Web URL Tracking

By Timothy Aaron Penner, Technical Services Team Member, 4D Inc.

Technical Note 11-20

Table of Contents

Abstract

The concept of URL/Link tracking is used for numerous things on the internet; one common thing this is used for is marketing campaigns.

When running a marketing campaign it is often times useful to track how many times each of the links used within the campaign were clicked. This information can be used to understand which links were clicked more frequently and can later be used to refine the campaign or give reports on how the campaign did.

This Technical Note describes a general approach of tracking the links for such campaigns. Included with this Technical Note is a sample database that demonstrates the concepts. Also included is an example HTML file to help demonstrate the functionality of the sample database.

Introduction

When talking about Marketing Campaigns, one question that is often times asked is 'how well did the campaign do'? This is commonly referred to as metrics.

The problem with collecting these metrics is that page clicks are usually logged on the server hosting the page. What if you are not hosting the page, or if at least you do not have access to the server logs to obtain such data, how could one tell the amount of traffic that the link generated?

This Technical Note describes an approach for handling just that, and also provides a sample database that demonstrates the approach.

Concepts behind URL/Link Tracking

The concept behind URL / Link tracking is rather simple:

A link exists that must be distributed and it is necessary to know how many people click the link.

This is accomplished in this Technical Note by using a 4D database to store the links in a table. Each link is its own record in the database with a counter field that is increased each time the link is accessed.

The links are presented to the end-user referenced by UUID instead of by the actual link, in this way the destination is obfuscated until translated by the 4D database. The true destination of the link is only known by the 4D database; therefore the UUID must be translated by the 4D database. When the 4D database translates the link it also increases the counter for the link while at the same time redirecting the end-user to the correct location on the web.

In this way, the counter is silently increased and the end-user gets the page they want.

Storing Links in the Database

A source document or snippet of HTML must first be parsed and the links found must then be saved in to the database. Once the links have been saved in the database, each with their own unique ID, the source document or snippet of HTML must have the links replaced with new ones that reference the UUID of the stored link.

For more information and an example approach, please see the '<u>How tracked links are stored in the database</u>' section within the '<u>Sample Database</u>' chapter of this document.

Retrieving Links from the Database

A method must exist in the database to look up the link by UUID and return the destination URL while at the same time increasing the counter. An entry point for the method must exist, either in *On Web Connection* or through *4DACTION*, so that a simple HTTP call to the 4D database does a *SEND HTTP REDIRECT* to the destination URL while also increasing the counter for the link.

For more information and an example approach, please see the '<u>How tracked links are resolved though the web</u>' section within the '<u>Sample Database</u>' chapter of this document.

Sample Database

Included with this Technical Note is a sample database that demonstrates some of the concepts discussed within this document.

The sample database goes a step further with the concept and implements campaigns.

