jPALIO SA

Agile Technology. Peace of Mind.

Documentation **jPALIO 7.4**

Table of Contents

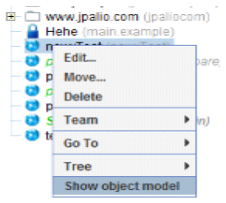
1. Syntax Analysis

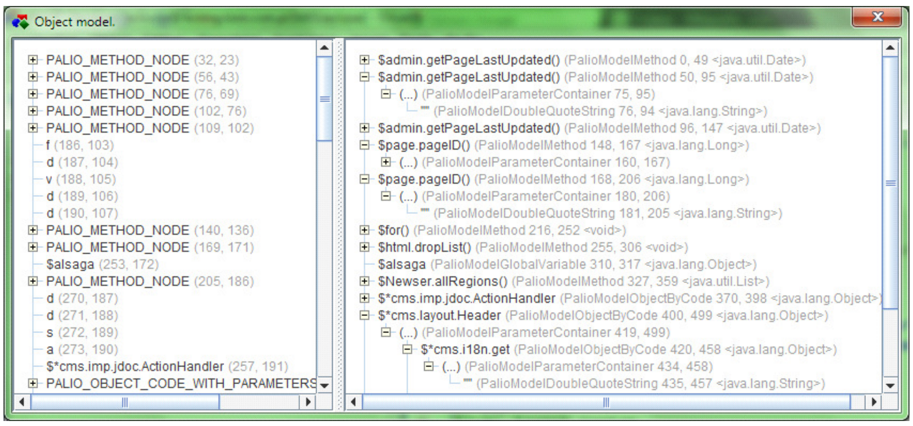
Since version jPALIO 7.4 jDESIGNER has built-in compiler at the client side. It allows to detect simple errors in code and warn programmers about it directly in code.

Compilation process and highlighting errors start always after opening object and after each change in code. At now designer has support objects of type jPalio.

1.1. Model of object's content

After generation the model by compiler there is a possibility to show model in popup window. To do this right click on the object in the editor tree and choose Show object model.

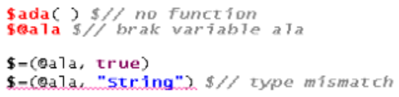




```
⊞ PALIO_METHOD_NODE (32, 23)
⊞ PALIO_METHOD_NODE (56, 43)
⊞ PALIO_METHOD_NODE (76, 69)
⊞ PALIO_METHOD_NODE (102, 76)
⊞ PALIO_METHOD_NODE (109, 102)
  f (186, 103)
  d (187, 104)
  v (188, 105)
  d (189, 106)
  d (190, 107)
⊞ PALIO_METHOD_NODE (140, 136)
⊞ PALIO_METHOD_NODE (169, 171)
  $alsaga (253, 172)
⊞ PALIO_METHOD_NODE (205, 186)
  d (270, 187)
  d (271, 188)
  s (272, 189)
  a (273, 190)
  $*cms.imp.jdoc.ActionHandler (257, 191)
⊞ PALIO_OBJECT_CODE_WITH_PARAMETERS
```

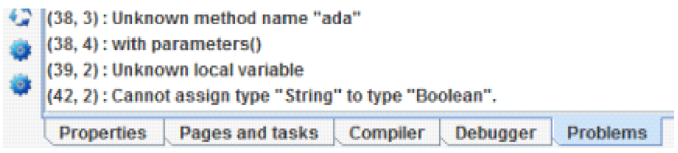
```
⊞ $admin.getPageLastUpdated() (PalioModelMethod 0, 49 <java.util.Date>)
⊟ $admin.getPageLastUpdated() (PalioModelMethod 50, 95 <java.util.Date>)
   ⊟ (...) (PalioModelParameterContainer 75, 95)
      └ "" (PalioModelDoubleQuoteString 76, 94 <java.lang.String>)
⊞ $admin.getPageLastUpdated() (PalioModelMethod 96, 147 <java.util.Date>)
⊟ $page.pageID() (PalioModelMethod 148, 167 <java.lang.Long>)
   ⊞ (...) (PalioModelParameterContainer 160, 167)
⊟ $page.pageID() (PalioModelMethod 168, 206 <java.lang.Long>)
   ⊟ (...) (PalioModelParameterContainer 180, 206)
      └ "" (PalioModelDoubleQuoteString 181, 205 <java.lang.String>)
⊞ $for() (PalioModelMethod 216, 252 <void>)
⊞ $html.dropList() (PalioModelMethod 255, 306 <void>)
  $alsaga (PalioModelGlobalVariable 310, 317 <java.lang.Object>)
⊞ $Newser.allRegions() (PalioModelMethod 327, 359 <java.util.List>)
⊞ $*cms.imp.jdoc.ActionHandler (PalioModelObjectByCode 370, 398 <java.lang.Object>)
⊟ $*cms.layout.Header (PalioModelObjectByCode 400, 499 <java.lang.Object>)
   ⊟ (...) (PalioModelParameterContainer 419, 499)
      ⊟ $*cms.i18n.get (PalioModelObjectByCode 420, 458 <java.lang.Object>)
         ⊟ (...) (PalioModelParameterContainer 434, 458)
            └ "" (PalioModelDoubleQuoteString 435, 457 <java.lang.String>)
```

Errors detected by the compiler are shown in the code by changing the style of the text or highlight text in the editor.



```
$ada( ) $// no function
$@ala $// brak variable ala

$-(@ala, true)
$-(@ala, "string") $// type mismatch
```

Hovering your mouse on an error in the code causes the appearance of the window displays the contents of the error. In addition, the tab below the editor problems is displaying a list of all errors together with the position where the error occurs. Clicking the mouse on the bug will automatically move to the caret where it occurs.



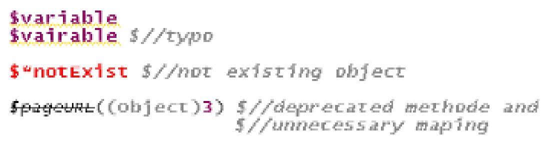
```
(38, 3) : Unknown method name "ada"
(38, 4) : with parameters()
(39, 2) : Unknown local variable
(42, 2) : Cannot assign type "String" to type "Boolean".
```

1.2. Inspections

After creating the object model there are made a number of code inspections. They are designed to detect more sophisticated bugs in the code.

Examples of inspections are:

- Search for non-existent jPALIO elements (objects, pages, privileges, etc.)
- Find typos in a global variables
- Unnecessary mapping parameters
- Select the deprecated methods

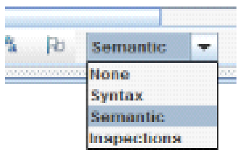


```
$variable
$vairable $//typo
$*notExist $//not existing object
$pageURL((object)3) $//deprecated methode and
                    $//unnecessary maping
```

1.3. Disabling compilation and inspection

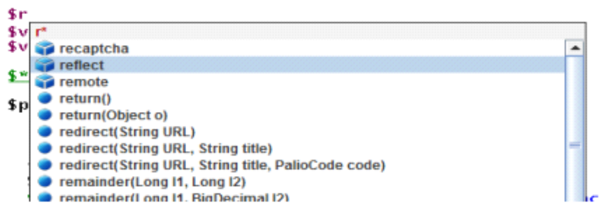
Performing an inspection is labor intensive for the CPU and can be disabled. Below the editor is a combobox in which you can choose one of the four options described below:

- None - there is no syntax highlighting in the editor
- Syntax – there is only a syntax analyze and text elements are colored only from information contained in syntax
- Compilation – syntax parsing is performed, and compilation is made. Coloring is basis on the model and syntax, together with an indication of fundamental errors
- Inspections – syntax parsing is done, the compilation is made, and additional inspections are done. On the basis of all such information is colored text, and selection errors.



