



Web Publisher
HTML Interface
Technische Handleiding

5 WEB PUBLISHER.....	1
1 INTRODUCTIE.....	4
2.1 HOE WERKT HET	4
<i>Template File</i>	4
2.2 CONFIGURATIE.....	4
3 HTML INTERFACE	5
3.1 WEERGAVE HTML ZOEKSCHERM	5
3.1.1 DE ZOEKOPDRACHT STUREN	6
<i>Catalogusnaam</i>	6
<i>Zoekreeks</i>	6
3.1.2 HET FORMAAT VAN HET RESULTAAT	7
<i>Zoekresultaat weergave Selectie</i>	7
<i>Zoekformulier selectie</i>	8
SELEKTIE VAN DE ZOEKRESULTAAT WEERGAVE	8
<i>Document Encoding Conversion</i>	9
<i>Thumbnail Size Limitation</i>	10
3.1.3 ZOEK OPDRACHT.....	10
<i>Junctors</i>	11
<i>Query Values</i>	12
<i>Properties</i>	12
<i>Properties Uitbreiden</i>	<i>Error! Bookmark not defined.</i>
<i>Meerdere properties</i>	13
3.1.4 RESULTATEN SORTEREN.....	14
3.2 DE RESULTAATPAGINA CREËREN.....	14
3.2.1 OVERVIEW.....	14
3.2.2 DE “TEMPLATE BESCHRIJVINGS TAAL”	15
<i>Comments</i>	16
<i>Includes</i>	16
3.2.3 DE REPLY PAGE LABEL	17
3.2.4 ZOEKFORMULIER.....	18
3.2.5 KNOPPEN DIE OP EEN RESULTAATPAGINA WORDEN GETOOND.	19
<i>Nieuwe Zoekopdracht</i>	19
<i>Verstuur een Zoekopdracht</i>	19
<i>Zoek in de huidige gevonden reeks</i>	20
<i>Zoek in de opgegeven reeks</i>	20
<i>Navigatie door de Resultaatpagina</i>	20
<i>“Collection Basket” Knop</i>	20
3.2.6 RESULT HEADING (AANTAL GEVONDEN RECORDS EN HOEVEEL ER GETOOND WORDEN.).....	21
3.2.7 RESULTAAT LIJST	21
3.2.8 DE INHOUD VAN DE RESULTAATWEERGAVE	22

<i>Fields</i>	22
Voor “string” velden.....	22
“encoding”.....	22
Opties voor velden van het type “Real”:	23
Opties voor velden van het type “Picture”.....	24
Opties voor velden van het type “Audio”.....	24
Opties voor velden van het type “Categories”.....	24
Asset Access Labels.....	24
3.2.9 RESULT FORMAAT OPTIES	28
3.3 ANDERE RESULTAAT PAGINA’S	28
3.3.1 ZOEKPAGINA	28
3.3.2 INFO PAGE	29
3.3.3 COLLECTION BASKET PAGINA DE CHECKOUT PAGINA	29
3.3.3A KNOPPEN WELKE TOEGANKELIJK ZIJN VANAF DE COLLECTION BASKET PAGINA	31
3.3.4 ANDERE LABELS	31
3.3.5 CATEGORIEN PAGINA (ZIE OOK SECTIE 4)	32
3.3.6 FOUTMELDINGEN	33
3.4 DATA DIE ZICH NIET IN CUMULUS BEVIND VIA WEBPUBLISHER BENADEREN	35
3.4.1 ZELF EEN EIGEN VARIABELE DEFINIEREN	35
3.4.2 WAARDES INSTELLEN EN OPVRAGEN	36
4 CATEGORIEN TABEL WEERGAVE	37
5 ASSET DOWNLOADEN	37
5.1 LINK CONSTRUCTIE	38
5.2 ASSET CONVERSIE	38
5.2.1 CONVERTERS EN OPTIES	38
5.2.2 VOORBEELDEN	39
6 COLLECTION BASKET	39
7 DE TEMPLATES VAN CUMULUS WEB PUBLISHER 5.0	40
7.1 THE STANDAARD TEMPLATES	40
7.1.1 STANDAARD RESULTAAT PAGINA	40
7.1.2 ANDERE PAGINA’S	41
7.1.3 CROSS PLATFORM INSTALLATIE	41
7.2 TEMPLATES CREËREN VAN DE HUIDIGE WEERGAVE BINNEN CUMULUS	41
8 WEB PUBLISHER VERSUS INTERNET MEDIA SERVER 4.0	42
8.1 NIEUWE FEATURES	42
8.1.1 HTML TEMPLATES OPNIEUW LADEN	42
8.1.2 RESULTAAT SORTEREN	42
8.1.3 ON-THE-FLY ASSET CONVERSIE	42
8.2 BACKWARD COMPATIBILITEIT	43

1 Introductie

The Cumulus Web Publisher is een Web interface voor Cumulus catalogi. Hij ontvangt zoekopdrachten via een CGI script welke wordt aangeroepen vanuit HTML pagina's en laat het resultaat zien als een lijst van gevonden voorwerpen. Vanuit deze weergave, zijn gebruikers in staat om het origineel te downloaden/previewen, of ze in de collectionbasket plaatsen, om deze later te downloaden, en om additionele informatie over de Asset op te vragen. Ook kan de categorieën tabel van de catalogus worden opgevraagd.

De Web Publisher ondersteund meerdere simultane gebruikers. De lay-out en inhoud van de HTML pagina's die door Canto worden gegenereerd zijn vrij in te stellen. Hoe zoekopdrachten worden ontvangen en afgehandeld kan voor een groot gedeelte worden afgehandeld door de HTML interface aan te passen. Dit document is bedoeld om meer inzicht te geven in hoe dit werkt.

2.1 Hoe Werkt het

Een gebruiker maakt een connectie met de webserver waar de Cumulus Webpublisher CGI op geïnstalleerd is. Hier staan een aantal statische html pagina's waarin de zoekopdracht kan worden ingevoerd, als er op submit gedrukt wordt zal er contact gezocht worden met de Cumulus CGI. Deze geeft een antwoord terug welke qua look and feel gebaseerd word op de cumulus template files.

Template File

Deze template files worden tijdens het opstarten van de CGI ingelezen. Hiermee kan per klant een verschillende look worden gegenereerd. De folder Templates bevind zich op een Macintosh in de Cumulus Server/Webpublisher for Single user folder. Onder NT bevindt deze zich in de map WINNT\system32\canto\components5 map. De applicatie leest de main.html uit de map templates. Alle files die via een Include statement geladen worden, worden geladen. Veranderingen in de templates worden direct doorgevoerd.

2.2 Configuratie

de HTML interface is geconfigureerd door de juiste bestanden te wijzigen. Dit zijn de zoek pagina's en de templates. De lay-out en de manier waarop tekst word weergegeven kan door middel van standaard HTML tags, de functionaliteit is afhankelijk van de uitbreidingen die Canto op de HTML taal heeft gemaakt zoals beschreven bij punt 3.

3 HTML Interface

De Web Publisher heeft een standaard direct te gebruiken zoek pagina.

Dit hoofdstuk gaat in op de details betreffende het aanpassen van deze pagina en hoe de Web Publisher deze genereert.

3.1 Weergave HTML Zoekscherm

Een zoekopdracht die verstuurd word, gaat als CGI call vanuit de Webpagina naar de Webpublisher toe.

Dit bestaat uit een HTML form (een <FORM> ... </FORM> constructie).

In het bijzonder staat in deze form de URL van de CGI applicatie.

Het argument word aan de CGI doorgegeven als een search of post string.

De zoekopdracht splits zich in de database die doorzocht moet worden, een selectie die het reply formaat bepaald, en de eigenlijke zoek velden.

De Web Publisher verwacht de data in HTML tag paren typisch voor HTML pagina's.

In de volgende hoofdstukken beschrijven we de Cumulus HTML tags.

Ook staan hier voorbeelden voor het gebruik van de Cumulus tags in een HTML form.

3.1.1 De zoekopdracht sturen

Catalogusnaam

naam: "catalogName"

waarde: de naam van een voor internet gedeelde Cumulus catalogus (case sensitive).

Een of meer catalogi waaraan de zoekopdrachten worden doorgegeven.

Ook alle volgende zoekopdrachten worden naar deze catalogi gestuurd.

Meerdere catalogi tegelijk doorzoeken is mogelijk, de resultaten worden in een gezamenlijke lijst getoond.

Om in meerdere catalogi tegelijk te zoeken moet de catalogName regel nogmaals in de pagina worden opgenomen met in de eerste regel catalogus 1 en op de tweede regel catalogus 2 of de catalogi moeten als comma gesepareerde lijst worden opgegeven.

Bijvoorbeeld catalogName = "catalog1",catalog2

Als er geen catalogus met de opgegeven naam gedeeld is voor webtoegang, word er een foutmelding getoond.

Deze functionaliteit word alleen vanuit de eerste zoekpagina ondersteund.

Als er geen catalogusnaam word opgegeven word automatisch in de eerste voor webtoegang gedeelde catalogus gezocht.

Als er helemaal geen catalogus gedeeld is word er een foutmelding getoond.

