Designing a Chart of Accounts and Taking Advantage of Oracle® E-Business Suite Release 12

an eprentise white paper

tel: 407.591.4950 | toll-free: 1.888.943.5363 | web: www.eprentise.com

Author: Helene Abrams

Published: May 15, 2012
www.eprentise.com

A chart of accounts is a list of accounts used by a business to classify financial transactions. In Oracle E-Business Suite (EBS), the chart of accounts is called the accounting flexfield and is one example of a number of key flexfields in EBS that store values in a table structure for integration with other parts of the enterprise resource planning system. A good chart of accounts provides flexibility for recording and reporting accounting information, provides structure for managing business uniformly, and enhances communication across all parts of the business. A great chart of accounts takes care and consideration in the design phase, with particular focus on five key criteria considerations that will maximize the life of the accounting flexfield. Companies that do not consider their long-term growth and change will eventually find that their chart of accounts is no longer suitable for tracking financial transactions according to the current state of the business.

Note: This article assumes you have a general understanding of the components of a chart of accounts in E-Business Suite. If you don't, or if it's been awhile, take a moment to brush up by clicking here for a quick crash course.

5 fundamental criteria for a good chart of accounts design

Criteria 1: One type of information

The first of these considerations is a crucial aspect of designing a long-lived chart of accounts: You should have one (and only one) type of information in each segment. If the type of information in a segment is not unique, the chart suffers from a vulnerable overlapping of information across segments, creating potential inaccuracies during reporting. For example, consider a company that defines two of its segments as Department and Location – ideally, the Department segment will contain only departmental information and the Location segment will contain only location information. However, if the company's Department segment contains values similar to"HR – Sacramento, CA", they would have failed to keep a single information type confined to a single segment since they have location information in their Department segment. If there are transactions posted to a code combination containing the value for "HR – Sacramento, CA" but containing the location "United States", a report on the location "California" would not contain this transaction even though the transaction is related to California. Keeping information types unique to a single segment eliminates this type of mistake and solidifies the consistency of a company's financial data. Additionally, it reduces the maintenance of keeping information accurate in multiple places. As another example, if your Cost Center segment has the same type of information as a Business Unit segment, there is no need to implement both.

Criteria 2: Information not repeated

For reasons similar to those discussed above, your accounting flexfield should not repeat information that exists in other modules of EBS. If you are implementing the Oracle Projects module in your system, there is no need to have a segment in your chart of accounts that is dedicated to tracking projects transactions. All of the information you need about those transactions will be available to you in the Projects module. As another example, if you are implementing Receivables, then there is no reason to have a Customer segment in your chart of accounts. The goal is still to keep types of information in discrete buckets, only here the buckets include all of the other applications in your EBS environment.

Criteria 3: Enough room to expand

A common downfall of chart of accounts design is to tailor the design to your current business without considering the future growth and change that are inevitable outcomes of a successful company. In the design phase, make sure to define your segment lengths to be long enough to accommodate values that will be added in the future. Failure to do so will make a segment susceptible to filling up – once all of the possible values are used, desperate measures are often taken when recording transactions that result in inaccurately recorded financial information, further resulting in reporting inaccuracies that can be detrimental, especially if the company is publicly traded and accuracy of information reaches legal territory. When designing roll-up group values, a general guideline is to increment consecutive values by at least 5 within each group; however, if the group is likely to be a high-growth area of the business, a better approach is to increment by at least 10. For example, if you have a Location segment with a high-growth "United States" roll-up group, allow enough room to add ten additional values between each of your lowest levels. A portion of your Location segment hierarchy might then look like this:

10000 US11000 Midwest

  11100 Detroit Metropolitan Area

    11110 Ann Arbor

    11120 Canton

    11130 Plymouth

20000 Canada

Overestimate your segment lengths. What may seem like a lot today might be wholly inadequate in a few years.

Criteria 4: Use logical ranges

Expanding on what was just discussed about segment lengths, you should take care to put values for each segment into logical ranges. A Project segment set up in an accounting flexfield might have a range of values such as 55000 – 55999 that represents a specific type of project. There is room in this structure for at least 999 different projects of that same type, and you can tell just by looking at the number what type of project it is. If the range were 550000 – 559999, there would be room for 9,999 projects. If you know the trends in your business, you can define an appropriate number of characters for both a segment and the ranges it contains. Don't hesitate to increase the length of a segment to accommodate a larger range. Increasing segment length to accommodate growth lets you build flexibility into the COA structure.

Ranging your values logically promotes streamlined reporting, security and maintenance. It enables you to minimize the number of cross-validation rules, security rules, and Financial Statement Generator (FSG) report definitions that are necessary to support operations by giving you the ability to define your rules and reports by ranges as opposed to individual values. Similarly, avoid using intelligent numbers in your accounting flexfield. For products, where intelligent numbers can help to identify a particular product family or subgroup to a customer, devise a scheme where you have both non-intelligent and intelligent numbers for a particular product. The non-intelligent numbers will be used for internal and financial reports but the intelligent numbers can be used in a customer-facing environment. As discussed previously, avoid using alpha characters, except potentially in parent values, because these will create problems in sequencing and sorting data in reports, assigning codes, using ranges, and when creating rules. If you do use letters, use only capital letters for consistency and to facilitate any queries you may need to run.

