#### Braiins Report Object Export Implementation Notes

26 May 2012

Table of Contents

[Purposes 1](#_Toc325786284)

[To keep a record of practical issues of implementing BROs. 1](#_Toc325786285)

[BRO Naming 2](#_Toc325786286)

[Purposes 2](#_Toc325786287)

[Mapping and Linking 2](#_Toc325786288)

[Conventions 2](#_Toc325786289)

[Brevity 2](#_Toc325786290)

[Common parts should have common names 2](#_Toc325786291)

[Structure 2](#_Toc325786292)

[Balance Sheet 2](#_Toc325786293)

[Profit & Loss Account 3](#_Toc325786294)

[Reconciling the Statutory P&L with the Detailed P&L 3](#_Toc325786295)

[Tricks of creating BROS 4](#_Toc325786296)

[Main steps 4](#_Toc325786297)

[Type (Set or Ele) 6](#_Toc325786298)

[BROS Dimensions 6](#_Toc325786299)

[Handling large numbers of Related entries 7](#_Toc325786300)

[Handling Strings in Tuples 8](#_Toc325786301)

[Operational points 8](#_Toc325786302)

[Modes of data entry 8](#_Toc325786303)

[Automatic versus Manual 8](#_Toc325786304)

[Data Readiness -Adjusted Closing Trial Balance versus Opening or Extended Trial Balance 8](#_Toc325786305)

[BROS Function – Sorting 9](#_Toc325786306)

[Operating and Non-Operating 10](#_Toc325786307)

[Issues arising 11](#_Toc325786308)

[Importing from 3rd Party Programs. 16](#_Toc325786309)

[Nominal Ledger Information 16](#_Toc325786310)

# Purposes

1. To keep a record of practical issues of implementing BROs.
2. Operational points
   1. To describe and considers modes of data entry

# To keep a record of practical issues of implementing BROs.

Small detailed points will be stored in the Comments section of the BROs table.

Where major issues arise, this will be the reference document to store such information

It is to be read in conjunction with the Bros.docx document.

# BRO Naming

## Purposes

The first level BRO names provide a unique reference system which can link internally to TxId and BRO only data, and both types of data can include Hypercubes and Dimensions.

They are used for referencing within the Report Generator part of BRAIINS.

Externally they provide the reference interface for mapping to other AP and Accounting systems CoAs and any other data they may contain.

## Mapping and Linking

It may seem odd to put this before listing out the BRO conventions and structures, but this a very important part of the system, and probably harder to change than an RG aspect, and has various aspects, such as whether an external item can be linked with 100% certainty to a precise TxId/Dimension combination, only to a TxId, or to a general TxId/Dimension combination, from which some manual intervention/decision will need to be made. This latter might mean allowing an imported piece of information to go to a RO filed, but generate a warning that this needs adjusting, and providing a list of legitimate choices.

Known facts

BROs have two reference systems, a Numeric one and an Name one (Alphanumeric)

Hypercubes and Dimensions have Numeric references

ands alpand HYrecubes

See also Importing from 3rd Party Programs.

## Conventions

### Brevity

If possible, a named should be reduced to the smallest number of characters that still allows unique identification.

So ***Investments*** can be shortened to ***Invest***, but not ***Inv***, since this might get confused with ***Inventory***.

### Common parts should have common names

Example, one can have Fixed Asset Investments and Current Asset Investments. It is tempting to abbreviate the former to ***FAI*** and the latter to ***CAI***. But it may be better to call both ***Investments***, or ***Invest*** to make it shorter. The rationale for this approach as is any type of investment can be recognised and searched on.

This should also make it easier to have computer searches and masking done.

## Structure

Note that I have avoided taking the top level (0) to be the name of any report type, but the nature of the information at the highest level. So for a Balance Sheet, as can be seen below, there are three names at level 0. These are Assets, Liabilities and Equity. This is quite adequate for determining any total or subtotal amounts.

### Balance Sheet

|  |  |  |
| --- | --- | --- |
| **Level** | **Name** | **Notes** |
| 0 | Assets |  |
| 1 | Fixed |  |
| 2 | IFA |  |
| 2 | TFA |  |
| 2 | FAI |  |
|  |  |  |
| 1 | Current |  |
| 0 | Liabilities |  |
| 0 | Equity |  |
| 1 | Shares |  |
| 1 | Reserves |  |
| 2 | P&L |  |

### Profit & Loss Account

|  |  |  |
| --- | --- | --- |
| **Level** | **Name** | **Notes** |
| 0 | RevOp |  |
| 1 | Turnover |  |
| 1 | OtherOpIncome |  |
| 0 | ExpOp |  |
| 1 | Stock |  |
| 1 | OwnWorkCapitalised |  |
| 1 | GoodsMaterials |  |
| 1 | CarriageComms |  |
| 1 | Machines |  |
| 1 | Premises |  |
| 1 | MarketingAdvertising |  |
| 1 | FeesServicesProf |  |
| 1 | Other |  |
| 1 | Financial |  |
| 1 | Personnel |  |
| 1 | DepnAmortImpair |  |
| 0 | RevNonOp |  |
| 1 | SubAssocPart |  |
| 0 | NonOpFinanceInvestment |  |
| 1 | InvestmentGainsLossesNetItemsHeading |  |
| 2 | InstrsNetBeforeTax |  |
|  |  |  |
| 0 | GainsLossesFromSaleOrTerminationOperations |  |
|  |  |  |
|  |  |  |
| 0 | OperatingActivitiesExceptionals |  |
|  |  |  |
|  |  |  |
| 0 | TaxOnOrdActivities |  |
| 1 | TotalUKForeignCurrentAfterAdjustsRelief |  |
| 1 | TotalUKForeignDeferred |  |
| 1 | AdvancedCorporationWrittenOffBack |  |
|  |  |  |
| 1 |  |  |
| 0 | ExpDividends |  |

### Reconciling the Statutory P&L with the Detailed P&L

At present there are inconsistencies between the Statutory P&L and Notes versus the Detailed P&L.

