
The StaticVars Class Library

StaticVars is an Integra runtime object class designed to store static information throughout the execution of an Integra profile. The actual object at runtime is called **stats**.

Though the class has the same name and has the same structure for the three Integra profile types (Word, Excel and Text), the availability (usability) of the properties and methods differ.

Integra's Advanced tab allows definition of event driven Callback script. The runtime object **stats** can be used within the Callback Script or within an Add-in Script Library, thus giving control over Integra's behaviour during its runtime events.

Below is a list of all the properties and methods of the StaticVars class. They are explained in detail throughout this section and availability will be indicated.

Note: ALL properties allow Read/Write access, however, it is indicated if Read Only is Recommended. For the latter you should not modify the value of the properties unless you are very confident in the use of Integra's runtime object classes.

1 StaticVars.AB_Doc [Class Property]

Version: 2.x
Availability: Word, Excel, Text, Export

Note: this property is not more supported since version 3.0.

Notes back-end document that contains the action buttons as selected in the Integra profile.

This property is not more in use and is kept only for compatibility reasons to Integra for Notes Release 2. The Action Buttons delivered with Integra for Notes Release 2 provided some Inplace-Editing capabilities, which could be attached together with an Excel spreadsheet or Word document send via email or saved in a Notes document.

Integra for Notes 3.0 contains new design elements for Inplace-Editing that can be inherited into existing development (Notes databases); replacing the Action Buttons functionality.

Please refer to the Inplace-Editing documentation for further details.

2 StaticVars.Action [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

Runtime COM action as selected on the Integra profile or by the user in the runtime dialogs (Prompt User option). Possible actions values are 'Activate', 'Print', 'Print/Preview', 'Save To File' and 'None'.

3 StaticVars.ActionOnly [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

The processing of an Integra export profile consists basically of four steps:

- 1) The user selects the profile and eventually determines runtime options
- 2) Integra is extracting the data from Notes and pushes them into Word or Excel
- 3) Depending on profile settings the user is presented with the output document
- 4) After the user closes the output document, Integra further process the document (e.g. sending as mail)

With Integra releases prior to 3.0 the Notes client was frozen for the time Word documents or Excel spreadsheets created by Integra where activated on screen (between step 3 and 4). With Integra 3.0 this limitation has been removed.

The ActionOnly property indicates whether Integra is running before the activation of Word or Excel or after the user closed the activated Word or Excel application.

4 StaticVars.AddCategory [Class Method (Function)]

Version: 2.x to 4.7
Availability: Word, Excel, Export

StaticVars.AddCategory (CatFormula As String)

The AddCategory is called for every exported Notes document. It will evaluate the formula as defined in the Category Formula field of the Integra profile. It converts and adds the resulting value to an array of category strings ().

Depending on the type of profile this array of categories is used to create a Category field in the Notes document or response document in which a Word or Excel document is saved back into.

In case of a the Save As Response or Attach To Document options the new Category field will contain only the one category value evaluated for the parent document, otherwise it will contain the entire list of category values stored as multi-value.

Parameters:

CatFormula (String)

The formula as defined in the Category field in the Integra profile. However, any valid Notes Formula will do. The formula will be evaluated in the context of the exported Notes document.

Return:

True if the evaluation was successfull, otherwise False.

5 StaticVars.AftWritMacr [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Export

After Write Macro.

Name of a Word or Excel macro to be executed after each Notes document is exported to Word or Excel. The execution of the macro follows the CB_AFTERWRITECOM event. This property is exposed to the profile creator in the advanced tab of the Integra profile. However, the name of the macro can be changed dynamically at runtime.

With Integra 4.1 two new Excel export methods have been introduced, namely Normal (using CSV) and XML. These two methods avoid the use of COM technology and are therefore much faster. However, this performance improvement comes with a reduced flexibility or usability when it comes to the Integra events. The main difference between the use of COM technology and the new CSV or XML export is that the latter do not write every single exported Notes document into rows in Excel, but rather collect all rows in either CSV format or XML format (MS Office 2003+). Once all Notes documents are exported to CSV or XML, the CSV or XML file is imported into the Excel spreadsheet. The consequence of this is that the CB_AFTERWRITECOM event becomes very limited in its use, thus the After Write Macro useless in this context.

6 StaticVars.AllIfNone [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export

When using the Integra profile option "Selected Documents In Active View" to let the user select the documents to be exported, the AllIfNone property of the global Integra runtime object stats controls whether all documents in the view are exported in case the user selects none or if the export will be aborted.

This property is exposed to the profile creator for all releases prior to 4.3 in the advanced tab of the Integra profile. Since Integra 4.3 this option is found in the Filtering & Sorting->Data Filtering tab.

Set default for Integra Quick Reports is False.

7 StaticVars.ApiExpRT [Class Property]

Version: 4.0 to 4.7
Availability: Word, Export

Notes API driven export for Notes rich text fields.

In all versions of Integra prior to 4.0, Notes rich text is exported using the Notes Client user interface. The method used is basically to open each rich text field with a special form, to select all rich text content and to copy it into the Windows Clipboard. The Clipboard content is then pasted into the Word document at the defined place.

This method has two side effects: flickering of the screen because of the opening and closing of UI documents and a very low performance for that exact same reason.

With Integra 4.0 a new Notes API driven method is used. The Notes API provides some undocumented functionality to export Notes rich text to a rich text file. This new method works much faster and also avoids the UI as the vehicle of getting rich text out of Lotus Notes. The new API export is the default method.

Sometimes, the fidelity of both methods differs, thus the old UI driven export might give better results. The example code showing below explains how to switch back to the old UI driven rich text export.

Example:
[UI driven rich text export](#)

8 StaticVars.ApilmpRT [Class Property]

Version: 4.0 to 4.7
Availability: Word, Export

Notes API driven import of Word document content into Notes rich text fields.

In all versions of Integra prior to 4.1, Word document content was imported into Notes rich text fields using the Notes Client user interface. The method used is basically to copy the Word document content into the Windows Clipboard, to open a rich text field with a special form and to past the Clipboard content into the rich text field. The temporary Notes document is then further processed to get the rich text into the target Notes document.

This method has two side effects: flickering of the screen because of the opening and closing of UI documents and a very low performance for that exact same reason.

With Integra 4.1 a new Notes API driven method is used. The Notes API provides some undocumented functionality to import Word rich text into a Notes rich text field. This new method works much faster and also avoids the UI as the vehicle of getting rich text into Lotus Notes. The new API import is the default method.

Sometimes, the fidelity of both methods differs, thus the old UI driven import might give better results. The example code showing below explains how to switch back to the old UI driven rich text import.

Example:
[UI driven rich text import](#)

9 StaticVars.AttName [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

The AttName property contains the formula string evaluated at runtime to produce a computed filename for the Word documents, Excel spreadsheets or textfiles to be attached to automated email or Notes documents (Integra profile options SendTo or Save As...).

The string has to evaluate to a valid filename without file-extension and without a file-path. The formula is usually pre-defined in the Integra profile in the Attachment Name-Prefix Formula field, but can be changed at runtime. Note: Integra assumes AttName to contain a formula; a simple string has to be in quotes.

Example:
[Dynamically changing the attachment name prefix for Save As / Send To](#)

10 StaticVars.AutoColWdth [Class Property]

Version: 4.0 to 4.7
Availability: Excel, Export

The AutoColWdth property controls whether the Excel spreadsheet is auto formatted once the export is complete with regards to the column width. If set to True the Excel spreadsheet is formatted, otherwise not.

This property is exposed to the profile creator for all releases prior to 4.3 in the advanced tab of the Integra profile. Since Integra 4.3 this option is found in the Excel tab. The option is also available in Quick Reports.

11 StaticVars.AutoOvrWrit [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export

If using the Filename field in the Integra profile to pre-determine the filename for the Word, Excel or Text filename to be created, the AutoOvrWrit property controls whether an existing file with the same filename is automatically overwritten or if the user is being prompted to confirm. If the profile is scheduled or used from a browser, the property is ignored and the file is always overwritten.

This property is exposed to the profile creator in the advanced tab of the Integra profile.

12 StaticVars.AutoRowHght [Class Property]

Version: 4.0 to 4.7
Availability: Excel, Export

The AutoRowHght property controls whether the Excel spreadsheet is auto formatted once the export is complete with regards to the row height. If set to True the Excel spreadsheet is formatted, otherwise not.

This property is exposed to the profile creator for all releases prior to 4.3 in the advanced tab of the Integra profile. Since Integra 4.3 this option is found in the Excel tab. The option is also available in Quick Reports.

13 StaticVars.Buttons [Class Property]

Version: 2.x
Availability: Word, Excel, Export

Note: this property is not more supported since version 3.0.

Contains a NotesRichTextItem object storing the Action Buttons selected in the Integra profile.

This property is not more in use and is kept only for compatibility reasons to Integra for Notes Release 2. The Action Buttons delivered with Integra for Notes Release 2 provided some Inplace-Editing capabilities, which could be attached together with an Excel spreadsheet or Word document send via email or saved in a Notes document.

Integra for Notes 3.0 contains new design elements for Inplace-Editing that can be inherited into existing development (Notes databases); replacing the Action Buttons functionality.

Please refer to the Inplace-Editing documentation for further details.

14 StaticVars.CatCount [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

When exporting selected documents from the currently active view, Integra will evaluate any category formula as defined in the executed Integra profile for each exported Notes document. CatCount contains the number of category elements created:

`stats.CatCount = ubound(stats.category) + 1`

Note: using the Category Formula field is limited to smaller data sets. The maximum number of elements stats.Categories depends on the Lotus Notes limit for array dimensions and last but not least the available memory.

15 StaticVars.Category [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

When exporting selected documents from the currently active view, Integra will evaluate the category formula stored in the Integra profile for each exported Notes document. Category[] is an array that contains the strings resulting from the evaluation of the category formula for each of the exported documents.

If one of the Save As... options is selected in the Integra profile, the category string will be created in a field Category in the Notes document together with the attached Word, Excel or Text file.

Note: using the Category Formula field is limited to smaller data sets. The maximum number of elements stats.Categories depends on the Lotus Notes limit for array dimensions and last but not least the available memory.

Example:
[Insert Excel row on value change of Category Formula](#)

16 StaticVars.CatStr [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

Integra has two ways of exporting data from a Notes database. The first one is giving the user the ability to select documents in a view and the second option is to export all documents within a pre-selected view (or folder). The CatStr() property is used only for the first option.

The CatStr[] property contains an array of combined strings per categorised column values for each of the documents in the document collection (selected documents in active view). Each category is separated with a "\" (back-slash). Integra has to evaluate the column formulas for each selected document per categorised column to build this formatted string of categories.

The CatStr[] property is used to identify a change in category from one exported document to the next exported document. Integra is using this property for only one particular standard profile (which is delivered with the Integra template) called "Export Current View To Excel".

The CatStr[] property is related to StaticVars.KeyArr() property which contains the sorting string to re-establish the original sorting of the view (please refer to the StaticVars.KeyArr[] property).

Example:
[Insert Excel row on value change of category value in stats.CatStr\[\]](#)

17 StaticVars.CbStatus [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

Integra uses this property to set the current event (name) before it is executing the Callback Formulas or Scripts stored in the Integra profile.

Example:
[Default advanced script structure explained](#)

18 StaticVars.ColCat [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The ColCat[] property is a dynamic array of type Integer. The upper bound of the array is set according to the number of sorted columns in the active view. The value stored for each sorted column is either True (column is categorised) or False (column is not categorised). E.g. if the currently active view has 5 columns and column 1 is sorted ascending and categorised, column 3 & 5 are sorted, but not categorised, then the upper bound of ColCat[] is 2 (3 elements) and the array element values are:

```
ColCat(0) = True  
ColCat(1) = False  
ColCat(2) = False
```

The property is only relevant in the context of exporting "Selected Documents in Active View".