Voorbeeld van een catalogus kiezen via een popup:

Choose a catalog:

```
<SELECT NAME="catalogName">
```

```
<OPTION SELECTED value="Flora"> Flowering plants
```

```
<OPTION value="Insects"> Insects
```

```
</SELECT>
```

Zoekreeks

naam: "searchRange"

waarde: "all", "found"

searchRange geeft aan of een zoekopdracht voor de gehele database geldt of dat er binnen de huidige selectie moet worden gezocht. Deze optie is niet beschikbaar in start zoek pagina's

Deze functionaliteit zit standaard in de "Search_all_button" en de "Search_found_button". De standaard is "All"

Voorbeeld:

Zoek in:

```
<INPUT TYPE="RADIO" NAME="searchRange" VALUE="all">
```

de hele database, of

```
<INPUT TYPE="RADIO" NAME="searchRange" VALUE="found">
```

de huidige selectie.

Dit toont twee radio buttons waarmee de zoekreeks kan worden gekozen.

3.1.2 Het formaat van het resultaat

In de start zoekpagina is het mogelijk om het resultaat format op verschillende manieren op te geven.

Allereerst kan de standard lay-out worden ingesteld. Dit bepaald globaal waar het resultaat getoond word, waar het formulier om de volgende zoekopdracht in te typen komt en of zo'n formulier direct op de zoekpagina of op een aparte pagina moet worden getoond.

De template voor dit zoekformulier kan worden gekozen en bepaald de weergave en functionaliteit van het zoekformulier.

Het resultaat formaat kan worden geselecteerd, dit bepaald de plaatsing in de resultaat lijst en de layout en inhoud van een gevonden item.

Alle opties zijn in het zoekformulier op te nemen zodat de gebruiker kan bepalen hoe zijn zoekresultaten getoond worden.

Zoekresultaat weergave Selectie

naam: "replyPage"

waarde: een van de template namen (zie sectie 3.2.3).

Kies het formaat van de resultaat pagina. Als de template met de opgegeven naam niet bestaat zullen de ingebouwde (saaie) standaard pagina's gebruikt worden.

Deze optie word normaal gesproken hard in de start zoekpagina gezet.

Als er niets opgegeven word, worden de default pagina's gebruikt.

Zoekformulier selectie

naam: "queryForm"

waarde: een query form template (see sectie 3.2.4).

Kies de zoekformulier template die gebruikt moet worden.

Als de template niet bestaat word default template gebruikt.

Deze optie wordt normaal gesproken in de start zoekenpagina hard opgegeven.

Alle volgende pagina's worden dan op deze pagina's gebaseerd.

Selectie van de zoekresultaat weergave

naam: "resultFormat"

waarde: een zoek resultaat template naam

hiermee wordt bepaald welke weergave word gebruik voor het resultaat van een zoekopdracht. Als de resultFormat niet word opgegeven word standaard een thumbnail, de bestandsnaam en de notes getoond.

voorbeeld:

Selecteer de resultaat weergave:

```
<INPUT TYPE="RADIO" NAME="resultFormat" VALUE="textAndThumbnail">
```

Thumbnail en Omschrijving

```
<INPUT TYPE="RADIO" NAME="resultFormat" VALUE="thumbnail"> Thumbnail
```

```
<INPUT TYPE="RADIO" NAME="resultFormat" VALUE="text"> Tekst
```

Hiermee kan worden gekozen hoe het zoekresultaat word getoond.

naam: "replyLength".

waarde: een nummer.

Het aantal items dat weergegeven word per resultaat pagina.

Het kan soms gebeuren dat het aantal met één wordt verhoogd of verlaagd om de beelden beter over de resultaat pagina's te verdelen.

Als er niets gedefinieerd is worden er automatisch 6 (of 5 dan wel 7) getoond.

Document Encoding Conversion

Web browser kunnen problemen hebben met 8-bit Ascii karakters op verschillen platformen, Op basis van de document encoding conventie die de gebruiker heeft geselecteerd,

Webpublisher heeft daarom de mogelijkheid om verstuurde tekens met een hoger hexadecimale waarde dan 0x7F waar een HTML alternatief voor is hiernaar te converteren.

(Dit ziet er dan bijvoorbeeld zo uit “Ä” dit is de hoofdletter A umlaut).

Deze alternatieven zijn niet afhankelijk van het platform waarop de server en/of de browser draaien.

Deze optie word middel de hierop volgende syntax geactiveerd

naam: “DocEncoding”

waarde: een van:

“IsoLatin”

“MacRoman”

“Plain”

“Disable”

“IsoLatin” en “MacRoman” gebruik de Iso Latin of Macintosh Roman character encoding om HTML alternatieven te genereren.

“Plain” Converteer niets maar zorgt er wel voor dat Speciale HTML karakters zoals <> geconverteerd worden alsmede dat return worden vervangen door de
 tag.

Om zelfs dit uit te schakelen kan er worden gekozen voor de optie disable.

Deze instellingen zijn vergelijkbaar met de optie “Encoding” die in 2.3.8 wordt omschreven. De functionaliteit van disable zit bij Encoding in de Pre optie.

Als er een waarde word opgegeven voor Encoding word deze gebruikt in plaats van DocEncoding.

De default is “IsoLatin”.

Deze optie wordt alleen op start zoekpagina's ondersteund.

Thumbnail Size Limitation

De grote van een thumbnail (minipict) kan gelimiteerd worden door de “thumbnailConstraint” te gebruiken.
waarde: een positief nummer.

De maximale thumbnail hoogte en breedte kan worden opgegeven in Pixels.
Thumbnail die groter zijn worden gescaled naar deze waarden.
Een waarde van 0 betekend dat er geen limit is.
Deze optie wordt alleen op start zoekpagina's ondersteund.

De Thumbnail size kan ook in de pagina waarin de thumbnail zich bevindt worden gelimiteerd dit word in sectie 3.2.8 omschreven.

3.1.3 Zoek opdracht

Net zoals de zoekopdracht in Cumulus Single User/Client, word ook een Webpublisher zoekopdracht gemaakt door een of meer criteria te combineren via en/of.

In cumulus worden dit Junctors genoemd.

De criteria worden opgebouwd uit de eigenschappen van een Asset zoals record name en een test zoals bijvoorbeeld is, is niet, bevat, etc. en een waarde waarop gezocht moet worden.

Alle eigenschappen waarop binnen Single User/Client kan worden gezocht kan ook binnen Webpublisher op gezocht worden.

Het word in de standaard interface echter als vaste paren aangeboden, een keuze is dan bijvoorbeeld Recordname Contains of Recordname is.

In SU/Client zijn dit twee losse keuzes.

Dit word dan ook niet als losse componenten gezien.

Een uitgebreide zoekopdracht ziet er dan als volgt uit.

```
<property> <value>  
<junctor> <property> <value>  
<junctor> <property> <value> ...
```

Hieronder is een voorbeeld van hoe deze zoekopdracht eruit kan zien.

```
name="prop" value="field[Categories] op[is]"  
name="qVal" value="South Africa"  
name="junc" value="and"  
name="prop" value="field[Notes] op[contains]"  
name="qVal" value="medical"  
(HTML tags zoals <INPUT> zijn niet vermeld)
```

Dit vertaald zich in:

“category is ‘South Africa’ en notes bevat ‘medical’.”

Net zoals in SU/Client worden lege velden genegeerd evenals de en/of die een zoekopdracht kan verbinden,

De paren worden gevormd door veld welke via de veldnaam word gedefinieerd en de test die uitgevoerd moet worden zoals “LessThan” of “Contains”.

Met de HTML interface is meer mogelijk dan met SU/Client, omdat het mogelijk is om de zoekopdracht naar meerdere databases tegelijk te sturen en toch het zoekresultaat in één pagina weer te geven.

Er kan via de “mulConts” optie op een aantal woorden die in een willekeurige volgorde voorkomen worden gezocht. Ze moeten dan wel allemaal voorkomen.

Het is ook mogelijk om met de informatie die in een zoekopdrachtveld ingevoerd word in meerdere Cumulusvelden te zoeken.

Webpublisher kijkt niet of de syntax van een zoekopdrachten correct is of dat de zoekopdracht zinvol is. Een test die op een niet bestaand veld wordt uitgevoerd word genegeerd. Het voordeel hiervan is dat er meerdere catalogi kunnen worden doorzocht die niet allemaal dezelfde velden bevatten.

Junctors

naam: “junc”

waarde: “and”, “or”, “butNot”, “outerAnd”, “outerOr”

De junctors verbinden zoekopdrachten.

Met de opdracht butNot kan een aantal record worden weggelaten uit een zoekopdracht.

OuterAnd en outerOr kunnen gebruik worden om een zoekopdracht te verkleinen.

Standaard word er van and gebruik van gemaakt.

Query Values

naam: "qVal"
waarde: Text

Dit is de waarde die word gebruikt voor een zoek opdracht. Standaard is er geen waarde toegekend. In het geval dat deze waarde leeg blijft word ze genegeerd, als er van outerAnd en outerOr gebruik word gemaakt worden die wel in stand gehouden om dat deze zich op een lager niveau van de zoekopdracht bevind, en de zoekopdracht zo één geheel blijft.

