



# Integration Service File share Documentation

## Document Control

**Author:** Tracesense Limited

**Change Authority:** Tracesense Limited

### Revision History:

Document Version	Date Issued	Status	Change Request #	Reason for Change
1.0		New	Release	

**This document will be kept under revision control.**

## Contents

INTRODUCTION.....	5
OVERVIEW.....	5
INSTALLATION.....	6
CONFIGURATION.....	6
IMPORT SYSTEM.....	7
SUPPORTED FILE TYPES.....	7
XML.....	7
CSV.....	7
RESULT ACTIONS.....	7
EXPORT SYSTEM.....	8
SUPPORTED FILE TYPES.....	8
XML.....	8
CSV.....	8
RESULT ACTIONS.....	8
SUPPORTED DATA TYPES.....	9
XML STRUCTURE.....	10
COMMODITY.....	10
Allergens.....	14
Each Commodity can have zero or many Allergens linked to it.....	14
Example File.....	15
CONTAINER.....	16
Example File.....	17
CUSTOMER.....	17
Addresses.....	19
Example File.....	20

LOCATION.....	21
Example File .....	21
PRODUCTION .....	22
Example File .....	25
PURCHASE ORDER.....	26
Purchase Order Lines .....	27
Example File .....	29
Example File – Purchase Order Fail .....	30
REASON .....	31
Example File .....	31
Example File – Reason Fail .....	32
RECIPE .....	33
Recipe Steps .....	35
Each Recipe can have zero or many Recipe Steps linked to it. ....	35
Example File .....	38
REQUIREMENT .....	43
Example File .....	44
SALESORDER.....	45
Sales Order Lines.....	46
Example File .....	47
Example File – Sales Order Fail.....	48
STOCK.....	49
Example File .....	51
STOCK LOCATION.....	52
Example File .....	52
STOCK UNIT .....	53

Example File .....	54
SUPPLIER .....	55
Example File .....	56
TRANSACTION .....	57
USER .....	63
Example File .....	63
EXAMPLE CSV HEADERS .....	64

# INTRODUCTION

The purpose of this manual is to provide all installers and users of Tracesense Integration with a reference manual containing instructions and examples of how to configure and use the application effectively and efficiently.

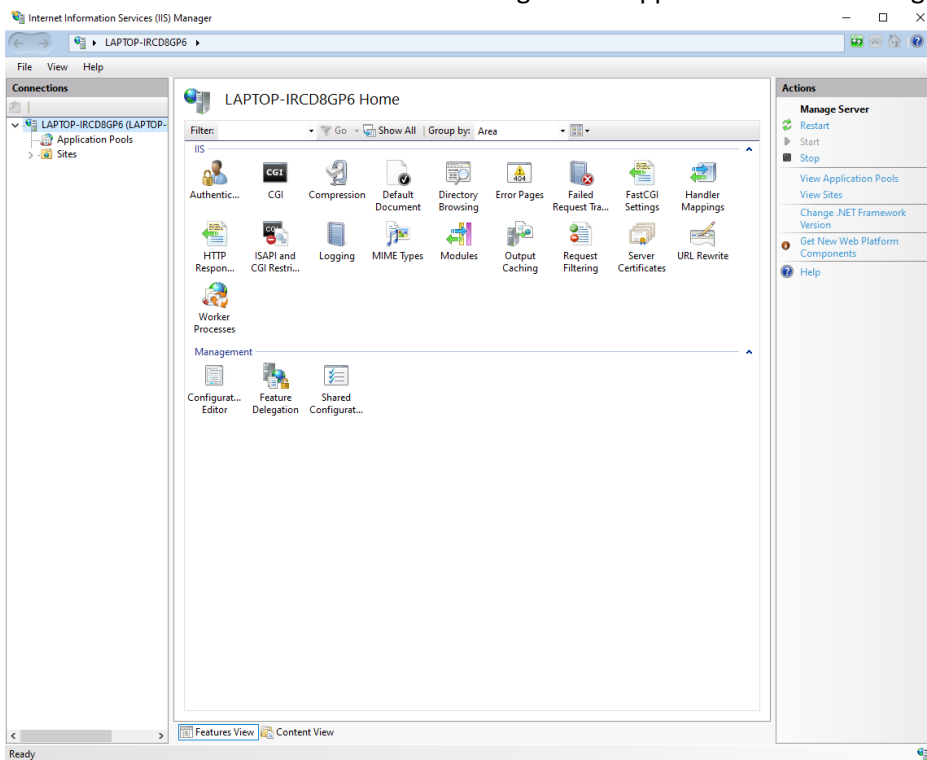
## OVERVIEW

The Integration software is a Django Web Application hosted on windows IIS, which provides a standardised utility for the import of data into, and the export of data from, the Tracesense Recipe Formulation System. This provides a means of communication between the Tracesense Recipe Formulation System and other third- party Enterprise Resource Planning (ERP) Systems and applications.

The application functions by monitoring for acceptable file formats (XML or CSV) in specified directories on the host, or a network accessible, computer system. The import files may be generated by third party software and placed in the monitored directories to transact information from one database format, into the Tracesense system.

The export module constantly listens for specific types of data in the Tracesense Recipe Formulation System and exports the data whenever the conditions are satisfied. The methods of export include full exports (complete dumps of specified data in the system) or update exports (whenever a change has occurred within the Tracesense system, warranting an export).

Once installed Windows IIS must be running for the application to be running.



## **INSTALLATION**

The Tracesense Integration will be enabled if you have purchased this module. Once the IIS application is set up then the application will be monitoring the correct directories for imports/exports.

## **CONFIGURATION**

The Tracesense Integration benefits from being highly configurable to suit the individual needs of each customer. Configuration can either be done through the provided User Interface (which only shows a limited number of settings), or via Tracesense Central which shows all available settings.

## **IMPORT SYSTEM**

### **SUPPORTED FILE TYPES**

The system supports two file types – XML (.xml) and CSV (.csv). The system is set up to accept both file types by default but can be configured to only accept a specified file type if required. Files must be placed in the 'Pending' directory (or the directory specified by the configuration), and if there are numerous files, they will be processed in chronological order.

### **XML**

If importing an XML file, it must conform to the schema declaration supplied (A digital version of this can be found in the 'Documents' directory once the application has been installed). Support for the IS1 schema has been implemented, ensuring backwards compatibility for all existing customers.

### **CSV**

If importing a CSV file, the system must be configured correctly to ensure it can determine the structure of the CSV file. This is either done by supplying a 'header' row in the CSV file to determine the columns, or by supplying a separate file detailing the columns and placing this in the 'CSVHeaders' directory, before setting the corresponding configuration.

### **RESULT ACTIONS**

The import system iterates through the files it finds in the specified 'pending' folder upon the 'Import Interval' being elapsed. For each file, the system will parse and import its contents into the system database.

Once the Integration has finished processing a file, it will be moved to the specified 'processed' directory. A file will be created in each of the specified 'success' and 'fail' directories. These files will show which entries have been successful, and which have failed.

Any 'failed' files will be placed into the correct failed directory and reasons for the failure will be in the system log file.

## **EXPORT SYSTEM**

### **SUPPORTED FILE TYPES**

The system supports two file types – XML (.xml) and CSV (.csv). The Export mechanism will export files in the file format specified in the configuration.

#### **XML**

By default, the exported XML will conform to the schema declaration supplied (A digital version of this can be found in the 'Documents' directory once the application has been installed).

The software can be configured to output XML conforming to a custom schema declaration rather than the default, providing high flexibility to all users (This option is particularly useful when upgrading a client site which may require the formats to remain in their current IS1 format).

#### **CSV**

The exported CSV file will either use the columns found in the corresponding CSV header, or it will use the system defaults (Appendix B).

### **RESULT ACTIONS**

Exported files will always be placed in the directory specified by the relevant configuration. No log files are generated when exporting files. If you expected to see an export, but one was not produced, it may be worthwhile checking the log files found in the local 'log' directory.



## SUPPORTED DATA TYPES

Type
Commodity
Customer
Location
Purchase Order
Question
Reason
Recipe
Requirement
Sales Order
Stock
Stock Location
Stock Unit
Supplier
User

## XML STRUCTURE

The following pages detail the structure of the various XML files that are processed within the system, with a full example for each specified type.

### COMMODITY

A Commodity Import provides the functionality to create or update commodities in the Tracesense Systems. An export provides details of newly created/edited Commodities.

#### Elements

The following table details the structure of a Commodity XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>accountLink</b>	<b>int</b>	-	<b>0</b>	
<b>allergenWarning</b>	<b>int</b>	-	<b>0</b>	
Used to determine whether an Allergen Warning should be displayed when this commodity is used on the Production Terminal [0 – No, 1 – YES].				
<b>bagMode</b>	<b>int</b>	-	<b>0</b>	
A value determining whether the default bag weight should be used (0 – disabled, 1 – enabled).				
<b>barcode</b>	<b>string</b>	<b>255</b>	<b>ST-21022018-A1</b>	
Used to set the barcode value for the commodity.				
<b>budgetCost</b>	<b>double</b>	-	<b>2.50</b>	
<b>caseQTY</b>	<b>double</b>	-	<b>10.0</b>	
Sets the total number of this Commodity that can be held within 1 case.				
<b>code</b>	<b>string</b>	<b>35</b>	<b>STSPRODUCT1</b>	<b>TRUE</b>
The Code to be used for this Commodity. This is a unique value and will always be represented in UPPER CASE throughout the Tracesense Application Suite.				
<b>costPerUnit</b>	<b>double</b>	-	<b>10.0</b>	

