Conveyorware Business distribution software

Conveyorware API User Guide
Application Program Interface

RESTful (Representational state transfer) API to submit orders, submit new inventory items, submit new customer accounts, request an inventory file, and request a shipping feedback. This API was written using Java and Spring MVC framework for running on IBMi midrange computer (former AS400). All URL samples in the guide are GET requests. If your GET request URI is too long and you get the following error:

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>414 Request-URI Too Large</TITLE>
</HEAD><BODY>
<H1>Request-URI Too Large</H1>
<P>The requested URL's length exceeds the capacity limit for this server.
</P>
</body></html>

You need to submit the request as a POST.
You can find examples of GET and POST requests here:
https://fmw.ferbermidwest.com/cwapi/CwApiHTTPClientExample.zip

Submit orders to the system

To submit orders you need to have an http client that will send a request in this format:
https://fmw.ferbermidwest.com/cwapi/ordersSubmit?orderFile=xml

Where:

Https is a secure socket layer protocol
fmw.ferbermidwest.com is the url
cwapi/ordersSubmit is the api path
orderFile is the parameter
xml is the parameter value

The Parameter value (XML) is a string that presents xml file.

Sample request of the order XML file:

<OrderSubmitRequest>
<UserId>USERID</UserId>
<Orders>
<Order>
<MainValues>
<CompanyNumber>9</CompanyNumber>
<CustomerNumber>99</CustomerNumber>
<OrderNumber>ordernumber</OrderNumber>

```
</MainValues>
<ShipTo>
<FullName>FullName</FullName>
<StreetAddress1>StreetAddress1</StreetAddress1>
<City>City</City>
<State>ST</State>
<Zip>11111</Zip>
<Country>US</Country>
</ShipTo>
<ShipVia>
<ShippingMethod>USFRST|FIRST</ShippingMethod>
<Residential>No</Residential>
</ShipVia>
<Items>
<Item>
<ItemNumber>ABC</ItemNumber>
<Qty>1</Qty>
</Item>
</Items>
</Order>
</Orders>
</OrderSubmitRequest>
```

XML string can be without any line breaks. The Server will send response in XML format after request is received.

Incoming request will be validated against the following schema:
https://fmw.ferbermidwest.com/cwapi/OrderSubmitRequest.xsd.

Sample of server response with failed validation:

```
<OrderSubmitRespond>
<OrdersSubmitted>
<MessageId>0002</MessageId>
<MessageText>false</MessageText>
</OrdersSubmitted>
<XmlFileError>
<MessageId>0006</MessageId>
<MessageText>
Message of XML Validator: org.xml.sax.SAXParseException: cvc-pattern-valid: Value '9999' is
not facet-valid with respect to pattern '[0-9]{1,3}' for type '#AnonType_CompanyNumber'.
</MessageText>
</XmlFileError>
<UserIdStatus>
<MessageId>0010</MessageId>
<MessageText>*ENABLED</MessageText>
</UserIdStatus>
</OrderSubmitRespond>
```

The XML schema (OrderSubmitRequest.xsd) describes all required fields for creating orders in ConveyorWare system. Some of the fields are optional, some not. Sample of XML file shows only required fields.

To prevent infinite writing of the data to ConveyorWare system you can send no more than 5.5MB of data in one request and no more than 10MB of data during one day. It is approximately 2,500 orders in one request and about 4,500 orders during the day. The system will not limit amount of requests per day. For example, if you sent 3 requests with size of 9MB and next request with size of 4MB, both will be accepted (total 13 Mb) but the next request will be rejected even if the request size is very small.

To successfully submit orders you need to have a valid user id which works in ConveyorWare at this url: https://fmw.ferbermidwest.com/cw. Also it can be helpful but not necessary to have any knowledge about manual order entry processing in ConveyorWare.

