
Product Name	Order Connector
Date	11/13/06

1.0 PRODUCT DESCRIPTION

The Order Connector allows integration of an external system with Great Plains. Customers, Sales Orders and Payments from customers are created by the interface using information saved in a “Buffer Zone”. The Order Connector consists of stored procedures that can be called from the external application or scheduled to be executed on the SQL server.

2.0 TECHNICAL DESCRIPTION

2.1 CUSTOMER AND ADDRESS INTERFACE

Customers and their respective Addresses shall be created using data received on the order.

- Customers shall be created by copying defaults from a Customer Master called “MASTER”. Therefore, when a new customer is created, all settings related to the customer configuration (except address) shall be taken from the “MASTER” customer.
- If the Customer ID already exists, the address information will be updated using the information on the order.
- The two possible Address Types shall be “BILLING” and “SHIPTO”.
- We will create a table called INTEGR8_Tax_Schedule_Mapping which will map the Customer’s State to an appropriate Tax Schedule ID in Great Plains
 - Tax calculations are only performed to verify the calculations performed at the web application. The tax amount calculated at the web application is always used, even if it is wrong.

2.2 SALES TRANSACTION INTERFACE

Sales Transactions and Credit Card Receipts shall be created in Great Plains according to the following flow:

- 1) Credit Card payment is confirmed by the website and it immediately saves information to the “buffer zone”
- 2) A timed process will scan the “buffer zone” for new orders and it will call stored procedure INTEGR8_Create_Sales_Order

-
- 3) The stored procedure INTEGR8_Create_Sales_Order shall obtain all data from the “buffer zone” and it shall create (or update) the Customer Master, the Sales Order Transaction and the Credit Card Payment.
 - 4) If the Sales Order is created successfully in Great Plains, stored procedure shall save a “SUCCESS” status to one of the tables in “buffer zone”. Otherwise, it will write a “FAILURE” status and an Error Description

2.3 THE “BUFFER ZONE”

The Buffer Zone consists of a set of tables in the COMPANY database in Great Plains.

The tables in the Buffer Zone shall be:

a) INTEGR8_Order_Header

Most of the fields in this table are named exactly as the fields in the table used by the website.

When creating a record in this table, use the following values:

Status = “NEW”

ErrorDescription = “”

DateTimeRequested = current date and time

DateTimeProcessed = ‘1900-01-01 00:00:00.0000’

Script for table creation:

```
-- The Status field shall contain the string 'NEW' for new orders. Once the
Order is imported, this field is updated with either 'SUCCESS' or 'FAILURE'
-- The ErrorDescription field shall contain the reason why the order was not
imported (if Status = 'FAILURE') or any warnings (for example, tax
calculation from web front is different from tax calculation in GP
-- The DateTimeProcessed is updated with the date and time the record was
processed by the interface
```

```
CREATE TABLE [dbo].[AS_Order_Header] (
    [IDOrder] [varchar] (20) NOT NULL ,
    [OrderDate] [varchar] (30) NULL ,
    [IDCustomer] [varchar] (15) NULL ,
    [Address1] [varchar] (30) NULL ,
    [Address2] [varchar] (30) NULL ,
    [Address3] [varchar] (30) NULL ,
    [City] [varchar] (30) NULL ,
    [StateCode] [varchar] (30) NULL ,
    [Zip] [varchar] (10) NULL ,
    [CountryCode] [varchar] (4) NULL ,
    [BillToEmail] [varchar] (100) NULL ,
    [BillToPhone] [varchar] (21) NULL ,
    [BillToFax] [varchar] (21) NULL ,
    [TaxAmount] [numeric] (19, 5) NULL ,
```

```

[FreightAmount] [varchar] (50) NULL ,
[MiscellaneousAmount] [numeric](19, 5) NULL ,
[DiscountAmount] [numeric](19, 5) NULL ,
[Total] [numeric](19, 5) NULL ,
[ShippingMethodCode] [varchar] (10) NULL ,
[ShipToName] [varchar] (64) NULL ,
[ShipToAddress1] [varchar] (30) NULL ,
[ShipToAddress2] [varchar] (30) NULL ,
[ShipToAddress3] [varchar] (30) NULL ,
[ShipToCity] [varchar] (50) NULL ,
[ShipToStateCode] [varchar] (30) NULL ,
[ShipToZip] [varchar] (10) NULL ,
[ShipToCountryCode] [varchar] (30) NULL ,
[ShipToEmail] [varchar] (100) NULL ,
[ShipToPhone] [varchar] (21) NULL ,
[OrderConfirmationNumber] [varchar] (30) NULL ,
[Status] [varchar] (15) NULL ,
[ErrorDescription] [varchar] (150) NULL ,
[DateTimeRequested] [datetime] NULL ,
[DateTimeProcessed] [datetime] NULL
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[AS_Order_Header] ADD
CONSTRAINT [PK_AS_Order_Header] PRIMARY KEY NONCLUSTERED
(
[idOrder]
) WITH FILLFACTOR = 80 ON [PRIMARY]
GO

