**Overview**

TridaBase, a Microsoft Office (Access) application was developed on 32 bit Windows/MS-Office by Epison Group, USA. The requirement of the current project was to make the application 64 bit compatible.

**Prelude**

It was found that there was a previous attempt to make this application 64 bit compatible but was abandoned since the replacement for the existing **Tree View ActiveX Control** (which is used in the main Navigation Screen) was not possible. **Microsoft doesn’t provide this Tree View control in its 64 bit Office versions.**

**After completion the application was tested on the following versions of MS-Access:**

1. **Access 2007**
2. **Access 2010**
3. **Access 2013**

**To make this application 32 and 64 bit compatible the following steps were taken:**

**Replace the existing Tree View ActiveX control:**

A Free Third Party Tree View Control Code from JKP ADS <http://www.jkp-ads.com> Build 025 is added/edited and implemented to replace the existing ActiveX Tree View Control

This JKP Ads Treeview calls 2 Class Modules

1. clsTreeView

2. clsNode

**Node Key Naming Logic followed as per old code:**

Root node key = ':P'

keys for projects = 'P' & projectID

key for Object Node under a project = "P" & projectID & ":O"

key for Object under Object node = "O" & objectID

etc.

**Add new events and event handler codes to new TreeView class**

1. Click

2. Double Click

3. Expand

4. Collapse

5. Key Down

**Add/Edit code in third party TreeView classes**

**Add new subs/functions TridaBase code**

**Edit Existing subs/functions TridaBase code**

**Make the existing code 64 bit compatible**

* Update Declare statements for api calls
* Update address pointers
* Update user defined types
* Use conditional compilation statements

**Update/Edit File Dialog box code**

**Update Change Data File feature**

**Update XML Import/Export**

**Update Backend data file link at start up**

**Handle MissingRequiredOption() in basUtility module**

**Fix Access version related bugs:**

In Access 2016 the XML Import/Export form was raising error: **“User Defined Type missing…”**

**Reason:**

The object variable **mxmlDoc** in class module **XmlExport** was defined as:

Private mxmlDoc As MSXML2.DOMDocument

Changed to:

Private mxmlDoc As MSXML2.DOMDocument**60**

When the "Create New Object" button in Project Form was clicked in Access 2007 it raised **Error:"3085 undefined function Nz() in expression"** but it worked without any error while ran in Access 2013.

**Reason:**

In Access 2007 the Nz() function when called from a query does not return 0 (as expected) when the evaluated argument is NULL.

The queries having the Nz() function called form sub setrowsource() in basutility module are the followings:

* **qlkpobjectparentobject**
* **qlkpDisplaymeasumentseriesvalues**
* **qlkpDisplayDerivedSeriesValues**

**Action taken:**

Replaced the **Nz()** function in the queries with **iif(isNull(<arg>),0,<arg>)**

Its working in Access 2007, 2010 and 2013

The old and edited queries are given below for record:

**qlkpobjectparentobject (Old) :**

SELECT qlkpDisplayObject.objectID, qlkpDisplayObject.Display, qlkpDisplayObject.identifier, qlkpDisplayObject.title FROM qlkpDisplayObject WHERE (((qlkpDisplayObject.objectID)<>Nz([Forms]![frmObject]![objectID]))) ORDER BY qlkpDisplayObject.SortOrder;

**qlkpobjectparentobject (Edited) :**

SELECT qlkpDisplayObject.objectID, qlkpDisplayObject.Display, qlkpDisplayObject.identifier, qlkpDisplayObject.title FROM qlkpDisplayObject WHERE (((qlkpDisplayObject.objectID)<>iif(isNull([Forms]![frmObject]![objectID]),0,[Forms]![frmObject]![objectID]))) ORDER BY qlkpDisplayObject.SortOrder;

**qlkpDisplaymeasumentseriesvalues (Old)**

SELECT tblMeasurementSeriesValues.measurementSeriesValuesID,

Nz([qlkpControlledVocInDefaultLanguage\_Variable]![TitleAndStandard],"") &

[variable] & " - " & IIf([unitless],"Unitless","") &

Nz([qlkpControlledVocInDefaultLanguage\_Unit]![TitleAndStandard],"") &

[unit] AS Display

FROM (tlkpValuesVariable LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Variable ON tlkpValuesVariable.controlledVocID = qlkpControlledVocInDefaultLanguage\_Variable.ControlledVocID) RIGHT JOIN ((tlkpUnit LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Unit ON tlkpUnit.controlledVocID = qlkpControlledVocInDefaultLanguage\_Unit.ControlledVocID) RIGHT JOIN tblMeasurementSeriesValues ON tlkpUnit.controlledVocID = tblMeasurementSeriesValues.unitID) ON tlkpValuesVariable.controlledVocID = tblMeasurementSeriesValues.variableID;