In addition to XML validation, the API does field value validation. API validates these fields with fatal result (rejecting orders): UserId, CompanyNumber, CustomerNumber, WarehouseNumber, SalesPersonNumber. Other fields without fatal result (orders created but not processed until user makes corrections): ShippingMethod, UPSOptionalService, ThridPartyBillingAccount, ThridPartyAddress, Zip code, among others.
Sample of server response with failed field validation:

<OrderSubmitRespond>
<OrdersSubmitted>
<MessageId>0002</MessageId>
<MessageText>false</MessageText>
</OrdersSubmitted>
<XmlFileError>
<MessageId>0005</MessageId>
<MessageText>XML file successfully validated</MessageText>
</XmlFileError>
<UserIdStatus>
<MessageId>0010</MessageId>
<MessageText>*ENABLED</MessageText>
</UserIdStatus>
<Orders>
<Order>
<OrderNumber>OrdNmb</OrderNumber>
<OrderSequenceInXmlFile>1</OrderSequenceInXmlFile>
<CustomerNumber>
<MessageId>0018</MessageId>
<MessageText>Customer Account 90000009909 does not exist</MessageText>
</CustomerNumber>
<WarehouseNumber>
<MessageId>0020</MessageId>
<MessageText>Warehouse 19 does not exist</MessageText>
</WarehouseNumber>
<SalesPersonNumber>
<MessageId>0024</MessageId>
<MessageText>Customer number is wrong</MessageText>

```
</SalesPersonNumber>
<USAZipCode>
<MessageId>0011</MessageId>
<MessageText>US Zip Code: 8306 must be 5 digits numeric field including leading
zeros</MessageText>
</USAZipCode>
<ItemNumber>
<MessageId>0023</MessageId>
<MessageText>Item Number: ABCABC is not in inventory file</MessageText>
</ItemNumber>
</Order>
</Orders>
</OrderSubmitRespond>
```

Sample of server respond with passed field value validation but with requirement of user manual corrections:

```
 <OrderSubmitRespond>
<OrdersSubmitted>
<MessageId>0001</MessageId>
<MessageText>true</MessageText>
</OrdersSubmitted>
<XmlFileError>
<MessageId>0005</MessageId>
<MessageText>XML file successfully validated</MessageText>
</XmlFileError>
<UserIdStatus>
<MessageId>0010</MessageId>
<MessageText>*ENABLED</MessageText>
</UserIdStatus>
<Orders>
<Order>
<OrderNumber>OrdNmb</OrderNumber>
<OrderSequenceInXmlFile>1</OrderSequenceInXmlFile>
<USAZipCode>
<MessageId>0011</MessageId>
<MessageText>US Zip Code: 8306 must be 5 digits numeric field including leading
zeros</MessageText>
</USAZipCode>
<ItemNumber>
<MessageId>0023</MessageId>
<MessageText>Item Number: ABCABC is not in inventory file</MessageText>
</ItemNumber>
</Order>
</Orders>
</OrderSubmitRespond>
```

Response with successful validation:

```
<OrderSubmitRespond>
<OrdersSubmitted>
<MessageId>0001</MessageId>
<MessageText>true</MessageText>
</OrdersSubmitted>
</OrderSubmitRespond>
```

Submit new inventory items to the system

To submit new inventory items you need to have an http client that will send a request in the following format:
https://fmw.ferbermidwest.com/cwapi/inventorySubmit?inventoryFile=xml

Where:

Https is a secure socket layer protocol
fmw.ferbermidwest.com is the url
cwapi/inventorySubmit is the api path
inventoryFile is the parameter
xml is the parameter value

The Parameter value (XML) is a string that represents the xml file.

Sample request of the submitted inventory XML file:

```
<InventorySubmitRequest>
<UserId>userid</UserId>
<InventoryFile>
<CompanyNumber>987</CompanyNumber>
<InventoryRecord>
<ItemNumber>A_AWEB_A1</ItemNumber>
<ItemDescription>ABC</ItemDescription>
</InventoryRecord>
<InventoryRecord>
<ItemNumber>A_AWEB_A2</ItemNumber>
<ItemDescription>ABCD</ItemDescription>
</InventoryRecord>
</InventoryFile>
</InventorySubmitRequest>
```

The XML string can be without any line breaks. The Server will send respond in XML format after request is received.

Incoming request will be validated against the following schema:
https://fmw.ferbermidwest.com/cwapi/InventorySubmitRequest.xsd.

The XML schema (InventorySubmitRequest.xsd) describes all fields for creating inventory records in the ConveyorWare system. Some of the fields are optional. Sample of the XML file shows only required fields.

