



Ministerie van Binnenlandse Zaken en  
Koninkrijksrelaties

## DWR 1.2 TO Search Enricher

Versie 1.2

|        |                   |
|--------|-------------------|
| Datum  | 16 september 2010 |
| Status | Definitief        |

## Inhoud

|                                                                                  |           |
|----------------------------------------------------------------------------------|-----------|
| <b>Inhoud.....</b>                                                               | <b>2</b>  |
| <b>Inleiding.....</b>                                                            | <b>3</b>  |
| <b>1 Extraheren van content uit documenten .....</b>                             | <b>4</b>  |
| <b>2 Automatische detectie van karakter encoding .....</b>                       | <b>5</b>  |
| 2.1 <i>HTML-gebaseerde inhoud .....</i>                                          | 5         |
| 2.2 <i>Vernauwing encoding op basis van een taalhint .....</i>                   | 5         |
| 2.3 <i>Uitsluiting encoding op basis van een 'ondersteunde talen' lijst.....</i> | 5         |
| <b>3 Automatische detectie van inhoudstaal .....</b>                             | <b>6</b>  |
| 3.1 <i>Aanpassen inhoud- en titel-veld.....</i>                                  | 6         |
| 3.2 <i>Corpus lengte voor detectie .....</i>                                     | 6         |
| <b>4 Content filtering.....</b>                                                  | <b>7</b>  |
| 4.1 <i>Key Content Extraction (KCE) .....</i>                                    | 7         |
| 4.2 <i>Tag stripping.....</i>                                                    | 7         |
| 4.3 <i>Datum conversie .....</i>                                                 | 7         |
| <b>5 Ondersteuning voor het ACI protocol.....</b>                                | <b>8</b>  |
| 5.1 <i>ACI enveloppen .....</i>                                                  | 8         |
| 5.2 <i>ACI deletes .....</i>                                                     | 8         |
| 5.3 <i>ACI commits.....</i>                                                      | 8         |
| <b>6 Ondersteuning voor meerdere taken .....</b>                                 | <b>9</b>  |
| <b>7 Ondersteuning voor zoekmachines .....</b>                                   | <b>10</b> |
| 7.1 <i>Batch-verwerking.....</i>                                                 | 10        |
| 7.2 <i>Search engine ondersteuning.....</i>                                      | 10        |
| <b>8 Verwerken van grote documenten tot secties.....</b>                         | <b>11</b> |
| <b>9 Configuratie .....</b>                                                      | <b>12</b> |
| <b>10 Licensering .....</b>                                                      | <b>13</b> |

## Inleiding

Inhoud die naar de search engine gestuurd moet worden, bestaat doorgaans uit zowel (een referentie naar) document *inhoud* als wel document *metadata*. Bij het ophalen van gegevens uit content bronnen – zoals het Rijksportaal – worden deze twee gescheiden entiteiten doorgaans los van elkaar opgegeven.

De Search Enricher biedt de mogelijkheid om deze twee samen te voegen en als een geheel aan te bieden aan de search engine. Tevens biedt de Search Enricher ondersteuning voor een aantal 'verrijkende' bewerkingen op de inhoud. Deze verrijkende bewerkingen worden in dit document uiteengezet.

## 1 Extraheren van content uit documenten

Content uit documenten kan door de Search Enricher geëxtraheerd worden, ongeacht welke document type. Dit wordt bereikt door slim gebruik te maken van bestaande extractiebibliotheken zoals Apache Tika.

Apache Tika biedt ondersteuning voor alle doorgaans relevante documenttypen zoals Text, HTML, RDF, Microsoft Office, OpenOffice, PDF, etc.

Het documenttype (in de vorm van het MIME type) wordt aangeboden aan de search engine in de vorm van het format parameter veld, wanneer dit zo geconfigureerd is (doorgaans *dcterms\_format*). Hiermee kunnen zoek front-ends vervolgens specifiek zoeken naar bepaalde bestandstypen.

Wanneer er geen format parameter veld opgegeven wordt in de configuratie zal er geen format aangeboden worden aan de search engine. Bovendien kan ook een default format gespecificeerd worden. In dit geval zal het gedetecteerde type genegeerd worden en dit default type er voor in de plaats genomen worden.

