

# What is XBRL?

Jul 1, 2021

*Lately I get a lot of feedback and requests regarding my open-source library `py-xbrl`, which makes me very happy. But since I unfortunately can't answer every request and many questions focus on the XBRL standard, I want to give a short technical overview of the XBRL standard with this blog post.*

## 1 Introduction

XBRL (**eX**tensible **B**usiness **R**eporting **L**anguage) is a established XML-based specification used for the **transmission of financial data between two parties**. Transferring financial data between two different parties seems trivial at first glance. However, the huge number of different companies from different industries with different internal accounting systems makes the creation of a common interface a challenge. This is where XBRL comes into play. XBRL allows a wide variety of financial information to be compiled and transmitted.

A very common use case is the transmission of quarterly reports and annual reports to authorities. For example, every U.S. company (over a certain size) must file quarterly reports (10-Q) and annual reports (10-K) with the SEC using XBRL. These submissions are publicly accessible through [SEC EDGAR](#).

## 2 The Taxonomy

Since there is a large number of financial information with different accounting systems (us-gaap, ifrs...), a common language framework must be established prior to the data transfer. In the XBRL context, this common language framework is called a **Taxonomy**. The taxonomy defines a list of **concepts** that can then be used in the instance document to tag certain numbers and ensures the integrity and syntactic correctness of the data.

*It has proven useful to create a separate taxonomy for each accounting system - e.g. IFRS, German GAAP (HGB), US GAAP.*

### 2.1 The Taxonomy Schema

The taxonomy schema is the heart of the taxonomy and defines the concepts of the taxonomy. The concepts will later be used by the creator of a financial report to tag certain

numbers.

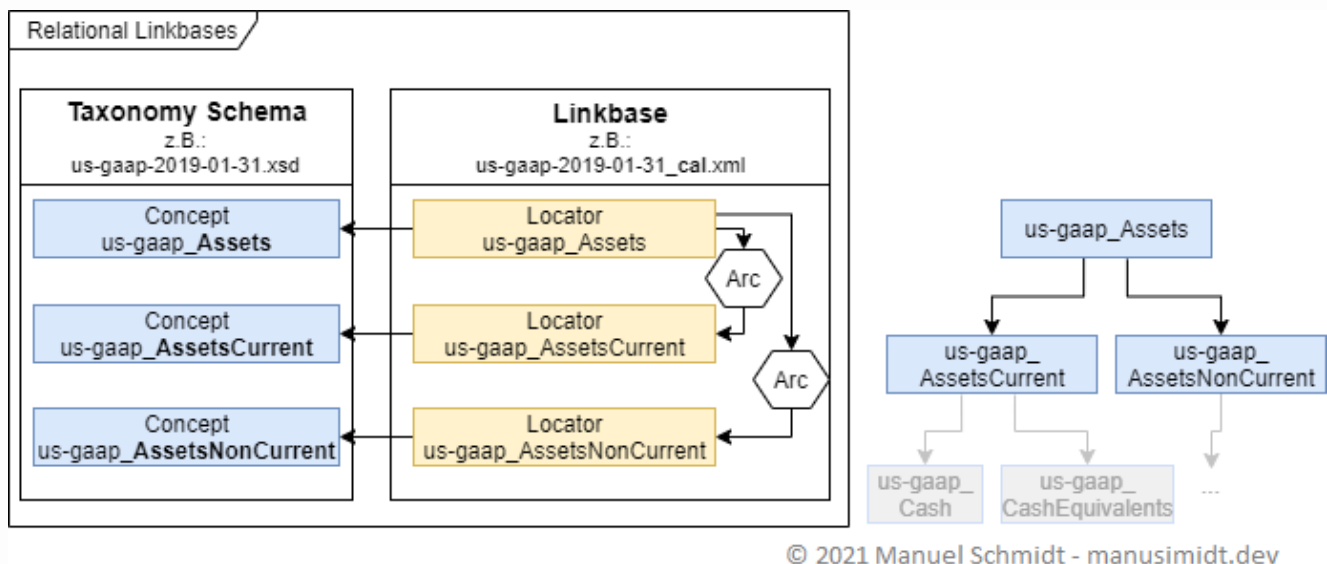
*An example: The US-GAAP taxonomy defines the concept "Cash and Cash Equivalents". Every company that creates a financial report based on us-gaap can use this concept ( `us-gaap:CashAndCashEquivalents` ) and associate it with a number in their financial report.*

## 2.2 Taxonomy Linkbases

Linkbases are individual XML files that bring structure to concepts and link them to additional information. This information can be, for example, user-friendly labels or references to authoritative literature. The linkbases are imported in the taxonomy schema. Linkbases can be divided into two main groups: **Relation Linkbases** and **Reference Linkbases**. Relation Linkbases create hierarchical relationships between multiple concepts. The interpretation of these hierarchical relationships is defined by the type of linkbase. Reference linkbases, on the other hand, add resources to concepts.

