

## Alphabet Rosetta accounting XBRL/GL Taxonomy

Alphabet Rosetta accounting XBRL/GL Taxonomy .....	1
1. Open document format and methods .....	1
2. Scope .....	1
3. Brief.....	2
4. History XBRL.....	3
5. XBRL Taxonomies is the foundation in interpreting.....	3
6. Methods are banned in standard organisations.....	3
7. History XBRL/GL.....	4
8. Accounting interchange methods – What is that? .....	5
9. Alphabet Rosetta XBRL/GL Taxonomy – specification .....	7

### 1. Open document format and methods

The Alphabet Rosetta accounting format and method as well as the Alphabet Rosetta accounting XBRL/GL Taxonomy are free and open for anyone to use:

- <http://www.alphabet.se/Rosetta/Rosetta.xsd>
- [http://www.alphabet.se/Rosetta/Rosetta\\_Specification.pdf](http://www.alphabet.se/Rosetta/Rosetta_Specification.pdf)
- [http://www.alphabet.se/XBRL/Rosetta\\_XBRL-GL\\_Taxonomy.pdf](http://www.alphabet.se/XBRL/Rosetta_XBRL-GL_Taxonomy.pdf)

.XGL is the work file extension for the Rosetta XBRL/GL taxonomy.

There are no Copyright or IPR demands on the file format, methods and use of these three documents and its content, from our side.

The only requirements are attribution, referring to Alphabet AB Rosetta when used.

### 2. Scope

This document describes the Alphabet Rosetta accounting XBRL/GL Taxonomy, the Alphabet Rosetta handling of XBRL/GL XML files.

It also highlights the technical grounds and interpretation issues of the XBRL/GL. Issues that are of importance to understand XBRL/GL.

It is in XBRL-terms a Taxonomy and is related to the “Alphabet Rosetta accounting file format and method” document, the Rosetta.xsd and the Alphabet Rosetta Accounting OS-independent C-Engine.

The Alphabet Rosetta accounting file format and method is used as the main basis of the Alphabet Rosetta Accounting OS-independent C-Engine basic module. The XBRL/GL-module handles importing and exporting data in XBRL/GL files to and from it, reading and writing.

The Alphabet Rosetta Accounting OS-independent C-Engine basic module, is the basis of Alphabet Rosetta Accounting programs / apps in any OS and is for possible OEM sales.

Samples in this document origins from the SIE official sample SIE4.se.

Special thanks to Gianluca Garbellotto XBRL/GL International to make us fully understand the XBRL/GL format.

### 3. Brief

The present update is referring to the XBRL Global Ledger Taxonomy Framework 2017- 2017- Public Working Draft 01 December 2016 <http://www.xbrl.org/int/gl/2016-12-01/gl-framework-2017-PWD-2016-12-01.html> (Note that all XML-tools do not work with XBRL right.)

The Alphabet Rosetta XBRL/GL interpret module is designed to be able to read and interpret as large parts as possible of the overall XBRL Global Ledger Taxonomy Framework.

The intention is to be able to read as many XBRL/GL files as possible and try to interpret them to something that is an accurate context.

The basic problem is that XBRL/GL is a file format but not a method of information handling using the file format (that SIE is). So the method of information handling is anything of the author of the imported files.

Its writing of XBRL/GL is designed to be as generic as possible with as much information content as possible.

Its written files do have its own style of XBRL/GL, trying to be understandable by others. The intension is to be able to write usable XBRL/GL files for other future purpose, users and other vendors programs/apps. This taxonomy in writing XBRL/GL files is based on the methods of the SIE4 specification (retrievable in English from <http://www.sie.se/>). This document is to understand the Rosetta taxonomy, the Rosetta XBRL/GL files.

This means that the Rosetta XBRL/GL module understands much more XBRL/GL tags (styles) than used writing.

The XBRL/GL should as all XBRL, be seen as a financial accounting report format rather than an information interchange file method between programs in the accounting chain of tasks (as SIE in fact is). This means for instance that we do not include unused accounts and **identifierReference** (that in SIE is referred to as accounting objects) that are included in the SIE and Rosetta module file writing.

There are no declaring sections in the Alphabet Rosetta accounting XBRL/GL Taxonomy, of the chart of accounts and objects as there is in SIE4 and the Alphabet Rosetta accounting file format. The reason is that it is financial reporting and not accounting data transfer. The Account and object data is embedded in the reports instead. Making the files longer but easier to read sections of it, using XML tools.

## 4. History XBRL

XBRL/GL is a file format by XBRL/GL International that derives from XBRL as an XBRL taxonomy (in the XBRL language/expressions) with the basic definitions made by XBRL International.

XBRL has its roots 1998 in The American Institute of Certified Public Accountants (AICPA) that was instrumental in pulling together what eventually became XBRL International. The intension has been followed up by US authorities and followed by authorities as well as Institute of Certified Public Accountants and Audits in a large number of countries worldwide, forming domestic XBRL organisations.

- <https://en.wikipedia.org/wiki/XBRL#History>
- <https://www.xbrl.org/>

Accountants and Audits and their organisations in a large number of countries worldwide firmly claims they want XBRL files.

## 5. XBRL Taxonomies is the foundation in interpreting

The experience from XBRL use is that it is heavily dependent on the local taxonomies based on a common XBRL set of XML tags to be used.

The XBRL standard XML tags are just fragment pieces and their relations must be defined, to be useable, be defined in a taxonomy. For instance a receiver in power of demanding reports (like an authority) defines an XBRL taxonomy for a certain report. Often the demanding party also specify a method in handling the file format.

