



Uniface 10

Arjen van Vliet

Solution Consultant

Monday, December 5, 2016

UNIFACE

Advanced Development Technology

Imagine always having driven
traditional cars...
...with traditional dashboards...



Then one day a new
revolutionary car arrives...
...with a new revolutionary
dashboard...



A conceptual image showing a road that splits into two paths. The path on the left is labeled 'Traditional' and the path on the right is labeled 'New'. The signs are green with white text. The background is a blue sky with clouds and a green field.

Traditional

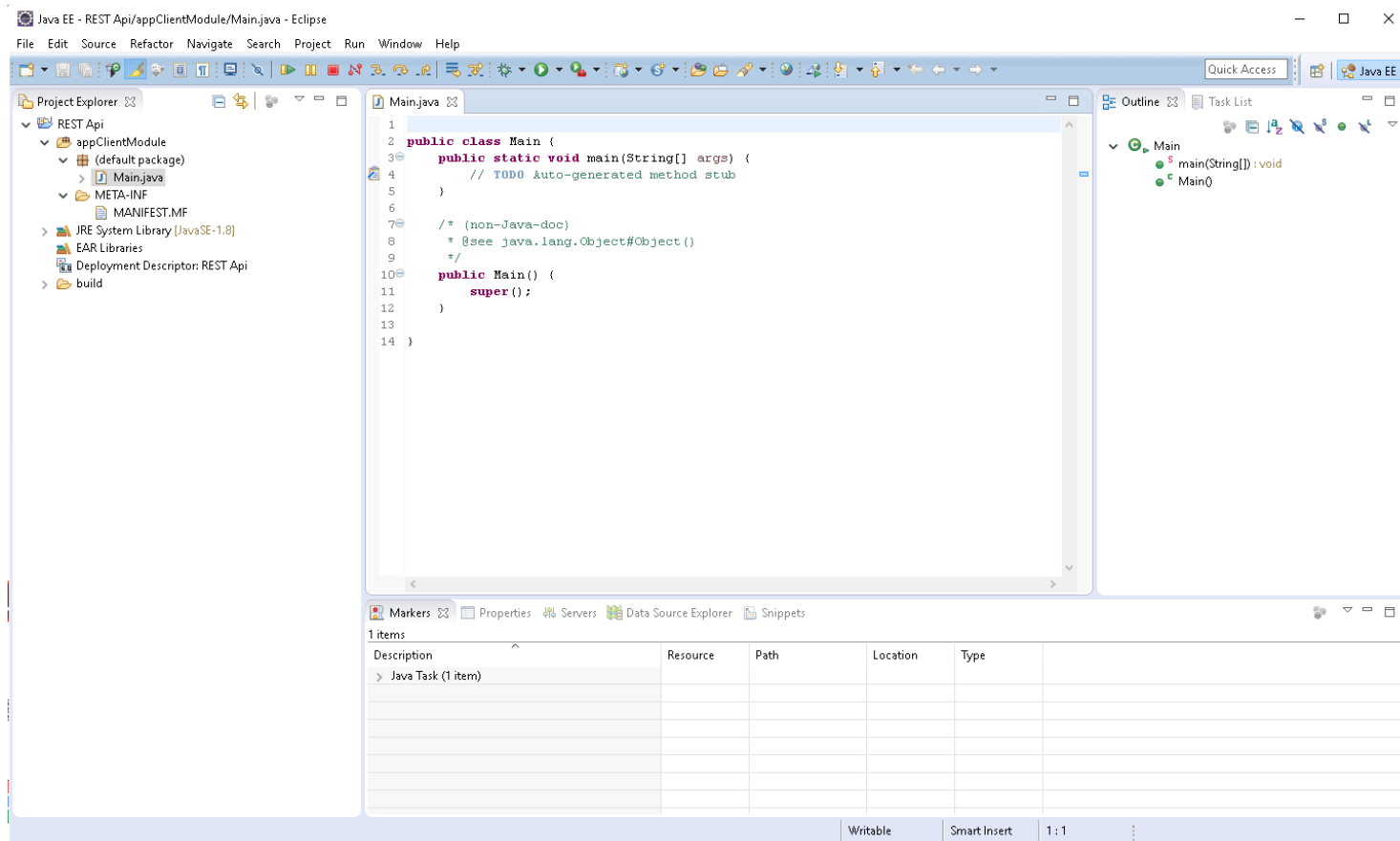
New



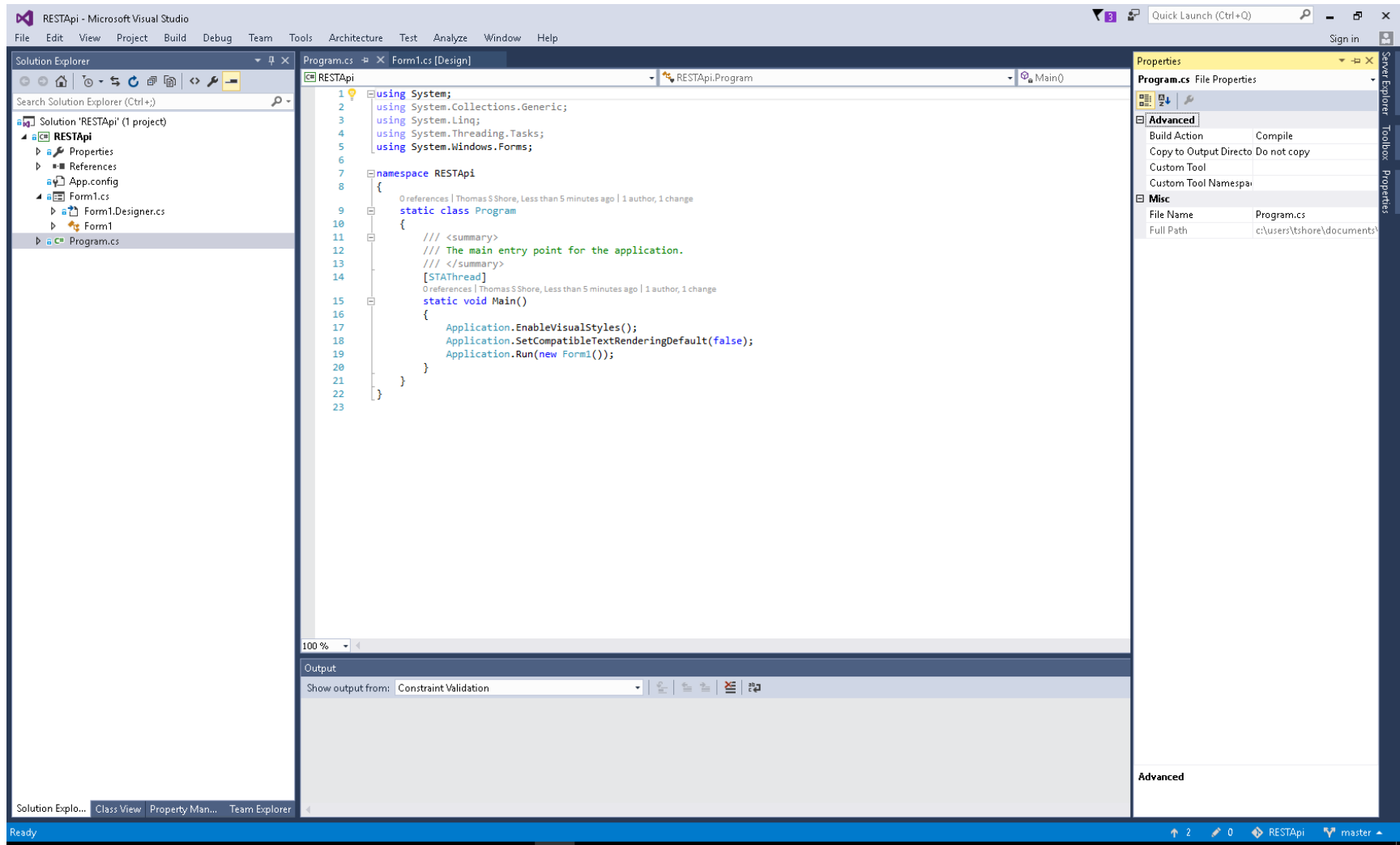
UNIFACE 10