It should be possible for any standard CoA to reach up into both the Statutory and Detailed P&L taxonomy and access all Money (DE) labels. But we know that this cannot be done directly. The proposed revised Detailed P&L will not solve this problem; it will merely re-arrange the problem.

It is and will remain the case that there will be Taxonomy Labels which cannot be accessed by a conventional CoA structure.

But BRAiiNs can handle these inconsistencies because of the extra features in BROs. One of the key ones for handling this particular mess is the Related Field. This allows us to “jump” from one structure to another. I need to work out the precise details of this, but in essence one builds a substructure of BRO names (Sets and Elements) below the TxId labels and then creates cross relationships at an intermediary level.

Tasks

Need to create a detailed of exactly what the differences are

# Tricks of creating BROS

There are various techniques which can save time and improve accuracy when creating BROS.

## Main steps

1. Download the TxIds using the BRO Prep Lookup Menu. (Save file and open in Excel).
2. Cut and paste desired section to a new Worksheet.
3. In New Worksheet
   1. Insert a few blank rows at the top
   2. Copy and paste Header Row and next line or two to provide column titles and lead values from Name Level 0 to required Name level from existing BROS GAAP spread sheet.
4. Move original section across far enough that it aligns from the correct Name column; typically N1
5. Copy and paste down known fixed values re Name 0, N1 as required e.g. Assets in Name 0.
6. Abstract Elements – not Headings.
   1. Look for TxIds which will have an “[A]” at the start and usually not the word “Heading” at the end. These rows can be removed. Beware that just occasionally there are some which end with “Heading” but should still be removed
      1. Example: In the table below, the first row is a legitimate Heading which can be used to form part of BRO names. But the second, although it ends with “heading” is not, and it plus the others shown below should all be removed

|  |
| --- |
| [A] 3698 OwnSharesHeading |
| [A] 5406 DimensionMembersWhichMustBeUsedToDistinguishTotalSharesIndividualShareTypesClassesHeading |
| [A] 5350 AllShareClassesDefault |
| [A] 5348 AllOrdinaryShares |
| [A] 5535 OrdinaryShareClass1 |

