Hypothetical Meals

Bulk Format EV 3 Import Proposal v2

Bulk Import Specifications

Format

The format of the file will be **CSV** (Comma Separated Value). This will allow users to easily create files from spreadsheet applications such as Excel and Google Sheets by “export as .csv file” functionality. This will provide an enjoyable experience for the user, as it requires low difficulty and has popular software programs that support this format. It will also allow existing spreadsheets (as stated in the Evolution 2 Introduction) to be used for import, as the format is likely easy to replicate from existing spreadsheets.

There are three different types of bulk imports: raw ingredient import, intermediate product import, and final product import. Each type of bulk import must be specified in a separate file. Mixing two or more types of bulk import into one file is not permitted.

General Requirements and Specifications

The following specifications apply to all bulk import files:

1. The import action shall only occur if the entire input is **free of name conflicts** or otherwise **problematic issues**; if such issues arise, the precise nature of the error should be presented to the administrator in enough detail that it can be corrected.
    a. Problematic issues include vendor doesn’t exist, capacity would be exceeded by bulk import, etc.
2. A double-quote appearing inside a field must be escaped by preceding it with another double quote. (e.g. "aaa","b""bb","ccc")
3. If commas are within any field, the field will be **double quoted** by the CSV format to allow for proper parsing.
4. The file could take either CRLF or LF as line breaks, but the choice of line break through a particular file should be consistent. Any occurrence of CRLF or LF MUST represent a line break. Use of CR and LF outside of line break sequences is forbidden.
5. Character encoding is in **UTF-8**.
6. All fields are **case insensitive**

Raw Ingredient Import

Inputs

Headers (correspond to fields below)

- INGREDIENT
- PACKAGE
- NATIVE UNIT
- UNITS PER PACKAGE
- PRICE PER PACKAGE
- VENDOR FREIGHT CODE
- TEMPERATURE

Fields

- *Ingredient* - unique name of the ingredient being added.
- *Package* - choice of “sack”, “pail”, “drum”, “supersack”, “truckload”, or “railcar”.
- *Native Unit* - string representing the unit of measure for the ingredient
- *Units Per Package* - positive, decimal value of units per package
- *Price Per Package* - positive, decimal value - specified up to hundredths place.
- *Vendor Freight Code* - alphanumeric freight code for an existing Vendor.
- *Temperature* - choice of “**frozen**”, “**refrigerated**” or “**room temperature**”.

Requirements and other specifications

1. Imported **Vendors** must already be **pre-existing** in the database;
2. Imported Ingredients do **NOT** have to be pre-existing, but can also already be in the database (e.g. adding more stock of an ingredient, if the system chooses to support this functionality)
    a. If an ingredient by name is pre-existing, the packaging, native units, units per package, and temperature state must match
        i. Price per package can be differing if it is from a different vendor
3. Price per package must be a positive value, specified up to the hundredths place.

Examples

Ingredient Import

Excel Representation

	A	B	C	D	E	F	G
1	INGREDIENT	PACKAGE	NATIVE UNIT	UNITS PER PACKAGE	PRICE PER PACKAGE	VENDOR FREIGHT CODE	TEMPERATURE
2	Carrots, pail	pail	lbs	15	20.15	DF2CK21	room temperature
3	Carrots, pail	pail	lbs	30	16.89	JK878DC	room temperature
4	Carrots - railcar	railcar	lbs	432	4000	DF2CK21	room temperature
5	Grapes - supersack	supersack	bunches	25	69.32	JK878DC	frozen
6	Potatoes - drum	drum	units	8	32.1	DF2CK21	refrigerated
7	Potatoes, sack	sack	units	4	6.01	KDC322D	refrigerated
8							

CSV Output from Excel

Example

```
INGREDIENT,PACKAGE,NATIVE UNIT,UNITS PER PACKAGE,PRICE PER
PACKAGE,VENDOR FREIGHT CODE,TEMPERATURE
"Carrots, pail",pail,lbs,15,20.15,DF2CK21,room temperature
"Carrots, pail",pail,lbs,30,16.89,JK878DC,room temperature
Carrots - railcar,railcar,lbs,432,4000,DF2CK21,room temperature
Grapes - supersack,supersack,bunches,25,69.32,JK878DC,frozen
Potatoes - drum,drum,units,8,32.1,DF2CK21,refrigerated
"Potatoes, sack",sack,units,4,6.01,KDC322D,refrigerated
```

Final Product Formula Import

Inputs

Headers (correspond to fields below)

- NAME
- PRODUCT UNITS
- DESCRIPTION
- INGREDIENT
- INGREDIENT UNITS

Fields

- *Name* - Unique name of the formula being added.
- *Product Units* - Integer number of product units the formula will produce
- *Description* - A long-form description/notes field about the formula and its product
- *Ingredient* - name of the ingredient being used for this formula
- *Ingredient Units* - floating-point number of the amount of a particular ingredient consumed in native units used in the formula.