1.4. Code complition feature

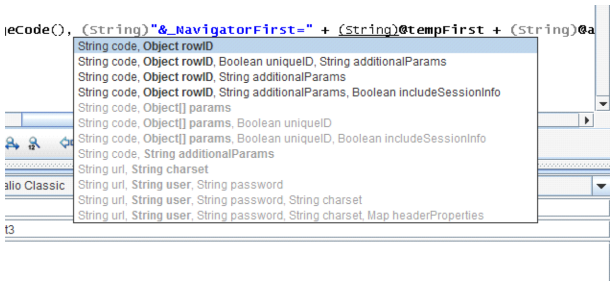
Now there are code complition feature after clicking Ctrl+Space. The editor can prompt jPalio function or object code.



```
$r
$v r
$v recaptcha
   reflect
$  remote
$p return()
   return(Object o)
   redirect(String URL)
   redirect(String URL, String title)
   redirect(String URL, String title, PalioCode code)
   remainder(Long l1, Long l2)
   remainder(Long l1, BigDecimal l2)
```

1.5. Parameter prompting

The editor is also available prompting the function parameters. When you place inside the parentheses of the method of carriage and click Ctrl + Shift + Space is prompting parameters.



```
eCode(), (String)"&_NavigatorFirst=" + (String)@tempFirst + (String)@a
String code, Object rowID
String code, Object rowID, Boolean uniqueID, String additionalParams
String code, Object rowID, String additionalParams
String code, Object rowID, String additionalParams, Boolean includeSessionInfo
String code, Object[] params
String code, Object[] params, Boolean uniqueID
String code, Object[] params, Boolean uniqueID, Boolean includeSessionInfo
String code, String additionalParams
String url, String charset
String url, String user, String password
String url, String user, String password, String charset
String url, String user, String password, String charset, Map headerProperties
```

2. Versioning

Versioning is a mechanism that allows the database to store the historical versions of objects for later tracking the change history of the object. Each user can work at any time preview archived content objects and restore their contents or fragments.

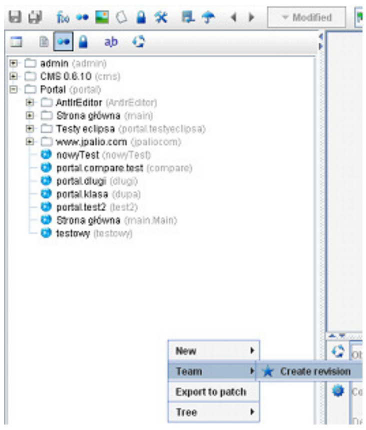
We have two types of archival objects:

- Automatic archiving – it is created automatically during working with object every 15 minutes from the last backup and when another user try to save this object. It is also created before deleting the object.
- Revision - created at the request of the user, it is based on window for creating a revisions, with comments and unique number. It fulfill a revisions from SVN or other version control systems.

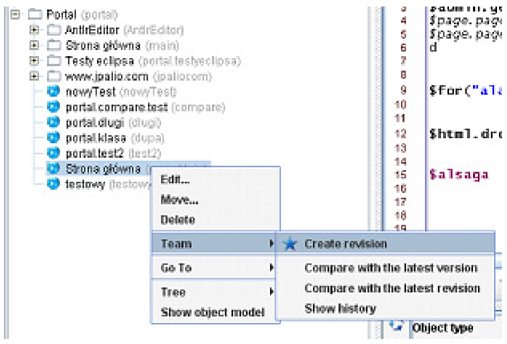
2.1. Creation of revision

In order to call window which is allow to creation of the revision, my must click the right mouse button in the window, containing the list of available objects and then choose from the menu: “Team” -> “Create Revision”. Depending on the current cursor position, we obtain the following results:

- Selected area which does not belong to any object



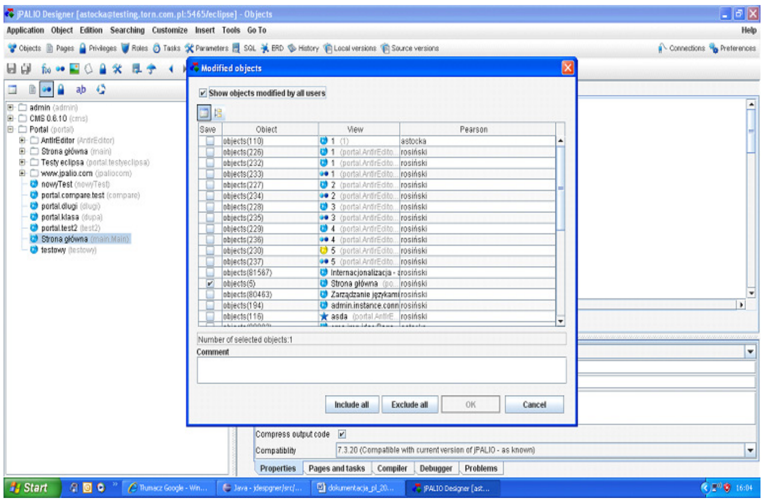
- Selected one of the available objects from the available list of objects – the chosen object is selected by default as an object to be archived.



2.2. List of modified objects

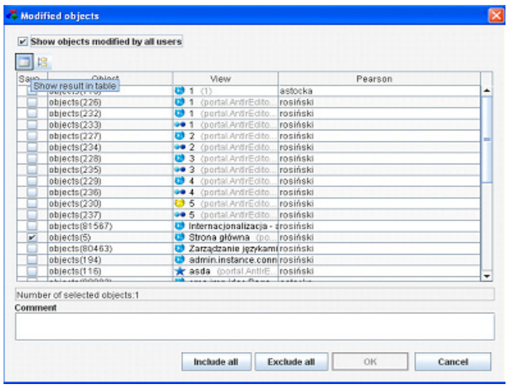
To create a revision you must select at least one object and add a comment.

By default, only objects modified by the user who wants to create a revision are shown. It is also possible to show all modified objects, you must only select the “Show objects modified by all users”, which was shown in the screenshot below.



The list of modified objects can be presented in two ways:

1. In the table


Modified objects

☑ Show objects modified by all users

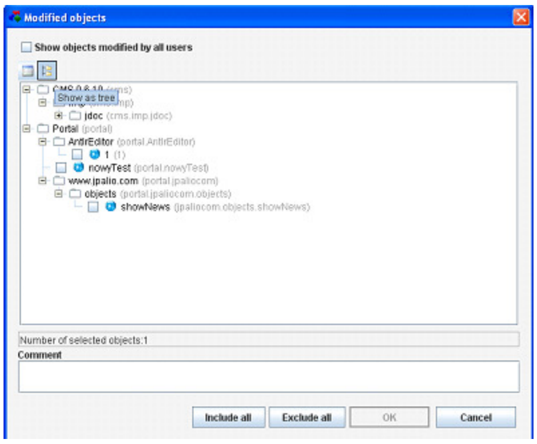
Save	Object	View	Pearson
		1 (1)	astocka
☐	objects(226)	1 (portal.AntlrEdito...	rosiński
☐	objects(232)	1 (portal.AntlrEdito...	rosiński
☐	objects(233)	1 (portal.AntlrEdito...	rosiński
☐	objects(227)	2 (portal.AntlrEdito...	rosiński
☐	objects(234)	2 (portal.AntlrEdito...	rosiński
☐	objects(228)	3 (portal.AntlrEdito...	rosiński
☐	objects(235)	3 (portal.AntlrEdito...	rosiński
☐	objects(229)	4 (portal.AntlrEdito...	rosiński
☐	objects(236)	4 (portal.AntlrEdito...	rosiński
☐	objects(230)	5 (portal.AntlrEdito...	rosiński
☐	objects(237)	5 (portal.AntlrEdito...	rosiński
☐	objects(81567)	Internacjonalizacja -	rosiński
✓	objects(5)	Strona główna (po...	rosiński
☐	objects(80463)	Zarządzanie językami	rosiński
☐	objects(194)	admin.instance.conn	rosiński
☐	objects(116)	asda (portal.AntlrE...	rosiński

Number of selected objects:1

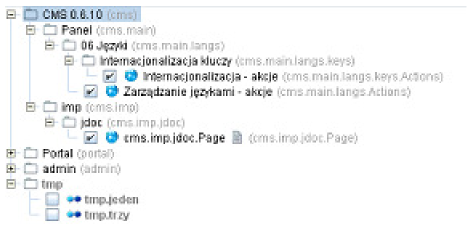
Comment

2. In the tree

In trees both modified objects and objects in the hierarchy above them are visible, what makes it easier to find concrete object, as well as facilities management.


Modified objects
Show objects modified by all users
CMS 0.6.10 (cms)
Show as tree
jdoc (cms.imp.jdoc)
Portal (portal)
AntlrEditor (portal.AntlrEditor)
1 (1)
nowyTest (portal.nowyTest)
www.jpalio.com (portal.jpaliocom)
objects (portal.jpaliocom.objects)
showNews (jpaliocom.objects.showNews)
Number of selected objects:1
Comment
Include all
Exclude all
OK
Cancel

Right-clicking on the folder displays menu which enables the automatic checking/unchecking all objects which are appear in this folder.