1. Abstract Elements – legitimate Headings.
   1. Look for Heading TxIds (which will have an “[A]” at the start and “Heading” at the end) and move them to the left to the first Tx field.
   2. Edit the name to something succinct and (where applicable) consistent with any conventions already in use.
   3. Copy this down to all Tx fields below, where there is a TxId field to the right. (When you encounter a TxId field in the column with no TxId in the row to the right, this means a new Name is required - this will be obvious, you will see another description beginning with an “[A], or you will see a Tuple type).
2. Tuples
   1. Recognition – The description will be in the wrong column of Period Type, and will end with “Grouping”.
   2. Actions
      1. **Copy** this across and edit as with Abstract Headings.
      2. Copy down the Tx columns below, following same rules as Abstract Headings.
      3. It may be useful to retain the Tuple number and TxId
         1. Tuple itself
            1. Filter all [Tuple nnn] rows
            2. Use the blank Description column and Concatenate Text+Tuple+TxId
            3. Then Copy and Paste Text to hard code
            4. Delete the now unwanted Text ,Tuple and TxId values
         2. Tuple lines
            1. Filter all [T nnn] values.
            2. Use the blank HyId column Concatenate Text+Tuple value
            3. Then Copy and Paste Text to hard code
            4. Then Cut this and move to the Description column.
            5. Delete the now unwanted Text and Tuple values
      4. Delete the Concrete value on the Tuple line
      5. Move the original description to align with the other descriptions
      6. Cut and paste the Tuple value to the end of the description (this is just done for ease of reference)
      7. Do the cut and paste for second and subsequent tuple values; these will be shown in a shortened form e.g. [T 19] rather than [Tuple 19]
3. Moving columns: At this point, the column numbers and order from the BRO Prep Lookup will still not match the order in BROSUK-GAAP. For the most part it is obvious what needs to be moved where and what can also be deleted.
   1. It is necessary to move the columns to the right at this stage to leave the columns N1 to N8 only containing legitimate data. Otherwise when you come to calculate what Type (Set or Ele) applies to each row, the blank cell check will be inaccurate.
   2. **Period Type** (no column header). The contents column filled **Instant** and **Duration** can be cleared as long as there is no other text in any of the cells (blanks can be ignored). This is one of the columns re-created by a Braiins import.
   3. **StartEnd column**. The contents of column containing [Start Label], [End Label] or [StartEnd] can be moved across to the StartEnd column (Col AF). The [End Label] can be deleted in due course. They will be duplicating the TxId already in [Start End] rows. But would suggest that this is done on an individual basis, since it gives one the opportunity to review whether there are matching Start and End TxId. *(In due course, you will need to enter the appropriate command and TxId values, but this is beyond scope of this section)*.
   4. Ultimately we want to put the column containing Taxonomy Names in to the **Descr** column; but this will be out of view on even the largest screen which is a nuisance if one is dealing with BRO Dimensions; where visual confirmation is very useful.
      1. If there are no BRO Dimensions to deal with, move contents straight to Descr column (Col AJ)
      2. If there are BRO Dimensions to deal with, move contents temporarily to the BD column (Col O), since this column is never needed at this stage. It is completed within Braiins as part of the ***Build Bro, Tuple & Zone Structs*** action. Since currently BD column is likely to have other stuff in it, most likely Data Type information, best to move contents of the five columns starting at BD three columns to the right. This will correctly align Data Type and Sign. Then just move what will now be the contents of the RO column back across to the BD column.
   5. **TxIds** – the [C] part of the TxIds can be removed, and contents moved across to the TxId column.
   6. **HyIds** – these can be moved across to the HyId column.
   7. **StartEnd**. These need moving across to the StartEnd column.
4. **Filling in fields**:
   1. **Data Type** – Check that there are no blank fields. These will have arisen when setting up Abstracts and Tuples, and are non TxId BROS. But they still need to have a Data Type. Most commonly this will be “Money”. Best way to check is to look at rows immediately below. Copy “Data Type” and “sign” (where applicable. Do not copy “HyId” and “Excl Dims” fields since these never apply to non TxId BROs.
   2. **Acct Types**. This will typically be BS, PL, CF. Although not an active field, it is a useful visual indicator. It applies to all Data Types.
   3. **Post Type**. There are three possible values, DE, Sch or left blank. ***DE and Sch only apply to Money type*** values. It differentiates what is considered part of the Double Entry (and therefore part of the Trial Balance), and what is considered supporting schedules, which will be related back to DE figures as appropriate.
   4. **Excl Dims.** What needs to be entered here will depend on the nature of the **HyId** in use, and what if any BRO Dimensions are active. Some common scenarios are:
      1. HyId which allows BRO Dims, but not none used **–** enter 43,44
      2. HyId which allows BRO Dims, Dim Function in use **–** enter 44
      3. HyId which allows BRO Dims, Dim Ageing in use **–** enter 43
      4. Any other type of HyId – leave blank (there are some rare exceptions to this rule).
   5. **Except**. This will always be left blank unless the this BRO is an exceptional item (and meets other criteria – see BROS.docx)
   6. **Sum Up**. This will either be “No”, “+” or “NA”. You may want to manually enter “No” for some BROS that you do not want totalled back up the tree. For any others, at this stage it is quicker to leave them blank and do a BRAIINS Import. The error report will quickly identify any that need adjusting, and why.
   7. **Context**. The only entry normally needed to be entered on an Import is Period. This applies all TxId BROS that are not part of a BRO Dimension (other than at the total level). So a quick way to complete them is:
      1. **Filter on TxId** – Un-tick ***Blank***
      2. **Filter on BD** – tick only ***Blank*** (This only applies if using BROS Dimensions, and the BD values have been entered. (See BROS Dimensions further on)
      3. Copy and paste “Period” down the **Context** column.
   8. StartEnd. These can be tricky because they may well make reference to a section of BROS that you are NOT currently importing. e.g. only doing ***Assets – Current***, and the ***inbuilt routines are very clever at spotting errors or fudges***. Solution
      1. If at least one TxId that is referred to exists in the current import, then put this TxId value in. For instance for Cash the ***StartEnd*** entry should be ***PostEnd 575,541 2646,2647,2648,2649***. But only the first two will be in this import – the others are in the Cashflow Section. Just put in ***PostEnd 575,541***, and put a note in the Comments field that this needs completion when all the parts are assembled, preferable with an easy to remember and search set of charactyers such as OS for outstanding. So in this case the entry could read ***OS Need to put in StartEnd column: PostEnd 575,541 2646,2647,2648,2649***.
      2. If no TxId exists in the current selection, at least one of the required TxIds will have to be copied across from an existing BROSUK-GAAP table. You will need to make sure that this added one leads back up to a level 0 either by copying across required rows from existing BROSUK-GAAP, or creating some dummy.
      3. This can trickier than i
5. **Type** – Set or Ele. Refer forward to separate section and then return to the next step.
6. **Other columns** e.g. BD, Incl Dims, DiMeId, Totalling, Sum To, Related, can be ignored. If any information needs to be entered in to any of them it can be easily done after an Export.
7. **BROS Dimensions**. Check if there are any. If so refer forward to next section and then return to this section, next step.
8. Copy and paste in to BRAIINS.
   1. Repeat above until all Error messages have been dealt with.

### Type (Set or Ele)

There are two issues here

* Is a Type Set or Ele(ment)
* Has a Type row been missed e.g. steps down must be in units of 1.

The correct Type can be visually recognised by looking at the line below and seeing if an additional BRO Name field has been used. But this can be tedious if dealing with many 10s of row.

Quick reliable method is as follows:

In the Type column enter the formula =IF(D7<D6,"Set","Ele")

In the Level column enter formula =IF(D6-D7>1,"GAP","")

In the BRO Name column enter the formula =COUNTBLANK(E6:M6)

(The actual column and row reference will depend on how many blank lines were inserted at the top, and whether any additional columns were inserted to the left to help sort row re Bros Dimensions)

This solves both problems

1. The Type field will be correctly shown as Set or Type
2. If a Step has been jumped, this will become obvious looking at the BRO Name fields and noting any which decrease by more than 1. (Or better still if you have used the formula above, you will see the word GAP in the Level column).

Obviously one must:

* Copy down all the fields to the required rows, and
* Copy and paste Text only in the Type cells to preserve the “Set” and “Ele” values rather than the underlying Formulae.
* Manually change the Type for the last row to “Ele” (it will calculate as “Set”) Or just extend the Count command one row further).

### BROS Dimensions

This relates to BROS Dimensions, such as Ageing or Function.

The key point is that the order wants to be sorted by BRO without Dimension, and then matching BROS with Dimension, in the correct dimension order.