## 2 Automatische detectie van karakter encoding

Tevens kan de Search Enricher door middel van een Character Encoding Detector op basis van de IBM ICU4J bibliotheek, een 'best-effort' poging doen om de encoding te detecteren van de opgegeven content. Uiteraard bestaat ook de mogelijkheid de encoding zelf te specificeren, namelijk via de HTTP interface. Tevens wordt getracht voor bepaalde situaties nog betere encodings-detectie toe te passen:

### 2.1 HTML-gebaseerde inhoud

In het geval van HTML-gebaseerde inhoud (text/html en application/xhtml+xml) wordt geen 'best-effort' poging gedaan om de encoding vast te stellen. In plaats daarvan wordt gebruik gemaakt van een volgens de RFC-3023 gespecificeerd proces om de encoding vast te stellen. Zo wordt er gekeken naar:

- *De opgegeven encoding.* Deze wordt gelijkgesteld aan het Content-Type HTTP header veld. (Bijv. 'Content-Type: text/html; charset=UTF-8')
- *De BOM.* Dit zijn de initiele bytes aan het begin van het document, welke eventueel endianness en dubbel of single-encoding vast kunnen stellen (Bijv. UTF-8, UTF-16BE, UTF-16LE)
- *De XML prolog.* Mits deze aan het begin van het document staat (Bijv. '<?xml version="1.0" encoding="UTF-8"?>')
- *Het Content-Type META-veld.* Mits dit in de eerste 4096 bytes van het document voorkomt. (Bijv. '<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />')

De META-detectie kijkt af van RFC 3023 en is toegevoegd na XML prolog detectie.

### 2.2 Vernauwing encoding op basis van een taalhint

Wanneer er een expliciete taal gespecificeerd is welke binnen ICU4J een toegesneden lijst met encodings heeft, zal alleen binnen deze encodings geselecteerd worden. In het geval van bijvoorbeeld "ja" zullen alleen UTF-8, UTF-16, Shift\_JIS, ISO-2022-JP en EUC\_JP overwogen worden.

### 2.3 Uitsluiting encoding op basis van een 'ondersteunde talen' lijst

Wanneer er een gelimiteerde lijst met mogelijke inhoudstalen gespecificeerd zijn, en er kan geen gebruik gemaakt worden van hints uit HTML meta-data, kan de zogenaamde 'plausibele encodingslijst' vernauwd worden tot enkel die encodings die binnen de gestelde talen vallen. Zo kan de lijst met mogelijke encodings voor de taal 'ja' (Japans), bestaande uit Shift\_JIS, ISO-2022-JP en EUC\_JP, uitgesloten worden wanneer de talen beperkt zijn tot bijvoorbeeld 'en, nl, ru'.

### 3 Automatische detectie van inhoudstaal

Bovendien kan de Search Enricher door middel van detectie op n-gram basis de taal van de inhoud bepalen. In de configuratie kunnen de volgende parameters hiervoor ingesteld worden:

- *Taal parameter.* Dit geeft aan wat de naam parameter om de taal in op te slaan moet zijn. Wanneer dit *niet* opgegeven is, wordt taal detectie uitgeschakeld.
- *Beschikbare talen.* Wanneer enkel uit een gelimiteerd aantal talen gekozen mag worden (omdat bijvoorbeeld de search engine zo is ingericht) kan er een lijst opgegeven worden met talen welke toegekend kunnen worden. Wanneer de gedetecteerde taal hier dan buiten valt, wordt de inhoud toegekend aan het standaardveld zonder nadere taal specificatie (bijvoorbeeld *content*), afhankelijk van de zoek machine implementatie
- *Standaard taal.* Wanneer er een standaard-taal ingesteld is, zal er geen poging gedaan worden om de taal te detecteren. Ook wanneer de taal reeds opgegeven is in de envelop zal de standaard taal genomen worden.

#### 3.1 Aanpassen inhoud- en titel-veld

Tot slot zullen de titel en inhoudsvelden aangepast worden zodat deze toegespitst kunnen worden op de gedetecteerde of opgegeven inhoudstaal. Afhankelijk van de zoekmachine implementatie wordt dit gedaan in de vorm van een veld met een taal-suffix (bijvoorbeeld 'content-nl', 'content-en', etc.) of het specificeren van de taal van de inhoud (bijvoorbeeld *DRELANGUAGE nl*).

#### 3.2 Corpus lengte voor detectie

De corpus lengte kan ingesteld worden met de parameter *enricher.language.analysis.consideration.threshold*. Deze staat standaard ingesteld op 8096 karakters, wat performance verruimt voor enige precisie. Wanneer deze op 0 wordt gelaten zal altijd alle tekst in overweging genomen worden, wat veel *n-gram counts* op kan leveren en derhalve de performance niet ten goede zal komen.

## 4 Content filtering

Met behulp van een filtering pipe-line kan met name HTML-gebaseerde content dynamisch aangepast worden. Zo is de kern functionaliteit van de Search Enricher het kunnen identificeren van zogenaamde 'key content'. Hierbij wordt met behulp van een heuristisch algoritme irrelevante content achterwege gelaten. Op deze manier kunnen bijvoorbeeld scripts, menu's en advertenties uit content gefilterd worden. Hierdoor blijft alleen de effectieve inhoud over.

