



eCompli submitOrder Web Service

Version 3.0

Document Revised August 24, 2009

The **submitOrder** web service is a function accessed through a Simple Object Access Protocol (SOAP) interface that sends online orders to eCompli and returns various order compliance parameters. This allows third-party Point of Sale (PoS) and order management systems to test orders for compliance in real time and generate accurate shipment reports using eCompli.

The eCompli **submitOrder** web service provides the following functionality:

- Supports orders for wine, beer, spirits, food and non-food items
- Supports direct-to-consumer, wholesale and self-distribution shipments
- Validates orders for beverage and shipment compliance in real time
- Supplies sales tax rates for each invoice line item, when applicable
- Allows invoices to be added to, modified and deleted from eCompli

The function and data structures used to submit order data and test order compliance are defined by a Web Services Description Language (WSDL) file located at:

https://www.compli-beverage.com/services/order_server_v3.php?wsdl.

The function looks like this:

```
result = submitOrder(orderLine)
```

1.0 Overview

An order typically contains a header (order number, order date, customer, destination, etc.) and one or more line items describing the product, quantity and price. When submitting order data using the above function, the header information and the data for a single line item are submitted in an **orderLine** data structure. When an order contains more than one line item, the **submitOrder** function is called for each line item, repeating the header information while incrementing the **orderLineNumber** element. The **orderLinesTotal** element specifies the total lines in the order.

Some **orderLine** elements contain data that is not part of an order. This data includes additional information about the product, such as the product class, as well as invoice and shipping information.

The **orderLine** structure contains two access elements and two control elements. The access elements are **accountNumber** and **accountKey**. These must be obtained from a Compli representative and included in each instance of **orderLine**. The access elements identify an eCompli enterprise and client account.

The **orderLine** structure contains a control element called **command**. This parameter specifies the action to be taken by eCompli. The commands are VALIDATE, ADD, MODIFY and DELETE. These commands are described in Section 2..

The **orderLine** structure also contains a control element called **mode**. The parameters for this element are TEST and EXECUTE. When the mode is set to EXECUTE, the **command** is performed and the order is updated in eCompli. If the parameter is TEST, the **result** structure returns the same status indicators as if the EXECUTE mode had been used but without changing the stored information in eCompli. If no **mode** parameter is provided, TEST is assumed. The **mode** element is ignored when the command is VALIDATE.

Once an **invoiceNumber** and **invoiceDate** are added to the order, the invoice information contained in the order is sent to the eCompli reporting engine. The reporting engine supports three types of shipments: direct shipments to consumers, wholesale shipments to distributors and self-distribution shipments to retailers. The type of shipment is indicated in the **shipType** element of **orderLine**. This element must always be included in every instance of **orderLine**.

All **orderLine** elements are listed in Table 1, below.

All **result** elements are listed in Table 2, below.

Table 1: orderLine Elements

Element	Type	Size	Notes
accountNumber	String	20	Alphanumeric
accountKey	String	20	Alphanumeric
command	String	20	VALIDATE, ADD, MODIFY, DELETE
mode	String	20	TEST, EXECUTE
orderNumber	String	100	Alphanumeric
orderLineNumber	Integer		Order item line number, begins with 1
orderLinesTotal	Integer		Total number of line items in this orderNumber
orderDate	String	10	YYYY-MM-DD
coName	String	60	Company name of distributor or retailer (shipType = W or S)
customerNameFirst	String	60	Customer first name (shipType = D) Alpha characters plus hyphen and apostrophe
customerNameLast	String	60	Customer first name (shipType = D) Alpha characters plus hyphen and apostrophe
customerAddress	String	60	Street address of consumer, distributor or retailer
customerCity	String	60	City of consumer, distributor or retailer
customerState	String	2	State of consumer, distributor or retailer (2 character abbreviation, ex: CA)
customerZip	String	5	Zip code of consumer, distributor or retailer (First 5 digits only)
customerBirthdate	String	10	YYYY-MM-DD (Not required for distributors or retailers)
buyerNameFirst	String	60	Alpha characters plus hyphen and apostrophe (shipType = D)
buyerNameLast	String	60	Alpha characters plus hyphen and apostrophe (shipType = D)
buyerAddress	String	60	Street address of buyerv (shipType = D)
buyerCity	String	60	City of buyer (shipType = D)
buyerState	String	2	State of buyer (2 character abbreviation, ex: CA) (shipType = D)
buyerZip	String	5	Zip code of buyer (First 5 digits only) (shipType = D)
buyerBirthdate	String	10	YYYY-MM-DD (shipType = D)
productSKU	String	30	Alphanumeric
productBrand	String	30	Alphanumeric
productName	String	40	Alphanumeric
productAppellation	String	40	Alphanumeric
productClass	String	7	WINE, BEER, SPIRITS, SPARK, CIDER
productVintage	String	4	YYYY or NV for none
productAlcohol	Float		Ex: 14.1 (no percent sign)
productsSample	Boolean		True = 1, False = 0 (productClass = WINE, SPARK, BEER, SPIRITS)
productsForeign	Boolean		True = 1, False = 0 (productClass = WINE, SPARK, BEER, SPIRITS)
bottleSize	Float		Numeric volume of bottle, ex: 1500 or 1.5 (productClass = WINE, SPARK, BEER, SPIRITS)
bottleSizeUnit	String	2	OZ=Ounce, G= Gallon, ML=Milliliters, L=Liters (productClass = WINE, SPARK, BEER, SPIRITS)
containerCount	Integer		Total shipping containers used for this invoice line
containerPack	Integer		Total items in shipping container
beveragePrice	Float		Price per item in USD (productClass = WINE, SPARK, BEER, SPIRITS)
foodPrice	Float		Price per food item (productClass = FOOD)
otherPrice	Float		Price per non-beverage, non-food item (productClass = OTHER)
salesTax	Float		Sales tax paid for this line item in USD
invoiceNumber	String	30	Invoice number assigned to this order at fulfillment
invoiceDate	String	10	YYYY-MM-DD
shipType	String	1	W=Wholesale, D=Direct, S=Self-Distribution
shipOnSite	Boolean		True = 1, False = 0 (productClass = WINE, SPARK, BEER, SPIRITS)
shipDate	String	10	YYYY-MM-DD
shipPrice	Float		Price to ship this line item in USD (no dollar sign)
shipCarrierCode	String	40	Ex: UPS, DHL, FXO, FEX
shipTrackingNumber	String	40	Alphanumeric