19 StaticVars.ColFrm [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The ColFrm[] property is a dynamic array of type String. The upper bound of the array is set according to the number of sorted columns in the active view. The value stored for each sorted column is either the fieldname or the Notes formula for the particular column. E.g. if the currently active view has 5 columns and column 1, 3 & 5 are sorted, then the upper bound of ColFrm[] is 2 (3 elements) and the array element values could be:

```
ColFrm(0) = "Name"  
ColFrm(1) = "Surname"  
ColFrm(2) = | Surname + ", " + Name |
```

20 StaticVars.ColNum [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The ColNum[] property is a dynamic array of type Integer. The upper bound of the array is set according to the number of sorted columns in the active view. The value stored for each sorted column is the effective column number in the NotesView.Columns array. E.g. if the currently active view has 5 columns and column 1, 3 & 5 are sorted, then the upper bound of ColNum[] is 2 (3 elements) and the array element values are:

ColNum(0) = 1
ColNum(1) = 3
ColNum(2) = 5

The property is only relevant in the context of exporting "Selected Documents in Active View".

21 StaticVars.ColRes [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The ColRes[] property is a dynamic array of type Integer. The upper bound of the array is set according to the number of sorted columns in the active view. The value stored for each sorted column is either True (the column shows response documents only) or False. E.g. if the currently active view has 5 columns and column 1, 3 & 5 are sorted, but non of them shows response documents only, then the upper bound of ColRes[] is 2 (3 elements) and the array element values are:

ColRes(0) = False
ColRes(1) = False
ColRes(2) = False

The property is only relevant in the context of exporting "Selected Documents in Active View".

22 StaticVars.ColSrt [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The ColSrt[] property is a dynamic array of type Integer. The upper bound of the array is set according to the number of sorted columns in the active view. The value stored for each sorted column is either False (column is sorted ascending) or True (column is sorted descending). E.g. if the currently active view has 5 columns and column 1 is sorted ascending, column 3 is sorted ascending and column 5 is sorted descending, then the upper bound of ColSrt[] is 2 (3 elements) and the array element values are:

ColSrt(0) = False
ColSrt(1) = False
ColSrt(2) = True

The property is only relevant in the context of exporting "Selected Documents in Active View".

23 StaticVars.Copies [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

For Integra export profiles using the Print option, the user is presented with a Word or Excel Print dialog. The Prntr property contains the name of the in the dialog selected printer.

24 StaticVars.CopyButtonSet [Class Method (Procedure)]

Version: 2.x
Availability: Word, Excel, Text, Export

CopyButtonSet (buttonset As String, target_fd As String, target_doc As notesdocument, ritem_trg As Variant)

Note: this method is not more supported since version 3.0.

Internal method called to copy the in the profile selected button set (see also) into any document in which the profile is set to attach the Excel, Word or Text output file.

This method is not more in use and is kept only for compatibility reasons to Integra for Notes Release 2. The Action Buttons delivered with Integra for Notes Release 2 provided some Inplace-Editing capabilities, which could be attached together with an Excel spreadsheet or Word document send via email or saved in a Notes document.

Integra for Notes 3.0 contains new design elements for Inplace-Editing that can be inherited into existing development (Notes databases); replacing the Action Buttons functionality. Please refer to the Inplace-Editing documentation for further details.

Parameters:

buttonset (String)

Name of the action button set as selected in the Integra profile (or any other available button set for that matter)

target_fd (String)

Name of the Notes rich text field in which to copy the button set

target_doc (NotesDocument)

Back-end handle to the Notes document containing the (above) rich text field

rtitem_trg (Variant)

Handle to the NotesRichTextItem in which to copy the button set

25 StaticVars.CurCatLevel [Class Property]

Version: 4.3 to 4.7
Availability: Excel, Export

New to Integra for Notes 4.3

CurCatLevel represents the category or sub-category level under which the currently exported document is shown under. The top level is represented by the value 0. The property is only available when exporting all documents in a (active) view using options that will let Integra choose to use the NotesViewNavigator class.

The conditions are the following (evaluated at runtime in order of presentation):

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	CurCatValue Available
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes	No
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	Yes
View is NOT categorised	NotesViewEntryCollection	n/a	No
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	No
stats.dosort = False	NotesViewEntryCollection	Yes	No
stats.flatcat = True	NotesViewEntryCollection	Yes	No
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	No
Otherwise	NotesViewNavigator	n/a	Yes

26 StaticVars.CurCatValue [Class Property]

Version: 4.3 to 4.7
Availability: Excel, Export

New to Integra for Notes 4.3

CurCatValue represents the value of the category or sub-category value under which the currently exported document is shown under. The property is only available when exporting all documents in a (active) view using options that will let Integra choose to use the NotesViewNavigator class.

The conditions are the following (evaluated at runtime in order of presentation):

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	CurCatValue Available
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	Yes
View is NOT categorised	NotesViewEntryCollection	n/a	No
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	No
stats.dosort = False	NotesViewEntryCollection	Yes	No
stats.flatcat = True	NotesViewEntryCollection	Yes	No
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	No
Otherwise	NotesViewNavigator	n/a	Yes

27 StaticVars.CurLabel [Class Property]

Version: 2.x to 4.7
Availability: Word, Export

This property is only used for Word export profiles of type Label. It stores the label number in which the currently processed Notes document is exported into. The value can be any number between 1 and the number of labels per page (as defined in the Integra profile)

The property is only relevant in the context Word Label profiles using Advanced COM as the method of export.

28 StaticVars.CurRow [Class Property]

Version: 2.x to 4.7
Availability: Excel, Export

The property stats.CurRow contains the current row number in the Excel spreadsheet into which Integra is writing the data of the currently exported Notes document or for import profiles the current row from which Integra is reading the data for the currently imported Notes document. Although defined as long, the valid value range is between 1 to 65535 (max number of rows in an Excel spreadsheet).

The property is only relevant in the context of exporting using the Advanced COM method.

29 StaticVars.CurViewNam [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export

The CurViewNam property reflects the view's name or alias as selected in the Integra profile if the profile is linked to a predefined view, otherwise it is the name or alias of the currently active view from which Integra has been initiated from.

30 StaticVars.DateText [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

At runtime Integra will set the DateText property to the current date-time value converted to a string data type.

In case of an export profile, Integra will write this value into a pre-defined Word field or Excel tag called `_DocumentDate`. When creating a new Integra export profile and editing the attached Word or Excel template, Integra can optionally add the Word fields or Excel tags to the template. When using this option, Integra will create a field or tag called `_DocumentDate`.

In Integra 2.x the Prompt User dialog (export only) actually included a field to allow the user to modify this date-time value. However, in Integra 3.0 this field has been removed from the Prompt User dialog, instead the profile creator can add some Lotus Script to the Script Callback field in the Integra profile to allow a user to set a different date-time before the Word or Excel document is created.

Example:
[Use of Inputbox\(\) to change stats.DateText](#)

31 StaticVars.Delete [Class Method (Procedure)]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

Default object class method called once Integra completes its processing cycle. The method calls depending on the profile type one of the following three methods:

[StaticVars.DeleteWord](#)
[StaticVars.DeleteText](#)

Each of the respective methods clears any other object assigned to any of the StaticVars properties of type object, closes any temporary files and removes the Word or Excel COM objects from memory (closing any eventually still open MS Office files).

32 StaticVars.DeleteText [Class Method (Procedure)]

Version: 2.x to 4.7
Availability: Text, Export, Import

This method is called from the default method. It clears any other object assigned to any of the StaticVars properties of type object and closes any temporary or output files.

33 StaticVars.DeleteWord [Class Method (Procedure)]

Version: 2.x to 4.7
Availability: Word, Export, Import

This method is called from the default method. It clears any other object assigned to any of the StaticVars properties of type object, closes any temporary files and calls the [StaticVars.QuitWord](#) method to remove the Word COM objects from memory (closing any eventually still open Word files).

34 StaticVars.DeleteXls [Class Method (Procedure)]

Version: 2.x to 4.7
Availability: Excel, Export, Import

This method is called from the default method. It clears any other object assigned to any of the StaticVars properties of type object, closes any temporary files and removes the Excel COM objects from memory (closing eventual any still open Excel files).

The method re-sets all Excel environment variables to its former state if changes where required for Integra's processing.

35 StaticVars.DelHidden [Class Property]

Version: 4.1 to 4.7
Availability: Excel, Export

With **Integra for Notes 3.0** a new profile has been introduced called Export Current View To Excel. This profile allows a user to export data from any view in any Notes database to an Excel spreadsheet without creating a custom Integra profile. The profile would simply export all columns in the view to an Excel spreadsheet.

Naturally views can contain hidden columns and views and a requirement exist to control whether those columns should be exported or not.

In the Integra versions 3 and 4 the profile contains an advanced callback script function call that had a parameter to control the behaviour (fourth parameter):

[Call ViewToExcelCallBack\(stats,export,true,true,false,false\)](#)

By default the hidden columns will not be exported. One can customise the behaviour by either changing the default parameter or by creating a copy of the profile with a different parameter setting.

With **Integra for Notes 4.1** the Export Current View To Excel profile has been superceeded by the Quick Reports Wizard which does mainly the same thing, but with some more user controlled features. The profile has subsequently been removed from the Integra template. The existing Export Current View To Excel profile will not be deleted by an upgrade to Integra for Notes 4.1, however, the profile will stop functioning.

Nevertheless, the original functionality is still supported and is now simply build into the core of Integra's run time libraries. To create a profile that would do exactly what the original Export Current View To Excel profile does one can follow these steps:

- Create a new Integra Excel export profile
- Select the for All Databases option in the Database tab
- Do not add any computed fields and do not create an Excel template by clicking the Edit Template button
- Enable the Format Excel output according to selected view (column) formatting
- Other options may be enabled or disabled as they seem fit.

The DelHidden property of the stats object has been introduced to allow the previously parameterised control on whether hidden columns are exported or not using the stats object in e.g. the CB_INITIALISE event of the advanced Callback Script.

36 StaticVars.Dellcon [Class Property]

Version: 4.1 to 4.7
Availability: Excel, Export

With **Integra for Notes 3.0** a new profile has been introduced called Export Current View To Excel. This profile allows a user to export data from any view in any Notes database to an Excel spreadsheet without creating a custom Integra profile. The profile would simply export all columns in the view to an Excel spreadsheet.

Naturally views can contain columns to display icons and a requirement exist to control whether those columns should be exported or not.

In the Integra versions 3 and 4 the profile contains an advanced callback script function call that had a parameter to control the behaviour (third parameter):

[Call ViewToExcelCallBack\(stats,export,true,true,false,false\)](#)

By default the columns displaying icons will not be exported. One can customise the behaviour by either changing the default parameter or by creating a copy of the profile with a different parameter setting.

With **Integra for Notes 4.1** the Export Current View To Excel profile has been superceeded by the Quick Reports Wizard which does mainly the same thing, but with some more user controlled features. The profile has subsequently been removed from the Integra template. The existing Export Current View To Excel profile will not be deleted by an upgrade to Integra for Notes 4.1, however, the profile will stop functioning.

Nevertheless, the original functionality is still supported and is now simply build into the core of Integra's run time libraries. To create a profile that would do exactly what the original Export Current View To Excel profile does one can follow these steps:

- Create a new Integra Excel export profile
- Select the for All Databases option in the Database tab
- Do not add any computed fields and do not create an Excel template by clicking the Edit Template button
- Enable the Format Excel output according to selected view (column) formatting
- Other options may be enabled or disabled as they seem fit.

The Dellcon property of the stats object has been introduced to allow the previously parameterised control on whether columns displaying icons are exported or not using the stats object in e.g. the CB_INITIALISE event of the advanced Callback Script.

37 StaticVars.DocCnt [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

The DocCnt property contains the number of actually exported documents for Word and Excel exports. DocCnt does not necessarily represent to the number of documents selected in the Notes view selected for the export, as one or more documents could be ignored due to filter conditions (e.g. Selection Formula field in Integra profile).

Example:
[Notify user of the total number of documents exported](#)

38 StaticVars.DocCol [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export

The stats.DocCol property contains the list of to be processed (exported) Notes documents.

Integra for Notes utilises all Notes classes that allow processing of documents in a view (NotesDocumentCollection, NotesView, NotesViewEntryCollection and NotesViewNavigator). Which class is used depends on profile settings and on the context the user is in when calling the Integra profile.

The condition for the use of the NotesDocumentCollection class (stats.DocCol) is that...