To prevent heavy load on the ConveyorWare system you can send no more than 5.5MB of data in one request and no more than 10MB of data during one day. It is approximately 2,500 inventory records in one request and about 4,500 inventory records during the day. The system will not limit amount of requests per day. For example, if you sent 3 requests with a size of 9MB and the next request with a size of 4MB, both will be accepted (total 13 Mb) but the next request will be rejected even if the request size is very small.

To successfully submit new inventory items you need to have a valid user id which works in ConveyorWare at this url: https://fmw.ferbermidwest.com/cw. Also it can be helpful but not necessary to have any knowledge about manual inventory entry processing in ConveyorWare.

In addition to XML validation, the API does field value validation. The API validates all fields with a fatal result (rejecting submitted inventory file) if something is wrong, such as: invalid user id, or duplicate items, or duplicate items upc,… . To make submitting go smoothly, it's recommended to create a few test items manually using opt 301 in main menu of ConveyorWare or Order Management inventory part (https://fmw.ferbermidwest.com/fmw).
Required fields are UserId, CompanyNumber, ItemNumber, and ItemDescription. If you will omit other fields in the xml file then system will use default values without any errors.
The schema (https://fmw.ferbermidwest.com/cwapi/InventorySubmitRequest.xsd) has a list and requirements of all fields in the inventory file. Additional information about fields can be found in the ConveyorWare field sensitive help.

Sample of server response with failed fields validation:

<InventorySubmitRespond>
<InventorySubmitted>
<MessageId>0002</MessageId>
<MessageText>false</MessageText>
</InventorySubmitted>
<XmlFileError>
<MessageId>0005</MessageId>
<MessageText>XML file successfully validated</MessageText>
</XmlFileError>
<UserIdStatus>
<MessageId>0010</MessageId>
<MessageText>*ENABLED</MessageText>
</UserIdStatus>
<InventoryRecord>
<CompanyNumber>987</CompanyNumber>
<ItemNumber>A_AWEB_A1</ItemNumber>
<ItemSequenceInXmlFile>1</ItemSequenceInXmlFile>
<ItemNumberError>
<MessageId>0030</MessageId>
<MessageText>Item Number A_AWEB_A1 already exists in inventory file (duplicate item)</MessageText>
</ItemNumberError>
</InventoryRecord>
</InventorySubmitRespond>

Response with successful validation:

<InventorySubmitRespond>
<InventorySubmitted>
<MessageId>0001</MessageId>
<MessageText>true</MessageText>
</InventorySubmitted>
</InventorySubmitRespond>

Request an inventory file

Inventory file can be requested in CSV or XML format.
To request a CSV file, use the following link:
https://fmw.ferbermidwest.com/cwapi/inventory.csv?user=userid&cmpn=cmp&whsn=whs

Where:

Https is a secure socket layer protocol
fmw.ferbermidwest.com is the url
cwapi/inventory.csv is the api path
user is the parameter name (user has to have valid user id in ConveyorWare System)
userid is the parameter value
cmpn is the parameter name – present company number in ConveyorWare System
cmp is the parameter value (it is a number between 1 and 999)
whsn is the parameter name – present warehouse number in ConveyorWare System
whs is the parameter value (it is a number between 1 and 999)

Because inventory numbers are time sensitive, the file name contains a company number, warehouse numbers, date and time when inventory was requested. For example, an error file name is error_9_1_20150209_141706.csv and a successful file name is inventory_009_001_20150209_141806.csv.

Samples of CSV files can be found in CwApiHTTPClientExample.zip

To request inventory file in XML format, use the following link:
https://fmw.ferbermidwest.com/cwapi/inventoryFile?inventory=xml

Where:

Https is a secure socket layer protocol
fmw.ferbermidwest.com is the url
cwapi/inventoryFile is the api path
inventory is the parameter
xml is the parameter value

The Parameter value (XML) is a string that presents an XML file.

Sample request of the inventory XML file:
<?xml version="1.0" encoding="UTF-8"?>
<InventoryFileRequest>
<UserId>USERID</UserId>
<CompanyNumber>9</CompanyNumber>
<WareHouseNumber>ALL</WareHouseNumber>
</InventoryFileRequest>

XML string can be without any line breaks. The Server will send response in XML format after request is received.

Incoming request will be validated against the following schema:
https://fmw.ferbermidwest.com/cwapi/InventoryFileRequest.xsd.