Criteria 5: No reliance on spreadsheets

Many companies ignore the high cost of maintaining spreadsheets because they have already purchased a spreadsheet package for a minimal license fee. Spreadsheets are great tools for short, small projects, but care should be taken when using them as an extensive resource, as is often done for financial reporting and consolidation. Spreadsheets are good choices when you need to organize simple data in a fast and cost-effective manner, but they are generally ill-equipped to handle accurately and efficiently transforming data from your financial systems to report results. Eliminating your reliance on spreadsheets to get the financial information you need is a best-practices consideration when designing your chart of accounts. Make sure that you will be able to get all of the information you need from the built-in reports of EBS. This will add value by giving you real-time access to information and eliminating numerous errors that are introduced when multiple employees use the same spreadsheets and are unaware of changes and updates made by others. Eliminating spreadsheets from financial efforts promotes a streamlined workflow, reduces both resource requirements and maintenance, and eliminates the need to integrate your EBS environment with 3rd party reporting applications.

You will get faster reports if you focus on the built-in reporting features of EBS. FSGs can be kept to a minimum if you take the time to design a robust master row set, allowing you to generate different reports without requiring that you rewrite each report. The time required to obtain a custom reports can be reduced from days to under an hour.

Taking advantage of Release 12 with a global chart of accounts

New accounting features introduced in EBS Release 12 promote the adoption of a single chart of accounts across the organization. Companies are given the opportunity to consolidate multiple accounting flexfields into a single flexfield or to restructure their existing flexfield to take advantage of R12 features that automatically take care of functionality that required their current charts to become overgrown and convoluted. In R12, sets of books are replaced by ledgers, which bring along with them the welcome addition of Subledger Accounting (SLA) that includes new features of Primary Ledgers, Secondary Ledgers, Ledger Sets, Reporting Currencies, and Autoaccounting Rules.

A ledger can have its own accounting method, calendar, currency, and chart of accounts. Ledgers can be combined into ledger sets for processing all transactions. This means that a user can enter a single transaction in the primary ledger, and by defining create accounting rules and events, can update multiple ledgers without having to reenter the same information for each transaction. Furthermore, ledger sets allow users to open and close periods, create journals, and translate and revalue balances across ledgers.

In a global environment, a company might have ledgers to comply with statutory requirements of a particular region or to report on the same data in a different way. For example, most organizations with foreign subsidiaries need to prepare their consolidated financial statements according to IFRS (International Financial Reporting Standards) and also US GAAP standards. Using SLA, a company can have one ledger that complies with IFRS and one that complies with GAAP accounting methods, post the transactions one time, and generate the consolidated reports from both ledgers. When a user posts a transaction type to the primary ledger, the "Create Accounting Function" uses a set of rules to update the statutory ledger. It is no longer necessary or a good idea to have multiple charts of accounts for reporting different ways – everything is taken care of by setting up different ledgers for different purposes, drastically reducing the time necessary for compliance with multiple requirements.

In R12, Oracle also introduces a single accounting engine to manage all posting activities into the general ledger. In prior releases, modules such as AR and AP contained their own rules for posting accounting events into the general ledger. The universal posting engine streamlines the close process. By applying standard accounting rules to all business transactions, SLA ensures consistent financial reporting. If the business uses a single chart of accounts, the user only needs to define the rules to create the transactions for posting to the GL or to subledgers one time, and then all the related subsidiary ledgers in a ledger set will be updated automatically. Users need to support only one set of rules, and maintenance of rules is simplified. A single chart of accounts reduces the time spent compiling, reconciling, and consolidating financial data from disparate systems and spreadsheets and reduces the close time between the different modules and the general ledger. With a single chart of accounts and SLA, accounting policies are standardized across the entire enterprise and everyone adheres to the same set of rules and definitions. The data remains consistent, has full drill-down and roll-up capability, auditability, and visibility into all of the activity for the entire ledger set. In addition, a single chart of accounts eliminates conversions required for data warehouse queries, facilitates the movement to a shared service center, and increases the level of enterprise governance and control of new code combinations.

Conclusion

Use the transition to R12 to redesign your chart of accounts if it is not optimized or to adopt a single chart of accounts if your organization is currently using multiple accounts. As you are redesigning, focus on the five fundamental criteria discussed previously. In respect to Subledger Accounting, design your new chart with Ledger Sets, Secondary Ledgers, and Autoaccounting rules in mind. If your company is a large, global organization, there are a few additional considerations and recommendations. Add an intercompany segment to you chart to take advantage of R12's Advanced Global Intercompany (AGIS). Add a segment to accommodate local requirements such as local bank accounts and statutory reporting. A Location segment is optional, but it helps with security and cross-validation rules. Implement other Oracle modules such as Project Accounting and Collections for detailed tracking at a local level. Finally, take advantage of R12's Multiple Reporting Currencies to report in different currencies.

With R12, the EBS accounting process has become more advanced at the application level, but this correlates into a simpler user experience, increased functionality, and reduced operational expense. In order to gain these advantages, it is highly advised to adopt a single chart of accounts and design it accordingly. Don't miss out on a great opportunity to get a step ahead of the competition.

Curious?
For more information, please call **eprentise** at **1.888.943.5363** or visit **www.eprentise.com.**

About eprentise

eprentise provides transformation software products that allow growing companies to make their Oracle® E-Business Suite (EBS) systems agile enough to support changing business requirements, avoid a reimplementation and lower the total cost of ownership of enterprise resource planning (ERP). While enabling real-time access to complete, consistent and correct data across the enterprise, **eprentise** software is able to consolidate multiple production instances, change existing configurations such as charts of accounts and calendars, and merge, split or move sets of books, operating units, legal entities, business groups and inventory organizations.