- the (export) profile is set "Selected Documents in Active View" and
- one or more (**but not all**) documents are selected (tick-marked) in the view

If ALL documents are tick-marked, or if NO documents are tick-marked and the profile option *Export ALL Notes documents if NONE are selected is enabled*, in both cases Integra will switch to one of the other more efficient classes.

The condition for the use of the respective classes is outlined in the table below...

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	stats.Entr y available
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes	No
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	Yes
View is NOT categorised	NotesViewEntryCollection	n/a	Yes
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	Yes
stats.DoSort = False	NotesViewEntryCollection	Yes	Yes
stats.FlatCat = True	NotesViewEntryCollection	Yes	Yes
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	Yes
Otherwise	NotesViewNavigator	n/a	Yes

The difference in the use of these Notes object classes lies in the performance and in available features. The following table shows the differences:

Class	Performance	Usage
NotesDocumentCollection	Slowest compared to other classes.	If users require to select documents manually in the view. Not suitable for large amounts of to be exported documents.
NotesViewEntryCollection	Fastest class to process documents in the view.	Used for large data collection, where ONLY document entries are required to be processed.
NotesViewNavigator	Performance depends on other settings and the design of the view. It is generally faster than the NotesDocumentCollection, however, performance can be reduced when handling view category and subtotal entries	This class needs to be used when it is required to handle the view category and subtotal entries.

Some combinations of settings might create a situation where no documents can be exported, e.g. if a full text filter or selection formula is used in addition to the main selection method. This would be determined at runtime and the user will be notified.

39 StaticVars.DoSort [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

Integra has two ways of exporting data from a Notes database. The first one is giving the user the ability to select documents in a view and the second option is to export all documents within a pre-selected view (or folder). With regards to sorting, the latter is not a problem. This as Integra can utilise the view's sorting to export data in the correct order and therefore the KeyArr() property is not used for pre-selected view exports.

To re-establish the original sorting of the view, Integra has to evaluate the column formulas for each selected document per sorted column and to build a single unicode string to reflect the very unique Notes sorting. It gets even more complex as a view might contain a mix of ascending and descending sorted columns. To achieve a proper sorting of this single string per document, descending column values have to be "negated". In simple terms, an "A" becomes a "Z", a "B" becomes a "Y" and so on. "Simple terms" in the sense that Notes (as mentioned earlier) is using unicode character sets and therefore a negated "A" is in matter of fact not a "Z". Besides that a unicode character sets include more than the 256 characters of the standard ASCII character set.

As a result of the very complex sorting procedure Integra is sorting a collection of documents precisely as listed in a Notes view. The sorting method itself is actually a QuickSort algorithm. Having explained this on a very high level, it becomes immediately apparent that the number of documents and the complexity in the view design has a huge impact on the performance for the "Selected Documents in Active View" option.

For exports of larger volumns of Notes documents where users are required to select documents in the active view, it is recommended to automate the process of moving selected documents in a folder and to run the Integra profile on this folder. This as Notes view and folder indexing is certainly much faster than Integra's sorting algorithm will ever be.

However, in some cases sorting might not even be an issue or it is feasible or preferred to let Excel do the sorting and grouping - Excel has very powerful sorting and grouping functionality. The Integra profile creator can use the Script Callback field to add some Lotus Script to set the DoSort property to False. In this case Integra will not sort the document collection, but export the documents in the sequence it received the collection.

The property is only relevant in the context of exporting "Selected Documents in Active View".

Example:

[Disable Integra's sorting for selected documents](#)

40 StaticVars.Entry [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export

The stats.Entry property points to the currently processed view entry (from which the to be exported Notes document derives - stats.ExpDoc).

Integra for Notes utilises all Notes classes that allow processing of documents in a view (NotesDocumentCollection, NotesView, NotesViewEntryCollection and NotesViewNavigator). Which class is used depends on profile settings and on the context the user is in when calling the Integra profile.

The condition for the use of the respective classes is outlined in the table below...

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	stats.Entr y available
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes	No
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	Yes
View is NOT categorised	NotesViewEntryCollection	n/a	Yes
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	Yes
stats.DoSort = False	NotesViewEntryCollection	Yes	Yes
stats.FlatCat = True	NotesViewEntryCollection	Yes	Yes
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	Yes
Otherwise	NotesViewNavigator	n/a	Yes

Some combinations of settings might create a situation where no documents can be exported, e.g. if a full text filter or selection formula is used in addition to the main selection method. This would be determined at runtime and the user will be notified.

41 StaticVars.EntryCol [Class Property]

Version: 4.2 to 4.7
Availability: Word, Excel, Text, Export

The stats.EntryCol property contains the list of to be processed (exported) Notes documents.

Integra for Notes utilises all Notes classes that allow processing of documents in a view (NotesDocumentCollection, NotesView, NotesViewEntryCollection and NotesViewNavigator). Which class is used depends on profile settings and on the context the user is in when calling the Integra profile.

The condition for the use of the respective classes is outlined in the table below...

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	stats.Entr y available
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes	No
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	Yes
View is NOT categorised	NotesViewEntryCollection	n/a	Yes
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	Yes
stats.DoSort = False	NotesViewEntryCollection	Yes	Yes
stats.FlatCat = True	NotesViewEntryCollection	Yes	Yes
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	Yes
Otherwise	NotesViewNavigator	n/a	Yes

The difference in the use of these Notes object classes lies in the performance and in available features. The following table shows the differences:

Class	Performance	Usage
NotesDocumentCollection	Slowest compared to other classes.	If users require to select documents manually in the view. Not suitable for large amounts of to be exported documents.
NotesViewEntryCollection	Fastest class to process documents in the view.	Used for large data collection, where ONLY document entries are required to be processed.
NotesViewNavigator	Performance depends on other settings and the design of the view. It is generally faster than the NotesDocumentCollection, however, performance can be reduced when handling view category and subtotal entries	This class needs to be used when it is required to handle the view category and subtotal entries.

Some combinations of settings might create a situation where no documents can be exported, e.g. if a full text filter or selection formula is used in addition to the main selection method. This would be determined at runtime and the user will be notified.

42 StaticVars.ExpDoc [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

For export profiles this property contains the handle to the currently processed Notes document object (back-end class). You can use the document handle to e.g. create temporary fields, to query fields for filter conditions or to add a date-time stamp on the exported document as a means to log a group of documents of a particular export.

Example:
[Coded filter condition & time-stamp for exported documents](#)

43 StaticVars.ExpDocUNID [Class Property]

Version: 4.3 to 4.7
Availability: Word, Excel, Text, Export

For export profiles this property contains the document's unique id (UNID) of the currently processed Notes document. The main reason for this property is to preserve a "link" to the last exported document once the main processing loop is completed (after which the stats.ExpDoc property is set to Nothing).

Integra uses this property to re-assign stats.ExpDoc when Integra's processing resumes after an activated Word or Excel document has been closed by the user.

44 StaticVars.ExpMaxChr [Class Property]

Version: 4.3 to 4.7
Availability: Excel, Export

Microsoft Excel has limitations with regards to the number of characters that can be exported into a single cell. The limit depends on the Excel version. In older version this limit was set at 1024 characters. More recent version (XP, 2003) have a higher limit (16384+). The default value of ExpMaxChr is set to 16384.

Previous releases of Integra for Notes did export a fixed maximum of 768 or 1024 characters per Excel cell.

45 StaticVars.ExpType [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export

The ExpType property contains either "WORD", "XLS" or "TEXT", depending on the type of profile executed.

46 StaticVars.FdAdds [Class Property]

Version: 2.x to 4.2
Availability: Word, Export

Note: this property and the described functionality is not more supported under Integra for Notes 4.3. You can use Microsoft Word mergefield properties or formula fields to achieve the same.

The FdAdds() array stores a space character (" ") for those fields in the Word template where the name of the field tag ends with the + symbol, otherwise the value is a null string ("").

The + symbol can be used as part of the Word MailMerge field tag name to indicate to Integra that you wish to add a space character to the contents of the corresponding field in the Notes document (listed in the Selected Field List) only if the contents of the Notes field is not blank.

e.g. you are creating an address label and the first line of your label has the three field tags <Title+><FirstName+><Name>, then Integra will add a space to the Title and/or FirstName when the contents of the corresponding Notes field is not empty. If you would place the field tags as follows, <Title> <FirstName> <Name>, then empty spaces will be added when one of the Notes fields is blank.

The Selected Field List is stored in the Export runtime object. Please refer to the related online documentation for further details.

47 StaticVars.FdCount [Class Property]

Version: 2.x to 4.7
Availability: Word, Export

In all Integra releases prior to Integra 4.3 the FdCount property would be the sum of all Word mailmerge and form fields found in the Word document. However, any mergefields located in header or footer sections would be ignored. For Label profiles, the number would be divided by the number of labels as set in the Integra profile.

From Integra 4.3 onwards, the FdCount property would be the number of mailmerge fields (only) found in the Word document (again without header and footer sections).

In Integra 4.3 the exact number of fields per Word header/footer or document section is available through the use of the ubound() function referencing .

48 StaticVars.FdIdxSet [Class Property]

Version: 3.x to 4.7
Availability: Word, Export

This property indicates whether the field index arrays () have been created or not. This is not happening as soon Integra is initializing the Integra objects, but when processing the first selected Notes document. As soon as the arrays are built, FdIdxSet is set to True (1).

49 StaticVars.FdIndex [Class Property]

Version: 2.x to 4.2
Availability: Word, Export

This is the (cross-reference) field index array. For each field in the Word template it stores the index for the corresponding field in the Selected Field List within the Integra profile.

Note: there has been a significant change in the data type and data content of FdIndex from version 4.3 onwards.

Integra 4.2 and earlier releases

The lower bound starts with 1, as Microsoft is starting the Word field tag index with 1. The upper bound is the number of field tags in the Word template.

e.g. if the first field tag in Word is named "CompanyName" and the corresponding field is the third field in the Selected Field List, then FdIndex(1) = 2, as Notes lower bound starts always from 0 (zero).

The Selected Field List is stored in the Export runtime object. Please refer to the related online documentation for further details.

50 StaticVars.FdIndex [Class Property]

Version: 4.3 to 4.7
Availability: Word, Export

This is the (cross-reference) field index array. For each field in the Word template it stores the index for the corresponding field in the Selected Field List within the Integra profile.

Note: there has been a significant change in the data type and data content of FdIndex from version 4.3 onwards.

Integra 4.3 onwards

IntegraFieldIndex is a class definition containing 4 properties, each an array of field indexes for the respective supported field types:

```
Public Class IntegraFieldIndex
    Public FDI Merge() As Long ' Index of Word mailmerge fields
    Public FDI OCX() As Long ' Index of Word OCX fields
    Public FDI Form() As Long ' Index of Word form fields
    Public FDI BookM() As Long ' Index of Word bookmarks
End Class
```

Note: In Notes array indexes start (by default) with 0, all as the lower bound of any array is 0 (zero) and the upper bound is the number of elements in the array - 1. This is different as compared to the MS Office object structure, where array boundaries are starting at one (lower bound) going up to the number of elements (upper bound). Keep this in mind for the following paragraphs!

For every section in the Word template that does contain any of the supported fields, stats.FdIndex would be redimensioned and the number of fields of each type would be assigned to the respective property of FdIndex(x). Whereas FdIndex for x = 0 would have the field counts for the Word template's body (if there are any), for x = 1 refers to the header or footer of the first section and so on.

Field references are starting with 1 counting up to the number of fields in the Selected Field List. Hence to refer to the correct field in for instance export.record.structure.FdNames(i) one has to subtract 1 from the cross-reference value stored in stats.FdIndex(x).FDIMerge(y).

So if for x = section (0 for body) and y = Word field index - 1 the reference value is 1 we are referring to the first field in the Selected Field List as i = 1 - 1 = 0 and export.record.structure.FdNames(0) is the name of the first field in the Selected Field List as defined in the Integra profile.

How to calculate the total number of cross-referenced fields:

Let's assume the Word template contains 2 form fields and 3 mailmerge fields in the body part, 4 fields in the header of the first section and 1 field in the footer of the first section. In this examples ubound of FdIndex is 2 (FdIndex(0) = body fields, FdIndex(1) = 1st section header fields, FdIndex(2) = 1st section footer fields).