Enterprise Edition

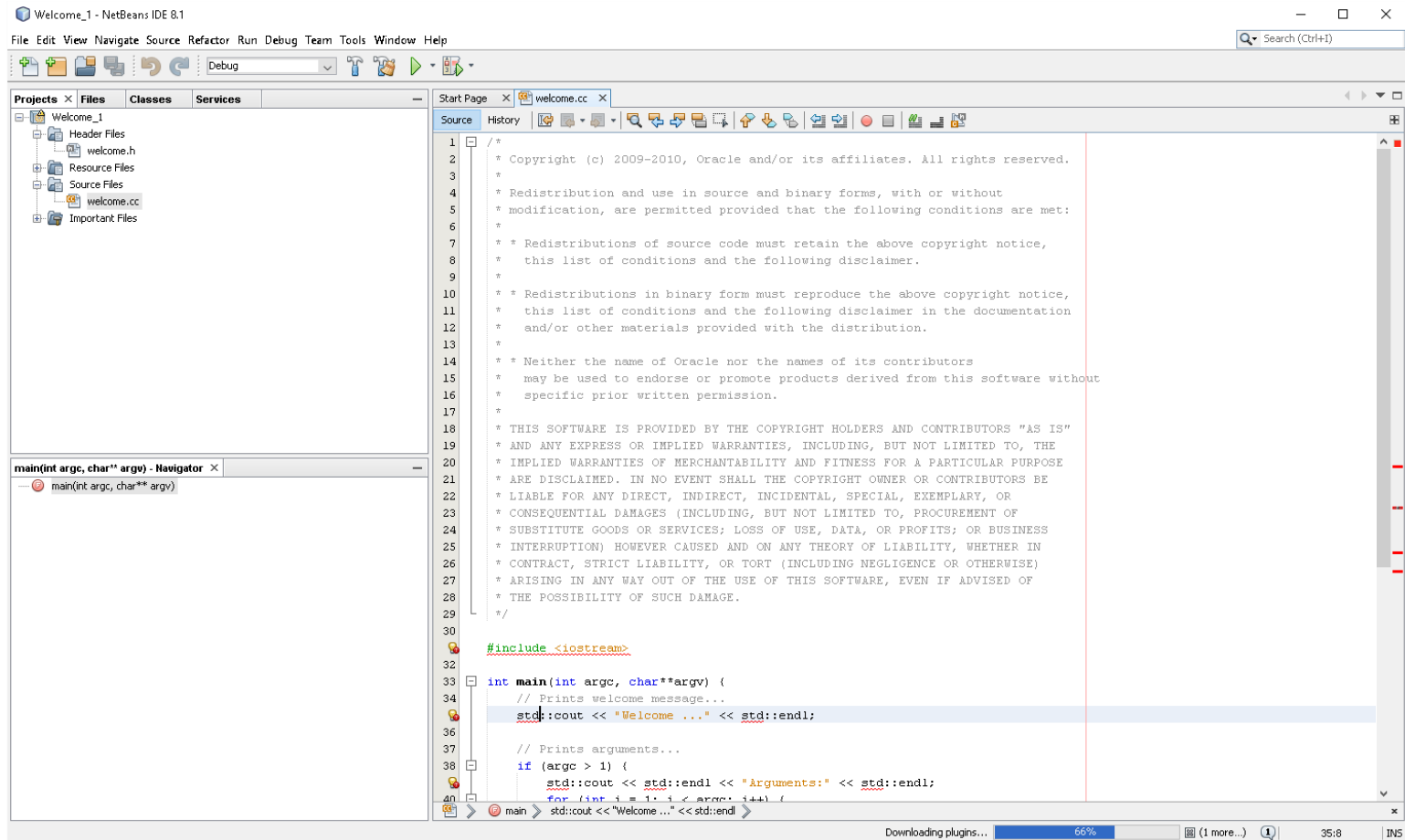
Uniface 10



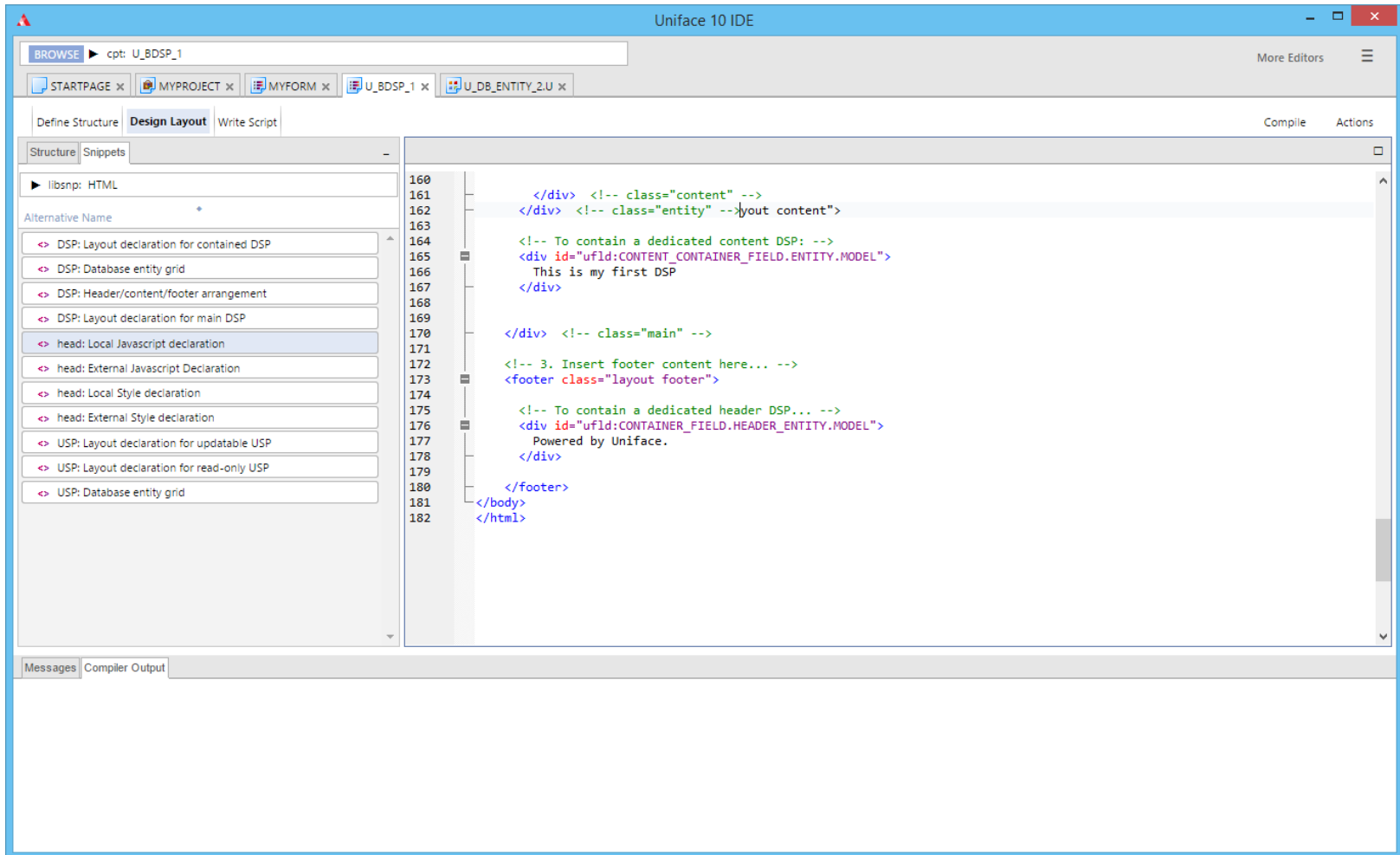
Uniface 10



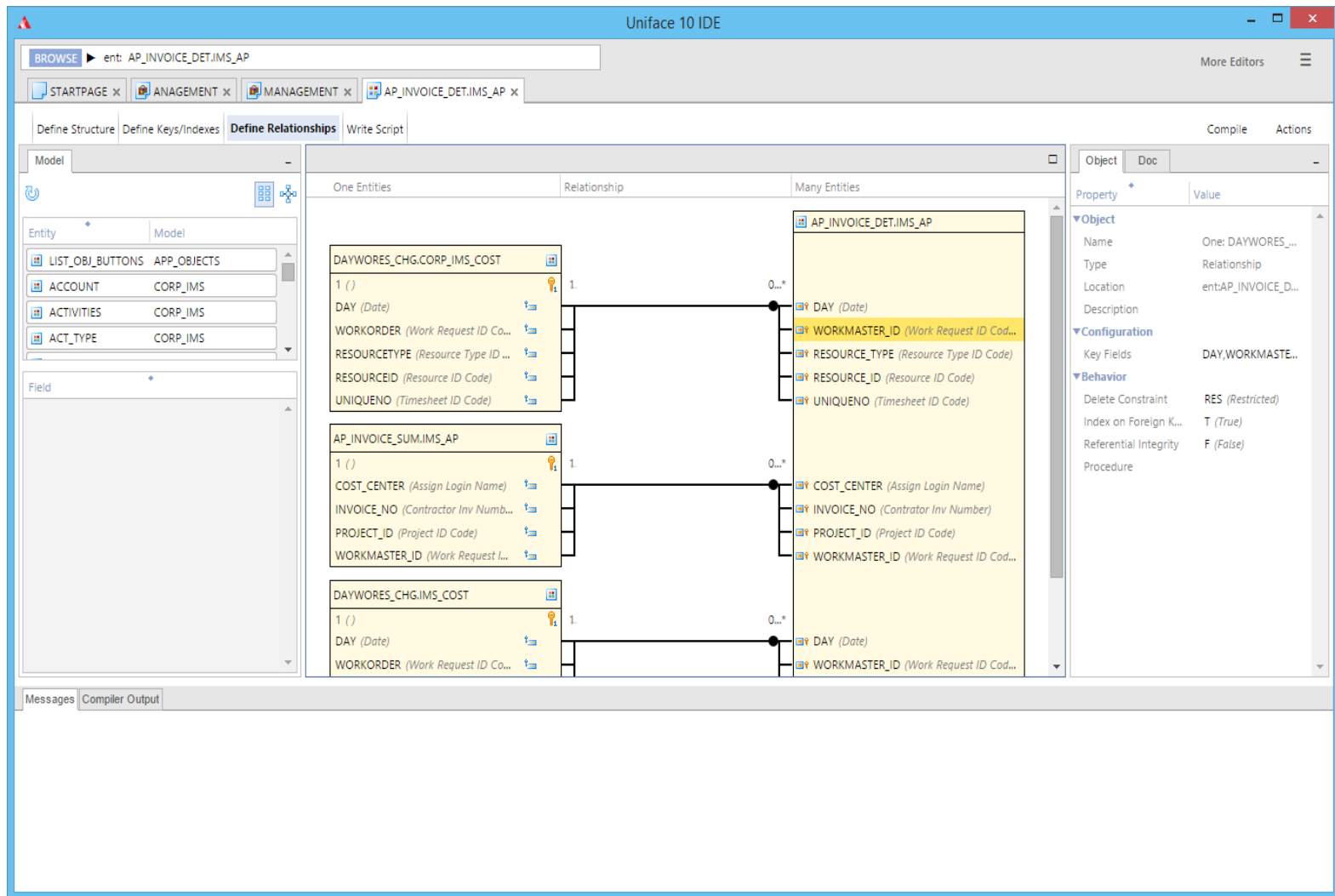
Uniface 10



Uniface 10



Uniface 10



What's the problem?

