

How to Write a Plug-In Interface

Overview

CommissionCalc™ is a software application which computes sales-based incentive from data in an ERP database. It is described at www.CommissionCalc.com.

This spec defines a standard set of tables, the "CC Input Tables", from which CommissionCalc can read data. The CC Input Tables are stored in a Microsoft SQL database.

Database Schema

The columns listed for each table below are required. In addition, you can add whatever other columns are used to compute your commission.

I. Accounts Receivable tables with transactional data.

These 2 tables contain financial transactions. They do not contain product data.

- A. CC_ArTransactions. This required table contains one row for each invoice, payment, credit memo, or debit memo. (See section V.D for a definition of *payment*.)

1. Required columns

- a) ArDocType. (See section V.B for a listing of document types.)
- b) ArDocId. Ordinarily, the ERP system assigns a unique ID to each CC_ArTransactions row. Use that number. For example, if the row represents an invoice, ArDocId is the invoice number. Rarely, the ERP system does not assign a unique ID to each CC_ArTransactions row. In this case, create an ID by concatenating columns or by using an ERP2CC identity column.
- c) CustomerId
- d) DocDate
- e) DueDate. If the row is not an invoice, Null.
- f) ArCreditType. (See section V.C for a listing of credit types. If your commission plan does not require this information, you may leave this column blank.)
- g) Amount. The net amount of the transaction in the home currency, after all discounts and other adjustments which are included in the row. If ArDocType is 'R', 'C', or 'P', the sign will always be negative. If ArDocType is 'I', the sign will ordinarily be positive; however, if the ERP system permits invoices for negative quantities and amounts, the sign may occasionally be negative. If ArDocType is 'D', the sign will always be positive.

2. Indexes

- a) Primary: ArDocType + ArDocId
 - b) Secondary: CustomerId + DocDate
- B. CC_ArApplications. This required table contains one row for each application of a payment, credit memo, or debit memo to an invoice. If a payment represents a single application of a check to an invoice, there is one row in CC_ArApplications for each payment row in CC_ArTransactions. If a payment represents a check which might be applied to several invoices, there may be multiple rows in CC_ArApplications for each payment; see Appendix A for an illustration of this multiple-row design.
- 1. Required columns
 - a) DateApplied
 - b) CustomerId
 - c) ApplyFromArDocType
 - d) ApplyFromArDocId
 - e) ApplyToArDocType. This must always be "I".
 - f) ApplyToArDocId
 - g) Amount. This is the net amount of the application, in the home currency, after all discounts and other adjustments included in the row. The sign should be negative if the application reduces the amount due and positive otherwise.
 - 2. Indexes
 - a) Primary: ApplyFromArDocType + ApplyFromArDocId.
ApplyToArDocType and ApplyToArDocId may be appended if it is desired to make this index unique, and therefore useable as a clustered index.
 - b) Secondary:
 - (1)ApplyToArDocType + ApplyToArDocId.
 - (2)CustomerId.

II. Sales History Tables.

These 2 tables normally contain data about product which has been invoiced or returned, including item numbers, quantities invoiced, prices, and the like. Exception: if commission is paid when orders are booked, sales orders can be copied to these tables *before* they are invoiced. Such orders should be coded as preliminary invoices; please discuss this with FTI.

- A. CC_InvoicedOrderHeader. This required table normally contains one row for each invoice or return. It has a one-to-many relationship with CC_InvoicedOrderLines. Because a single sales order can be shipped in multiple parts, with each part having a separate invoice, it has a many-to-one relationship with sales orders.
- 1. Required Columns
 - a) Link to AR invoice or return (i.e. credit memo).
 - (1)ArDocType. (See section V.B for a listing of document types.)
 - (2)ArDocId
 - b) CustomerId

