

## Special Issues

### Inventory Valuation

The system provides a mechanism for entering “inventory on hand” counts and calculating the value of the inventory on hand. This is done in the “Enter Inventory” screen. Upon entering this screen, the system displays only those items that have “Show On Inventory Forms” turned ON. If you see items you do not wish to see, go into the item record and change the “Show On Inventory Forms” button to off. And vice versa. Be aware also that the system may not show some items on the count forms, but it will display ALL items that have ANY activity on the various Inventory Reports. This fact has many ramifications for the inventory value the system calculates. It is entirely possible, for instance, to buy a product that has been turned off for showing on count sheets, and count a product that has never been bought nor used. This could be valid and desirable in some circumstances. But it could also lead to gross inaccuracies as well. You are responsible for knowing your inventory and reviewing the many system reports and keeping your data current and valid. We feel we have a system that makes this very easy to do and yet provides the opportunity to have very detailed and accurate inventory activity reports. Normally common logic is all that is needed. One does not have to be a computer whiz to get it right. Hopefully the following discussions of Inventory Valuation will be helpful in understanding our system logic.

### Counting

Your first line of defense is in the actual counting of your inventory. Normally this is done using an inventory form printed for this purpose. It is important to print these forms on the same day the count is to occur so that the form contains the latest system info. Using a stash of copied forms you printed last month will get you into trouble as things often change within the system and data on those forms may now be invalid. After the forms are filled out during a physical count, someone will type those amounts into the “Enter Inventory” screen. That is the perfect time to correct inventory problems. When you enter an amount in the “count” column and tab out, the system will use the valuation method chosen for that inventory item in conjunction with its current “Pack Unit Setup” to calculate the cost of the on hand amount you just entered.

The important thing to always be on the look out for when you enter inventory is the Unit of Measure for the item being counted. Everything depends upon this “Inv Unit” being correctly setup, and correctly interpreted at count time.

Ending Inventory Entry					
Prod No	Description	Brand	Vendor	Pack Unit	Inv Unit
5916	Ice Cream	Barbers	Barber	3 Gal. Tub	Gallon
50849	Margarine Prints	Kraft	Kraft	30/1#	Pound
1234	Monterey Jack Cheese	Kraft	Kenneth	4/10# Pkg.	Case
30163	Sour Cream	Breakstone	Kraft	4/5# Cont'r	Cont'r
B*9	Table Butter Blend	House Item	Batch	Pound	Pound
75047220	Whipping Cream, 36%	Olde Style	Kraft	12/Quarts	Quart

Make sure you check the Inv Unit (Unit of Measure) to count!

In other words, if the UM is “Contr” and someone just counted this item in “Pounds”, inventory value will not be correct. A common mistake is when two people count the same item in two locations and one counts “pounds” while the other counts something else (“each” etc). It is important to review the “Inv Unit” on the count form and make sure everyone understands what they are counting. Sometimes it is simply impossible to count what is printed on the form because the Admin just didn’t “get it” and put in anything at all just to make an invoice come out right. This happens especially with an Admin who simply had little training when he/she took over the job from a buddy or has little experience in the kitchen. We think anyone who is in charge of inputting new products at invoicing time should help count inventory from time to time. This will probably help clean up your inventory file better than any other one thing you can do.

**Advice:** Teach those taking count to make notes on the form that Admin can understand in order to properly correct the Inventory Record Setup screen. The best time to clean up your inventory file and deal with ill-conceived Pack Unit Setups is at the Inventory Count Entry screen.

As you type in the count, review the “Inv Unit”. If it does not seem right or definitely does not match what was counted, change the UM immediately in the inventory setup screen (double click the line to get there) or re-count using the proper unit of measure. In any case, do not type in a number that is obviously a wrong count for the Inv Unit showing on this screen. You will simply get “garbage in/garbage out” results. If you do change it you MUST be aware of the consequences if that item was entered into an invoice with the existing “wrong” pack unit in play. This most often means simply don’t change the existing “Pricing Unit” so that it is just not the same as what may be in existing invoices. If historical invoices are not an issue, there will be no consequences. Invoices will ONLY be an issue if the product is setup for

FIFO ***and*** your count exceeds the last purchased amount. Only then will the system go back through past invoices trying to calculate the “real” value of the items you just counted. The safest correction to an inventory item is a simple change to the “count” unit. You can change the entire pack unit or add a missing count unit and the product will still calculate properly through past invoices as long as you don’t completely change the way the system calculates purchase/pricing units. In other words, it is probably not a good idea to change the Pricing Unit from “case” to “pound” at the end of the month. It would be safe to ADD a count unit of “pound” as an extra column for a case pack unit as long as it properly equates to the existing case. If you do need to change the product setup entirely you are probably best advised to get through the closing and change it at the very beginning of the new month before you enter any new invoices.