Properties

naam: "prop"
waarde: Een veldnaam en een test parameter zoals:
"field[<some name>] op[<some operation>]" of
"field[<some name>] operation[<some operation>]"

Het veld word opgegeven zoals het in Catalog Properties van de Cumulus Applicatie word weergegeven. De namen zijn Case sensitive. Als er in de naam [of] gebruikt worden moeten deze via het escape character (de \ word gebruikt om aan te geven dat het volgende teken geen speciale betekenis heeft de \ moet zelf ook op die manier gebruikt worden. Dit word de escape character genoemd.) uitgesloten worden.

Als alternatief kan ook een interne identificatie dit is bijvoorbeeld een Windows GUID zoals "{F054A8B3-B6DC-11d3-9AC4-0080AD783086}".

De ID van een veld kan word bepaald door middel van een Cumulus record export file. Dit is een tekst bestand en bevat veldnamen en ID's in zijn Header. Veld ID's kunnen ook via de API worden uitgevonden.

Hieronder een voorbeeld.

(In Visual Basic)

"Color Mode," is MsgBox

CumulusApp.FrontCollection.Catalog.RecordLayout.FieldDefinitions.Item ("Color Mode").FieldUID

(in Appelscript)

get field uid of field definition "Color Mode" of record layout of catalog of front collection

Het word aangeraden om interne ID's te gebruiken als de naam van het veld tekens bevat die geen onderdeel uitmaken van de 7-bit ASCII tabel.

Dit zijn bijvoorbeeld umlaut en tekens met accenten erop.

Dit voorkomt problemen als er vanaf verschillende OS platformen word gezocht.

Een operation kan bestaan uit "startsWith," "contains," "containsNot," "is,"

"isNot," "lessThan," "lessEqual," "greaterThan," "greaterEqual,"

"hasValue," "hasNoValue" and "mulConts."

Let erop dat een veld op de juiste manier geïndexeerd moet zijn om erin te kunnen

zoeken. Dat betekent dat als op de inhoud van een veld met de optie **is** gezocht wordt dit veld voor sorteren geïndexeerd moet zijn.

Om **contains** te gebruiken moet indexed for containment search aanstaan.

Voor "**hasValue**," "**hasNoValue**" is dit niet nodig omdat er alleen wordt gekeken of het veld een waarde heeft.

waarde: "thumbnailSimilarTo"

Bij deze optie wordt gezocht naar een thumbnail die op de thumbnail van het opgegeven record lijkt.

waarde: "ctgBestMatch"

Gebruik de categorie die het meest lijkt op de waarde die ingevoerd is bij **category is**.

Deze optie wordt gebruikt zodat niet de exacte naam van een categorie gebruikt hoeft te worden.

Meerdere properties

Meerdere properties kunnen doorzocht worden op één opgegeven waarde.

Dit wordt bereikt door de verbindingen tussen de waarden weg te halen inclusief de "qVal" name tags. De waarde die zich aan het einde bevindt zal worden gebruikt.

Voorbeeld (zonder HTML tags):

name="prop" value="field[Notes] op[mulConts]"

name="junc" value="or"

name="prop" value="field[Categories] op[contains]"

name="qVal" (waarde die uit een invoer veld komt)

Deze zoekopdracht doorzoekt notes en categorieën op de waarde die is opgegeven.

3.1.4 Resultaten Sorteren

De resultaten van een zoekopdracht kunnen net als in SU/Client collections gesorteerd worden

Dit sorteren kan op elk veld welke is geïndexeerd voor sorteren De sortering kan oplopend of aflopend zijn.

naam: “sorting”

waarde: Een veld naam en een sorteer zoals in:

“field[<some name>] direction[Ascending]” of “...direction[Descending]”

Het resultaat van een zoekopdracht word gesorteerd op het opgegeven veld.

Het veld wordt op dezelfde manier opgegeven als een zoekopdracht.

Als er geen sorteer richting word opgegeven word het resultaat oplopend gesorteerd.

Er word een foutmelding gegenereerd als een veld opgegeven word welke niet is geïndexeerd voor sorteren.

3.2 De resultaatpagina creëren

3.2.1 Overview

De resultaatpagina;s worden gegenereerd vanuit template bestanden.

Deze worden door Web Publisher ingelezen tijdens het opstarten.

De bestanden bevinden zich in de templates map. De templates map bevindt zich in de components5 map. De map components5 bevindt zich onder windows NT/2000 in de map winnt\system32\canto\. Onder Unix en MacOS bevindt de components map zich in de map waarin ook de Cumulus applicatie staat.

De template folder moet het file main.html bevatten, dit bestand word als eerste door de applicatie gelezen. In dit bestand kunnen alle waarden staan. Het is ook mogelijk om via een zogenaamde include een ander bestand te laden.

Dit is de manier waarop Canto standaard geïnstalleerd word.

Dit zijn de waarden die in de templates worden gedefinieerd.

- Reply Page
- Query form
- Result heading and buttons
- Result items
- Buttons leading to other pages.

De reply page die het standaard zoekresultaat bevat is gegenereerd van de template “REPLY_PAGE”. Dit is een van de template beginpunten. Zo zijn deze er ook voor categorie pagina's, de collection basket etc.

Hier staat HTML code in voor wat betreft headers statische tekst, tags voor opmaak, achtergrondkleur en achtergrondbeelden etc.

Er kunnen ook referenties in staan naar andere template die één van de volgende opties beschrijven.

- een formulier waarin vervolg zoekopdrachten kunnen worden ingegeven.
- een bovenschrift bij de resultaatpagina's
- een resultaat pagina
- navigatie knoppen om naar de volgende of vorige resultaatpagina te gaan.
- knoppen om naar een zoekpagina buttons verwijzen, naar een collection basket of een lijst met categorie

Elk item wat getoond word op de resultaatpagina wordt gegenereerd op basis van de "RESULT_ITEM" template.

Deze beschrijft hoe de pagina getoond moet worden met betrekking to de thumbnail, record name, omschrijving, bestandsgrote etc., er kan ook een link naar de info pagina op staan, een link naar het origineel of een ander beeld zoals bijvoorbeeld een lage resolutie beeld.

Het is mogelijk om een nieuwe zoekpagina weer te geven op dezelfde pagina als het resultaat als om een link op te nemen welke verwijst naar de zoekpagina.

Die pagina wordt dan vanuit een andere template opgebouwd.

3.2.2 De "Template beschrijvings taal"

De template bestanden bevatten beschrijvingen voor de verschillende onderdelen van een resultaat pagina. Dit zijn onder andere dingen als hoeveel items worden tegelijk weergegeven op één resultaat pagina.

Een template is als volgt opgebouwd.

label = { value }

or

label (argument) = { value }

de label bevat de template naam of de naam van de instelling die gewijzigd word.

Deze labels worden in dit document omschreven.

De label moet als eerste op de regel staan gevolgd door een = op dezelfde regel.

Als de waarde die aan de label word toegewezen zich op dezelfde regel bevind hoeven de { } er niet omheen.

De argument optie kan worden gebruikt om een naam aan de template toe te wijzen.

Dit zorgt ervoor dat de template via die naam aan te spreken is.

Dit zijn de queryform template, de reply page template, en alle templates die met resultaat weergave te maken hebben.

Deze templates worden vanuit een start zoekpagina aangeroepen middels de "queryForm" call, de "replyPage" call en "resultFormat" call, (zie ook 3.1.2 en 3.1.3).

Als een label geen argument (naam) meekrijgt word deze gebruikt als de standaard weergave. Dit betekend dat als er met een argument naar een label wordt verwezen en deze bestaat niet of er is geen label opgegeven de resultaatpagina op basis van deze standaard definities word weergegeven.

Er zijn ook labels welke nog extra waardes kunnen hebben.
Dit ziet er dan als volgt uit.

```
<% label argument-tag=argument ... %>•
```

Alle optie die meegegeven worden moeten tussen <% en %> staan.

De opties bestaan uit de optienaam = “optiewaarde” de optiewaarde moet tussen “” op het moment dat deze uit meerdere woorden bestaat.

Niet alle labels hebben dit soort opties.

Als er wel opties zijn , hoeven deze niet vermeld te worden, als ze niet vermeld worden wordt de standaard waarde gebruikt.

Webpublisher heeft standaard templates voor elke mogelijke zoek en reply pages.

Deze waarden worden gebruikt als er geen templates zijn of de gedefinieerde templates niet bestaan.

Comments

Een template mag comments bevatten deze moeten worden voorafgegaan door “;”.

Comments kunnen niet middenin een regel staan.

Lege regels worden genegeerd.

Includes

Er kan gebruik worden gemaakt van zogenaamde includes.

```
#include “BasketPage.html”
```

Hiermee word de inhoud van de file BasketPage.html op dat punt ingevoegd.

Het voordeel is dat een eigen template definitie niet elke keer opnieuw ingetyped hoeft te worden.

Uiteraard kan er ook via een pad verwijzing geinclude worden.

```
#include “Pages/BasketPage.html”
```

Alle labels namen, opties etc. zijn niet case-sensitive.

3.2.3 De Reply Page Label

label: "REPLY_PAGE"

Als optie kan er een naam aan de template meegegeven worden.

Bijvoorbeeld REPLY_PAGE (topitreply)

Als er geen naam word opgegeven word de REPLY_PAGE als standaard gebruikt.

Uiteraard word de eerste definitie die gevonden word gebruikt.

Deze definitie worden in een start zoekenpagina opgegeven.