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
The cost per unit of this Commodity.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the commodity is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>defaultBagWeight</b>	<b>double</b>	<b>-</b>	<b>5.0</b>	
The default bag weight of the commodity.				
<b>defaultExpiryDays</b>	<b>int</b>	<b>-</b>	<b>5</b>	
A value used when automatically setting the expiry date of a produced commodity				
<b>defaultLocation</b>	<b>string</b>	<b>20</b>	<b>FREEZER01</b>	
<b>denominationType</b>	<b>int</b>	<b>-</b>	<b>0</b>	
<b>dribbleWeight</b>	<b>double</b>	<b>-</b>	<b>0.0</b>	
The weight used in conjunction with auto-filling, to determine when the switch will slow down.				
<b>fillerNumber</b>	<b>int</b>	<b>-</b>	<b>0</b>	
The electrical output associated with the ingredient, a direct physical link to a PLC number.				
<b>forceChangeLotWeight</b>	<b>double</b>	<b>-</b>	<b>15.0</b>	
The value at which the currently selected LOT should be changed when being used in Production.				
<b>globalStockDetID</b>	<b>int</b>	<b>-</b>	<b>0</b>	
Used to set a Global Stock Det for this commodity				
<b>GTIN</b>	<b>string</b>	<b>255</b>	<b>12345678912345</b>	
The GTIN for the commodity (GTIN-13 or GTIN-14).				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>grams</b>	<b>int</b>	-	<b>0</b>	
<b>inFlightReCalc</b>	<b>int</b>	-	<b>0</b>	
The value used to calculate the next actual required amount on the next weighing if the inflight weight value was used previously, ensuring that the overall amount will add up.				
<b>inFlightWeight</b>	<b>float</b>	-	<b>0.0</b>	
The weight transferred which will still be transferred after the system has shut off weighing (product which may still remain in the pipes after shut off).				
<b>Instructions</b>	<b>string</b>	<b>2048</b>	<b>An Instruction</b>	
The instruction that should be displayed when using this Commodity in Production.				
<b>labelPrinterTypeID</b>	<b>int</b>	-		
<b>lastStockIn</b>	<b>string*</b>	-	<b>01/01/2018</b>	
A date representing the last date/time Stock of this Commodity was Received or Produced.				
<b>maxPalletQTY</b>	<b>int</b>	-	<b>1000</b>	
<b>name</b>	<b>string</b>	<b>255</b>	<b>STS PRODUCT 1</b>	<b>TRUE</b>
The name of the commodity. If the element is omitted, the name will be derived from the commodity code instead.				
<b>nonStockItem</b>	<b>int</b>	-	<b>0</b>	
A value to determine whether this Commodity is a 'Non-Stock' item (0 – Stock Item, 1 – Non-Stock Item)				
<b>notes</b>	<b>string</b>	<b>1024</b>	<b>NEW PRODUCT</b>	

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
Any notes that should be attached to this Commodity.				
<b>productOfCommodity</b>	<b>string</b>	<b>35</b>		
If the Commodity is the Product of another Commodity, the Parent Commodity Code must be entered.				
<b>reOrderWeight</b>	<b>double</b>	<b>-</b>	<b>25.0</b>	
The Weight at which an order for this Commodity should be placed.				
<b>reOrderQTY</b>	<b>double</b>	<b>-</b>	<b>10.0</b>	
The Quantity at which an order for this Commodity should be placed.				
<b>scale</b>	<b>int</b>	<b>-</b>	<b>0</b>	
<b>sdFactor</b>	<b>double</b>	<b>-</b>	<b>0.0</b>	
<b>stockUnit</b>	<b>string</b>	<b>-</b>	<b>KG</b>	<b>TRUE</b>
The Stock Unit used for this Commodity.				
<b>strictLocation</b>	<b>int</b>	<b>-</b>	<b>0</b>	
<b>superLotID</b>	<b>int</b>	<b>-</b>		
<b>syncNotesInstructions</b>	<b>string</b>	<b>-</b>		
<b>tare</b>	<b>double</b>	<b>-</b>	<b>1.0</b>	
The tare weight of the commodity.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>type</b>	<b>int</b>	-	<b>0</b>	
The type of commodity being imported (0 – ingredient, 1 – recipe).				
<b>UPC</b>	<b>string</b>	<b>255</b>		
Sets the UPC for this Commodity.				
<b>validUntil</b>	<b>string*</b>	-	<b>12/10/2021</b>	
Sets the Date for which this commodity valid until.				

\* This value must be in the same Date Format as determined by the related Integration Configuration.

### *Allergens*

Each Commodity can have zero or many Allergens linked to it.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>code</b>	<b>string</b>	<b>50</b>	<b>EGG</b>	
The Code of an Allergen that is found within this Commodity				

## Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <commodities>
    <commodity>
      <accountLink />
      <allergens>
        <allergen>
          <code> EGG </code>
        </allergen>
      </allergens>
      <allergenWarning />
      <bagMode>0</bagMode>
      <bagName />
      <barcode />
      <budgetCost>0</budgetCost>
      <caseQTY />
      <code>WF001</code>
      <costPerUnit>1</costPerUnit>
      <deleted />
      <defaultBagWeight>25</defaultBagWeight>
      <defaultExpiryDays>0</defaultExpiryDays>
      <defaultLocation />
      <denominationType />
      <dribbleWeight>0</dribbleWeight>
      <fillerNumber>0</fillerNumber>
      <forceChangeLotWeight>1</forceChangeLotWeight>
      <globalStockDetID />
      <GTIN />
      <grams>0</grams>
      <inFlightReCalc>0</inFlightReCalc>
      <inFlightWeight>0</inFlightWeight>
      <instructions />
      <labelPrinterTypeID />
      <lastStockIn />
      <lotIncrement>1</lotIncrement>
      <maxPalletQTY />
      <messageListID />
      <name>White Flour</name>
      <nonStockItem />
      <notes />
      <productOfCommodity />
      <reOrderWeight />
      <reOrderQTY>0</reOrderQTY>
      <scale>1</scale>
      <sdFactor>0</sdFactor>
      <stockUnit>KG</stockUnit>
      <strictLocation>0</strictLocation>
      <superLotID />
      <syncNotesInstructions />
      <tare />
      <type>0</type>
      <UPC />
      <tolerances />
    </commodity>
  </commodities>
</Tracesense>
```

---

## CONTAINER

A Container Import provides the functionality to create or update containers in the Tracesense Systems.

### Elements

The following table details the structure of a Customer XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>code</b>	<b>string</b>	<b>50</b>	<b>CONTAINER1</b>	<b>TRUE</b>
The Code to be used for this Container. This is a unique value and will always be represented in UPPER CASE throughout the Tracesense Application Suite.				
<b>name</b>	<b>string</b>	<b>50</b>	<b>TEST CONTAINER ONE</b>	<b>TRUE</b>
The name of the Container.				
<b>description</b>	<b>string</b>	<b>100</b>	<b>TEST CONTAINER DESC</b>	<b>TRUE</b>
Any description that should be applied to this Container.				
<b>location</b>	<b>string</b>	<b>20</b>	<b>WEIGHUP-4</b>	
The Location of this container. If not provided, will use default location.				
<b>Type</b>	<b>string</b>	<b>20</b>	<b>TYPE1</b>	
The Type of this container. If not provided, will use default Type.				
<b>group</b>	<b>int</b>	<b>20</b>		
The Group this container belongs to. If not provided, will use default group.				
<b>tare</b>	<b>string</b>	<b>-</b>	<b>1.5</b>	
The Tare weight of the container.				



### Example File

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <containers>
    <container>
      <code>CONTAINER1</code>
      <name>TEST CONTAINER ONE</name>
      <description>TEST CONTAINER DESC</description>
      <tare>1.5</tare>
      <type>TYPE1</type>
      <group>MAIN</group>
      <location>WEIGHUP-4</location>
    </container>
  </containers>
</Tracesense>
```

### CUSTOMER

A Customer Import provides the functionality to create or update customers in the Tracesense Systems. Anexport provides details of newly created/edited Customers.

### Elements

The following table details the structure of a Customer XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>code</b>	<b>string</b>	<b>40</b>	<b>CUSTOMER1</b>	<b>TRUE</b>
The Code to be used for this Customer. This is a unique value and will always be represented in UPPER CASE throughout the Tracesense Application Suite.				
<b>contactName</b>	<b>string</b>	<b>50</b>	<b>TEST CONTACT 1</b>	
The main Contact Name for this Customer.				
<b>contactEmail</b>	<b>string</b>	<b>70</b>	<b>info@acustomer.com</b>	
The main Contact Email for this Customer.				
<b>customerGroupID</b>	<b>int</b>	<b>-</b>		
Provides a way of specifying this Customer belongs to a parent group				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Customer is marked as deleted (FALSE - not deleted, TRUE – deleted).				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>fax</b>	<b>string</b>	<b>20</b>	<b>01254 685201</b>	
A value representing the fax contact number for this Customer				
<b>frequency</b>	<b>int</b>	<b>-</b>		
<b>invoiceAddress</b>	<b>string</b>	<b>150</b>		
The address at which to send Invoice's that relate to this Customer.				
<b>mobileTelephone</b>	<b>string</b>	<b>20</b>	<b>07980674149</b>	
A mobile telephone number that can be used for this Customer				
<b>name</b>	<b>string</b>	<b>120</b>	<b>TEST CUSTOMER 1</b>	<b>TRUE</b>
The name of the Customer.				
<b>notes</b>	<b>string</b>	<b>255</b>	<b>TEST CONTACT NOTES</b>	
Any notes that should be applied to this Customer.				
<b>nextVisit</b>	<b>string*</b>	<b>-</b>	<b>25/12/2018</b>	
The date of the Customers next visit.				
<b>settings</b>	<b>int</b>	<b>-</b>		
<b>shortestUseBy</b>	<b>int</b>	<b>-</b>	<b>5</b>	
A value that specifies the 'shortest use by' value for this Customer.				
<b>siteCode</b>	<b>string</b>	<b>-</b>		
<b>specialInstructions</b>	<b>string</b>	<b>250</b>	<b>A SPECIAL INSTRUCTION</b>	
Any Special Instructions that should be applied to this Customer				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>telephone</b>	<b>string</b>	<b>60</b>	<b>01254 685200</b>	
The main telephone number for this Customer.				
<b>website</b>	<b>string</b>	<b>50</b>	<b>www.Tracesense.com</b>	
The Customers website URL.				

\* This value must be in the same Date Format as determined by the related Integration Configuration.

#### Addresses

Each Customer can have zero or many Addresses linked to it.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>address</b>	<b>string</b>	<b>255</b>		<b>TRUE</b>
The Address.				
<b>addressCode</b>	<b>string</b>	<b>255</b>	<b>ADDRESS-1</b>	<b>TRUE</b>
The Address Code.				
<b>customerCode</b>	<b>string</b>	<b>40</b>	<b>CUSTOMER1</b>	
The Customer that relates to this Address				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the commodity is marked as deleted (FALSE - not deleted, TRUE – deleted).				

## Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <customers>
    <customer>
      <code>CUSTOMER1</code>
      <contactName>Test Contact 1</contactName>
      <contactEmail>info@Tracesense.com</contactEmail>
      <customerGroupID>1</customerGroupID>
      <deleted />
      <fax>01254 685201</fax>
      <frequency>0</frequency>
      <invoiceAddress>12 Manse Avenue, Armadale, Eh483hs
</invoiceAddress>
      <mobileTelephone>07980674149</mobileTelephone>
      <name>Test Customer 1</name>
      <notes>Test Contact Notes</notes>
      <nextVisit>01/12/2017</nextVisit>
      <settings>0</settings>
      <shortestUseBy>1</shortestUseBy>
      <siteCode />
      <specialInstructions />
      <telephone>01254 685200</telephone>
      <website>www.Tracesense.com</website>
      <customerAddresses>
        <customerAddress>
          <address>Tracesense HQ, Greenbank Technology Park, Challenge Way,
Blackburn, BB1 5QB</address>
          <addressCode>ADDRESS-1</addressCode>
          <customerCode>CUSTOMER1</customerCode>
          <deleted />
        </customerAddress>
        <customerAddress>
          <address>Tracesense South, Redbank Technology Park, Victory Way, London, SW1W
0DT</address>
          <addressCode>ADDRESS-2</addressCode>
          <customerCode>CUSTOMER1</customerCode>
          <deleted />
        </customerAddress>
      </customerAddresses>
    </customer>
  </customers>
</Tracesense>
```

---

## LOCATION

A Location Import provides the functionality to create or update Locations in the Tracesense Systems. Anexport provides details of newly created/edited Locations.

### Elements

The following table details the structure of a Location XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>autoLogOffDelay</b>	<b>int</b>	<b>-</b>	<b>0</b>	
The amount of time (in seconds) before an inactive Operator should be logged out of the system.				
<b>code</b>	<b>string</b>	<b>20</b>	<b>WEIGHUP-4</b>	<b>TRUE</b>
The Location Code.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>description</b>	<b>string</b>	<b>50</b>	<b>Recipe Weighup 4</b>	
The Description of the Location.				
<b>locationType</b>	<b>string</b>	<b>10</b>		
<b>portionScaleOnly</b>	<b>int</b>	<b>-</b>	<b>0</b>	
Determines whether this Location can only hold Portion Scales.				

### Example File

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <locations>
    <location>
      <autoLogOffDelay>0</autoLogOffDelay>
      <code>WEIGHUP-4</code>
      <deleted />
      <description> Recipe Weigh Up Location 4</description>
      <locationType />
      <portionScaleOnly />
    </location>
  </locations>
</Tracesense>
```

## PRODUCTION

A Production Export provides details of production completed within the Tracesense System. This includes data relating to the commodity produced, and the commodities consumed.

### Elements

The following table details the structure of a Production XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>batchNumber</b>	<b>int</b>	-	<b>943</b>	
The Batch Number that has been/is being produced.				
<b>batchSequence</b>	<b>int</b>	-	<b>0</b>	
The Batch Sequence value.				
<b>code</b>	<b>string</b>	<b>35</b>	<b>TESTREC1</b>	
The code of the Commodity that is required to be produced.				
<b>cost</b>	<b>double</b>	-	<b>5.50</b>	
The cost of the Production.				
<b>created</b>	<b>string*</b>	-	<b>17/10/2018</b>	
<b>completed</b>	<b>string*</b>	-	<b>17/10/2018</b>	
The date that this Production was completed.				
<b>expiryDate</b>	<b>string*</b>	-	<b>17/12/2018</b>	
The Expiry Date of the Produced Commodity.				
<b>jobNumber</b>	<b>string</b>	<b>22</b>	<b>JOB-01</b>	
The Job Number of the Job that is producing/has produced this entity.				
<b>jobStatus</b>	<b>int</b>	-	<b>1</b>	
The status of the Job (0 – Created, 1 – In Progress, 2 – Complete, 3- Aborted).				

<b>lotNumber</b>	<b>string</b>	<b>30</b>	<b>13/271017</b>	
The unique reference to this Stock.				
<b>name</b>	<b>string</b>	<b>255</b>	<b>STS PRODUCT 1</b>	
The name of the commodity.				
<b>producedMixSize</b>	<b>double</b>	<b>-</b>	<b>100.00</b>	
The size of the mix produced.				
<b>productionLocation</b>	<b>string</b>	<b>20</b>	<b>ANY</b>	
The Location where this Production is/was taking place.				
<b>salesOrder</b>	<b>string</b>	<b>255</b>		
If the production has been scheduled from to meet the requirement of a Sales Order, the related Sales Order Order Number will be given.				
<b>scheduledMixSize</b>	<b>double</b>	<b>-</b>	<b>100.00</b>	
The size of the mix that was scheduled.				
<b>scheduledQuantity</b>	<b>double</b>	<b>-</b>	<b>100</b>	
The quantity that the scheduled job should produce. This has an attribute of “units” that will detail what Stock Unit the quantity is measured in.				
<b>siteId</b>	<b>int</b>	<b>-</b>		
The ID of the Site where this has been produced.				
<b>started</b>	<b>string*</b>	<b>-</b>	<b>12/10/2018</b>	
The Date/Time that this production was started.				
<b>quantity</b>	<b>Double</b>	<b>-</b>	<b>100</b>	
The quantity that the job produced. This has an attribute of “units” that will detail what Stock Unit the quantity is measured in.				

## Consumption

Each Production export can have zero or many Consumption entries linked to it.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>code</b>	<b>string</b>	<b>35</b>	<b>3RM0725</b>	
The code of the Commodity that is required to be produced.				
<b>containerCode</b>	<b>string</b>	<b>255</b>	<b>VC13693-1</b>	
The code of the Container that was used.				
<b>created</b>	<b>string*</b>	<b>-</b>	<b>17/10/2018</b>	
The date that this consumption was processed.				
<b>lineNumber</b>	<b>int</b>	<b>-</b>		
<b>lotNumber</b>	<b>string</b>	<b>30</b>	<b>13/271017</b>	
The unique reference to this Stock.				
<b>name</b>	<b>string</b>	<b>255</b>	<b>Triethanolamine</b>	
The name of the commodity.				
<b>quantity</b>	<b>Double</b>	<b>-</b>	<b>100</b>	
The quantity that has been consumed. This has an attribute of "units" that will detail what Stock Unit the quantity is measured in.				
<b>stockLocation</b>	<b>string</b>	<b>20</b>	<b>ANY</b>	
The Location where this Production is/was taking place.				
<b>cost</b>	<b>double</b>	<b>-</b>	<b>5.50</b>	
The cost of the Production.				
<b>sequenceNo</b>	<b>Int</b>	<b>-</b>	<b>1</b>	
The sequence number of commodity.				



\* This value must be in the same Date Format as determined by the related Integration Configuration.

### Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <production>
    <produced status="I">
      <batchNumber>943</batchNumber>
      <batchSequence>0</batchSequence>
      <code>2TWM003</code>
      <cost />
      <created>16/11/2017 15:19:18</created>
      <completed />
      <expiryDate />
      <jobNumber>N01145290001</jobNumber>
      <jobStatus>1</jobStatus>
      <lotNumber />
      <name>Aloe Vera Gel (2017)</name>
      <producedMixSize>100.00</producedMixSize>
      <productionLocation>ANY</productionLocation>
      <salesOrder />
      <scheduledMixSize>100.00</scheduledMixSize>
      <scheduledQuantity units="Kg">100</scheduledQuantity>
      <siteId />
      <started>16/11/2017 15:19:55</started>
      <stockProductionLocation>BUILDS</stockProductionLocation>
      <quantity units="Kg">100</quantity>
      <consumption>
        <consumed>
          <code>3RM0725</code>
          <containerCode>VC13693</containerCode>
          <created>17/11/2017 16:20:21</created>
          <lineNumber />
          <lotNumber>138685</lotNumber>
          <name>Triethanolamine 99% BP(TGA)</name>
          <quantity units="kg">10.32</quantity>
          <stockLocation>WHDISP</stockLocation>
          <cost>0.00</cost>
          <sequenceNo>1</sequenceNo>
        </consumed>
      </consumption>
    </produced>
  </production>
</Tracesense>
```

---

## PURCHASE ORDER

A Purchase Order Import provides the functionality to create or update Purchase Orders in the Tracesense Systems. An export provides details of newly created/edited Purchase Orders and gives details of any receipts that may have been completed.

### Elements

The following table details the structure of a Purchase Order XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>createdDate</b>	<b>string*</b>	-	<b>01/01/2018</b>	
The Date the Purchase Order was Created.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>deliveryDate</b>	<b>string*</b>	-	<b>01/01/2018</b>	
The Date the Purchase Order is/was delivered.				
<b>orderNumber</b>	<b>string</b>	<b>255</b>	<b>PO-001</b>	<b>TRUE</b>
The unique Order Number for the Purchase Order.				
<b>orderReference</b>	<b>string</b>	<b>255</b>	<b>PO-001</b>	
An Order Reference for the Purchase Order.				
<b>siteCode</b>	<b>string</b>	<b>10</b>	<b>HEADOFFICE</b>	
The Site at which the Purchase Order is due to be delivered.				
<b>supplierCode</b>	<b>string</b>	<b>10</b>	<b>SUPPLIER1</b>	
The Code of the Supplier for this Purchase Order.				

\* This value must be in the same Date Format as determined by the related Integration Configuration.

### Purchase Order Lines

Each Purchase Order can have zero or many Purchase Order Lines linked to it. When exporting a Purchase Order, it is possible to configure the system so that it only exports Purchase Order Lines that have been received, a full export, or only an export on completion.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>commodityCode</b>	<b>string</b>	<b>35</b>	<b>TESTING002</b>	<b>TRUE</b>
The Code of the Commodity for this Purchase Order Line.				
<b>cost</b>	<b>double</b>	<b>-</b>	<b>2.00</b>	
The Cost of the Purchase Order Line.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>denominationType</b>	<b>int</b>	<b>1</b>	<b>0</b>	
A value determining the Denomination Type of this Purchase Order Line.				
<b>expiryDate</b>	<b>string*</b>	<b>-</b>	<b>19/05/2018</b>	
The Expiry Date of the Stock that has been received as part of this Purchase Order Line.				
<b>lotNumber</b>	<b>string</b>	<b>255</b>		
The LotNumber that this Purchase Order Line was received into.				
<b>notes</b>	<b>string</b>	<b>40</b>	<b>URGENT JOB</b>	
Any notes that should be applied to this Purchase Order Line.				
<b>orderedQuantity</b>	<b>double</b>	<b>-</b>	<b>5.00</b>	
The Quantity ordered.				
<b>orderedRequired</b>	<b>double</b>	<b>-</b>	<b>5.00</b>	
The remaining Quantity to be received.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>receivedDate</b>	<b>string*</b>	-	<b>22/02/2018</b>	
The Date that this Purchase Order Line was received.				
<b>receivedQuantity</b>	<b>double</b>	-	<b>0.0</b>	
The Quantity of Stock received.				
<b>stockUnit</b>	<b>string</b>	<b>10</b>	<b>Kg</b>	
The Stock Unit that this Purchase Order Line will be measured in.				

## Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <purchaseOrders>
    <purchaseOrder>
      <createdDate>29/09/2017</createdDate>
      <deleted />
      <deliveryDate>30/09/2017</deliveryDate>
      <orderNumber>PO-001</orderNumber>
      <orderReference>PO-001</orderReference>
      <siteCode>HEADOFFICE</siteCode>
      <status>0</status>
      <supplierCode>SUPPLIER1</supplierCode>
      <purchaseOrderLines>
        <purchaseOrderLine>
          <commodityCode>TESTING02</commodityCode>
          <cost>2.00</cost>
          <deleted />
          <denominationType>0</denominationType>
          <expiryDate />
          <lotNumber />
          <notes />
          <orderedQuantity>5</orderedQuantity>
          <orderedRequired>5</orderedRequired>
          <receivedDate />
          <receivedQuantity>0</receivedQuantity>
          <stockUnit>KG</stockUnit>
        </purchaseOrderLine>
        <purchaseOrderLine>
          <commodityCode>TESTING03</commodityCode>
          <cost>3.00</cost>
          <deleted />
          <denominationType>0</denominationType>
          <expiryDate />
          <lotNumber />
          <notes />
          <orderedQuantity>5</orderedQuantity>
          <orderedRequired>5</orderedRequired>
          <receivedDate />
          <receivedQuantity>0</receivedQuantity>
          <stockUnit>KG</stockUnit>
        </purchaseOrderLine>
      </purchaseOrderLines>
    </purchaseOrder>
  </purchaseOrders>
</Tracesense>
```