As an example, lets say that you buy an item by the case, and you break it out to inventory it by the container (6 per case). The case price on the last invoice was 29.28. Admin entered an invoice for 2 cases and had to setup the item since it was new to the system. This admin knew nothing about the item or how it would be counted, so he/she simply entered the easiest thing possible, a simple “case” pack unit. The invoice will look and calculate just fine.

	Purchase Unit	Pricing Unit			Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
Inventory Units	Case	Case			Each
Modifiers	1	1	1	1	1
Price	29.28	0.00	0.00	29.2800	

At Month End you count 10 containers of this item, but it is obviously not setup correctly since it is stored and counted as individual small containers and there is no way you have 10 cases. You decide that you need to change the product to match the count taken so you make the following change.

	Purchase Unit	Pricing Unit			Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
Inventory Units	Contr	Contr			Each
Modifiers	1	1	1	1	1
Price	29.28	0.00	0.00	29.2800	

**BEWARE!** You just changed the Pricing Unit and the math logic underneath it will no longer match your invoice reality. When you count this item, you

will definitely be placing a very high value on it! Those 10 containers you counted actually cost you 4.88 each, NOT 29.28 each. A difference of \$234 in value!

**CORRECT SETUP AND LOGIC:** Follow the 4-step Cimarron Logic:

1. **PURCHASE UNIT:** The vendor makes you buy it by the “CASE”.
2. **PRICING UNIT:** The vendor prices it by the “CASE” on invoices. Admin sees the “Pricing Unit” column on invoices so they will see and enter the proper number of cases being purchased and your invoice will match the vendor’s invoice.
3. **COUNT UNIT:** You **COUNT** by the container. There are 6 containers in a case. So if you count 10 containers the system calculates 1.6666 cases. The system will use past invoices to see how much you paid for those 1.6666 cases and your total inventory value will calculate properly.
4. **PORTION UNIT:** Always try to carry the logic into the last column.
- 5.

*This setup will calculate your container count properly and still work properly with the “wrong” setup in existing invoices using FIFO (see first setup screen above).*

	Purchase Unit		Pricing Unit		Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
<b>Inventory Units</b>	Case	Case	Cont'r		Ounce
<b>Modifiers</b>	1	1	6	1	16
<b>Price</b>		29.28	4.88	0.00	0.3050

## Valuation Methods

The Cimarron Restaurant Solutions System provides for 3 inventory valuation methods. The user can control how value is calculated for each individual item by clicking the appropriate method in the Inventory Record Setup Screen.

Inventory Value Method				
<input checked="" type="radio"/> FIFO	<input type="radio"/> LAST PRICE	<input type="radio"/> AVERAGE		
UNIT TO COUNT				
Case	Case	Cont'r		Ounce
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
<input type="checkbox"/> EACH		<input type="text" value="0"/>		

Each method has its pros and cons, and the literature is rife with descriptions and endless discussions of the merits of one method over another. We think the user should be empowered to use what is best for his operation and for each type product.

### FIFO

FIFO (First In First Out) works by going back through previous invoices in the (reverse) order they were created and calculating the value of the counted item as it was purchased. Assume you bought 1 bag of carrots for \$1 on the first day this month. Then you bought two bags the next week at \$3 per bag, and 2 bags the last week for \$1 per bag again. Now assume that you counted 3 bags of carrots on hand. Using FIFO, the system will find the last invoice and calculate 2 bags at a value of \$1 each, then move back to the next invoice and calculate the remaining bag counted at \$3. Total of \$5 value (2 X \$1) + (1 X \$3).

FIFO GOTHAS! It is extremely important to understand that if you move back through invoices calculating what you just counted, then the invoices in question must have been PRICED in like manner, or the FIFO method is likely to result in outrageously incorrect values! Additionally, if the item is linked to other items, you must be able to COUNT them in like manner.

Using the above example lets assume that you bought two 25-pound bags of carrots at \$7.80/ bag. Admin setup a pack unit of “bag” without thinking and simply made the item match the invoice.