But that taxonomy can't be used for general use. We should not underestimate the problems of agreeing in common file formats and information handling methods. This is how XBRL is always based on the freedom of writing your own taxonomy. But an own taxonomy must be agreed among its users to be technically possibly used.

## 6. Methods are banned in standard organisations

XBRL is a wannabe official international standard organisation in accounting and financial reporting,

By this XBRL is tightly connected to that methods in the use of the standard, are banned in international standard committee work of organisations related to the ISO and UN hierarchies of standard bodies. It is considered as proprietary solutions and out of scope.

Methods of using for instance standard file formats (agreements on how to use a file format into the last bolts and nuts), are essential for that information interchange actually do work.

### 6.1. Somebody else must define the methods

Somebody else, that is not a standard organisation like a user, vendor, user or vendor society, must define the "proprietary" methods.

That is why XBRL is so dependent on the issuer of the taxonomies, because the issuer is not the international standard organisation. Due to the international standard organisation syndrome, XBRL itself can't issue a method specification. But users, vendors or governmental societies (like the IRS or AICPA) can (not being a standard organisation, and must it be done outside the XBRL framework).

Typical suppliers of methods are public authorities or huge manufacturers defining methods of use, demanding all dependent parties to use a format and method. We see this in governmental reporting and

for instance in car manufacturing supply chains. Cases of local working implementations, not applicable outside its domains.

We have seen especially in the 1980ies large system vendors totally dominating the “standard” method market. As such also sidestepping completely the standard organisations by making own versions of the file formats, very similar to the official standards. Them making their very proprietary solutions dominating the market, no one else making method definitions of the public standards. The environment name the proprietary solutions everybody use, as “standards” despite the total proprietary nature of it. It is a vendor enclosure of an open market.

The SIE organisation is a vendor society agreeing that accounting data should be interchangeable with a format and method. This to ensure that pre-systems for payables, receivables, salaries, inventories and post systems like income tax declaration programs are compatible between the vendors of the vendor society. This is possible when one vendor is not that huge to consider keeping the rest of the market out of its customers, in moments when all wants generic interchange.

In fact moments when all vendors agree of common information interchange are rare, most of the time all do not want to agree. Typical is that the SIE haven’t made anything successful since 1994.

The interesting aspect of the SIE4 is the market force in free open solutions supporting a free and open market of applications. A market of application fragments that can and do interact with each other. The force of SIE is so strong that huge global well known system suppliers, must provide SIE-support in order to sell in Sweden, else customers will not buy the system. This making SIE applicable worldwide, because there are SIE-support available in all systems marketed in Sweden, even though the supplier do not market SIE-support outside Sweden. Most vendors wants, more or less, to enclose the market.

It means that SIE is globally the only vendor independent working, file format exchange of accounting data, between applications.

Alphabet AB do not try to dominate or control the market but show the way, how the proprietary solutions should/could be used. Alphabet AB is interested in expand the features in the global market of accounting software and make it suitable for mobile phone and tablet accounting. Using what is already successful and deploy it globally, to the demand. The basic idea is to develop the SIE4 success for global multiple language use and support the demand of new totally digitalised accountings, internet banking integration and a more useful legal fundamental (the EU VAT directive) view of invoices.

No one can say that XBRL haven’t tried. In 2007 in the board of XBRL Sweden, the Swedish audits organisation made huge pushes to have XBRL/GL being used and supported by the program vendors. SIE decided after debate later not to implement SIE on XBRL/GL.

The Rosetta XBRL/GL taxonomy is the SIE methods implemented on XBRL/GL on the Rosetta format, an enhanced SIE file format.

## **7. History XBRL/GL**

There is a huge need for information exchange of bookkeeping accounting data between accounting computer programs/apps, to income tax declaration programs, financial analyse and from pre-systems like salaries, inventories and payables/receivables.

This is origins of the Swedish user/vendor society SIE and its SIE file format including information interchange methods made 1991-1994.

The same basic idea led to the formation of the of the XBRL/GL Working Group with its XBRL/GL to be a general ledger file format for general purpose use, a general XBRL ledger taxonomy.

- <https://www.xbrl.org/tag/xbrl-gl/>
- <https://www.xbrl.org/the-consortium/get-involved/gl/>
- <https://www.xbrl.org/the-standard/what/global-ledger/>

This is a huge step into something general usable, but still lacks methods because XBRL is a wannabe official international standard organisation. XBRL can't work with the methods and someone else has to take that responsibility in order to get something routinely working.

In fact there are no generally known method specifications of XBRL/GL. It means that all XBRL/GL use is local adaptations.

This means that the Rosetta, in reading and interpreting, beside the Rosetta XBRL/GL taxonomy, try to use some common sense and good guessing in interpreting non Rosetta taxonomy XBRL/GL files.

Rosetta is using the SIE methods in the Rosetta XBRL/GL taxonomy and so in its written Rosetta XBRL/GL files.

## 8. Accounting interchange methods – What is that?

There are three general levels of Accounting interchange methods:

1. The GAAP used (specified in the files and can vary quite)
2. The tax legislation (the EU VAT directive)
3. The technical implementation issues of information interchange files

Without the methods wrong interpretation of the information will be the general problem. It shows in symptoms like sums are not correct or the accounting has a very unfamiliar look and presentation (makes it hard to understand).