If the TxIds were left in the default order, this would all need to be manually which is slow, tedious and error prone.

The objective is to get them sorted in the right order, with all correct details completed (or removed) with simple block commands.

1. Need to copy the Concrete Description down the TxId columns following the same rules as with Abstract TxIds (one never normally needs do this with Concrete TxIds since they will self-populate. So only normal reason would be to use a short Name other than the one the Taxonomy provides)
2. Repeat for Dimensions
3. BRO Dimensions at the Total level e.g. Debtors – Make common changes (see 10) above.
4. BRO Dimensions at a at an Individual level e.g. DebtorsDueWithinOneYear, DebtorsDueAfterOneYear
   1. **Type** will be ***Map*** for all
   2. Remove any entries from the following columns because they do not use BRO Dimensions:
      1. Data Type
      2. Sign
      3. Acct Types
      4. Post Type
      5. RO
      6. HyId ???
      7. Excl Dims ???
      8. Incl Dims
      9. DiMeId
      10. Except
      11. Amort
      12. Sum Up
      13. Totalling
      14. Sum To
      15. Related
      16. StartEnd
      17. Context **HyId** values for Money TxIds
5. Columns for Fill and Sort. These two actions each require a column. You can create a new column for each e.g. new Col A and Col B, but bear in mind that at this stage there are three existing columns on the left which are unused, ID (Col A), Level (Col C) and BRO Name (Col D). So these can be used, and then the contents deleted when the exercise is complete.
6. In left most new or unused Column (example Column A), put Fill values against each BRO Dimension, including TxId without any Dimension e.g. Debtors
7. Check if count for all Dimensions is the same.
   1. It is assumed that the order of the TxIds for each dimension will already be in the same order (as defined by the UK GAAP Taxonomy, but just do a quick visual check to make sure this is the case)
   2. If either the count or the order are not the same, look to see which are missing or out of order, and adjust subsequent Fill numbers. For example in Debtors, CalledUpShareCapitalNot-paidCurrentAsset, appears as both “total” and “Less than 1 year”, but not “Greater than 1 year”.
   3. I cannot see any fully automatic way of doing this. One cannot sort by Element because that may be different between the Dimensions e.g. “Debtors” matches “DebtorsDueWithinOneYear” and “DebtorsDueAfterOneYear”. Just these three by themselves would come out in the wrong order. In addition there are other Labels starting with the Label “Debtor..” that can get interspersed. Nevertheless it might help to visually pinpoint the “missing” items. If not just scan manually down the column.
   4. Whichever method is used, put them back in default order and adjust numbers i.e. delete and move up, so all numbers match the equivalent TxId description
8. In a new or unused column to the right of the one used for Fill e.g. Col B, enter a value against each Dimension in the required order e.g. for Debtors
   * A for all Un-aged
   * B for all < 1year
   * C for all >1 year.
9. Sort. One can now sort by Fill e.g. Col A and then by Dimension order e.g.Column B
10. Check if count for all Dimensions is the same. (part 2)

### Handling large numbers of Related entries

OS

### Handling Strings in Tuples

Strings in Tuples are one of the instances when a repeated TxId is NOT wanted.

The BROS import routine keeps a record of all the possible Tuples that a String can belong to.

It is best to keep the TxId strings in some common statement area;, and simply strip out the duplicates when advised by the BRO Import routine e.g.

|  |  |  |
| --- | --- | --- |
| BroName | TxId | DataType |
| AcPol.FinancialRiskCapitalRiskManagement.FinancialInstruments.Credit.OnFinancialAssets.DescrSpecificClass | **1563** | **String** |
| AcPol.FinancialRiskCapitalRiskManagement.FinancialInstruments.Credit.OnFinancialAssets.MaximumExposureSpecificType | 3189 | Money |
| AcPol.FinancialRiskCapitalRiskManagement.FinancialInstruments.Credit.OnFinancialAssets.DescrCollateralHeldSecurityOtherEnhancementsSpecificType | 1346 | String |
| AcPol.FinancialRiskCapitalRiskManagement.FinancialInstruments.Credit.OnFinancialAssets.DescrQualitySpecificTypeThatAreNeitherPastDueNorImpaired | 1360 | String |
| AcPol.FinancialRiskCapitalRiskManagement.FinancialInstruments.Credit.OnFinancialAssets.WhichTermsRenegotiatedCarryingValue | 2088 | Money |
| PL.Interest.Receivable.AmountRecordedImpairAllowanceAccountForSpecificTypeFinancialAsset | **1563** | **String** |
| PL.Interest.Receivable.AmountRecordedImpairAllowanceAccountForSpecificTypeFinancialAsset | 223 | Money |
| FinInstruments.FairValueFAFL.GeneralDescrValuationFinAssetsLiabs.MethodAssumptionsUsedInDeterminingInstrValues | **1563** | **String** |
| FinInstruments.FairValueFAFL.GeneralDescrValuationFinAssetsLiabs.MethodAssumptionsUsedInDeterminingInstrValues | 1448 | String |
| FinInstruments.FairValueFAFL.GeneralDescrValuationFinAssetsLiabs.ChangeInTechnique | 1324 | String |
| FinInstruments.FairValueFAFL.GeneralDescrValuationFinAssetsLiabs.PolicyOnRecognisingDifferenceBetweenInitialRecognitionEstablishedByTechniqueThatDate | 3878 | String |
| FinInstruments.FairValueFAFL.GeneralDescrValuationFinAssetsLiabs.AggregateIncrInYetToBeRecognisedArisingFromDifferenceBetweenFromTechnique | 145 | Money |
|  | **1563** | **String** |
| InvestmentGainsLossesNetItems.FinInstrsNetBeforeTax.ImpairReversalAssetsIncomeStatementAnalysis.Impair.HeldtomaturityInvests.DirectlyReducedToSpecificClass | 2560 | Money |
| InvestmentGainsLossesNetItems.FinInstrsNetBeforeTax.ImpairReversalAssetsIncomeStatementAnalysis.Impair2.AvailForSaleInvests.SpecificType | 2592 | Money |

The string TxId 1563 shold be placed in some “common” part such as Statements, and removed from each of these parts.