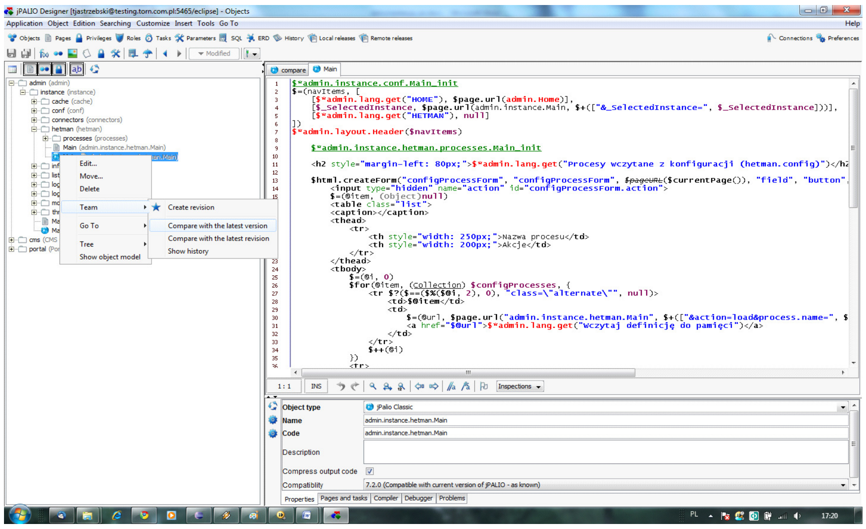

CMS 0.6.10 (cms)
Panel (cms.main)
06 Języki (cms.main.langs)
Internacjonalizacja kluczy (cms.main.langs.keys)
Internacjonalizacja - akcje (cms.main.langs.keys.Actions)
Zarządzanie językami - akcje (cms.main.langs.Actions)
imp (cms.imp)
jdoc (cms.imp.jdoc)
cms.imp.jdoc.Page (cms.imp.jdoc.Page)
Portal (portal)
admin (admin)
tmp
tmp.jeden
tmp.trzy

In order to distinguish objects removed from the existing, they are highlighted in gray. An example was presented in the figure above.

2.3. Comparison revisions

In order to compare two different versions of an object, click the right mouse button on the selected object, which brings up a menu, then select "Team", then choose one of available options:

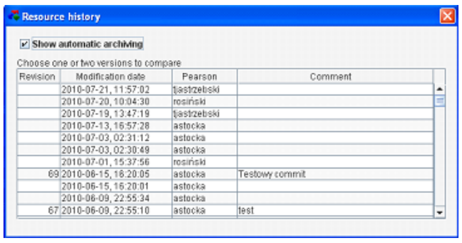
1. “Compare with the latest version” – compares the local version of the object, with the latest archives version of the object
2. “Compare with the latest revision” – compares the local version of the object, with the latest archives version of the object, which was included in revision
3. “Show history” – displays a window containing the history of the object


jPALIO Designer [tjastrzebski@testing.torn.com.pl:5465/eclipse] - Objects
Application Object Edition Searching Customize Insert Tools Go To
Help
Objects Pages Privileges Roles Tasks Parameters SQL ERD History Local releases Remote releases
Connections Preferences
Modified
admin (admin)
instance (instance)
cache (cache)
conf (conf)
connectors (connectors)
hetman (hetman)
processes (processes)
Main (admin.instance.hetman.Main)
Edit...
Move...
Delete
Team
Go To
Tree
Show object model
Create revision
Compare with the latest version
Compare with the latest revision
Show history
compare
Main
1 : 1
INS
Inspections
Properties
Pages and tasks
Compiler
Debugger
Problems
PL
17:20

```
$*admin.instance.conf.Main_init
$=(navItems, [
    [$*admin.lang.get("HOME"), $page.url(admin.Home)],
    [$_SelectedInstance, $page.url(admin.instance.Main, $+(["&_SelectedInstance=", $_SelectedInstance]))],
    [$*admin.lang.get("HETMAN"), null]
])
$*admin.layout.Header($navItems)

    $*admin.instance.hetman.processes.Main_init

    <h2 style="margin-left: 80px;">$*admin.lang.get("Procesy wczytane z konfiguracji (hetman.config)")</h

    $html.createForm("configProcessForm", "configProcessForm", ~~$pageURL~~($currentPage()), "field", "button",
        <input type="hidden" name="action" id="configProcessForm.action">
        $=(@item, (Object)null)
        <table class="list">
        <caption></caption>
        <thead>
            <tr>
                <th style="width: 250px;">Nazwa procesu</td>
                <th style="width: 200px;">Akcje</td>
            </tr>
        </thead>
        <tbody>
            $=(@i, 0)
            $for(@item, (Collection) $configProcesses, {
                <tr $?($==($%($@i, 2), 0), "class=\"alternate\"", null)>
                    <td>$@item</td>
                    <td>
                        $=(@url, $page.url("admin.instance.hetman.Main", $+(["&action=load&process.name=", $
                        <a href="$@url">$*admin.lang.get("Wczytaj definicję do pamięci")</a>
                    </td>
                </tr>
                $++(@i)
            })
            <tr>
```

Object type	jPalio Classic
Name	admin.instance.hetman.Main
Code	admin.instance.hetman.Main
Description	
Compress output code	✓
Compatiblity	7.2.0 (Compatible with current version of jPALIO - as known)

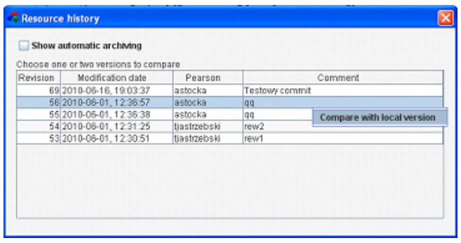
In the “Resource history” window, only the modifications that were included in the revision are displayed by default. If you select “Show automatic archiving” , you will see automatic archiving of the objects too.



Choose one or two versions to compare

Revision	Modification date	Pearson	Comment
	2010-07-21, 11:57:02	tjastrzebski	
	2010-07-20, 10:04:30	rosiński	
	2010-07-19, 13:47:19	tjastrzebski	
	2010-07-13, 16:57:28	astocka	
	2010-07-03, 02:31:12	astocka	
	2010-07-03, 02:30:49	astocka	
	2010-07-01, 15:37:56	rosiński	
69	2010-06-15, 16:20:05	astocka	Testowy commit
	2010-06-15, 16:20:01	astocka	
	2010-06-09, 22:55:34	astocka	
67	2010-06-09, 22:55:10	astocka	test

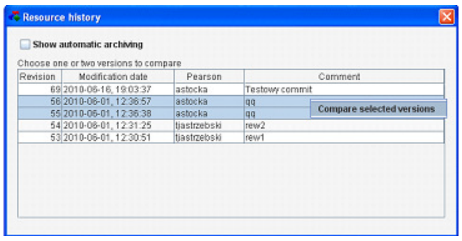
When you select one of the archival version, and pressed the right mouse button menu appears on the "Compare with local version”, which makes comparison of selected versions of the object with the local version.



Choose one or two versions to compare

Revision	Modification date	Pearson	Comment
69	2010-06-16, 19:03:37	astocka	Testowy commit
56	2010-06-01, 12:36:57	astocka	qq
55	2010-06-01, 12:36:38	astocka	qq
54	2010-06-01, 12:31:25	tjastrzebski	rew2
53	2010-06-01, 12:30:51	tjastrzebski	rew1

When you select two of the archival versions, and pressed the right mouse button menu appears on the "Compare selected versions”, which makes comparison between two selected archival versions.

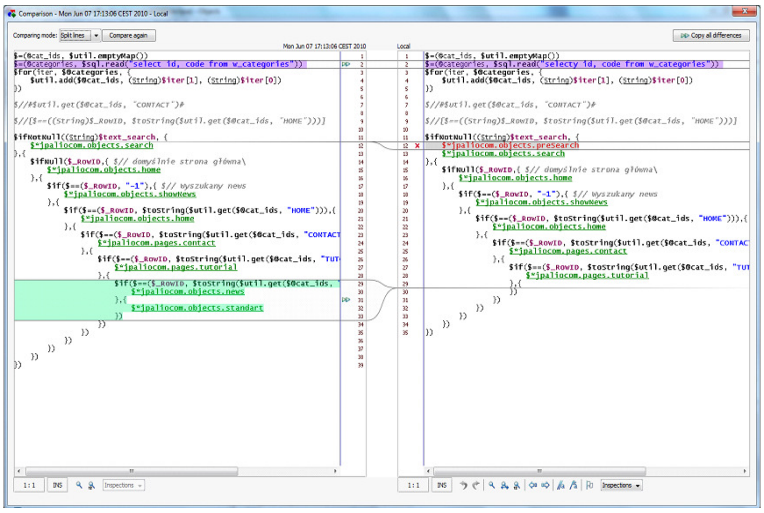


Choose one or two versions to compare

Revision	Modification date	Pearson	Comment
69	2010-06-16, 19:03:37	astocka	Testowy commit
56	2010-06-01, 12:36:57	astocka	qq
55	2010-06-01, 12:36:38	astocka	qq
54	2010-06-01, 12:31:25	tjastrzebski	rew2
53	2010-06-01, 12:30:51	tjastrzebski	rew1

3. Object comparison window

The main window contains two adjacent editors: left – showing archival object's content and right containing local version of the object. Over the editors there are labels describing which version there is into editor.