- We gave you a clear concise execution model that:
 1. The first person understood
 2. The next person didn't
 3. So you got Unibol

Unibol

```
selectdb (count(a_num)) from "bills_index" u_where (sess_number.bills_index = sess_number.dummy) & (chamber.bills_index = chamber.dummy) & (bill_number.bills_index = bill_number.dummy) & (a_num.bills_index > $$temp_a_num) & (a_or_b.bills_index = $$temp_a_or_b) to ($90)

while ($90 > 0)
  selectdb (min(a_num)) from "bills_index" u_where (sess_number.bills_index = sess_number.dummy) & (chamber.bills_index = chamber.dummy) & (bill_number.bills_index = bill_number.dummy) & (a_num.bills_index > $$temp_a_num) & (a_or_b.bills_index = $$temp_a_or_b) to ($$temp_a_num)
  $$temp_a_text = "FIND ONE"
  run "bh_find_a_level"
  message "INDEXED UNDER %%%$temp_a_text"
  length $$temp_a_text
  $51 = $result
  $53 = 79
  $$temp_text = $$temp_a_text
  call checkE
  inversion.dummy = "%%inversion.dummy%%$temp_text%^"
  $99 = $99 + 1
  $$temp_b_num = 0
  $91 = 0
  selectdb (count(b_num)) from "bills_index" u_where (sess_number.bills_index = sess_number.dummy) & (chamber.bills_index = chamber.dummy) & (bill_number.bills_index = bill_number.dummy) & (a_num.bills_index = $$temp_a_num) & (b_num.bills_index > $$temp_b_num) & (a_or_b.bills_index = $$temp_a_or_b) to ($91)
  while ($91 > 0)
    selectdb (min(b_num)) from "bills_index" u_where (sess_number.bills_index = sess_number.dummy) & (chamber.bills_index = chamber.dummy) & (bill_number.bills_index = bill_number.dummy) & (a_num.bills_index = $$temp_a_num) & (b_num.bills_index > $$temp_b_num) & (a_or_b.bills_index = $$temp_a_or_b) to ($$temp_b_num)
    $$temp_b_text = "FIND ONE"
    run "bh_fnd_b_level"
    inversion.dummy = "%%inversion.dummy  "
    length $$temp_b_text
    $51 = $result
    $53 = 75
    $$temp_text = $$temp_b_text
    call checkE
    inversion.dummy = "%%inversion.dummy%%$temp_text%^"
```


[illegible]

UNIFACE
User Events

Why should you fix this?

- Efficiency
- Consistency
- Reduce the “that’s different”

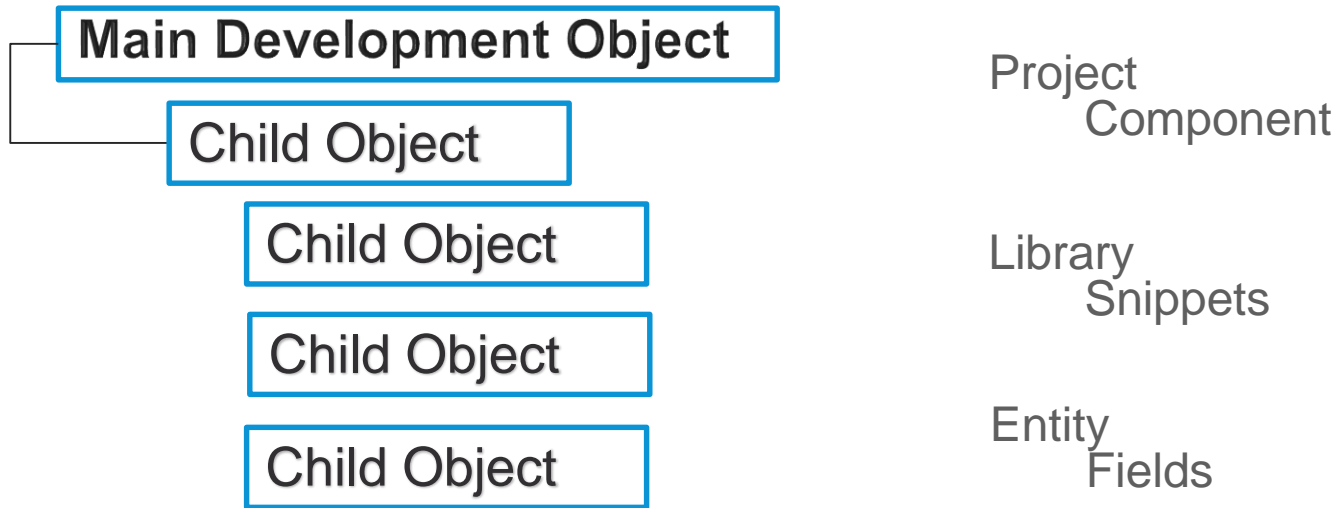
So what have we done?

- Normalize the development tooling
 - Navigation
 - Context
 - Single click
 - Compact design

Key Concepts

- Development objects
- Snippets
- Templates and palettes
- Containers (vs triggers)
- Bulk activities
- Drag and drop
- Properties inspector
- Inheritance of ProcScript
- Modelled components and properties.