	Purchase Unit	Pricing Unit			Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
<b>Inventory Units</b>	Bag	Bag			Each
<b>Modifiers</b>	1	1	1	1	1
<b>Price</b>	7.80	0.00	0.00	7.8000	

Two weeks later another invoice came in from another vendor and Admin (without thinking again) input this new vendor’s product to match the new vendor’s pricing method (using “Pounds”). This happens to be in line with the actual pound price of Vendor A so really the price has not changed. And when Admin types in 25 Qty on the invoice, again our invoice is going to match Vendor B’s invoice.

	Purchase Unit	Pricing Unit			Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
Inventory Units	Each	Each			Each
Modifiers	1	1	1	1	1
Price	0.31	0.00	0.00	0.3100	

We now link the two carrot products from Vendor A and Vendor B (see the linking section elsewhere). Month End arrives and we count 40 pounds of carrots. We would rightly expect the system to extend the value for carrots to \$12.40 (40lbs X .31). HOWEVER, the system (using FIFO) went back to the last invoice paid and found 25 “each” at .312 and (correctly) calculated 25 of our counting units as worth \$7.80. It then went to the previous invoice and calculated 2 more of our count units at \$7.80 as worth a total of \$15.60. We are now up to an inventory value of \$23.35 and the system still has 13 more “counting units” to find in invoices and render values for (the 40 units counted, less the first invoice qty of 25 and the next invoice qty of 2). These are going to be some expensive carrots when it finishes! And the only reason is that Admin setup the product without understanding the Inventory Pack Unit Rules of Logic.

Let’s setup the product(s) so they work properly for both the invoice and for the counting procedure at Month End.

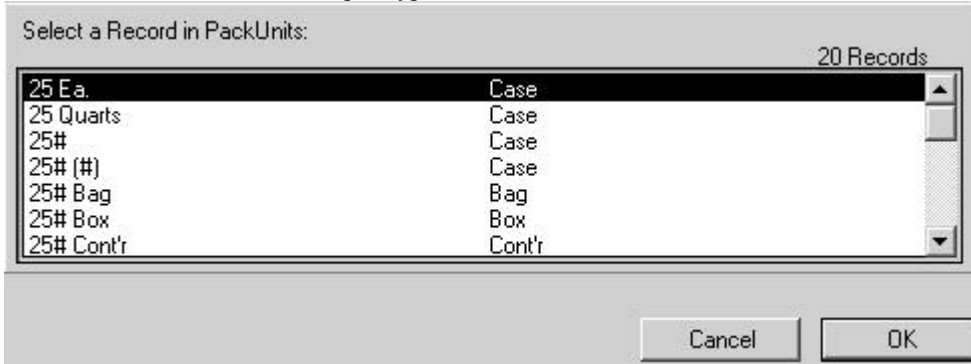
First lets review the 4 Inventory Pack Unit Rules of Logic:

1. PURCHASE UNIT: The vendor makes you buy it by something.
2. PRICING UNIT: The vendor prices it by something on invoices. This is most often the same but can be something different.
3. COUNT UNIT: You COUNT by something. In general the counting unit should always be the same for all linked products. Every product should have a “count” unit that makes sense on count forms to the guy doing the counting. It can be the same as the pricing unit as long as that makes sense and is REALITY. AND all linked items must contain the same counting unit.
4. PORTION UNIT: Always try to break down an item into the smallest unit the item will break down to (ounce, mg, ml, each, etc)

With this in mind, let’s setup the first carrots product for Vendor A. We get these carrots in a bag, and the invoices look like they are priced by the bag.

A little investigation will quickly inform us that these bags always contain 25 pounds and that the kitchen guys throw them into a big plastic container and always COUNT them by the pound. So we need to set the item up to accommodate both parties. We choose “25# Bag” from the master pack unit popup and it fills in the master pack unit setup for us.

**TIP:** Our system comes with over 1200 built in pack units. You are strongly encouraged to use our units. Place your cursor into the pack unit field and type AS LITTLE AS POSSIBLE to get a list of pack units to choose from. In the above example, type “25” and tab. You will see this list:



Note that there are 20 choices on this list. If you don't find what you are looking for, type something else. Or type FEWER CHARACTERS! The system does a look up on what you typed so the less you type, the more you will find. Or put another way, the more you type the more limited your list will get. In some cases like “bag” above, the system will find only ONE choice and simply fill it in for you. You will not see the list. If the system finds nothing, it will tell you so and give you the choice to make up your own pack unit. DO NOT MAKE UP YOUR OWN unless you are certain you have exhausted our choice list options. If this is the case, please email us the new pack unit and we'll add it to our list (or inform you of a better choice we DO have).