It is possible to retrieve partial number of records in inventory file. Every stock item has status (Active or Inactive), so you can retrieve records only with one status.

Sample request of the inventory file only with items that have one status:

```
<?xml version="1.0" encoding="UTF-8"?>
<InventoryFileRequest>
<UserId>USERID</UserId>
<CompanyNumber>9</CompanyNumber>
<WareHouseNumber>1</WareHouseNumber>
<ItemStatus>ACTIVE</ItemStatus>
</InventoryFileRequest>
```

ItemStatus field can only have the following values "ALL", "ACTIVE", "INACTIVE".
Also, you can request particular inventory items but no more than 100. In this case the ItemStatus field will be ignored.

Sample of requesting particular item numbers:

```
<?xml version="1.0" encoding="UTF-8"?>
<InventoryFileRequest>
<UserId>USERID</UserId>
<CompanyNumber>9</CompanyNumber>
<WareHouseNumber>1</WareHouseNumber>
<ItemNumber>A</ItemNumber>
<ItemNumber>ABC</ItemNumber>
</InventoryFileRequest>
```

Sample of a response if something went wrong:

```
<InventoryFileRespond>
<XMLFileError>
<MessageId>0016</MessageId>
<MessageText>Company 987 does not exist</MessageText>
<MessageId>0019</MessageId>
<MessageText>Warehouse 1 is not assign to company 987</MessageText>
</XMLFileError>
</InventoryFileRespond>
```

Response with successful validation of inventory file request:

```
<InventoryFile>
<CompanyNumber>009</CompanyNumber>
<WarehouseyNumber>001</WarehouseyNumber>
<RespondDate>20150211</RespondDate>
<RespondTime>124903</RespondTime>
<InventoryRecord>
```

```xml
<StockNumber>A</StockNumber>
<ItemDescription>TEST AAA</ItemDescription>
<OnHand>1160</OnHand>
<Allocated>0</Allocated>
<Available>1160</Available>
<Short>0</Short>
<BackOrder>0</BackOrder>
<CrdHold>3</CrdHold>
<FutureShip>0</FutureShip>
<Tentative>0</Tentative>
<Shipped>0</Shipped>
<ReorderPoint>30</ReorderPoint>
<MfgrAbrv>A</MfgrAbrv>
<UPC>N1234</UPC>
<Price>500.00</Price>
<Cost>50.00</Cost>
<Weight>222.0000</Weight>
<Status/>
<Kit/>
</InventoryRecord>
<InventoryRecord>
<StockNumber>ABC</StockNumber>
<ItemDescription>TEST ITEM</ItemDescription>
<OnHand>10</OnHand>
<Allocated>0</Allocated>
<Available>10</Available>
<Short>0</Short>
<BackOrder>10</BackOrder>
<CrdHold>20</CrdHold>
<FutureShip>0</FutureShip>
<Tentative>0</Tentative>
<Shipped>0</Shipped>
<ReorderPoint>0</ReorderPoint>
<MfgrAbrv>A</MfgrAbrv>
<UPC>ABC</UPC>
<Price>20.00</Price>
<Cost>10.00</Cost>
<Weight>1.0000</Weight>
<Status/>
<Kit/>
</InventoryRecord>
</InventoryFile>
```

Request a shipping feedback

To request a Shipping Feedback Request XML file, use the following link:
https://fmw.ferbermidwest.com/cwapi/shippingfeedback?shippingfeedback=xml

Where:

Https is a secure socket layer protocol
fmw.ferbermidwest.com is the url
cwapi/shippingfeedback is the api path
shippingfeedback is the parameter
xml is the parameter value

The Parameter value (XML) is a string that presents an XML file.
Sample request of the shippingfeedback XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<ShippingFeedBackRequest>
 <UserId>USERID</UserId>
 <CompanyNumber>9</CompanyNumber>
 <DateFrom>2015-02-15</DateFrom>
 </ShippingFeedBackRequest>
```

Or with all possible options:

```
<?xml version="1.0" encoding="UTF-8"?>
<ShippingFeedBackRequest>
 <UserId>USERID</UserId>
 <CompanyNumber>9</CompanyNumber>
 <CustomerNumber>99</CustomerNumber>
 <DateFrom>2015-02-15</DateFrom>
 <DateTo>2015-02-18</DateTo>
 <OmitItemSerialNumber>Yes</OmitItemSerialNumber>
</ShippingFeedBackRequest>
```

Or with requesting shipping feedbacks by particular order numbers. Number of requested shipping records cannot be more than 100. If request was created by order numbers than DateFrom and DateTo are ignored.