The following code line would return the total number of cross-referenced Word form fields:

```
ubound(stats.FdIndex(0).FDI Form)+1
```

Of course the above script is only accurate as we assumed for the example that there are only form fields in body part of the Word template.

To get the total number of fields (irrespective of the type) in the Word template referencing to a field in the Selected Field List one can use following script:

```
NoOfBodyFields = 0

For i = 0 to ubound(stats.FdIndex)
  if ubound(stats.FdIndex(i).FDMerge) > 0 then
    NoOfBodyFields = NoOfBodyFields + ubound(stats.FdIndex(i).FDMerge) + 1
  elseif stats.FdIndex(i).FDMerge(0) > 0 then
    NoOfBodyFields = NoOfBodyFields + 1
  end if

  if ubound(stats.FdIndex(i).FDIOCX) > 0 then
    NoOfBodyFields = NoOfBodyFields + ubound(stats.FdIndex(i).FDIOCX) + 1
  elseif stats.FdIndex(i).FDIOCX(0) > 0 then
    NoOfBodyFields = NoOfBodyFields + 1
  end if

  if ubound(stats.FdIndex(i).FDIForm) > 0 then
    NoOfBodyFields = NoOfBodyFields + ubound(stats.FdIndex(i).FDIForm) + 1
  elseif stats.FdIndex(i).FDIForm(0) > 0 then
    NoOfBodyFields = NoOfBodyFields + 1
  end if

  if ubound(stats.FdIndex(i).FDIBookm) > 0 then
    NoOfBodyFields = NoOfBodyFields + ubound(stats.FdIndex(i).FDIBookm) + 1
  elseif stats.FdIndex(i).FDIBookm(0) > 0 then
    NoOfBodyFields = NoOfBodyFields + 1
  end if
Next
```

51 StaticVars.FileID [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

Integra for Notes 3.x and earlier

The FileID property contains the file id of the CSV or custom textfile when running a Text profile.

Integra for Notes 4.x and later

For XML or Text profiles the FileID property contains the file id of the XML, CSV or custom textfile.

For Word and Excel export profile using the Normal export method, FileID contains the file id of the temporary CSV file, which is then imported into Excel or used as Word data control file.

The file id is created using the LotusScript function freefile()

52 StaticVars.FileID2 [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Export

The FileID2 property contains the file id of a second temporary textfile for web enabled Excel profiles which are executed on a non Windows server or on a Windows server without MS Office being installed. The file contains the values for fields linked to single cells (rather than row/column data).

The file id is created using the LotusScript function freefile().

53 StaticVars.FileName [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

Exports

The FileName property contains the file- and pathname under which the new Word, Excel or Textfile will be saved as (Integra export profile). The value is defaulted to either the filename defined in the Integra profile or to a random filename located in the Integra temporary directory.

Imports

The FileName property contains the filename of the Word, Excel or Textfile to be imported (Integra import profile). The value is defaulted to either the filename defined in the Integra profile or has to be selected by the user at runtime. For the latter a file select dialog will be presented to the user.

54 StaticVars.FileSelectWordDocs [Class Method (Function)]

Version: 2.x to 4.7
Availability: Word, Import

StaticVars.FileSelectWordDocs (Pathname As String)

The FileSelectWordDocs method is an internal Integra for Notes function called for a Word import profiles without pre-defined import filename. FileSelectWordDocs opens the OpenFileDialog allowing the user to select multiple Word document files.

The selected filenames (including the pathname) are stored in the `StaticVars.FList` property.

Parameters:

Pathname (String)

Default path the OpenFileDialog is opened with.

Return:

Number of files (Integer) selected and stored in `StaticVars.FList`.

55 StaticVars.FlatCat [Class Property]

Version: 4.1 to 4.7
Availability: Excel, Export

Flat Category

Integra for Notes 4.1 can export data from a categorised Notes view into an Excel spreadsheet in three ways:

1. It can keep the views formatting and categorisation as is, creating sort of an image of the Notes view in an Excel spreadsheet (FlatCat = False)
2. It can remove any categorisation levels/indentations and export the column values of categorised columns per Notes document (Flat Category; FlatCat = True)
3. It can export categorised and non categorised views and categorise (group) them in Excel using the new Grouping & Sorting feature of Integra 4.1 (Flat Category; FlatCat = True)

Option 1 and 2 can be controlled using the FlatCat property. If the Grouping & sorting feature is used, the FlatCat property is always True. The FlatCat property is exposed to the profile creator in the advanced tab of the Integra profile and to the end user in step 4 of the Quick Reports Wizard.

56 StaticVars.FList [Class Property]

Version: 2.x to 4.7
Availability: Word, Import

StaticVars.FList() As String

The Flist[] property is a dynamic array. For Word import profiles where no filename is pre-defined in the Integra profile, the user is presented with the Windows File-Open dialog. FList[] contains the list of files (path- and filename) as selected in the dialog.

The property can be used to programmatically define the list of files to be imported in the CB_INITIALISE event in the Script Callback field of the Integra profile.

57 StaticVars.ForceNavExp [Class Property]

Version: 4.3 to 4.7
Availability: Word, Excel, Text, Export

New to Integra for Notes 4.3

Integra for Notes utilises all Notes classes that allow processing of documents in a view (NotesDocumentCollection, NotesView, NotesViewEntryCollection and NotesViewNavigator). Which class is used depends on profile settings and on the context the user is in when calling the Integra profile.

The ForceNavExp property is relevant only when exporting all documents in a (active) view. The property is exposed in the UI of the Integra profile.

If ForceNavExp is True then Integra is using the NotesViewNavigator class to process the documents in the view. The NotesViewNavigator class is the only class that will return all view entries including category or total entries. If ForceNavExp is False then the class used depends on other profile settings.

The conditions are the following (evaluated at runtime in order of presentation):

Condition	Used Class	Exposed to UI (No = Advanced Script Only)
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes
stats.UseAllEntries = False	NotesViewNavigator	No
stats.ForceNavExp = True	NotesViewNavigator	Yes
View is NOT categorised	NotesViewEntryCollection	n/a
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes
stats.DoSort = False	NotesViewEntryCollection	Yes
stats.FlatCat = True	NotesViewEntryCollection	Yes
stats.FstCatOnly = True	NotesViewEntryCollection	Yes
Otherwise	NotesViewNavigator	n/a

58 StaticVars.FstCall [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

This FstCall property is set to True for the first processed (exported) Notes document, otherwise it is set to False.

59 StaticVars.FstCatOnly [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel

Export First Category Only

In cases where categorised columns of Notes views are based on multi-value fields, Notes documents can occur under multiple categories. When Integra exports data from categorised views it can either export the selected documents for each category they appear under (stats.FstCatOnly = False) or for the first category only (stats.FstCatOnly = True).

This property is exposed to the profile UI since version 4.0.

60 StaticVars.FTFilter [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export

FTFilter contains the full text filter string as defined in the Integra profile. However, the code sample below shows an example on how a filter condition can be set at runtime. The full text filter is executed after the CB_BFOREEXPORT event.

Example:
[Present custom dialog to set full text filter condition at runtime](#)

61 StaticVars.GCReload [Class Property]

Version: 3.x to 4.7
Availability: Excel, Export, Import

This property is only relevant for profiles using the Advanced COM export method.

All MS Excel releases earlier than MS Office Version 10 do have a memory leakage in the Excel COM objects. For those MS Excel releases Integra for Notes therefore runs a garbage collection for every 200 rows written to a spreadsheet. The garbage collection basically saves the spreadsheet to file, closes the Excel application and removes all COM objects from memory. Then it creates a new Excel application object and re-opens the spreadsheet.

The GCReload (garbage collection reload) property indicates when a garbage collection took place. The garbage collection procedure sets this property to true. This property tells Integra that with the next document to be written to Excel it needs to rebuild the internal cache of Excel tag objects, provided of course that the Use Tags option in the Integra profile is set to Yes.

62 StaticVars.GC_Delay [Class Property]

Version: 3.x to 4.7
Availability: Excel, Export, Import

This property is only relevant for profiles using the Advanced COM export method.

All MS Excel releases earlier than MS Office Version 10 do have a memory leakage in the Excel COM objects. For those MS Excel releases Integra for Notes therefore runs a garbage collection for every 200 rows written to a spreadsheet. The garbage collection basically saves the spreadsheet to file, closes the Excel application and removes all COM objects from memory. Then it creates a new Excel application object and re-opens the spreadsheet.

For some PC workstations without harddisk having a shared Network drive on which the Notes Client is installed, timing problems can occur when the garbage collection is executed. This depends on the Network performance. The GC_Delay property allows to insert a delay before Integra creates a new Excel application object and re-opens the Excel spreadsheet. The delay is in seconds, 0 (zero) means no delay (default).

If during runtime a file sharing access error occurs or the Excel spreadsheet is opened read only and a file save error occurs, you might need to use the GC_Delay property to force a delay. In our experience a delay of 1-2 seconds is sufficient, but again it depends on the Network performance in your environment.

Example:

'Callback Script Example for a Excel Export Profile

```
Const CB_INITIALISE = 3
Const CB_BFORREADNOTES = 6
Const CB_BFORWRITECOM = 1
Const CB_AFTERWRITECOM = 9
Const CB_BFORCOMACTION = 5
Const CB_BFORNOTESACTION = 2
Const CB_TERMINATE = 4

select case stats.cbstatus
case CB_INITIALISE
    ' force a 1 second garbage collection delay
    stats.GC_Delay = 1
case CB_BFORREADCOM
case CB_BFORWRITECOM
case CB_AFTERWRITECOM
case CB_BFORCOMACTION
case CB_BFORNOTESACTION
case CB_TERMINATE
end select
```

63 StaticVars.HasFooter [Class Property]

Version: 3.x to 4.7
Availability: Word, Export

The HasFooter property is used to optimise runtime performance of Word export profiles. By default Integra assumes that the attached Word template has footer sections containing eventual mailmerge fields (HasFooter = True). The profile developer can use the Script Callback field to set this property in the CB_INITIALISE event to False. The effect is that Integra will not search the footer section for contained mailmerge fields.

MS Word is organising the mailmerge fields not in one global field container object, but as properties of several different object containers. Fields contained in the main document part are organised as a property of the Word document object; fields located in, for instance, headers and footers are organised within the header and footer object, which in turn are organised in the section objects. Fields contained in headers and footers are therefore scattered throughout the sections object tree.

As a result the performance of Integra can be noticeable increased by setting the HasFooter property to False, provided that no fields are located in the template's footer sections.

This property is exposed to the profile UI since version 3.

64 StaticVars.HasHeader [Class Property]

Version: 3.x to 4.7
Availability: Word, Export

The HasHeader property is used to optimise runtime performance of Word export profiles. By default Integra assumes that the attached Word template has header sections containing eventual mailmerge fields (HasHeader = True). The profile developer can use the Script Callback field to set this property in the CB_INITIALISE event to False. The effect is that Integra will not search the header sections for contained mailmerge fields.

MS Word is organising the mailmerge fields not in one global field container object, but as properties of several different object containers. Fields contained in the main document part are organised as a property of the Word document object; fields located in, for instance, headers and footers are organised within the header and footer object, which in turn are organised in the section objects. Fields contained in headers and footers are therefore scattered throughout the sections object tree.

As a result the performance of Integra can be noticeably increased by setting the HasHeader property to False, provided that no fields are located in the template's header sections.

This property is exposed to the profile UI since version 3.

65 StaticVars.HasTemplate [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

Excel profiles

The HasTemplate property indicates whether the Integra (export) profile has a template attached to it (True) or not (False). If the Integra profile does not contain a template, Integra will write the data into a blank Excel spreadsheet using its default formatting.

Word profiles

This property has been available for Excel export profiles since version 2, however, for Word export profiles this property is only relevant since version 4.2.

If a profile has no template (HasTemplate = False) and the profile is set to use the Normal export method, then Integra will create a blank Word document and if MS Office is installed (hence COM is available) will create the list of mailmerge fields in the Word document separated by a carriage return. The user has the opportunity to finalise the document once it is activated.

Using Advanced COM as the method of export will cause a runtime error if the profile has no template.

66 StaticVars.ImpDoc [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Import

Currently processed Notes (target) document. You can use the handle to the Notes back-end document currently processed (created) to create additional fields, for instance to add a date-time stamp on the imported document as a means to log when the import happened with one common date-time value for all documents imported with the particular execution of the Integra import profile.