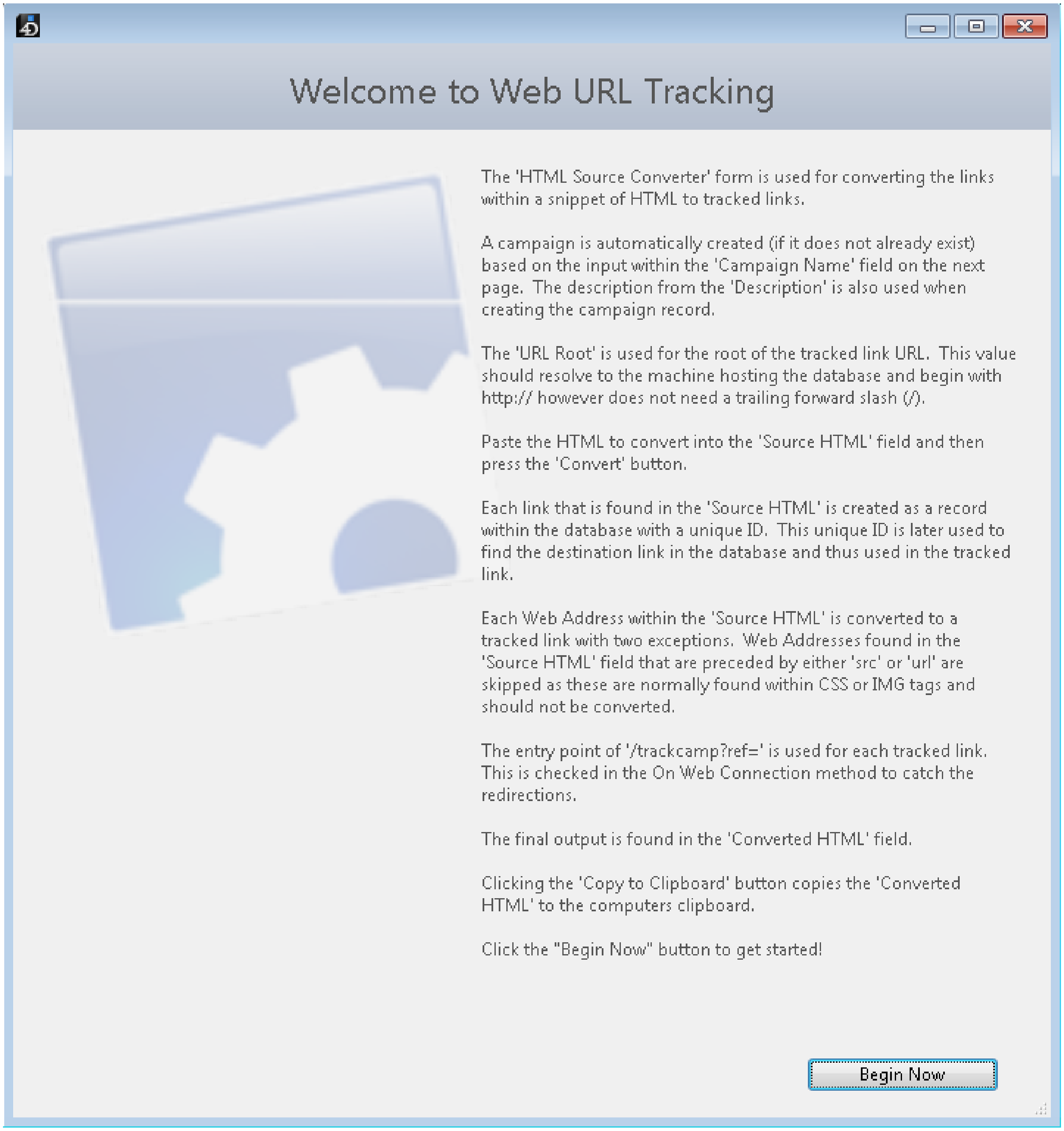
Usage of Campaigns

Campaigns are an easy way to group the links and are similar to projects. Campaigns are useful when a URL might be used multiple times but need to be tracked separately.

For example, the main URL of a website (<u>www.myproduct.com</u>) might be re-used in multiple communications. Say this URL is used in a communication regarding the release of a 'New Version' as well as a separate communication regarding a 'Trade Show'. The use of campaigns allows these to be tracked independently from each other. So the clicks from the 'New Version' campaign are counted separately from the clicks for the 'Trade Show' campaign.

HTML Source Converter

When the demo database is launched the HTML Source Converter is automatically displayed:



Welcome to Web URL Tracking

The 'HTML Source Converter' form is used for converting the links within a snippet of HTML to tracked links.

A campaign is automatically created (if it does not already exist) based on the input within the 'Campaign Name' field on the next page. The description from the 'Description' is also used when creating the campaign record.

The 'URL Root' is used for the root of the tracked link URL. This value should resolve to the machine hosting the database and begin with http:// however does not need a trailing forward slash (/).

Paste the HTML to convert into the 'Source HTML' field and then press the 'Convert' button.

Each link that is found in the 'Source HTML' is created as a record within the database with a unique ID. This unique ID is later used to find the destination link in the database and thus used in the tracked link.

Each Web Address within the 'Source HTML' is converted to a tracked link with two exceptions. Web Addresses found in the 'Source HTML' field that are preceded by either 'src' or 'url' are skipped as these are normally found within CSS or IMG tags and should not be converted.

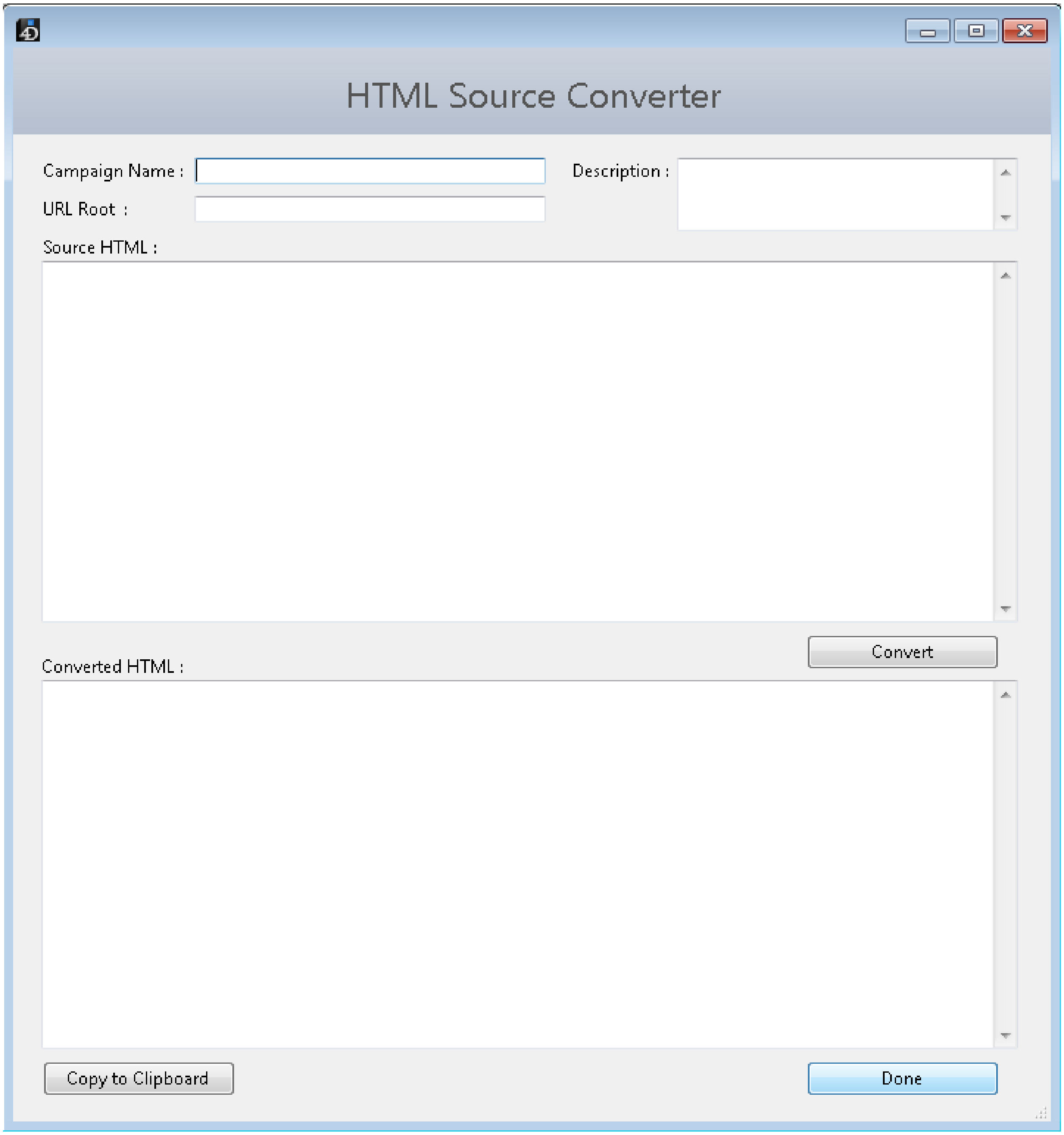
The entry point of '/trackcamp?ref=' is used for each tracked link. This is checked in the On Web Connection method to catch the redirections.