```
<?xml version="1.0" encoding="UTF-8"?>
<ShippingFeedBackRequest>
 <UserId>USERID</UserId>
 <CompanyNumber>9</CompanyNumber>
 <OrderNumber>AAAAAA</OrderNumber>
<OrderNumber>BBBBBBB</OrderNumber>
…. Not more than 100
 </ShippingFeedBackRequest>
```

If XML request by only one order number:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ShippingFeedBackRequest>
 <UserId>USERID</UserId>
 <CompanyNumber>9</CompanyNumber>
 <OrderNumber>AAAAAA</OrderNumber>
</ShippingFeedBackRequest>
```

Incoming request will be validated against the following schema:
https://fmw.ferbermidwest.com/cwapi/ShippingFeedBackRequest.xsd.

Fields CustomerNumber, DateTo, and OmitItemSerialNumber are optional. If CustomerNumber is not present in request then response will have shipping information for all existing customers. If DateTo is not present in request then response will have only one day (ShipFrom) shipping information. DateFrom must be before DateTo, otherwise dates will be reversed. If DateFrom is submitted without DateTo, all of the records will be in the response for one day shipping information. If both date fields are present, a maximum of 5,000 records will be returned in the response. If OmitItemSerialNumber is not present in request, then response will not include item serial numbers.

Error checking is done in the same way as in other api options. UserId, CompanyNumber, and CustomerNumber must be valid in ConweyorWare System.

XML Response sample:

```xml
<ShippingFeedBackFile>
<CurrentLibrary>D000000001</CurrentLibrary>
<ShippingRecord>
<CompanyNumber>9</CompanyNumber>
<CustomerNumber>99</CustomerNumber>
<CustomerName>TEST CUSTOMER</CustomerName>
<ShipDate>20140219</ShipDate>
<OrderNumber>221144</OrderNumber>
<CustomerOrderNumber>223344</CustomerOrderNumber>
<ShippingCompanyName/>
<ShippinAttentionName>VERONICA CISOWSKI</ShippinAttentionName>
<InvoiceTotal>19.90</InvoiceTotal>
<Carrier>USPS</Carrier>
<ShipService>FIRST</ShipService>
<ThirdPartyAccountNumber/>
<WarehouseNumber>1</WarehouseNumber>
<UOM>LB</UOM>
<Currency>USD</Currency>
<NumberOfPackages>1</NumberOfPackages>
<TotalWeight>5.0</TotalWeight>
<TotalFreight>2.09</TotalFreight>
<AllShippedItems>
```

```xml
<Item>
<ItemNumber>FREIGHT</ItemNumber>
<ItemDescription>FREIGHT CHARGES</ItemDescription>
<OrderQty>1</OrderQty>
<BackOrderQty>0</BackOrderQty>
<ShipQty>1</ShipQty>
<ItemPrice>3.95</ItemPrice>
</Item>
<Item>
<ItemNumber>1364</ItemNumber>
<ItemDescription>EA TOO MUCH FUN TOO DEMA</ItemDescription>
<OrderQty>1</OrderQty>
<BackOrderQty>0</BackOrderQty>
<ShipQty>1</ShipQty>
<ItemPrice>15.95</ItemPrice>
</Item>
</AllShippedItems>
<Package>
<TrackingNumber>9400110200830159994669</TrackingNumber>
<Weight>5.0</Weight>
<Freight>2.09</Freight>
<PackageNumber>1</PackageNumber>
<PackageItems>
<Item>
<ItemNumber>1364</ItemNumber>
<ItemDescription>EA TOO MUCH FUN TOO DEMA</ItemDescription>
<PackageItemQty>1</PackageItemQty>
<PackageItemPrice>15.95</PackageItemPrice>
</Item>
</PackageItems>
</Package>
</ShippingRecord>
<ShippingRecord>
<CompanyNumber>16</CompanyNumber>
<CustomerNumber>99</CustomerNumber>
<CustomerName>TEST CUSTOMER</CustomerName>
<ShipDate>20140219</ShipDate>
<OrderNumber>221148</OrderNumber>
<CustomerOrderNumber/>
<ShippingCompanyName/>
<ShippinAttentionName>MARIAN CHAPMAN</ShippinAttentionName>
<InvoiceTotal>13.90</InvoiceTotal>
<Carrier>USPS</Carrier>
<ShipService>FIRST</ShipService>
<ThirdPartyAccountNumber/>
<WarehouseNumber>1</WarehouseNumber>
<UOM>LB</UOM>
<Currency>USD</Currency>
<NumberOfPackages>1</NumberOfPackages>
<TotalWeight>3.0</TotalWeight>
```

```xml
<TotalFreight>1.93</TotalFreight>
<AllShippedItems>
<Item>
<ItemNumber>FREIGHT</ItemNumber>
<ItemDescription>FREIGHT CHARGES</ItemDescription>
<OrderQty>1</OrderQty>
<BackOrderQty>0</BackOrderQty>
<ShipQty>1</ShipQty>
<ItemPrice>3.95</ItemPrice>
</Item>
<Item>
<ItemNumber>1106</ItemNumber>
<ItemDescription>MOON: CD I LOVE MUSIC CD</ItemDescription>
<OrderQty>1</OrderQty>
<BackOrderQty>0</BackOrderQty>
<ShipQty>1</ShipQty>
<ItemPrice>9.95</ItemPrice>
</Item>
</AllShippedItems>
<Package>
<TrackingNumber>9400110200829089454916</TrackingNumber>
<Weight>3.0</Weight>
<Freight>1.93</Freight>
<PackageNumber>1</PackageNumber>
<PackageItems>
<Item>
<ItemNumber>1106</ItemNumber>
<ItemDescription>MOON: CD I LOVE MUSIC CD</ItemDescription>
<PackageItemQty>1</PackageItemQty>
<PackageItemPrice>9.95</PackageItemPrice>
</Item>
</PackageItems>
</Package>
</ShippingRecord>
</ShippingFeedBackFile>
```