**qlkpDisplaymeasumentseriesvalues (Edited)**

SELECT tblMeasurementSeriesValues.measurementSeriesValuesID,

iif(isNull([qlkpControlledVocInDefaultLanguage\_Variable]![TitleAndStandard]),"",[qlkpControlledVocInDefaultLanguage\_Variable]![TitleAndStandard]) &

[variable] & " - " & IIf([unitless],"Unitless","") &

iif(isNull([qlkpControlledVocInDefaultLanguage\_Unit]![TitleAndStandard]),"",[qlkpControlledVocInDefaultLanguage\_Unit]![TitleAndStandard])

&

[unit] AS Display

FROM (tlkpValuesVariable LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Variable ON tlkpValuesVariable.controlledVocID = qlkpControlledVocInDefaultLanguage\_Variable.ControlledVocID) RIGHT JOIN ((tlkpUnit LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Unit ON tlkpUnit.controlledVocID = qlkpControlledVocInDefaultLanguage\_Unit.ControlledVocID) RIGHT JOIN tblMeasurementSeriesValues ON tlkpUnit.controlledVocID = tblMeasurementSeriesValues.unitID) ON tlkpValuesVariable.controlledVocID = tblMeasurementSeriesValues.variableID;

**qlkpDisplayDerivedSeriesValues (Old)**

SELECT tblDerivedSeriesValues.derivedSeriesValuesID, Nz([qlkpControlledVocInDefaultLanguage\_Variable]![TitleAndStandard],"") & [variable] & " - " & IIf([unitless],"Unitless","") & Nz([qlkpControlledVocInDefaultLanguage\_Unit]![TitleAndStandard],"") & [unit] AS Display

FROM (tlkpValuesVariable LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Variable ON tlkpValuesVariable.controlledVocID = qlkpControlledVocInDefaultLanguage\_Variable.ControlledVocID) RIGHT JOIN ((tlkpUnit LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Unit ON tlkpUnit.controlledVocID = qlkpControlledVocInDefaultLanguage\_Unit.ControlledVocID) RIGHT JOIN tblDerivedSeriesValues ON tlkpUnit.controlledVocID = tblDerivedSeriesValues.unitID) ON tlkpValuesVariable.controlledVocID = tblDerivedSeriesValues.variableID;

**qlkpDisplayDerivedSeriesValues (Edited)**

SELECT tblDerivedSeriesValues.derivedSeriesValuesID,

iif(isNull([qlkpControlledVocInDefaultLanguage\_Variable]![TitleAndStandard]),"",[qlkpControlledVocInDefaultLanguage\_Variable]![TitleAndStandard])

&

[variable] & " - " & IIf([unitless],"Unitless","") &

iif(isNull([qlkpControlledVocInDefaultLanguage\_Unit]![TitleAndStandard]),"",[qlkpControlledVocInDefaultLanguage\_Unit]![TitleAndStandard]) & [unit] AS Display

FROM (tlkpValuesVariable LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Variable ON tlkpValuesVariable.controlledVocID = qlkpControlledVocInDefaultLanguage\_Variable.ControlledVocID) RIGHT JOIN ((tlkpUnit LEFT JOIN qlkpControlledVocInDefaultLanguage AS qlkpControlledVocInDefaultLanguage\_Unit ON tlkpUnit.controlledVocID = qlkpControlledVocInDefaultLanguage\_Unit.ControlledVocID) RIGHT JOIN tblDerivedSeriesValues ON tlkpUnit.controlledVocID = tblDerivedSeriesValues.unitID) ON tlkpValuesVariable.controlledVocID = tblDerivedSeriesValues.variableID;

**Update code in class formresize to be 64 bit compatible**

**Clean up third party unnecessary codes except the following form control related codes:**

* cboEntity
* cboId
* cmdNewProject
* cmdRefresh
* cmdOptions
* cmdDelete
* cmdXmlImportExport
* cmdCloseForms
* lblFunction12Message
* cmdClose

**Populate tree with Project data at form load to speed up performance and show some message while projects load**

**Change Application/Version**

* The Function DataFileVersion() of module basUtility looks up the current data file version data from table tblVersionHistory
* Inserted new version (136) record in tblVersionHistoryApplication with comments
* Added check in frmBackEndUpdate. If version <136 call new added function Update135To136() and set refresh controlvocabularycache and set MinimumProgramVersion =136 in table TblOption
* TblOptionApplication: Set value of MinimumBackEndVersion = 136