### 8.1. Methods of the GAAP used

The GAAP (Generally Accepted Accounting Principles) are similar but different in every country, defined domestically by the Institute of Certified Public Accountants and Audits or/and governmental bodies (that often are the same).

Typical differences in GAAP are like in Sweden non-transparent pre-systems are the general rule and illegal in most other countries. It means the accounting only contains the periodic sums of the actions in salary systems, payables and receivables.

Other issues are like the Swedish GAAP (BFN) demands paper invoices to be stored despite they being scanned and interpreted by use of OCR. This because the government general persistent resisting the implementation of the EU digital signature directive (1999/93/EG), now the EU digital signature regulation (910/2014 to be implemented 2016). In Sweden there are a large number of invoice scanning service bureaus and provider of scanning systems for the paper invoice receiving entities. No digital signing, the scanning party can't verify its interpretations digitally, not fulfil the demands of the GAAP. Else the scanned invoices would be certified and no paper bales needed, and very soon all accounting would become completely digital.

The missing digital signature infrastructure makes the VAT directive article 233 impossible to implement (and the VAT directive article 233 is excluded in the Swedish VAT law). It is also one of the major factors that makes true digital receipts in the daily store trade impossible to

be implemented successfully, receipts can't be signed by the issuing company, the concept of digital signatures of legal persons. Huge sums are invested by large retail chains in delivering digital receipts to their registered customers, when the banks refuse and the government resists. But with no method of standard connection to accounting it will end up in very poor results. Only governments can initiate digital signature and digital receipt (huge future markets) infrastructures.

Digital signatures would be used by everyone many times every day in all part of trade. But the firm resistance from governments (the lack of drive by the present EU commission of its own good work) is the day.

The GAAP is declared in XBRL/GL and in the Rosetta format. Reading foreign accounting files knowledge in local GAAPs are essential.

## **8.2. Methods of the tax legislation (the EU VAT directive)**

The VAT legislation is the primary legislation of accounting in every country. Not harmonised to the VAT law, the tax reports are invalid and about every entity of accounting has to do tax reports.

The major topic is the definition of the invoice and the handling of invoices in accounting, as accounting and tax reporting tools and objects.

In Europe a common VAT directive that is the father of all its member states VAT legislation is a very detailed, commercially outstanding concise and specific common work. It means that an invoice (article 226) is (computer-) logically the same invoice in every member state.

We are using the SIE methods of the EU VAT directive.

## **8.3. Methods of the technical implementation issues**

The SIE specification is focused on the technical implementation of its file format (that actually could be any file format) and is much more important than the file format syntax itself.

This describes the internal technical logics of how the file format should be used. For instance SIE says that the balances should be the sum of the accounting vouchers entries with accounting vouchers header dates of the balance period and nothing else. That the Opening balance should be defined by using the SIE Opening balance tags.

There are quite a lot of variants in internal logics in different accounting software. So an accounting program that uses an accounting entries to define the Opening balances have to remove it and put the data into SIE Opening balance tags. Else receiving programs will not understand the content of the file.

We are using the SIE methods in the Rosetta XBRL/GL taxonomy.

## 9. Alphabet Rosetta XBRL/GL Taxonomy – specification

The XBRL/GL allows more than one entity in one file.

The Alphabet Rosetta XBRL/GL Taxonomy and the Alphabet Rosetta (and SIE4) accounting file format can only handle one entity in one file, can't handle files with multiple entities.

However Alphabet Rosetta XBRL/GL Taxonomy can handle data from multiple accounting years of the same entity.

Our documentation starts with the Rosetta XBRL/GL taxonomy writing, and at the end there is a documentation in extended reading capabilities of any XBRL/GL file.

### 9.1. XBRL/GL XML structure - `cor:accountingEntries`

When Rosetta is using a linear XML format of purpose-components.

XBRL/GL is using beside its initial declarations, for everything, one reusable non-purpose general XML structure:

- `gl-cor:accountingEntries`
  - `gl-cor:documentInfo`
  - `gl-cor:entityInformation`
  - `gl-cor:entryHeader`
    - `gl-cor:entryDetail`
    - `gl-cor:xbrlInfo`

It being repeated for different use using different flags to declare for what use, like:

- `gl-cor:entriesType`
- `gl-cor:qualifierEntry`
- `gl-cor:xbrlInclude`
- `gl-bus:reportingCalendarPeriodType`

This makes the XBRL/GL definitions (XSD) shorter but nested. It is harder to read without good XBRL-tools or applications for XBRL/GL.

Another fundamental difference is that Rosetta see one journal per Ledger (accounting year) while XBRL/GL se one journal per series of an accounting year. It is quite a mental difference.

The documents are defined by:

- **Balance reports**
  - `gl-cor:entriesType` are tagged `gl-cor:entriesType` with `>balance<`
    - With the general balances of the Ledger tagged `gl-cor:qualifierEntry` with `>balance-brought-forward<`
    - With the period balances of the Ledger tagged `gl-cor:qualifierEntry` with `>standard<`
- **Balance reports of an object**
  - `gl-cor:entriesType` are tagged `gl-cor:entriesType` with `>mapping<`
    - With the general balances of the Ledger tagged `gl-cor:qualifierEntry` with `>balance-brought-forward<`
    - With the period balances of the Ledger tagged `gl-cor:qualifierEntry` with `>standard<`
- **Vouchers** (the journals, that are separated, one journal for each Voucher ID series)
  - `gl-cor:entriesType` are tagged `gl-cor:entriesType` with `>journal<`