Voorbeeld/Default:

```
REPLY_PAGE = {
<HTML> <HEAD>
<TITLE> Query Result </TITLE>
<BODY>
<% QUERY_FORM %>
<hr>
<CENTER><H2> <% RESULT_HEADING %> </H2></CENTER>
<CENTER> <% NEXT_BUTTON text="Next Page" %>
<% PREVIOUS_BUTTON text="Previous Page" %>
</CENTER>
<% RESULT_LIST %>
<hr>
</BODY></HTML>
}
```

In dit voorbeeld word een zoek scherm weergegeven daarna een resultaatweergave (aantal gevonden records en van hoeveel gevonden records.) Dan twee knoppen om naar de volgende dan wel de vorige pagina te gaan met daarna de resultaatlijst.

De knoppen hebben hun naam als optie.

Het resultaatformaat alsmede het zoekformulier worden bepaald door de opties die door de start zoekenpagina zijn gedefinieerd welke ons naar deze resultaatpagina gebracht heeft.

Voorbeeld/TopIT:

```
REPLY_PAGE (topitreply) = {
<HTML> <HEAD>
<TITLE> Zoekresultaat</TITLE>
<BODY>
<% QUERY_FORM %>
<hr>
<CENTER><H2> <% RESULT_HEADING %> </H2></CENTER>
<CENTER> <% NEXT_BUTTON text="Volgende Pagina" %>
<% PREVIOUS_BUTTON text="Vorige Pagina" %>
```

```
</CENTER>
<% RESULT_LIST %>
<hr>
</BODY></HTML>
}

<input type="hidden" name="replyPage" value="topitreply">
```

3.2.4 Zoekformulier

Een zoekformulier (query form) definiëren is vergelijkbaar met een zoekpagina definiëren.

Het verschil is dat de <FORM> en </FORM> tags door de applicatie zelf worden gegenereerd en de Submit Button middels een label wordt gegenereerd.

label: "QUERY_FORM"

Er moeten een naam achter de query_form tag opgegeven worden zodat de form vanuit een start zoekpagina kan worden aangeroepen.

Als er geen naam wordt opgegeven wordt die pagina als de standaard gebruikt wanneer er een <% QUERY_FORM%> in een replypage wordt opgegeven.

Voorbeeld:

```
QUERY_FORM = {
<INPUT TYPE="hidden" NAME="junc" VALUE="outerAnd">
<INPUT TYPE="hidden" NAME="prop" VALUE="field[Notes] op[mulConts]">
<INPUT TYPE="hidden" NAME="junc" VALUE="or">
<INPUT TYPE="hidden" NAME="prop" VALUE="field[Categories] op[contains]">
<INPUT TYPE="TEXT" NAME="qVal" SIZE=35 MAXSIZE=70>
<% SEARCH_BUTTON text="Search" %> }
```

3.2.5 Knoppen die op een resultaatpagina worden getoond.

Alle knoppen die op een resultaatpagina worden getoond worden door labels gegenereerd.

De tekst van een button dan wel het beeld dat moet worden weergegeven kan opgegeven worden. Dit ziet er dan bijvoorbeeld zo uit:

tekst

```
<% NEXT_BUTTON TEXT=" Next Page " %>
```

beeld

```
<% NEXT_BUTTON IMAGE="next.jpeg" %>
```

De tekst die op het beeld volgt definieert de url naar dit beeld.

Wanneer er een beeld gedefinieerd wordt, word alles wat na de image tag wordt weergegeven aan de HTML input tag meegegeven.

Bijvoorbeeld:

```
<% NEXT_BUTTON IMAGE="next.jpeg" BORDER=0 ALIGN=center%>
```

wordt dan:

```
<INPUT NAME="(internal name)" TYPE=image SRC="next.jpeg" BORDER=0  
ALIGN=center>
```

Alle HTML optie die voor de Image tag staan worden genegeerd. De border word in de huidige link kleur getoond. Als er geen border wordt opgegeven word standaard border = 0 gebruikt (geen border)

Er zijn knoppen voor het versturen van een zoekopdracht, het navigeren door de resultaatpagina's en voor het oproepen van extra pagina's zoals bijvoorbeeld de "collection basket" of een categorieëntabel.

In een resultaat pagina definitie kan de Info pagina opgeroepen worden middels een info knop of link.

Nieuwe Zoekopdracht

Terug naar de zoekpagina

label: "NEW_BUTTON"

Standaard tekst: "Start Over"

Verstuur een Zoekopdracht

Zoek in de gehele database

label: "SEARCH_ALL_BUTTON"

Standaard tekst: "Find"

Zoek in de huidige gevonden reeks

Deze knop verschijnt alleen wanneer het resultaat uit meer dan één pagina bestaat.

label: "SEARCH_FOUND_BUTTON"

Standaard tekst: "Find Again"

Verstuur een zoekopdracht

Zoek in de opgegeven reeks

Zoek in de reeks welke door de "searchRange" optie is bepaald.

label: "SEARCH_BUTTON"

Standaard tekst: "Find"

Navigatie door de Resultaatpagina

Knoppen die naar de volgende en vorige pagina met zoekresultaten leid.

Als er geen vorige dan wel volgende pagina bestaat word die knop niet getoond.

label: "PREVIOUS_BUTTON"

Standaard tekst: "Previous"

label: "NEXT_BUTTON"

Standaard tekst: "Next"

"Collection Basket" Knop

Deze knop toont de geselecteerde records.

"BASKET_PAGE" (zie ook sectie 3.3)

label: "BASKET_BUTTON"

Standaard tekst: "Collection Basket"

Categorieën Tabel Knop

Deze knop toont een categorieën tabel die gedefinieerd word via de

"CATEGORY_PAGE" template (zie ook sectie 3.3)

label : "CATEGORY_BUTTON"/"KEYWORD_BUTTON"

Standaard tekst: "Category Search"

3.2.6 Result Heading (Aantal gevonden records en hoeveel er getoond worden.)

De result heading wordt gegenereerd door de RESULT_HEADING label.

Deze label kan niet hard gedefinieerd worden aangezien er een aantal variabele waardes zijn.

Er zijn een aantal speciale labels die gebruikt kunnen worden om deze heading te definiëren, dit zijn “ItemsFound”, “FirstItem”, “LastItem

labels:

- “RESULT_HEADING_NONE”
- “RESULT_HEADING_SINGLE”
- “RESULT_HEADING_ALL”
- “RESULT_HEADING_RANGE”

Er kan een naam aan de RESULT_HEADING tags gehangen worden zodat deze vanuit een start zoekpagina kan worden gedefinieerd.

Ook hier geldt weer dat als er geen naam als optie wordt meegegeven de template als de standaard word gebruikt.

waarde:

de tekst moet op dezelfde pagina als de label worden gezet.

Voorbeeld

RESULT_HEADING_NONE = No items found

RESULT_HEADING_SINGLE = One item found

RESULT_HEADING_ALL = <%itemsFound%> Items Found, Displaying All

RESULT_HEADING_RANGE = { <%itemsFound%> Items Found, Displaying:
<%firstItem%> - <%lastItem%> }

3.2.7 Resultaat Lijst

De resultaatlijst word gedefinieerd door de “RESULT_LIST” label

Deze label kan niet hard worden gedefinieerd. De resultaatlijst wordt opgebouwd vanuit een item template en een row separator template.

(geen “item separator”, omdat er op een rij meerdere items since several items can appear on a single row).

Er zijn ook verschillende resultaat lijst instellingen, deze worden omschreven in de hier opvolgende secties.

label: “ROW_SEPARATOR”

HTML code die het rij scheidingsteken definieert.

voorbeeld/default:

```
<hr>
```

label: "RESULT_ITEM"

HTML code en inhoud label verwijzingen die het weergeven van een item bepalen.
Deze definitie word ook gebruikt voor het weergeven van de collection basket.

voorbeeld:

```
RESULT_ITEM = {  
<TABLE WIDTH=100%>  
<TR><TD ALIGN="LEFT">  
<% field name=Thumbnail %></TD>  
<TD ROWSPAN=2>  
<% field name=Notes %> </TD></TR>  
<TR><TD ALIGN="LEFT" VALIGN="TOP">  
<% field name="Record Name" %> </TD></TR>  
</TABLE>  
}
```

3.2.8 De inhoud van de resultaatweergave

Alle labels die in de "RESULT_ITEM" label kunnen worden geplaatst

De "Field" label word gebruikt om de inhoud van een veld te tonen.

Achter de field tag moet de naam van een veld zoals het in Cumulus onder "Catalog Properties." De naam word case-sensitive vergeleken.

Een veld kan ook via "ID" argument worden aangeropen. Dit is de interne identificatie van het veld. (Bijvoorbeeld een Windows GUID). Deze ID kan bepaald worden door een record export te doen. De id's staan in de header van deze export.

(Zie ook sectie 3.1 voor de eigenschappen van een zoekopdracht).

De waarde die weergegeven wordt correspondeert met de huidige Asset tags.

Als een veld geen waarde heeft of niet bestaat word er niets weergegeven.

Fields

Opties:

"defaultValue"

De waarde die wordt weergegeven op het moment dat er geen waarde is voor het desbetreffende veld. Standaard wordt er niets weergegeven.

Argumenten voor verschillende soorten velden

Voor "string" velden

"length"

De maximale lengte die word weergegeven

Standaard worden alle tekens getoond.

