USING XOG QUERY API

Table of contents

Introduction

Normally data is XOGed into and out of Clarity with the Object API. Though there is a large number of sample files and schemas that is also perceived as limitation: no other data can be transferred and the filters are predefined. There is also the XOG query API allows to read out any data from Clarity database in a supported manner. It is not documented to the same extend and as the XOG Object API. The query API is only for reading data and not for writing. The output format is different from the output of the Object API and depends on how the query is designed. The design can be made so that with a little editing the output can be used for writing through the Object API. The possibility to customize the data transfer justifies the effort.

Example: Reading resource with manager filter

Setup 1

Resource manager is not one of the predefined filters for reading resource data, but a query can be easily filtered for resource manager.

```
SELECT RES.ID RID
    ,CASE
    WHEN (RES.PERSON_TYPE  = 300)
    THEN 'EMPLOYEE'
    WHEN (RES.PERSON_TYPE  = 301)
    THEN 'CONTRACTOR'
    END employmentType
    ,RES.EXTERNAL_ID  externalId
    ,RES.INCLUDE_FLAG includeInDatamart
    ,RES.IS_ACTIVE  isActive
    ,RES.IS_EXTERNAL isExternal
    ,MGR.USER_NAME   managerUserName       --RES.MANAGER_ID
    ,RES.UNIQUE_NAME resourceId
    ,CASE
    WHEN (RES.RESOURCE_TYPE  = 0)
    THEN 'LABOR'
    WHEN (RES.RESOURCE_TYPE  = 1)
    THEN 'EQUIPMENT'
```

```sql
        WHEN (RES.RESOURCE_TYPE = 2)
        THEN 'MATERIAL'
        WHEN (RES.RESOURCE_TYPE  = 3)
        THEN 'EXPENSE'
        END resourceType -- 0 = LABOR 1 = EQUIPMENT 2 = MATERIAL 3 = EXPENSE
        ,USR.USER_NAME username --RES.USER_ID
        ,RES.EMAIL emailAddress
        ,RES.FIRST_NAME firstName
        ,RES.LAST_NAME lastName
        ,CAL.HOURS_PER_DAY availability
        , PRR.PRISOPEN openForTimeEntry
        ,CASE
        WHEN (PRR.PRTRACKMODE  = 0)
        THEN 'None'
        WHEN (PRR.PRTRACKMODE  = 1)
        THEN 'Other'
        WHEN (PRR.PRTRACKMODE = 2)
        THEN 'Clarity'
        END trackMode
        , PRR.PRUSERFLAG1  userFlag1
        ,PRR.PRUSERFLAG2 userFlag2
        , CAL.PRNAME baseCalendar

 FROM  PRJ_RESOURCES PRR,   PRCALENDAR CAL, SRM_RESOURCES RES
 LEFT JOIN CMN_SEC_USERS MGR on RES.MANAGER_ID = MGR.ID
 LEFT JOIN CMN_SEC_USERS USR on RES.USER_ID = USR.ID
 Where RES.ID = PRR.PRID
    AND PRR.PRCALENDARID = CAL.PRID
    AND MGR.ID IS NOT NULL
    AND MGR.LAST_NAME LIKE'K%'
    AND MGR.FIRST_NAME LIKE'M%'
```

This and all other queries are for MS SQL server

After that is turned into NSQL and stored it can be used for reading resource data. Note that there are a number of where clauses in the query. This is because after query is saved minor modification do not create structural changes.

The following properties file is used

```
servername=localhost
username=kinnunenad
password=***********
portnumber=80
sslenabled=false
fipsenabled=false

output=../Output/resquery_output1.xml
input=../Input/query_read.xml
```

The input file used is

```xml
<Query xmlns="http://www.niku.com/xog/Query">
<Code>zresquery</Code>
</Query>
```

and the bat file to launch the transfer is

```
time
call xog -propertyfile query.properties
time
```

The purpose of the time statements is to allow simple clocking of the run.

