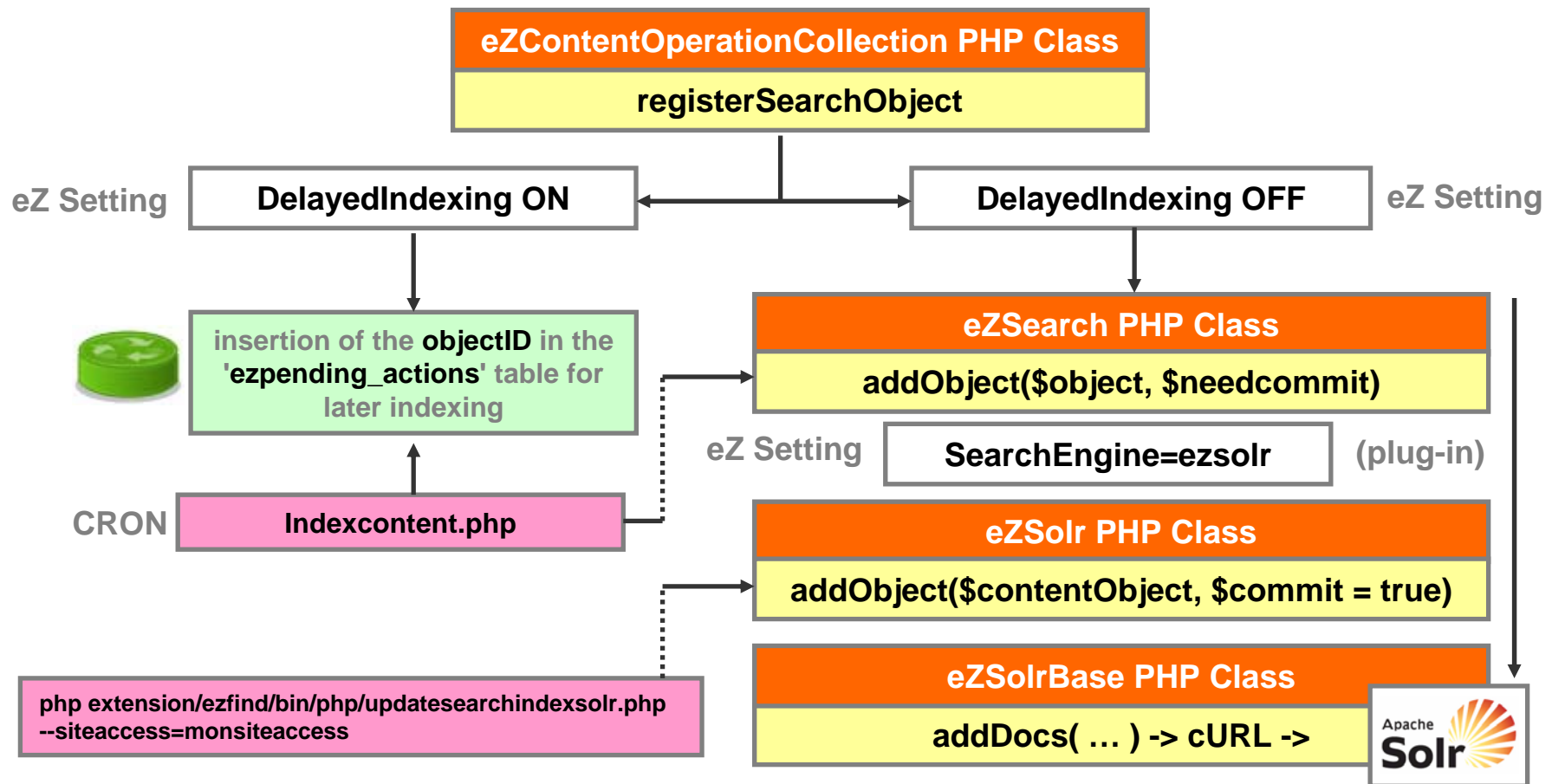
Chapter 1 : Introduction about eZ Find

► How eZ Find work



Chapter 1 : Introduction about eZ Find

- How eZ Find pushes content from eZ Publish to Solr



eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Summary :

Chapter 1 : Introduction about eZ Find

Chapter 2 : **Speed-up eZ Find development tasks**

Chapter 3 : Fields & Datatypes in Solr and eZ Find

Chapter 4 : Indexing additional fields in Solr

Chapter 5 : Enhance eZ Find using the Solr syntax

Chapter 6 : Conclusion

Chapter 2 : Speed-up eZ Find development tasks

- ▶ Re-index some content after a modification
- ▶ Re-indexing all content to merely test one single, minor modification's impact on one's application can quickly become a drawn-out process. A few, hidden, life-saver arguments exist in the concerned script **/bin/php/updatesearchindexsolr.php**, allowing for pointing to:
 - a root node
 - an offset
 - a limit
- ▶ It is mandatory to use the 3 parameters simultaneously :

```
phpextension/ezfind/bin/php/updatesearchindexsolr.php  
--siteaccess=mysiteaccess --topNodeID=2546 --offset=0 --limit=10
```

Chapter 2 : Speed-up eZ Find development tasks

► Check what Solr actually indexed

Solr Admin (example)

gandbox-desktop:8983
 cwd=/home/gandbox/www/myblog/extension/ezfind/java SolrHome=solr/./

Solr [SCHEMA] [CONFIG] [ANALYSIS] [SCHEMA BROWSER]
 [STATISTICS] [INFO] [DISTRIBUTION] [PING] [LOGGING]

App server: [JAVA PROPERTIES] [THREAD DUMP]

Make a Query [FULL INTERFACE]

Query String:

Assistance [DOCUMENTATION] [ISSUE TRACKER] [SEND EMAIL]
 [SOLR QUERY SYNTAX]