Table 2: result Elements		
Element	Type	Notes
accountNumber	String	Alphanumeric
orderNumber	String	Alphanumeric
orderLineNumber	Integer	
orderLinesTotal	Integer	
resultCode	Integer	See Table 7, page 21
resultMessage	String	
salesTaxRate	Float	ex: 0.06 = 6%
stateLiters	Float	Remaining liters winey can ship below limit
customerLiters	Float	Remaining liters consumer can purchase below limit
validState	Integer	See Table 8, page 23
validZip	Integer	See Table 8
validLicense	Integer	See Table 8
validBrand	Integer	See Table 8
validStateLimit	Integer	See Table 8
validCustomerLimit	Integer	See Table 8
validCustomerAge	Integer	See Table 8
validBuyerAge	Integer	See Table 8
orderCompliance	Integer	See Table 8

2.0 orderLine Commands

VALIDATE Command

The VALIDATE command is used to test and retrieve order compliance, sales tax rates and age verification for a specified order line item without modifying the contents of eCompli. When the VALIDATE command is sent before adding an order line to eCompli, the parameters included in the **orderLine** structure are tested. Any particular compliance test can be conducted by providing only those parameters required to support a desired test. For example, including only the **customerState** and **customerZip** parameters will return results for **validState**, **validZip** and **validLicense**. When an order contains more than one line item, as indicated by the **orderLinesTotal** parameter, the data for each line is buffered and used in subsequent compliance tests for that **orderNumber**. The buffer is emptied when the last order line item is received.

When the VALIDATE command is sent after adding an order line to eCompli, the stored data is tested. In this case, only the parameters required to identify the order line item must be included in the **orderLine** structure. See Table 3.

*Note: when a third-party is purchasing product as a gift for the customer, the buyerXXX parameters must be included to return a valid result for the buyer's Age Verification test.

IMPORTANT: When validating order with multiple line items prior to adding them to the reporting engine, all lines of the order must be validated before adding any of them to ensure accurate results for the **stateLiters** and **customerLiters** parameters in the **result** structure.

Table 3: Required Parameters for Full Validation	
VALIDATE before ADD*	VALIDATE after ADD
accountNumber	accountNumber
accountKey	accountKey
command	command
orderNumber	orderNumber
orderLineNumber	orderLineNumber
orderLinesTotal	
orderDate	
customerNameFirst	
customerNameLast	
customerAddress	
customerCity	
customerState	
customerZip	
productClass	
containerCount	
containerPack	
bottleSize	
bottleSizeUnit	
shipType	
shipOnSite	

ADD Command

The ADD command inserts the parameters provided by **orderLine** into eCompli when the **mode** parameter is EXECUTE. The minimum parameters of the ADD command when the **productClass** is a beverage are shown in Table 3.

When the ADD command is first received by eCompli, it examines the **productSKU** parameter to determine if the product has been stored in the eCompli product database for the specified **accountNumber**. If a product “lookup” fails, then the **productXXX** parameters in the **orderLine** structure are examined. If any required product parameters are missing, a failure is indicated in the **result** structure.

The minimum **orderLine** elements for both conditions are shown in Table 4.

The **orderLine** elements supplied with the ADD command will generally reflect the available order information at the time of purchase. This means that parameters such as **invoiceNumber**, **invoiceDate**, **shipDate**, **shipPrice**, **shipCarrierCode** and **shipTrackingNumber** will not be included. These parameters would typically be provided later using the MODIFY command. However, all elements shown in Table 1 may be included in the ADD command if available.

The **result** structure returned for the ADD command contains valid indicators for all order compliance tests, sales tax rates and age verification.

Note: Products are not imported into the eCompli product database using the **submitOrder** web services interface. Products must be entered into eCompli manually, or by uploading a product list in spreadsheet (CSV) format, or by using the **submitProduct** web services interface.

*IMPORTANT: Some states require that beverage products or brands shipped directly to consumers be registered. After products are imported into the eCompli product database, a registration request must be submitted for any products that will ship to these states. The **result** structure will return 0 (Validation Fail) in the **validBrand** element for any unregistered products purchased from these states. If a product has not been imported and a lookup fails, the **result** structure returns 2 (Validation Not Possible) in the **validBrand** element.*

Table 4: Minimum Parameters	
ADD	ADD with Lookup
accountNumber	accountNumber
accountKey	accountKey
command	command
mode	mode
orderNumber	orderNumber
orderLineNumber	orderLineNumber
orderLinesTotal	orderLinesTotal
orderDate	orderDate
customerNameFirst	customerNameFirst
customerNameLast	customerNameLast
customerAddress	customerAddress
customerCity	customerCity
customerState	customerState
customerZip	customerZip
productSKU	productSKU
productBrand	
productName	
productAppellation	
productClass	
productVintage	
productAlcohol	
productIsSample	
productIsForeign	
containerCount	containerCount
containerPack	containerPack
beveragePrice	beveragePrice
salesTax	salesTax
bottleSize	bottleSize
bottleSizeUnit	bottleSizeUnit
shipType	shipType
shipOnSite	shipOnSite

MODIFY Command

The MODIFY command is used to update or add parameters to order line items that have been previously inserted into eCompli using the ADD command. Typical elements that are updated include invoicing and shipping formation at the time of order fulfillment. However, any parameter may be changed using this command with the exception of **orderNumber**, **orderLineNumber** and **orderLinesTotal**. The minimum parameters for the MODIFY command are shown in Table 5.