Output 1

The format of the output file is not at all similar to the output from the Object API

```xml
- <QueryResult xmlns="http://www.niku.com/xog/Query">
    <Code>zresquery1</Code>
  - <Records>
    - <Record>
        <includeindatamart>1</includeindatamart>
        <openfortimeentry>0</openfortimeentry>
        <employmenttype>EMPLOYEE</employmenttype>
        <isexternal>0</isexternal>
        <managerusername>kinnunenad</managerusername>
        <lastname>Ueberoth</lastname>
        <resourcetype>LABOR</resourcetype>
        <firstname>Ed</firstname>
        <userflag1>0</userflag1>
        <userflag2>0</userflag2>
        <username>enduser</username>
        <externalid>null</externalid>
        <isactive>1</isactive>
        <emailaddress>jotakin</emailaddress>
        <rid>5003041</rid>
        <trackmode>None</trackmode>
        <resourceid>enduser</resourceid>
        <basecalendar>Standard</basecalendar>
      </Record>
    - <Record>
        <includeindatamart>1</includeindatamart>
```

Basically the tag labels are there but there so much additional tags that it is impractical to edit the result into an input file for XOG Object API. The query can be used as a data provider for a portlet and the portlet looks a little cleaner

employmentType	externalId	includeinDatamart	isActive	isExternal	managerUserName	resourceId	resourceType	use
EMPLOYEE			1	1	0 kinnunenad	Administrator	LABOR	Admir
EMPLOYEE			1	1	0 kinnunenad	labor_res	LABOR	labor_
			1	1	0 kinnunenad	Equipment	EQUIPMENT	
			1	1	0 kinnunenad	Res_material	MATERIAL	
EMPLOYEE			1	1	0 kinnunenad	enduser	LABOR	endus

The data can be exported to Excel which cleans all of the tags which no good either. You would have to create them manually into the spreadsheet.

Setup 2

The next approach is to develop the query so that it will create the tags and that the output will require only a little editing if any.

SELECT @SELECT:DIM:USER_DEF:IMPLIED:U:U.Userdata:Userdata@

```
FROM
(SELECT RID,
 ('<Resource employmentType="'+employmentType
+'" externalId="'+externalId
+'" includeInDatamart="'+includeInDatamart
+'" isActive="'+isActive
+'" isExternal="'+isExternal
+'" managerUserName="'+managerUserName
+'" resourceId="'+resourceId
+'" resourceType="'+resourceType
+'" username="'+username+'">'
+'<PersonalInformation'
+' emailAddress="'+emailAddress
+'" firstName="'+firstName
+'" lastName="'+lastName
+'"/> <ManagementInformation availability="'+availability
+'" openForTimeEntry="'+openForTimeEntry
+'" trackMode="'+trackMode
+'" userFlag1="'+userFlag1
+'" userFlag2="'+userFlag2+'"/>'
+' <CustomInformation/> <General/> <OBSAssocs completed="false"/> <SkillAssocs/>'
+' <Calendar baseCalendar="'+baseCalendar
+'" resetCalendar="false"/> </Resource>'
) Userdata

FROM
(SELECT RES.ID RID,
    convert(varchar,ISNULL(CASE
    WHEN (RES.PERSON_TYPE  = 300)
    THEN 'EMPLOYEE'
    WHEN (RES.PERSON_TYPE  = 301)
    THEN 'CONTRACTOR'
    END,'')) employmentType
    ,convert(varchar,ISNULL(RES.EXTERNAL_ID,RES.ID)) externalId
    ,convert(varchar,ISNULL(RES.INCLUDE_FLAG,'0')) includeInDatamart
    ,convert(varchar,ISNULL(RES.IS_ACTIVE,'0'))  isActive
    ,convert(varchar,ISNULL(RES.IS_EXTERNAL,'0'))  isExternal
    ,convert(varchar,ISNULL(MGR.USER_NAME,''))  managerUserName      --RES.MANAGER_ID
    ,RES.UNIQUE_NAME resourceId
    ,CASE
    WHEN (RES.RESOURCE_TYPE  = 0)
    THEN 'LABOR'
    WHEN (RES.RESOURCE_TYPE  = 1)
    THEN 'EQUIPMENT'
    WHEN (RES.RESOURCE_TYPE = 2)
    THEN 'MATERIAL'
    WHEN (RES.RESOURCE_TYPE  = 3)
    THEN 'EXPENSE'
    END resourceType -- 0 = LABOR 1 = EQUIPMENT 2 = MATERIAL 3 = EXPENSE
    ,convert(varchar,ISNULL(USR.USER_NAME,'')) username --RES.USER_ID
    ,convert(varchar,ISNULL(RES.EMAIL,'')) emailAddress
    ,ISNULL(RES.FIRST_NAME,'') firstName
    ,ISNULL(RES.LAST_NAME,'') lastName
    ,convert(varchar,CAL.HOURS_PER_DAY) availability
    ,convert(varchar,PRR.PRISOPEN) openForTimeEntry
    ,convert(varchar,(CASE
    WHEN (PRR.PRTRACKMODE  = 0)
    THEN 'None'
    WHEN (PRR.PRTRACKMODE  = 1)
    THEN 'Other'
    WHEN (PRR.PRTRACKMODE = 2)
    THEN 'Clarity'
    END)) trackMode
    ,convert(varchar,ISNULL(PRR.PRUSERFLAG1,''))  userFlag1
    ,convert(varchar,ISNULL(PRR.PRUSERFLAG2,'')) userFlag2
    , CAL.PRNAME baseCalendar
```

```
FROM  PRJ_RESOURCES PRR,   PRCALENDAR CAL, SRM_RESOURCES RES
LEFT JOIN CMN_SEC_USERS MGR on RES.MANAGER_ID = MGR.ID
LEFT JOIN CMN_SEC_USERS USR on RES.USER_ID = USR.ID


Where RES.ID = PRR.PRID
    AND PRR.PRCALENDARID = CAL.PRID
     AND MGR.ID IS NOT NULL
    AND MGR.LAST_NAME LIKE'K%'
    AND MGR.FIRST_NAME LIKE'M%'
) RESU) U
WHERE   @FILTER@
```