The final output is found in the 'Converted HTML' field.

Clicking the 'Copy to Clipboard' button copies the 'Converted HTML' to the computers clipboard.

Click the "Begin Now" button to get started!

After clicking on the 'Begin Now' button to get started, the following form is displayed:



HTML Source Converter

Campaign Name :

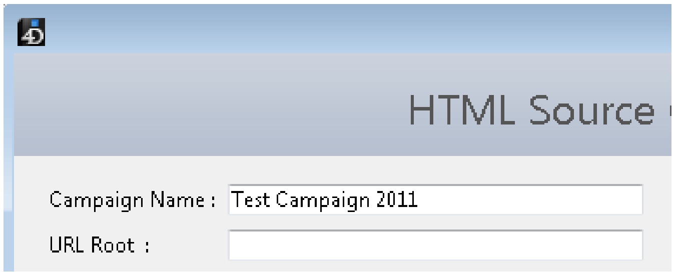
URL Root :

Description :

Source HTML :

Converted HTML :

Enter a campaign name to be used in the 'Campaign Name' field:



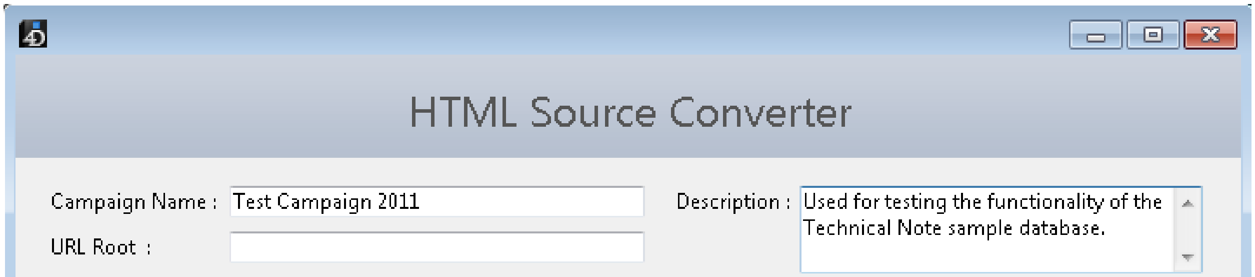
HTML Source

Campaign Name : Test Campaign 2011

URL Root :

The information entered into the 'Campaign Name' field is used for storing the links once the 'Convert' button is clicked. A parent record is used for the campaign while each link of the campaign is stored in a related table as a child of the parent record.

Enter a Description for the campaign in the Description field:



HTML Source Converter

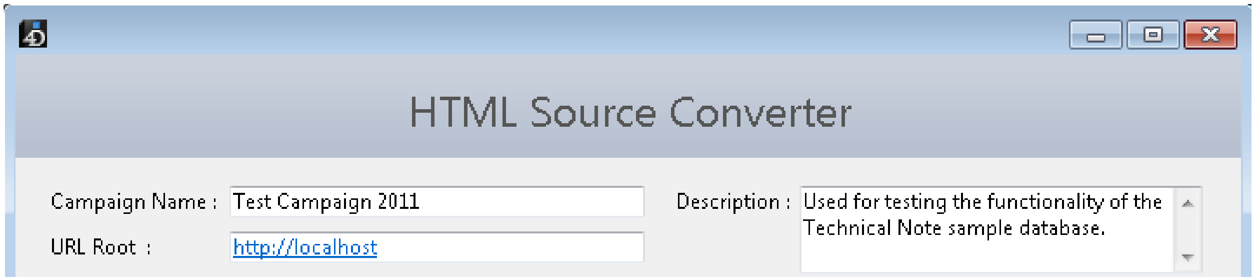
Campaign Name : Test Campaign 2011

URL Root :

Description : Used for testing the functionality of the Technical Note sample database.

The description is saved with the parent record and is later displayed when viewing the statistics of the campaign.