Once we fill in “25# bag” as our pack unit, the system will fill in the rest of the math modifiers. Please note that this setup satisfies our 4 rules of pack unit logic. Admin will see the invoice Pricing Unit terminology “bag” and will purchase each bag at 7.80; Kitchen staff will see the correct Inventory Counting Unit “pound” displayed on Inventory forms on count night; the system will correctly calculate the math, turning the 30 pounds counted into 1.2 bags worth 9.36.

	Purchase Unit		Pricing Unit		Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
<b>Inventory Units</b>	Bag	Bag	Pound		Ounce
<b>Modifiers</b>	1	1	25	1	16
<b>Price</b>	7.80	0.31	0.00	0.0195	

Now lets set up the second product so it correctly identifies all our Pack Units and the underlying math the system needs to properly perform FIFO Calculations. Vendor B invoices clearly indicate this vendor sells carrots by the pound. We have no idea how they show up, whether in boxes, bags, or loose. The invoice just states “pounds”. So we set this up a bit differently. Note that our counting unit and pricing unit are the same. The admin will clearly see the pricing unit and when typing in 25 qty (pounds) will see the correct price extended. And the kitchen staff will clearly see “pound” on the count form at month end. Likewise, the system will know how to correctly extend a count of 30 to 7.75. (Note that this vendor is selling carrots for .31/pound (exactly!) whereas the first vendor’s price is 7.80 per 25# bag. The system carries this math (7.80/25) out to 4 digits underneath so Vendor A’s real price is .312/pound. You can see that reflected in the portion unit price: .0194 vs .0195)

	Purchase Unit	Pricing Unit			Portion Unit
	1 ↓	2 ↓	3	4	5 ↓
<b>Inventory Units</b>	Pound	Pound			Ounce
<b>Modifiers</b>	1	1	1	1	16
<b>Price</b>	0.31	0.00	0.00	0.0194	

Now what happens when we link these two and perform FIFO underneath? Again we use the same count sheet of 30 pounds. The first thing the system sees is that this item is “linked”. So it finds all invoices for all the linked items and sorts them so that they line up the way they were entered. Then the system goes to the latest invoice and finds the last linked item you purchased (Vendor B Carrots#2) and uses the value for 25 pounds at .31/pound. This leaves us with 5 more pounds to find, so the system goes to the next invoice in line and finds that we bought a 25-pound bag and takes out 5 pounds from it at .312/pound. Our true FIFO value for the 30 pounds of carrots is 9.31.

### Linking and FIFO

FIFO will not be effective at count time for many products if you are not linking them properly. This means that if you purchase equivalent items from several vendors, OR even more usual, very similar items from the same vendor and do not link these products, you will a) end up with several

similar items on count forms and b) probably enter a count for only one of them, possibly even one that you bought LONG ago at a very different price than what you pay now. FIFO will correctly value that one item. But you just missed the value fluctuations for all those “similar” items you did not count. Had you linked them, you would have a) seen only one of the similar items on your forms, and b) they would have all been put through the FIFO valuation process. Linking is a very important facet of FIFO. And it is very much overlooked. We hope this long discussion helped give you some ideas about cleaning up your inventory.

### Last Price Paid

This has been our traditional valuation method. It is the simplest, and it is probably accurate to 98% in a busy environment and up to 99.9% in one where you purchase extremely often. In such an environment, it would be really rare that you counted items that span several invoices with highly variable costs. Where it falls down is when you have linking on for items from different vendors with wildly varying prices. You may get a good deal from one of the vendors at month end and buy something you weren't yet out of just to take advantage of the price. That last purchase just swapped onto the count sheet and all the other items will now get valued at the low-end price. Properly implemented and managed, FIFO will fix that scenario.

### Average

This is probably a rarely needed valuation method, but it may be better for some items than Price Last Paid, and may give a slightly different analysis tool than FIFO. It averages the price paid this month and last. It does not go back through invoices. It uses the monthly purchase count and monthly purchase amount taking place within the current period, lumped with the count and value you started the month with. It is truly different than counting up the true value with FIFO, even if the system were to use the very same date range invoices. And it is probably more useful as an analysis tool than as a “valuation” method. It could be useful at the right time for the right item so we include it.