## 9.2. The declarations - xbrli

The XBRL/GL file always includes a file header of declarations.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Created by Alphabet Rosetta SIE-XBRL/GL Viewer version 1.000, 2018-12-03 -->
<!-- Alphabet XBRL GL interpretation of SIE type Ledger -->
<!-- See: www.alphabet.se/XBRL -->
<!-- This XBRL GL file is a converted Swedish SIE file -->
<!-- It means its taxonomy is based on the SIE-logics -->
<!-- About SIE see http://www.sie.se/english/english.asp -->
<xbrli:xbrl xmlns:xbrli="http://www.xbrl.org/2003/instance"
  xmlns:xbrll="http://www.xbrl.org/2003/linkbase"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:gl-cor="http://www.xbrl.org/int/gl/cor/2016-12-01"
  xmlns:gl-muc="http://www.xbrl.org/int/gl/muc/2016-12-01"
  xmlns:gl-bus="http://www.xbrl.org/int/gl/bus/2016-12-01"
  xmlns:gl-plt="http://www.xbrl.org/int/gl/plt/2016-12-01"
  xmlns:iso4217="http://www.xbrl.org/2003/iso4217"
  xmlns:iso639="http://www.xbrl.org/2005/iso639"
  xsl:schemaLocation="http://www.xbrl.org/int/gl/plt/2016-12-01
    http://www.xbrl.org/taxonomy/int/gl/2016-12-01/plt/case-c-b-m-u/gl-plt-all-2016-12-01.xsd">
  <xbrll:schemaRef xlink:type="simple" xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase"
    xlink:href="http://www.xbrl.org/taxonomy/int/gl/2016-12-01/plt/case-c-b-m-u/gl-plt-all-2016-12-01.xsd" />
  <xbrli:unit id="NotUsed">
    <xbrli:measure>pure</xbrli:measure>
  </xbrli:unit>
  <xbrli:unit id="sek">
    <xbrli:measure>iso4217:SEK</xbrli:measure>
  </xbrli:unit>
  <xbrli:context id="id-123456-7890">
    <xbrli:entity>
      <xbrli:identifier scheme="http://www.alphabet.se/XBRL">Testbolaget</xbrli:identifier>
    </xbrli:entity>
    <xbrli:period>
      <xbrli:startDate>2009-01-01</xbrli:startDate>
      <xbrli:endDate>2010-12-31</xbrli:endDate>
    </xbrli:period>
  </xbrli:context>
```

Alphabet Rosetta XBRL/GL Taxonomy do not read the XML schema declarations etc with exceptions of

The `xbrli:unit` refers to the default currency of the accounting. An accounting must have a default currency (or none but is).

As XBRL/GL is a reporting format the currency is noted with every amount data in the reports and journals. An accounting can only have one common currency. All other currency data is quantity information.

All used currencies must be declared under the `xbrli:unit` tag.

```
<xbrli:unit id="sek">
  <xbrli:measure>iso4217:SEK</xbrli:measure>
</xbrli:unit>
```

In XBRL/GL the `xbrli:context id` must be defined and used regarding all data referring to the entity with the same `contextRef` tag. The `xbrli:identifier scheme` only refers to this document.

We use the equivalent to Rosetta as `contextRef` tag:

- Accounting > Entity > LegalRegId

```
<xbrli:context id="id-123456-7890">
  <xbrli:entity>
    <xbrli:identifier scheme="http://www.alphabet.se/XBRL">Testbolaget</xbrli:identifier>
  </xbrli:entity>
```

The `xbrli:startDate` and `xbrli:endDate` refers to the first and the last accounting entry days the document refers to.

```
<xbrli:period>
  <xbrli:startDate>2009-01-01</xbrli:startDate>
  <xbrli:endDate>2010-12-31</xbrli:endDate>
</xbrli:period>
</xbrli:context>
```

### 9.3. The `gl-cor:entityInformation`

The entity information section is repeated for every `gl-cor:accountingEntries`, report or journal series.

### 9.4. Common data

The XBRL/GL `gl-cor:documentInfo` declares the report content.

```
<gl-cor:documentInfo>
  <gl-cor:entriesType contextRef="id-551264-1241">balance</gl-cor:entriesType>
  <gl-cor:uniqueID contextRef="id-551264-1241">0</gl-cor:uniqueID>
  <gl-cor:language contextRef="id-551264-1241">iso639:sv</gl-cor:language>
  <gl-cor:creationDate contextRef="id-551264-1241">2018-12-03</gl-cor:creationDate>
  <gl-bus:creator contextRef="id-551264-1241">Jan</gl-bus:creator>
  <gl-cor:entriesComment contextRef="id-551264-1241">Balanser</gl-cor:entriesComment>
  <gl-cor:periodCoveredStart contextRef="id-551264-1241">1995-01-01</gl-cor:periodCoveredStart>
  <gl-cor:periodCoveredEnd contextRef="id-551264-1241">1995-12-31</gl-cor:periodCoveredEnd>
  <gl-bus:sourceApplication contextRef="id-551264-1241">Alphabet Rosetta</gl-bus:sourceApplication>
  <gl-muc:defaultCurrency contextRef="id-551264-1241">iso4217:sek</gl-muc:defaultCurrency>
</gl-cor:documentInfo>
```

#### 9.4.1. The `gl-cor:language` tag

XBRL/GL only support one text line for each information text tag.

Rosetta supports multiple languages and use the information text of the setting of the application. The `gl-cor:language` tag is used to declare what language setting the report is generated under (what Rosetta texts are used).

Non-technical pure information texts are:

- `gl-cor:entriesComment`
- `gl-bus:entryOrigin`
- `gl-cor:entryComment`
- `gl-cor:accountMainDescription`
- `gl-cor:mainAccountTypeDescription`
- `gl-cor:identifierDescription`
- `gl-bus:identifierPurpose`
- `gl-cor:detailComment`
- `gl-cor:postingStatusDescription`

#### 9.4.2. The `gl-cor:creationDate` and `gl-bus:creator` tags

These tags documents the creator of the file and the creation date.

Equivalent to Rosetta:

- Accounting > GeneratingUser

Equivalent to parameters in SIE:

- #GEN

#### 9.4.3. The `gl-cor:periodCoveredStart`, `gl-cor:periodCoveredEnd`

These tags documents the report frame in time.

We use the Equivalent data in Rosetta:

- Accounting > LedgersChart > Ledger > YearStart / YearEnd

We use the Equivalent data in SIE:

- #RAR

#### 9.4.4. The `gl-bus:sourceApplication` tags

These tags documents the creating program of the file.

Equivalent to Rosetta:

- `Accounting > Program`

Equivalent to parameters in SIE:

- `#PROGRAM`

#### 9.5. Balance reports

The XBRL/GL `gl-cor:documentInfo` declares the report content.

A balance basic report looks like this and the `gl-cor:entriesType >balance<` tells it is a balance report.