"encoding"

Hiermee wordt de conversie van tekst naar HTML verzorgt.

Opties:

Plain:

Hierbij worden de speciale HTML karakters '<', '>', '"' and '&' vertaald naar html alternatieven. Dit zijn platformafhankelijke beschrijvingen van deze tekens. Line Breaks (Return) worden vervangen door "
".

"MacRoman," "IsoLatin":

converteert 8-bit ASCII karakters van Macintosh roman of ISO 8859 codering naar HTML alternatieven. Als toevoeging op de waardes die al door plain geconverteerd worden.

"restrictedMacRoman," "restrictedIsoLatin":

converteert 8-bit ASCII karakters van Macintosh roman of ISO 8859 codering naar HTML alternatieven. Als toevoeging op de waardes die al door plain geconverteerd worden.

HTML character zoals '<', '>', '"' and '&' en
 worden niet vertaald.

Hiermee kan bijvoorbeeld een string veld in een HTML TEXTAREA veld worden getoond.

"Pre": reformatie.

De tekst word letterlijk weergegeven, er vind geen conversie plaats.

Als er geen conversie plaats moet vinden maar men kan geen gebruik maken van de "Pre" optie (bijvoorbeeld om HTML weer te geven in het notes veld) dan kan daar middels een escape character (de \ backslash) voor gezorgd worden.

Het escape character (de \ backslash) moet dan voor het desbetreffende teken geplaatst worden. ('<', '>', '&', '"') met een backslash.

Voorbeeld:

```
"...gelinkt naar \<A HREF=\"/test.html\"> test \</A>..."
```

Er zijn nog additionele opties om een pad weer te geven.

Namelijk:

"Mac": display a path in Macintosh format,

"Url": in Url format (Blanks replaced by "%20" etc),

"Unix": in Unix formaat,

"Dos": in Dos formaat.

De standaard is **"IsoLatin"**.

Opties voor velden van het type "Real":

"measurement"

Hiermee word de maat weergave van dit veld bepaald

Opties zijn :

"0" : inches

"1" : centimeters

"2" : millimeters

"3" : punten

"4" : pixels

"5" : picas

Opties voor velden van het type “Picture”

Optie:

“constraint”

de maximum grote voor het getoonde beeld in pixels.

Deze waarde beperkt ook de grote van de verstuurd beeldinformatie.

De optie overschrijft de waarde die globaal toegekend is via “thumbnailConstraint” setting (zie ook sectie 3.1.2)

Optie

“framewidth”, “frameheight”

De framegrote in pixels, als deze waarden worden opgegeven heeft de thumbnail een onzichtbaar frame van deze grote.

HTML Argumenten zoals alignment worden doorgegeven aan de onderliggende “IMG” tag.

Opties voor velden van het type “Audio”

Velden van het type “Audio” worden weergegeven middels links die naar een Quikcktime audio stream verwijzen.

Er is hiervoor de optie die de weergave van de link tekst weergeeft.

Deze optie is “linkText”

Default word de veldnaam gebruikt.

Opties voor velden van het type “Categories”

Bij de categorieën worden de volgende opties ondersteund.

Optie:

“separator”

Een tekst die van de categorieën een lijst maakt.

Standaard word een return gebruikt.

Optie:

“length”

Het aantal nummers dat maximaal wordt getoond.

Standaard worden alle getoond.

Asset Access Labels

De gecatalogiseerde “assets” kunnen benadert worden via een hyperlink.

Een link kan door Web Publisher gegenereerd worden middels de “Link/EndLink” tags, of hij kan handmatig gegenereerd worden middels de AssetLinkURL” tag welke een complete url genereert welke verwijst naar de asset.

Web Publisher kan op het moment dat de link gegenereerd wordt bekijken of de asset benadert kan worden en als dit niet zo is wordt de link niet gegenereerd. Als dit niet gecontroleerd word en de asset is niet benaderbaar word er een error gegenereerd.

Als de eerste optie word gekozen vertraagd dit de opbouw van de pagina met de tijd die het duurt om de links te checken.

Het is mogelijk om de asset te converteren naar een lage resolutie JPEG, of om een link te hebben vanuit de OPI lage resolutie file naar het Hoge resolutie beeld.
Zie ook hoofdstuk 5 voor details.

label: "Link", "EndLink"

Opties:

"Alt" = "dontShow"

"Target" = "" Dit zorgt ervoor dat de preview in een ander venster wordt getoond.
Het werkt net als de Netscape HTML "Target" tag.

"Direct Path"

Een url relatief aan de server rootfolder, dit kan gebruikt worden voor links naar beelden die niet gecatalogiseerd zijn. Zoals bijvoorbeeld lage resolutie bestanden of een gestufte versie.

Voorbeeld:

```
<% link directPath="/Cumulus%20Previews/<% fileName encoding=URL%>.jpeg" %>  
<% link directPath="/Cumulus%20Previews/<% fileName encoding=URL%>.sit" %>
```

Let op het is mogelijk om labels te gebruiken binnen andere label definities.

Alles wat tussen de link argumenten staat word getoond als een link die behoort bij het beeld wat getoond word.

Als de Alt= dontShow" optie is toegevoegd word de link niet getoond op het moment dat er iets mis gaat bij het genereren van de link.

Anders staat de tekst er wel maar wordt het niet als een link getoond.

Voor informatie over deze constructie zie sectie 5.

"AssetLinkURL"

Optie:

"AltText"

een tekst die getoond wordt in plaats van de URL.

Standaard is dit de naam van de asset of unknown als deze niet bekend is.

"Key"

De key wordt gebruikt om de link te "encrypten" dit is alleen nuttig als de link gebruikt word in combinatie met de collection basket.

De collection basket CGI verwacht een "encrypted" asset link.en gebruikt deze key bij het "decrypten", deze key moet hier gedefinieerd worden.

Zowel de Link als de AssetLinkURL tag kunnen de Asset Conversion

Module aanroepen middels de volgende tag:

"Converter"

De case sensitive naam of het ID van de conversie module, deze naam is de naam zoals

deze binnen de applicatie onder Asset Storage Modules is opgegeven.

Optie:

“Param”

de parameters die aan de Conversie Module moeten worden doorgegeven.

Zie ook hoofdstuk 5.

Deze parameters moeten altijd opgegeven worden anders zou er een dialoogvenster waarin optie ingesteld kunnen worden op de server getoond kunnen worden.

“Fallback” = “Unconverted.”

Als deze optie gespecificeerd is wordt het origineel gebruikt op het moment dat er iets mis gaat met de conversie.

Default wordt het origineel niet gebruikt maar komt er een foutmelding.

Zowel “Link” als de “AssetLinkURL” tag mogen het checken van de toegankelijkheid van een asset uitstellen middels de volgende optie:

“resolving” = “later.”

Als deze optie gespecificeerd wordt, wordt de link gegenereerd ongeacht of de asset toegankelijk is. Dit is sneller maar kan links genereren naar bestanden die niet bestaan. Standaard wordt er meteen gechecked of een bestand bestaat.

Zowel de “Link” als de “AssetLinkURL” tag kunnen met opties gebruikt worden.

Dit kan middels de “passToLink” constructie.

Deze opties worden geactiveerd middels de “usePassedArgs” = “Yes.”.

Als “usePassedArgs” = “Yes.” is opgegeven worden de passToLink opties die zijn meegegeven gebruikt voor het creëren van de link.

Het doorgeven van linkopties.

Als een zoekresultaat actie is kunnen hier extra opties aan meegegeven worden die later gebruikt kunnen worden voor het creëren van een link.

Dit kan gebruikt worden voor de Collection basket pagina, bijvoorbeeld om een link die op de Checkout pagina wordt getoond te controleren.

Dit werkt alle op de collection basket pagina.

De HTML opties worden via een HTML INPUT tag met de naam “assetLinkArg” en het speciale label “<%passToLink%>.” Dit kan bijvoorbeeld gebruikt worden om een naam van een convertersietool aan een link mee te geven,

```
<INPUT NAME=assetLinkArg VALUE=<%passToLink Converter=”Pixel Image Converter”%>>
```

Alle opties die op een link van toepassing zijn kunnen hier ingevoerd worden.

De opties werken alleen op de Collectionbasket pagina en moeten dan ook binnen de <%basketOnly%>, <%endBasketOnly%> labels staan zodat deze niet op de standaard resultaatpagina verschijnen, deze pagina’s worden namelijk vanuit dezelfde template gegenereerd.

Andere labels

“Info_Button”

Standaard tekst: "Info" (Zie ook sectie 3.2.5)

Bij deze label wordt een informatie pagina getoond met de specifieke informatie over de asset. Zie ook sectie 3.3.2

"InfoLink", "EndInfoLink"

Optie;

"Target" = "" (alleen voor de InfoLink)

dit zorg ervoor dat de pagina in een nieuw venster geopend word..

Alles wat tussen deze labels staat ziet er uit als een link naar de informatie van de betreffende Asset.

"BasketMark"

Opties

"toBasketText", "inBasketText"

Met deze opties kan de tekst opgegeven worden die getoond word, de "toBasketText" word getoond op de resultaat pagina, de "inBasketText" word gebruikt in de collection basket zelf.