Another use is to look-up the target database for the existence of a document with a unique key value that is part of the imported fields. In case a document already exist with the same key, the documents field values will be overwritten, if no document exist with the key, a new document will be created.

Example:

[Import new Notes documents or update existing ones](#)

67 StaticVars.InitialiseFilename [Class Method (Function)]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export, Import

InitialiseFilename(record As Variant)

Depending on the profile type it calls one of the following StaticVars methods:

```
InitialiseFileName = (record)
InitialiseFileName = InitialiseFileNameWord(record)
InitialiseFileName = InitialiseFileNameText(record)
```

Parameters:

record (Variant)

Handle to the export.record object (export as FileExport, record as FileRecord)

Return:

True or False depending on what the respective above listed function returns.

68 StaticVars.InitialiseFilenameText [Class Method (Function)]

Version: 2.x to 4.7

Availability: Text, Export, Import

InitialiseFilenameText(record As Variant)

Initialises the output filename according to the pre-defined profile setting or randomly generated by Integra. If required the user will be presented with a FileSave or FileOpen dialog. The latter is mostly the case for import profiles. The file name is stored in .

Parameters:

record (Variant)

Handle to the record object (FileRecord)

Return:

True if the filename could be set or generated, otherwise False.

69 StaticVars.InitialiseFilenameWord [Class Method (Function)]

Version: 2.x to 4.7

Availability: Word, Export, Import

InitialiseFilenameWord(record As Variant)

Initialises the output filename according to the pre-defined profile setting or randomly generated by Integra. If required the user will be presented with a FileSave or FileOpen dialog. The latter is mostly the case for import profiles. The file name is stored in .

Parameters:

record (Variant)

Handle to the record object (FileRecord)

Return:

True if the filename could be set or generated, otherwise False.

70 StaticVars.InitialiseFilenameXls [Class Method (Function)]

Version: 2.x to 4.7

Availability: Excel, Export, Import

InitialiseFilenameXls(record As Variant)

Initialises the output filename according to the pre-defined profile setting or randomly generated by Integra. If required the user will be presented with a FileSave or FileOpen dialog. The latter is mostly the case for import profiles. The file name is stored in .

Parameters:

record (Variant)

Handle to the record object (FileRecord)

Return:

True if the filename could be set or generated, otherwise False.

71 StaticVars.InitialiseTmpFile [Class Method (Procedure)]

Version: 3.x to 4.7
Availability: Word, Excel, Export

InitialiseTmpFile()

Depending on the profile type it calls one of the following StaticVars methods to open a random named temporary text file (CSV) for export profiles using the Normal export method. The ID for the opened file is stored in stats.FileID, thus indicating the success or failure of the method.

[InitialiseTmpFileWord\(\)](#)

72 StaticVars.InitialiseTmpFileWord [Class Method (Procedure)]

Version: 3.x to 4.7
Availability: Word, Export

InitialiseTmpFileWord()

Opens a random named temporary text file (CSV) for Word export profiles using the Normal export method. The ID for the opened file is stored in , thus indicating the success or failure of the method.

73 StaticVars.InitialiseTmpFileXls [Class Method (Procedure)]

Version: 2.x to 4.7
Availability: Excel, Export

InitialiseTmpFileXls()

Opens a random named temporary text file (CSV) for Excel export profiles using the Normal export method. The ID for the opened file is stored in , thus indicating the success or failure of the method.

74 StaticVars.InitialiseUserProfile [Class Method (Procedure)]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export, Import

InitialiseUserProfile(file As String, agent As String, fileuse As String, filetype As String)

All users have a hidden profile document storing runtime information. Integra creates this profile automatically with the first time a user runs Integra. As soon as an Integra profile is executed it will call this method to initialise the processing of the profile. A handle to the users profile is available in the stats.usrdoc property. The following fields on the user profile are initialised:

```
stats.usrdoc.NotesAction = stats.NotesAction
stats.usrdoc.Action       = stats.action
stats.usrdoc.FileName     = file
stats.usrdoc.Agent       = agent
stats.usrdoc.FileUse      = fileuse
stats.usrdoc.FileType     = filetype
```

' in addition for browser executed profiles

```
stats.webactdoc.ProfType = "Excel Spreadsheet"
or
stats.webactdoc.ProfType = "Word Document"
```

Parameters:

file (String)

Name of the output file either as pre-defined in the Integra profile or otherwise generated by Integra.

agent (String)

Name of the Integra agent to be called by the Integra background windows.

fileuse (String)

Either "Import" or "Export".

filetype (String)

Depending on the profile type either one of the following: "Report", "Mailmerge", "Single Document" or "Labels".

75 StaticVars.IsQuickRep [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Export

The IsQuickRep property indicates whether the Integra profile is a (saved) Quick Report (True) or not (False).

In version 4.1 this property was only relevant for Excel profiles. Quick Reports for Word Mailmerge and Labels has been introduced in Integra 4.2.

76 StaticVars.KeyArr [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

Integra has two ways of exporting data from a Notes database. The first one is giving the user the ability to select documents in a view and the second option is to export all documents within a pre-selected view (or folder). With regards to sorting, the latter is not a problem. This as Integra can utilise the view's sorting to export data in the correct order and therefore the KeyArr[] property is not used for pre-selected view exports.

It is the first option, where users are selecting documents within the active view where Integra has to work on a document collection which is not sorted. This is actually a very unfortunate limitation of Notes document collections.

To re-establish the original sorting of the view for the exported document collection, Integra has to perform a very complex sorting procedure. Please refer to the StaticVars.DoSort property for more details.

The KeyArr[] property contains the above described sorting strings for each of the selected documents.

The property is only relevant in the context of exporting "Selected Documents in Active View"

77 StaticVars.KeyCnt [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

StaticVars.KeyCnt As Long

The KeyCnt property is the counter for the processed (exported) documents. It starts with 0 (zero) for the first exported document and counts up to the number of documents exported - 1 (StaticVars.KeyNum - 1).

If the Integra profile is using the "Selected Documents in Active View" option then the KeyCnt property can be used as the index to the currently processed (exported) Notes document listed in the StaticVars.KeyArr[] property. Please refer to the property.

78 StaticVars.KeyNum [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The KeyNum property contains the number of documents to be exported.

If the Integra profile is using the "Selected Documents in Active View" option then the KeyNum property is equals to the number of elements in the StaticVars.KeyArr[] array, otherwise it is equals to the number of documents selected in the pre-defined view.

79 StaticVars.LabelMergeWiz [Class Property]

Version: 4.3 to 4.7
Availability: Word, Export

New to Integra for Notes 4.3. Only available with Microsoft Office 10 or later

Represents the value of the advanced option of the profile field Export Method ("Launch Word Label/Mailmerge Wizard on Activate" = True, "Complete Word Merge before Activate" = False).

If LabelMergeWiz is set to True, the activated Word mailmerge document will be opened in Word's Mailmerge Wizard otherwise the Word merge process will be completed.

80 StaticVars.Language [Class Property]

Version: 4.3 to 4.7
Availability: Word, Excel, Text, Export, Import

New to Integra for Notes 4.3

Contains a lower case 2 character language code according to the user's Integra or Notes Client language preferences. The code is used to look-up language specific text for runtime message. Default is "en" for English.

81 StaticVars.LastFName [Class Property]

Version: 4.2 to 4.7
Availability: Word, Export

Used to temporarily store the filename of the current Word document whenever the filename (stats.Filename) is changed due to file handling processes, such as PDF conversion. After the conversion process the filename (stats.Filename) is restored.

This property is only used in the context of advanced COM Word mailmerge profiles.

82 StaticVars.LaunchMacro [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Export

Before Launch Macro.

Name of a Word or Excel macro to be executed after the last exported Notes document and before the created Word or Excel document is launched or otherwise processed (Send To, Save As...). The execution of the macro follows the CB_BFORCOMACTION event. This property is exposed to the profile creator in the advanced tab of the Integra profile. However, the name of the macro can be changed dynamically at runtime.

With Integra 4.1 two new Excel export methods have been introduced, namely Normal (using CSV) and XML. These two methods avoid the use of COM technology and are therefore much faster. However, this performance improvement comes with a reduced flexibility or usability when it comes to the Integra events. The main difference between the use of COM technology and the new CSV or XML export is that the latter do not write every single exported Notes document into rows in Excel, but rather collect all rows in either CSV format or XML format (MS Office 2003+). Once all Notes documents are exported to CSV or XML, the CSV or XML file is imported into the Excel spreadsheet. The consequence of this is that the CB_BFORCOMACTION event becomes very limited in its use, thus the Launch Macro useless in this context.

83 StaticVars.LstCall [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

This LstCall property is set to True after the last processed (exported) Notes document, otherwise it is set to False.

This property is for read only purposes, do not modify the value; it cannot be used in the CB_BFORWRITECOM event to identify the last exported document; instead use the StaticVars.KeyCnt and StaticVars.KeyNum properties to do this.

84 StaticVars.LstSet [Class Property]

Version: 4.0 to 4.7
Availability: Excel, Export

Last Call Settings

After Integra exports the last row to an Excel spreadsheet it will set certain Excel tags used by Integra e.g. for a later import of the same Excel spreadsheet with a modified data set (_Datablock, _Columnheader, _Columnfooter).

It will also populate the reserved Integra Excel tags _DocumentDate, _DocumentTitle, _DataRows.

With this property one can control whether Integra is calling the export.XlsLastCallSettings() method or not. By default the method is executed.

85 StaticVars.MailAction [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

This property is used in conjunction with the Notes Action option Send To. Depending on the profile type and other options set on the profile, one of the following three mail options can be set on the profile which is consequently stored as text string in the StaticVars.MailAction property:

Immediate:

When selecting this option, Integra for Notes will send any mail without further confirmation or intervention by the user. This option implies that at least one of the three formula fields Send To, CC and BCC does return an email address. If no email address is returned when evaluating the formulas, no mail is being send. If the Integra for Notes log database is installed, Integra for Notes will create an entry in the log file.

Confirm:

When selecting this option, Integra for Notes will present a confirmation dialog prior to sending out the email. The dialog will allow to enter or select recipients from the names and addressbook, to enter a subject and a heading for the body text. If any formula has been defined in the profile for the formula fields Send To, CC, BCC or Subject, the evaluated result of these formulas will be presented as default values in the dialog.

Memo:

When selecting this option, Integra for Notes will create a new memo in the user's mail database. This option will allow the user to take full advantage of all the mail options available in the Notes mail database. As for the Confirm option, the fields Send To, CC, BCC and Subject are defaulted to the evaluated result of the formulas of the corresponding formula fields in the Integra for Notes profile.

86 StaticVars.MailAsLink [Class Property]

Version: 4.2 to 4.7
Availability: Word, Excel, Export

Notes back-end document object that contains the Notes document which is to be send as document link. This property is used in conjunction with the stats.SendAsFormat and the stats.NotesAction property, both of which are exposed in the profile UI.

When using the Send To and Save As... options (stats.NotesAction) together with the Send Document Link to Saved Attachment (stats.SendAsFormat) then Integra for Notes will assign the document handle in which it saved the Word or Excel attachment to the stats.MailAsLink property. This document handle is by default the same that is assigned to stats.SavDoc.

Using advanced script, one can assign any document handle to which a document link is to be send. The appropriate event to do this, is the CB_BFORNOTESACTION event.

87 StaticVars.MailConf [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

This property is used in conjunction with the Notes Action option Send To. If this property is set to True, Integra will present a confirmation dialog prior to sending out an email attachment (Word or Excel). This dialog allows to define the recipients, subject and body heading of the mail.

In Integra for Notes Release 2 this setting was not exposed to the profile creator as an options field in the profile. The only way to control the property was from within the Script Callback field, adding some Lotus Script. By default this property was set to False.

In Integra for Notes Release 3 this property is exposed to the profile creator with an additional Send To option allowing the profile creator to choose between Integra creating a new Memo in the user's mail database, opening the Integra Mail-Confirmation dialog or sending the mail without further confirmation.

However, the MailConf property remains in use and is initialised according to the profile's setting.

