

WORK IN PROGRESS**dITo: Global Actions, manually improved**

1	What's it good for?	2
1.1	the DITo initiative: a mutual share of knowhow and experience	2
1.2	the DITo Contact	2
1.3	act as a true professional	2
2	Why do "Global Action" needs improvement?	3
2.1	Leftovers after deleting components	3
2.2	Bugs in using component delete in Global Actions	3
3	A strategy to find component-related data in the repository	3
3.1	A small experimental repository	4
3.2	Some actions to take:	4
3.3	Examine a "select all" export file	4
3.4	An incomplete example of find_formloc.erg	4

Disclaimer: test-package is provided "as is". Use it on your own risk, so run tests with sufficient backups.

1 What's it good for?

1.1 the DITo initiative: a mutual share of knowhow and experience

Instead of waiting for CPWR to provide some solution, **do IT *ourself*** (dITo) invites to share all you know about uniface land with the community.

1.2 the DITo Contact

eMail to **ulrichmerkel@web.de**; see www.uli-merkel.de for more about dITo.

1.3 act as a true professional

If you do not have a separate "sandbox" or at least a separate repository (database) to have a completely isolated experiment environment; but **experiment in the same area as your productive data**:

Be careful not to delete production data if you do more than reading.

Please have in mind that experiments may fail.

BE CAREFUL, BACKUP as often as possible.

2 Why do "Global Action" needs improvement?

Something seems wrong when we want to get rid of test/experimental/backup components in our development environment.

Most of us want to delete anything which is connected with a component including signatures etc. Anything with the component name has to be purged from the repository.

As most of us do NOT model signatures at first and start creating a component from these signatures, but just use the form editor as a start, there is no need to set signatures back to a "modeled" state they never had in their lifecycle.

2.1 Leftovers after deleting components

Source: <http://unifaceinfo.com/forum/uniface9/cleanup-old-signatures/#p6048447>

Hello Everyone,

How do I cleanup the signatures from uniface components that have been removed?

Thanks,

Col Douglass

2.2 Bugs in using component delete in Global Actions

Source: <http://unifaceinfo.com/forum/unifacedevelopment/global-actions-component-delete/#p6048445>

I am doing a tidy up of redundant entities and components.

I tried going in to Go To->Administration->Global Actions, entering the pattern for the components now redundant, and then Edit->Delete

This seems to have deleted all the components, and without any warning messages at that. However, the signatures don't seem to have been set to modeled.

I have run a /cln and a full recompile (which gave me the message that there was no implementation for these signatures).

The signature cleaner download can't find the signatures, and I have just spent half an hour finding them all manually and deleting them.

Is the Delete function in the Global Actions safe to use on components? Have I left corrupt data in my IDF somewhere?

Iain

2.3 So we need our own defined way of cleaning the repository

And because we can NOT trust the documentation of the repository which is delivered, we have to find out on our own which records have to be deleted.

3 A strategy to find component-related data in the repository

3.1 A small experimental repository

If we want to know the fields where component names are held in the uniface repository, all we need is a hand ful of forms including all possibilities what we can do with forms, and let IDF machinery do it's job (opening "Component Integration").

It's best to start this with an empty repository, do our specifications and export using the "Select All" button to file select_all.xml at the end.

3.2 Some actions to take:

Create form "DITOFIND01"

- component variables
- operation with parameters
- entity with fields
- compile

Create form "DITOFIND02"

- HEADER frame with dummy field
- operation with parameters
- modify DITOFIND01 that this operation is called with activate
- compile both forms

Open "Component Integration", wait for synchronisation

3.3 Examine a "select all" export file

We export using the "Select All" button to file select_all.xml to get our base data.

What we want to know is the name and entity for any occurencs which includes the name of the components we created, all starting ">DITOFI" here and the first line of the description block which starts "<DSC ".

A simple FIND command which is available under Windows, will do the job for us:

```
find /n "<DSC " select_all.xml > find_formloc.entity
find /n ">DITOFI" select_all.xml > find_formloc.forms
find /n "<DSC " select_all.xml > find_formloc.tmp
find /n ">DITOFI" select_all.xml >> find_formloc.tmp
sort find_formloc.tmp /O find_formloc.erg
```

3.4 An incomplete example of find_formloc.erg

After a little manual editing (because the line numbers are sorted as strings), we get:

```
[6]<DSC name="UFORM" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[172]<DAT name="ULABEL">DITOFIND01</DAT>
[319]<DAT name="ULABEL">DITOFIND02</DAT>
[2160]<DSC name="UXGROUF" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[2305]<DAT name="UFORM">DITOFIND01</DAT>
[2314]<DAT name="UFORM">DITOFIND01</DAT>
[2342]<DAT name="UFORM">DITOFIND02</DAT>
```

```
[2745]<DSC name="UXFIELD" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[2871]<DAT name="UFORM">DITOFIND01</DAT>
[2880]<DAT name="UFORM">DITOFIND01</DAT>
[2889]<DAT name="UFORM">DITOFIND01</DAT>
[2898]<DAT name="UFORM">DITOFIND01</DAT>
[2907]<DAT name="UFORM">DITOFIND01</DAT>
[3350]<DAT name="UFORM">DITOFIND02</DAT>
[3362]<DSC name="UXREGS" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[3409]<DSC name="UCSCH" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[3454]<DSC name="UCTABLE" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[3502]<DSC name="UCGROUP" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[4803]<DSC name="UCFIELD" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[5766]<DSC name="UCKEY" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[5822]<DSC name="UCRELSH" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[5883]<DSC name="UAPPL" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[5975]<DSC name="ULIBR" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[5995]<DSC name="USOURCE" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6276]<DSC name="USUBS" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6297]<DSC name="USSPEC" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6344]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6348]<DAT name="UDEFIMPLNAM">DITOFIND01</DAT>
[6355]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6358]<DAT name="UDEFIMPLNAM">DITOFIND02</DAT>
[6365]<DSC name="USOPER" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6396]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6403]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6410]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6417]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6424]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6431]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6438]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6445]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6452]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6459]<DSC name="USPARM" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6496]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6506]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6516]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6526]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6536]<DSC name="USIMPL" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6571]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6572]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6580]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6581]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6589]<DSC name="USIOPER" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6628]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6629]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6638]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6639]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6648]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6649]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6658]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6659]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6668]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6669]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6678]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6679]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6688]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6689]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6698]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6699]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6708]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6709]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6718]<DSC name="USIPARM" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6759]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6760]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6769]<DAT name="USSPECNAM">DITOFIND01</DAT>
[6770]<DAT name="UIMPLNAM">DITOFIND01</DAT>
[6779]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6780]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6789]<DAT name="USSPECNAM">DITOFIND02</DAT>
[6790]<DAT name="UIMPLNAM">DITOFIND02</DAT>
[6799]<DSC name="USLINK" model="DICT" system="S" pseudo ="73" level="1" nouupdate="0"
[6820]<DAT name="USSPECNAMCAL">DITOFIND01</DAT>
[6821]<DAT name="USSPECNAMDES">DITOFIND02</DAT>
```

3.5 A concept for the "delete component complete" component

All we have to do is to get the fields which holds our formnames, paint the entities on a form and fill the formname fields with our profile. Sometimes, like in USLINK, formnames may appear in the role of caller as well as of destination, so we need two separate retrieves.

Because the metadictionary does not hold all the relationships necessary, it's the best to paint all entities next to each other and delete them one by one (still having the "delete the many before you delete the one" rule in mind).