Comparison - Mon Jun 07 17:13:06 CEST 2010 - Local
Comparing mode: Split lines
Compare again
Copy all differences
Mon Jun 07 17:13:06 CEST 2010
Local
1 : 1
INS
Inspections

```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("select id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTACT
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                        $if($==($_RowID, $toString($util.get($@cat_ids, '
                            $*jpaliocom.objects.news
                        },{
                            $*jpaliocom.objects.standart
                        })
                    })
                })
            })
        })
    })
})
```

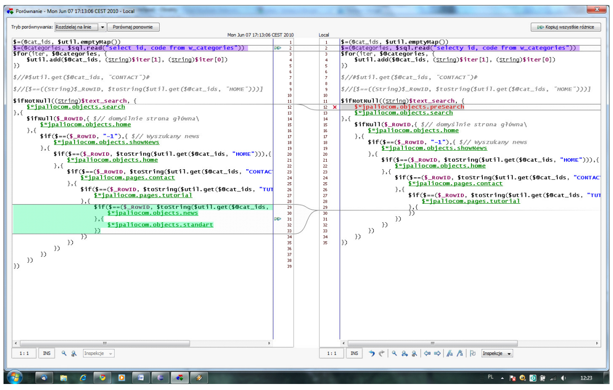
```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("selecty id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.presearch
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                    })
                })
            })
        })
    })
})
```

Right editor with local version is integreted with the main editor in designer application, it means that all changes made here are automaticly applying in the main editor and saving also is the same as save on main application level.

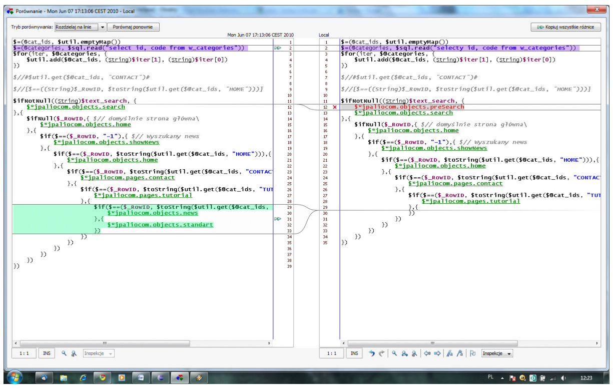
3.1. Markers overlaying differs

Main functionality od the comparison frame is showing differs betweene two difrent version of the object. Graphicly it's represented by translucent markers overlaying corespodnding code fragments. Depending on the type of changes made in the code there are three general differences represented by independent markers:

- Addition – in local version there is a code fragment, that didn't exsist into archived content. It's posible to remove added fragment from local version of the object by pressing button. Added code is marked in grey.


Porównanie - Mon Jun 07 17:13:06 CEST 2010 - Local
Tryb porównywania: Rozdzielaj na linie
Porównaj ponownie
Kopiuj wszystkie różnice
Mon Jun 07 17:13:06 CEST 2010
Local
1 : 1
INS
Inspekcje

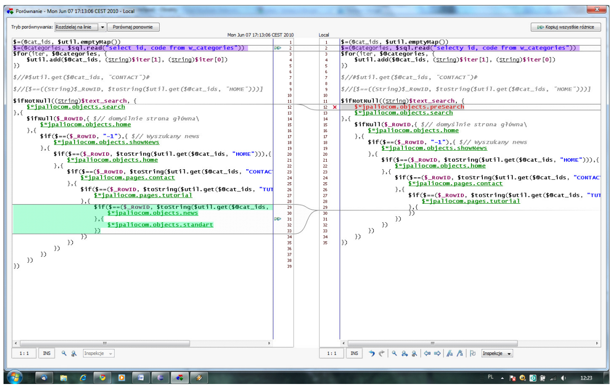
- Removal – in local version don't exsist the fragment, that exsisted in archived content. The fragment is marked in green and it's available to copy the fragment to local editor by pressing button.


Porównanie - Mon Jun 07 17:13:06 CEST 2010 - Local
Tryb porównywania: Rozdzielaj na linie
Porównaj ponownie
Kopiuj wszystkie różnice
Mon Jun 07 17:13:06 CEST 2010
Local
1 : 1
INS
Inspekcje

```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("select id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                        $if($==($_RowID, $toString($util.get($@cat_ids,
                            $*jpaliocom.objects.news
                        },{
                            $*jpaliocom.objects.standart
                        })
                    })
                })
            })
        })
    })
})
```

```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("selecty id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.preSearch
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                    })
                })
            })
        })
    })
})
```

- Edition – the fragment occurs both in local and archived version, but their contents are different. The fragments in both editors are marked in blue color. Action fired on button is replacing local fragment by their corresponding fragment in history.


Porównanie - Mon Jun 07 17:13:06 CEST 2010 - Local
Tryb porównywania: Rozdzielaj na linie
Porównaj ponownie
Kopiuj wszystkie różnice
Mon Jun 07 17:13:06 CEST 2010
Local
1 : 1
INS
Inspekcje

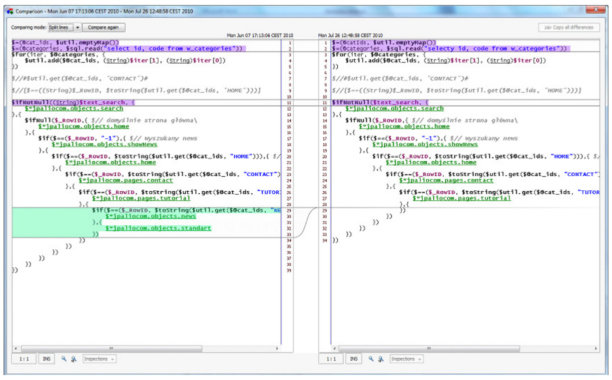
```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("select id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                        $if($==($_RowID, $toString($util.get($@cat_ids,
                            $*jpaliocom.objects.news
                        },{
                            $*jpaliocom.objects.standart
                        })
                    })
                })
            })
        })
    })
})
```

```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("selecty id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.preSearch
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                    })
                })
            })
        })
    })
})
```

In right corner of the comparison window there is a button "Copy all differencess". It's dedicated to remove all differencess by moving archived content (left editor) to local version of the object. After presing the button all content of the local object's version is replacing by archived code collected in left editor. This functionality may be used to reverting archived versions.

3.2. Comparing two archival versions of the object

The window allows not only compare local version with archival but also it's posible to compare both archivals. In the context it's available only preview contents of the object under the versions without any changes. It's clear that icons on marker to move content form one editor to other is disabled, the right editor is in read-only mode and the button "Copy all differences" is inactive.


Comparison - Mon Jun 07 17:13:06 CEST 2010 - Mon Jul 26 12:48:58 CEST 2010
Comparing mode: Split lines
Compare again
Copy all differences
Mon Jun 07 17:13:06 CEST 2010
Mon Jul 26 12:48:58 CEST 2010
1:1
INS
Inspections

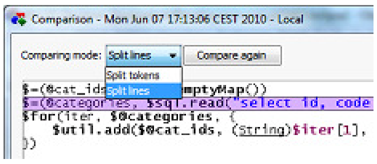
```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("select id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{ $/
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTACT")
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUTOR
                        $*jpaliocom.pages.tutorial
                    },{
                        $if($==($_RowID, $toString($util.get($@cat_ids, "NE
                            $*jpaliocom.objects.news
                        },{
                            $*jpaliocom.objects.standart
                        })
                    })
                })
            })
        })
    })
})
```

```
$=(@catIds, $util.emptyMap())
$=(@categories, $sql.read("selecty id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME")))]

$ifNotNull($text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{ $/
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTACT"]
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUTOR
                        $*jpaliocom.pages.tutorial
                    },{
                    })
                })
            })
        })
    })
})
```

3.3. Comparison algorithm

Comparison engine is fired even befor displaying its on window and it isn't refreshig after making any changes in local editor so appearing new difference or removing old are not recognise automaticly and not marked in markers overlay. To refresh and re-compare versions you need to press "Compare again" button that it's situated in the left top corner of the window.