```

b) AS_Order_Detail

Script for table creation:

```
-- The Order Detail table shall contain all line items from the order. The
defaults are pulled from the Item Master from Great Plains. If the Item does
not exist in Great Plains, the order is not imported
```

```

CREATE TABLE [dbo].[AS_Order_Detail] (
[IDOrder] [varchar] (20) NOT NULL ,
[LineItemSequence] [int] NOT NULL ,
[ItemNumber] [varchar] (25) NULL ,
[ItemDescription] [varchar] (50) NULL ,
[QTYOrdered] [numeric](19, 5) NULL ,
[UnitPrice] [numeric](19, 5) NULL
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[AS_Order_Detail] ADD
CONSTRAINT [PK_AS_Order_Detail] PRIMARY KEY NONCLUSTERED
(

```

```

        [idOrder],
        [LineItemSequence]
    ) WITH FILLFACTOR = 80 ON [PRIMARY]
GO

```

c) AS_Credit_Card

Script for Table Creation

-- The Credit Card table shall contain all amounts tendered via Credit Card. The table allows more than one Credit Card accepted per order. For each card, increment SeqCode by 1.

```

CREATE TABLE [dbo].[AS_Credit_Card] (
    [IDOrder] [varchar] (20) NOT NULL ,
    [CardType] [varchar] (50) NULL ,
    [CardNumber] [varchar] (80) NULL ,
    [Expiration] [varchar] (30) NULL ,
    [SeqCode] [varchar] (6) NULL ,
    [AuthorizationCode] [varchar] (150) NULL
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[AS_Credit_Card] ADD
    CONSTRAINT [PK_AS_Credit_Card] PRIMARY KEY NONCLUSTERED
    (
        [idOrder]
    ) WITH FILLFACTOR = 80 ON [PRIMARY]
GO

```

c) AS_Tax_Schedule_Mapping

This table will be used by both the Customer Interface and the Sales Transaction Interface to determine what Tax Schedule ID from Great Plains shall be applicable.

Script for Table Creation:

-- The tax schedule mapping table shall contain one record for each state where the company has presence. Therefore orders placed by customers from these states shall be taxed. This table is primarily
 -- used to compare the tax calculated by the shopping cart application against the tax calculated by Great Plains.

```

CREATE TABLE [dbo].[AS_Tax_Schedule_Mapping] (
    [StateCode] [varchar] (4) NOT NULL ,
    [TaxScheduleID] [varchar] (25) NULL
) ON [PRIMARY]
GO

```

```

ALTER TABLE [dbo].[AS_Tax_Schedule_Mapping] ADD

```

```
CONSTRAINT [PK_AS_Tax_Schedule_Mapping] PRIMARY KEY NONCLUSTERED
(
    [StateCode]
) WITH FILLFACTOR = 80 ON [PRIMARY]
GO
```

2.4 ADDITIONAL DEFINITIONS

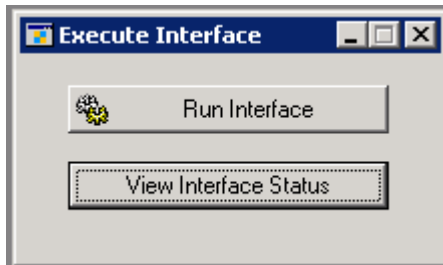
All orders created through the interface shall have the following defaults:

- a) When Customers are created, the code shall also create the Customer Master Summary Table (RM00103). When orders are created, the fields Unposted Sales Amounts, Unposted Cash Amounts and On Order Amount shall be updated
- b) Document ID for the Sales Order shall be obtained from the Order ID field from the Sales Order Processing Setup window.
- c) Default Site ID (and all Line Item Site IDs) will be obtained from the Site ID field from the Sales Order Processing Setup window.
- d) The Sales Order Number in Great Plains will come from field INTEGR8_Order_Header.idOrder. Therefore, the order number will NOT come from the Order ID Next Number field from the Sales Order Setup window.
- e) All GL account numbers will come from both the Customer Card and the Item Card
- f) Order Confirmation Number from the website shall be saved to Text Field 3 (in the Sales User-Defined Fields Entry).
- g) Batch IDs shall be named as follows:
"INT" + "-" + MM/DD/YYYY
- h) Credit Card payments from the website shall be "attached" to the Sales Order as a Credit Card Payment type in the Sales Payment Entry window.
- i) All items in sales orders shall be allocated from inventory irrespective of their current available quantity
- j) If the order is created successfully in Great Plains, the procedure INTEGR8_Create_Sales_Order shall save the value "SUCCESS" to field Status. Otherwise, it will save the value "FAILURE" and one of the following Error Descriptions mentioned in item (K)
- k) These are the possible errors to be detected during sales order creation:
 - Order Number Already Exists in Great Plains
 - Required Order Fields are Missing
 - Credit Card Type Does Not Exist in GP
 - Shipping Method Does Not Exist in GP

2.5 THE INTERFACE LOG AND EXECUTING THE ORDER CONNECTOR ON DEMAND

A log is created every time the Order Connector is executed. This log can be viewed within Great Plains via the Import Process Status Inquiry window, which is opened via the Execute Interface window. The Execute Interface window also allows the user to run the interface on demand.

Path: Tools >> Utilities >> Sales >> Order Interface



When the user selects the View Interface Status option, the Import Process Status Inquiry window is opened:

Developers can access the Log by querying tables ALB30100 and ALB30200 in the company database.