Enter the domain name or IP address of the machine hosting the 4D database, preceded with http:// without a trailing slash, into the URL Root field:



HTML Source Converter

Campaign Name : Test Campaign 2011

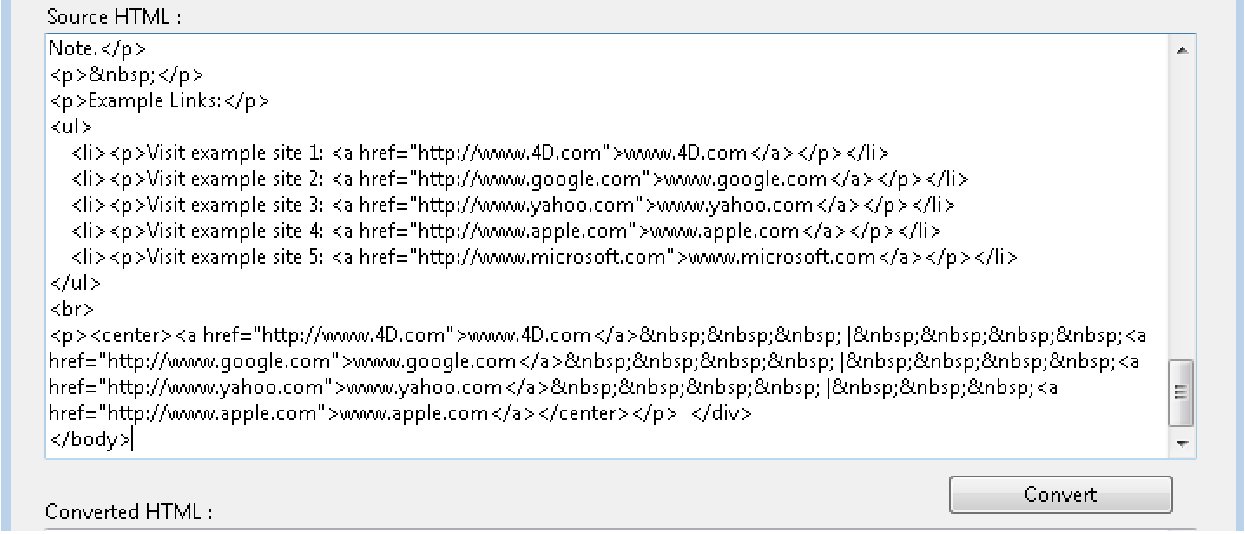
URL Root : http://localhost

Description : Used for testing the functionality of the Technical Note sample database.

In the sample screenshot shown above, a URL root of 'http://localhost' is used.

NOTE: localhost resolves to the local machine and thus will not be suitable unless the database is running on the same machine that displays the HTML page.

Take the source html to convert and paste it in the field named 'Source HTML':

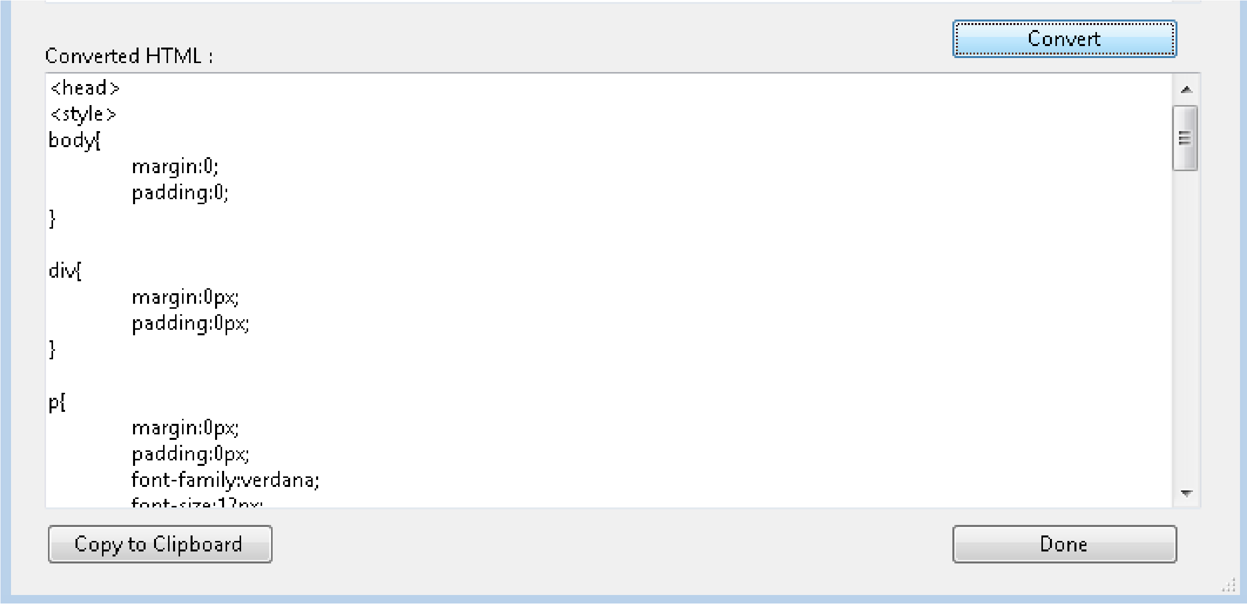


```
Note.</p>
<p>&nbsp;</p>
<p>Example Links:</p>
<ul>
   <li><p>Visit example site 1: <a href="http://www.4D.com">www.4D.com</a></p></li>
   <li><p>Visit example site 2: <a href="http://www.google.com">www.google.com</a></p></li>
   <li><p>Visit example site 3: <a href="http://www.yahoo.com">www.yahoo.com</a></p></li>
   <li><p>Visit example site 4: <a href="http://www.apple.com">www.apple.com</a></p></li>
   <li><p>Visit example site 5: <a href="http://www.microsoft.com">www.microsoft.com</a></p></li>
</ul>
<br>
<p><center><a href="http://www.4D.com">www.4D.com</a>&nbsp;&nbsp;&nbsp; |&nbsp;&nbsp;&nbsp;&nbsp;<a
href="http://www.google.com">www.google.com</a>&nbsp;&nbsp;&nbsp;&nbsp; |&nbsp;&nbsp;&nbsp;&nbsp;<a
href="http://www.yahoo.com">www.yahoo.com</a>&nbsp;&nbsp;&nbsp;&nbsp; |&nbsp;&nbsp;&nbsp;<a
href="http://www.apple.com">www.apple.com</a></center></p>   </div>
</body>
```