```
<gl-cor:documentInfo>
  <gl-cor:entriesType contextRef="id-551264-1241">balance</gl-cor:entriesType>
  <gl-cor:uniqueID contextRef="id-551264-1241">0</gl-cor:uniqueID>
  <gl-cor:language contextRef="id-551264-1241">iso639:sv</gl-cor:language>
  <gl-muc:defaultCurrency contextRef="id-551264-1241">iso4217:sek</gl-muc:defaultCurrency>
</gl-cor:documentInfo>
```

The balance report has a and to describe the balance information.

```
<gl-cor:entryHeader>
  <gl-cor:postedDate contextRef="id-551264-1241">1995-03-01</gl-cor:postedDate>
  <gl-cor:entryType contextRef="id-551264-1241">standard</gl-cor:entryType>
  <gl-bus:entryOrigin contextRef="id-551264-1241">Årligen</gl-bus:entryOrigin>
  <gl-cor:entryComment contextRef="id-551264-1241">Balanser</gl-cor:entryComment>
  <gl-cor:qualifierEntry contextRef="id-551264-1241">balance-brought-forward</gl-cor:qualifierEntry>
  <gl-cor:entryDetail>
    <gl-cor:account>
      <gl-cor:accountMainID contextRef="id-551264-1241">1010</gl-cor:accountMainID>
      <gl-cor:accountMainDescription contextRef="id-551264-1241">Kassa</gl-cor:accountMainDescription>
      <gl-cor:mainAccountType contextRef="id-551264-1241">asset</gl-cor:mainAccountType>
      <gl-cor:mainAccountTypeDescription contextRef="id-551264-1241">Tillgångar</gl-cor:mainAccountTypeDescription>
      <gl-cor:accountPurposeDescription contextRef="id-551264-1241">BAS96</gl-cor:accountPurposeDescription>
    </gl-cor:account>
    <gl-cor:amount unitRef="sek" decimals="2" contextRef="id-551264-1241">307500.00</gl-cor:amount>
    <gl-cor:postingDate contextRef="id-551264-1241">1995-03-01</gl-cor:postingDate>
  </gl-cor:entryDetail>
</gl-cor:entryHeader>
```

##### 9.5.1. The `gl-cor:uniqueID` tag

In SIE and Rosetta the Ledgers (accounting years) are numbered from the present 0 and backwards into the past, normally with the present first.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Id`

Equivalent to parameters in SIE:

- `#RAR`
- `#IB etc`

##### 9.5.2. The `gl-cor:postedDate` tag

In SIE and Rosetta the Ledgers (accounting years) are numbered from the present 0 and backwards into the past, normally, the present first.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > YearEnd`

Equivalent to parameters in SIE:

- `#RAR`

### 9.5.3. The `gl-cor:postingDate` tag

The tag is specify the period end date, that in normal balance reports are the same as the `gl-cor:postedDate` tag.

### 9.5.4. The `gl-cor:accountPurposeDescription` tag

This tag is used for the GAAP (and chart of accounts) being used.

Equivalent to Rosetta:

- `Accounting > Gaap`

Equivalent to parameters in SIE:

- `#KPTYP`

### 9.5.5. The `gl-cor:xbrlInclude` tag

This tag is used for the GAAP (and chart of accounts) being used.

- `Opening balance >beginning_balance<`
- `Forward balance >ending_balance<`
- `Periodic balance >standard<`

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Report > BalanceAccount > Opening`
- `Accounting > LedgersChart > Ledger > Report > BalanceAccount > Forward`
- `Accounting > LedgersChart > Ledger > Report > BalanceAccount > Month`

Equivalent to parameters in SIE:

- `#IB`
- `#UB`
- `#PBUDGET`
- `#PSALDO`

### 9.5.6. Period reports

The XBRL/GL don't have any tools to define the length of periodic data so the Rosetta taxonomy only work with monthly periodic reports. The Rosetta works also with quarterly, tertial, semi-annual and annual reports and the Rosetta reading facilities generates them from months.

#### 9.5.6.1. The `gl-cor:qualifierEntry` tag

In SIE and Rosetta the Ledgers (accounting years) are numbered from the present 0 and backwards into the past, normally, the present first.

Equivalent to parameters in SIE:

- `Balances >balance-brought-forward<`
- `Periodic >standard<`

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > YearEnd`

Equivalent to parameters in SIE:

- `#RAR`

### 9.5.6.2. The `gl-cor:entryType` tag – for Budget data

The tag is always put as `>standard<` in balance reports except for periodic budget information and then set as `>budget<`.

Equivalent to Rosetta:

- Accounting > LedgersChart > Ledger > Report > BalanceAccount > Month > Budget

Equivalent to parameters in SIE:

- #PBUDGET

### 9.5.6.3. The `gl-cor:postingDate` tag

The tag is specify the period end date of the monthly period.

Equivalent to Rosetta:

- Accounting > LedgersChart > Ledger > Report > BalanceAccount > Month > End

Equivalent to parameters in SIE:

- #PSALDO
- #PBUDGET

## 9.5.7. Object reports

The XBRL/GL `gl-cor:documentInfo` declares the report content.

A object balance report looks like a balance report and the `gl-cor:entriesType >mapping<` tells it is an object report.

### 9.5.7.1. The `gl-cor:identifierReference` tag

The tag documents what object is being mapped.