# Operational points

## Modes of data entry

This splits into to two areas

Automatic versus Manual

Closing Trial Balance versus Opening or Extended Trial Balance

### Automatic versus Manual

This is simply whether data is imported directly in from an electronic source via a mapping file, or posted manually by a User.

The two methods are not mutually exclusive, but issues of control and feed-back arise where both methods are used.

### Data Readiness -Adjusted Closing Trial Balance versus Opening or Extended Trial Balance

To re-iterate, BRAIINS is not designed to be a general accounting system. Its basic unit or level of “granularity” is a GL balance.

But it should allow for the varying degrees of “final account readiness” from the Entity.

The degrees might be described as follows (note that they are in decreasing degrees of completeness)

**Adjusted Closing Trial Balance**

This is what would be expected from larger Entities with more sophisticated accounting systems or from Agents who use their own processes or tools to get to this point.

This means that all BRAIINS would need to hold is a single Closing Trial Balance for each year.

**Unadjusted Closing Trial Balance**

Essentially this is what would typically be produced by a SME entity using a program such as Sage Line 50, TAS Books or Intuit. The trial balance merely reflects the balance of the daily operations without any special adjustments to reflect period ends or adjustments and additions to provide required accounting information for regulatory accounts

This means that BRAIINS would need to hold a Trial Balance plus various forms of Adjustment Journals. These Adjustment journals could be shown in the form of an Extended Trial Balance, but this is not essential.

End of period entries would cover:

Adjusting entries

* Unexpired costs
* Unearned income
* Unrecorded expenses
* Unrecorded revenues

Change in value of Fixed Assets

* Depletion: Depreciation/amortisation
* Impairment: Provision for depreciation

Change in value of current assets

* Valuation allowances
* Provision for depreciation (AR, Inventories, Current Investments)

Ending inventory

Error correction

Closing entries

In incorporating these entry types, BRAIINS can also carry out much more data validation e.g. there is an entry in an Inventory NL account, but nothing shown in the Ending Inventory Journal. Or even more simply, no entries have been posted in any of the Adjusting Entry type journals = highly unlikely.

## BROS Function – Sorting

If sorting out a load of BROS that involve e BROS Function there are a number of things that can make this easier.

Assuming that a whole mass of TxId have been downloaded using *Bro Data Prep Version of Lookup of UK-GAAP Taxonomy Element(s)* the following steps help

1. Keep the TxIds in downloaded order
2. Do the normal steps (see above).
3. Create or Use blank a column to Fill with the number of rows in each Function e.g. Total, Less than 1 year, Greater than 1 years or if PL, Total, Cos, Distrib, Admin.

**Incomplete Records**

Essentially there are no meaningful accounts, just bags of source and some rough schedules.

This is the most complex for BRAIINS to handle. Essentially it is dealing with Daybooks (=detailed transactions) rather than Trial Balances (=summarised figures).

I am going to leave this area suspended as at the time of writing (18 April 2012).

The reasons are:

1. Suspect very few entities actually work from incomplete records; and certainly if they are ones needing to report under UK GAAP (or an IFRS).
2. Therefore we could waste a lot of time just working through the requirements (let alone the much greater work involved should we bother to construct any additional modules) for no practical value.

Revision

Consider that we avoid doing Incomplete Records at all and merely direct any User wanting to use such an entry method to one of the On-Line systems designed specifically for this purpose.

We then just import the trial balance.

Potential programs

HMRC are now listing programs that meet their

<http://www.hmrc.gov.uk/softwaredevelopers/mobile-apps/record-keeping.htm>

**QuickFile**

http://www.quickfile.co.uk/

Looks neat and it is free

NB. The founder/programmer behind QuickFile, Glenn Drake, has a sister company, Company Wizard.

<http://company-wizard.co.uk/>

He mentions using Shoebox and Dropbox to store copies documents.

<http://www.quickfile.co.uk/help/Receipt_Archiving_with_DropBox_Shoeboxed.aspx>

Shoebox

<https://www.shoeboxed.com/sbx-home/>

# Operating and Non-Operating

Both UK GAAP and UK IFRS differentiate between Operating and Non- Operating Revenue and Costs.

(You will note that I have currently used this as the major split between Costs and Revenue groups in BROS at present).

The significance is follows:

Operating relates to the major areas of business of an entity

Non-operational relates to the general handling of ancillary and usually financial aspects e.g. the putting of surplus cash into interest bearing accounts.

Not only does the P&L account make this distinction, so does the Cashflow. So not only is it important to differentiate, but it must (or should) be used as a basis for reconciling what is happening within the P&L to what happens in the Cashflow.

What makes this whole area fraught, is that one cannot simply say for instance that interest or dividends paid or received must be Non-Operational, or vice versa. Depending on the type of business, and the even circumstances within a business, something like “Bank Interest Paid” could be either. (Whether it could be both in the same Entity is more problematic again).