Once the content is verified to be accurate, click on the 'Convert' button to find all links and convert them to tracked links.

NOTE: URL's that are preceded with 'URL' or 'SRC' are not converted. The 'URL' notation is often found in CSS and the 'SRC' notation is found in both image and script tags. Neither of which should be converted.

After clicking on the 'Convert' button, the converted source with tracked links is available in the 'Converted HTML' field:



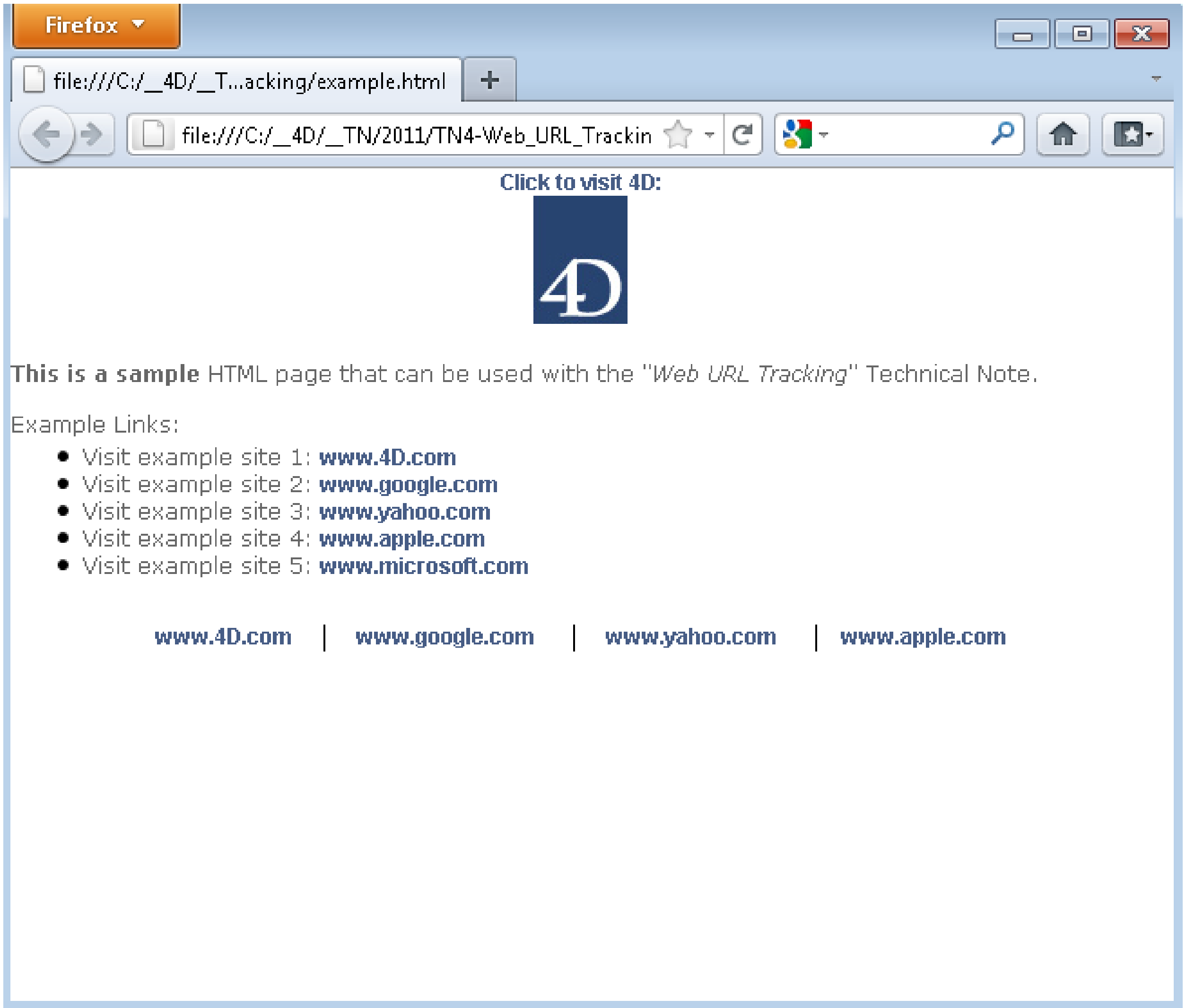
```
<head>
<style>
body{
            margin:0;
            padding:0;
}

div{
            margin:0px;
            padding:0px;
}

p{
            margin:0px;
            padding:0px;
            font-family:verdana;
            font-size:12px;
```

Select the text from the ‘Converted HTML’ field and copy it to the clipboard (or use the ‘Copt to Clipboard’ button). This is the HTML that includes tracked links and should be used in the communication.