88 StaticVars.MailSent [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

This property is used in conjunction with the Notes Action option Send To. It indicates whether sending the mail was successful (True) or if any error occurred (False). Currently this property is of little use as there is no event after the CB_BFORNOTESACTION event, other than the CB_TERMINATE event. It is mainly used by Integra's internal processing.

89 StaticVars.Memo [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

Notes back-end document object that contains the memo document created for sending a Word or Excel attachment (SendTo option in the Integra profile). The object is set prior to the CB_BFORNOTESACTION event. This event is actually triggered twice: before sending an email and before saving a Word or Excel attachment back into a Notes document.

90 StaticVars.MergeChr [Class Property]

Version: 3.x to 4.7
Availability: Word, Export

The MergeChr property contains the character used to separate exported Notes documents collected (merged) in one Word document. The property applies for Word export profiles with the options Mailmerge & Activate/Preview or Mailmerge & Save As... with Save ONE Attachment in a Single Word Document.

By default the MergeChr property contains a Chr(12), which is the page break character used by MS Word. If the default value is not changed this means that each exported Notes document results in one or more Word pages separated with a page break before the next collection of pages begins.

Since release 4.1 following acronyms are valid:

```
stats.mergechr = "CR"  
stats.mergechr = "WDLINBREAK"  
stats.mergechr = "WDPAGEBREAK"  
stats.mergechr = "WDCOLUMNBREAK"  
stats.mergechr = "WDSECTIONBREAKNEXTPAGE"  
stats.mergechr = "WDSECTIONBREAKCONTINUOUS"  
stats.mergechr = "WDSECTIONBREAKKEVENPAGE"  
stats.mergechr = "WDSECTIONBREAKODDPAGE"  
stats.mergechr = "WDLINBREAKCLEARLEFT"  
stats.mergechr = "WDLINBREAKCLEARRIGHT"  
stats.mergechr = "WDTXTWRAPPINGBREAK"
```

At runtime these will be replaced by their respective MS Word constant. Of course any other character combination can be assigned.

91 StaticVars.MsoAbsent [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Export

The MsoAbsent property indicates whether Microsoft Office is available on the server (False) or not (True). This property is only relevant for scheduled or web-enabled profiles as these are executed on the server. Integra checks at runtime whether the operating system is Windows based or not and if it is Windows based whether MS Office is installed or not. Only if the latter is the case the property is set to True otherwise False.

Note: the property is available in Integra for Notes 4.1, but is not completely supported. It just indicates whether the server is Windows based or not. It does not indicate the presence of MS Office.

92 StaticVars.New [Class Method (Procedure)]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

Default object class method called with the beginning of Integra's processing cycle. The method initialises the stats runtime object.

93 StaticVars.NotesAction [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

Depending on the profile type and other options set on the profile, either none or a combination of the following options can be set on the profile which is consequently stored as text string in the StaticVars.NotesAction property.

None (or blank):

If the property NotesAction is a blank string or is set to None then no Notes Action will be performed.

Send To:

The Word or Excel document is going to be send as mail attachment. Please refer also to the MailAction property.

Save As Document:

The Word or Excel document is attached to a new Notes main document.

Save As Response:

The Word or Excel document is attached to a new Notes response document. The parent document is the currently exported Notes document

Attach To Document:

The Word or Excel document is attached to the currently exported Notes document.

The Notes Actions None, Save As... and Attach To Document are exclusive, however, a combination of Send To and one of the Save As... or the Attach To Document option is possible. If Send To is combined with one of these options then both options are separated by a semicolon. E.g. "Send To; Save As Document".

94 StaticVars.NotesDB [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export, Import

The NotesDB property contains the file- and pathname of the Notes source database Integra is exporting from or importing to. The value is defaulted to the database name defined in the Integra profile.

95 StaticVars.NotesSrv [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export, Import

The NotesSrv property contains the server name where the Notes source database is located from which Integra is exporting from or importing to. The value is defaulted to the server name defined in the Integra profile.

96 StaticVars.Password [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

Integra supports the Word and Excel protection mechanism. In the Integra profile as well as in the runtime Prompt User dialog it is possible to select one or more (Excel only) protection options. Integra is generating at runtime a random password. In the Prompt User dialog the user can actually overwrite the password.

The password is stored in the Password property and can be changed programmatically. for instance, in the CB_INITIALISE event.

97 StaticVars.PBar [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export, Import

StaticVars.PBar As ProgressBar

The PBar property contains the object handle to Integra's runtime progress bar. Please refer to the WinAPIWrapper documentation for information about the properties and methods of the progress bar object class.

98 StaticVars.Prntr [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The Prntr property contains the name of the printer to be used for export profiles which use the Print option. Prntr is defaulted to the Word or Excel default printer or to whatever printer is selected when the profile options are set to present the user the print dialog prior to printing.

99 StaticVars.ProcStat [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export, Import

New to Integra for Notes 4.1 Build 58

The ProcStat property controls the runtime processing (loop). During normal processing the value of ProcStat is set to 1 (one), which indicates to continue processing the events normally. Detailed processing control can be achieved with the Continue property of the runtime export or import object. Setting the value of ProcStat to 0 (zero) will abort the processing immediately. Any set COM or Notes actions will not be executed. Within the processing loop (for an export this would be between the CB_BFORREADNOTES and the CB_AFTERWRITECOM event) the value of ProcStat can also be set to 2 (two), which will let Integra to process the same exported Notes document again.

Note: Using one of the above mentioned events to set ProcStat to 2 can lead to an endless loop processing one and the same Notes document repeatedly if no proper conditioning is put in place.

One of the positive effects of the introduction of this property is that Integra will automatically set ProcStat to 0 (zero) to abort the export or import as soon as a runtime error is triggered. In earlier releases and builds, Integra continued processing with the next event following the one in which the error occurred, resulting in nasty repeated error messages for every single Notes document exported.

100 StaticVars.ProDoc [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text

Contains the backend handle to the currently processed Integra profile. All profile information is read into Integra's runtime objects before the CB_INITIALISE event.

101 StaticVars.Profile [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

Name of the profile selected by the user and currently executed.

102 StaticVars.Protect [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

The Protect property contains the Word or Excel protection option as set on the profile or as chosen at runtime in the Prompt User dialog.

103 StaticVars.PrtDialog [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

When using the Print option for Word or Excel export profiles the PrtDialog option controls whether the user is presented with the Word or Excel print dialog. If first set to True, PrtDialog is set to False after the dialog was indeed presented.

104 StaticVars.QuitWord [Class Method (Function)]

Version: 2.x to 4.7
Availability: Word, Export, Import

This method is called from the method. It removes the Word COM objects from memory (closing any eventually still open Word files).

The method re-sets all Word environment variables to its former state if changes where required for Integra's processing.

Return:

True if there where any Word COM objects still in memory that could be closed successfully, otherwise False.

105 StaticVars.ReadMethod [Class Property]

Version: 4.3 to 4.7
Availability: Excel, Export

The ReadMethod property indicates whether the list of to be exported fields refers to

fields and view columns values (Read_From_DocFields)

or

column values only (Read_From_ColValues).

Integra determines the value at runtime. The default value is Read_From_DocFields. Should the Selected Field List of the Integra profile contain only view column references then stats.ReadMethod is set to Read_From_ColValues.

106 StaticVars.RecCnt [Class Property]

Version: 2.x to 4.7
Availability: Text, Export

The RecCnt property contains the number of actually exported documents for Text exports. RecCnt does not necessarily represent to the number of documents selected in the Notes view selected for the export, as documents could be ignored due to filter conditions (e.g. Selection Formula field in Integra profile).

Example:

[Notify user of the total number of documents exported](#)

107 StaticVars.Record [Class Property]

Version: 4.3 to 4.7
Availability: Word, Export

Contains a handle to export.record (FileExport.FileRecord). Please refer to the documentation of the respective class.

108 StaticVars.ROS_Mail [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export

An Integra profile can be configured to run with the browser, on a scheduled basis and manually by the user. For the latter however, the Integra profile provides an option to run the profile either on the client, optionally on the server or always on the server (advanced tab). The last two options mean that the profile will though manually initiated, not run on the client, but on the server. The profile will not be processed immediately on the server, but with the next scheduled processing of other scheduled profiles. As the result is produced on the server and not on the client, there is no direct link between the user and the Excel or Word document. Therefore Integra stores in the ROS_Mail property the user's mail address and saves it temporarily until the scheduled process on the server can pick up the information to whom to send the result to.

109 StaticVars.Rows [Class Property]

Version: 2.x to 4.7
Availability: Excel, Import

Number of Excel rows

For Integra Excel import profiles this property contains the number of rows that Integra will import into Notes. Integra will calculate the value depending on whether the import is using Excel tags or not.

If the Use Tags option is set to Yes Integra will look for a tag called _DATABLOCK and will determine the number of rows within the cell range with that name.

If Use Tags is set to no Integra will determine the number of rows referring to the number of rows in the used range (UsedRange property of the Excel spreadsheet).

110 StaticVars.RTExpForm [Class Property]

Version: 4.2 to 4.7
Availability: Word, Export

The RTExpForm property contains the name of the form used for non API rich text export to Word documents. By default the related stats.ApiExpRT property is True and therefore the stats.RTExpForm property is irrelevant. If stats.ApiExpRT is set to false then stats.RTExpForm contains the form to be used to display the rich text on screen that will then be copied to the Windows clipboard and then pasted into the Word document.

Both properties are not exposed in the profile UI and only be changed using advanced scripting.

Using non API export:

The Lotus Notes client always had issues with in-line images, in particular when it comes to exporting them. As Integra is using the Notes API for rich text export, Integra would simply fail to export some documents containing in-line images (mainly if of type bitmap .bmp).

Integra allows the alternate use of a copy and paste method rather than the Notes API. The copy and paste method is UI driven, opening the rich text in a new document and literally copying and pasting the rich text into the Windows clipboard. A profile developer can switch between both methods using advanced script. The following line entered into the CB_INITIALISE event would change to the copy and paste method:

```
stats.ApiExpRT = False
```

Changing the default copy and paste export form:

When using Copy & Paste as the method of exporting Notes rich text, Integra by default is using a form stored in the Integra database (CopyAndPasteRT) to open the rich text on screen. The reason for it is that it is less intrusive than the alternative: creating a form in the source database design.

However, it comes with reduced performance as the rich text has first to be copied into a hidden document in the Integra database before it can be opened with Integra's own form. It also can cause loss in fidelity.

Assigning any other form name than "CopyAndPasteRT" to stats.RTExpForm will let Integra locate and use this form in the source database rather than using Integra's own form. The effect of this is that the rich text does not need to be copied across to the Integra database and therefore the process is much faster. It also eliminates possible problems with copying in-line images across.

The Integra database contains a sample form that can be copied into any source database for the use of the copy and paste export method. Once copied into the design of the source database, the design inheritance has to be removed. The form is called "CustomCopyAndPasteRT".

Note: by default the form contains a rich text field called Body. If the rich text field to be exported from the source database has a different name, this field has to be renamed accordingly. If there are more than one rich text fields to be exported, a rich text field per to be exported rich text field (with the corresponding name) has to be created in this form.

Example:

UI driven rich text export

111 StaticVars.RTImpForm [Class Property]

Version: 4.2 to 4.7
Availability: Word, Import

!!! Note: not implemented yet !!!

The RTImpForm property contains the name of the form used for non API rich text import from Word documents. By default the related stats.ApilmpRT property is True and therefore the stats.RTImpForm property is irrelevant. If stats.ApilmpRT is set to false then stats.RTImpForm contains the form to be used to paste the from the Word document copied rich text content before it is saved into the Notes target document.

Once implemented the same explanation as for stats.RTExpForm can be added here (of course focusing on the import of rich text).

112 StaticVars.RunOnServer [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

Indicates whether the profile is to be executed on the client (0), optionally executed on the server (1) or always on the server (2). In case of Optionally Run On Server, the user will be prompted at runtime to choose the appropriate action. However, to be prompted, the Prompt User option in the Word or Excel tab has to be selected.

113 StaticVars.SavDoc [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

Notes back-end document object that contains the Notes document created for saving a Word or Excel attachment (Save As... options in the Integra profile). The object is set prior to the CB_BFORNOTESACTION event. This event is actually triggered twice: before sending an email and before saving a Word or Excel attachment back into a Notes document.