Current Time: Sun May 09 13:31:10 CEST 2010
 Server Start At: Sun May 09 11:42:48 CEST 2010

```

-<response>
  -<lst name="responseHeader">
    <int name="status">0</int>
    <int name="QTime">2</int>
    -<lst name="params">
      <str name="indent">on</str>
      <str name="start">0</str>
      <str name="q">test</str>
      <str name="version">2.2</str>
      <str name="rows">10</str>
    </lst>
  </lst>
  -<result name="response" numFound="83" start="0">
    -<doc>
      -<arr name="attr_author_s">
        <str>tset</str>
      </arr>
      -<arr name="attr_author_t">
        <str>tset</str>
      </arr>
      -<arr name="attr_contact_t">
        <str>test@test.fr</str>
      </arr>
      -<arr name="attr_text_t">
        <str>test</str>
      </arr>
      -<arr name="attr_title_s">
        <str>test test test</str>
      </arr>
    </doc>
  </result>

```

Fieldname

Value

<http://localhost:8983/solr/admin/>

Chapter 2 : Speed-up eZ Find development tasks

► Debugging directly in Solr

```

gandbox@gandbox-desktop: ~/www/myblog/extension/ezfind/java
Fichier Edition Affichage Terminal Aide
e t+attr appellations s+attr author t+attr contact t+attr couleur s+attr intro t+attr photo s+attr rating s+attr subtitle t+attr tags lk+attr text t+at
r texte t+attr title t+attr todrink s+attr url t+attr zoom t+meta name t+meta owner name t&hl.fl=attr adresse t+attr appellations s+attr author t+attr
contact t+attr couleur s+attr intro t+attr photo s+attr rating s+attr subtitle t+attr tags lk+attr text t+attr title t+attr todrink s+at
r url t+attr zoom t&wt=phpspellcheck.collate=true&rows=10&spellcheck.onlyMorePopular=true&hl.snippets=1&facet.sort=false&facet.sort=false&start=0&q=
test&spellcheck.dictionary=default} hits=83 status=0 QTime=16
9 mai 2010 13:19:38 org.apache.solr.core.SolrCore execute
INFO: [] webapp=/solr path=/select params={facet=true&enableElevation=false&sort=score+desc&facet.limit=20&facet.limit=20&hl.simple.pre=<b>&hl=true&ve
rsion=2.26&q=meta installation id s:a78872aec0d46b773cbd03ae2ce25508*1.5+meta language code s:fre-FR*1.2&fl=meta guid s+meta installation id s+meta ma
in url alias s+meta installation url s+meta id si+meta main node id si+meta language code s+meta name t+score+meta published dt+meta path string s&for
ceElevation=false&hl.simple.post=</b>&facet.field=subattr date-year dt&facet.field=subattr date-yearmonth dt&qt=ezipublish&fq=meta path si:2&fq=(+meta
installation id s:a78872aec0d46b773cbd03ae2ce25508)+AND+(+meta language code s:fre-FR+)+AND+meta is invisible b:false&fq=meta contentclass id si:24
&fq=meta language code s:fre-FR&hl.requireFieldMatch=false&hl.fragsize=200&facet.mincount=1&facet.mincount=1&facet.offset=0&facet.offset=0&indent=on&
f+attr adresse t+attr intro t+attr photo s+attr tags lk+attr text t+attr title t+attr todrink s+meta name t+meta owner name t&hl.fl=attr adresse t+at
r intro t+attr photo s+attr tags lk+attr text t+attr title t+attr todrink s&wt=php&rows=10&hl.snippets=1&facet.sort=false&facet.sort=false&start=0&q=}
hits=38 status=0 QTime=2
9 mai 2010 13:20:38 org.apache.solr.core.SolrCore execute
INFO: [] webapp=/solr path=/select params={spellcheck=true&facet=true&enableElevation=true&sort=score+desc&facet.limit=30&facet.limit=30&facet.limit=3
0&spellcheck.q=test&hl.simple.pre=<b>&hl=true&version=2.26&q=meta installation id s:a78872aec0d46b773cbd03ae2ce25508*1.5+meta language code s:fre-FR*1
.2&fl=meta guid s+meta installation id s+meta main url alias s+meta installation url s+meta id si+meta main node id si+meta language code s+meta name
t+score+meta published dt+meta path string s&forceElevation=false&hl.simple.post=</b>&facet.field=meta contentclass id si&facet.field=subattr rating-n
ame s&facet.field=attr tags lk&spellcheck.count=1&qt=ezipublish&fq=(+meta installation id s:a78872aec0d46b773cbd03ae2ce25508)+AND+(+meta language code
s:fre-FR+)+AND+meta is invisible b:false&fq=meta contentclass id si:24+OR+meta contentclass id si:27+OR+meta contentclass id si:26+OR+meta contentc
lass id si:28+OR+meta contentclass id si:24+OR+meta contentclass id si:13&fq=meta language code s:fre-FR&hl.requireFieldMatch=false&hl.fragsize=200&fa
cet.mincount=1&facet.mincount=1&facet.mincount=1&facet.offset=0&facet.offset=0&facet.offset=0&indent=on&spellcheck.extendedResults=true&fq=attr adresse
e t+attr appellations s+attr author t+attr contact t+attr couleur s+attr intro t+attr photo s+attr rating s+attr subtitle t+attr tags lk+attr text t+at
r texte t+attr title t+attr todrink s+attr url t+attr zoom t+meta name t+meta owner name t&hl.fl=attr adresse t+attr appellations s+attr author t+attr
contact t+attr couleur s+attr intro t+attr photo s+attr rating s+attr subtitle t+attr tags lk+attr text t+attr title t+attr todrink s+at
r url t+attr zoom t&wt=phpspellcheck.collate=true&rows=10&spellcheck.onlyMorePopular=true&hl.snippets=1&facet.sort=false&facet.sort=false&start=0&q=}
hits=38 status=0 QTime=17
9 mai 2010 13:20:39 org.apache.solr.core.SolrCore execute
INFO: [] webapp=/solr path=/select params={facet=true&enableElevation=false&sort=score+desc&facet.limit=20&facet.limit=20&hl.simple.pre=<b>&hl=true&ve
rsion=2.26&q=meta installation id s:a78872aec0d46b773cbd03ae2ce25508*1.5+meta language code s:fre-FR*1.2&fl=meta guid s+meta installation id s+meta ma
in url alias s+meta installation url s+meta id si+meta main node id si+meta language code s+meta name t+score+meta published dt+meta path string s&for
ceElevation=false&hl.simple.post=</b>&facet.field=subattr date-year dt&facet.field=subattr date-yearmonth dt&qt=ezipublish&fq=meta path si:2&fq=(+meta
installation id s:a78872aec0d46b773cbd03ae2ce25508)+AND+(+meta language code s:fre-FR+)+AND+meta is invisible b:false&fq=meta contentclass id si:24
&fq=meta language code s:fre-FR&hl.requireFieldMatch=false&hl.fragsize=200&facet.mincount=1&facet.mincount=1&facet.offset=0&facet.offset=0&indent=on&
f+attr adresse t+attr intro t+attr photo s+attr tags lk+attr text t+attr title t+attr todrink s+meta name t+meta owner name t&hl.fl=attr adresse t+at
r intro t+attr photo s+attr tags lk+attr text t+attr title t+attr todrink s&wt=php&rows=10&hl.snippets=1&facet.sort=false&facet.sort=false&start=0&q=}
hits=38 status=0 QTime=2