An example HTML file is included with this Technical Note and is used in the above screenshots. Here is what the page looks like when rendered in a web browser:


Firefox
file:///C:/__4D/__T...acking/example.html
file:///C:/__4D/__TN/2011/TN4-Web_URL_Trackin

Click to visit 4D:

This is a sample HTML page that can be used with the "*Web URL Tracking*" Technical Note.

Example Links:

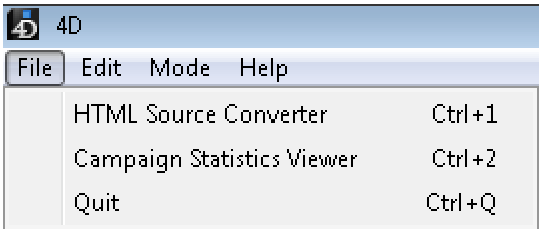
- Visit example site 1: **www.4D.com**
- Visit example site 2: **www.google.com**
- Visit example site 3: **www.yahoo.com**
- Visit example site 4: **www.apple.com**
- Visit example site 5: **www.microsoft.com**

www.4D.com | **www.google.com** | **www.yahoo.com** | **www.apple.com**

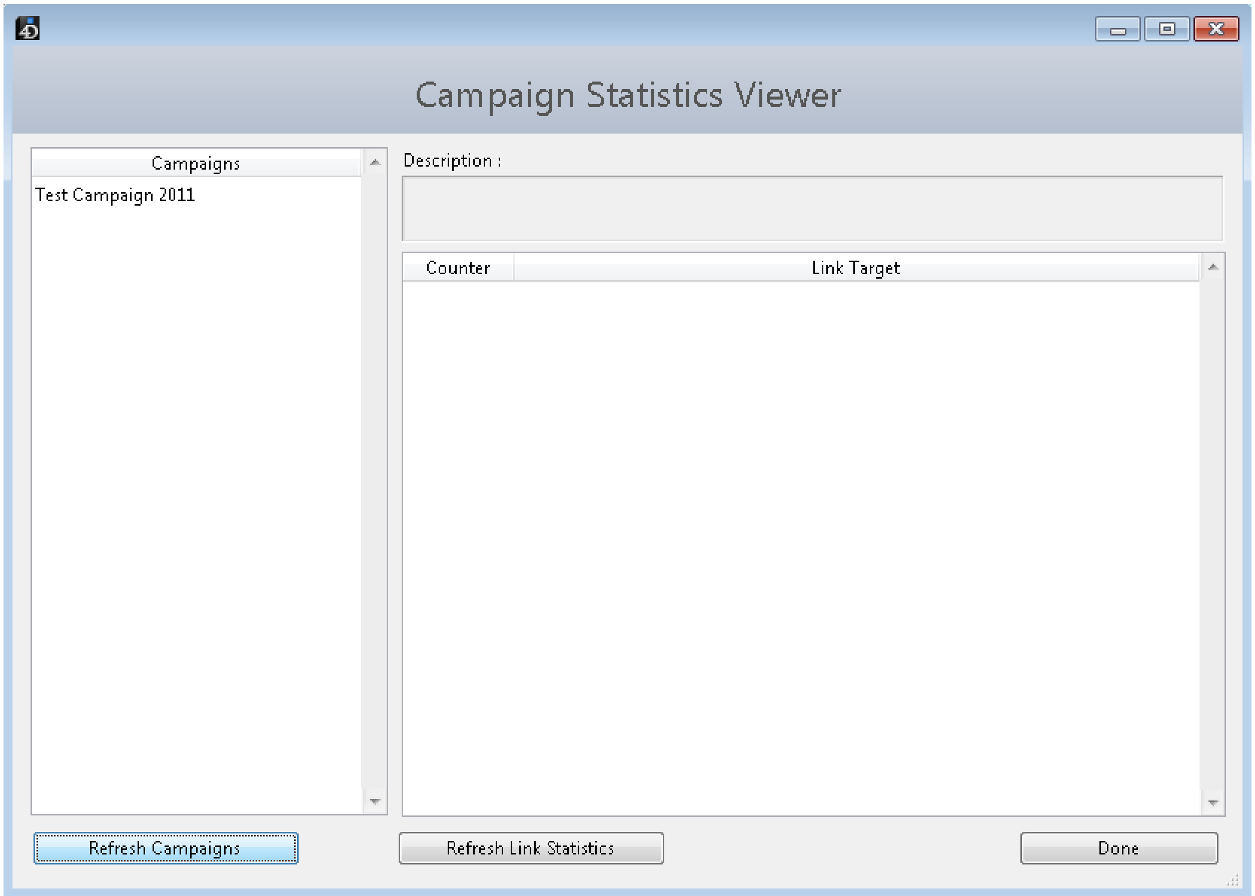
This file is saved with the Technical Note as example.html and converted.html; where the converted.html file has the converted source and the example.html is the original source file.