International Accounting Standard 7 (IAS7) does not prescribe any particular classification

Extract from International Reporting Standard – A Practical Guide” by Alan Melville p 242

*“IAS7 requires that cash flows arising from interest received, dividends received, interest paid and dividends paid should be shown separately in the cash flow statement. However, there is no consensus as to the classification of these items. Interest received and dividends received are usually classified as cash flows arising from investing activities, but the correct treatment of interest paid and dividends paid is less clear. For instance:*

1. *Interest paid on a loan obtained for operating purposes might be classified under the operating activities. Alternatively, it could be argued that interest paid should be classified under finance activities since it is a cost of obtaining finance.*
2. *Dividends paid might be classified under operating activities so that users may determine an entity’s ability to pay dividends out of operating cash flows. Alternatively, it could be argued that dividends paid are a cost of obtaining financial resources and so should be classified under finance activities.*

*IAS7 does not prescribe any particular classification for these items but does require that each of them should be classified in a consistent manner from period to period.*

*Cash inflows and outflows arising from income taxes must be disclosed separately and should be classified under operating activities unless specifically associated with investing or financing activities.”*

So looking back at this from the Entity’s point of view; having an account called “Bank Overdraft Interest” is easy to understand. But it would be far less easy for a bookkeeper to know whether some bank interest should be classified as Operating or Non-operating. (It may well be that this distinction is drawn at the end of the year, and applied in the form of a percentage or ratio split, just as might be the case of some other Natural Accounts between the Functions of CoS, Distribution or Admin.

Likewise having Bank Loan Interest appearing as 317 and 318 and then 410 and 411, as it does in SAPA is also less than obvious and user friendly (Note by the way that SAPA contains the advice “*(use a/c's 410 & 411 for statutory disclosure)*”. In fact this statement is inaccurate, either or both are required for correct statutory disclosure. I think that this advice has been in their since the early to middle 1990s.

Now we already have our expenses classified by their Nature e.g. wages, and then by their Function e.g. Administration.

Really, any Natural Account which can be subsequently divided as to Function is by definition an Operational Cost. Or we might say that we could add a 4th Function “Finance and Investing”, and simply know that this was the only Non Operating Function

We could look at it in at least three ways

**Method 1**

CoS

Distribution

Admin

Finance and Investing

**Method 2**

Operating

* Cos
* Distrib
* Admin

Non-Operating

**Method 3**

Keep the two areas quite distinct. So leave BROS Functions as is, and adds additional Dimensions for Operating and Non Operating.

In theory I guess an Operating Cost Bank Charge could be split between Cos, Distrib. and Admin.