```

Active console opened

Copy MESSAGE

INFO: [] webapp=/solr path=/select
params={ ... MESSAGE ... } status=400
QTime=5

Past MESSAGE : <http://localhost:8983/solr/select/?MESSAGE>

The obtained result is the exact output sent by Solr to eZ Find before transformation and display of the results. Using this trick is pretty useful when debugging, by, for example, directly manipulating the messages to retrieve the expected result.

eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Summary :

Chapter 1 : Introduction about eZ Find

Chapter 2 : Speed-up eZ Find development tasks

Chapter 3 : **Fields & Datatypes in Solr and eZ Find**

Chapter 4 : Indexing additional fields in Solr

Chapter 5 : Enhance eZ Find using the Solr syntax

Chapter 6 : Conclusion

Chapter 3 : Fields & Datatypes in Solr and eZ Find

► The Solr-side naming of fields



eZ Publish

eZ Find

Indexing process

Attribute Name : **Title** | Attribute datatype : **ezstring** | Class Name : **Article**

ezfind.ini

```
DatatypeMap[ezstring]=text  
DatatypeMapSort[ezstring]=string  
DatatypeMapFacet[]  
DatatypeMapFilter[]
```

attr_[contentattributename]_[contentattributetype]

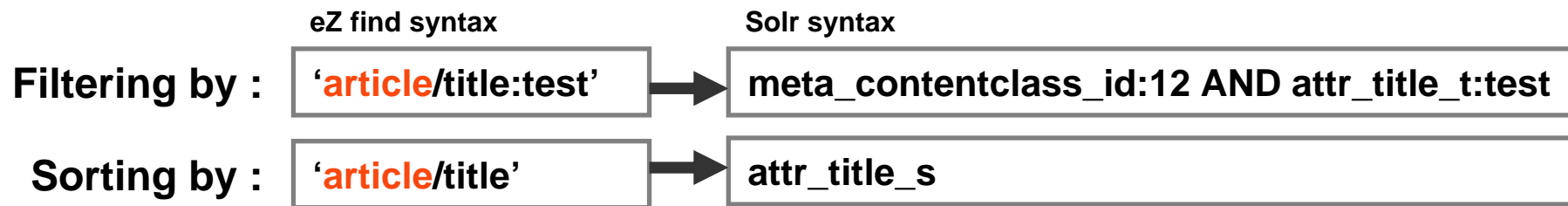


Solr Field 1 : **attr_title_s**
Solr Field 2 : **attr_title_t**

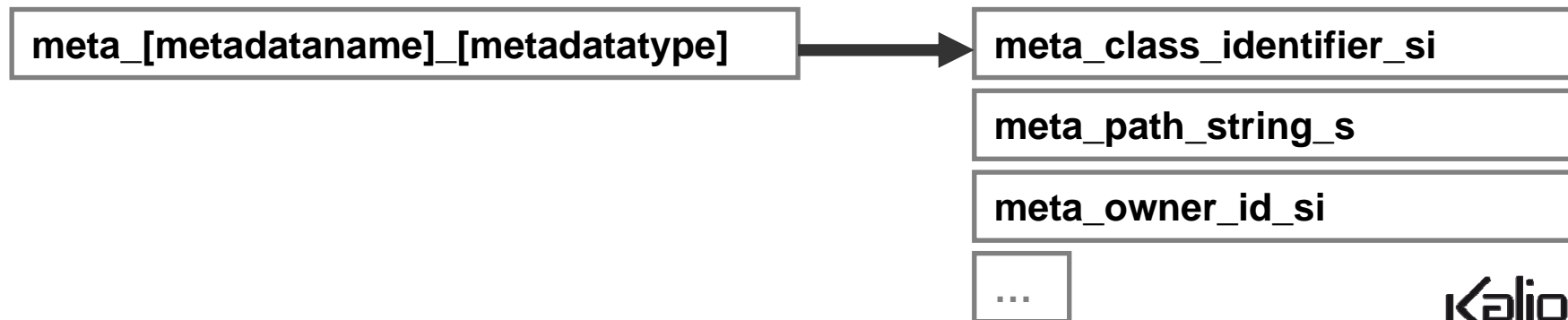
Note the absence of the content class identifier, opening for nice perspectives like filtering on several content classes having identical names

Chapter 3 : Fields & Datatypes in Solr and eZ Find

► The Solr-side naming of fields



► Metadata fields mapping



Chapter 3 : Fields & Datatypes in Solr and eZ Find

► Sub-attributes names (not used by default)

```
subattr_[contentattributename]-[contentsubattributename]_[contentsubattributetype]
```

► Sub-attributes usage ?