```
<gl-cor:identifierReference>
  <gl-cor:identifierCode contextRef="id-551264-1241">1</gl-cor:identifierCode>
  <gl-cor:identifierDescription contextRef="id-551264-1241">Ekonomi</gl-cor:identifierDescription>
  <gl-cor:identifierType contextRef="id-551264-1241">other</gl-cor:identifierType>
  <gl-cor:identifierCategory contextRef="id-551264-1241">1</gl-cor:identifierCategory>
  <gl-bus:identifierPurpose contextRef="id-551264-1241">Kostnadsställe</gl-bus:identifierPurpose>
</gl-cor:identifierReference>
```

### 9.5.7.2. The `gl-cor:identifierCode` tag

The tag is specify the Id of the Object being mapped .

Equivalent to Rosetta:

- Accounting > LedgersChart > Ledger > Report > Object > Id

Equivalent to parameters in SIE:

- #OIB
- #OUB
- #PSALDO
- #PBUDGET

### 9.5.7.3. The `gl-cor:identifierCategory` tag

The tag is specify the Id of the Dimension being mapped.

The `gl-cor:identifierExternalReference` tag defines if there are a super Dimension related to.

Equivalent to Rosetta:

- Accounting > LedgersChart > Ledger > Report > Object > Dimension > Id

Equivalent to parameters in SIE:

- #OIB
- #OUB
- #PSALDO
- #PBUDGET

### 9.5.7.4. The `gl-cor:identifierType` tag

The tag is specify if any XBRL/GL predefined identifiers being used.

- > customer<
- > employee<
- > vendor<
- > other<

## 9.6. Journals

The XBRL/GL `gl-cor:documentInfo` declares the report content.

The `gl-cor:entriesType >journal<` tells it is a voucher list.