Table 5: Minimum Parameters
MODIFY
accountNumber
accountKey
command
mode
orderNumber
orderLineNumber
shipType
(Additional parameters as required)

The **result** structure returned for the MODIFY command contains valid indicators for all compliance tests, sales tax rates and age verification, based on all of the parameters of the updated invoice line.

DELETE Command

The DELETE command removes an entire order from eCompli engine, based on the **orderNumber** parameter. All line items are deleted.

To ensure synchronization between the order system using the **submitOrder** web service and eCompli, it is not possible to delete a single invoice line. If one or more invoice lines must be removed from an invoice, the system using the web services interface must delete the entire order from eCompli and reinsert the remaining line items using the ADD command. The **orderLineNumber** and **orderLinesTotal** parameters must be modified prior to insertion to reflect the new number of line items in the order.

Table 6: Minimum Parameters
DELETE
accountNumber
accountKey
command
mode
orderNumber

The minimum parameters for the DELETE command are shown in Table 6.

3.0 orderLine Element Definitions

complexType: **orderLine**

Type: Struct

The **orderLine** data structure contains the elements and parameters passed in the **submitOrder()** function.

Element Name: **accountNumber**

Type: String

Required

This element contains an alpha-numeric identifier that corresponds to a an eCompli client enterprise and account. This number is issued by Compli upon request.

Element Name: **accountKey**

Type: String

Required

This element contains an alpha-numeric password that corresponds to a an eCompli client **accountNumber**. The **accountKey** is issued by Compli with the **accountNumber**.

Element Name: **command**

Type: String

Required

Defined Parameters: VALIDATE, ADD, MODIFY, DELETE

VALIDATE: Performs order compliance tests on line items without modifying the content of the eCompli database. This command also returns the applicable sales tax and age verification, if subscribed. See Example 1 at the end of this document.

ADD: Inserts an order line item into the eCompli database, returns the results of the order compliance tests, applicable sales tax and age verification, if subscribed. See Example 2 at the end of this document.

MODIFY: Updates one or more data items of an order line item previously added to the eCompli database. Only required parameters and parameters to be modified must be specified in **orderLine**. See Example 3 at the end of this document.

DELETE: Removes all order line items associated with **orderNumber** from the eCompli database. See Example 4 at the end of this document.

Element Name: **mode**

Type: String

Conditional

Defined Parameters: TEST, EXECUTE

TEST: Use this parameter during setup and testing of the SOAP client. All return parameters reflect the results of the **command** performed without modifying the content of the eCompli database.

EXECUTE: This parameter must be indicated for the **command** to modify the eCompli database.

Note: The **mode** is ignored when the **command** is VALIDATE.

Element Name: **orderNumber**

Type: String

Required

This element contains the unique alpha-numeric identifier for the order.

Element Name: **orderLineNumber**

Type: Integer

Required

Orders that contain multiple line items are submitted one line item at a time. This parameter identifies the order line number, beginning with 1.

Element Name: **orderLinesTotal**

Type: Integer

VALIDATE: Required, ADD: Required, MODIFY: Required, DELETE: Not Used

This parameter indicates the total number of line items in the order. When present using the VALIDATE command, the order line items are buffered. When each line item is received, the validation tests for state aggregate and consumer shipping limits are calculated using the accumulated line items.

Element Name: **orderDate**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

Contains the order date (date of purchase) in the following format: YYYY-MM-DD (Example: 2008-08-01 is August 1st, 2008).

Element Name: **coName**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the full name of the recipient of the shipment. When the **shipType** parameter is W (Wholesale), this element contains the name of the distributor. When the **shipType** parameter is S (Self-Distribution), this element contains the name of the retailer. When the **shipType** is D (Direct), this element is ignored.

Element Name: **customerNameFirst**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the first name of the consumer who is receiving the shipment. This parameter is required only when the **shipType** parameter is D (Direct). When the **shipType** is W (Wholesale) or S (Self-Distribution), this element is ignored.

Element Name: **customerLastName**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the last name of the consumer who is receiving the shipment. This parameter is required only when the **shipType** parameter is D (Direct). When the **shipType** is W (Wholesale) or S (Self-Distribution), this element is ignored.

Element Name: **customerAddress**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the street address of the recipient of the shipment.

Element Name: **customerCity**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the full city name of the recipient of the shipment. The name must be in a format recognized by the U.S. Postal Service. See <http://zip4.usps.com/zip4/citytown.jsp>.

Element Name: **customerState**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the 2 letter state abbreviation of the recipient of the shipment. The state abbreviation must be recognized by the U.S. Postal Service.

Element Name: **customerZip**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the 5 digit zip code of the recipient of the shipment. The zip code must be recognized by the U.S. Postal Service.

Element Name: **customerBirthdate**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains the customer's birth date in the following format: YYYY-MM-DD. The **customerBirthdate** element is expected only when the **shipType** parameter is D (Direct).

Element Name: **buyerNameFirst**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

This element contains the first name of the person who is making the purchase when different than the consumer, e.g. the purchase is a gift. The **buyerNameFirst** element is examined only when the **shipType** parameter is D (Direct-to-Consumer).

Element Name: **buyerNameLast**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

This element contains the last name of the person who is making the purchase when different than the consumer, e.g. the purchase is a gift. The **buyerNameLast** element is examined only when the **shipType** parameter is D (Direct).