Example:
[Add fields to documents created with the Save As... options](#)

114 StaticVars.SaveAsPDF [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

Indicates whether the Excel or Word output is to be converted to PDF or not. A value of 0 (zero) means no conversion to PDF, a value of 1 means that the output is converted before launching Word or Excel and a value of 2 means the output is converted after Word or Excel are launched or after the user closed the Office application respectively. If the PDF conversion fails, Integra will protect the Word or Excel file with MS Office' own protection methods to prevent further modifications to the document.

The default value reflects the settings in the Integra profile. If the Prompt User option is selected the user can overwrite the default setting.

115 StaticVars.SendAsFormat [Class Property]

Version: 4.2 to 4.7
Availability: Word, Excel, Export

This property is used in conjunction with the stats.NotesAction property. Both properties are exposed in the profile UI.

When using the Send To and Save As... options (stats.NotesAction), the output file (Word, Excel or PDF document) can be send as attachment (), it can be converted back into Notes rich text before sending it, or a link to a Notes document containing the Word or Excel attachment will be send.

SendAsFormat will reflect one of the following options selected in the Integra profile:

Send Attachment (Copy) = 0
Send As Notes Rich Text = 1
Send Document Link to Saved Attachment = 2:

116 StaticVars.SendAsRTF [Class Property]

Version: 4.3 to 4.7
Availability: Word, Excel, Export

New to Integra for Notes 4.3

The SendAsRTF option allows in combination with the Send To option to send the Excel or Word output as Notes rich text, building a native part of the Notes mail body field rather than being sent as a file attachment. This is a useful feature to run mail shots where it is not certain whether the recipients do have MS Office installed.

Note: Integra is using the (undocumented) Notes rich text import API feature to import the Word or Excel file into Notes rich text. There are limitations as to the fidelity of this import and the size of Word or Excel document that can be imported.

117 StaticVars.SetWordDocOptions [Class Method (Procedure)]

Version: 3.x to 4.7
Availability: Word, Export, Import

StaticVars.SetWordDocOptions (fstcall As Integer)

Integra requires the below listed Word environment variables (options) to be set in a particular way. This method sets and re-sets the variables according to the given parameter value.

stats.WordDoc.BuiltinDocumentProperties(1).value	= stats.title
stats.WordDoc.UpdateStylesOnOpen	= False
stats.WordAppl.Options.PrintBackGround	= False
stats.WordAppl.Options.BackGroundSave	= False
stats.WordAppl.Options.CheckGrammarAsYouType	= False
stats.WordAppl.Options.CheckSpellingAsYouType	= False
stats.WordAppl.Options.SaveInterval	= 0

Parameters:

fstcall (Integer)

If fstcall (first call) is True then all Word environment variables set by this method are cached and then set as required, otherwise they are set back to their original value.

118 StaticVars.SetXlsApplOptions [Class Method (Procedure)]

Version: 3.x to 4.7
Availability: Excel, Export, Import

StaticVars.SetXlsApplOptions (XlsAppl As Variant, XlsWBook As Variant, fstcall As Integer)

Integra requires the below listed Excel environment variables (options) to be set in a particular way. This method sets and re-sets the variables according to the given parameter value.

XlsWBook.EnableAutoRecover = False

Parameters:

XlsAppl (Variant)

Handle to the Excel COM Application object (as stored in stats.XlsAppl)

XlsWBook (Variant)

Handle to the Excel COM Workbook object (as stored in stats.XlsWBook)

fstcall (Integer)

If fstcall (first call) is True then all Excel environment variables set by this method are cached and then set as required, otherwise they are set back to their original value.

119 StaticVars.ShowUserDialog [Class Method (Function)]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export, Import

StaticVars.ShowUserDialog (dlg_form as string, dlg_title as string)

This method of the stats object allows to call custom dialogs with the least possible parameters to e.g. enable users to enter filter conditions at runtime. The method requires only two parameters: the dialog form or subform and the dialog title. The custom form or subform has to be created in the Integra database. The method calls the DialogBox() function of the Utilities script library, which provides a view more parameters:

' Please refer to the Utilities script library documentation for the details re the additional parameters
 ShowUserDialog = Dialogbox(dlg_form,True,True,False,False,False,False,dlg_title,True,False)

The DialogBox function in turn calls the NotesUiWorkspace.DialogBox() method in the context of the user's runtime profile (hidden Notes document), in which any field input is saved.

Note: if you require a user to input additional information at runtime you need to use this method of the stats object.

Example:

' For this sample we assume that there is a custom form called "ReportPeriodDialog" created in the Integra database,
 ' containing two fields named "Date_From" and "Date_To" to enter a date period.

```
Const CB_INITIALISE = 3
Const CB_BFOREXPORT = 10
Const CB_BFORREADNOTES = 6
Const CB_BFORWRITECOM = 1
Const CB_BFORCOMACTION = 5
Const CB_BFORNOTESACTION = 2
Const CB_TERMINATE = 4
```

```
Select Case stats.cbstatus
Case CB_INITIALISE
Case CB_BFOREXPORT
Dim period_from As Variant
Dim period_to As Variant
Dim usrdoc As notesdocument
```

' Use either the simplified stats.DialogBox() method...

```
export.continue = stats.ShowUserDialog("ReportPeriodDialog","Report Period Dialog")
```

' ...or alternatively the more advanced DialogBox() function of the Utilities library.

' You can modify the other parameters to change the behaviour of the dialog box.

```
' export.continue = Dialogbox("ReportPeriodDialog",True,True,False,False,False,False,"Report Period Dialog",True,False)
```

If export.continue Then

' The user's input is stored in the user's hidden database profile.

```
Set usrdoc = GetUserProfile()
```

' With Integra 4.3 it is recommended to use following code line instead

```
' Set usrdoc = stats.Utilities.utlGetUserProfile()
```

```
period_from = usrdoc.Date_From(0)
period_to = usrdoc.Date_To(0)
```

```
stats.FTFilter = | FIELD PostedDate >= | & period_from & | AND FIELD PostedDate <= | & period_to
End If
```

```
Case CB_BFORREADNOTES
Case CB_BFORWRITECOM
Case CB_BFORCOMACTION
Case CB_BFORNOTESACTION
Case CB_TERMINATE
End Select
```

120 StaticVars.SortGroup [Class Property]

Version: 4.0 to 4.7
Availability: Excel, Export

This property contains the internal representation of the profil's or user's (Quick Reports) sorting and grouping definition, as well as further properties and methods to process the definition.

The object class is defined here...

121 StaticVars.SrcDB [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export

The SrcDB property contains the handle to the Notes database (NotesDatabase class) from where the user initiated the Integra profile.

122 StaticVars.SrcDB2 [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Text, Export

The SrcDB2 property contains the handle to the Notes database (NotesDatabase class) into which any Word or Excel documents are saved back as attachments. Depending on the profile settings this can be either the source database (in which case stats.SrcDB and stats.srcDB2 are pointing to the same database) or any other Notes database as selected in the Integra profile.

123 StaticVars.Template [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Export

The Template property contains the path- and filename of the temporarily detached Word or Excel template stored in the Integra profile. Integra is detaching the profile in its temporary directory.

124 StaticVars.TimeStamp [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

When the object is initialized, the TimeStamp property will be set to the current date and time. The TimeStamp property can be utilized to mark Notes documents exported or imported in a particular profile execution with a common time-stamp.

Example:

[Common time-stamp for imported documents](#)
[Coded filter condition & time-stamp for exported documents](#)

125 StaticVars.Title [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The Title property contains by default the name of the profile. If the template contains a field or tag named _DocumentTitle, the Title property will be written into the _DocumentTitle field/tag.

If the Integra profile uses the Prompt User option, then the user can change the Title property in the Prompt User runtime dialog. The profile developer can also programmatically change the document title in the CB_INITIALISE event of the Scrip Callback field.

126 StaticVars.TmpFileNam2 [Class Property]

Version: 4.0 to 4.7
Availability: Excel, Export

Exports

The TmpFileNam2 property contains the file- and pathname of a second temporary CSV file which is created when using the Normal export method for web-enabled Excel profiles running on a non Windows Domino server or on Windows Domino Server without Microsoft Office installed. The second temporary file is required to write the data for single cell tags.

127 StaticVars.TmpFileName [Class Property]

Version: 4.0 to 4.7
Availability: Word, Excel, Export

Exports

The TmpFileName property contains the file- and pathname of the temporary CSV or XML file which is created when using the Normal or XML (Excel only) export method.

128 StaticVars.TxtExpDelCR [Class Property]

Version: 4.3 to 4.7
Availability: Text, Export

If the TxtExpDelCR property is set to True (default), any carriage return contained in the text exported to the CSV file is removed. For most CSV import filters the default behaviour is required to create a valid CSV format. Unless required this property should not be changed.

129 StaticVars.UnicodeSet [Class Property]

Version: 4.3 to 4.7
Availability: Word, Excel, Text, Export, Import

In Integra for Notes 4.2 the use of the unicode character set UTF-16 was introduced for its Normal export method. Prior to Integra 4.2 the temporary text files used for the Normal export were simple DOS text format. With the implementation of UTF-16 it became possible to export special characters such as German vowels (ÖÄÜöäüß).

Integra 4.3 now allows users to set their own export preference with regards to the unicode character set used for the exporting of data. All character sets supported by Lotus Notes are now also supported by Integra for Notes 4.3.

It is also possible using advanced script to force a particular unicode characters set on a per profile basis, effectively overwriting the user preferences. The new *unicodeset* property of the *stats* runtime object can be used to change the unicode character set used for a profile. For instance, the following code placed in the CB_INITIALISE event will change the unicode character set used to Japanese characters:

Note: Text import profiles do currently not support unicode character set.

Example:
[Forcing a specific unicode character set](#)

130 StaticVars.UsDatFmt [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

The UsDatFmt property determined the format of the date string presentation (format) of stats.DateText.

In **Integra 2.x** the Prompt User dialog (export only) actually included a field to allow the user to modify this date-time value. However, in **Integra 3.0** this field has been removed from the Prompt User dialog, instead the profile creator can add some Lotus Script to the Script Callback field in the Integra profile to allow a user to set a different date-time before the Word or Excel document is created.

With **Integra for Notes 4.1** the Export Current View To Excel profile has been superceeded by the Quick Reports Wizard which does mainly the same thing, but with some more user controlled features. The profile has subsequently been removed from the Integra template. The existing Export Current View To Excel profile will not be deleted by an upgrade to Integra for Notes 4.1, however, the profile will stop functioning.

Note: since Integra for Notes 4.1 the UsDatFmt property is no more in use. It has remained for backwards compatibility reasons, but it's setting has no effect.

131 StaticVars.UserVar [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export, Import

This variable can be used to forward any information from one Integra for Notes event to another and from one processed Notes document to the next one.

As the Lotus Script stored in the Callback Script field (Advanced tab) is executed for each of the Integra for Notes run-time events, any variable declaration within the Callback Script is lost after the code is executed for a particular event. To pass on information from event to event or from one processed document to the next one the developer can store temporary variables, for instance, a counter a totals in the *stats.UserVar* property.

Any data type that can be assigned to a variant is allowed. It is perfectly possible to get a handle to a back-end document of a different database and to temporarily store the handle to it in *stats.UserVar*.

Limitations:

To allow a user to run an Integra for Notes profile that for instance activates a Word document on screen to afterwards send the modified Word document as an email attachment, Integra for Notes has to wait for the user to save and close the Word document before it can send it. To accomplish this it seems logical that Integra for Notes would need to freeze the Notes client before it can call the CB_BFORNOTESACTION event and send the mail. As a matter of fact, Integra for Notes 2.x did exactly that.

However, since Integra for Notes 3.x the Notes UI is unlinked from the activated Word or Excel document and therefore allows the user to continue doing something else within the Notes client while having an unfinished Integra for Notes profile running. To achieve this, Integra for Notes actually activates any Word or Excel document and then completes the profile execution before calling the CB_BFORNOTESACTION event. With the completion of the profile execution all run-time objects are actually freed from memory. A timer document in the Notes client waits for the user to close the Word or Excel document and as soon that happens, re-initiates the Integra for Notes profile execution with a re-entry point where it had previously stopped; recovering the run-time objects to the same status as they had been before the previous execution stopped. As a result, the user can freely switch between Notes and Word or Excel.