Bovendien wordt er een gemakkelijk mechanisme geboden om de filters uit te bereiden, zodat ook additionele extractie mogelijkheden geboden kunnen worden. Zo kunnen gemakkelijke nieuwe bestandstypen en dergelijke aangesloten worden.

### 4.1 Key Content Extraction (KCE)

Wanneer KCE voor een bepaalde 'job' aangezet wordt, zal deze key content extraction toepassen op de daarvoor geschikte inhoud. Dit is doorgaans HTML-gebaseerde content.

Key content extraction werkt – naast een lijst met gedeeltelijke *class* of *id* namen – met een heuristische detectie welke kijkt naar het aantal woorden in de tekst. Zo kunnen vervolgens menu-items en dergelijke makkelijk uit de tekst gefilterd worden, wanneer deze maar uit een of enkele woorden bestaan (tot een bepaald maximum) terwijl de gehele tekst uit enkele honderden woorden bestaat.

### 4.2 Tag stripping

Naast KCE kan er ook gekozen worden voor een lichtere vorm van content extractie. Hierbij kan dan in een eerdere stap in het enrichment proces indicators toegevoegd worden welke content markeren als irrelevant. Er kan in de configuratie vervolgens een strip-tag opgegeven worden welke dan uit de HTML document-content gefilterd wordt. Dit is doorgaans 'noindex'.

### 4.3 Datum conversie

Omdat de inkomende datum niet altijd voldoet aan de gewenste datum, kan er gekozen worden om voor bepaalde metadatavelden de datum te converteren.

Hiervoor kunnen een aantal data formaten opgegeven worden, welke in de opgegeven volgorde toegepast worden op de inhoud van het veld totdat er een geldige conversie heeft plaatsgevonden. Bovendien kan de *locale* van de binnenkomende datums gespecificeerd worden (zodat maand- en dag-namen goed geconverteerd worden). Vervolgens wordt het formaat opgegeven waar de datum naartoe geconverteerd kan worden. Dit datum-formaat kan voldoen aan de Java datum standaard (zie de SimpleDateFormat documentatie) maar er kan ook 'EPOCH' opgegeven worden om de datum te representeren als het aantal seconden sinds 01-01-1970 00:00:00.

## 5 Ondersteuning voor het ACI protocol

ACI is een standaard om zoekmachine gerelateerde acties uit te voeren. Zo kan via het ACI protocol onder andere een gecombineerde set van zowel metadata als document inhoud aangeboden worden.

### 5.1 ACI enveloppen

De Search Enricher biedt ondersteuning voor de zogenaamde 'XML enveloppen' om gecombineerde metadata en document content aan te bieden. Hierbij kunnen referenties naar zowel lokale bestanden als intranet- of internet-urls opgegeven worden.

Een voorbeeld van een envelop is:

```
<autn:import xmlns:autn="http://schemas.autonomy.com/aci/">
  <autn:document>
    <autn:fetch url="/content/agd.html" deleteoriginal="true" />
  </autn:document>

  <autn:envelope>
    <autn:stubidx><![CDATA[
      #DREDBNAME ProfilerWebsites
      #DREREFERENCE http://www.minbzk.nl/foo
      #DREFIELD ..
    ]]></autn:stubidx>
  </autn:envelope>
</autn:envelope>
```

### 5.2 ACI deletes

De Search Enricher kan tevens gebruikt worden om documenten uit een zoekindex te verwijderen. Dit kan op zowel documentniveau als op veldniveau. Waarbij op documentniveau enkele records verwijderd kunnen worden op een unique-key basis. En op veldniveau alle documenten met een gegeven veldwaarde verwijderd kunnen worden.

### 5.3 ACI commits

De Search Enricher biedt tot slot de mogelijkheid om een commit-opdracht naar de zoek machine door te sturen. Hiermee wordt alle verkeer naar de zoekmachine toe geabstraheerd.

## 6 Ondersteuning voor meerdere taken

Door gebruik te maken van het *jobs* principe kunnen er meerdere taken aangemaakt worden binnen de configuratie van een enkele Search Enricher server instantie. Jobs kunnen in het configuratiebestand opgegeven worden, en configuratieopties zoals key content extraction, strip tags, datum extractie, de zoekmachine collectie en de zoekmachine kunnen per *job* opgegeven worden.

## 7 Ondersteuning voor zoekmachines

Door middel van de ACI abstractielaag kunnen meerdere zoekmachines aangesproken worden, eventueel zelfs door elkaar heen. Per *job* kan de zoek machine collectie opgegeven worden.