---

## Example File – Purchase Order Fail

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <purchaseOrders>
    <purchaseOrder>
      <createdDate>05/02/18</createdDate>
      <deleted />
      <deliveryDate />
      <orderNumber>PO-001</orderNumber>
      <orderReference>REF-001</orderReference>
      <siteCode />
      <status>0</status>
      <supplierCode />
      <purchaseOrderLines>
        <purchaseOrderLine>
          <commodityCode />
          <cost>4.80</cost>
          <deleted />
          <denominationType />
          <expiryDate />
          <lotNumber />
          <notes />
          <orderedQuantity>10</orderedQuantity>
          <orderedRequired>10</orderedRequired>
          <receivedDate />
          <receivedQuantity>0</receivedQuantity>
          <stockUnit>kg</stockUnit>
        </purchaseOrderLine>
        <purchaseOrderLine>
          <commodityCode />
          <cost>9.76</cost>
          <deleted />
          <denominationType />
          <expiryDate />
          <lotNumber />
          <notes />
          <orderedQuantity>45</orderedQuantity>
          <orderedRequired>45</orderedRequired>
          <receivedDate />
          <receivedQuantity>0</receivedQuantity>
          <stockUnit>kg</stockUnit>
        </purchaseOrderLine>
      </purchaseOrderLines>
      <failure>
        <message>Object Validation Error</message>
        <cause>The PurchaseOrderHeader Being Imported Failed Validation</cause>
        <errors>
          <error>The [Lines[0].Commodity] Element Failed Validation. Either a matching
[Commodity] was not found, or a [Commodity] was not supplied</error>
          <error>The [Lines[1].Commodity] Element Failed Validation. Either a matching
[Commodity] was not found, or a [Commodity] was not supplied</error>
        </errors>
      </failure>
    </purchaseOrder>
  </purchaseOrders>
</Tracesense>
```

---

## REASON

A Reason Import provides the functionality to create or update Reasons in the Tracesense Systems. An export provides details of newly created/edited Reasons.

### Elements

The following table details the structure of a Reason XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Reason is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>name</b>	<b>string</b>	<b>60</b>	<b>Contaminated</b>	<b>TRUE</b>
The Reason.				

### Example File

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <reasons>
    <reason>
      <deleted />
      <name>Contaminated</name>
    </reason>
  </reasons>
</Tracesense>
```

## Example File – Reason Fail

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <commodities />
  <customers />
  <locations />
  <purchaseOrders />
  <reasons>
    <reason>
      <deleted />
      <name />
      <failure>
        <message>Incorrect CSV line structure detected</message>
        <cause>The number of columns specified in one or more CSV lines, does not match the
number of columns provided in the CSV Header [2]. See 'errors' node for failed lines</cause>
        <errors>
          <error>LostInTransit, error, errorcolumn</error>
          <error>AnotherFailReason, this, should, fail </error>
          <error>Quarantined, too many, columns</error>
        </errors>
      </failure>
    </reason>
  </reasons>
  <recipes />
  <salesOrders />
  <sites />
  <stockHeaders />
  <stockLocations />
  <stockUnits />
  <suppliers />
  <users />
</Tracesense>
```

---



## RECIPE

A Recipe Import provides the functionality to create or update recipes in the Tracesense Systems. An export provides details of newly created/edited recipes.

### Elements

The following table details the structure of a Recipe XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>addToStock</b>	<b>int</b>	-	<b>0</b>	
Value used to set whether production of this Recipe should be added to stock.				
<b>code</b>	<b>string</b>	<b>35</b>	<b>TESTING002</b>	<b>TRUE</b>
The Code of the Commodity for this Recipe. If this does not exist, then a commodity will be created using this value as both it's Code and Name.				
<b>created</b>	<b>string</b>	-	<b>15/04/2019</b>	
The Date this Recipe was created.				
<b>defaultMixSize</b>	<b>double</b>	-	<b>100.0</b>	
Used to set the Default Mix Size when this Recipe is produced.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Reason is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>labelField1-20</b>	<b>string</b>	<b>255</b>		
There are 20 x Label Fields that can be populated with any string value as required. These values are then available to be used when printing labels for this Recipe.				
<b>maxProducts</b>	<b>int</b>	-	<b>30</b>	
The maximum number of products that can be made of this Recipe in one batch.				
<b>minProducts</b>	<b>int</b>	-	<b>1</b>	
The minimum number of products that can be made of this Recipe in one batch.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>name</b>	<b>string</b>	<b>255</b>	<b>STS PRODUCT 1</b>	<b>TRUE</b>
The name of the commodity. If the element is omitted, the name will be derived from the commodity code instead.				
<b>premix</b>	<b>int</b>	<b>-</b>	<b>0</b>	
Indicates whether this Recipe can be used as a Premix or not (0 – No, 1 – Yes).				
<b>productionLocationCode</b>	<b>string</b>	<b>10</b>	<b>ANY</b>	
Specifies which Production Location can be used for this Recipe.				
<b>stockProductionLocationCode</b>	<b>string</b>	<b>50</b>	<b>RAW STOCK</b>	
Specifies which Stock Production Location will be used for this Recipe.				
<b>productPerMinute</b>	<b>int</b>	<b>-</b>	<b>10</b>	
Indicates the Products Per Minute for this Recipe during production.				
<b>requireSignOff</b>	<b>int</b>	<b>-</b>	<b>0</b>	
Specifies whether a supervisor must sign off the completion of production when using this recipe.				
<b>tareWeight</b>	<b>double</b>	<b>-</b>	<b>5.0</b>	
The default tare weight of the Recipe.				
<b>type</b>	<b>int</b>	<b>-</b>	<b>0</b>	<b>TRUE</b>
Indicates what type of Recipe this is. Value must be one of: [0 – Batch, 1- Portion, 2-Packaging, 3-Main_Portion, 4-JobRecipe, 5-Meat, 6-Meat ]				
<b>weighProduction</b>	<b>int</b>	<b>-</b>	<b>0</b>	
Indicates whether the production should be weighed upon production of this recipe being completed.				
<b>yield</b>	<b>double</b>	<b>-</b>	<b>200</b>	
The expected yield of this Recipe.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>maxLifeTime</b>	<b>Int</b>	-	<b>60</b>	
Lifetime before batch expires. Needs to be expressed in Minutes. Minimum value expected = 60. For example, maxLifeTime = 1800 sets it to 1 Day & 6 Hours.				

### *Recipe Steps*

Each Recipe can have zero or many Recipe Steps linked to it.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>active</b>	<b>int</b>	-	<b>1</b>	
Indicates whether this Recipe Step is Active or not (0 – No, 1 – Yes).				
<b>answer 1-4</b>	<b>string</b>	<b>25</b>		
Allows the input of 4 potential answers if this Step is determined to be a Question.				
<b>code</b>	<b>string</b>	<b>35</b>	<b>TESTING002</b>	
The Code of the Commodity for this Recipe. If this does not exist, then a commodity will be created using this value as both it's Code and Name.				
<b>delay</b>	<b>double</b>	-	<b>0</b>	
The number of seconds that should the system should pause for before this Recipe Step is presented to an Operator.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Reason is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>freeAnswer</b>	<b>int</b>	-	<b>0</b>	
Used when creating a Question step. This indicates whether an Operator can type a 'Free text' answer, or whether they must select a pre-set answer (0 – No, 1 – Yes).				
<b>instruction</b>	<b>string</b>	<b>50</b>	<b>PUT GLOVES ON</b>	
An instruction that should be presented to the Operator when dealing with this Recipe Step.				

<b>XML NODE</b>	<b>FIELD TYPE</b>	<b>FIELD LENGTH</b>	<b>EXAMPLE</b>	<b>REQUIRED</b>
<b>instructionDisplayTime</b>	<b>int</b>	-	<b>10</b>	
The amount of time (in seconds) that an Instruction should be displayed.				
<b>lowerTolerance</b>	<b>double</b>	-	<b>0.05</b>	
The Lower Tolerance for this Recipe Step.				
<b>name</b>	<b>string</b>	<b>255</b>	<b>STS PRODUCT 1</b>	<b>TRUE</b>
The name of the commodity. If the element is omitted, the name will be derived from the commodity code instead.				
<b>negativeWeighing</b>	<b>int</b>	-	<b>0</b>	
Specifies whether this Recipe Step should be determined as a Negative Weigh step (0 – No, 1 – Yes).				
<b>platform</b>	<b>int</b>	-	<b>1</b>	
Dictates which Platform should be used when weighing this Recipe Step.				
<b>productionLocationCode</b>	<b>string</b>	<b>20</b>	<b>RAW STOCK</b>	
Indicates which ProductionLocation should be used to execute this Recipe Step.				
<b>question</b>	<b>string</b>	<b>255</b>		
If this Recipe Step is a Question, then the text to be displayed must be entered here.				
<b>scaleTargetQuantity</b>	<b>int</b>	-		
<b>scaleTolerance</b>	<b>int</b>	-		
<b>schedule</b>	<b>int</b>	-	<b>1</b>	
Indicates whether this Recipe Step should be scheduled automatically when required.				
<b>section</b>	<b>string</b>	<b>255</b>		

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>sequenceNumber</b>	<b>int</b>	-		
Specifies the order of Recipe Steps. If no value is provided, then the order the Recipe Steps are listed in the import file will be used.				
<b>targetQuantity</b>	<b>double</b>	-	<b>50.0</b>	
The Target Quantity for this Recipe Step.				
<b>temperatureUnit</b>	<b>int</b>	-		
Specifies which unit of Temperature should be used (0 – Celsius, 1 – Fahrenheit, 2 – Kelvin)				
<b>type</b>	<b>int</b>	-	<b>0</b>	<b>TRUE</b>
<p><b>Indicates the Recipe Step type. This must be one of the following:</b></p> <p>(0 – Weigh On Weigh Scale), (1 – Non_Weighed), ([2 – Message], {3 – Submix}, (4 - Change Container), (5 – Switch Scale 1), (6 – Switch Scale 2), (7 – Question), (8 – Discrete Quantity), (9 – Proportional Submix), (10 – Possible Ingredient), (11 – Timed Ingredient), (12 – Scaleable Ingredient), (13 -Weigh On Scale Only), (14 – Use Packagaing), (15 – Main Portion), (16 – Add Container), (17 – Premix), (19- Temperature Probe), (99 – Collate Submix).</p>				
<b>upperTolerance</b>	<b>double</b>	-	<b>0.05</b>	
The Lower Tolerance for this Recipe Step.				

#### Tolerances

If no tolerance nodes are provided and the recipe step does not already exist, then first the commodity is checked to see if global tolerances apply if not then configuration values are used to determine tolerances. If no tolerance nodes are provided and the recipe step already exists then the database values are retained.

## Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <recipes>
    <recipe>
      <addToStock>1</addToStock>
      <code>EXAMPLERECIPE</code>
      <created />
      <defaultMixSize />
      <deleted />
      <labelField1 />
      <labelField20 />
      <maxProducts>0</maxProducts>
      <minProducts>0</minProducts>
      <name />
      <premix>0</premix>
      <productionLocationCode>ANY</productionLocationCode>
      <productPerMinute />
      <requireSignOff>0</requireSignOff>
      <stockProductionLocationCode>POSITIVE RELEASE</stockProductionLocationCode>
      <tareWeight>0</tareWeight>
      <type>0</type>
      <weighProduction>0</weighProduction>
      <yield>1</yield>
      <maxLifeTime>1800</maxLifeTime>
      <recipeSteps>
        <recipeStep>
          <active>1</active>
          <answer1 />
          <answer4 />
          <commodityCode>WF001</commodityCode>
          <delay>0</delay>
          <deleted />
          <freeAnswer>0</freeAnswer>
          <instruction>Add Flour To Mix</instruction>
          <instructionDisplayTime>0</instructionDisplayTime>
          <lowerTolerance>0.5</lowerTolerance>
          <name>White Flour</name>
          <negativeWeighing>0</negativeWeighing>
          <platform>0</platform>
          <productionLocationCode />
          <question />
          <scaleTargetQuantity>1</scaleTargetQuantity>
          <scaleTolerance>1</scaleTolerance>
          <schedule>0</schedule>
          <section />
          <sequenceNumber>1</sequenceNumber>
          <targetQuantity>50</targetQuantity>
          <temperatureUnit />
          <type>0</type>
          <upperTolerance>0.5</upperTolerance>
        </recipeStep>
      </recipeSteps>
    </recipe>
  </recipes>
</Tracesense>
```