Uniface 10 – Development Objects

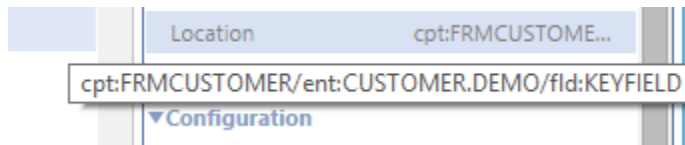


Whole unit can be exported

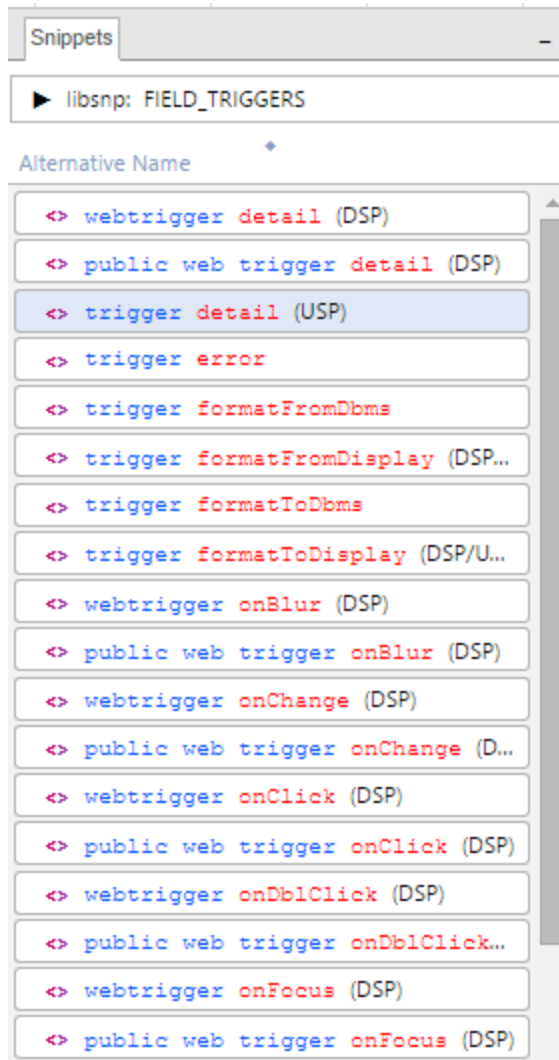
Uniface 10 – Development Object URLs

Each development object, property, and script module has a unique URL that can be used for navigation purposes.

URLSegment{/URLSegment}N



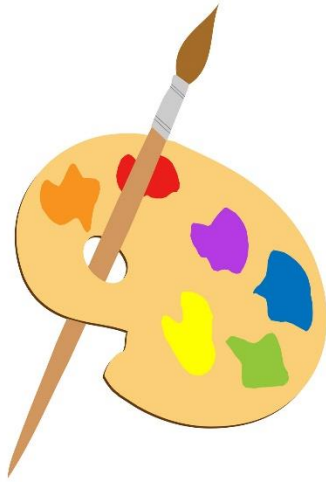
Uniface 10 - Snippets



A collection of named objects organized into libraries.

All available in:
uniface\misc\utemplates.xml

Uniface 10 — Palettes / Templates








Palette

- Template 1
- Template 2

ALLOW_BROWSE_TEMPLATES=1


Palettes

prj: DEMO			
Name	Description	Date Modified	Window
 BASIC_TEMPLATES	Palette of templates for main development obj...	02-aug-16 14:36:55	N/A
 COMPANYSTANDARDS	Company Standards 9/2016	29-aug-16 15:28:38	N/A
 DEMO		29-aug-16 16:42:17	1
 UTEMPLATES	Uniface default templates and palettes	02-aug-16 15:02:38	N/A
 U_PROJECT	New Project	30-apr-15 16:28:36	N/A