However, this advantage comes with a limitation in the storage of temporary variables. To forward any data stored in the *stats.UserVar* property to the CB_BFORNOTESACTION event, the data has to be temporarily saved in the user's Integra for Notes profile. Obviously a Notes document has some limitations as to what data types it can store. For instance it is not possible to save a Notes back-end document object handle, but it can store the UNID of the latter. So to pass on a back-end document handle the developer has to store the UNID and to set the back-end document handle in the CB_BFORNOTESACTION using the *NotesDatabase.GetDocumentByUNID()* method.

132 StaticVars.UseViewFrmt [Class Property]

Version: 4.0 to 4.7
Availability: Excel, Export

The UseViewFrmt property has been introduced in connection with Integra Quick Reports. If True Integra will try to export the Notes view to Excel as close as possible to "as seen on screen", including column and data formatting (such as font sizes and types). When using this option, Integra will overwrite any settings in a template attached to the Integra profile.

Integra for Notes utilises all Notes classes that allow processing of documents in a view (NotesDocumentCollection, NotesView, NotesViewEntryCollection and NotesViewNavigator). Which class is used depends on profile settings and on the context the user is in when calling the Integra profile.

The table below indicates for which of the conditions (hence used classes) Integra will include category and (sub-)total entries besides the font, color and data type formatting:

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	Includin g Categories & Totals
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes	Yes
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	No
View is NOT categorised	NotesViewEntryCollection	n/a	No
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	No
stats.DoSort = False	NotesViewEntryCollection	Yes	No
stats.FlatCat = True	NotesViewEntryCollection	Yes	No
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	No
Otherwise	NotesViewNavigator	n/a	Yes

133 StaticVars.UsrDoc [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export, Import

This property contains the document handle to the user's hidden database profile, which is used to store temporary runtime information such as fields from arequesting e.g. filter conditions from the user. Any information entered by the user will be stored in the corresponding fields in this profile document. The profile document can also be used in the advanced script to store further information that is to be carried forward between the different Integra events (similar to the stats.uservar property). However, any information stored in the user's profile remains available even after a particular Integra profile is executed. For instance values entered in fields of a custom dialog will be cached and the dialog will open with these (default) values the next time a profile using the same custom dialog form is executed.

134 StaticVars.Utilities [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export, Import

This property contains the subroutines and functions of the Utilities script library as methods of the Utilities class. Please refer to the Utilities script library documentation for more details.

The reason for the implementation of a Utilities class lies in Integra's capability of allowing the inclusion of custom script libraries. The way Lotus Script works with regards to script library inclusion (Use statement) is that subroutines and methods are only known to code that follows the Use statement for the included script library.

Any custom script library included into the CustomScript library of Integra as well as any code added into the CustomScript library itself, does not know anything about the subroutines and functions in the Utilities script library (as the Use statement for it follows later on the chain of included script libraries).

As the Utilities script library contains many useful functions, these have been used by many in the advanced script fields of the Integra profile. However, when the code becomes increasingly long and complex it gets to a point where moving the advanced script into a custom script library is necessary.

However, in all Integra version prior to 4.1 one would lose the ability to call the subroutines and functions in the Utilities script library when moving the advanced script code into a custom script library.

From Integra 4.1 onwards a developer can include the Utilities subroutines and functions by calling them the Utilities property of the StaticVars class (stats object). There is though a minor difference in the call of the subroutines and functions, in that all of them require to be called with a preceding "util" prefix, for instance the Utility function ReplaceStr(...) need to be called stats.Utilities.utilReplaceStr(...)

Note: it is recommended to use the new calling convention even for advanced code stored in the Integra profile to make code transition to custom script libraries seamlessly.

135 StaticVars.View [Class Property]

Version: 2.x to 4.7
Availability: Word, Excel, Text, Export

If the profile is set to use the "Selected Documents in Active View" option, then this property contains the back-end Notes view object of the currently opened Notes UI view. If the profile is set to use the "All Documents in View" option, then this property contains the back-end object of the view selected in the profile.

136 StaticVars.ViewNav [Class Property]

Version: 4.1 to 4.7
Availability: Word, Excel, Text, Export

StaticVars.ViewNav As NotesViewNavigator

Integra for Notes utilises all of the available Lotus Notes classes to process the to be exported documents. Which of the classes is used depends on the profile settings and the context the user is in.

The conditions are the following (evaluated at runtime in order of presentation):

Condition	Used Class	Exposed to UI (No = Advanced Script Only)	stats.Entr y available
export "Selected Documents in Active View" AND one or more (but not all) documents in the view are tick-marked	NotesDocumentCollection	Yes	No
stats.UseAllEntries = False	NotesViewNavigator	No	Yes
stats.ForceNavExp = True	NotesViewNavigator	Yes	Yes
View is NOT categorised	NotesViewEntryCollection	n/a	Yes
stats.SortGroup.Enabled = True	NotesViewEntryCollection	Yes	Yes
stats.DoSort = False	NotesViewEntryCollection	Yes	Yes
stats.FlatCat = True	NotesViewEntryCollection	Yes	Yes
stats.FstCatOnly = True	NotesViewEntryCollection	Yes	Yes
Otherwise	NotesViewNavigator	n/a	Yes

The difference in the use of these Notes object classes lies in the performance and in available features. The following table shows the differences:

Class	Performance	Usage
NotesDocumentCollection	Slowest compared to other classes.	If users require to select documents manually in the view. Not suitable for large amounts of to be exported documents.
NotesViewEntryCollection	Fastest class to process documents in the view.	Used for large data collection, where ONLY document entries are required to be processed.
NotesViewNavigator	Performance depends on other settings and the design of the view. It is generally faster than the NotesDocumentCollection, however, performance can be reduced when handling view category and subtotal entries	This class needs to be used when it is required to handle the view category and subtotal entries.

Some combinations of settings might create a situation where no documents can be exported, e.g. if a full text filter or selection formula is used in addition to the main selection method. This would be determined at runtime and the user will be notified.

137 StaticVars.WdFdTypes [Class Property]

Version: 4.3 to 4.7
Availability: Word, Export

Prior to version 4.3 Integra supported only Word mailmerge and Form fields. At runtime Integra would assume that both field types are in use and would scan the Word template for both supported field types.

Integra 4.3 supports now the following Word field types:

- Merge fields
- Form fields
- Bookmarks
- OCX fields

In addition to supporting more field types, it also allows through the profile UI to determine which field types are actually in use in the Word template. Selecting only the used field types will let Integra scan the template for contained fields of the selected types only, thus improving the runtime performance.

The WdFdType property reflects the profile's field type selection as a bitwise AND ...

Linked Merge fields in template = 1
 Linked Form fields in template = 2
 Linked Bookmarks in template = 8
 Linked OCX fields in template = 4

E.g. if Merge fields and Bookmarks are selected the value of WdFdType would be 9 (1 + 8).

138 StaticVars.WebActDoc [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Export

Notes back-end document object that contains the Web Activity document. The Web Activity document is created for all profiles run through a browser to store runtime relevant information, such as runtime error messages. More importantly it stores the Word or Excel file created during the export, that is to be pushed (downloaded) to the browser.

The Web Activity documents can be accessed via the Actions menu: Actions->Administration->Web Activity (to open this view, a user requires the Admin user role). A scheduled agent is cleaning up processed Web Activity documents after latest 24 hours. This agent is automatically enabled/disabled when the Integra database is enabled/disabled for Web usage through the Configuration screen.

139 StaticVars.WordAppl [Class Property]

Version: 2.x to 4.7
Availability: Word, Export, Import

The WordAppl property contains the by Integra at runtime created handle to the Word application COM object. This allows the Integra for Notes profile creator to use the properties and methods of the COM object class to do further processing of the Word document beyond the functionality delivered with Integra for Notes.

For the details please refer to Microsoft's documentation of the MS Word COM object classes. You can find them online on the Microsoft websites and if MS Office is completely installed also in the following file:

C:\Program Files\Microsoft Office\Office10\1033\VBAWD10.CHM

Note: the above path is only an indication where to look for the help file. The path and filename itself depends on the MS Office version and the location of the installation. The example shows the filename for MS Office XP in the default installation directory.

Example:
[Calling Word macros at specific events](#)

140 StaticVars.WordDoc [Class Property]

Version: 2.x to 4.7
Availability: Word, Export, Import

The WordDoc property contains the by Integra at runtime created handle to the Word document COM object. This allows the Integra for Notes profile creator to use the properties and methods of the COM object class to do further processing of the Word document beyond the functionality delivered with Integra for Notes.

For the details please refer to Microsoft's documentation of the MS Word COM object classes. You can find them online on the Microsoft websites and if MS Office is completely installed also in the following file:

C:\Program Files\Microsoft Office\Office10\1033\VBAWD10.CHM

Note: the above path is only an indication where to look for the help file. The path and filename itself depends on the MS Office version and the location of the installation. The example shows the filename for MS Office XP in the default installation directory.

Example:
[Removing Word template comments before Save As... or Send To...](#)

141 StaticVars.WriteMethod [Class Property]

Version: 3.x to 4.7
Availability: Word, Excel, Text, Export

The WriteMethod property indicates the method Integra is using to export data to Word or Excel as defined in the Advanced tab of the Integra profile. WriteMethod can be one of the following constants:

Excel export:		
Export_Using_COM	0	(Advanced COM)
Export_Using_XML	1	(XML, MS Office XP 2003+)
Export_Using_XSV	2	(Normal, Excel CSV format)
Word export:		
Export_Using_COM	0	(Advanced COM, default Integra 4.1)
Export_Using_WDS	2	(Word Data Source, Integra 4.2+)

Text export:		
Export_Using_CSV	0	(CSV format, Excel compatible)
Export_Using_XML	1	(XML)
Export_Using_TXT	2	(Custom delimited text format)

142 StaticVars.XlsAppl [Class Property]

Version: 2.x to 4.7
Availability: Excel, Export, Import

The XlsAppl property contains the by Integra at runtime created handle to the Excel application COM object. This allows the Integra for Notes profile creator to use the properties and methods of the COM object class to do further processing of the Excel application beyond the functionality delivered with Integra for Notes.

For the details please refer to Microsoft's documentation of the MS Excel COM object classes. You can find them online on the Microsoft websites and if MS Office is completely installed also in the following file:

C:\Program Files\Microsoft Office\Office10\1033\VBAXL10.CHM

Note: the above path is only an indication where to look for the help file. The path and filename itself depends on the MS Office version and the location of the installation. The example shows the filename for MS Office XP in the default installation directory.

143 StaticVars.XlsSheet [Class Property]

Version: 2.x to 4.7
Availability: Excel, Export

The XlsSheet property contains the by Integra at runtime created handle to the Excel worksheet COM object. This allows the Integra for Notes profile creator to use the properties and methods of the COM object class to do further processing of the Excel worksheet beyond the functionality delivered with Integra for Notes.

For the details please refer to Microsoft's documentation of the MS Excel COM object classes. You can find them online on the Microsoft websites and if MS Office is completely installed also in the following file:

C:\Program Files\Microsoft Office\Office10\1033\VBAXL10.CHM

Note: the above path is only an indication where to look for the help file. The path and filename itself depends on the MS Office version and the location of the installation. The example shows the filename for MS Office XP in the default installation directory.

Example:
[Removing Excel template comments before Save As... or Send To...](#)

144 StaticVars.XlsWBook [Class Property]

Version: 2.x to 4.7
Availability: Excel, Export, Import

The XlsWBook property contains the by Integra at runtime created handle to the Excel workbook COM object. This allows the Integra for Notes profile creator to use the properties and methods of the COM object class to do further processing of the Excel workbook beyond the functionality delivered with Integra for Notes.

For the details please refer to Microsoft's documentation of the MS Excel COM object classes. You can find them online on the Microsoft websites and if MS Office is completely installed also in the following file:

C:\Program Files\Microsoft Office\Office10\1033\VBAXL10.CHM

Note: the above path is only an indication where to look for the help file. The path and filename itself depends on the MS Office version and the location of the installation. The example shows the filename for MS Office XP in the default installation directory.

Example:
[Changing the order of Excel Workbook sheets](#)