Door de optie "BasketMark" wordt een checkbox getoond welke gevolgd word door de tekst zoals gedefinieerd door de "toBasketText", "inBasketText" opties.

Als de box gechecked/geunchecked is word zodra de pagina via één van de knoppen op de pagina de asset toegevoegd dan wel verwijderd uit de collection basket.

De back en forward knoppen van de browser kunnen dit effect niet bereiken omdat deze de CGI niet aanroepen.

"basketOnly," "endBasketOnly,"

"resultListOnly," "endResultListOnly"

Deze tags zorgen ervoor dat er afhankelijk van of de resultaatpagina of de collection basket getoond word zichtbaar is.

Bijvoorbeeld:

"<% basketOnly%>a<% endBasketOnly%>"

Zal op de collectionbasket pagina als een zichtbare a worden getoond, terwijl hij in de resultaat pagina als commentaar in de html staat.

Dus als "<!--a-->", dit betekend dat hij daar niet zichtbaar is.

Omdat de definities gedeeld worden. tussen de resultaatpagina en de collection basket.

Deze tags kunnen er voor zorgen dat de resultaatpagina en de collection basket pagina er verschillend uitzien.

3.2.9 Resultaat Formaat Opties

label: "ROW_LENGTH"

Hiermee word aangegeven hoeveel items er op één regel moeten worden getoond. Standaard is de waarde van row_length 1.

label: "PAGE_LENGTH"

Hiermee word aangegeven hoeveel items er op de resultaatpagina moeten worden getoond.

Standaard is de waarde van page_length 6.

label: "ITEM_ALIGN", "ITEM_VALIGN"

Deze optie kan één van de volgende waardes hebben:

- "TOP"
- "MIDDLE"
- "BOTTOM"
- "LEFT"
- "CENTER"
- "RIGTH"

Deze waardes bepalen de horizontale en verticale uitlijning van de weergegeven items.

ROW_LENGTH word alleen gebruikt als de waarde groter is dan 1

De standaard waarden voor "ITEM_ALIGN" "ITEM_VALIGN" zijn "CENTER en "MIDDLE"

3.3 Andere Resultaat Pagina's

3.3.1 Zoekpagina

label: "QUERY_PAGE"

Met deze label word een zoekpagina gedefinieerd die verschijnt op de resultaatpagina's. Uiteraard moet daar deze label aangeropen worden. De standaardpagina's bevatten een referentie naar deze label.

Als deze label niet bestaat word een standaard definitie gebruikt.

3.3.2 Info Page

label: "INFO_PAGE"

Met deze label word een informatiepagina gedefinieerd die verschijnt als op de link op de resultaatpagina's word geklikt.

Uiteraard moet deze link in de replypage definite gezet worden. De standaardpagina's bevatten een referentie naar deze label.

Als deze label niet bestaat word een standaard definitie gebruikt.

3.3.3 Collection Basket Pagina de Checkout pagina

Deze pagina's worden vanuit de zoekpagina's aangeroepen middels de replypage label.

Dit betekend dan ook dat wanneer we een reply page als volgt hebben gedefinieerd:

```
<input type="hidden" name="replyPage" value="topit">
```

Wel de bijbehorende Basketpage en Infopage moeten definiëren als:

BASKET_PAGE(topit)
BASKE T_PAGE(topit)
CATEGORY_PAGE(topit)
CHECKOUT_ITEM(topit)
CHECKOUT_PAGE(topit)
EMPTY_BASKET_LIST(topit)
INFO_PAGE(topit)

Met deze label word collection basket pagina gedefinieerd die verschijnt nadat op de basketbutton geklikt word die op de resultaatpagina's.

Uiteraard moet deze label in de replypage definite aangeroepen worden. De standaardpagina's bevatten een referentie naar deze label.

Als deze label niet bestaat word een standaard definitie gebruikt.

Deze template mag HTML code bevatten alsmede een referentie naar de "BASKET_LIST" label (Zie ook sectie 6).

Net als bij resultaat formaat is er geen eigen definite voor BASKET_LIST, deze lijst word gegenereerd vanuit de "RESULT_ITEM" en "RESULT_ITEM_SEPARATOR" labels.

"EMPTY_BASKET_LIST"

De html die binnen dit label staat word getoond op het moment dat er geen items in de collection basket staan.

example/default: EMPTY_BASKET_LIST = <center> (Collection Basket is Empty)</center>

label: "CHECKOUT_PAGE"

Hiermee word de checkout pagina gedefinieerd.

Deze pagina word gegenereerd op het moment dat op de "Checkout" button geklikt word

van de collection basket pagina.

Deze template bevat een formulier welke een CGI aanstuurt die een handeling met de ingevoerde data uitvoert.

Er kan binnen het formulier een referentie worden opgenomen naar de "CHECKOUT_LIST" label, deze lijst is gegenereerd middels de "CHECKOUT_ITEM" label.

bijvoorbeeld:

```
CHECKOUT_PAGE = {
<HTML><HEAD>
(...)
<BODY>
(...)
<FORM ACTION="Collbasket.cgi" METHOD="post">
<CENTER><TABLE>
<%checkout_list%>
</TABLE></CENTER>
(...)
Naam:
<INPUT TYPE=text NAME=username SIZE=25 MAXSIZE=40>
e-mail:
<INPUT TYPE=text NAME=eMail SIZE=35 MAXSIZE=50>
<INPUT TYPE=submit VALUE=" Send Data ">
</FORM></BODY>
```

label: "CHECKOUT_ITEM"

Deze label kan HTML code en label referentie bevatten voor de items in de "RESULT_ITEM" label

voorbeeld:

```
CHECKOUT_ITEM = {
<TR><TD><%field name="Record Name"%> </TD>
<TD><INPUT TYPE="checkbox" checked NAME="item"
VALUE="<% field name="Asset Name" encoding=MAC%">" >
Confirm processing</TD></TR>
}
```

3.3.3a Knoppen welke toegankelijk zijn vanaf de Collection Basket Pagina

“BACK_TO_SEARCH_BUTTON”

Middels deze knop kan men terug naar de zoekpagina vanaf de Collection Basket Pagina. Deze knop kan ook vanuit de categorieën en de informatie pagina aangeroepen worden. Als ze daar gebruikt worden zullen ze automatisch worden omgeven door een <FORM>..</FORM> construct anders zou het een niet werkende knop zijn. Het is mogelijk het begin en eind van deze constructie te beïnvloeden middels de <%bform%> (begin formulier) en <%eform%> (eindig formulier) label te plaatsen op respectievelijk het begin een eindpunt van het formulier. Standaard bevat deze kop de tekst back to search page.

“CLEAR_BASKET_BUTTON”

Als deze knop word aangesproken word de collection basket geleegd. De knop is alleen zichtbaar als er documenten in de collection basket staan. Standaard tekst van deze knop is “Remove All”.

CHECKOUT_BUTTON”

Met deze knop word een pagina aangeroepen waarop de item die zich in de collection basket bevinden kunnen worden afgehandeld via een andere CGI dan de webpublisher cgi.

De knop is alleen zichtbaar als er documenten in de collection basket staan. Standaard tekst van deze knop is “Proceed”.

3.3.4 Andere Labels

label: “WebUser”

Deze label toont de gebruikersnaam zoals deze is aan de webserver is opgegeven (middels een inlogscher).

De waarde is leeg op het moment dat er niet word ingelogd.

“catalogName”

De naam van de catalogus waarin op dit moment gezocht word.

Als er in meerdere catalogi tegelijk gezocht word de eerste getoond.

label: “catalogNames”

Met deze label kan het scheidingsteken opgegeven worden voor de namen in de catalogNames lijst.

Standaard word “,” (een komma en een spatie) gebruikt.

label: “queryContents”

Hiermee kan de inhoud van de laatste zoekopdracht worden getoond als optie kan het maximale aantal woorden worden opgegeven.

label: "empty"

Deze label verdwijnt in het niets en is bedoeld om een formulier welke door de Cumulus cgi gegenereerd word zoals bijvoorbeeld de REPLY_PAGE en BASKET_PAGE eerder te laten beginnen. Er moet dan op de plaats waar het formulier moet beginnen een <%empty%> tag opgegeven worden. Als deze niet opgegeven word begint het formulier bij de eerste label die aangeroepen worden en houd hij op bij de laatste.

3.3.5 Categorieën Pagina (Zie ook Sectie 4)

label: "CATEGORY_PAGE"

Hierin kunnen zoals HTML codes als label referenties opgegeven worden om de weergave van de categorieën Pagina op te geven.

In deze template moet een verwijzing staan naar de "CategoryTable" label.

Default voorbeeld:

```
CATEGORY_PAGE = {  
<HTML><HEAD><TITLE> Category Table </TITLE>  
<BODY>  
<CENTER>  
<H1> Categories </H1>  
<H2> Showing the category table of <% catalogName %> </H2>  
</CENTER>  
<% categoryTable %>  
</BODY>  
</HTML>
```

label: "categoryTable"

optie: "Encoding" (Zie sectie 3.2.8)

opties:

- "listStart"
- "listEnd"

- “itemStart”
- “itemEnd”
- “indent”,
- “unindent”

dit zijn de HTML tags die bepalen hoe de categorieën tabel wordt getoond.

Standaard zijn dit “”, “”, “”, “”, “”, “”.