```
<gl-cor:documentInfo>
<gl-cor:entriesType contextRef="id-551264-1241">journal</gl-cor:entriesType>
<gl-cor:language contextRef="id-551264-1241">iso639:sv</gl-cor:language>
<gl-cor:creationDate contextRef="id-551264-1241">2018-12-03</gl-cor:creationDate>
<gl-bus:creator contextRef="id-551264-1241">Jan</gl-bus:creator>
<gl-cor:entriesComment contextRef="id-551264-1241">Verifikationer</gl-cor:entriesComment>
<gl-cor:periodCoveredStart contextRef="id-551264-1241">1995-01-01</gl-cor:periodCoveredStart>
<gl-cor:periodCoveredEnd contextRef="id-551264-1241">1995-03-01</gl-cor:periodCoveredEnd>
<gl-bus:sourceApplication contextRef="id-551264-1241">Alphabet Rosetta</gl-bus:sourceApplication>
<gl-muc:defaultCurrency contextRef="id-551264-1241">iso4217:sek</gl-muc:defaultCurrency>
</gl-cor:documentInfo>
...
<gl-cor:entryHeader>
<gl-cor:postedDate contextRef="id-551264-1241">1995-03-01</gl-cor:postedDate>
<gl-cor:enteredBy contextRef="id-551264-1241">unknown</gl-cor:enteredBy>
<gl-cor:enteredDate contextRef="id-551264-1241">1995-06-08</gl-cor:enteredDate>
<gl-cor:entryNumber contextRef="id-551264-1241">9</gl-cor:entryNumber>
<gl-cor:entryComment contextRef="id-551264-1241">Betaling leverantörsskuld</gl-cor:entryComment>
<gl-cor:entryDetail>
<gl-cor:account>
<gl-cor:accountMainID contextRef="id-551264-1241">2110</gl-cor:accountMainID>
<gl-cor:accountMainDescription contextRef="id-551264-1241">Leverantörsskulder</gl-cor:accountMainDescription>
<gl-cor:mainAccountType contextRef="id-551264-1241">liability</gl-cor:mainAccountType>
<gl-cor:accountPurposeDescription contextRef="id-551264-1241">BAS96</gl-cor:accountPurposeDescription>
</gl-cor:account>
<gl-cor:amount unitRef="sek" decimals="2" contextRef="id-551264-1241">10000.00</gl-cor:amount>
<gl-cor:identifierReference>
<gl-cor:identifierCode contextRef="id-551264-1241">A1</gl-cor:identifierCode>
<gl-cor:identifierDescription contextRef="id-551264-1241">Byggprojektet</gl-cor:identifierDescription>
<gl-cor:identifierType contextRef="id-551264-1241">other</gl-cor:identifierType>
<gl-cor:identifierCategory contextRef="id-551264-1241">6</gl-cor:identifierCategory>
<gl-bus:identifierPurpose contextRef="id-551264-1241">Projekt</gl-bus:identifierPurpose>
</gl-cor:identifierReference>
<gl-cor:postingStatus contextRef="id-551264-1241">posted</gl-cor:postingStatus>
</gl-cor:entryDetail>
</gl-cor:entryDetail>
```

### 9.6.1. The `gl-cor:uniqueID` tag

Each Voucher Id series forms an independent journal with the `gl-cor:uniqueID` as Series.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Id > series`

Equivalent to parameters in SIE:

- `#VER regdate series`

### 9.6.2. The `gl-cor:enteredDate` and `gl-cor:enteredBy` tags

These tags documents the user of registration of the accounting Voucher entry and the creation date.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Accountant`

Equivalent to parameters in SIE:

- `#VER regdate sign`

### 9.6.3. The `gl-cor:postedDate` tag

Voucher date /that is the report trigger and matches the balance periods).

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Date`

Equivalent to parameters in SIE:

- `#VER verdate`

### 9.6.4. The `gl-cor:entryNumber` tag

Voucher date.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Id`

Equivalent to parameters in SIE:

- `#VER verno`

### 9.6.5. The `gl-cor:entryComment` tag

Voucher text.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Title`

Equivalent to parameters in SIE:

- `#VER vertex`

### 9.6.6. Voucher Entry data, the `gl-cor:entryDetail` tag

#### 9.6.6.1. The `gl-cor:amount` tag

An accounting can only have one currency for its entire content. All other currency data is quantity information and in general not accounted.

XBRL/GL is a report specification and demands the number of decimals and the currency to be noted.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Entry > Amount`

Equivalent to parameters in SIE:

- `#TRANS amount`

#### 9.6.6.2. The `gl-cor:postingDate` tag

Voucher entry date (only as a note, has no technical purpose).

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Entry > AuxDate`

Equivalent to parameters in SIE:

- `#TRANS transdate`

#### 9.6.6.3. The `gl-cor:documentReference` tag

Voucher entry external file reference.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Entry > Invoice > OriginalInvoiceFileUrl`

#### 9.6.6.4. The `gl-cor:postingStatus` tag

Voucher entry Active or Deprecated entry status.

The tag is specify if any XBRL/GL predefined identifiers being used.

- `>posted< // Active`
- `>cancelled< // Deprecated`

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Entry > Deprecated`

Equivalent to parameters in SIE:

- `#BTRANS`

#### 9.6.6.5. The `gl-cor:detailComment` tag

Voucher entry text.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Entry > Title`

Equivalent to parameters in SIE:

- `#TRANS transtext`



### 9.6.7. Quantity data, the `gl-bus:measurable` tag

#### 9.6.7.1. The `gl-bus:measurableQuantity` tag

Account ID.

In SIE it must be a numeric value, but in Rosetta and XBRL/GL it must be of an alpha-numeric value.

Equivalent to Rosetta:

- `Accounting > LedgersChart > Ledger > Voucher > Entry > Quantity`

Equivalent to parameters in SIE:

- `#TRANS` quantity

#### 9.6.7.2. The `gl-bus:measurableUnitOfMeasure` tag

Account Name.

Equivalent to Rosetta:

- `Accounting > AccountsChart > Account > QuantityUnit`

Equivalent to parameters in SIE:

- `#ENHET`

### 9.6.8. Account data, the `gl-cor:account` tag

The Account data is used in `gl-cor:entryDetail` in all reports/lists and do not support the Account coding features of Rosetta, but most other data.

#### 9.6.8.1. The `gl-cor:accountMainID` tag

Account ID.

In SIE it must be a numeric value, but in Rosetta and XBRL/GL it must be of an alpha-numeric value.

Equivalent to Rosetta:

- `Accounting > AccountsChart > Account > Id`

Equivalent to parameters in SIE:

- `#KONTO`

#### 9.6.8.2. The `gl-cor:accountMainDescription` tag

Account Name.

Equivalent to Rosetta:

- `Accounting > AccountsChart > Account > Name`

Equivalent to parameters in SIE:

- `#KONTO`

### 9.6.8.3. The `gl-cor:accountPurposeDescription` tag

This tag is used for the GAAP (and chart of accounts) being used.

Equivalent to Rosetta:

- Accounting > Gaap

Equivalent to parameters in SIE:

- #KPTYP

### 9.6.8.4. The `gl-cor:mainAccountType` Account types

This tag group the accounts in different subgroups of Active, Passive, Revenues and Costs..

Equivalent to Rosetta:

- Accounting > AccountsChart > Account > Type

Equivalent to parameters in SIE:

- #KTYP

Rosetta Account types are written as XBRL/GL `gl-cor:mainAccountType` Account types:

- Tangible
  - asset
- Stock
  - asset
- Receivables
  - asset
- Assets
  - asset
- Equities
  - equity
- Liabilities
  - liability
- Revenues
  - income
- Supplies
  - expense
- Expenses
  - expense
- Staff
  - expense
- LossGains
  - loss
- Financial
  - comprehensive-income
- Auxiliary
  - Other

### 9.6.8.5. Account types balances, noted but not used

Account type balances are written (with no `gl-cor:accountMainID` noted but a `gl-cor:mainAccountType`) only for the reader. Also the Account type group balances are written (with no `gl-cor:accountMainID` or `gl-cor:mainAccountType`) noted, only for the reader but only with a `gl-cor:mainAccountTypeDescription`.

These are not used when reading a XBRL/GL file by the Rosetta XBRL/GL Taxonomy engine module. It processes the sums of the Account balances and the Account type balances.

## 9.6.9. Entity data, the `gl-cor:entityInformation` tag

### 9.6.9.1. The `gl-bus:organizationIdentifiers` tag

The XBRL/GL handles organisational identifiers of different kind in by declaring what:

- `Legal` // Legal Reg number
- `Vat`// Vat Reg number
- `Statistical` // Statistical class
- `Jurisdiction` // What law is valid, country code

### 9.6.9.2. The `gl-bus:organizationAddress` tag

The Rosetta Address field is multiple in order to handle the collapse of the ISO and the Universal Postal Union UPU Standard S42: International postal address components and templates.

XBRL/GL don't handle the situation and implemented US addressing.