Natively, the subattribute concept is not or little used, because the standard state and features of eZ Publish does not require it massively. **It however is here as an opening for advanced usages, and it is, for instance, a great tool to extend eZ Find and index additional fields.**

Example : subattr_relatedimage-alttext_s

Index your images metadata fields

Example : subattr_relatedobject-title_s

Index all attributes of an object's related objects

Chapter 3 : Fields & Datatypes in Solr and eZ Find

ezfsolrdocumentfieldobjectrelation

Last updated: Monday 19 April 2010 20:57

UNIX name	Owner	Status	Version	Compatible with
ezfsolrdocumentfieldobjectrelation	gilles guirand	beta	0.1	4.x

enhanced the default ezfind/classes/ezfsolrdocumentfieldobjectrelation.php

What's for ?

This PHP class enhance the default **ezfind/classes/ezfsolrdocumentfieldobjectrelation.php**

The default current ezfind/classes/ezfsolrdocumentfieldobjectrelation.php is not usable for objet relation(s) faceting. This PHP class enable objet relation(s) faceting using a specific attribute or the 'name' attribute

Template using

```
{set search=fetch( ezfind , search ,
    hash( query , $search_text,
        'facet', array(
            hash('field', 'myclass/myattribute/mysubattribute')
        )
    )
)}
```

OR

```
{set search=fetch( ezfind , search ,
    hash( query , $search_text,
        'facet', array(
            hash('field', 'myclass/myattribute/name')
        )
    )
)}
```

Object relation datatype mapping

Check my contribution :

[ezfsolrdocumentfieldobjectrelation](#), indexing all attributes of an object's related objects, storing them as subattributes.

This then opens for applying all sorts of operations to there subattributes (search, filtering, faceting), using the **'myclass/myattribute/mysubattribute'** syntax.

Chapter 3 : Fields & Datatypes in Solr and eZ Find

► Field type management on the Solr side

Open and read the file : `/ezfind/java/solr/conf/schema.xml`.

This configuration file contains the hard-coded definition for a certain amount of fields (metadata fields for instance), but also defines the so-called **dynamic fields** :

Solr relies on several configuration files, one of them is used to tell him that :

- `'_s'` at the end of a field name means **string**,
- `'_t'` for **text**, etc.

```
<dynamicField name="*_i" type="int" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_f" type="float" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_d" type="double" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_si" type="sint" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_sf" type="sfloat" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_sd" type="sdouble" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_s" type="string" indexed="true" stored="true" multiValued="true" termVectors="true"/>
<dynamicField name="*_sl" type="slong" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_l" type="long" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_t" type="text" indexed="true" stored="true" multiValued="true" termVectors="true"/>
<dynamicField name="*_b" type="boolean" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_dt" type="date" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_random" type="random" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_k" type="keyword" indexed="true" stored="true" multiValued="true"/>
<dynamicField name="*_lk" type="lkeyword" indexed="true" stored="true" multiValued="true"/>
<!-- some trie-coded dynamic fields for faster range queries -->
<dynamicField name="*_ti" type="tint" indexed="true" stored="true"/>
<dynamicField name="*_tl" type="tlong" indexed="true" stored="true"/>
<dynamicField name="*_tf" type="tfloat" indexed="true" stored="true"/>
<dynamicField name="*_td" type="tdouble" indexed="true" stored="true"/>
<dynamicField name="*_tdt" type="tdate" indexed="true" stored="true"/>
<!-- geopoint for geospatial/location searches, boosting, ... -->
<dynamicField name="*_gpt" type="geopoint" indexed="true" stored="true"/>
```

Chapter 3 : Fields & Datatypes in Solr and eZ Find

Field type management on the Solr side

- This files can also be used to define more complex behaviours for given eZ Publish datatypes, like the keywords datatype (ezkeyword). Two different field types definitions can be found
 - **'keyword'** for case-sensitive cases
 - **'lkeyword'** for lower-case cases
- This example, keywords fields management, teaches a lot about Solr configuration. One can note the way Solr filters are called, how coma-based word separations are handled (PatternTokenizerFactory), case-sensitivity management (LowerCaseFilterFactory), duplicate removal (RemoveDuplicatesTokenFilterFactory), etc.

```
<fieldtype name="lkeyword" class="solr.TextField" positionIncrementGap="100">
  <analyzer type="index">
    <tokenizer class="solr.PatternTokenizerFactory" pattern=", *" />
    <filter class="solr.TrimFilterFactory" />
    <filter class="solr.StopFilterFactory" ignoreCase="true" words="stopwords.txt"/>
    <filter class="solr.LowerCaseFilterFactory"/>
    <filter class="solr.RemoveDuplicatesTokenFilterFactory"/>
  </analyzer>
  <analyzer type="query">
    <tokenizer class="solr.PatternTokenizerFactory" pattern=", *" />
    <filter class="solr.TrimFilterFactory" />
    <filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt" ignoreCase="true" expand="true"/>
    <filter class="solr.StopFilterFactory" ignoreCase="true" words="stopwords.txt"/>
    <filter class="solr.LowerCaseFilterFactory"/>
    <filter class="solr.RemoveDuplicatesTokenFilterFactory"/>
  </analyzer>
</fieldtype>
```

eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Summary :

Chapter 1 : Introduction about eZ Find

Chapter 2 : Speed-up eZ Find development tasks

Chapter 3 : Fields & Datatypes in Solr and eZ Find

Chapter 4 : Indexing additional fields in Solr

Chapter 5 : Enhance eZ Find using the Solr syntax

Chapter 6 : Conclusion

Chapter 4 : Indexing additional fields in Solr

- Presenting the case : Develop a year and year & month filter with eZ Find

The screenshot shows a web interface for archives. It is divided into two sections: 'ARCHIVES PAR ANNÉES :' and 'ARCHIVES PAR MOIS / ANNÉES :'. The first section lists years from 2010 to 2008. The second section lists months from May 2010 down to December 2008. A mouse cursor is hovering over 'Janvier 2010', which has opened a sub-menu showing 'Archives : janvier 2010' with '2 Billet(s)'. Two yellow callout boxes with arrows point to the interface: one points to the 'March' entry (which is not visible in the list) with the text 'Hide empty month (March : 0 items)', and the other points to the '2 Billet(s)' text with the text 'Items count'.