---

## REQUEST - ALLFIELDS.XML

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <requests>
    <request>
      <datatype>[A]</datatype>
      <criteria>
        <code condition=[B]></code>
        <name />
        <descriptionsingular />
        <descriptionplural />
        <fromdate />
        <todate />
        <ordernumber />
        <suppliercode />
        <customercode />
        <status />
        <orderby />
        <receivedstart />
        <receivedend />
        <expirestart />
        <expireend />
        <limit />
        <text />
        <deleted />
        <minimumstocklevel />
      </criteria>
    </request>
  </requests>
</Tracesense>
```

---

The 'datatype' node [A] must always be populated. It can contain any of the following options:

*COMMODITY, CUSTOMER, LOCATION, PRODUCEDHEADER, PURCHASEORDER, QUESTION, REASON, RECIPE, SALESORDER, STOCK, STOCKLOCATION, STOCKUNIT, SUPPLIER, USER.*

Each 'Criteria' node [B] must contain the 'condition' attribute. The 'condition' attribute can contain any of the following options:

ASCENDING, CONTAINS, DESCENDING, EQUALS, GREATERTHAN, LESSTHAN, INCLUSIVE, STARTSWITH

## REQUEST - ALL STOCK BELOW LEVEL.XML

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <requests>
    <request>
      <datatype>STOCK</datatype>
      <criteria>
        <code></code>
        <orderby condition="descending">NAME</orderby>
        <receivedstart />
        <receivedend />
        <expirestart />
        <expireend />
        <limit />
        <deleted />
        <minimumstocklevel condition="lessthan">10</minimumstocklevel>
      </criteria>
    </request>
  </requests>
</Tracesense>
```

---

This will return a STOCKEXPORT file, containing only those entries with a stock level of less than 10.

## REQUEST - ALL STOCK FOR NAMED COMMODITY.XML

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <requests>
    <request>
      <datatype>STOCK</datatype>
      <criteria>
        <code condition="equals">TESTING01</code>
        <orderby condition="ascending">CODE</orderby>
        <receivedstart />
        <receivedend />
        <expirestart />
        <expireend />
        <limit />
        <deleted />
        <minimumstocklevel />
      </criteria>
    </request>
  </requests>
</Tracesense>
```

---

This will return a STOCKEXPORT file, containing only those stock entries where the Commodity Code is equal to "TESTING01", ordered by the 'code' node.



## REQUEST-ALLSTOCKPOSITIVE.XML

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <requests>
    <request>
      <datatype>STOCK</datatype>
      <criteria>
        <code></code>
        <orderby condition="ascending">CODE</orderby>
        <receivedstart />
        <receivedend />
        <expirestart />
        <expireend />
        <limit />
        <deleted />
        <minimumstocklevel condition="greaterthan">0</minimumstocklevel>
      </criteria>
    </request>
  </requests>
</Tracesense>
```

---

This will return a STOCKEXPORT file, containing only those stock entries where the current stock level is greater than '0', ordered by the 'code' node (ascending).

## REQUEST-ALLSTOCKUNEXPIRED.XML

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <requests>
    <request>
      <datatype>STOCK</datatype>
      <criteria>
        <code></code>
        <orderby condition="ascending">CODE</orderby>
        <receivedstart />
        <receivedend />
        <expirestart condition="greaterthan">27/10/2017</expirestart>
        <expireend />
        <limit />
        <deleted />
        <minimumstocklevel condition="greaterthan">0</minimumstocklevel>
      </criteria>
    </request>
  </requests>
</Tracesense>
```

---

This will return a STOCKEXPORT file, containing only those stock entries where the current stock expiry date is greater than '27/10/2017' and the current stock level is greater than '0', ordered by the 'code' node (ascending).

## REQUEST - RECIPEBYCODE.XML

---

```
< ?xml version="1.0" encoding="utf-8"? >
<Tracesense>
  <requests>
    <request>
      <datatype>RECIPE</datatype>
      <criteria>
        <code condition="equals">WB-001</code>
        <orderby />
        <limit />
        <deleted />
      </criteria>
    </request>
  </requests>
</Tracesense>
```

---

This will return a RECIPEEXPORT file, containing only the recipe with a code that equals 'WB-001'.

## REQUIREMENT

A Requirement Import provides the functionality to create or update Requirements in the TracesenseSystems. An export provides details of newly created/edited Requirements.

### Elements

The following table details the structure of a Requirement XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>batchCount</b>	<b>int</b>	-	<b>1</b>	
The number of Batches required.				
<b>commodityCode</b>	<b>string</b>	<b>35</b>	<b>TESTREC1</b>	<b>TRUE</b>
The code of the Commodity that is required to be produced.				
<b>created</b>	<b>string*</b>	-	<b>01/12/2018</b>	
The Date the requirement was created. If no value is given, then this will default to the current date and time.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Reason is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>jobNumber</b>	<b>string</b>	<b>22</b>	<b>JOB-01</b>	
The Job Number for this Requirement.				
<b>mixSize</b>	<b>double</b>	-	<b>100</b>	
The required Mixsize for this Requirement.				
<b>notes</b>	<b>string</b>	<b>40</b>	<b>URGENT JOB</b>	
Any notes that should be applied to this Requirement.				
<b>productCount</b>	<b>double</b>	-	<b>250</b>	
The number of Products that are required.				
<b>productionDate</b>	<b>string*</b>	-	<b>03/12/2018</b>	
The date that the production for this Requirement is to be scheduled.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
schedule	boolean	5	FALSE	
Determines whether this Requirement should be automatically scheduled.				
shift	string	255		
Any notes that should be shown to the shift completing the requirement.				

\* This value must be in the same Date Format as determined by the related Integration Configuration.

*Example File*

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <requirements>
    <requirement>
      <batchCount />
      <commodityCode />
      <created>
      <deleted />
      <jobNumber />
      <mixSize />
      <notes />
      <productCount />
      <productionDate />
      <schedule />
      <shift />
    </requirement>
  </requirements>
</Tracesense>
```

**NOTE:** On Requirements Import, Entry Types are treated on priority bases if all elements are provided. As following

- 1- Batch Count & Mix Size
- 2- Products
- Mix Size
- Batch Count

So, if (Products) element is provided with (Batch Count & Mix Size), IS will ignore (Products) and use (Batch & Mix), Similarly if (Products) element is provided with either (Batch Count) OR (Mix Size), IS will ignore (Batch Count / Mix Size) and use (Products) instead.