Element Name: **buyerAddress**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

This element contains the street address of the purchaser. The **buyerAddress** element is examined only when the **shipType** parameter is D (Direct).

Element Name: **buyerCity**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

This element contains the full city name of the purchaser. The name should be in a format recognized by the U.S. Postal Service. See <http://zip4.usps.com/zip4/citytown.jsp>. The **buyerCity** element is examined only when the **shipType** parameter is D (Direct).

Element Name: **buyerState**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

This element contains the 2 letter state abbreviation of the purchaser. The state abbreviation must be recognized by the U.S. Postal Service. The **buyerState** element is examined only when the **shipType** parameter is D (Direct).

Element Name: **buyerZip**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

This element contains the 5 digit zip code of the purchaser. The zip code must be recognized by the U.S. Postal Service. The **buyerZip** element is examined only when the **shipType** parameter is D (Direct).

Element Name: **buyerBirthdate**

Type: String

VALIDATE: Conditional, ADD: Conditional, MODIFY: Conditional, DELETE: Not Used

Contains the purchaser's birth date in the following format: YYYY-MM-DD. The **buyerBirthdate** element is expected only when a purchaser different than the recipient is included and the **shipType** parameter is D (Direct).

Element Name: **productSKU**

Type: String

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element contains the manufacturer's product code, inventory code or Stock-Keeping Unit (SKU) number. This parameter is used to look up product information stored in the eCompli database.

Element Name: **productBrand**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains the brand name of the product. It is not required when the **productClass** is FOOD or OTHER. When **productClass** is a beverage, this element may be required if the **productSKU** does not match a product in the eCompli database.

Element Name: **productName**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains the product name or other appropriate descriptive identification.

When productClass indicates a beverage, this element may not be required when the **productSKU** parameter is included. If the eCompli reporting engine is unable to find a product match, then the **productName** parameter is required.

This element is required when the **productClass** is FOOD or OTHER.

Element Name: **productAppellation**

Type: String

VALIDATE: Optional, ADD: Optional, MODIFY: Optional, DELETE: Not Used

This element contains the product appellation, or geographical region where the product is produced. This parameter is optional, but may be required for some states' reports.

Element Name: **productClass**

Type: String

VALIDATE: Required, ADD: Required, MODIFY: Optional, DELETE: Not Used

Defined Parameters: WINE, SPARK, CIDER, BEER, SPIRITS, FOOD, OTHER

This element contains the product type or class. The defined parameters below are currently recognized by eCompli.

WINE: All vinous beverages except sparkling wines

SPARK: Sparkling wines

CIDER: Alcoholic beverages fermented from fruit other than grapes

BEER: All beer and malt beverages

SPIRITS: Distilled alcoholic beverages

FOOD: Any item for human consumption except alcoholic beverages

OTHER: Any non-food or non-beverage item

Element Name: **productVintage**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains the 4 digit vintage year of a beverage product. If no vintage applies, the element can remain empty or the parameter "NV" can be used. This element may not be required when the **productSKU** parameter is included. If eCompli is unable to find a product match, then the **productVintage** parameter is required. This element is ignored when the **productClass** is FOOD or OTHER.

Element Name: **productAlcohol**

Type: Float

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains a decimal number indicating a beverage product's percent of alcohol (Example: 14.5). This element may not be required when the **productSKU** parameter is included. If eCompli is unable to find a product match, then the **productAlcohol** parameter is required. This element is ignored when the **productClass** is FOOD or OTHER.

Note: Do not include a percent sign.

Element Name: **productIsSample**

Type: Boolean

VALIDATE: Optional, ADD: Optional, MODIFY: Optional, DELETE: Not Used

Defined Parameters: false (default), true

If the product is delivered as a sample, this element must contain "1" for "Yes", otherwise "0" for "No". If this element is null, "0" is assumed.

Element Name: **productIsForeign**

Type: Boolean

VALIDATE: Optional, ADD: Optional, MODIFY: Optional, DELETE: Not Used

Defined Parameters: false (default), true

If the product is manufactured outside the United States, this element must contain "1" for "Yes", otherwise "0" for "No". If this element is null, "0" is assumed.

Element Name: **containerCount**

Type: Integer

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element indicates the number of containers shipped for this line item.

Element Name: **containerPack**

Type: Integer

VALIDATE: Optional, ADD: Required, MODIFY: Optional, DELETE: Not Used

This element indicates the product count in a shipping container.

Element Name: **beveragePrice**

Type: Float

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element is required when the **productClass** indicates a beverage. This element indicates the price charged for an individual item in U.S. dollars (Example: 49.95). The value should include the price paid ONLY for the beverage item. It should NOT include any taxes or shipping charged to the recipient. Do not include a dollar sign.

Note: When the productClass indicate beverage, the **beveragePrice** must include a value greater than zero when the **shipType** parameter is "D" (direct to consumer shipments). For wholesale shipments, this parameter may be zero except for shipments to Washington state.

Element Name: **salesTax**

Type: Float

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element indicates the sales tax paid by the consumer for this line item in U.S. dollars (Example: 3.59). Do not include a dollar sign. This element is only required when **shipType** = D (Direct).

Element Name: **foodPrice**

Type: Float

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element indicates the amount paid per item when the productClass is FOOD. This element is required by some states when **shipType** = D (Direct).

Element Name: **otherPrice**

Type: Float

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element indicates the amount paid per item when the productClass is OTHER. This element is required by some states when **shipType** = D (Direct).

Element Name: **bottleSize**

Type: Float

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element indicates the volume of an individual bottle in the units indicated by **bottleSizeUnit**. Volumes of 1000 milliliters or higher must be expressed as liters (ex: 1.5L = 1500ML). This element may not be required when the **productSKU** parameter is included. If eCompli is unable to find a product match, then the **bottleSize** parameter is required. This element is ignored when the **productClass** is FOOD or OTHER.