So Method 1 is probably the truest and the most simple

# Issues arising

|  |  |  |
| --- | --- | --- |
| Topic | Ref | Details |
| Money-Instant-Start/End with Ageing | 1 | When BROs was being designed we did not think there would (or could) be a valid combination of a Money Instant with separate Start End values which also required Ageing (BRO Dim 44)  But Obligations under finance leases and hire purchase contracts does appear to require both  [C] 3387 [H 13] [Money Cr Instant] [StartEnd] Obligations under finance lease and hire purchase contracts  [C] 3389 [H 1,13] [Money Cr Instant] [StartEnd] Obligations under finance lease and hire purchase contracts within one year  [C] 3388 [H 13] [Money Cr Instant] [StartEnd] Obligations under finance lease and hire purchase contracts after one year  But within Balance Sheet Creditors we do have such a thing.  This is strange because the level above Obligations FLHP, Creditors, is a simple Money Instant.  In addition when thinking of the schedule nature of the ObligationFLHP it is very odd to separate by period.  Note in the extract below that where there is further analysis of the Obligations FLHP >1 into 1-2, 2-5 and >5, none of these Instants have Start End  [A] 3390 Obligations under finance leases and hire purchase contracts - Maturity Profile  [C] 3389 [H 1,13] [Money Cr Instant] **[StartEnd]** Obligations under finance lease and hire purchase contracts within one year  [C] 3392 [H 1] [Money Cr Instant] Obligations under finance leases and hire purchase contracts between one to two years  [C] 3393 [H 1] [Money Cr Instant] Obligations under finance leases and hire purchase contracts between two to five years  [C] 3391 [H 1] [Money Cr Instant] Obligations under finance leases and hire purchase contracts after five years  [C] 1593 [H 1] [String Duration] Description of terms of repayment and rate of interest for finance leases and hire purchase contracts after five years  [C] 2265 [H 1] [Money Dr Instant] Future finance charges  The BrosStartEndWip SSs do show 3387,3389 and 3388 as Start/End Instants and list the Sum End TxIds required.  Unfortunately I failed to realise the significance of the <1 and >1.  Scope of the problem  I cannot see any other TxIds involving Ageing that have StartEnds  Possible solutions   1. Modify BROS Import php to handle this.   But this is not a 5 minute job, and it is only required for two (related) TxIds   1. Construct special BROS import lines   I have dealt with the CashFlow items by no longer treating them as Ageing Dimensions.  This works fine in that   1. The Cash Flow can still sum up. 2. It is dubious that each Ageing item was directly comparable (in contrast to all other Creditors and Debtors. E.g.   Not sure if:  CapitalElementFinanceRentalPayments (TxId 505) is the same as  IncrInFinanceHirePurchaseDueWithinOneYearReflectedInCashOutflowInflow (TxId 2687) and  IncrInFinanceHirePurchaseDueAfterOneYearReflectedInCashOutflowInflow (TxId 2684)  Likewise not sure if  NewFinance (TxId 3322) is equivalent to  IncrInFinanceHirePurchaseDueWithinOneYearFromOtherNoncashChanges (TxId 2686) and  IncrInFinanceHirePurchaseDueAfterOneYearFromOtherNoncashChanges (TxId 2683)  Have treated each ObligationFLHP Set as a non-Taxonomy Bro  For the Balance sheet I have entered the following TxIds as straight elements with no Ageing Dim 44   |  |  |  |  | | --- | --- | --- | --- | | FinLeasesHPContractsAll |  |  | 3387 | | FinLeasesHPContractsLess1Yr |  |  | 3389 | | FinLeasesHPContractsMore1Yr |  |  | 3388 |   Then created three further non-Taxonomy BROs and should them as Related To each of the above.  This seems to work fine for the Element, but not for the two Map Bros. Get import error message that cannot use Braiins Dimension maps with non-Taxonomy elements   |  |  |  | | --- | --- | --- | | 1635 | Map BS Creditors FLHP <1 | This Braiins Dimension Map is missing a TxId. All maps are expected to have a TxId. | | 1636 | Map BS Creditors FLHP >1 | This Braiins Dimension Map is missing a TxId. All maps are expected to have a TxId. |   So questions then are   1. Ignoring the non TxId aspect for a moment, does this fudge actually work. 2. If so,    1. Is it easy to allow BROs Dims to work with non-Taxonomy Bros    2. Would this be useful in its own right e.g. for users to create schedules.   **ANSWER**  **DJH added ability for BRO Dims to handle Non TxId**  Now looks like below (extract)   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Type** | **Level** | **Bro Name** | **BD** | **TxId** | **HyId** | **Excl Dims** | **SUp** | **Related** | **StartEnd** | | Set | 3 | Liabilities.CreditorsBorrowings.CredTot.ObligationsUnderFinanceLeaseHirePurchaseContracts |  | 3387 | 13 | 43,44 | + |  | SumEnd  505,2681,2738,3322 | | Ele | 3 | Liabilities.CreditorsBorrowings.CredTot.ObligationsUnderFinanceLeaseHirePurchaseContractsWithinOneYear |  | 3389 | 13 | 43,44 | + |  | SumEnd  2680,2685,2686,2687 | | Ele | 3 | Liabilities.CreditorsBorrowings.CredTot.ObligationsUnderFinanceLeaseHirePurchaseContractsAfterOneYear |  | 3388 | 13 | 43,44 | + |  | SumEnd  2679,2682,2683,2684 | | Ele | 3 | Liabilities.CreditorsBorrowings.CredTot.FLHP |  |  | 13 | 43 | + | Equal To: Liabilities.CreditorsBorrowings.CredTot.ObligationsUnderFinanceLeaseHirePurchaseContracts |  | | Map |  | Liabilities.CreditorsBorrowings.CredTot.FLHP:Ageing.<1 | <1 |  |  |  |  | Equal To: Liabilities.CreditorsBorrowings.CredTot.ObligationsUnderFinanceLeaseHirePurchaseContractsWithinOneYear |  | | Map |  | Liabilities.CreditorsBorrowings.CredTot.FLHP:Ageing.>1 | >1 |  |  |  |  | Equal To: Liabilities.CreditorsBorrowings.CredTot.ObligationsUnderFinanceLeaseHirePurchaseContractsAfterOneYear |  |   Note re the above cases also need to add Related To data.  The summing status will need to be reviewed. May well double sum on the above settings. |
| BRO Ageing when no TxId | 2 | Related to the above is the situation where a TxId needs the BRO Ageing, but it only exists in an aged state, i.e. the equivalent to a MAP with no equivalent Ele or Set.  The only cases that I have come across relates to Creditors.   |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Type** | **Le** | **Bro Name** | **TxId** | **Data Type** | **Sign** | **AcTy** | **PoTy** | **HyId** | **Excl Dims** | **Context** | | Set | 3 | Liabilities.CreditorsBorrowings.CredTot.DividendsProposed | 1781 | Money | Credit | BS | DE | 13 | 43 | Period | | Map |  | Liabilities.CreditorsBorrowings.CredTot.DividendsProposed:Ageing.<1 |  |  |  |  |  |  |  |  | | Ele | 4 | Liabilities.CreditorsBorrowings.CredTot.DividendsProposed.Ordinary | 1783 | Money | Credit | BS | DE | 13 | 43 | Period | | Map |  | Liabilities.CreditorsBorrowings.CredTot.DividendsProposed.Ordinary:Ageing.<1 |  |  |  |  |  |  |  |  | | Ele | 4 | Liabilities.CreditorsBorrowings.CredTot.DividendsProposed.CumulativePreference | 1782 | Money | Credit | BS | DE | 13 | 43 | Period | | Map |  | Liabilities.CreditorsBorrowings.CredTot.DividendsProposed.CumulativePreference:Ageing.<1 |  |  |  |  |  |  |  |  | |
| Structure | 3 | Background.  BROs incorporate all the aspects one would expect from a CoA plus all other information that could be possible held for an Entity. It also has its own built in logic to deal with relationships between different pieces of related information e.g. a “NL” Set to hold totals for a number of “NL” Elements or sub-sets.  Concrete taxonomy elements can be viewed in two main ways or Types; Presentation or Definition. Often they are identical. Many TxIds appear more than once within the Taxonomy (in either view). But within BROs they can only appear once. (BROs is looking to store the value so should never have duplicates of the same TxId/HyId combination.  The chosen method of building the BROs structure has been using the Presentation view.  Where a TxId/Hy Id combination does appear more than once we need to decide which “appearance” (if either?) is the best one to go for. Initially whether it makes any functional difference e.g. ability to create complete sets.  Have experimented with two main methods so far.   1. Link back to the Primary report e.g. make *Provisions for liabilities and charges* a subset of Balance Sheet. 2. Follow order in the presentation tree, ignoring Abstract levels above. So *Provisions for liabilities and charges* becomes a Level 0 Set.   The first method makes it clear how everything ties back to the main reports. But is does tend to imply a direct relationship with the Primary Report layout. The second method uses less Name levels to present the same information. The second method implies that the SumTo or Related columns might be required more often to tie back to Primary reports. But this is not certain. The BROs do not really need to make all these connections, many of this type could be left to the RG. |
| TxId with no Hypercube  TxId 5462 | 4 | TxId 5462. It is shown as not belong to any HypedrCube.  But if add to BROs with no HyId it gets rejected. Equally if say it belongs to HyId 1, it gets rejected because it does not.  02 - Business Report Information  [A] 5360 Business report information [heading]  [C] 5462 [EntityAccounts Duration] Entity accounts type  [C] 5361 [H 1] [String Duration] Business report name  [C] 5387 [H 1] [String Duration] Description of business report |
| Money TxId wrongly defined as Duration not Instant | 5 | *4550 13 Money Debit Duration [Start Label] StocksPaymentsOnAccountTradingActivities*  *2709 13 Money Debit Duration IncreaseDecreaseInPaymentsOnAccountForStocksTradingActivities*  This is clearly a balance, so one would expect it to be an Instant.  Same logic re TxIds 4547,4548 and 4549. |
| Operating and Non-Operating | 6 | Legitimate vertical separation within The Income Statement (P&L).  UK GAAP does have a Dimension for Operating Costs. |
| Accounts: Natural and Function | 7 | Creating the inter-relationship  *(CW 26/5/2012 – not sure what I meant when I entered the above – have left inb case in case I recall meaning)* |
| Concrete item wrongly described as Abstract | 8 | [A] 1348 [T 16] Description of compound financial instruments with multiple embedded derivatives  Probably because it has been defined as an Abstract TxId the matching Tuple is incorporated in the same field (when exported from BRO Prep Elements Look Up). |
| TxId repeats – priority system | 9 | There are many TxIds that appear more than once.  The XBRL UK Preparers and Developers Guide sect 4.10 makes it clear that all occurrences of a Taxonomy value must be shown.  OS Have put the TxId value (4275) against the STGRL. Logic being that STRGL is higher up the reporting chain, HyId 14 includes HyId 1, and this line is more of a data entry line.  Points to keep in mind:   1. BRO is designed to be a sort of “super CoA”, so makes sense to follow basic rules of a CoA. E.g do not have two or more “Accounts” which store the same information. 2. But whether in BROs or RG we need to allow for multiple occurrences. They could be allowed within BRO by ensuring that only one occurrence could not be an RO. (Many of the repeated TxIds would never be other than RO) 3. Where a TxId can appear in different places in a set of accounts can vary. E,g, 3990 ProfitLossForPeriod could show in the *P&L Account*, and the *Balance Sheet*, *BS Notes-Reserves* and *Reconciliation of movements in shareholders' funds*. Looking at the HMRC AAAA example accounts, it is quite legitimate for the same Taxonomy Lable to appear more than once with the same context. (The Developers Guide section 4.8 only talks about unique vales, not unique occurrences).   **Following discussion with DJH 18 May 2012**  Would relax rule about only having one occurrence of TxId i.e. could have multiple BROs utilising the same TxId. Rules:   1. Duplicated only allowed where they use the Equal To command in the Related field and 2. The RO field contains an RO 3. Cross reference would be initially by BRO name 4. Will consider allowing TxId directly in the Related field (much shorter than BRO name) 5. DJH had considered using the Id field, but agreed this is too probe to changing   CW further  Questions re StartEnd.  Presently eliminate the End copy.  If allowing multiple occurrences of a TxId would it make sense to leave in the End value but make it clear what it was e.g.  Assets.Fixed.IFA.CostOrValuation.Cost  becomes  Assets.Fixed.IFA.CostOrValuation.Cost:Start  Assets.Fixed.IFA.CostOrValuation.Cost:End  Think that this is already handled within the BRO logic.  (CW26/5/2012 – this whole area is under review, with allowed duplicates, Aliases, Master/Slaves etc. |