The NSLQ is very simple because the standard SQL in the wrapper is deliberately designed for that. The operational items are at the beginning of the line "like comma first" as suggested by Paul Maxwell which makes error checking easier.

Output 2

This time the output in the portlet is not very usable

but after export to Excel it is the body of the XML file for input

the xml output needs more editing

```xml
<QueryResult xmlns="http://www.niku.com/xog/Query">
  <Code>zresquery</Code>
- <Records>
  - <Record>
      <userdata><Resource employmentType="" externalId="5004042" includeInDatamart="1" isA
        isExternal="0" managerUserName="kinnunenad" resourceId="Equipment" resourceType="
        username=""><PersonalInformation emailAddress="" firstName="" lastName="Res Eq"/>
        <ManagementInformation availability="8" openForTimeEntry="1" trackMode="Clarity" use
        userFlag2="0"/> <CustomInformation/> <General/> <OBSAssocs completed="false"/> <
        <Calendar baseCalendar="Standard" resetCalendar="false"/> </Resource></userdata>
    </Record>
  - <Record>
      <userdata><Resource employmentType="" externalId="5004043" includeInDatamart="1" isA
        isExternal="0" managerUserName="kinnunenad" resourceId="Res_material" resourceType
        username=""><PersonalInformation emailAddress="" firstName="" lastName="Res Mat"/
        <ManagementInformation availability="8" openForTimeEntry="1" trackMode="Clarity" use
        userFlag2="0"/> <CustomInformation/> <General/> <OBSAssocs completed="false"/> <
        <Calendar baseCalendar="Standard" resetCalendar="false"/> </Resource></userdata>
    </Record>
  - <Record>
      <userdata><Resource employmentType="EMPLOYEE" externalId="5003041" includeInDatam
        isActive="1" isExternal="0" managerUserName="kinnunenad" resourceId="enduser"
        resourceType="LABOR" username="enduser"><PersonalInformation emailAddress="jotak
        firstName="Ed" lastName="Ueberoth"/> <ManagementInformation availability="8"
        openForTimeEntry="0" trackMode="None" userFlag1="0" userFlag2="0"/> <CustomInforr
        <General/> <OBSAssocs completed="false"/> <SkillAssocs/> <Calendar baseCalendar="
        resetCalendar="false"/> </Resource></userdata>
    </Record>
  - <Record>
      <userdata><Resource employmentType="EMPLOYEE" externalId="5004041" includeInDatam
        isActive="1" isExternal="0" managerUserName="kinnunenad" resourceId="labor_res"
        resourceType="LABOR" username="labor_res"><PersonalInformation emailAddress="non
        firstName="Labor" lastName="Res"/> <ManagementInformation availability="8" openForT
        trackMode="Clarity" userFlag1="0" userFlag2="0"/> <CustomInformation/> <General/>
        completed="false"/> <SkillAssocs/> <Calendar baseCalendar="Standard" resetCalendar=
        </Resource></userdata>
    </Record>
  - <Record>
      <userdata><Resource employmentType="EMPLOYEE" externalId="5005049" includeInDatam
        isActive="1" isExternal="0" managerUserName="kinnunenad" resourceId="Administrator"
        resourceType="LABOR" username="Administrator"><PersonalInformation emailAddress="
        firstName="adm" lastName="Administrator Win"/> <ManagementInformation availability=
        openForTimeEntry="0" trackMode="None" userFlag1="0" userFlag2="0"/> <CustomInforr
        <General/> <OBSAssocs completed="false"/> <SkillAssocs/> <Calendar baseCalendar="
        resetCalendar="false"/> </Resource></userdata>
    </Record>
  </Records>
</QueryResult>
```