Usually in this case, a [template operator](#) is developed which builds the appropriate SQL queries. This can quickly become complicated, and often has sharp limitations ([eZArchive](#))

eZ Archive classical SQL limitation : Only the 'publication_date' parameter is taken into account, and no room is left for using a content-class-specific date attribute

Chapter 4 : Indexing additional fields in Solr

► Indexing Year and Year/Month values in Solr

Create your own PHP class to manage the '**ezdate**' datatype

eZ Find settings ([ezfind.ini](#), to be overridden in the ezfind.ini.append.php file of your extension) allow for delegating the indexing process of an eZ Publish datatype to a given PHP class

```
[SolrFieldMapSettings]
CustomMap[ezdate]=ezfSolrDocumentFieldDate
```

/extension/myextension/classes/ezfsolrdocumentfielddate.php

```
<?php
class ezfSolrDocumentFieldDate extends ezfSolrDocumentFieldBase ← extends ezfSolrDocumentFieldBase
{
    public static function getFieldname( eZContentClassAttribute $classAttribute, $subAttribute = null, $context = 'search' )
    {
        // return the fieldname like : attr_mydate_d
    }

    public function getData()
    {
        // return the array keys (fieldname => value), like : array('attr_mydate_dt' => '2010-04-30T00:00:00Z')
    }
}
?>
```

Chapter 4 : Indexing additional fields in Solr

► Indexing Year and Year/Month values in Solr

Role of the getFieldName() method

This method is invoked the attributes names (within eZ Find) to Solr field names. For instance, when building a facet using the following syntax : 'mycontentclass/mydateattribute', this method should return 'attr_mydateattribute_dt'.

```

const DEFAULT_SUBATTRIBUTE_TYPE = 'date';

public static function getFieldName( eZContentClassAttribute $classAttribute, $subAttribute = null, $context = 'search' )
{
    switch ( $classAttribute->attribute( 'data_type_string' ) )
    {
        case 'ezdate' :
        {
            if ( $subAttribute and $subAttribute !== '' )
            {
                // A subattribute was passed
                return parent::generateSubattributeFieldName( $classAttribute,
                    $subAttribute,
                    self::DEFAULT_SUBATTRIBUTE_TYPE );
            }
            else
            {
                // return the default field name here.
                return parent::generateAttributeName( $classAttribute, self::getClassAttributeType( $classAttribute, null, $context ) );
            }
        } break;

        default:
        {} break;
    }
}

```

Important : to make sure the written code is generic enough, and avoid hard-coding the Solr field names, we will use the handy **generateSubattributeFieldName** and **generateAttributeName** methods.

Chapter 4 : Indexing additional fields in Solr

► Indexing Year and Year/Month values in Solr

```

public function getData()
{
    $contentClassAttribute = $this->ContentObjectAttribute->attribute( 'contentclass_attribute' );

    switch ( $contentClassAttribute->attribute( 'data_type_string' ) )
    {
        case 'ezdate' :
        {
            $returnArray = array();

            // Get timestamp attribute value
            $value = $this->ContentObjectAttribute->metaData();

            // Generate the main fileName attr_XXX_dt
            $fieldName = parent::generateAttributeFieldName( $contentClassAttribute,
                self::DEFAULT_ATTRIBUTE_TYPE );
            $returnArray[$fieldName] = parent::convertTimestampToDate( $value );

            // Generate the yearmonth subattribute fileName subattr_year_dt
            $fieldName = parent::generateSubattributeFieldName( $contentClassAttribute,
                'year',
                self::DEFAULT_SUBATTRIBUTE_TYPE );

            $year = date("Y", $value); // Get Year value : 2010
            $returnArray[$fieldName] = parent::convertTimestampToDate( strtotime($year.'-01-01') );

            // Generate the yearmonth subattribute fileName subattr_yearmonth_dt
            $fieldName = parent::generateSubattributeFieldName( $contentClassAttribute,
                'yearmonth',
                self::DEFAULT_SUBATTRIBUTE_TYPE );

            $month = date("n", $value); // Get Month value : 3
            $returnArray[$fieldName] = parent::convertTimestampToDate( strtotime($year.'-'. $month.'-01') );

            return $returnArray;
        }
        break;
        default:
        {
            break;
        }
    }
}

```

Role of the getData() method

This method is invoked to extract data from eZ Publish, and prepare it prior to indexing in Solr. This method is the place to add additional fields like **'year'** et **'yearmonth'**.