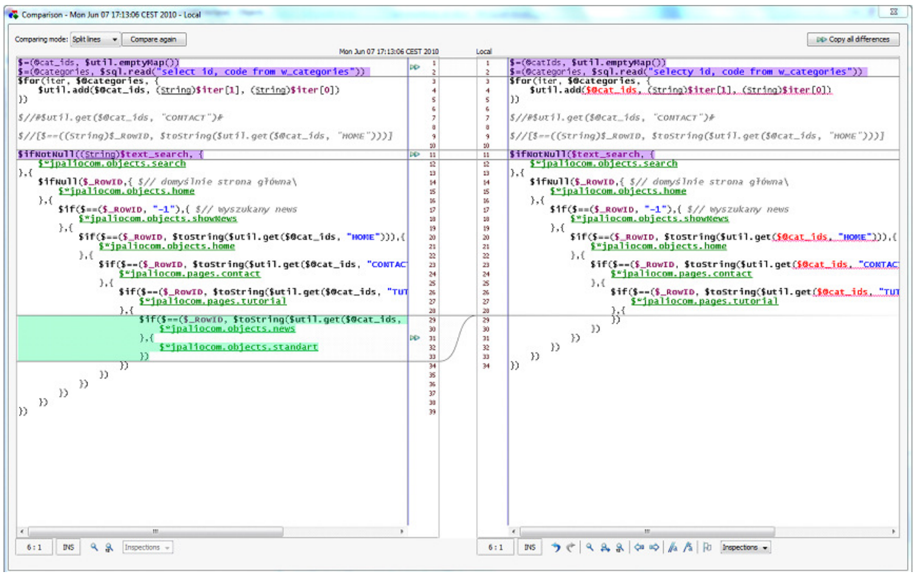
Re-comparing feature allows us to chose comparison mode - the way how the algorithm should divide the code into elements to compare. The single element has the property, that the matching elements in both codes are compared like common strings – char by char. It's mean that even small differ on one char causes that the elements are treated like different. Only complete compatibility between elements may be recognized as identical.


Comparison - Mon Jun 07 17:13:06 CEST 2010 - Local
Comparing mode: Split lines
Compare again
Split tokens
Split lines

```
$=(@cat_ids                mptyMap())
$=(@categor ies, $sql.read("select id, code
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1],
})
```

To chose comparing mode you need only select from situated next to "Compare again" button list proper option. There are two comparison mode possible now:

- Split lines – default mode, all lines is accepted as single elements for the comparison algorithm. Even one char differ in the matching lines is recognizing as all line is differ.



Comparison - Mon Jun 07 17:13:06 CEST 2010 - Local

Comparing mode: Split lines

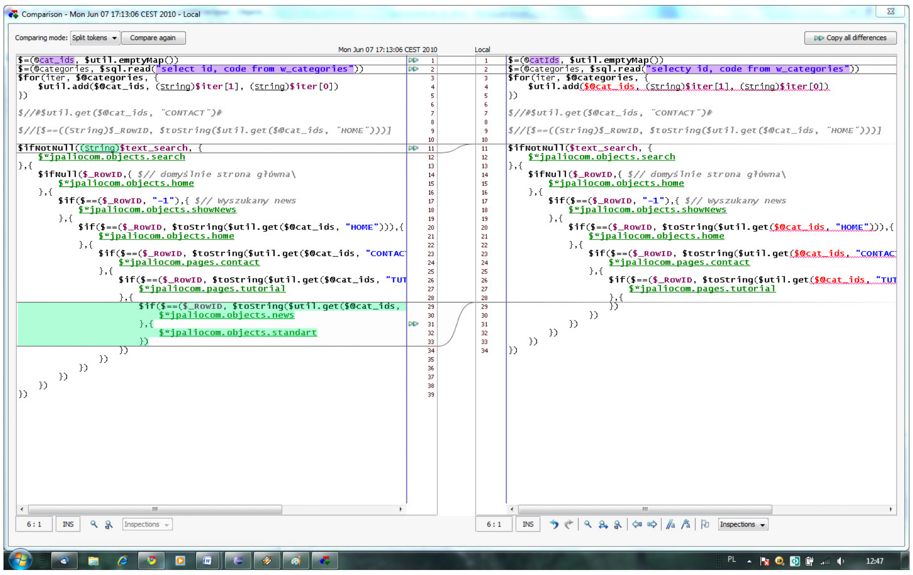
Mon Jun 07 17:13:06 CEST 2010

```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("select id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME"))))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                        $if($==($_RowID, $toString($util.get($@cat_ids,
                            $*jpaliocom.objects.news
                        },{
                            $*jpaliocom.objects.standart
                        })
                    })
                })
            })
        })
    })
})
```

Local

```
$=(@catIds, $util.emptyMap())
$=(@categories, $sql.read("selecty id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME"))))]

$ifNotNull($text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                    })
                })
            })
        })
    })
})
```

- Split tokens – algorithm divide code into fragments based on lexicographical code analysis. Single token is often shorter that single line and match to jPALIO language elements like variable name, object's code, string or function name. Because of to the tokens recognizing is using syntax analyze even small syntax error may cause, that tokenizer recognize two different tokens sets so you can see that caparisoning versions look like completely different despite in the fact it may not be true (General it can appears when you don't close the bracket or leave unquote).



Comparison - Mon Jun 07 17:13:06 CEST 2010 - Local

Comparing mode: Split tokens

Mon Jun 07 17:13:06 CEST 2010

```
$=(@cat_ids, $util.emptyMap())
$=(@categories, $sql.read("select id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME"))))]

$ifNotNull((String)$text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                        $if($==($_RowID, $toString($util.get($@cat_ids,
                            $*jpaliocom.objects.news
                        },{
                            $*jpaliocom.objects.standart
                        })
                    })
                })
            })
        })
    })
})
```

Local

```
$=(@catIds, $util.emptyMap())
$=(@categories, $sql.read("selecty id, code from w_categories"))
$for(iter, $@categories, {
    $util.add($@cat_ids, (String)$iter[1], (String)$iter[0])
})

$//#$util.get($@cat_ids, "CONTACT")#

$//[$==((String)$_RowID, $toString($util.get($@cat_ids, "HOME"))))]

$ifNotNull($text_search, {
    $*jpaliocom.objects.search
},{
    $ifNull($_RowID,{ $// domyślnie strona główna\
        $*jpaliocom.objects.home
    },{
        $if($==($_RowID, "-1"),{ $// wyszukany news
            $*jpaliocom.objects.showNews
        },{
            $if($==($_RowID, $toString($util.get($@cat_ids, "HOME"))),{
                $*jpaliocom.objects.home
            },{
                $if($==($_RowID, $toString($util.get($@cat_ids, "CONTAC
                    $*jpaliocom.pages.contact
                },{
                    $if($==($_RowID, $toString($util.get($@cat_ids, "TUT
                        $*jpaliocom.pages.tutorial
                    },{
                    })
                })
            })
        })
    })
})
```

4. Administration Panel

Administration panel is “rich” version of already known administration page – html.info. Due to the lack of authorization, it's recommended to disable access to old administration page – html.info (you can do it by removing few lines from web.xml – Tomcat's or another container's servlets configuration file).

4.1. Access

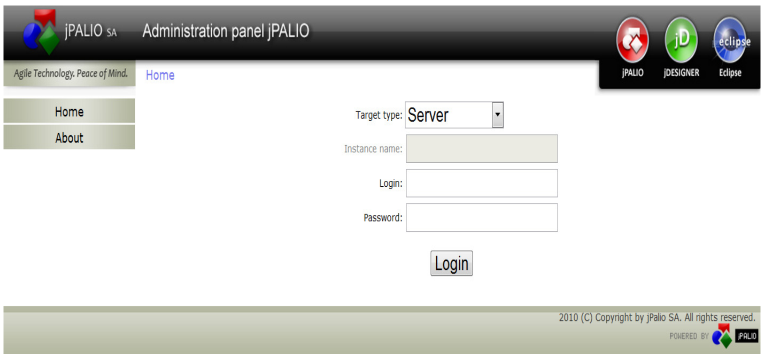
Administration panel works as common web application. To access administration panel, just type url in your browser: http://server_address/palio/html.run, where:

- server_address is host name (including optional port number after colon),
- palio (name of Tomcat's or another container's context).