Element Name: **bottleSizeUnit**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

Defined Parameters: OZ, G, ML, L

This element indicates the unit of measure for the bottle volume amount indicated by **bottleSize** (Example: ML).

OZ: Ounces

G: Gallons

ML: Milliliters

L: Liters

This element may not be required when the **productSKU** parameter is included. If eCompli is unable to find a product match, then the **bottleSizeUnit** parameter is required. This element is ignored when the **productClass** is FOOD or OTHER.

Element Name: **invoiceNumber**

Type: String

VALIDATE: Optional, ADD: Optional, MODIFY: Optional, DELETE: Not Used

This element contains the invoice number assigned to this order when the order is fulfilled. When the **invoiceNumber** element is included, the **invoiceDate** element is required. Once the invoice number and date have been added to the order line item, the item is submitted to the eCompli reporting engine.

Element Name: **invoiceDate**

Type: String

VALIDATE: Optional, ADD: Optional, MODIFY: Optional, DELETE: Not Used

This element contains the invoice date assigned to this order when the order is fulfilled. When the **invoiceDate** element is included, the **invoiceNumber** element is required. Once the invoice number and date have been added to the order line item, the item is submitted to the eCompli reporting engine.

Element Name: **shipType**

Type: String

VALIDATE: Required, ADD: Required, MODIFY: Required, DELETE: Not Used

Defined Parameters: D, W, S

This element indicates the type of recipient to whom product is shipped.

D: Direct shipment to a consumer

W: Wholesale shipment to a distributor

S: Self-Distribution shipment to a retailer

Element Name: **shipOnSite**

Type: Boolean

VALIDATE: Conditional, ADD: Required, MODIFY: Optional, DELETE: Not Used

Defined Parameters: false (default), true

The **shipOnSite** element indicates if the order was received at the place of business (i.e. a winery or tasting room) . To indicate an On Site order, this element should be set to 1. If the order was not received on site, this element should be set to 0. This element is required for accurate order compliance testing when the **shipType** is D (Direct). This element is ignored when the **productClass** is FOOD or OTHER.

Element Name: **shipDate**

Type: String

VALIDATE: Optional, ADD: Optional, MODIFY: Optional, DELETE: Not Used

Contains the shipment date in the following format: YYYY-MM-DD.

Element Name: **shipPrice**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains the total shipping cost for this line item in US dollars (ex: 3.45). Do not include a dollar sign. This element should be included, but is only required by some states and is ignored by the eCompli reporting engine if 0.

Element Name: **shipCarrierCode**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

Defined Parameters: DHL, UPS, FEX (FedEx), FXO (FedEx Air)

This element contains a code indicating the authorized shipper who delivers the product. This element is only required by some states and is ignored by the eCompli reporting engine if null. For more information, contact a Compli representative.

Element Name: **shipTrackingNumber**

Type: String

VALIDATE: Optional, ADD: Conditional, MODIFY: Optional, DELETE: Not Used

This element contains a code indicating the carrier's tracking number. This parameter is only required by some states and is ignored by the eCompli reporting engine if null. For more information, contact a Compli representative.

4.0 result Element Definitions

complexType: **result**

Type: Struct

The **result** data structure contains the elements and parameters returned to the SOAP client after calling the **submitOrder()** function. See Examples 5 and 6 at the end of this document.

Element Name: **accountNumber**

Type: String

This element contains the account number used in the transaction.

Element Name: **orderNumber**

Type: String

This element contains the order number used in the transaction.

Element Name: **orderLineNumber**

Type: String

This element contains the line item number used in the transaction.

Element Name: **orderLinesTotal**

Type: String

This element contains the total line items indicated in the transaction.

Element Name: **resultCode**

Type: Integer

This element contains a numeric code indicating the result of the transaction. All transactions ending in success are indicated with 1. All other **resultCode** parameters indicate an error as described in Table 7.

Table 7: Result Codes	
CODE	DESCRIPTION
101	Missing or unrecognized accountNumber
102	Missing or unrecognized accountKey
103	Invalid command (Use VALIDATE, ADD, MODIFY or DELETE)
104	Missing parameter (Indicated in resultMessage)
105	Order line not found (MODIFY command only)
106	Incorrect invoiceDate format (Use YYYY-MM-DD)
107	Incorrect customerState format (Use 2 character abbreviation, ex: CA)
108	Unrecognized productClass (Use WINE, SPARK, CIDER, BEER, SPIRITS)
109	Incorrect productVintage format (Use YYYY or NV)
110	Incorrect productAlcohol format (Use NN.N without percent sign)
111	Unrecognized containerType (Use BOTTLE or CASE)
112	Unrecognized bottleSizeUnit (Use OZ, G, ML, L)
113	Unrecognized shipType (Use D, W, S)
114	Web Services Offline (Temporary Service Failure)
115	Duplicate invoice line (Invoice Line is already in reporting engine)
116	No elements to modify (MODIFY command only)
118	Incorrect customerBirthdate format (Use YYYY-MM-DD)
119	Incorrect buyerBirthdate format (Use YYYY-MM-DD)
120	Unsuccessful product lookup
121	Interface Busy (Another user is modifying the selected accountNumber)
122	Incorrect shipDate Format (Use YYYY-MM-DD)
123	Incorrect customerBirthdate Format (Use YYYY-MM-DD)
124	Incorrect buyerBirthdate Format (Use YYYY-MM-DD)
125	Duplicate product found during SKU lookup
126	Incorrect orderDate format (Use YYYY-MM-DD)
127	orderNumber exceeds 100 characters
128	invoiceNumber exceeds 30 characters

Element Name: **resultMessage**

Type: String

This element contains a human-readable text message describing the result of the transaction.