'mycontentclass/mydateattribute/year',
translated in Solr under :
'subattr_date-year_dt'

'mycontentclass/mydateattribute/yearmonth',
translated in Solr under :
'subattr_date-yearmonth_dt'

Chapter 4 : Indexing additional fields in Solr

► Indexing Year and Year/Month values in Solr

Building the facet navigation through a template

```

{def $search_yearmonth=fetch( ezfind, search,
  hash( 'query' , '',
    'facet', array(
      hash('field', 'billet/date/year',
        'sort', 'alpha',
        'limit', 20 ),
      hash('field', 'billet/date/yearmonth',
        'sort', 'alpha',
        'limit', 20 )
    ),
    'class_id', array('billet'),
    'subtree_array', array(2)
  )
)
}

{def $search_extras_year=$search_yearmonth['SearchExtras'].facet_fields[0].nameList|reverse}
{def $search_extras_yearmonth=$search_yearmonth['SearchExtras'].facet_fields[1].nameList|reverse}
{def $date_count = 0
  $date_ts = 0}

```

Diagram illustrating the mapping of facet fields to subattributes:

- `billet/date/year` is mapped to `subattr_date-year_dt`.
- `billet/date/yearmonth` is mapped to `subattr_date-yearmonth_dt`.

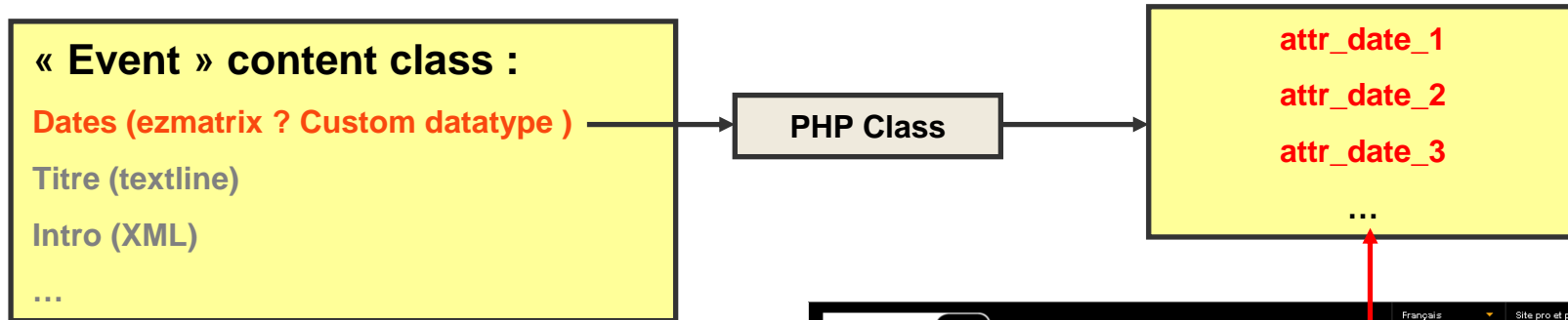
Note : The 'sort', 'alpha' statement does not actually specify an alphabetical sort. It rather helps specifying that no 'count' sort should occur (number of items matching a given facet). In this case, Solr automatically uses an 'increasing' sort, based on its index and the datatype of the concerned field (this explains the usage of the [reverse](#) operator to get an 'increasing' list).

You could manage year / month faceting using eZ Find / Solr in another way. This use case help to learn how manipulate subattributes and additional fields.

Chapter 4 : Indexing additional fields in Solr

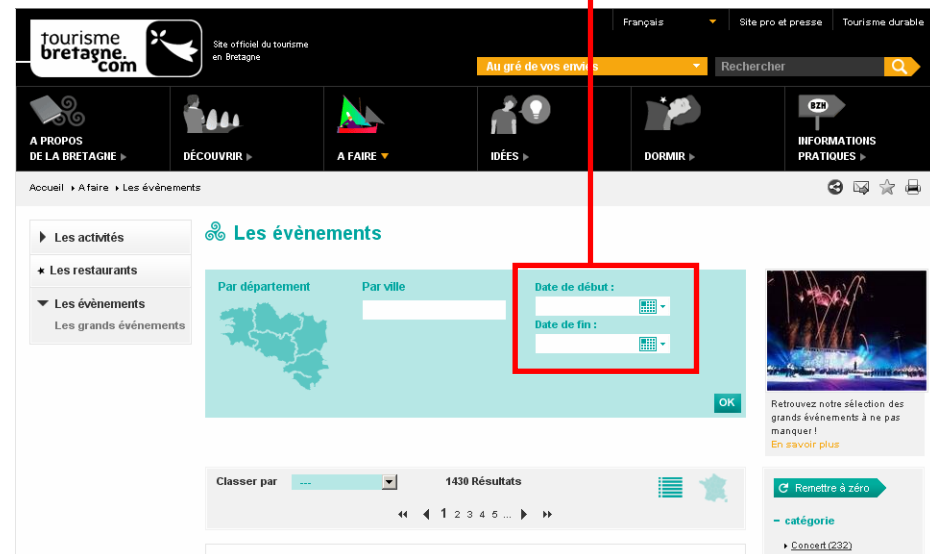
► Other real life exemple

Filtering events using a date range, (multi dates event... each month event)



Front_end filtering

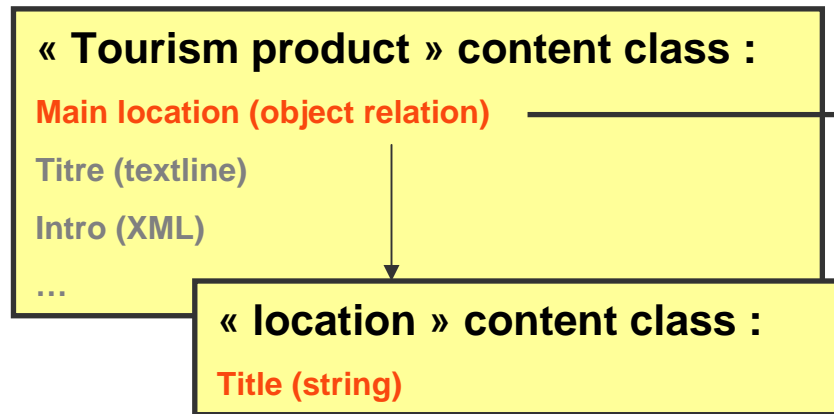
<http://www.tourismebretagne.com>



Chapter 4 : Indexing additional fields in Solr

Other real life exemple

Faceting on object(s) relation subattribute



<http://projects.ez.no/ezfsolrdocumentfieldobjectrelation>



Front_end filtering
<http://www.tourismebretagne.com>



eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Summary :

Chapter 1 : Introduction about eZ Find

Chapter 2 : Speed-up eZ Find development tasks

Chapter 3 : Fields & Datatypes in Solr and eZ Find

Chapter 4 : Indexing additional fields in Solr

Chapter 5 : Enhance eZ Find using the Solr syntax

Chapter 6 : Conclusion

Chapter 5 : Enhance eZ Find using the Solr syntax

- Is it possible to mix eZ Find and Solr syntax ?