### 2.2.1 Relation Linkbases

With the help of relation linkbases, it is possible to represent hierarchical structures between the concepts of the taxonomy. These hierarchical structures are created using **XLink** and extended links. The locators here reference the concepts of the taxonomy, which are declared in the taxonomy schema. These locators are then linked via arcs. So, in XLink jargon, this is **linking two remote resources**, where the remote resources are concepts of the taxonomy. In the image below, a locator pointing to the concept "Assets" is linked via two arcs to the hierarchically underlying concepts "Current Assets" and "Non-Current Assets".



The hierarchies created in the linkbase have different meanings depending on the type of the linkbase.

**Calculation Linkbase:** The Calculation Linkbase defines simple arithmetic relationships between individual concepts. If the above example were a calculation linkbase, it would

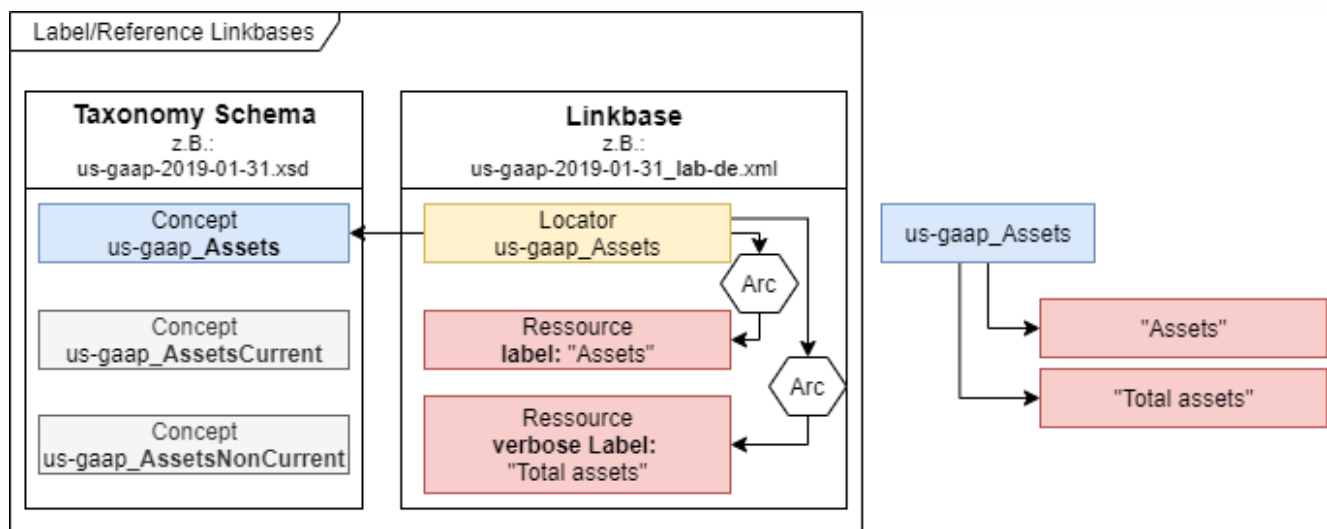
define the following equation:  $us-gaap\_Assets = us-gaap\_AssetsCurrent + us-gaap\_AssetsNonCurrent$ .

**Presentation Linkbase:** The presentation linkbase describes the order in which the concepts of the taxonomy should be arranged. The above example would subordinate the *us-gaap\_AssetsCurrent* and *us-gaap\_AssetsNonCurrent* concepts to the *us-gaap\_Assets* concept.

**Definition Linkbase:** The definition linkbase allows to create various other logical connections between concepts. For example, a link with the arcrole "essence-alias" can be used to emphasize that two concepts cover the same or very similar subject matter.

## 2.2.2 Reference Linkbases

In contrast to relation linkbases, which create connections between different concepts, resource linkbases connect concepts with related resources. These resources can be, for example, different types of labels for the concepts. So, in XLink jargon, here a **remote element is linked to several local elements** via arcs. The figure below shows a possible structure of a label linkbase. Here, the concept "Assets" is linked to the labels "Assets" and "Total Assets". Depending on the use of the concept in the financial report, one of the two labels can be selected later.



© 2021 Manuel Schmidt - manusimdt.dev

**Label Linkbase:** The Label Linkbase links concepts with one or more reader-friendly labels. It is also possible to link labels in different languages.

**Reference Linkbase:** The reference linkbase can be used to create links between concepts and documents outside of XBRL/XML. Most often, these external documents are laws or policies that govern the calculation, disclosure, or presentation of these concepts.

## 3 Instance Documents

The taxonomies discussed previously create the basic structure of reporting in XBRL and usually relate the concepts using regulatory guidelines or link them to other resources such

as labels. The Instance Document builds on top of it and uses the concepts created in the taxonomy to tag the numbers to be reported, giving them meaning and structure. Thus, the **instance document** embodies the actual business report, for example, the annual financial statement.

There are two different types of instance documents: the classic XBRL instance document (XML) and the iXBRL instance document (HTML). Both types are based on the same elements but are written in different file types.

## 3.1 Important Elements

The core elements of every Instance document are the **Facts**. A fact is a number (*150000*) combined with a context (*from Jan-Dec 2020 for company xyz*), a **Unit** (*USD*) and tagged with a **Concept** from the taxonomy (*us-gaap\_Revenue*). The **Context** defines the time frame and the company to which the fact belongs.