Requirements and other specifications

1. The formula name **cannot be pre-existing.** In other words, the name of the imported final product formula may not be the name of another final/intermediate product that is already in the system.
2. Ingredients specified in the formula must already be **pre-existing** in the database.
3. Product Units must be a **positive integer** value.
4. Ingredient Units are **floating point** quantity measured in native units.
5. Each row should only contain one ingredient and one ingredient units, in separate columns.
6. If a formula requires more than one ingredient, only the first column specifying the formula name must be the same in later rows, the other two columns for product units and description will be ignored. Only the product units and description from the earliest row of the formula will be used.
7. Formula for one product should be grouped into consecutive rows. For example, if another row appears with “Cake” in the formula field below “Soup”, it would be considered a formula for a new product and would result in a duplication of formula name error, causing the bulk-import to fail.

Examples

Excel Representation

	A	B	C	D	E
1	NAME	PRODUCT UNITS	DESCRIPTION	INGREDIENT	INGREDIENT UNITS
2	Cake	1	This is a cake	Butter	2
3	Cake			Egg	1
4	Cake			Sprinkles	400
5	Soup	2	This is soup	Tomato	4
6	Soup			Milk, 2%	1

CSV Output from Excel

```
NAME,PRODUCT UNITS,DESCRIPTION,INGREDIENT,INGREDIENT UNITS
Cake,1,This is a cake,Butter,2
Cake,,,Egg,1
Cake,,,Sprinkles,400
Soup,2,This is soup,Tomato,4
Soup,,,"Milk, 2%",1
```

Intermediate Product Import

Inputs

Headers (correspond to fields below)

- NAME
- PRODUCT UNITS
- DESCRIPTION
- PACKAGE
- NATIVE UNIT
- UNITS PER PACKAGE
- TEMPERATURE
- INGREDIENT
- INGREDIENT UNITS

Fields

- *Name* - Unique name of the intermediate product formula being added.
- *Product Units* - Integer number of product units the formula will produce
- *Description* - A long-form description/notes field about the formula and its product
- *Package* - choice of “sack”, “pail”, “drum”, “supersack”, “truckload”, or “railcar”.
- *Native Unit* - string representing the unit of measure for the intermediate product
- *Units Per Package* - positive, decimal value of units per package
- *Temperature* - choice of “**frozen**”, “**refrigerated**” or “**room temperature**”.
- *Ingredient* - name of the ingredient being used for this formula
- *Ingredient Units* - floating-point number of the amount of a particular ingredient consumed in native units used in the formula.

Requirements and other specifications

1. The **name** of the imported intermediate product formula may not be the name of another raw ingredient, intermediate product, or final product that is already in the system.
2. Product Units must be a **positive integer** value.
3. Ingredients specified in the formula must already be **pre-existing** in the database.
    a. Implication: This means importing nested intermediate products in one file is not permitted. For example, if the user wants to import an intermediate product A that uses ingredient R, and intermediate product B that uses intermediate product A, formula for A and that for B must be put in different files and imported in that order.
4. Ingredient Units are **floating point** quantity measured in native units.
5. Each row should only contain one ingredient and one ingredient units, in separate columns.
6. If a formula requires more than one ingredient, only the first column specifying the formula name must be the same in later rows, the other two columns for product

units, description,package, native unit, units per package, and temperature will be ignored. The system will use those values as specified by the first row of the formula.

7. Formula for one product should be grouped into consecutive rows. For example, if another row appears with “seasoning” in the name field below “sugar coating”, it would be considered a formula for a new product and would result in a duplication of formula name error, causing the bulk-import to fail.

Examples

Excel Representation

	A	B	C	D	E	F	G	H	I
1	NAME	PRODUCT UNITS	DESCRIPTION	PACKAGE	NATIVE UNIT	UNITS PER PACKAGE	TEMPERATURE	INGREDIENT	INGREDIENT UNITS
2	Refined Oil	15	better oil	drum	gallons	18	room temperature	Peanut Oil	20
3	Seasoning	5	extraordinary taste	sack	pounds	25	room temperature	Salt	16
4	Seasoning							Black Pepper	11
5	Sugar Coating	7	improves taste and look	sack	pounds	24	room temperature	Sugar	5
6	Sugar Coating							Water	28
7									

CSV Output from Excel

NAME,PRODUCT UNITS,DESCRIPTION,PACKAGE,NATIVE UNIT,UNITS PER PACKAGE,TEMPERATURE,INGREDIENT,INGREDIENT UNITS
Refined Oil,15,better oil ,drum,gallons,18,room temperature,Peanut Oil,20
Seasoning,5,extraordinary taste,sack,pounds,25,room temperature,Salt,16
Seasoning,,,,,,,Black Pepper,11
Sugar Coating,7,improves taste and look,sack,pounds,24,room temperature,Sugar,5
Sugar Coating,,,,,,,Water,28

Acknowledgement

This file is created based on the effort of the format committee, feedback from Professor Bletsch, and csv standards proposed here