Templates | Model | Objects

▶ prj: COMPANYSTANDARDS

Alternative Name

 Modeled DB entity with Status Fields

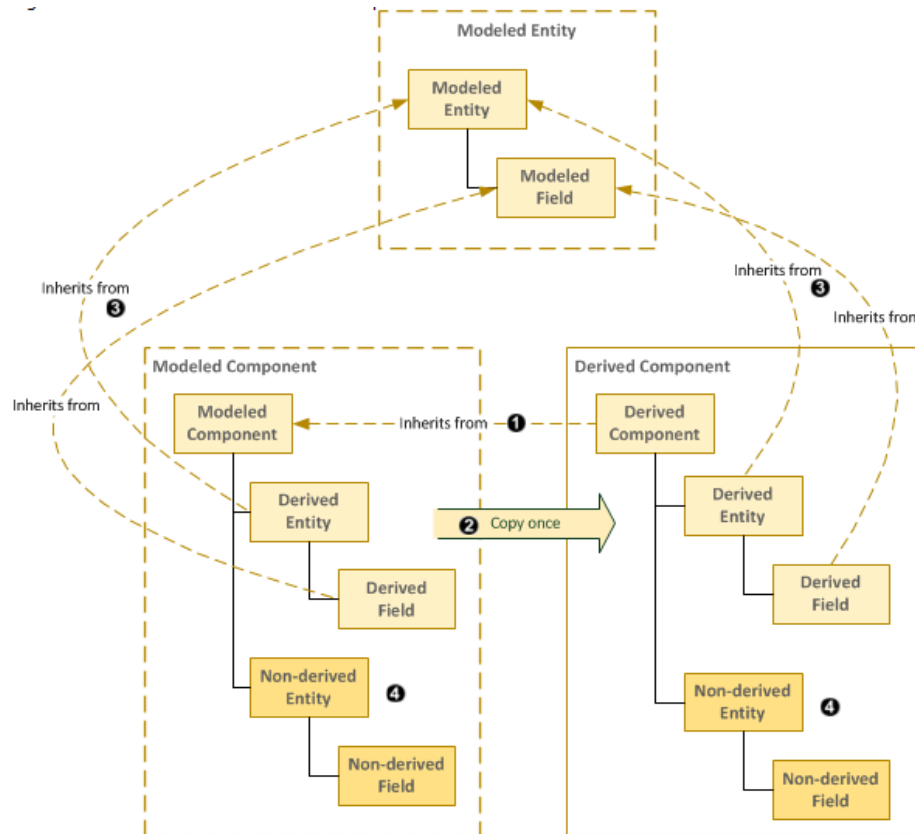
Palettes

Name	Inherits	Type	Description	Database Beha...	Data Access	Database Path	Database Interf...
MODENTSTATUS.U	F (False)	Entity	Basic database...	Y (In Database)	D (Via DBMS p...	DEF (DEFAULT)	

Name	Inherits	Type	Description	Is External	Data Type	Database Interface
KEYFIELD	F (False)	Field	Key field	T (True)	S (String)	C40
FIELD	F (False)	Field	In database field	T (True)	S (String)	C40
CREATEDON	F (False)	Field	Created On	T (True)	D (Date)	D
CREATEDBY	F (False)	Field	Created By	T (True)	S (String)	C40
UPDATEDON	F (False)	Field	Updated On	T (True)	D (Date)	D
UPDATEDBY	F (False)	Field	Updated By	T (True)	S (String)	C40

Uniface 10 - Templates

- Actually Modeled components



Containers



Container Scoping

```
1  Declarations
2  ; Your component defines here (optional)...
3
4  ; Your component VARIABLES block here (optional)...
5  ; All declarations will be available to objects in the component
6  ; including entities and fields
7
8  variables
9  string availableToEveryone
10 endvariables
11
12
13 Script
14 operation exec
15 public web
16 ; Your SCOPE block here (optional)...
17 ; Your VARIABLES block here (optional)...
18
19 ; Your implementation here...
20
21 ; Return execution status:
22 ; return 0 ; return >=0 typically indicates a succesful execution
23 ; return -1 ; return <0 typically indicates an error during execution
24 return 0
25
26 end
```

Container Scoping

```
1  ┌─ Declarations
2  │   ; These variables are available to the entity and it's fields
3  │   ┌─ variables
4  │   │   string ONLYforthisEntity
5  │   └─ endvariables
1  └─ Collection Script
1  └─ Occurrence Script
```

Collection Script — process a collection of the entity occurrences.

Occurrence Script container—used to define ProcScript modules that process each occurrence.

Properties Inspector

Object	
Property	Value
▼ Object	
Name	KEYFIELD.CUSTOM...
Type	Label
Location	cpt:FRMCUSTOME...
Alternative Name	KEYFIELD.CUSTOM...
▼ Configuration	
Inherits	T (True)
Inherits From	ent:CUSTOMER.DE...

Object	
Property	Value
▼ Object	
Name	DIFFER
Type	Label
Location	DIFFER
Alternative Name	DIFFER
▼ Configuration	
Inherits	T (True)
Inherits From	DIFFER

KEYFIELD.CUSTOMER

T (True)

Label

N/A

N/A

N/A

NAME.CUSTOMER

T (True)

Label

N/A

N/A

N/A

X-Position 1
Y-Position 0
X-size 8
Y-size 1
Hor. Alignment L (Left)
Vert. Alignment T (Top)
Font
Transparent (As defined in INI-f...
Index Color -1 (Follows system ...
Repeat per Occ. N (Do not repeat)

X-Position DIFFER
Y-Position 0
X-size DIFFER
Y-size 1
Hor. Alignment L (Left)
Vert. Alignment T (Top)
Font
Transparent (As defined in INI-f...
Index Color -1 (Follows system ...
Repeat per Occ. N (Do not repeat)