YES, but don't think about it : BAD PRACTICE. This potentially endangering the lower layer's evolutivity (Solr), and your own project evolutivity

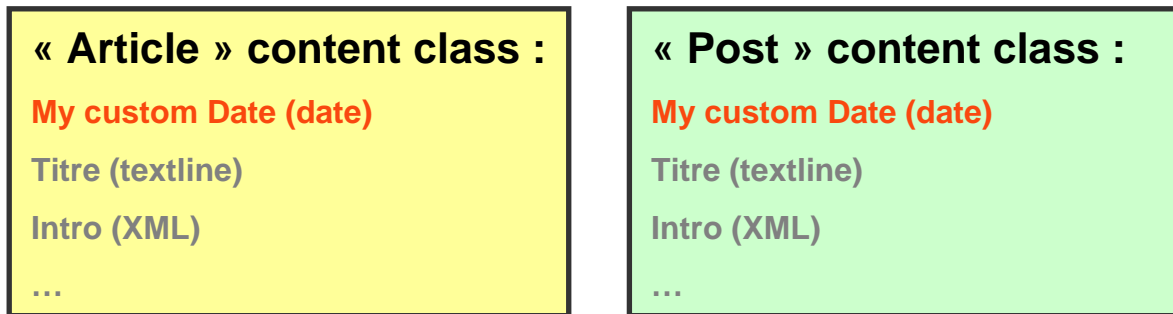
YES, it could make my day (and save my project)

```
{def $search=fetch( ezfind, search,  
  hash( query , '',  
        'class_id', array('post', 'article'),  
        'limit', 10,  
        'sort_by', hash('attr_date_dt', 'desc')  
  )  
)}
```

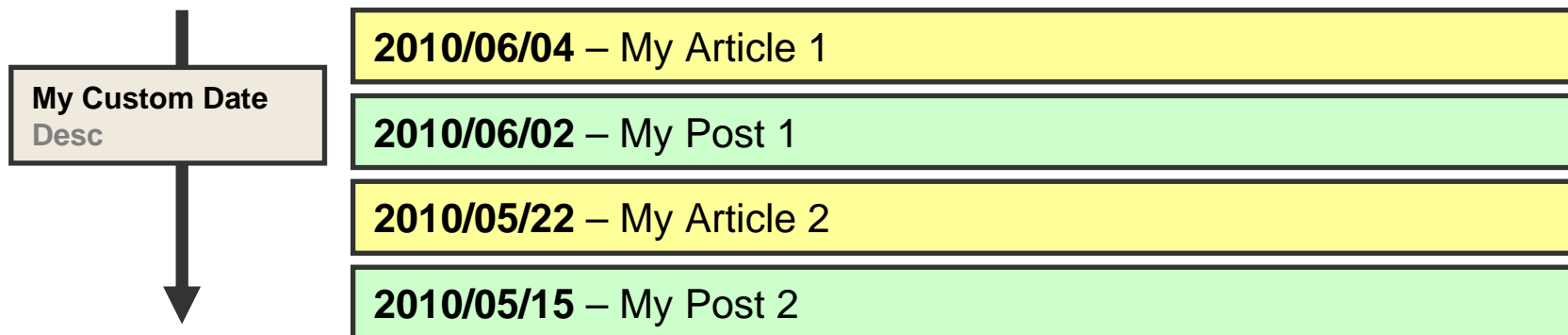
Solr field name

Chapter 5 : Enhance eZ Find using the Solr syntax

- Resolve a common issue : Sort an attribute present in several content classes



Fetching and mixing « article » and « news » content result :



Good luck without using eZ Find : custom Fetch, amazing SQL...

Chapter 5 : Enhance eZ Find using the Solr syntax

► Resolve a common issue : Sort an attribute present in several content classes

The solution, using eZ Find :

Remember ! The fortunate absence of the content class identifier in the field name means we can leverage this homonymy as we wish, through searches, filters or sorts depending

« eZ Publish » Fetch : Only 1 content class

```
{def $search_result = fetch( 'content', 'list', hash( 'parent_node_id', 2,
  'class_filter_type', 'include',
  'class_filter_array', array(24),
  'sort_by', array( array( 'attribute', false(), 'post/date' ) ),
  'limit', 10,
  'depth', 3
  )
}}
```

« eZ Find » Fetch : several content class, using the field homonymy

```
{def $search=fetch( ezfind, search,
  hash( query, '',
    'class_id', array('post', 'article'),
    'limit', 10,
    'sort_by', hash('attr_date_dt', 'desc')
  )
}}
```

Solr field name

Chapter 5 : Enhance eZ Find using the Solr syntax

► Complex search filters

Remember ! Solr is a Lucene service : http://lucene.apache.org/java/2_9_1/queryparsersyntax.html

```
'filter', array('NOT (attr_title_t:ez+find) OR attr_intro_t:ez+find) )')
```

Only returns results which contain the **'ez find'** or **'eZ Find'** expression in the **'title'** or **'Intro'** attributes.