## SALESORDER

A Sales Order Import provides the functionality to create or update Sales Orders in the Tracesense Systems. An export provides details of newly created/edited Sales Orders and gives details of any dispatches that may have been completed.

### Elements

The following table details the structure of a Sales Order XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>customerCode</b>	string	40	CUSTOMER1	TRUE
The Date the Purchase Order was Created.				
<b>customerReference</b>	string	30	TRACESENSE01	TRUE
The Customer Reference for this Sales Order.				
<b>deliveryAddress</b>	string	255		
The Address at which the Sales Order should be dispatched too.				
<b>deliveryDate</b>	string*	-	03/03/2018	
The Date that the Sales Order should be delivered.				
<b>orderDate</b>	string*	-	01/03/2018	
The Date the Sales Order was created.				
<b>orderNumber</b>	string	255	PO-001	TRUE
The unique Order Number for the Purchase Order.				
<b>user</b>	string	25	DANOTHER	
The Code of the User who took this order.				

\* This value must be in the same Date Format as determined by the related Integration Configuration.

### Sales Order Lines

Each Sales Order can have zero or many Sales Order Lines linked to it. When exporting a Sales Order, it is possible to configure the system so that it only exports Sales Order Lines that have been dispatched, a full export, or only an export on completion.

<b>XML NODE</b>	<b>FIELD TYPE</b>	<b>FIELD LENGTH</b>	<b>EXAMPLE</b>	<b>REQUIRED</b>
<b>commodityCode</b>	<b>string</b>	<b>35</b>	<b>TESTREC01</b>	<b>TRUE</b>
The Code of the Commodity for this Sales Order Line.				
<b>customerField1-10</b>	<b>string</b>	<b>255</b>		
There are 10 x Customer Fields that can be populated with any string value as required.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Reason is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>denominationType</b>	<b>int</b>	<b>1</b>	<b>0</b>	
A value determining the Denomination Type of this Sales Order Line.				
<b>dispatchedOutstanding</b>	<b>double</b>	<b>-</b>	<b>0.0</b>	
The total quantity outstanding.				
<b>dispatchedQuantity</b>	<b>double</b>	<b>-</b>	<b>0.0</b>	
The total quantity that has been dispatched.				
<b>dispatchedRequired</b>	<b>double</b>	<b>-</b>	<b>0.0</b>	
The remaining quantity that is required to be dispatched.				
<b>instruction</b>	<b>double</b>	<b>-</b>	<b>5.00</b>	
An instruction for this Sales Order Line.				
<b>printLabel</b>	<b>int</b>	<b>-</b>		
Used to determine whether a label should be printed when the Sales Order Line is marked as dispatched.				
<b>schedule</b>	<b>Boolean</b>	<b>5</b>	<b>FALSE</b>	

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
Used to create Requirement on Sales Order Import.				

### Example File

```

<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <salesOrders>
    <salesOrder>
      <customerCode>A Generic Customer</customerCode>
      <customerReference>TRACESENSE01</customerReference>
      <deliveryAddress>1 Sales Road, Blackburn, Lancashire, BB5 7NQ</deliveryAddress>
      <deliveryDate>42804</deliveryDate>
      <orderDate>42795</orderDate>
      <orderNumber>SALES0001</orderNumber>
      <user>RE001</user>
      <salesOrderLines>
        <salesOrderLine>
          <commodityCode>TESTREC01</commodityCode>
          <customerField1 />
          <customerField2 />
          <customerField3 />
          <customerField4 />
          <customerField5 />
          <customerField6 />
          <customerField7 />
          <customerField8 />
          <customerField9 />
          <customerField10 />
          <deleted />
          <denominationType>0</denominationType>
          <dispatchedOutstanding>0</dispatchedOutstanding>
          <dispatchedRequired>100</dispatchedRequired>
          <dispatchedQuantity>100</dispatchedQuantity>
          <instruction>An Instruction</instruction>
          <schedule>>false</schedule>
          <printLabel />
        </salesOrderLine>
      </salesOrderLines>
    </salesOrder>
  </salesOrders>
</Tracesense>

```

## Example File – Sales Order Fail

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <commodities />
  <customers />
  <locations />
  <purchaseOrders />
  <reasons />
  <recipes />
  <salesOrders>
    <salesOrder>
      <customerCode>BBEE</customerCode>
      <customerReference>B338</customerReference>
      <deliveryAddress />
      <deliveryDate>05/02/18</deliveryDate>
      <orderDate>05/02/18</orderDate>
      <orderNumber>A100</orderNumber>
      <user />
      <salesOrderLines>
        <salesOrderLine>
          <commodityCode>IS2TEST01</commodityCode>
          <customerField1 />
          <customerField2 />
          <customerField3 />
          <customerField4 />
          <customerField5 />
          <customerField6 />
          <customerField7 />
          <customerField8 />
          <customerField9 />
          <customerField10 />
          <deleted />
          <denominationType />
          <dispatchedOutstanding>0</dispatchedOutstanding>
          <dispatchedRequired>0</dispatchedRequired>
          <dispatchedQuantity>0</dispatchedQuantity>
          <instruction />
          <schedule>false</schedule>
          <printLabel />
        </salesOrderLine>
      </salesOrderLines>
      <failure>
        <message>Object Validation Error</message>
        <cause>The SalesOrderHeader Being Imported Failed Validation</cause>
        <errors>
          <error>The [User] Element Failed Validation.'User' must not be empty.</error>
        </errors>
      </failure>
    </salesOrder>
  </salesOrders>
  <sites />
  <stockHeaders />
  <stockLocations />
  <stockUnits />
  <suppliers />
  <users />
</Tracesense>
```

---



## STOCK

A Stock Import provides the functionality to create or update Stock in the Tracesense Systems. An export provides details of newly created/edited Stock.

### Elements

The following table details the structure of a Stock XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>bagWeight</b>	<b>double</b>	-	<b>25</b>	
The bag weight of this Stock entry.				
<b>commodityCode</b>	<b>string</b>	<b>35</b>	<b>TESTING02</b>	<b>TRUE</b>
The code of the Commodity that this Stock relates too.				
<b>cost</b>	<b>double</b>	-	<b>2.00</b>	
The cost of the Stock.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Reason is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>expiryDate</b>	<b>string*</b>	-	<b>21/02/2019</b>	
The Date that this Stock will expire. If no value is provided, then the expiry date will be calculated using the 'noOfExpiryDays' value stored against the Commodity.				
<b>lotNumber</b>	<b>string</b>	<b>30</b>	<b>13/271017</b>	<b>TRUE</b>
The unique reference to this Stock.				
<b>receivedDate</b>	<b>string*</b>	-	<b>10/02/2018</b>	
The date the Stock was received.				
<b>receivedWeight</b>	<b>double</b>	-	<b>2525.00</b>	
The total weight of the received Stock.				
<b>siteld</b>	<b>int</b>	-		

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>supplierCode</b>	<b>string</b>	<b>10</b>	<b>GENERIC</b>	
The code of the Supplier that this Stock was received from.				
<b>supplierName</b>	<b>string</b>	<b>50</b>	<b>GENERIC SUPPLIER</b>	
The name of the Supplier that this Stock was received from.				
<b>totalQuantity</b>	<b>double</b>	<b>-</b>	<b>2525.00</b>	
The total quantity of stock held within this LOT. This a sum of the values found within the related <stockDetails> nodes.				

\* This value must be in the same Date Format as determined by the related Integration Configuration.

#### StockDetails

Each Stock Header can have one or many StockDetails linked to it.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the commodity is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>stockLocation</b>	<b>string</b>	<b>50</b>	<b>POSITIVE RELEASE</b>	
The Stock Location Code where this Stock is located.				
<b>quantity</b>	<b>double</b>	<b>-</b>	<b>2525.00</b>	<b>TRUE</b>
The quantity of stock found in this Stock Detail.				

## Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockHeaders>
    <stockHeader>
      <bagWeight>25</bagWeight>
      <commodityCode>TESTING02</commodityCode>
      <cost>2.00</cost>
      <deleted />
      <expiryDate>02/10/2021</expiryDate>
      <lotNumber>13/271017</lotNumber>
      <receivedDate>27/10/2017</receivedDate>
      <receivedWeight>2525</receivedWeight>
      <siteId />
      <supplierCode>GENERIC</supplierCode>
      <supplierName>Generic Supplier</supplierName>
      <totalQuantity>2,525.00</totalQuantity>
      <stockDetails>
        <stockDetail>
          <deleted />
          <quantity>2,525.00</quantity>
          <stockLocation>POSITIVE RELEASE</stockLocation>
        </stockDetail>
      </stockDetails>
      <supplierBatchNumbers>
        <supplierBatchNumber>
          <number>SUP-Batch-1</number>
        </supplierBatchNumber>
      </supplierBatchNumbers>
    </stockHeader>
  </stockHeaders>
</Tracesense>
```

---

## STOCK LOCATION

A Stock Location Import provides the functionality to create or update Stock Locations in the TracesenseSystems. An export provides details of newly created/edited Stock Locations.