Viewing Campaign Statistics

This section covers how to view campaign statistics. In the demo select the 'File' menu and 'Campaign Statistics Viewer':

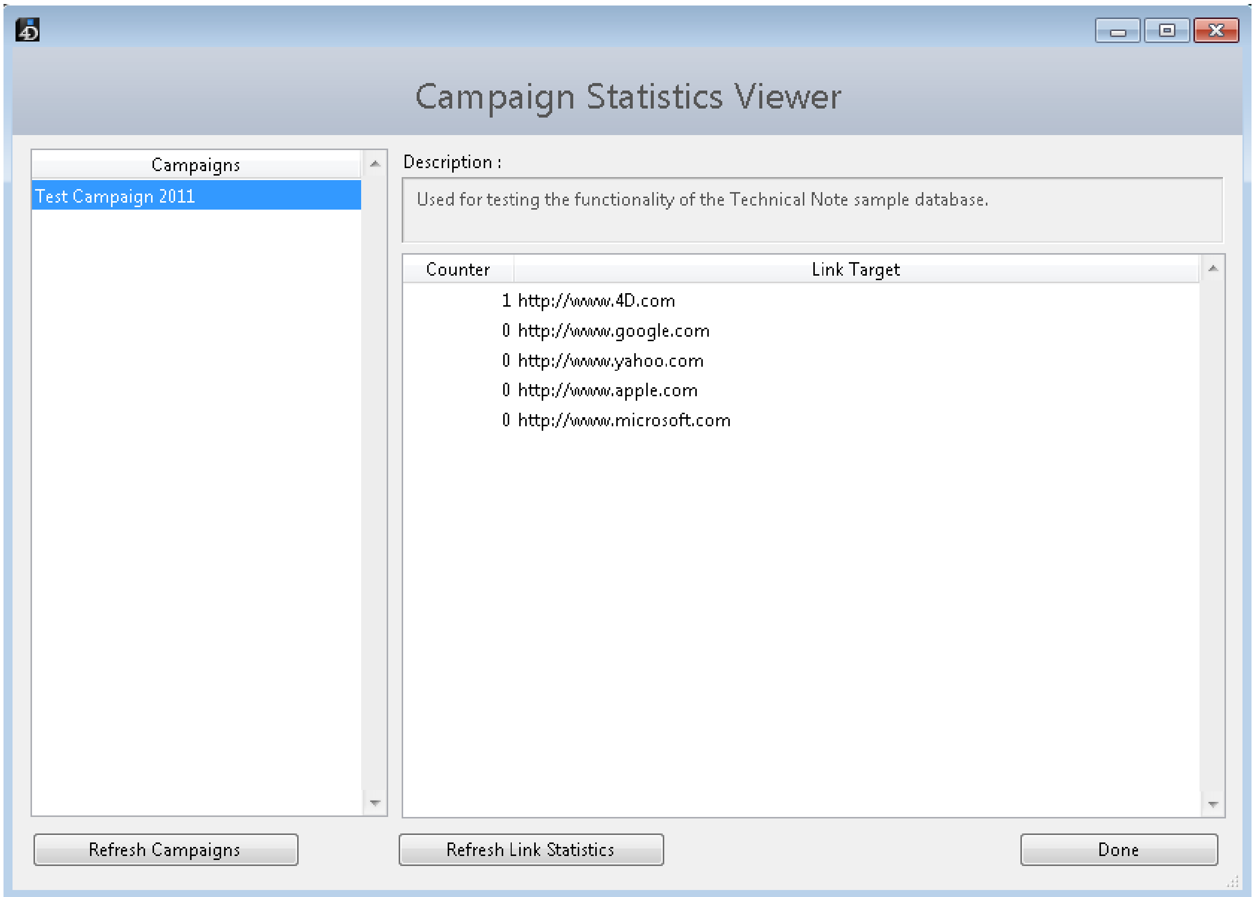

4D
File
Edit
Mode
Help
HTML Source Converter Ctrl+1
Campaign Statistics Viewer Ctrl+2
Quit Ctrl+Q

After clicking on 'Campaign Statistics Viewer' the following form is displayed:



The campaigns are listed on the left. The list can be refreshed by clicking on the 'Refresh Campaigns' button. This is helpful if a campaign is added after this form is open.

Selecting a campaign from the list will load the details of that campaign; the result may look similar to this screenshot:



Test Campaign 2011

Description : Used for testing the functionality of the Technical Note sample database.

Counter	Link Target
1	http://www.4D.com
0	http://www.google.com
0	http://www.yahoo.com
0	http://www.apple.com
0	http://www.microsoft.com

The statistics of the selected campaign can be refreshed by clicking on the 'Refresh Link Statistics' button. This button is helpful when a campaign is active; meaning links are being accessed and counters are being increased.

A row exists for each link that is being tracked, alongside the URL is a counter that represents the number of times that specific link was accessed.

How tracked links are stored in the database

The ‘Convert’ button on the ‘HTML Source Converter’ form calls the ***_convertSource*** project method; here is the code for that project method:

```
C_LONGINT($a)
C_TEXT($destinationURL;$redirectURL;Source;ConvertedSource;campaign_name;fqdn_t)
ARRAY TEXT(Links_at;0)
UTIL_FindAllLinks (Source;->Links_at)
ConvertedSource:=Source
For ($a;1;Size of array(Links_at))
 If (is_not_SRC (->Links_at{$a};->ConvertedSource))
 $destinationURL:=Links_at{$a}
 C_TEXT($thisLink)
 $thisLink:= storeLink($destinationURL;campaign_name;campaign_description)
 $redirectURL:=fqdn_t+"/trackcamp?ref="+$thisLink
 ConvertedSource:=Replace string(ConvertedSource;$destinationURL;$redirectURL;1)
 End if
End for
```