Bovendien kunnen zogenaamde 'batch opdrachten' snel verwerkt worden en vervolgens in bulk aangeboden worden aan de zoekmachine.

### 7.1 Batch-verwerking

Bij batch-verwerking worden meerdere XML enveloppen tegelijkertijd verwerkt en daarna ook in batch-vorm naar de zoek machine doorgestuurd. Hierbij worden grote batches opgeknipt tot een bepaalde (instelbare) hoeveelheid, om zo snelle verwerking te garanderen en bovendien de zoek machine niet te overbelasten. Verwerkte enveloppen worden dan per die hoeveelheid aangeboden aan de zoekmachine.

### 7.2 Search engine ondersteuning

Momenteel ondersteunt de Search Enricher de zoek machines Apache Solr en Autonomy IDOL.

## 8 Verwerken van grote documenten tot secties

Wanneer er veel grote documenten door de Search Enricher verwerkt moeten worden, kan er gekozen worden om "word breaking" toe te passen op de verwerking van het document. Hierbij wordt een document opgeknipt in zogenaamde *sections*. Er kan hiervoor een onder- en bovengrens opgegeven worden, waartussen het document afgebroken moet worden. Door de delta afdoende groot te houden zullen zinnen en paragrafen netjes afgesloten worden.

## 9 Configuratie

De Search Enricher maakt gebruik van een enkel bestand waarin zowel de globale als de per-*job* configuratie in staat. De locatie van dit bestand wordt uit de Web applicatie context variabele '*enricher.settings.file*' gehaald. Een voorbeeld configuratie bestand is:

```
enricher.generator.aci.add.uri = http://127.0.0.1:9001/DREADD
enricher.generator.aci.commit.uri = http://127.0.0.1:9001/DRESYNC
enricher.generator.aci.delete.reference.uri =
http://127.0.0.1:9001/DREDELETEREF
enricher.generator.aci.user.agent = MinBZK Zoek-en-Vind Content
Enricher 1.0

enricher.generator.solr.uri = http://localhost:8983/solr/

enricher.generator.documents.per.upload = 5000

enricher.envelope.word.break.min = 300
enricher.envelope.word.break.max = 500

# ImportEnvelope

enricher.envelope.job.ImportEnvelope.generator.type = ACI
enricher.envelope.job.ImportEnvelope.db.name = ProfilerWebsites
enricher.envelope.job.ImportEnvelope.result = /usr/local/search-
enricher/processed/%y/%m/%d
enricher.envelope.job.ImportEnvelope.strip.tags = noindex
enricher.envelope.job.ImportEnvelope.kce.enabled = true

enricher.envelope.job.ImportEnvelope.format.detection.parameter =
dcterm_format
# enricher.envelope.job.ImportEnvelope.format.detection.default =
text/html

enricher.envelope.job.ImportEnvelope.language.detection.parameter =
dcterm_language
enricher.envelope.job.ImportEnvelope.language.detection.supported =
nl, en
# enricher.envelope.job.ImportEnvelope.language.detection = nl

# ImportEnvelopeKluwer

enricher.envelope.job.ImportEnvelopeKluwer.generator.type = ACI
enricher.envelope.job.ImportEnvelopeKluwer.db.name = ProfilerKluwer
enricher.envelope.job.ImportEnvelopeKluwer.result=/usr/local/search-
enricher/processed/%y/%m/%d
enricher.envelope.job.ImportEnvelopeKluwer.strip.tags = noindex
enricher.envelope.job.ImportEnvelopeKluwer.date.convert.PUBDATE.from =
"dd MMMM yyyy", "dd-MM-yyyy", "yyyy-MM-dd HH:mm"
enricher.envelope.job.ImportEnvelopeKluwer.date.convert.PUBDATE.to =
EPOCH
enricher.envelope.job.ImportEnvelopeKluwer.date.convert.PUBDATE.locale
= NL enricher.envelope.job.ImportEnvelopeKluwer.kce.enabled = true

# Generic NULL-job, where the actual database is taken from the IDX
file

enricher.envelope.job.null.generator.type = ACI
enricher.envelope.job.null.result=/usr/local/search-
enricher/processed/%y/%m/%d
enricher.envelope.job.null.strip.tags = noindex
enricher.envelope.job.null.kce.enabled = true
```

## 10      Licensering

De Search Enricher is gebaseerd op het Key Content Extraction algoritme van het WebNews Crawler project. Dit project valt onder de GPL licentie, versie 2 (of hoger), en zodoende valt de Search Enricher hier ook onder. Er is gekozen voor een GPL licentie v3 voor de Search Enricher.

Omdat de Search Enricher als een *stand-alone* service opereert, en alleen communiceert over een netwerkverbinding, heeft de licentie geen consequenties voor overige custom software binnen het landschap.