### Elements

The following table details the structure of a Stock Location XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>code</b>	<b>string</b>	<b>50</b>	<b>FRZ</b>	<b>TRUE</b>
The Stock Location Code.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Stock Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>name</b>	<b>string</b>	<b>28</b>	<b>FREEZER</b>	<b>TRUE</b>
The Name of the Stock Location				
<b>returnsLocation</b>	<b>int</b>	<b>-</b>		
<b>type</b>	<b>string</b>	<b>255</b>		
The type of Stock Location. This can be any value, such as “MAIN STOCK” or “DEMO STOCK”.				

### Example File

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockLocations>
    <stockLocation>
      <code>FREEZER</code>
      <deleted />
      <name>Freezer</name>
      <returnsLocation />
      <type />
    </stockLocation>
  </stockLocations>
</Tracesense>
```

## STOCK UNIT

A Stock Unit Import provides the functionality to create or update Stock Units in the Tracesense Systems. Anexport provides details of newly created/edited Stock Units.

### Elements

The following table details the structure of a Stock Unit XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>conversionFactor</b>	<b>double</b>	-		
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Stock Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>divisible</b>	<b>int</b>	-		
Specifies whether this Stock Unit is divisible (0 – It is not divisible, 1 – It is divisible).				
<b>massVolume</b>	<b>int</b>	-		
<b>singularDescription</b>	<b>string</b>	<b>10</b>	<b>KG</b>	<b>TRUE</b>
The Singular Description for this Stock Unit				
<b>pluralDescription</b>	<b>string</b>	<b>10</b>	<b>KGS</b>	<b>TRUE</b>
The Plural Description for this Stock Unit				

### Example File

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockUnits>
    <stockUnit>
      <conversionFactor />
      <deleted />
      <divisible>1</divisible>
      <massVolume>0</massVolume>
      <pluralDescription>ROLL</pluralDescription>
      <singularDescription>ROLL</singularDescription>
    </stockUnit>
  </stockUnits>
</Tracesense>
```

---

## SUPPLIER

A Supplier Import provides the functionality to create or update Suppliers in the Tracesense Systems. Anexport provides details of newly created/edited Suppliers.

### Elements

The following table details the structure of a Supplier XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>address</b>	<b>string</b>	<b>150</b>	<b>1 Supplier St, Bury</b>	
The Supplier Address.				
<b>code</b>	<b>string</b>	<b>10</b>	<b>SUPPLIER1</b>	<b>TRUE</b>
The Code for this Supplier.				
<b>contactEmail</b>	<b>string</b>	<b>70</b>	<b>Sup1@supplier.com</b>	
The Email Address for this Supplier.				
<b>contactName</b>	<b>string</b>	<b>50</b>	<b>Mrs A Supplier</b>	
The Supplier Contact Name.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Stock Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>fax</b>	<b>string</b>	<b>20</b>	<b>01254 685201</b>	
A value representing the fax contact number for this Customer				
<b>name</b>	<b>string</b>	<b>50</b>	<b>TEST SUPPLIER 1</b>	<b>TRUE</b>
The Name for this Supplier.				
<b>notes</b>	<b>string</b>	<b>250</b>	<b>A TEST NOTE</b>	
A Note that should be stored against this Supplier.				
<b>telephone</b>	<b>string</b>	<b>20</b>	<b>01254 685200</b>	
The main Telephone Number for this Supplier.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
website	string	50	www.Tracesense.com	
The Website URL of this Supplier.				

*Example File*

```

<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <suppliers>
    <supplier>
      <address>Supplier St,Supplier</address>
      <code>SUPPLIER1</code>
      <contactEmail>supplier@tracesense.com</contactEmail>
      <contactName>Mr Supplier</contactName>
      <deleted />
      <fax />
      <name>Test Supplier 1</name>
      <notes />
      <telephone>01254 685200</telephone>
      <website />
    </supplier>
  </suppliers>
</Tracesense>

```



## TRANSACTION

A Transaction Import provides the functionality to perform Transactions on Stock within the Tracesense Systems. A Transaction Export will be created when stock is modified in some way: Adjustment, Cost Adjustment, Expiry Change, Move, Receive or Withdraw.

The XML created has the same structure regardless of the transaction type. Only elements related to the transaction type will hold a value. For example, a 'Move Transaction Export' will not populate the 'despatchDate' element. Please view the 'Example Files' for further assistance.

### Elements

The following table details the structure of a Stock Unit XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>commodityCode</b>	<b>string</b>	<b>35</b>	<b>WF001</b>	<b>TRUE</b>
The Commodity Code that represents the Stock referenced.				
<b>despatchDate</b>	<b>string*</b>	<b>-</b>	<b>01/01/2018</b>	
A value providing the date that this transaction was despatched.				
<b>expiryDate</b>	<b>string*</b>	<b>-</b>	<b>01/01/2018</b>	
Specifies the Expiry Date of the Stock referenced.				
<b>fromCost</b>	<b>double</b>	<b>-</b>	<b>10.00</b>	
The old Cost value of the Stock referenced.				
<b>fromDate</b>	<b>string*</b>	<b>-</b>	<b>01/01/2018</b>	
The old Expiry Date value of the Stock referenced.				
<b>fromLocation</b>	<b>string</b>	<b>50</b>	<b>RAW STOCK</b>	
The old Location value of the Stock referenced.				
<b>fromQuantity</b>	<b>double</b>	<b>-</b>	<b>10.00</b>	
The old Quantity value of the Stock referenced.				
<b>lotNumber</b>	<b>string</b>	<b>30</b>	<b>LOT01/22022108</b>	
The lot Number that represents the Stock referenced.				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>purchaseOrderNumber</b>	<b>string</b>	<b>255</b>	<b>ORDER010203</b>	
The Purchase Order Order Number of the Stock Referenced.				
<b>reason</b>	<b>string</b>	<b>60</b>	<b>LOST</b>	
The Reason as to why this Transaction has occurred.				
<b>receivedDate</b>	<b>string*</b>	<b>-</b>	<b>05/03/2018</b>	
The Date the Stock referenced was Received.				
<b>salesOrderNumber</b>	<b>string</b>	<b>255</b>	<b>CUST01-255</b>	
The Sales Order Order Number of the Stock Referenced.				
<b>timestamp</b>	<b>string*</b>	<b>-</b>	<b>22/02/2018</b>	<b>TRUE</b>
The Date/Time that this Transaction was executed.				
<b>toCost</b>	<b>double</b>	<b>-</b>	<b>10.00</b>	
The new Cost value of the Stock referenced.				
<b>toDate</b>	<b>string*</b>	<b>-</b>	<b>01/01/2018</b>	
The new Expiry Date value of the Stock referenced.				
<b>toLocation</b>	<b>string</b>	<b>50</b>	<b>RAW STOCK</b>	
The new Location value of the Stock referenced.				
<b>toQuantity</b>	<b>double</b>	<b>-</b>	<b>10.00</b>	
The new Quantity value of the Stock referenced.				
<b>transationType</b>	<b>string</b>	<b>-</b>	<b>STOCK MOVE</b>	<b>TRUE</b>
Details the type of Transaction. Must be equal to one of the following values: [STOCK ADJUSTMENT, COST ADJUSTMENT, STOCK EXPIRY CHANGE, STOCK MOVE, STOCK RECEIVE, STOCK WITHDRAW].				

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
quantity	double	-	20.00	
The quantity of stock affected by this Transaction.				

This value must be in the same Date Format as determined by the related Integration Configuration.

**Required Fields** In the examples below, required fields are highlighted in grey for particular transactiontype.

*Example File – Stock Receive*

[This export can be enabled/disabled as required]

```

<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockTransactions>
    <stockTransaction>
      <commodityCode>TESTING01</commodityCode>
      <despatchDate />
      <expiryDate>22/02/2022</expiryDate>
      <fromCost />
      <fromDate />
      <fromLocation />
      <fromQuantity />
      <lotNumber>TEST-LOT-111</lotNumber>
      <purchaseOrderNumber />
      <reason />
      <receivedDate />
      <salesOrderNumber />
      <timestamp>03/07/2019</timestamp>
      <toCost />
      <toDate />
      <toLocation>FREEZER</toLocation>
      <toQuantity />
      <transactionType>STOCK RECEIVE</transactionType>
      <quantity>111.000</quantity>
    </stockTransaction>
  </stockTransactions>
</Tracesense>

```

## Example File – Stock Withdraw

[This export can be enabled/disabled as required], Quantity can only be set to negative for stock withdraw.

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockTransactions>
    <stockTransaction>
      <commodityCode>TESTING01</commodityCode>
      <despatchDate />
      <expiryDate />
      <fromCost />
      <fromDate />
      <fromLocation>FREEZER</fromLocation>
      <fromQuantity />
      <lotNumber>ING01-LOT1</lotNumber>
      <purchaseOrderNumber />
      <reason />
      <receivedDate />
      <salesOrderNumber />
      <timestamp>03/07/2019</timestamp>
      <toCost />
      <toDate />
      <toLocation />
      <toQuantity />
      <transactionType>STOCK WITHDRAW</transactionType>
      <quantity>-74.000</quantity>
    </stockTransaction>
  </stockTransactions>
</Tracesense>
```

## Example File – Adjustment Stock Level

[This export can be enabled/disabled as required]

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockTransactions>
    <stockTransaction>
      <commodityCode>WF001</commodityCode>
      <despatchDate />
      <expiryDate />
      <fromCost />
      <fromDate />
      <fromLocation>Freezer02</fromLocation>
      <fromQuantity>250.00</fromQuantity>
      <lotNumber>LOT01</lotNumber>
      <purchaseOrderNumber />
      <reason>Lost</reason>
      <receivedDate />
      <salesOrderNumber />
      <timestamp>29/09/2017</timestamp>
      <toCost />
      <toDate />
      <toLocation />
      <toQuantity>200.00</toQuantity>
      <transactionType>STOCK ADJUSTMENT</transactionType>
      <quantity>-50.00</quantity>
    </stockTransaction>
  </stockTransactions>
</Tracesense>
```

## Example File – Adjustment Cost

[This export can be enabled/disabled as required]