Submit new customer accounts to the system

To submit new customer accounts you need to have an http client that will send a request in the following format:
https://fmw.ferbermidwest.com/cwapi/customerSubmit?customerFile=xml

Where:

Https is a secure socket layer protocol
fmw.ferbermidwest.com is the url
cwapi/customerSubmit is the api path
customerFile is the parameter
xml is the parameter value

The Parameter value (XML) is a string that represents the xml file.

Sample request of the submitted customer XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<CustomerSubmitRequest>
<UserId>userid</UserId>
<CustomerFile>
<CompanyNumber>9</CompanyNumber>
   <CustomerRecord>
    <BillingAddress>
      <BillToFullName>John Smith</BillToFullName>
      <BillToCompany>XYZ Company</BillToCompany>
      <BillToStreetAddress1>1 Street</BillToStreetAddress1>
      <BillToCity>City</BillToCity>
      <BillToState>MI</BillToState>
      <BillToZipCode>48484</BillToZipCode>
      <BillToCountry>US</BillToCountry>
    </BillingAddress>
   </CustomerRecord>
   <CustomerRecord>
    <BillingAddress>
      <BillToFullName>John Black</BillToFullName>
      <BillToCompany>XYZ Company</BillToCompany>
      <BillToStreetAddress1>1 Street</BillToStreetAddress1>
      <BillToCity>City</BillToCity>
      <BillToState>MI</BillToState>
      <BillToZipCode>48484</BillToZipCode>
      <BillToCountry>US</BillToCountry>
    </BillingAddress>
   </CustomerRecord>
   <CustomerRecord>
    <BillingAddress>
      <BillToFullName>John White</BillToFullName>
      <BillToCompany>XYZ Company</BillToCompany>
      <BillToStreetAddress1>1 Street</BillToStreetAddress1>
```