4.2. Authorization

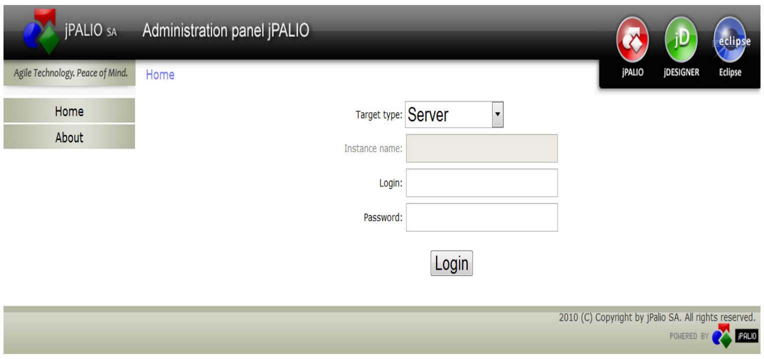
Access to the panel requires authority and only users with expected privileges can use it. There are 2 type of authority:

- Logging to server – After it you have full access to manage server and all of their instances too. To login to sever you should select on home page option Server from “Target type” list and fill valid login and password to the server administration account (more information can be found in documentation of jPALIO server configuration file – config.xml)


jPALIO SA
Administration panel jPALIO
Agile Technology. Peace of Mind.
jPALIO
jDESIGNER
Eclipse
Home
Home
About
Login
2010 (C) Copyright by jPalio SA. All rights reserved.
POWERED BY jPALIO

Target type: Server
Instance name:
Login:
Password:

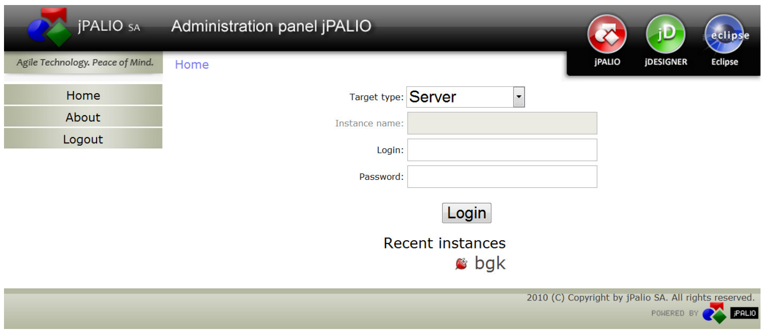
- Logging to instance – The panel allows you to sign directly into the specified instance without need of logging to the server first. To follow this case, you can use your private login and password valid in jDESIGNER application to connect to the instance (specified account must have “designer” role associated). If you want to log into specified instance you should select on home page Instance option from “Target type” list and fill below “Instance name” where you have a jDESIGNER account.



It is possible to set home page to automatically connect to the specified instance. To do this, just type SelectedInstance=instance_name parameter in url, eg.

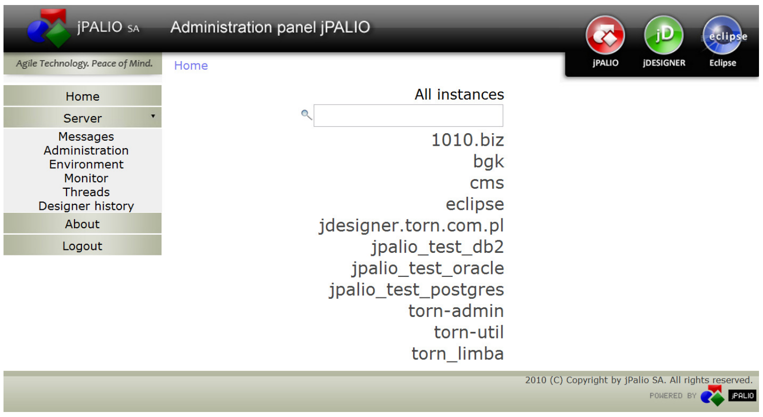
address http://server_address/palio/html.run?_SelectedInstance=foo automatically open logging page to foo instance.

User, who logs directly to an instance, may still log into another instances at the same time. Home page – besides logging form – always displays instances that you are already logged in and you can quickly switch to these instances just by clicking on their names.



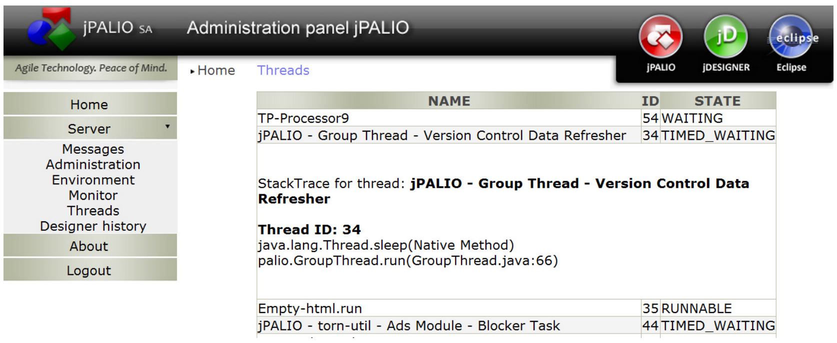
4.3. Server management

If you are logged to server, you can see list of all instances started on the server. When you click on their names you will be able to manage the selected instance. (More details about instance management will be discussed later).



You can perform administration tasks using menu on the left side of panel.

- Messages – list of messages that server "wants" to display to the user.
- Administration – The main administration actions page. Recently there is only „Rescan instances directory" button, enabling rescan the server directory to refresh instances list and start all newly added instances (without restarting of server).
- Environment – Information page displays server's environment parameters.
- Monitor – list of all Groovy classes hold by JVM. Additionally you can see here how many versions of the class exists (saving Groovy class through jDESIGNER application creates new version of that class, some data however may still reference to the "previous" version).
- Threads – List of all server threads. When you click any thread, it appends stack trace content for this thread.

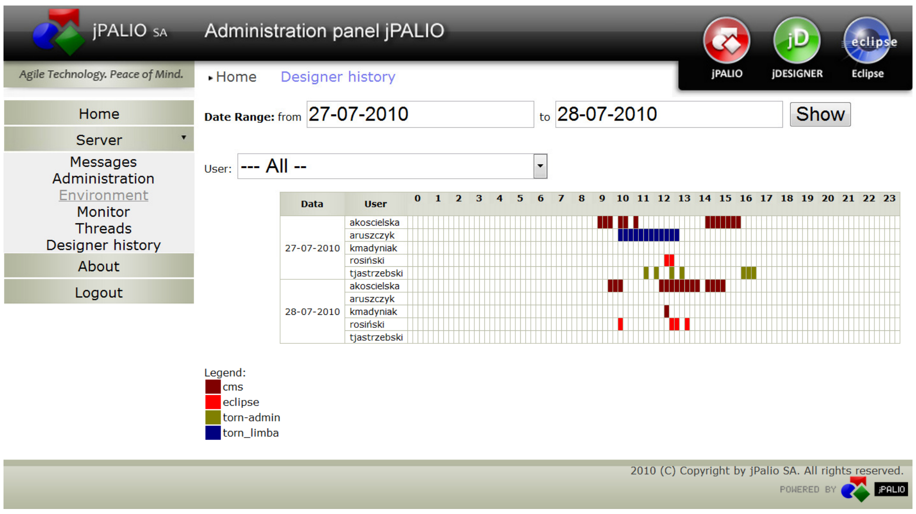


NAME	ID	STATE
TP-Processor9	54	WAITING
jPALIO - Group Thread - Version Control Data Refresher	34	TIMED_WAITING

```
StackTrace for thread: jPALIO - Group Thread - Version Control Data
Refresher

Thread ID: 34
java.lang.Thread.sleep(Native Method)
palio.GroupThread.run(GroupThread.java:66)
```

NAME	ID	STATE
Empty-html.run	35	RUNNABLE
jPALIO - torn-util - Ads Module - Blocker Task	44	TIMED_WAITING

- Designer history – module dedicated to tracing of user activity. This activity for selected instances is presented using sheet with 15 minutes resolution. You can filter presented data by selecting particular developer from the User list. When cursor is over non-empty cell, tooltip with detailed data is displayed.