“indent” en “unindent” worden gebruikt om hiërarchische niveaus van de categorie te laten zien (zoals deze hiërarchie ook binnen Cumulus bestaat).

3.3.6 Foutmeldingen

Webpublisher genereert foutmeldingen op basis van de “ERROR_PAGE” template. Deze template moet een verwijzing hebben naar de labels “messageHeading” en “messageBody”.

Dit wordt dan als foutmelding getoond.

“ERROR_PAGE”

voorbeeld/standaard:

```
ERROR_PAGE = {  
<HTML><HEAD><TITLE> Error </TITLE>  
<BODY>  
<H2> <% messageHeading %> </H2>  
<% messageBody %>  
</BODY>  
</HTML>  
}
```

label: “messageHeading”, “messageBody”

opties:

```
“connectionExpired”  
“connectionDenied”,  
“accessDenied”  
“databaseNotFound”  
“unhandledError”,  
“unspecificQuery”  
“linkExpired”  
“noDatabaseOpen”,  
“noAudioData”  
“badSearchValue”  
“templateParsingError”
```

default:

```
messageHeading(connectionExpired) = { Connection has expired }  
messageBody(connectionExpired) = { Sorry, your connection has timed out. Please
```

```
submit a new search. }
messageHeading(connectionDenied) = { No Admittance }
messageBody(connectionDenied) = { Sorry, too many users are currently logged in.
Please try again later. }
messageHeading(accessDenied) = { Access Denied }
messageBody(accessDenied) = { Sorry, you are not allowed to connect to this
catalog. }
messageHeading(databaseNotFound) = { Catalog Not Found }
messageBody(databaseNotFound) = { Sorry, the catalog you tried to connect to is not
available. If the problem persists, please contact
the site administrator. }
messageHeading(unspecificQuery) = { Query too unspecific }
messageBody(unspecificQuery) = { Sorry, the query you entered was too
unspecific. Please try again with a more precise
entry. }
messageHeading(linkExpired) = { Link Expired }
messageBody(linkExpired) = { Sorry, this link is not valid anymore.}
messageHeading(noDatabaseOpen) = { No Catalog Found }
messageBody(noDatabaseOpen) = { Sorry, no catalog is currently available for
searching. If the problem persists, please contact
the site administrator.}
messageHeading(noAudioData) = { No Audio Data }
messageBody(noAudioData) = { Sorry, could not create audio data for this link. If
the problem persists, please contact the site
administrator.}
messageHeading(badSearchValue) = { Unexpected Search Entry }
messageBody(badSearchValue) = { Sorry, your search condition contains invalid
data.}
messageHeading(recordDeleted ) = { Record Deleted }
messageBody(recordDeleted) = { This record has been deleted from the catalog.
}
messageHeading(templateParsingError) = { Template Parsing Error }
messageBody(templateParsingError) = { Error parsing HTML Templates:
<%templateErrorMessage%> (Error number
<%errorNumber%>)}
(the "templateParsingError" message takes the internal error message and number and
inserts it at the <% templateErrorMessage %> and <%errorNumber%> label reference)
messageHeading( unhandledError) = { We Regret... }
messageBody( unhandledError) = { An unhandled error ocurred during transaction
(Error number <%errorNumber%>). }
(the "unhandled Error" message takes the error number and inserts it at the
<%errorNumber%> label reference. This reference can be omitted if the error number
should not be displayed)
```

3.4 Data die zich niet in Cumulus bevindt via Webpublisher benaderen

Het is mogelijk om met webpublisher extra informatie te publiceren (informatie die niet in de Cumulus database staat).

De data wordt dan aan de webpublisher “gevoerd”.

Later kan deze data opgehaald worden en getoond op een resultaatpagina (ook info collectionbasket etcetera).

De data wordt aan een zelf gedefinieerde variabele toegewezen.

Deze data is vervolgens de hele sessie beschikbaar en kan middels een label die naar deze zelf gedefinieerde variabele verwijst opgevraagd worden.

3.4.1 Zelf een eigen variabele definiëren

Een eigen variabele moet in de HTML Templates worden gedefinieerd voor hij gebruikt kan worden

Dit ziet er dan als volgt uit.

```
#defvar VariableName
```

Deze definitie moet op één lijn staan.

“VariableName” mag uit letters en cijfers bestaan er mogen geen spaties in voorkomen.

3.4.2 Waardes instellen en opvragen

De waarde van een variabel kan worden ingesteld in een HTML form middels een `<INPUT>`

tag, zoals:

```
<INPUT TYPE=HIDDEN NAME=VariableName VALUE="example value">
```

De waarde kan vervolgens weer opgevraagd worden middels een label

Bijvoorbeeld: `<% VariableName %>`

Dit kan vanuit een van de templates zoals die hierboven zijn beschreven

In dit voorbeeld word de waarde "example value" getoond .

4 Categorieën Tabel Weergave

The Web Publisher genereert een pagina waarop een Categorieën tabel word getoond op het moment dat de CGI aangeroepen word met de optie “search”, submit button name of “categories”, of wanneer de category knop op een reply page word ingedrukt.

De tabel word getoond zoals hij in Cumulus zichtbaar is op het moment dat alle categorieën hiërarchisch getoond worden.

Elke categorie die getoond word is een link welke de database doorzoekt voor deze betreffende categorie.

Wat betreft de weergave van de categorieën tabel worden de instellingen gebruikt, zoals die in de pagina zijn opgegeven, die de tabel heeft aangeroepen

Voorbeeld.:

```
<FORM ACTION="cuwp_cgi.exe$search" METHOD="POST">
<INPUT TYPE="HIDDEN" NAME="catalogName" VALUE="Flora">
<!-- the database name -->
<INPUT TYPE="HIDDEN" NAME="replyFormat" VALUE="textAndThumbnail">
<!-- the reply format -->
(... code for query submission...)
<INPUT TYPE="SUBMIT" Name="categories" VALUE="Category Search">
<!-- the button -->
</FORM>
```

Open resultaatpagina welke ook een zoekpagina bevat kan een knop getoond worden welke de Categorieën Tabel door middel van het volgende label.

```
<% CATEGORY_BUTTON text=" Category Search" %>
```

Deze knop toond de categorieën tabel van de huidige actieve catalogus.

De weergave wordt bepaald door de definitie van de resultaatpagina zoals in het zoekformulier is opgegeven.

De categorieën pagina word getoond op basis van de template “CATEGORY_PAGE”, waarop een “categoryTable” label moet staan. Zie ook sectie 3.3.4

5 Asset Downloaden

Assets kunnen worden gedownload, als ze toegankelijk zijn voor de Web Publisher server.

Dit wil dan ook zeggen dat de documenten op de server zelf aanwezig zijn of via de Web Publisher server machine te benaderen (op een gemounte server bijvoorbeeld.)

De webserver zelf (dit hoeft namelijk niet de Web Publisher server te zijn) hoeft geen toegang te hebben tot de assets.

De datatransfer word volledig door de Web Publisher server afgehandeld en gelogd/

De download log entry word pas gemaakt op het moment dat de download klaar is.

5.1 Link Constructie

Elk karakter dat in een template staat kan veranderd worden in een download link naar de getoonde asset.

Dit wordt gedaan door de referentie te omvatten met de <% Link %>, <% EndLink %> tags.

Harde pad data kan worden opgegeven middels de “directPath” tag. Dit kan bijvoorbeeld worden gebruikt om een link te creëren naar een lage resolutie versie van het origineel, of bijvoorbeeld een ingepakt (.sit) bestand in plaats van een link naar het origineel.

Deze tag is alleen aanwezig voor compatibiliteit de Cumulus Internet Media Server 4.0, het word dan ook afgeraden deze te gebruiken.

Een betere manier om ditzelfde te bereiken is door gebruik te maken van de Asset Converter Modules om on the fly previews te creëren.

De link word getoond in de resultaat pagina op het moment dat de asset waarnaar gerefereerd word toegankelijk is, en of de desbetreffende gebruiker (als er geen gebruikers zijn is dit uiteraard de guest) de “transfer assets” permissie heeft binnen Cumulus.

Als niet aan deze criteria word voldaan, word geen link getoond maar alleen de tekst.

Deze kan ook verdwijnen door de extra optie <% Link Alt=“DontShow” %> . op te nemen binnen de linkconstructie.

Het kijken of een asset toegankelijk is kan enige tijd in beslag nemen.

Dit kan worden uitgezet worden door de “resolving=later” optie in the link constructie te gebruiken.

In dit geval word de link meteen gecreëerd, uiteraard word er dan altijd een link getoond ongeacht of de asset toegankelijk is. Er word op dat moment een foutmelding getoond aan de gebruiker.

5.2 Asset Conversie

Standaard word de originele data gedownload als er op een download link geklikt word.

Het is mogelijk de data te laten behandelen door een Asset Conversie module om een lage resolutie versie te verkrijgen of bijvoorbeeld een sample welke binnen een OPI proces gebruikt kan worden.

5.2.1 Converters en Opties

Canto Software levert een pixel image converter en een OPI converter.

The pixelimage converter maakt een JPEG van de Asset en heeft de volgende opties.
param=“Scale=100% JPEGQuality=3”

De “Scale” optie omschrijft het formaat van de JPEG welke gemaakt wordt.