Element Name: **salesTaxRate**

Type: Float

This element provides the applicable sales tax rate expressed as a decimal number (Example: 0.06 is 6%) if the **customerCity** , **customerState** and **customerZip** parameters are supplied by **orderLine** using any command except DELETE. This element returns 0 if no tax applies or -1 if the tax rate could not be retrieved due to missing or unrecognized customer parameters.

Element Name: **stateLiters**

Type: Float

This element returns the amount of product, in liters, that the winery may ship to **customerState** before exceeding the aggregate state limit. If the **validStateLimit** parameter is set to 3 (Not Applicable), the **stateLiters** parameter will be 0 and should be ignored. If stateLiters is a negative number, it is indicating the amount of product in liters that the winery has shipped in excess of the state aggregate limit.

Important: this parameter includes the amount of product invoiced and shipped to date, as well as any product on the current invoice line and any invoice lines previously submitted using the VALIDATE command.

Element Name: **customerLiters**

Type: Float

This element returns the amount of product, in liters, that the winery may ship to **customerName** in **customerState** before exceeding the consumer limit. If the **validConsumerLimit** parameter is set to 3 (Not Applicable), the **consumerLiters** parameter will be 0 and should be ignored.

Note: in some cases, **validCustomerLimit** is calculated based on **customerAddress**, **customerCity** and **customerZip**. As of this writing, states that limit consumer purchases based on the shipping address include Hawaii and Louisiana. Consequently, these parameters should be included when validating a state's consumer limits.

Important: this parameter includes the amount of product invoiced and shipped to date, as well as any product on the current invoice line and any invoice lines previously submitted using the VALIDATE command.

Order Compliance Result Codes

The **result** structure indicates the outcome of several beverage order compliance tests, as described below. The result of each test is indicated by one of the compliance codes in Table 8:

Table 8: Compliance Codes	
CODE	DESCRIPTION
0	Order compliance failed
1	Order compliance passed
2	Unable to perform compliance test (Insufficient data provided)
3	Not applicable (No compliance required for this item)
4	Not subscribed (contact a Compli representative)

Element Name: **validState**

Type: Integer

Indicates if the **customerState** allows shipments of the **shipType** and **productClass**.

Element Name: **validZip**

Type: Integer

Indicates if the **customerZip** allows shipments of the **shipType** and **productClass**.

Element Name: **validLicense**

Type: Integer

Indicates if the company identified by **accountNumber** holds a valid license to ship product to the **customerState** for the **shipType** and **productClass**. In states where a license is not required, this element returns "3" (Not applicable).

Element Name: **validBrand**

Type: Integer

Indicates if the product or brand has been registered in the **customerState** for the given **shipType**. The product must be identified by using the **productSKU** element in the **orderLine** structure for a reliable result. In states where product or brand registration is not required, this element returns "3" (Not applicable).

Element Name: **validStateLimit**

Type: Integer

Indicates if the company identified by **accountNumber** has shipped product to **customerState** in excess of the aggregate limit for the current year. Not all states enforce aggregate limits. In these cases, this element returns "3" (Not applicable).

Element Name: **validCustomerLimit**

Type: Integer

Indicates if the company identified by **accountNumber** has shipped product to **customerName** or, **customerAddress**, **customerCity**, **customerState** and **customerZip** in excess of the consumer limit, based on the month indicated by **orderDate**. Not all states enforce consumer limits. In these cases, this element returns "3" (Not applicable).

Element Name: **validCustomerAge**

Type: Integer

Indicates if **customerName** is 21 years or older. Age verification is performed only for invoices with **shipType** set to 'D' (Direct-to-Consumer). Only the first line of an order (**orderLineNumber** set to 1) is tested. Once the customer's age has been verified, it is stored for future reference to reduce subscription charges. The **validCustomerAge** parameter is set to '3' (not applicable) for orders with the **orderLineNumber** set greater than 1. This test requires a subscription. See a Compli representative for more information.

Element Name: **validBuyerAge**

Type: Integer

Indicates if **buyerName** is 21 years or older. Age verification is performed only for invoices with **shipType** set to 'D' (Direct-to-Consumer). Only the first line of an order (**orderLineNumber** set to 1) is tested. Once the customer's age has been verified, it is stored for future reference to reduce subscription charges. The **validBuyerAge** parameter is set to '3' (not applicable) for orders with the **orderLineNumber** set greater than 1. This test requires a subscription. See a Compli representative for more information.

Element Name: **orderCompliance**

Type: Integer

If any of the order compliance tests above return a "0" (Fail), this element returns "0". Otherwise, this element returns "1" (Pass).