Date Range: from 27-07-2010 to 28-07-2010

User: --- All --

Data	User
27-07-2010	akoscielska
	aruszczyk
	kmadyniak
	rosiński
	tjastrzebski
28-07-2010	akoscielska
	aruszczyk
	kmadyniak
	rosiński
	tjastrzebski

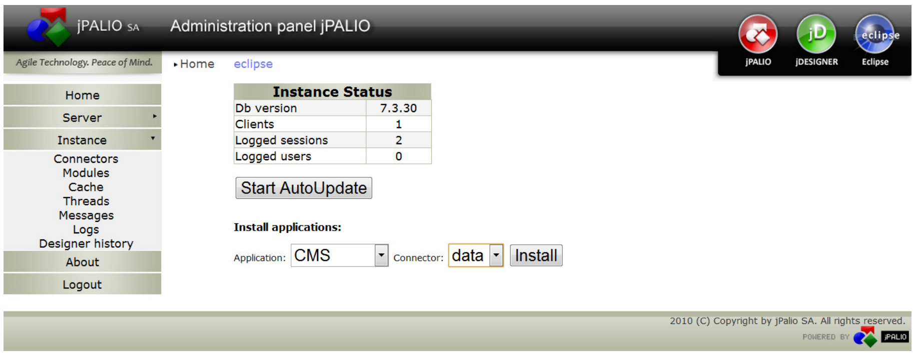
Legend:
- 🟥 cms
- 🟥 eclipse
- 🟩 torn-admin
- 🟦 torn_limba

2010 (C) Copyright by jPalio SA. All rights reserved.

4.4. Instance management

The aim of the functionality featured by the administration panel in the area of instance management is to complement features of jDESIGNER application.

- Main instance management page – besides information table there are here also „Start AutoUpdate” button, that allows you to force re-execution of automatic update of database structure (scanning of existing structure and filling gaps). You can also use form “Install application” to install database structures for external application to any selected connector. At the moment, there are 3 applications possible to install structures for: CMS – Content Management System, Newsletter – application which dispatch mails to registered subscribers and Ads – advert server. CMS application requests installed structures for all three applications, CMS, Newsletter and Ads. You can access this page always by clicking “Instance” on the left menu.



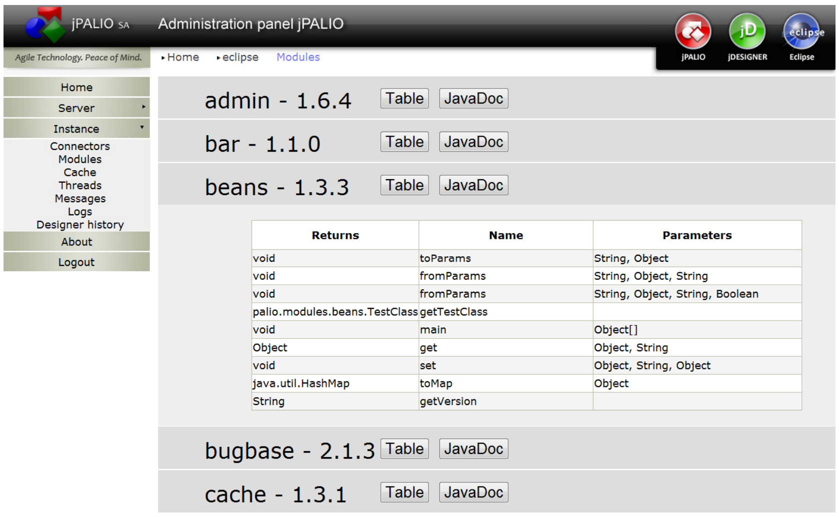
Instance Status	
Db version	7.3.30
Clients	1
Logged sessions	2
Logged users	0

Install applications:

Application: CMS Connector: data

- Connectors – displays connector list and allows management of them.

- Modules – Table of all available modules in managed instance. It’s possible to view only headers of all functions available within each module or detailed JavaDoc documentation.



admin - 1.6.4

bar - 1.1.0

beans - 1.3.3

Returns	Name	Parameters
void	toParams	String, Object
void	fromParams	String, Object, String
void	fromParams	String, Object, String, Boolean
palio.modules.beans.TestClass	getTestClass	
void	main	Object[]
Object	get	Object, String
void	set	Object, String, Object
java.util.HashMap	toMap	Object
String	getVersion	

bugbase - 2.1.3

cache - 1.3.1

- Cache – cache management module. Allows not only listing all caches in the instance but also flushing content from selected or all buffers.

- Threads – exact as threads table in server management, but shows only threads owned by current instance.
- Messages – list of user messages which describes state of instance.
- Logs – interactive view of instance's logs which uses AJAX technology (web equivalent of tail command known from Linux / Unix)
- Designer history – Developers activity tracing module. Exact as "Designer history" in server management except that shows only activity in the selected instance.

5. Management of active patches

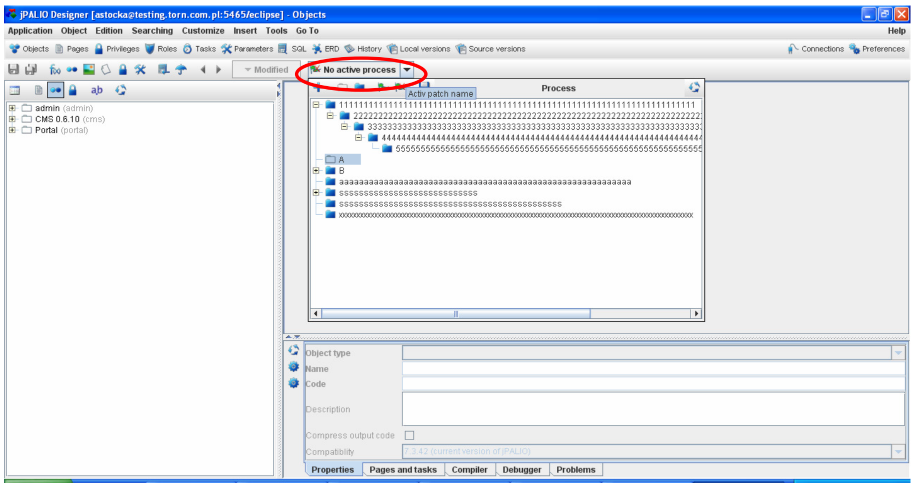
Since version jPALIO 7.4 management of patches has been transferred to module „Objects". In the panel appearing in a top part of window, button has been added which displays active patch.

5.1. Management of active patches in the module „Objects"

Depending on your patch it is possible to obtain the following results:

- The name of patch, that is active.
- "No active process", when there is no active patch.
- The patch name on a red background, in case you try to set closed patch as an active.

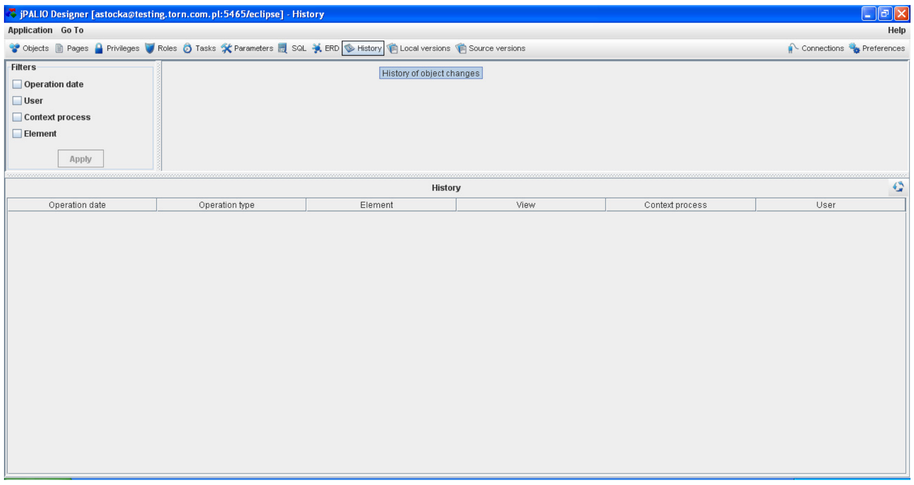
Described button is located as it's presented on the picture below:



Pressing the button displays a window containing a list of available patches as well as change the active patches.