Als het “%” achtervoegsel word weggelaten word de waarde als aantal pixels gelezen en wordt op die manier de grote in pixel gelimiteerd.

De “JPEGQuality” optie accepteert waardes van 0 (erg slechte kwaliteit) tot en met 10.

De OPI support module zorgt voor het terugvinden van het juiste OPI beeld.

De opties zijn param=“GetHiRes” welke een link creëert naar het hoge resolutie beeld.

param=“GetLowRes” welke een link creëert naar het lage resolutie beeld

De converters kunnen middels de (user interface) naam aangeroepen of middels hun ID welke aan het Operating System is opgegeven.

Deze ID's zijn.

Pixel Image Converter: {6244BF10-976D-11d3-9F4F-0080AD783050}

OPI System Support: {FCD43330-CD7C-11d2-8FA2-0000C0E166DC}

5.2.2 Voorbeelden

Om een link te creëren naar een lage resolutie OPI beeld welke niet verschijnt als het origineel niet word gevonden.

```
<%link converter="OPI System Support" param="GetLowRes" alt=dontShow%>Get Low Resolution<%endLink%>
```

Om een lage resolutie beeld te creëren van een asset en het origineel te gebruiken op het moment dat de asset niet geconverteerd kan worden:

```
<IMG SRC="<%assetLinkURL converter="Pixel Image Converter" param="Scale=400 JPEGQuality=2" fallback=unconverted%>" ALT="<%field name="Asset Name"%> Preview">
```

Let op de converters worden aangeroepen middels hun taalafhankelijke user interface namen

Als er gebruikt word gemaakt van bijvoorbeeld de Duitse versie van Cumulus zullen de namen anders zijn.

Het is altijd mogelijk de convertor middels zijn ID aan te roepen.

```
<%link converter="{FCD43330-CD7C-11d2-8FA2-0000C0E166DC}" param="GetLowRes" alt=dontShow%>Get Low Resolution<%endLink%>
```

6 Collection Basket

De resultaatpagina's kunnen checkboxes bevatten welke door een gebruiker geselecteerd kunnen worden zodat die asset in de "collection basket" word gezet.

Deze worden per sessie in één lijst gezet, er kunnen items van meerdere zoekopdrachten in één collection basket terechtkomen.

Op het moment dat de gebruiker naar de collection basket gaat krijgt hij daar de huidige selectie te zien en kunnen er nog items uit de lijst verwijderd worden.

Een "Check out" knop leid dan naar een andere pagina vanwaar een cgi kan worden aangeroepen welke met de geselecteerde assets een actie uitvoert.

Hier kan ook extra informatie betreffende de gebruiker/bestelling worden gevraagd uiteraard moet de CGI welke de afhandeling verzorgt met deze data overweg kunnen.

De checkboxes verschijnen op de resultaat pagina op het moment dat daar de "BasketMark." Label aan is toegevoegd.

Als een checkbox aangevinkt is word de asset aan de collection basket lijst toegevoegd zodra de pagina word verlaten middels één van de knoppen welke beschreven worden in sectie 3.2.5.

Als deze label op de collection basket pagina verschijnt kan de checkbox gebruikt worden om items te verwijderen uit de lijst.

De Collection Basket CGI welke word meegeleverd met Cumulus 5 creëert links naar de items die getoond worden op de Collection Checkout page.

7 De Templates van Cumulus Web Publisher 5.0

7.1 The Standaard Templates

De Template folder van deze versie van Web Publisher bevat de volgende files

- 1) Main.html
- 2) QueryForms.html
- 3) SearchResult.html
- 4) SearchResult no Basket.html
- 5) CollBasketSample.html
- 6) defaults.html

Deze bestanden bepalen de opbouw van pagina's zoals beschreven in hoofdstuk 3. Het bestand "Main.html" bevat geen template definities, maar wordt gebruikt om de andere templates aan te roepen.

7.1.1 Standaard Resultaat Pagina

De resultaatpagina's worden gedefinieerd in de "SearchResult.html" en "SearchResult no Basket.html". De collection basket is alleen beschikbaar vanuit de "SearchResult.html."

Hoe de pagina's eruit zien kan dan ook heel verschillend zijn afhankelijk van of de collection basket gebruikt wordt.

Welke resultaatpagina wordt getoond hangt af van de "replyPage" optie welke binnen de zoekpagina is opgegeven.

Alle zoekpagina's worden gedefinieerd in het bestand "QueryForms.html."

Er zijn twee soorten zoekpagina's deze komen overeen met de simplesearch en de compoundsearch zoekpagina's. Elke pagina wordt twee keer gedefinieerd, één keer met basket en één keer zonder basket. Dit komt omdat de layout met en zonder basket iets verschilt.

Welke wordt getoond is afhankelijk van de "queryForm" optie welke op de start zoekenpagina wordt opgegeven.

Het resultaat kan standaard op drie manieren worden getoond:

- text view
- thumbnail view
- text and thumbnail view

Elke definitie is aanwezig in zowel de "SearchResult.html" als de "SearchResult no Basket.html" pagina.

Welke van de pagina's getoond wordt is afhankelijk van de "resultFormat" optie

Deze optie wordt via de resultaat formaat selectie checkboxes die zichtbaar zijn op de start zoekenpagina's.

7.1.2 Andere Pagina's

Voor de meeste andere pagina's zoals de Category pagina en de collectionBasket pagina, worden de standaard definities gebruikt.

Deze staan de het bestand "defaults.html."

Hier bevinden zich ook de foutmeldingen en andere optie die hoofdstuk 3 beschreven zijn. Ook de informatie pagina is hier gedefinieerd.

7.1.3 Cross Platform Installatie

Als de Web Publisher server component de CGI op verschillende platformen zijn geïnstalleerd moet het pad welke naar de collection basket verwijst veranderd worden. Deze bevind zich in de CollBasket.html.

Met een Windows CGI is het pad /scripts/collbasket5/collbasket.dll

Met een Mac OS CGI is het CollectionBasket.acgi

7.2 Templates creëren van de huidige weergave binnen Cumulus

Het script "Create Web Publisher Template" maakt het mogelijk om een template definitie van de huidige weergave binnen een Cumulus application te maken.

Als deze functie vanuit het juiste Cumulus menu commando aangeroepen is, word er een bestand gecreëerd genaamd "myTemplates.html."

Dit bestand bevat voornamelijk resultaat pagina definities, deze zijn alle aan te roepen middels de replyPage=myTemplates en resultFormat=myTemplates.

Uiteraard moet dit bestand dan middels de defaults.html geinclude worden en in de "Templates" folder gezet worden.

Vervolgens kunnen ze aangeroepen worden vanuit de zoekpagina's middels bovenstaande commando.

De template die gegenereerd word bevat geen definitie voor een zoekformulier aangezien de Cumulus weergave ook geen zoekformulier heeft.

Welk zoekformulier word gebruikt word dus bepaald door de optie die in de start zoekenpagina is opgegeven.

8 Web Publisher versus Internet Media Server 4.0

8.1 Nieuwe Features

8.1.1 HTML Templates Opnieuw Laden

De HTML Templates worden opnieuw geladen als de WebPublisher CGI word aangeroepen.

Veranderingen aan de templates worden doorgevoerd zodra ze opgeslagen zijn. Als er problemen zijn met het interpreteren van de templates word er een foutmelding in de log gezet en deze word in de web browser getoond de eerste keer dat het probleem zich voordoet. Alle volgende zoekopdrachten worden door de default pagina templates afgehandeld.

8.1.2 Resultaat Sorteren

Het is nu net als binnen de cumulus applicatie mogelijk om resultaten te sorteren en een sorteerrichting mee te geven.

8.1.3 on-the-fly Asset Conversie

De manier waarop de assets worden benaderd is behoorlijk veranderd in versie 5.

Dit is mede omdat assets niet perse direct toegankelijk zijn.

Daarom wordt het downloaden van de asset nu afgehandeld via de Web Publisher.

Ook is het nu mogelijk om on the fly conversies te doen van assets.

Bijvoorbeeld in een lage resolutie JPEG.

8.2 Backward Compatibiliteit

De HTML interface van Webpublisher komt voor een groot gedeelte overéén met de interface zoals deze voor Cumulus internet media server bestond.

Het is dan ook mogelijk om de eigen pagina's van Cumulus 4 met kleine aanpassing te gebruiken met Cumulus 5.

Het word vanuit Canto aangeraden om de nieuwe collection basket te gebruiken omdat de manier waarop download worden verzorgd is veranderd.

De naam van de catalogus welke doorzocht moet worden moet case-sensitive worden opgegeven.

De manier waarop de download en de preview link gemaakt worden is veranderd.

Het grootse verschil is dat de optie "resolveFor" niet meer bestaat.

Als de bestandsgrote weergegeven word, word de eenheid er automatisch achter gezet:

- Byte
- Kilobyte
- Megabyte

Het is niet nodig om de tekst kB in de HTML templates te zetten.

Het resolutie veld kan standaard niet doorzocht worden in Cumulus 5, als de "oude" compoundSearch pagina gebruikt word (hierin kan het veld resolutie wel doorzocht worden)

In dit geval moet het veld "horizontal resolution" voor zoeken geïndexeerd worden.

· De fileLocation en webPassword labels die in versie 4 wel gebruikt kunnen worden maar deze zijn in versie 5 verwijderd.