```
        <BillToCity>City</BillToCity>
        <BillToState>MI</BillToState>
        <BillToZipCode>48484</BillToZipCode>
        <BillToCountry>US</BillToCountry>
      </BillingAddress>
    </CustomerRecord>
  </CustomerFile>
</CustomerSubmitRequest>
```

The XML string can be without any line breaks. The Server will send respond in XML format after request is received.

Incoming request will be validated against the following schema:
https://fmw.ferbermidwest.com/cwapi/CustomerSubmitRequest.xsd.

The XML schema (CustomerSubmitRequest.xsd) describes all fields for creating new customer account records in the ConveyorWare system. Some of the fields are optional. Sample of the XML file shows only required fields.

To prevent heavy load on the ConveyorWare system you can send no more than 5.5MB of data in one request and no more than 10MB of data during one day. It is approximately 2,500 customer records in one request and about 4,500 customer records during the day. The system will not limit amount of requests per day. For example, if you sent 3 requests with a size of 9MB and the next request with a size of 4MB, both will be accepted (total 13 Mb) but the next request will be rejected even if the request size is very small.
To successfully submit new customer accounts you need to have a valid user id which works in ConveyorWare at this url: https://fmw.ferbermidwest.com/cw. Also it can be helpful but not necessary to have any knowledge about manual customer accounts entry processing in ConveyorWare.

In addition to XML validation, the API does field value validation. The API validates all fields with a fatal result (rejecting submitted customer file) if something is wrong, such as: invalid user id, or duplicate customer number, or wrong state abbreviation,… . To make submitting go smoothly, it's recommended to create a few test customer accounts manually using opt 201 in main menu of ConveyorWare or Order Management Add Customer Account page (https://fmw.ferbermidwest.com/fmw).
Required fields are UserId, CompanyNumber, and any other fields that have requirement minOccurs='1' in the XML schema. If you will omit optional fields in the XML file then system will use default values without any errors. Default values are created in admin ConveyorWare menu options 625 - Initial Setup-Company Definitions or 626 - Change Company Definitions. If the customer number is not in the customer record of XML file then the system will create a customer account with the next available number.
The schema (https://fmw.ferbermidwest.com/cwapi/CustomerSubmitRequest.xsd) has a list and requirements of all fields in the customer file. Additional information about fields can be found in the ConveyorWare field sensitive help.

Sample of server response with failed fields validation:

<CustomerSubmitRespond>
<CustomerSubmitted>
<MessageId>0002</MessageId>
<MessageText>false</MessageText>
</CustomerSubmitted>
<XmlFileError>
<MessageId>0005</MessageId>
<MessageText>XML file successfully validated</MessageText>
</XmlFileError>
<UserIdStatus>
<MessageId>0010</MessageId>
<MessageText>*ENABLED</MessageText>
</UserIdStatus>
<CustomerRecord>
<CompanyNumber>9</CompanyNumber>
<CustomerSequenceInXmlFile>1</CustomerSequenceInXmlFile>
<CustomerNumberError>
<MessageId>0051</MessageId>
<MessageText>Customer Number 122 already exists in customer file (duplicate Customer Number)</MessageText>
</CustomerNumberError>
<BillToStateError>
<MessageId>0054</MessageId>
<MessageText>St is invalid Bill To State abbreviation for country US</MessageText>
</BillToStateError>
<ShipToStateError>
<MessageId>0058</MessageId>
<MessageText>ST is invalid Ship To State abbreviation for Canadian province </MessageText>
</ShipToStateError>
</CustomerRecord>
</CustomerSubmitRespond>

Response with successful validation:

<CustomerSubmitRespond>
<CustomerSubmitted>
<MessageId>0001</MessageId>
<MessageText>true</MessageText>
</CustomerSubmitted>
</CustomerSubmitRespond>

File CwApiHTTPClientExample.zip
(https://fmw.ferbermidwest.com/cwapi/CwApiHTTPClientExample.zip)
includes examples of http clients, needed jar files, schemas (OrderSubmitRequest.xsd,
InventorySubmitRequest.xsd, CustomerSubmitRequest.xsd, InventoryFileRequest.xsd,
ShippingFeedBackRequest.xsd), error_messages.properties file, samples of CSV and XML files
with all required and optional fields, and PDF file with this user guide
(ConveyorWareAPIUserGuide.pdf)

All described downloads can be found at https://fmw.ferbermidwest.com/cwapi/.

Technical support is available by phone or email.

Phone: 248 650 2870

Email: info@conveyorware.com