5.2. Renaming of module „Project” to „History”

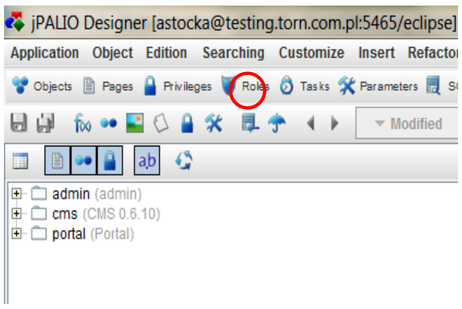
After transferring of active patches management to module “Objects”, module “Projects” contains only history viewer. Because of this, module “Project” has been renamed to “History”.



6. Uploading patches

New version of jPALIO server significantly extends configuration of patches uploading process. Patch uploading configuration window has been divided into three independent tabs. Tab “Compatible” holds configuration known from previous versions. The other two newly added tabs describing full advanced and simplified configuration. Main feature of introduced changes is possibility to determine taken actions for each type of elements separately. (e.g. you can upload roles, privileges, roles to privileges assignments without uploading pages). In addition to objects and multimedia defined a new type of action “Upload but overwrite only older” (the action causes replacement of element by element from patch only if modification time of element from patch is newer than modification time of element on target server). There is also another feature which gives you capability to enable extra action to be taken on uploaded tasks (you can choose to automatically enable/disable/leave as it is in the patch all tasks uploaded from patch).

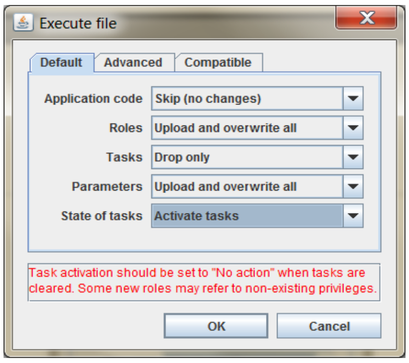
At the same time due to transferring active patches management to the module “Objects”, “Upload patch” action was also transferred to that module.


jPALIO Designer [astocka@testing.torn.com.pl:5465/eclipse]
Application Object Edition Searching Customize Insert Refacto
Objects Pages Privileges Roles Tasks Parameters
Modified
admin (admin)
cms (CMS 0.6.10)
portal (Portal)

6.1. Tab „Default”

Tab „Default” is default, simplified form of new configuration. In accordance with the new configuration (similar to the „Advanced” tab), user must determine expected actions to be taken for each type of uploaded elements. In case of „Default” tab, options are grouped in a form convenient for standard users:

- Option „Application code” is action selection for elements: directories, MIME types, objects, multimedia, pages, privileges and privileges to pages assignations,
- option „Roles” is responsible for roles and privileges assigned to roles,
- option „Tasks” is responsible for actions taken on tasks and groups of tasks,
- options „Parameters” concerns global parameters (exactly as in “Advanced” tab),
- option „State of tasks” describes extra action which may be performed on all uploaded tasks (automatic activation/deactivation/leaving as it was in the patch).



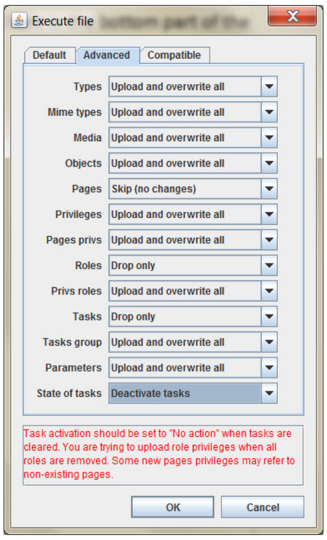
Application code: Skip (no changes)
Roles: Upload and overwrite all
Tasks: Drop only
Parameters: Upload and overwrite all
State of tasks: Activate tasks

Task activation should be set to "No action" when tasks are cleared. Some new roles may refer to non-existing privileges.

Selection of action “Upload but overwrite only older” in the “Application code” option consideres only objects and multimedia. For other elements there action is identical to action ”Upload and overwrite all”.

6.2. Tab „Advanced”

Tab „Advanced” allows to selection individual actions for each type of elements (as in the image below). User is responsible for choosing such configuration which is sensible. Because many of these is potentially misleading, you should take a look for hints written in red font in the bottom part of the window.



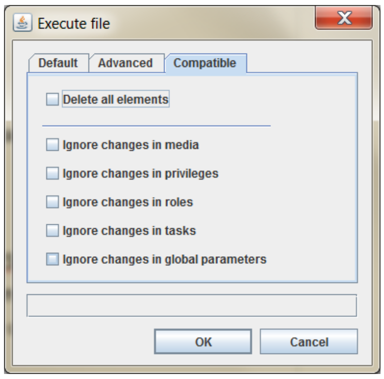
Types	Upload and overwrite all
Mime types	Upload and overwrite all
Media	Upload and overwrite all
Objects	Upload and overwrite all
Pages	Skip (no changes)
Privileges	Upload and overwrite all
Pages privs	Upload and overwrite all
Roles	Drop only
Privs roles	Upload and overwrite all
Tasks	Drop only
Tasks group	Upload and overwrite all
Parameters	Upload and overwrite all
State of tasks	Deactivate tasks

Task activation should be set to "No action" when tasks are cleared. You are trying to upload role privileges when all roles are removed. Some new pages privileges may refer to non-existing pages.

Except of non-joining elements into larger groups, this tab is fully compatible to the "Default" tab.

6.3. Tab "Compatible"

This tab has been transferred unchanged from previous versions of jDESIGNER. Aim of the existence of this tab is to ensure continuity without forcing users to learn new functionality. However, it is recommended to use "Default" or "Advances" tab due to more flexible configuration.



- ☐ Delete all elements
- ☐ Ignore changes in media
- ☐ Ignore changes in privileges
- ☐ Ignore changes in roles
- ☐ Ignore changes in tasks
- ☐ Ignore changes in global parameters

7. Automatic update

jPALIO 7.4 introduced a major extension to AutoUpdate. Since that moment, each process of automatic update of database structures will be preceded by scanning of actual database structure. So the update process may not only update expected structures but also make up any deficiencies.

Because of that, users who interfere with the system tables, should properly configure AutoUpdate process, which tables or sequences should be ignored.

7.1. Instance configuration

To mark which tables or sequences should be ignored, you have to add the following entries to the instance configuration file. For these tables, there will be also ignored constraints (foreign keys) which refer to those tables.

```
<autoupdate>

        <table name="P_OBJECTS" action="skip"/>

       <sequence name="P_OBJECTS_S" action="skip"/>

</autoupdate>
```

In most cases, configuration of AutoUpdate may be skipped.

7.2. Creating/updating structures for external applications

AutoUpdate process has been extended on other non-system structures. It may be used to create or update structures for several applications like CMS, Newsletter or Ads – these processes may be launched within administration panel.

7.3. Principles of AutoUpdate

Algorithm of automatic update is based on several principles:

- externally added unknown tables or columns will not be removed,
- only selected tables or columns (which were hard-coded into update algorithm) will be removed,
- missing tables, columns and constraints will be restored,
- size of column may be expanded, never shortened – even when expected size is lower than actual,
- column constrains which that interfere with expected constrains will be removed and replaced by expected ones..

8. Server configuration

8.1. Server administrator account

Account of server administrator is used to manage the server. In administration panel you can sign directly into particular instance or whole server (in these case you have access to all instances). Server administrator account is defined in config.xml file. Password to the account may be given in overt or encoded form. If you decide to give encoded form, to generate a password, you have to use a hash function and then Base64 encoding. In these case you have to type a name of used hash function (SHA1, MD5, etc.). In case of skipped hashing, as a password just type your original password in overt form (no Base64)

```
<admin login="admin" password="password" algorithm="sha1"/>
```

To get encoded password, you can use Linux / Unix command:

```
echo -n "hasło" | openssl dgst -sha1 -binary | openssl base64
```

8.2. Selection of automatically started version of jDESIGNER'a

When you click on the link jDESIGNER next to the name of an instance in the administration panel (or html.info), appropriate version of application is started. Till now, mapping of server versions to client versions was created arbitrarily. Since jPALIO 7.4, selection is done automatically. Because of this, we give you an opportunity to influence, what version of application will be launched for your server. You can add the following entry in configuration file

```
<designer version="latest"/>
```

if you want to start always latest version (appropriate for your server version) or

```
<designer version="stable"/>
```

if you want to start most stable version for your server. Parameter “version” is optional and default choice is to start stable version.

jPALIO SA
ul. Joteyki 20
02-317 Warszawa

Tel: +48 22 424 87 88
Fax: +48 22 313 23 37

Email: kontakt@jpalio.com

www.jpalio.com