Editing

The editing method it is open the file in IE which set the line breaks and copy the content to notepad for editing and repeat the process until the XML file is in the proper format. FF displays and allows copying in a slightly different manner and is therefore less convenient

Items to be removed
From the beginning of the file

 <QueryResult xmlns="http://www.niku.com/xog/Query">
 <Code>zresquery</Code>
 <Records>

From the end of the file

 </Records>
 </QueryResult>

- all dashes

</Record>
<Record>
</userdata>
<userdata>

Then add standard header and footer

```
<?xml version="1.0" encoding="UTF-8"?>
<NikuDataBus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../xsd/nikuxog_resource.xsd">
   <Header action="write" externalSource="NIKU" objectType="resource" version="12.1.0.5849"/>
   <Resources>


   </Resources>
</NikuDataBus>
```

Always save as UTF-8

Final output

The final version of the file is proper format for XOG Object API and ready fro writing into another system.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<NikuDataBus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../xsd/nikuxog_resource.xsd">
 <Header action="write" externalSource="NIKU" objectType="resource" version="12.1.0.5849" />
<Resources>
<Resource employmentType="" externalId="5004042" includeInDatamart="1" isActive="1" isExternal="0" managerUserName="kinnunenad" resource
 <PersonalInformation emailAddress="" firstName="" lastName="Res Eq" />
 <ManagementInformation availability="8" openForTimeEntry="1" trackMode="Clarity" userFlag1="0" userFlag2="0" />
 <CustomInformation />
 <General />
 <OBSAssocs completed="false" />
 <SkillAssocs />
 <Calendar baseCalendar="Standard" resetCalendar="false" />
 </Resource>
<Resource employmentType="" externalId="5004043" includeInDatamart="1" isActive="1" isExternal="0" managerUserName="kinnunenad" resource
 <PersonalInformation emailAddress="" firstName="" lastName="Res Mat" />
 <ManagementInformation availability="8" openForTimeEntry="1" trackMode="Clarity" userFlag1="0" userFlag2="0" />
 <CustomInformation />
 <General />
 <OBSAssocs completed="false" />
 <SkillAssocs />
 <Calendar baseCalendar="Standard" resetCalendar="false" />
 </Resource>
<Resource employmentType="EMPLOYEE" externalId="5003041" includeInDatamart="1" isActive="1" isExternal="0" managerUserName="kinnunenad"
 <PersonalInformation emailAddress="jotakin" firstName="Ed" lastName="Ueberoth" />
 <ManagementInformation availability="8" openForTimeEntry="0" trackMode="None" userFlag1="0" userFlag2="0" />
 <CustomInformation />
 <General />
 <OBSAssocs completed="false" />
 <SkillAssocs />
 <Calendar baseCalendar="Standard" resetCalendar="false" />
 </Resource>
<Resource employmentType="EMPLOYEE" externalId="5004041" includeInDatamart="1" isActive="1" isExternal="0" managerUserName="kinnunenad"
 <PersonalInformation emailAddress="none" firstName="Labor" lastName="Res" />
 <ManagementInformation availability="8" openForTimeEntry="1" trackMode="Clarity" userFlag1="0" userFlag2="0" />
 <CustomInformation />
 <General />
 <OBSAssocs completed="false" />
 <SkillAssocs />
 <Calendar baseCalendar="Standard" resetCalendar="false" />
 </Resource>
<Resource employmentType="EMPLOYEE" externalId="5005049" includeInDatamart="1" isActive="1" isExternal="0" managerUserName="kinnunenad"
 <PersonalInformation emailAddress="jotakin" firstName="adm" lastName="Administrator Win" />
 <ManagementInformation availability="8" openForTimeEntry="0" trackMode="None" userFlag1="0" userFlag2="0" />
 <CustomInformation />
 <General />
 <OBSAssocs completed="false" />
 <SkillAssocs />
 <Calendar baseCalendar="Standard" resetCalendar="false" />
 </Resource>
 </Resources>
 </NikuDataBus>
```

REVISIONS

v.01 April 13 20011

The content of these pages is presented as personal views only and not as any sort of advice or instruction.