- c) Addresses. These columns are used for ERP systems which store addresses in a separate table and reference them by an address code. If the addresses are stored in columns which will be part of `CC_InvoicedOrderHeader`, they can be {blank} or Null.
 - (1) `BillToAddressId`
 - (2) `ShipToAddressId`
 - 2. Indexes
 - a) Primary: `ArDocType + ArDocId`.
 - b) Secondary: `CustomerId`.
- B. CC_InvoicedOrderLines. This required table contains one row for each item on the invoice, return, or sales order.
 - 1. Required columns
 - a) Link to AR invoice or return (i.e. credit memo).
 - (1) `ArDocType`. (See section V.B for a listing of document types.)
 - (2) `ArDocId`
 - b) Line identification
 - (1) `OrderLineSeq`.
 - (2) `OrderLineKitSeq`. Usually zero. `OrderLineKitSeq` is a second segment for `OrderLineSeq` which is intended to support "kitting", a feature of some ERP systems which allows invoice lines to contain subordinate lines representing kit components.
 - c) Line data: `ItemId`
 - 2. Primary key: `ArDocType + ArDocId + OrderLineSeq + OrderLineKitSeq`.

III. Master Data

- A. CC_ItemMaster. Optional table.
 - 1. Required column: `ItemId`
 - 2. Primary key: `ItemId`
- B. CC_CustomerMaster. Required table.
 - 1. Required columns
 - a) `CustomerId`
 - b) `CustomerName`
 - 2. Primary key: `CustomerId`
- C. CC_SalespersonMaster. Required table.
 - 1. Required columns
 - a) `SalespersonId`
 - b) `Name`
 - 2. Primary key: `SalespersonId`
- D. CC_Addresses. Optional table.
 - 1. Required columns

- a) CustomerId
 - b) AddressId. This code must be unique for any given customer.
It may also be globally unique, but that is not required.
2. Primary key: CustomerId + AddressId.

IV. ERP Commission Data. These tables are optional and are usually not used; the data is often stored in the sales history tables instead. Typically they are used if (a) you want the ERP system to identify an unlimited number of salespeople for each order or order line or (b) the ERP system has similar tables so using these simplifies programming ERP2CC. You may use either one or both of these tables.

A. CC_CommissionSplits

- 1. Data columns
 - a) ArDocType
 - b) ArDocId
 - c) CustomerId
 - d) SplitsSeq, where SplitsSeq is 1 for the first salesperson associated with an ERP order header, 2 for the second, etc.
- 2. Primary key: ArDocType + ArDocId + SplitsSeq

B. CC_CommissionLineSplits.

- 1. Data columns
 - a) ArDocType
 - b) ArDocId
 - c) OrderLineSeq
 - d) CustomerId
 - e) SplitsSeq, where SplitsSeq is 1 for the first salesperson associated with an ERP order line, 2 for the second, etc.
- 2. Primary key: ArDocType + ArDocId + OrderLineSeq + SplitsSeq

