UN/CEFACT New Projects : Information note

This page last changed on Nov 04, 2015 by marchal.

UN/CEFACT has supported the accounting industry through the development of EDIFACT and XML messages. Today the Core Component Library (CCL) has rich and broad support for the accounting industry.
The project will complement the existing support to address specific requirements for archiving of accounting systems and requirements for cloud-based systems, as identified by the project team.

Current State

The following projects have contributed accounting entities to the CCL:

- Accounting Bundle Collection
- Accounting Chart of Accounts
- Accounting Entry
- Accounting Ledger
- Accounting Trial Balance
- Accounting Reporting
- Accounting Message
- Accounting Journal

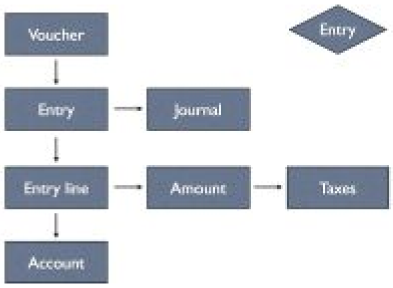
These projects addressed a number of business needs, including the archiving of an accounting system. Indeed under many jurisdictions the accountant or the organisation that maintain accounting records must preserve those records for several years. This is true for organisations of all sizes and, if the records are kept electronically (or, more accurately, «given that nowadays the records are kept electronically») then it may be required to archive them electronically as well.
To support this and other business needs identified by the projects, a significant number of accounting entities were introduced in the Core Component Library. To highlight the main Aggregate Core Components (ACC), they are:

- Accounting Account
- Accounting Account Boundary
- Accounting Account Classification
- Accounting Account Credit Risk
- Accounting Account Dimension
- Accounting Account Pattern
- Accounting Book
- Accounting Characteristic
- Accounting Check
- Accounting Entry
- Accounting Entry Line
- Accounting Journal
- Accounting Line Index
- Accounting Line Monetary Value
- Accounting Period
- Accounting Voucher

Within UN/CEFACT these ACCs are used by several sectors including accounting but also insurance, procurement and the cross-industry invoice (CI trade). They are, of course, available for all sectors and all users of the CCL.

Archiving documents

Initially archiving an accounting system raised a challenge: size.
The records include many documents (called accounting vouchers in the CCL). Those documents can be electronic documents or scans from paper documents and the scans can be very large. Yet they may need to be preserved alongside the accounting books.
However because they can be so large, the accounting entry project decided to organise the message such that they would not be repeated. So when looking at the AAA Entry_ Daybook message, it is organised as a series of vouchers (the large entities) from which the accounting entries are derived. See the diagram below:



To take a concrete example, let's consider a business meal. For simplicity, we'll assume that the employee paid for the meal and is seeking reimbursement. The employee would typically prepare a business expenses form. This form is recorded in an instance of the Aggregate Business Information Entity «AAA Entry_ Accounting Voucher» (XML element).

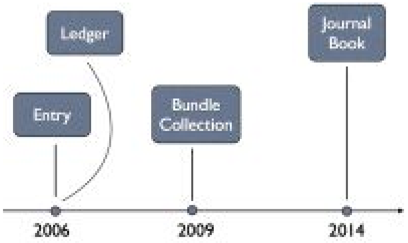
The voucher element will be associated with one instance of «AAA Entry_ Accounting Entry.» The entry itself will consist of several «AAA Entry_ Accounting Entry Line» to record charging the voucher to the accounts. This would typically result in two or three lines:

- a credit line to the third party account «employees reimbursement» (maybe with a sub-account per employee)
- a debit line to the expense account «domestic travel»
- depending on local legislation, a debit line may be required to record the taxes

This is a simple example so the voucher is recorded with only one entry line. Obviously more complex cases are possible. For example, the employee that travels may seek reimbursement for several business meals.

He or she would still fill in a business expense form (voucher) but the accountant may record a separate entry for every meal. And yet the voucher scan would appear only once in the message, which was a design requirement.

While efficient, this organisation is not natural to business users since the most typical groupings, per journal or per period, are only retained indirectly. Effectively a technical consideration has been given proeminence over the business practices. In practice it is not an insurmountable problem since it's easy to sort the entry through any criteria, including the more business-sensible ones.

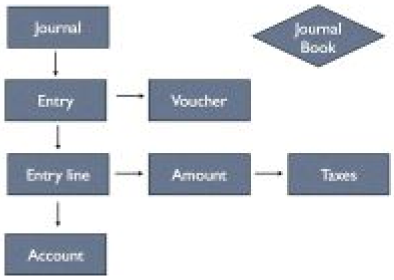


Yet in 2009, the accounting bundle was created to transmit and/or archive vouchers in a separate flow. Effectively the accounting bundle enables a separation between the accounting records (in accounting entry) and the vouchers that justify the records (in accounting bundle) and it neatly solves the problem of transferring multiple large documents. They are transferred in the bundle once and referenced (instead of copied) from the accounting entries.

The JournalBook project first aims to propose a more business-oriented alternative for users of the CCL, the alternative would organise the records around journals which complements nicely the bundle collection approach.

The following diagram helps illustrate the differences between entry and the new Journal Book project.

It should be noted that the CCL already provides different business views of the accounting information with the ledger business entities. The project is therefore coherent with past developments in UN/CEFACT.



Cloud/process/business rules

While preparing this document, the project team identified another need to help support cloud-based solutions such as web-based interactive record entry. Implementing it would further enhance the flexibility of the CCL.
Specifically cloud needs to deal with scalability: it is important to address a larger set of user as efficiently as possible. Increasingly the technical solution is to deport some of the processing to a browser/client, e.g. with JavaScript.
This however requires:

- structured information to be rendered on the client (e.g. the «raw» line information to be rendered as a table on the browser rather than an HTML table). See frameworks like AngularJS, Backbone or Ember
- efficient transfer of information to lower both the bandwidth requirements and the load on the browser

One solution to offer the later is automatic counterpart: when recording an entry line, it is seldom necessary to enter the counterpart line. The accounting application can, in many cases, automatically generate the debit line for a credit line (or vice-versa).
Likewise it is more efficient, in an interactive scenario, for the client to transmit only the recorded line and trust the server to create the counterpart.
Should this be part of a standard? The project will review these needs as well.

Conclusion

Support for accounting is already rich in the CCL. The project will further increase the usefulness of the CCL for accounting needs.