Example 1: orderLine (command = VALIDATE, shipType = D)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:ecompli_webservice_v3">
<SOAP-ENV:Body>
<tns:submitOrder xmlns:tns="urn:ecompli_webservice_v3">
<orderLine xsi:type="tns:orderLine">
<accountNumber xsi:type="xsd:string">ACCT55555555</accountNumber>
<accountKey xsi:type="xsd:string">KEY123456789</accountKey>
<command xsi:type="xsd:string">VALIDATE</command>
<mode xsi:type="xsd:string">EXECUTE</mode>
<orderNumber xsi:type="xsd:string">00001</orderNumber>
<orderLineNumber xsi:type="xsd:integer">1</orderLineNumber>
<orderLinesTotal xsi:type="xsd:integer">3</orderLinesTotal>
<orderDate xsi:type="xsd:string">2008-08-01</orderDate>
<coName xsi:type="xsd:string"></coName>
<customerNameFirst xsi:type="xsd:string">John</customerNameFirst>
<customerNameLast xsi:type="xsd:string">Doe</customerNameLast>
<customerAddress xsi:type="xsd:string">123 Broad Street</customerAddress>
<customerCity xsi:type="xsd:string">Paso Robles</customerCity>
<customerState xsi:type="xsd:string">CA</customerState>
<customerZip xsi:type="xsd:string">93423</customerZip>
<customerBirthdate xsi:type="xsd:string">1960-03-20</customerBirthdate>
<buyerNameFirst xsi:type="xsd:string"></buyerNameFirst>
<buyerNameLast xsi:type="xsd:string"></buyerNameLast>
<buyerAddress xsi:type="xsd:string"></buyerAddress>
<buyerCity xsi:type="xsd:string"></buyerCity>
<buyerState xsi:type="xsd:string"></buyerState>
<buyerZip xsi:type="xsd:string"></buyerZip>
<buyerBirthdate xsi:type="xsd:string"></buyerBirthdate>
<productSKU xsi:type="xsd:string">W-123-01</productSKU>
<productBrand xsi:type="xsd:string">Compli</productBrand>
<productName xsi:type="xsd:string">Old Vines Zin</productName>
<productClass xsi:type="xsd:string">WINE</productClass>
<productVintage xsi:type="xsd:string">2006</productVintage>
<productAlcohol xsi:type="xsd:float">14.5</productAlcohol>
<productIsSample xsi:type="xsd:boolean">0</productIsSample>
<productIsForeign xsi:type="xsd:boolean">0</productIsForeign>
<bottleSize xsi:type="xsd:float">750</bottleSize>
<bottleSizeUnit xsi:type="xsd:string">ML</bottleSizeUnit>
<containerPack xsi:type="xsd:integer">1</containerPack>
<containerCount xsi:type="xsd:integer">3</containerCount>
<beveragePrice xsi:type="xsd:float">32.95</beveragePrice>
<foodPrice xsi:type="xsd:float">0.00</foodPrice>
<otherPrice xsi:type="xsd:float">0.00</otherPrice>
<salesTax xsi:type="xsd:float">1.98</salesTax>
<invoiceNumber xsi:type="xsd:string"></invoiceNumber>
<invoiceDate xsi:type="xsd:string"></invoiceDate>
<shipType xsi:type="xsd:string">D</shipType>
<shipOnSite xsi:type="xsd:boolean">0</shipOnSite>
<shipDate xsi:type="xsd:string"></shipDate>
<shipPrice xsi:type="xsd:string"></shipPrice>
<shipCarrierCode xsi:type="xsd:string"></shipCarrierCode>
<shipTrackingNumber xsi:type="xsd:string"></shipTrackingNumber>
</orderLine>
</tns:submitOrder>

```

Example 2: orderLine (command = ADD, shipType = D)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:ecompli_webservice_v3">
<SOAP-ENV:Body>
<tns:submitOrder xmlns:tns="urn:ecompli_webservice_v3">
<orderLine xsi:type="tns:orderLine">
<accountNumber xsi:type="xsd:string">ACCT55555555</accountNumber>
<accountKey xsi:type="xsd:string">KEY123456789</accountKey>
<command xsi:type="xsd:string">ADD</command>
<mode xsi:type="xsd:string">EXECUTE</mode>
<orderNumber xsi:type="xsd:string">00001</orderNumber>
<orderLineNumber xsi:type="xsd:integer">1</orderLineNumber>
<orderLinesTotal xsi:type="xsd:integer">3</orderLinesTotal>
<orderDate xsi:type="xsd:string">2008-08-01</orderDate>
<coName xsi:type="xsd:string"></coName>
<customerNameFirst xsi:type="xsd:string">John</customerNameFirst>
<customerNameLast xsi:type="xsd:string">Doe</customerNameLast>
<customerAddress xsi:type="xsd:string">123 Broad Street</customerAddress>
<customerCity xsi:type="xsd:string">Paso Robles</customerCity>
<customerState xsi:type="xsd:string">CA</customerState>
<customerZip xsi:type="xsd:string">93423</customerZip>
<customerBirthdate xsi:type="xsd:string">1960-03-20</customerBirthdate>
<buyerNameFirst xsi:type="xsd:string"></buyerNameFirst>
<buyerNameLast xsi:type="xsd:string"></buyerNameLast>
<buyerAddress xsi:type="xsd:string"></buyerAddress>
<buyerCity xsi:type="xsd:string"></buyerCity>
<buyerState xsi:type="xsd:string"></buyerState>
<buyerZip xsi:type="xsd:string"></buyerZip>
<buyerBirthdate xsi:type="xsd:string"></buyerBirthdate>
<productSKU xsi:type="xsd:string">W-123-01</productSKU>
<productBrand xsi:type="xsd:string">Compli</productBrand>
<productName xsi:type="xsd:string">Old Vines Zin</productName>
<productClass xsi:type="xsd:string">WINE</productClass>
<productVintage xsi:type="xsd:string">2006</productVintage>
<productAlcohol xsi:type="xsd:float">14.5</productAlcohol>
<productIsSample xsi:type="xsd:boolean">0</productIsSample>
<productIsForeign xsi:type="xsd:boolean">0</productIsForeign>
<bottleSize xsi:type="xsd:float">750</bottleSize>
<bottleSizeUnit xsi:type="xsd:string">ML</bottleSizeUnit>
<containerPack xsi:type="xsd:integer">1</containerPack>
<containerCount xsi:type="xsd:integer">3</containerCount>
<beveragePrice xsi:type="xsd:float">32.95</beveragePrice>
<foodPrice xsi:type="xsd:float">0.00</foodPrice>
<otherPrice xsi:type="xsd:float">0.00</otherPrice>
<salesTax xsi:type="xsd:float">1.98</salesTax>
<invoiceNumber xsi:type="xsd:string"></invoiceNumber>
<invoiceDate xsi:type="xsd:string"></invoiceDate>
<shipType xsi:type="xsd:string">D</shipType>
<shipOnSite xsi:type="xsd:boolean">0</shipOnSite>
<shipDate xsi:type="xsd:string"></shipDate>
<shipPrice xsi:type="xsd:string"></shipPrice>
<shipCarrierCode xsi:type="xsd:string"></shipCarrierCode>
<shipTrackingNumber xsi:type="xsd:string"></shipTrackingNumber>
</orderLine>
</tns:submitOrder>

