

Script utili

In questa sezione sono riportati alcuni script (lua, xslt, ecc...) realizzati nel corso degli anni per l'esecuzione di alcune attività sui record.

addAltreUOR e addAltreUORint (script lua per aggiungere codici in diritto Altre UOR)

Lo script consente di aggiungere massivamente codici di strutture sul diritto Altre UOR (visualizzazione e intervento) degli utenti a partire da un file xml strutturato in un formato specifico.

Il file xml ha la seguente struttura:

altreuor.xml

```
<?xml version="1.0"?>
<lista_utenti>
<utente id="matricola_utente1" diritto="CODUOR1.CODUOR2,..." />
<utente id="matricola_utente2" diritto="CODUOR1.CODUOR2,..." />
<utente id="matricola_utente..." diritto="CODUOR1.CODUOR2,..." />
</lista_utenti>
```

Il file xml deve essere denominato **altreuor.xml** per lo script dedicato al diritto Altre UOR in visualizzazione o **altreuorint.xml** per lo script che interviene sul diritto Altre UOR in intervento.

Lo script lua che interviene sul diritto Altre UOR in visualizzazione è il seguente:

addAltreUOR.lua

```
local dbg = require 'xw.dbg' ({active=false, key='acl'})
local xw = require('extraway')
xw.addPackage("$.xwTools") ;

--[[
--]]
function addAltreUOR()

    local sourceFileFullPath = xw.getCurrentArcName() .. ".script/altreuor.xml" ;
    local sourceRecord = xw.prepareRecordFromFile(sourceFileFullPath) ;
    if ( not(sourceRecord:isValid()) ) then
        xw.log.msg ('file non valido') ;
        return 'false' ;
    end

    local targetNodeSet = sourceRecord.selectNodes ('/lista_utenti/utente', nil, 1) ;
    if ( targetNodeSet:isValid() and 0 < #targetNodeSet ) then
        local targetNode = targetNodeSet.getFirst() ;
        while ( targetNode:isValid() ) do
            local idUtente = targetNode:getAttribute('id') ;
            local idDiritto = targetNode:getAttribute('diritto') ;
            local risultatoAggiornamento = aggiornaRecord(idUtente ,
"/persona_interna/personal_rights/right[@cod='DW-DOC-RIS5']", idDiritto)
            targetNode = targetNodeSet.getNext() ;
        end
    end

end

function aggiornaRecord(idUtente , pathElementoDaAggiornare, valoreDaUtilizzare)

    -- deactivate triggers
    xw.triggersOff() ;

    local recordIsModified = false ;
```



```
local targetRecordSet = xw.executeQuery('([UD,/xw/@UdType/]="persona_interna") AND
([/persona_interna/@matricola/]=" ' .. idUtente .. '")') ;
if ( targetRecordSet.isValid() and 0 < #targetRecordSet ) then
    local targetRecord = targetRecordSet.getFirst() ;
    while ( targetRecord.isValid() ) do
        recordIsModified = false ;
        targetRecord_LockResult = targetRecord:lock() ;
        if ( targetRecord_LockResult == true ) then

            local targetRecordRootNode =
targetRecord:selectFirstNode("/persona_interna", nil) ;

            local elementoDaAggiornare = targetRecord:selectFirstNode(
pathElementoDaAggiornare, nil) ;
            if ( elementoDaAggiornare.isValid() ) then
                elementoDaAggiornare:setValue(valoreDaUtilizzare) ;
                recordIsModified = true ;
            end

            if ( recordIsModified ) then
                local outcome = targetRecord:save() ;
                if ( outcome <= 0 ) then
                    targetRecord:unlock() ;
                    xw.log.msg('save error') ;
                end
            else
                targetRecord:unlock() ;
            end
        else
            xw.log.msg('lock error') ;
        end
        targetRecord = targetRecordSet.getNext() ;
    end
end

end

-- activate triggers
xw.triggersOn() ;

return 'true' ;
end

addAltireUOR = {
    addAltireUOR = addAltireUOR,
    addAltireUOR_signature = {
        outParms = {"result"}
    }
} ;
```

Mentre lo script che interviene sul diritto Altre UOR in intervento è il seguente:

[addAltireUORint.lua](#)

```
local dbg = require 'xw.dbg' ({active=false, key='acl'})
local xw = require('extraway')
xw.addPackage("$.xwTools") ;

--[[
--]]
function addAltireUORint()

    local sourceFileFullPath = xw.getCurrentArcName() .. ".script/altreuorint.xml" ;
    local sourceRecord = xw.prepareRecordFromFile(sourceFileFullPath) ;
```



```
if ( not(sourceRecord:isValid()) ) then
    xw.log.msg ('file non valido') ;
    return 'false' ;
end

local targetNodeSet = sourceRecord:selectNodes ('/lista_utenti/utente', nil, 1) ;
if ( targetNodeSet:isValid() and 0 < #targetNodeSet ) then
    local targetNode = targetNodeSet:getFirst() ;
    while ( targetNode:isValid() ) do
        local idUtente = targetNode:getAttribute('id') ;
        local idDiritto = targetNode:getAttribute('diritto') ;
        local risultatoAggiornamento = aggiornaRecord(idUtente ,
"/persona_interna/personal_rights/right[@cod='DW-D0C-RIS13']", idDiritto)
        targetNode = targetNodeSet:getNext() ;
    end
end

end

function aggiornaRecord(idUtente , pathElementoDaAggiornare, valoreDaUtilizzare)

    -- deactivate triggers
    xw.triggersOff() ;

    local recordIsModified = false ;

    local targetRecordSet = xw.executeQuery('([UD,/xw/@UdType/]="persona_interna") AND
([/persona_interna/@matricola/]=" ' .. idUtente .. '")') ;
    if ( targetRecordSet:isValid() and 0 < #targetRecordSet ) then
        local targetRecord = targetRecordSet:getFirst() ;
        while ( targetRecord:isValid() ) do
            recordIsModified = false ;
            targetRecord_LockResult = targetRecord:lock() ;
            if ( targetRecord_LockResult == true ) then

                local targetRecordRootNode =
targetRecord:selectFirstNode("/persona_interna", nil) ;

                local elementoDaAggiornare = targetRecord:selectFirstNode(
pathElementoDaAggiornare, nil) ;
                if ( elementoDaAggiornare:isValid() ) then
                    elementoDaAggiornare:setValue(valoreDaUtilizzare) ;
                    recordIsModified = true ;
                end

                if ( recordIsModified ) then
                    local outcome = targetRecord:save() ;
                    if ( outcome <= 0 ) then
                        targetRecord:unlock() ;
                        xw.log.msg('save error') ;
                    end
                else
                    targetRecord:unlock() ;
                end
            else
                xw.log.msg('lock error') ;
            end
            targetRecord = targetRecordSet:getNext() ;
        end
    end

    -- activate triggers
    xw.triggersOn() ;
```



```
        return 'true' ;
end

addAltreUORint = {
    addAltreUORint = addAltreUORint,
    addAltreUORint_signature = {
        outParms = {"result"}
    }
} ;
```