Note the usage of the **'text'** (**_t**) of the **'title'** attribute, bringing **case-insensitivity**, unlike the **'string'** type

```
'filter', array('attr_title_s:[A TO G] AND ezf_df_text:google~0.7')
```

Only returns results of which the **'title'** starts by A,B,C,D, E or F (G excluded), and the content of which approximately contains the **'google'** expression (means it may also contain : Google, iGoogle, etc.).

Note : the **'ezf_df_text'** field is built dynamically, by copying the content of all of the document's **'string'**, **'text'** ou **'keyword'** fields. See the [schema.xml](#) file, and the definition of these ["copyField"](#) fields for more details.

eZ Conference 2010
June 24th



eZ Find 2.2 Customization & Advanced development

Summary :

Chapter 1 : Introduction about eZ Find

Chapter 2 : Speed-up eZ Find development tasks

Chapter 3 : Fields & Datatypes in Solr and eZ Find

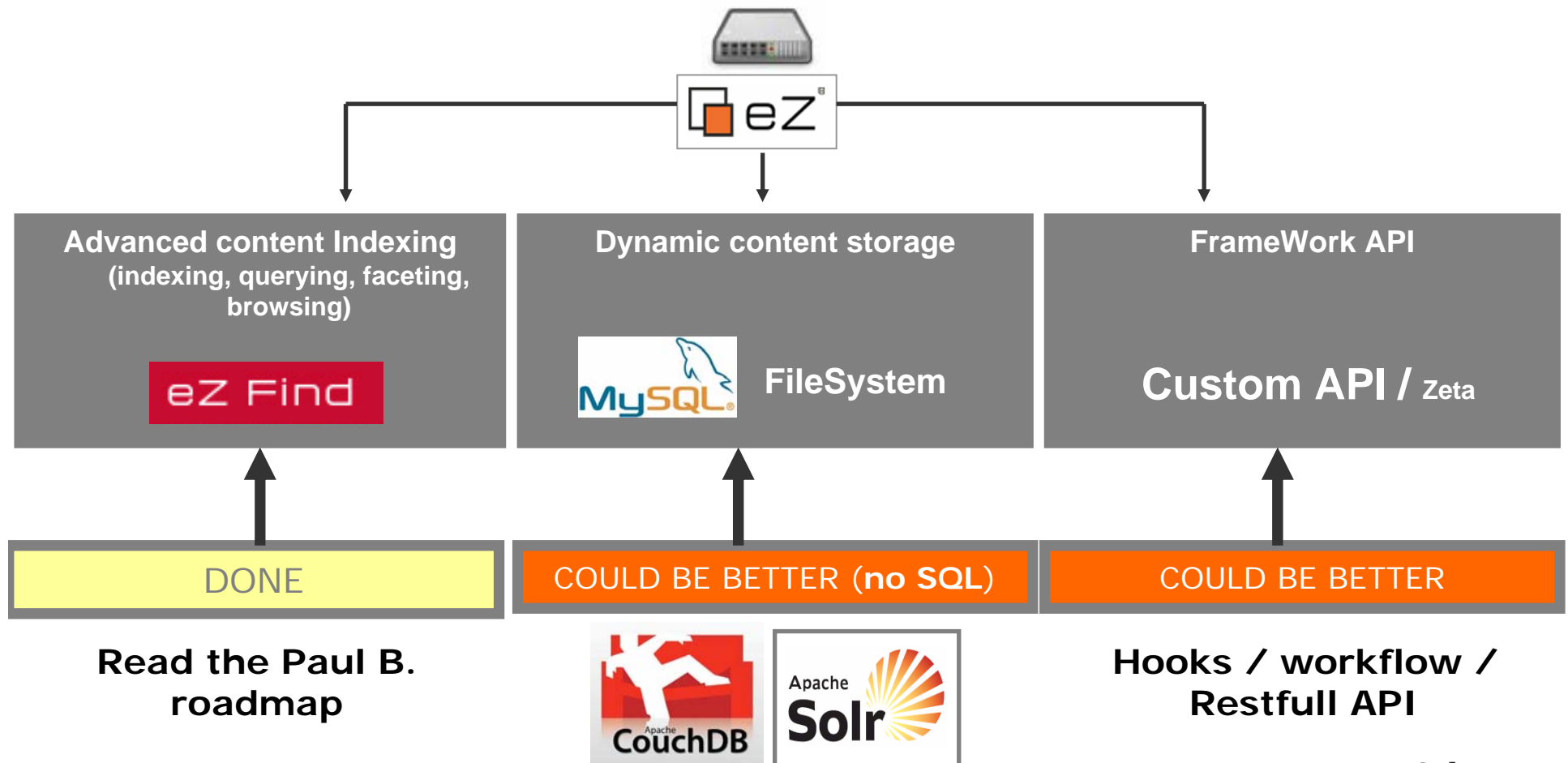
Chapter 4 : Indexing additional fields in Solr

Chapter 5 : Enhance eZ Find using the Solr syntax

Chapter 6 : **Conclusion**

Chapter 6 : Conclusion

► eZ Find : first step to the next generation CMS ?



eZ Conference 2010
June 24th



Follow me (and find eZ Find tutorials) :

FR : <http://www.gandbox.fr>

A screenshot of the Gandbox.fr website. The header includes a navigation menu with 'Blog', 'ma Cave', 'A propos', and 'Connexion'. The main content area features a large image of two glowing circular lights. Below the image, there is a section titled 'Développement avancé avec eZ Find (partie 3 : tirer profit de la syntaxe Solr)'. A sidebar on the right contains a 'WIDGET' section and a 'CALENDRIER' for June 2010.

EN : <http://share.ez.no>

A screenshot of the share.ez.no website. The header includes a navigation menu with 'Forums', 'Articles', 'Tutorials', 'Blogs', 'Projects', 'Directory', 'Community Program', and 'Team'. The main content area features a large article titled 'Advanced development with eZ Find - part 2 : Indexing additional fields in Solr'. A sidebar on the right contains a profile for 'gilles guirand' and a search bar.