# Importing from 3rd Party Programs.

Depending up on the program, we should be able to transfer every scrap of information.

Certainly that is what we are aiming to do with regard to SAPA.

These notes are angled very much from this perspective.

## Nominal Ledger Information

We know that the CoA of SAPA is a very mongrel animal.

Although rarely identified as such, it contains:

1. Straight posting DE accounts.
2. “Control” accounts – equivalent to a Set in BROs
3. “Contra” accounts – that is accounts which belong to with each other and net off even though the accounts may be made of a mixture of expected Debit and Credit values, and Duration and Instant types. The main example is Fixed asset accounts in the Balance sheet; Cost and Valuation accounts (Debit Instant), Accumulated Depreciation (Credit Instant) and Depreciation Charge in Period (Credit Duration)
4. Summary or Subtotal types of accounts. Examples of these would be Direct Costs and Indirect Costs. There is no way one would pot detail information to such accounts. (In fact it is hard to see how such accounts can be used at all within SAPA, but that is another matter.
5. Memo accounts. That is accounts which strictly speaking are not part of the CoA, and in particular do not form part of the DE posting system. But there will be associations between these and proper DE accounts. This might be One to Many, Many to One or Many to Many.
6. System Defined and User Defined. Some fields are very precise and detailed in their content, and can be linked straight through to an exact TxId/DimId match. Others may only be able to be matched with certainty to a TxId, but only a higher level of DimId or even no DimId at all. Very few, if any, SAPA fields are so “free form” that they could take on any value.
7. Sub codes – for the most part sub codes in SAPA have a precise use. Where they do not, then the same observations apply as made about User Defined fields above.