Proc Inheritance

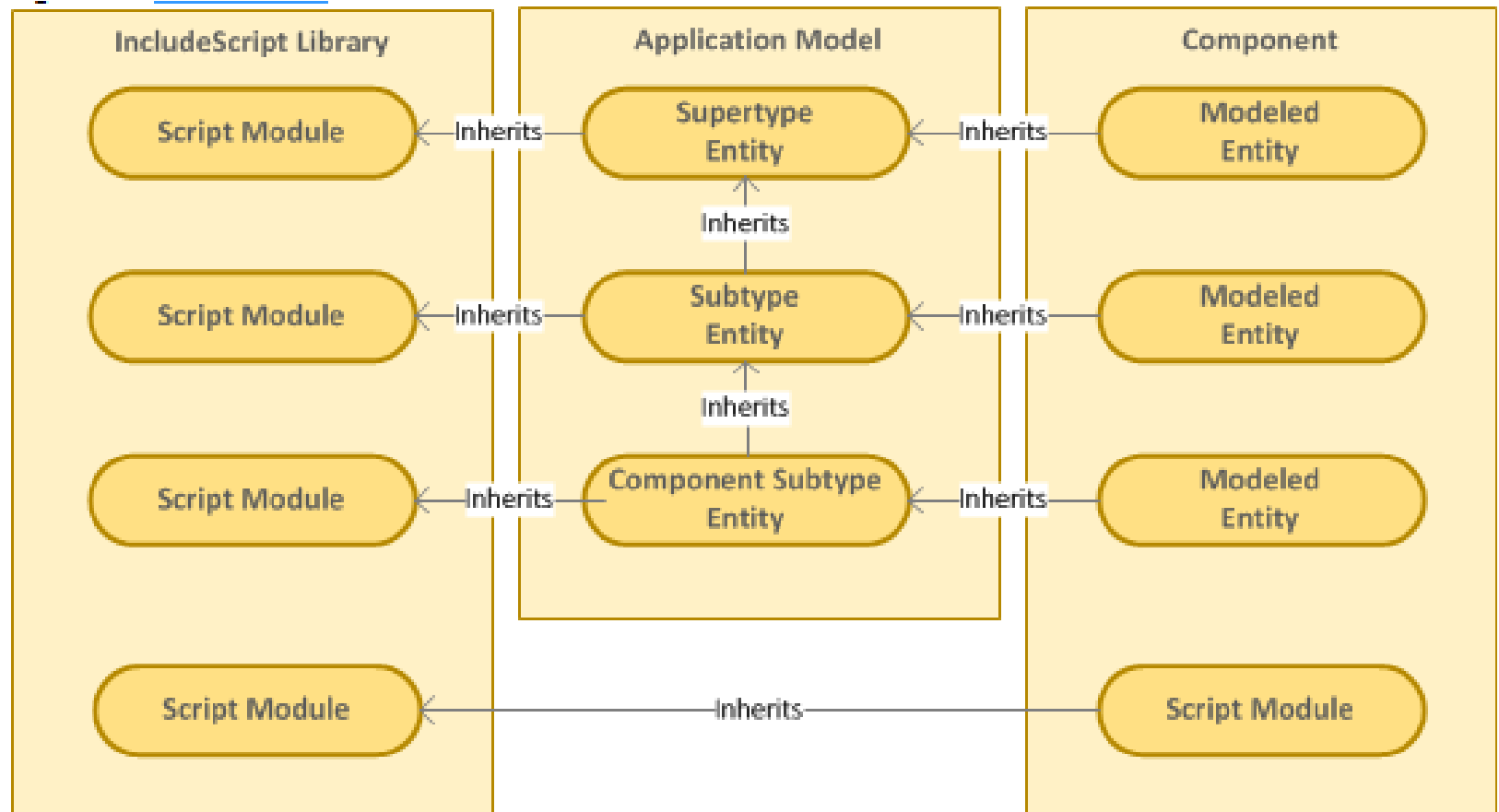
At Compilation time:

All inherited and locally-defined script modules. For a duplicate the last one is compiled into the component. (Modules defined in the component override those defined in modeled entities.)

All constants declared in the component and its sub-objects. The last used will be the one defined for the last field.

All ProcScript entry modules. Treated as if they are all defined at the component level no matter where they are defined.













Object Inheritance



Compiled Modules Inspector

Modules		
I	Type	Name
▶	occTrigger	read
▶	occTrigger	lock
▶	occTrigger	validatekey
▶	occTrigger	write
▶	occTrigger	delete

Compiled Modules Inspector

Icon	Meaning
	Entry is defined locally.
	Entry is inherited.
	Entry replaces an inherited entry.
	Entry replaces an entry defined in another <code>object</code> .
	Operation is defined locally.
	Operation is inherited.
	Operation replaces an inherited operation.
	Operation replaces an operation defined elsewhere, for example in an IncludeScript.
	Trigger is defined locally.
	Trigger is inherited.
	Trigger replaces an inherited trigger.
	Trigger replaces a trigger defined elsewhere, for example in an IncludeScript.

Repository Changes

- Model name is no longer. It simply defines namespace.
- ULANA.*, USYSANA.* - Used for classic deployment
- Type Library definitions
- Signature keywords
- Component integration workspace

Repository Changes

Documentation fields are now longer

Code containers live in one field

UDESCR is now W64

Thank You!

Q&A

UNIFACE
User Events