```
<element ref="gl-bus:organizationAddressName" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressDescription" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressPurpose" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressLocationIdentifier" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationBuildingNumber" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressStreet" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressStreet2" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressCity" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressStateOrProvince" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressZipOrPostalCode" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressCountry" minOccurs="0" maxOccurs="1"/>
<element ref="gl-bus:organizationAddressActive" minOccurs="0" maxOccurs="1"/>
```

So we support four address levels by using them in the order:

1. `gl-bus:organizationBuildingNumber`
2. `gl-bus:organizationAddressStreet`
3. `gl-bus:organizationAddressStreet2`
4. `gl-bus:organizationAddressStateOrProvince`

Equivalent to Rosetta:

- `Accounting > Entity > Address`

Equivalent to parameters in SIE:

- `#ADDRESS`

## 9.7. Interpret reading requirements

The XBRL/GL `gl-cor:documentInfo` declares the report content.

### 9.7.1. The issue of identify a Ledger from an XBRL/GL file

If the `gl-cor:uniqueID` tag is not present in the XBRL/GL file in balances, the Rosetta XBRL/GL Taxonomy engine module tries to get a Ledger Id (an accounting year) from dates, or will fail.

- `gl-bus:fiscalYearStart`
- `gl-bus:fiscalYearEnd`

This means that `gl-bus:fiscalYearStart` and `gl-bus:fiscalYearEnd` are requirements for balance and object reports.

For voucher lists (without balances) the `gl-bus:fiscalYearStart` are the requirements to be able to read the list.

### 9.7.2. The minimum data requirements

#### 9.7.2.1. Balance reports

The `gl-cor:entriesType`, `gl-cor:xbrlInclude`, `gl-cor:qualifierEntry`, `gl-bus:fiscalYearStart`, `gl-bus:fiscalYearEnd` and account and amount must be set.

#### 9.7.2.2. The `gl-cor:entriesType` tag

The following is supported by the `gl-cor:entriesType` tag:

- account
  - Tries to read it as an accounts chart, supported
- balance
  - Tries to read it as balance report
- entries
  - Tries to read it as a list of vouchers
- journal
  - Tries to read it as a list of vouchers
- ledger
  - Not supported
- assets
  - Not supported
- trialbalance
  - Tries to read it as balance report
- taxtables
  - Not supported
- mapping
  - Tries to read it as object balance report
- versioning
  - Not supported
- master\_file
  - Not supported
- trade\_documents
  - Not supported
- profile\_compliant
  - Not supported
- other
  - Not supported

### 9.7.2.3. The `gl-cor:qualifierEntry` tag

The `balance-brought-forward` is interpreted as a balance report in combination with `gl-cor:xbrlInclude` set to `beginning_balance` or `ending_balance` are accepted.

The `standard` is interpreted as a month periodic balance / budget report in combination with `gl-cor:xbrlInclude` set to `period_change is` accepted.

The `other` is not accepted.

The combination with `gl-cor:entryType` set to `budget` is interpreted as a month periodic budget report, and `gl-cor:entryType` set to `standard` is interpreted as a month periodic balance report

The following `gl-cor:entryType` are not accepted (report not read):

- adjusting
- comparative
- external-accountant
- passed-adjusting
- eliminating
- proposed
- recurring
- reclassifying
- simulated
- tax
- other

### 9.7.2.4. Account types

All accounts are Assets by default and interpreted from XBRL/GL `gl-cor:mainAccountType` to Rosetta Account types:

- asset
  - Assets
- equity
  - Equities
- contr-to-equity
  - Equities
- distr-from-equity
  - Equities
- liability
  - Liabilities
- income
  - Revenues
- expense
  - Expenses
- gain
  - LossGains
- loss
  - LossGains
- comprehensive-income
  - Financial
- other
  - Auxiliary

#### **9.7.2.5. Voucher lists**

The `gl-cor:entriesType`, `gl-bus:fiscalYearStart` and voucher date and voucher entry account and amount must be set.

The `gl-cor:postingStatus` set to "cancelled" is supported as a deprecated entry. All others are not read and entry is supposed to be posted OK.

### **9.7.3. The details supplementary supported reading**

#### **9.7.3.1. The `gl-cor:signOfAmount` tag**

If it is "-" or "minus" the amount will be made negative.

#### **9.7.3.2. The `gl-cor:debitCreditCode` tag**

If it is "D" or "d" the amount will be made negative.