

# Data Channels

Data channels create a named pipe to run [profiles](#) from user space. One channel per line. Each line is comma delimited and contain channel name, domain, profile, allowed groups.

You use them with [PluginDataChannel](#). Starting in [Tiki6](#), a [payment](#) parameter can be added and thus, the data channel will only be run when the payment is done.

Please see:

<http://profiles.tiki.org/Data+Channels>

A good example is [http://profiles.tiki.org/Project\\_Management](http://profiles.tiki.org/Project_Management)

## 1.1. Usage with trackerfilter/trackerlists

The idea of it is to let you set up a trackerfilter/trackerlist with checkboxes, let the user select the ones they want, and then run a datachannel (profile) on each item.

Example:

Test page

---

### 1.1.1. Tracker list multi select tester

```
{trackerfilter filters="2/d:3/d:4/d" noflipflop="y" trackerId="1" fields="2:3:4:5" other_filters="1" checkbox="0/itemId"
wiki="tracker checkbox test tpl" } --- {DATACHANNEL(channel="test_channel" emptyCache="none")}
itemId,external=input[name='itemId[]']:checked surname,external=input[name^='surname[']
forename,external=input[name^='forename['] {DATACHANNEL}
```

Pretty tracker template: tracker checkbox test tpl

---

### 1.1.2. tracker checkbox test tpl

```
{if $checkbox} {$item.field_values[$checkbox.ix].value|escape} <input type="{ $checkbox.type}"
name="{ $checkbox.name}[]" value="{if $checkbox.ix >
-1} {$item.field_values[$checkbox.ix].value|escape} {else} {$item.itemId} {/if}" />{/if} __ {$f_2} {$f_3} {$f_4}:__ {$f_5}
<input type="hidden" name="surname[{$item.itemId}]" value="{ $f_3}" /> <input type="hidden"
name="forename[{$item.itemId}]" value="{ $f_4}" />
```

Note

---

You need is\_html enabled and HTMLPurify off at the moment (as of August 2010) for the hidden inputs to work in this example. Could be replaced by {HTML} or some other plugins probably.

## 1.2. Example usage with hidden fields and Tracker plugin

I show this example because external fields only seem to work with tracker fields of type text, not drop-downs. Tracker number and field numbers need to be adapted to your Tiki site.

This data channel will show how to create a tracker item and a group and a wiki page which shows the tracker item.

## 1.2.1. Create Tracker 2

### Fields

- Field 5 (Town) is of type "text" (this would also work with the *external* data channel field syntax)
- Field 6 (Country) is of type "Country selector", which looks like a Drop-down
- Field 7 (Continent) is of type "Drop-down"

Gory details of fields extraction

## 1.2.2. Create data channel

Create Data channel in *Admin* → *Admin Home* → *Profiles* → *Advanced*

New\_Town,tiki://local,template\_Add\_New\_Town,Admins

This creates a data channel which only admins can use.

## 1.2.3. The page which calls the data channel

This contains the Tracker plugin, the datachannels plugin and the JQ plugin which binds them together

### Create New Town

```
-=Create New Town=- {tracker trackerId="2" fields="5:6:7" showtitle="n" formtag="n"}
{DATACHANNEL(channel="New_Town" buttonLabel="Create" class="channel" debug="y" )} Town,hidden=
Country,hidden= Continent,hidden= {DATACHANNEL} {JQ()} // On reporte les changements
$("input[name='ins_5']").bind('change', function() { valeur=$(this).val(); // This is the syntax for text fields
$("input[name='Town']").val(valeur); }); $("select[name='ins_6']").bind('change', function() {
valeur=$(this).find("option:selected").val(); // This is the syntax for drop-down fields $("input[name='Country']").val(valeur);
}); $("select[name='ins_7']").bind('change', function() { valeur=$(this).find("option:selected").val(); // This is the syntax for
drop-down fields $("input[name='Continent']").val(valeur); }); {JQ}
```

The JQ plugin will need approval

[View details](#) → [Approve](#)

## 1.2.4. The page which contains the data channel code

### template\_Add\_New\_Town

```
This is the template for the data channel {CODE(caption="YAML" wrap="1" colors="yaml")} mappings: #Rename variable for
permissions (aka groups) varNameTown: $profilerequest:Town$Unnamed Town$ permissions: #Creates a group named after
the Town name, with options varNameTown: autojoin: n description: $profilerequest:Continent$ $ - $profilerequest:Country$
$ #The group description uses other variables objects: - type: group id: varNameTown allow: #Permissions of group
varNameTown - group_view - group_view_members - group_add_member - group_remove_member - group_join objects: - type:
tracker_item # Create the tracker item ref: town_file_id # This is how we reference the tracker item from within the wiki page
data: tracker: 2 status: open values: - [ 5, $profilerequest:Town$Unnamed Town$ ] - [ 6, $profilerequest:Country$ $ ] - [ 7,
$profilerequest:Continent$ $ ] - type: wiki_page # Create a page data: name: Town $profilerequest:Town$Unnamed Town$
content: wikicontent:template_Town_Page # the content of the page is defined it this template mode: create
wiki_authors_style: none {CODE}
```

## 1.3. The page which contains the template for the wiki page which is created with the data channel

This is just to illustrate the use of references.

## template\_Town\_Page

```
-=Town $profilerequest:Town$Unnamed Town$=- Blah blah {trackerlist trackerId="2" itemId="$profileobject:town_file_id$"}  
More text
```

alias

---

[data channel](#) | [datachannel](#) | [DataChannels](#)