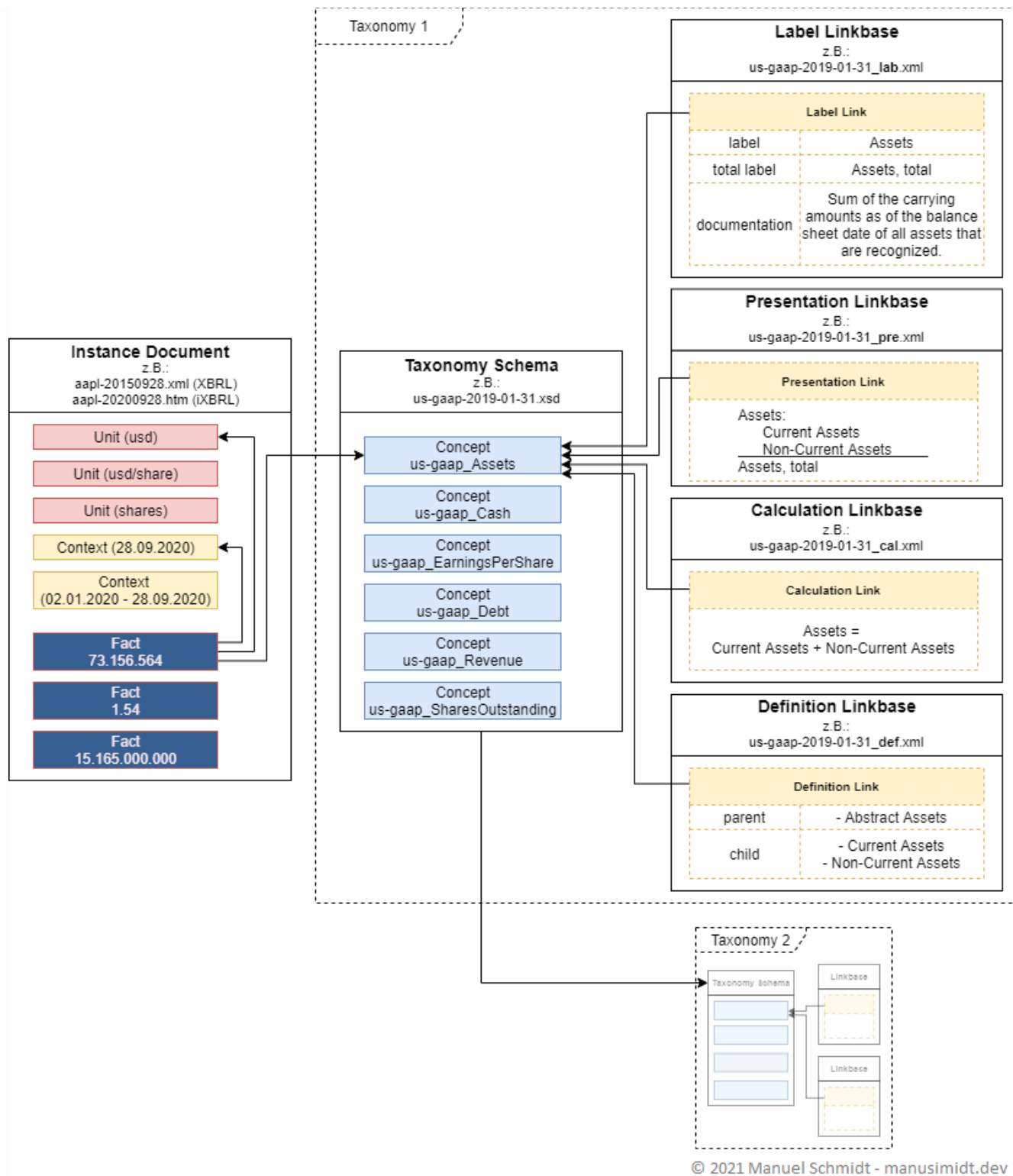
To be able to tag Facts in an XBRL Instance Document, each Instance Document must reference at least one Taxonomy. After referencing a taxonomy, any concepts defined in the taxonomy can be used to tag the facts in the instance document.

## 3.2 inline vs default Instance Documents

The issue with classic XBRL instance documents is that the filer has to provide two separate reports. One that is human readable and the XBRL Instance Document (which is basically an XML file). This leads to additional work and can lead to inconsistencies between the two files. A **inline XBRL (iXBRL) Instance Document** makes it possible to make XBRL taggings directly in an HTML file. Compared to XML files, HTML files have the decisive advantage that they can be read by an end user directly in the browser. Thus, the filer only has to create one document.





# 4 Putting it all together

The following figure shows all elements of an XBRL submission discussed before and their interrelationships.



## Manuel Schmidt

Manuel Schmidt  
hello@schmidt-manuel.de

 manusimidt  
 manusimidt  
 manusimidt  
 manusimidt

Hey, i am Manuel. I am interested in data science and machine learning.