```

Example 3: orderLine (command = MODIFY)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:ecompli_webservice_v3">
<SOAP-ENV:Body>
<tns:submitOrder xmlns:tns="urn:ecompli_webservice_v3">
<orderLine xsi:type="tns:orderLine">
<accountNumber xsi:type="xsd:string">ACCT55555555</accountNumber>
<accountKey xsi:type="xsd:string">KEY123456789</accountKey>
<command xsi:type="xsd:string">MODIFY</command>
<mode xsi:type="xsd:string">EXECUTE</mode>
<orderNumber xsi:type="xsd:string">00001</orderNumber>
<orderLineNumber xsi:type="xsd:integer">1</orderLineNumber>
<invoiceNumber xsi:type="xsd:string">A1234001</invoiceNumber>
<invoiceDate xsi:type="xsd:string">2009-08-24</invoiceDate>
<shipDate xsi:type="xsd:string">2009-08-24</shipDate>
<shipType xsi:type="xsd:string">D</shipType>
<shipOnSite xsi:type="xsd:string">0</shipOnSite>
<shipPrice xsi:type="xsd:string">3.49</shipPrice>
<shipCarrierCode xsi:type="xsd:string">UPS</shipCarrierCode>
<shipTrackingNumber xsi:type="xsd:string">67765453434450009989</shipTrackingNumber>
</orderLine>
</tns:submitOrder>

```

Note: unused elements not shown

Example 4: orderLine (command = DELETE)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:ecompli_webservice_v3">
<SOAP-ENV:Body>
<tns:submitOrder xmlns:tns="urn:ecompli_webservice_v3">
<orderLine xsi:type="tns:orderLine">
<accountNumber xsi:type="xsd:string">ACCT55555555</accountNumber>
<accountKey xsi:type="xsd:string">KEY123456789</accountKey>
<command xsi:type="xsd:string">DELETE</command>
<mode xsi:type="xsd:string">EXECUTE</mode>
<invoiceNumber xsi:type="xsd:string">00001</invoiceNumber>
</orderLine>
</tns:submitOrder>
```

Note: unused elements not shown.

Example 5: result: (command = VALIDATE)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:ecompli_webservice_v3">
<SOAP-ENV:Body>
<ns1:submitOrderResponse xmlns:ns1="urn:ecompli_webservice_v3">
<result xsi:type="tns:result">
<accountNumber xsi:type="xsd:string">ACCT55555555</accountNumber>
<orderNumber xsi:type="xsd:string">00001</orderNumber>
<orderLineNumber xsi:type="xsd:integer">3</orderLineNumber>
<orderLinesTotal xsi:type="xsd:integer">3</ orderLinesTotal>
<resultCode xsi:type="xsd:integer">1</resultCode>
<resultMessage xsi:type="xsd:string">Success: Validation Complete</resultMessage>
<salesTax xsi:type="xsd:integer">0.06</salesTax>
<stateLiters xsi:type="xsd:integer">0</stateLiters>
<customerLiters xsi:type="xsd:integer">11.25</customerLiters>
<validState xsi:type="xsd:integer">1</validState>
<validZip xsi:type="xsd:integer">1</validZip>
<validLicense xsi:type="xsd:integer">1</validLicense>
<validBrand xsi:type="xsd:integer">3</validBrand>
<validStateLimit xsi:type="xsd:integer">3</validStateLimit>
<validCustomerLimit xsi:type="xsd:integer">1</validCustomerLimit>
<validCustomerAge xsi:type="xsd:integer">4</validCustomerAge>
<validBuyerAge xsi:type="xsd:integer">4</validBuyerAge>
<orderCompliance xsi:type="xsd:integer">1</orderCompliance>
</result>
</ns1:submitOrderResponse>
```

Example 6: result: (command = ADD)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:ecompli_webservice_v3">
<SOAP-ENV:Body>
<ns1:submitOrderResponse xmlns:ns1="urn:ecompli_webservice_v3">
<result xsi:type="tns:result">
<accountNumber xsi:type="xsd:string">ACCT55555555</accountNumber>
<orderNumber xsi:type="xsd:string">00001</orderNumber>
<orderLineNumber xsi:type="xsd:integer">1</orderLineNumber>
<orderLinesTotal xsi:type="xsd:integer">1</orderLinesTotal>
<resultCode xsi:type="xsd:integer">1</resultCode>
<resultMessage xsi:type="xsd:string">Success: Invoice Line Added</resultMessage>
<salesTax xsi:type="xsd:integer">0.06</salesTax>
<stateLiters xsi:type="xsd:integer">0</stateLiters>
<customerLiters xsi:type="xsd:integer">11.25</customerLiters>
<validState xsi:type="xsd:integer">1</validState>
<validZip xsi:type="xsd:integer">1</validZip>
<validLicense xsi:type="xsd:integer">1</validLicense>
<validBrand xsi:type="xsd:integer">3</validBrand>
<validStateLimit xsi:type="xsd:integer">3</validStateLimit>
<validCustomerLimit xsi:type="xsd:integer">1</validCustomerLimit>
<validCustomerAge xsi:type="xsd:integer">4</validCustomerAge>
<validBuyerAge xsi:type="xsd:integer">4</validBuyerAge>
<orderCompliance xsi:type="xsd:integer">1</orderCompliance>
</result>
</ns1:submitOrderResponse>
```