V. Miscellaneous Information

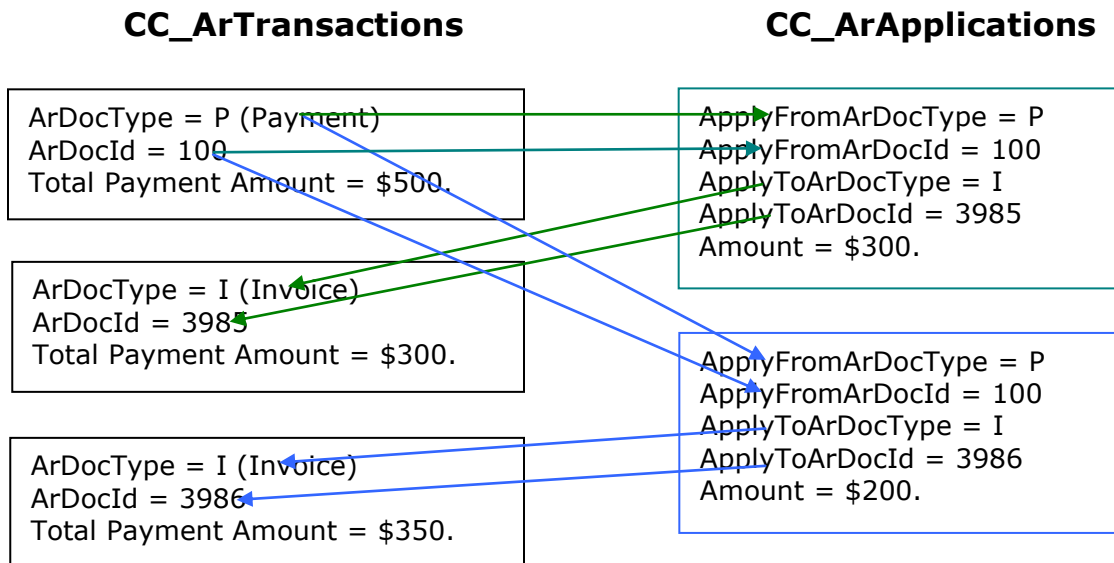
- A. Table Name Suffixes. CommissionCalc might be executed by multiple users simultaneously, all using the same SQL database. Therefore, two or more copies of each of the above tables, and any ERP2CC work tables, might exist simultaneously. To make this possible, the actual database name for each table will have a suffix consisting of an underscore and the computer name of the workstation on which CommissionCalc is executing. For example, if CommissionCalc is executed on computer ABC, the first table above will actually be named "CC_InvoicedOrderHeader_ABC". (Implementation note: Do not use the Windows API function GetComputerName or the VB.Net property My.Computer.Name to read the computer name because they convert lower case letters to capitals; Microsoft acknowledges that this (or, anyway, part of it) is a defect which they will change in a future release. ERP2CC can use the VB.Net property Dns.GetHostName() because this returns a properly capitalized name regardless of the version of Windows being used.)
- B. ArDocType. CHAR(1). The following values can be used. A plug-in interface need not use all of these values. Types A and S can be used only if CC_CommissionSplits and CC_CommissionLineSplits are not used.
1. I = Invoice
 2. P = Payment
 3. R = Return (i.e. credit with item data)
 4. C = Credit Memo (i.e. credit without item data)
 5. D = Debit Memo (i.e. no item data)
 6. A = Adjustment.
CommissionCalc treats these exactly the same as type C. It is provided because some accounting systems have separate adjustment transactions and it might be desirable to maintain the distinction between A & C for user convenience. Note: With some accounting systems, adjustments can change values which credit memos cannot change, such as due date. CommissionCalc does not support changing these values.
 7. S = Discount for prompt payment.
- C. ArCreditType. CHAR(1). Many ERP systems have columns in AR transaction tables which are dedicated to specific, common types of credits. ERP2CC will create separate CC_ArApplications rows for these credits and use ArCreditType indicate the source of the credit as follows. If a specific ERP system requires more than the following codes, please discuss the requirements of that system with FTI.
1. W = write off
 2. P = discount for prompt payment
 3. M = miscellaneous credit (i.e. any credit other than the above)
 4. {blank} = ArDocType <> 'C'
- D. Two alternative definitions of *payment* are used by different ERP systems. A *payment* can represent a single check, wire transfer, or other transfer

of funds; thus, one payment could be applied to several invoices. Alternatively, a "payment" can be the portion of a check, etc. which is applied to a single invoice. It will probably simplify development of ERP2CC if you use the same definition as your ERP software uses.

Appendix A

Payment Application

The following diagram shows how a \$500 check would be applied to 2 invoices when the first definition of *payment* in section V.D is used. The first invoice is paid in full. The second invoice still has an open balance of \$150.



NB: Each invoice, payment, or other A/R document is represented by exactly one row in **CC_ArTransactions**, regardless of how many other A/R documents it is associated with. Instead of splitting payments into multiple rows to represent multiple applications or splitting invoices to reflect the amount paid by each check, transactions are distributed via the **CC_ArApplications** table.