---

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockTransactions>
    <stockTransaction>
      <commodityCode>WF001</commodityCode>
      <despatchDate />
      <expiryDate />
      <fromCost>1.00</fromCost>
      <fromDate />
      <fromLocation>Fridge01</fromLocation>
      <fromQuantity />
      <lotNumber>LOT01</lotNumber>
      <purchaseOrderNumber />
      <reason>Unknown</reason>
      <receivedDate />
      <salesOrderNumber />
      <timestamp>29/09/2017</timestamp>
      <toCost>3.50</toCost>
      <toDate />
      <toLocation />
      <toQuantity />
      <transactionType>COST ADJUSTMENT</transactionType>
      <quantity />
    </stockTransaction>
  </stockTransactions>
</Tracesense>
```

---

## Example File – Expiry Change

[This export can be enabled/disabled as required]

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockTransactions>
    <stockTransaction>
      <commodityCode>WF001</commodityCode>
      <despatchDate />
      <expiryDate />
      <fromCost />
      <fromDate>16/09/2017</fromDate>
      <fromLocation>Freezer02</fromLocation>
      <fromQuantity />
      <lotNumber>LOT01</lotNumber>
      <purchaseOrderNumber />
      <reason>Unknown</reason>
      <receivedDate />
      <salesOrderNumber />
      <timestamp>29/09/2017</timestamp>
      <toCost />
      <toDate>25/12/2017</toDate>
      <toLocation />
      <toQuantity />
      <transactionType>STOCK EXPIRY CHANGE</transactionType>
      <quantity />
    </stockTransaction>
  </stockTransactions>
</Tracesense>
```

## Example File – Move

[This export can be enabled/disabled as required]

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <stockTransactions>
    <stockTransaction>
      <commodityCode>WF001</commodityCode>
      <despatchDate />
      <expiryDate />
      <fromCost />
      <fromDate />
      <fromLocation>Fridge01</fromLocation>
      <fromQuantity />
      <lotNumber>LOT01</lotNumber>
      <purchaseOrderNumber />
      <reason />
      <receivedDate />
      <salesOrderNumber />
      <timestamp>29/09/2017</timestamp>
      <toCost />
      <toDate />
      <toLocation>Production</toLocation>
      <toQuantity />
      <transactionType>STOCK MOVE</transactionType>
      <quantity>100.00</quantity>
    </stockTransaction>
  </stockTransactions>
</Tracesense>
```

## USER

A User Import provides the functionality to create or update Users in the Tracesense Systems. An export provides details of newly created/edited Users.

### Elements

The following table details the structure of a User XML file.

XML NODE	FIELD TYPE	FIELD LENGTH	EXAMPLE	REQUIRED
<b>code</b>	<b>string</b>	<b>25</b>	<b>AUSER</b>	<b>TRUE</b>
The Code for this User.				
<b>name</b>	<b>string</b>	<b>40</b>	<b>Mr A User</b>	<b>TRUE</b>
The Users name.				
<b>deleted</b>	<b>boolean</b>	<b>5</b>	<b>FALSE</b>	
A value determining whether the Stock Location is marked as deleted (FALSE - not deleted, TRUE – deleted).				
<b>password</b>	<b>string</b>	<b>260</b>		
The user password. This value will be encrypted and stored securely.				

### Example File

```
<?xml version="1.0" encoding="utf-8"?>
<Tracesense>
  <users>
    <user>
      <code>AN001</code>
      <deleted />
      <name>Andrew North</name>
      <password>AN01$</password>
    </user>
  </users>
</Tracesense>
>
```

## EXAMPLE CSV HEADERS

If the CSV file does not contain a header line, and the system cannot find a related '.csvh' file in the 'CSVHeaders' directory, then the corresponding header below will be used. Any custom '.csvh' files must only use elements found in the examples that follow.

### COMMODITY

"CODE", "NAME", "TYPE", "UNIT", "COSTPERUNIT", "DEFAULTBAGWEIGHT",  
"DEFAULTEXPIRYDAYS", "PRODUCTOFCOMMODITY", "SCALE", "BARCODE", "UPC", "GTIN",  
"NONSTOCKITEM", "ALLERGENWARNING", "NOTES", "INSTRUCTIONS", "LOTINCREMENT",  
"BAGMODE", "DENOMINATIONTYPE", "BAGNAME", "DEFAULTSTOCKLOCATION", "STRICTLOCATION",  
"CASEQUANTITY", "SDFACTOR", "LASTSTOCKIN", "FILLERNUMBER", "TARE",  
"GLOBALSTOCKDETID", "FORCECHANGELOTWEIGHT", "SUPERLOTID", "REORDERWEIGHT",  
"REORDERQUANTITY", "BUDGETCOST", "LABELPRINTERTYPEID", "MAXPALLETQUANTITY",  
"ACCOUNTLINK", "SYNCSINSTRUCTIONS", "DRIBBLEWEIGHT", "INFLIGHTWEIGHT",  
"INFLIGHTRECALC", "DELETED", "VALIDUNTIL"

### CONTAINER

"CODE", "NAME", "DESCRIPTION", "LOCATION", "TYPE", "GROUP", "TARE"

### CUSTOMER

"CODE", "NAME", "INVOICEADDRESS", "TELEPHONE", "MOBILE", "FAX", "WEBSITE",  
"CONTACTNAME", "CONTACTEMAIL", "SHORTESTUSEBY", "SPECIALINSTRUCTIONS", "DELETED",  
"SITEID", "FREQUENCY", "NEXTVISIT", "SETTINGS", "CUSTOMERGROUP", "NOTES",  
"ADDRESSCODE", "ADDRESS", "ADDRESSDELETED"

### LOCATION

"CODE", "DESCRIPTION", "LOCATIONTYPE", "AUTOLOGOFFDELAY", "DELETED",  
"PORTIONSCALEONLY"

### PURCHASE ORDER

"ORDERNUMBER", "SUPPLIERCODE", "ORDERREFERENCE", "DELIVERYDATE", "CREATED",  
"DELETED", "SITEID", "COMMODITYCODE", "ORDEREDQUANTITY", "COST", "UNIT"

### REASON

"NAME", "DELETED"

### RECIPE

"RECIPECODE", "RECIPENAME", "COMMODITYCODE", "DELETED", "INSTRUCTION",  
"PRODUCTIONLOCATIONCODE", "STOCKPRODUCTIONLOCATIONCODE", "YIELD",  
"DEFAULTMIXSIZE", "MINPRODUCTS", "MAXPRODUCTS", "ADDSTOCK", "PRODUCTPERMINUTE",  
"REQUIRESIGNOFF", "WEIGHPRODUCTION", "TAREWEIGHT", "TYPE", "PREMIX",  
"LABELFIELD 1", "LABELFIELD 2", "LABELFIELD3", "LABELFIELD4", "LABELFIELD5",  
"LABELFIELD 6", "LABELFIELD 7", "LABELFIELD8", "LABELFIELD9", "LABELFIELD10",  
"LABELFIELD11", "LABELFIELD12", "LABELFIELD13", "LABELFIELD14", "LABELFIELD15",  
"LABELFIELD16", "LABELFIELD17", "LABELFIELD18", "LABELFIELD19", "LABELFIELD20",  
"MESSAGELISTID", "MAXLIFETIME", "LABELPRINTERTYPEID", "TARGETQUANTITY", "UNITS",  
"UPPERTOLERANCE", "LOWERTOLERANCE", "SCHEDULE", "STEPINSTRUCTION",  
"INSTRUCTIONDISPLAYTIME", "STEPTYPE", "NONWEIGHED", "SCALETARGETQUANTITY",  
"SCALETOLERANCE", "QUESTION", "ANSWER1", "ANSWER2", "ANSWER3", "ANSWER4",



"ANSWERA", "ANSWERB", "ANSWERC", "ANSWERD", "FREEANSWER", "TEMPERATUREUNIT",  
"LOWERLOWTOLERANCE", "UPPERHIGHTOLERANCE"

## REQUIREMENT

"JOBNUMBER", "COMMODITYCODE", "DELETED", "CREATED", "BATCHESREQUIRED", "MIXSIZE", "PRODUCTS",  
"NOTES", "PRODUCTIONDATE", "SHIFT", "SCHEDULE"

## SALES ORDER

"ORDERNUMBER", "DATEORDERPLACED", "CUSTOMERCODE", "CUSTOMERREFERENCE",  
"DELIVERYDATE", "DELIVERYADDRESS", "STATUS", "USERCODE", "DELETED",  
"DATECOMPLETED", "SPECIALINSTRUCTIONS", "PSTATUS", "DELIVERYROUTEID", "URGENT",  
"SITEID", "DSDCUSTOMERID", "ASN", "COMMODITYCODE", "QUANTITY", "OUTSTANDING",  
"DISPATCHED", "DENOMINATIONTYPE", "LINEINSTRUCTIONS", "CUSTOMERFIELD 1",  
"CUSTOMERFIELD2 ", "CUSTOMERFIELD 3", "CUSTOMERFIELD4 ", "CUSTOMERFIELD 5",  
"CUSTOMERFIELD6 ", "CUSTOMERFIELD 7", "CUSTOMERFIELD8 ", "CUSTOMERFIELD 9",  
"CUSTOMERFIELD10"

## STOCK

"SUPPLIERCODE", "COMMODITYCODE", "LOTNUMBER", "RECEIVEDDATE", "EXPIRYDATE",  
"BAGWEIGHT", "RECEIVEDWEIGHT", "COST", "DELETED", "STOCKLOCATIONCODE", "QUANTITY",  
"INCREMENT"

## STOCK LOCATION

"CODE", "NAME", "DELETED", "RETURNSLOCATION", "TYPE"

## STOCK TRANSACTION

"TRANSACTIONTYPE", "TIMESTAMP", "COMMODITY", "LOTNUMBER", "FROMLOCATION", "TOLOCATION",  
"FROMQUANTITY", "TOQUANTITY", "QUANTITY", "FROMDATE", "TODATE", "FROMCOST", "TOCOST",  
"PURCHASEORDER", "SALESORDER", "REASON", "RECEIVEDDATE", "EXPIRYDATE", "DESPATCHDATE"

## SUPPLIER

"CODE", "NAME", "DELETED", "CONTACTEMAIL", "CONTACTNAME", "FAX", "ADDRESS",  
"NOTES", "TELEPHONE", "WEBSITE"

## USER

"CODE", "NAME", "PASSWORD", "DELETE"