The first thing this project method does is fill an array of all the links found using the ***UTIL_FindAllLinks*** project method with the following code:

```
C_TEXT($1;$myString)
C_POINTER($2)
ARRAY TEXT($2->;0)
ARRAY LONGINT($posFound_a;0)
ARRAY LONGINT($lengthFound_a;0)
C_LONGINT($start)
C_TEXT($stringNew;$pattern)
C_BOOLEAN($found)

$start:=1
$found:=False

$pattern:="(http|https|ftp)"  // Patterns that contain http or https, or ftp
$pattern:=$pattern+"\\://"  // Adding :// to the pattern
$pattern:=$pattern+"[a-zA-Z0-9\\-\\.]+"  //Matches the first part of the domain
$pattern:=$pattern+"\\.[a-zA-Z]{2,4}"  //Match the second part of the domain
$pattern:=$pattern+"(:[a-zA-Z0-9]*)?/?"  //Match the port number and the slash
$pattern:=$pattern+"([a-zA-Z0-9\\-\\._?\\,'/\\+%\\$#\\=~\\:\\&])*" //Reserved
$pattern:=$pattern+"[^\\\"\\.\\,\\)\\(\\s\\']"  //Excluded chars

$myString:=$1

Repeat
 $found:=Match regex($pattern;$myString;$start;$posFound_a;$lengthFound_a)
 $stringNew:=Substring($myString;$posFound_a{0};$lengthFound_a{0})
 If ($found)
  APPEND TO ARRAY($2->;$stringNew)
 End if
 $start:=$posFound_a{0}+$lengthFound_a{0}
Until (Not($found))
```

After obtaining an array of links found, each link is looped upon and checked to see if it is preceded with either URL or SRC using the ***is_not_SRC*** project method with the following code:

```
  // $1 = pointer to link to find
  // $2 = source var to seach
C_BOOLEAN($0)
If (Count parameters=2)
C_POINTER($1;$2)
C_LONGINT($pos)
C_TEXT($sub)
$pos:=Position($1->;$2->)
$sub:=Substring($2->;$pos-6;6)
If ((Position("src";$sub)#0) | (Position("url";$sub)#0))
 $0:=False
Else
 $0:=True
End if
Else
 $0:=False
End if
```

If the link is not preceded with either SRC or URL the conversion takes place using the ***storeLink*** project method and the following code:

```
  // returns redirect ID if record already exists otherwise creates it
  // $1 = URL to store
  // $2 = Campaign name to use for storing URL's
  // $3 = Campaign description

If (Count parameters=3)
 C_TEXT($0;$1;$2;$3)
 C_TEXT($campaignID_t;$urlID_t)

 QUERY([Web_URL_Campaigns];[Web_URL_Campaigns]Campaign_Name=$2)
 If (Not(Records in selection([Web_URL_Campaigns])=1))
   // record not found so create it
  CREATE RECORD([Web_URL_Campaigns])
  [Web_URL_Campaigns]Campaign_Name:=$2
  [Web_URL_Campaigns]Campaign_Description:=$3
  SAVE RECORD([Web_URL_Campaigns])
 End if
 $campaignID_t:=[Web_URL_Campaigns]ID

 QUERY([Web_URL_Tracking];[Web_URL_Tracking]Campaign_ID=$campaignID_t;*)
 QUERY([Web_URL_Tracking];[Web_URL_Tracking]Target_URL=$1)
 If (Not(Records in selection([Web_URL_Tracking])=1))
   // record not found so create it
  CREATE RECORD([Web_URL_Tracking])
  [Web_URL_Tracking]Campaign_ID:=$campaignID_t
  [Web_URL_Tracking]Target_URL:=$1
  SAVE RECORD([Web_URL_Tracking])
 End if

 $urlID_t:=[Web_URL_Tracking]ID
 $0:=$urlID_t

 UNLOAD RECORD([Web_URL_Campaigns])
 UNLOAD RECORD([Web_URL_Tracking])
End if
```

How tracked links are resolved through the web

The tacked link is in the form of a URL like the following:

```
http://webtest.4d.com/trackcamp?ref=9F120303E83B614EAC1A371BE47C2064
```

In the above example the UUID is 9F120303E83B614EAC1A371BE47C2064, which is the ID of the link that is tracked within the 4D database.

The '***On Web Connection***' method of the sample database is altered to check for the entry point of '*trackcamp?ref=*' as depicted in the following code snippet:

```
C_TEXT($1;$2;$3;$4;$5;$6;$url_ID)
If (Position("/trackcamp?ref=";$1)#0)
  $url_ID:=Substring($1;16;32)
  getLink ($url_ID)
End if
```

When a URL comes in matching '*/trackcamp?ref=*' the ***getLink*** project method is called with the tracked link UUID as the parameter.

The ***getLink*** method is executed in the following way:

```
C_TEXT($uid;$1)
If (Count parameters=1)
  $uid:=$1
  QUERY([Web_URL_Tracking];[Web_URL_Tracking]ID=$uid)
  If (Records in selection([Web_URL_Tracking])=1)
    // redirect found
    [Web_URL_Tracking]Counter:=[Web_URL_Tracking]Counter+1
    SAVE RECORD([Web_URL_Tracking])
    SEND HTTP REDIRECT([Web_URL_Tracking]Target_URL)
  Else
    // redirect not found
    // handle it
  End if
End if
```

In the code above, if the UUID of the tracked link is found, a redirection is carried out using ***SEND HTTP REDIRECT***, otherwise there is a section in the code left for the developer to implement some type of error handling (such as redirecting to a pre-defined page).

Conclusion

This Technical Note describes a general approach for tracking links within a marketing campaign. Included with this Technical Note is a sample database to help demonstrate the concepts described. This information should help the 4D developer to track links within their own marketing campaigns.