

11i WIP Assembly Completion & Overcompletion Transactions in Average Costing

*An Oracle White Paper
December, 2002*

11i WIP Assembly Completion & Overcompletion Transactions in Average Costing

ABSTRACT

The purpose of this white paper is to provide an understanding of Work In Process (WIP) Final Assembly Completion transactions and of how these transactions impact the unit cost of the final assembly when the costing method used is Average Costing. When assemblies being built on WIP jobs are completed, they are moved into the completion subinventory. In Average Costing, the cost of completion determines the unit cost of the completed assemblies based on certain criteria. This triggers a recalculation of the average unit cost of the assembly in inventory. The situation derives an additional flavour if there is a business need to perform overcompletion transactions.

As a new feature introduced in Release 11i, standard discrete jobs and repetitive schedules with a routing can have overcompletion performed on them at any stage of the routing. How this will impact the final unit cost of the assembly again depends on when the overcompletion was done (at what stage in the routing), what is the Supply Type of the components involved and whether the completion transaction was a Final Completion.

SCOPE & APPLICATION

This paper is intended for an audience familiar with Work In Process transactions, and involved in evaluating their costing implications for business or academic reasons. To keep the outcome of our study lucid and simple, we shall be operating under the following conditions:

- The transactions are performed in an Average Costing organization. Working on the Vision database, we have selected organization – Dallas Manufacturing (M3) as our Average Costing organization.
- The scope of our investigation is restricted to a Standard Discrete Job.
- Only Material and Resource cost elements are being considered to evaluate the costing outcome.
- The Standard Lot Size for Costing is 1. (The standard lot size is not very important in average costing. However, “1” is not an interesting number.)
- The items are neither lot nor serial controlled.
- The Auto Compute Final Completion option is disabled.

The costing implications of the completion transactions are analysed through a series of test cases, carefully designed to cover the main permutations possible with a specific set of parameters.

UNDERSTANDING OVERCOMPLETION AND COMPLETION ENHANCEMENTS

Overcompletion and Completion Enhancements are introduced as new features in WIP Release 11i. With reference to Overcompletions :

- You can complete more assemblies than the start quantity through a discrete job or a repetitive schedule whether or not the assembly has a routing.
- You can complete more assemblies from an operation step than those that exist at that operation step.

- You can establish specific tolerance levels for overcompletions. If the transaction makes the completed quantity exceed the tolerance, then the transaction fails. Overcompletion tolerance can be defined at two levels of control :

- o At the Item attributes level :
Inventory: Items > Organization Items > (T) Work In Process

The Tolerance Type can be defined either as Percent or as Amount.
This is followed by the Tolerance Value (Fig1).

- o At the Discrete Job level :
WIP: Discrete > Discrete Jobs > (T) More

The overcompletion tolerance level setup at the job level (Fig 2.) defaults from what is defined at the item attributes level. But the system allows you to override the default values defined at the item level.

The screenshot shows the SAP 'Organization Item (M3)' form for 'Dallas Manufacturing'. The 'Item' is 'Fountain Pen'. The 'Work In Process' tab is active. The 'Overcompletion' section is expanded, showing 'Tolerance Type' as 'Percent' and 'Tolerance Value' as '200'. The 'Supply' section shows 'Type' as 'Push'. The 'Scheduling Penalty Per Day' section has empty input fields for 'Inventory Carry' and 'Operation Slack'. The 'Build in WIP' checkbox is checked.

Fig 1. Overcompletion tolerance setup in the Item definition form

The screenshot shows the 'Discrete Jobs (M3)' form with the following fields and values:

- Job: 48033
- Type: Standard
- Assembly: Fountain Pen
- Class: Discrete
- Status: Released
- UOM: Ea
- Quantities: Start 4, MRP Net 4
- Dates: Start 06-DEC-2002 00:00:00, Completion 06-DEC-2002 00:00:00
- Overcompletion: Tolerance Type Percent, Tolerance Value 200

Buttons at the bottom include Sales Orders, Operations, and Components.

Fig 2. Overcompletion tolerance setup at the job level

- Finally, you can specify tolerance level for overcompletion in the WIP Parameters form
WIP: Setup >Parameters > (T) Other

The screenshot shows the 'Work in Process Parameters (M3)' form with the 'Other' tab selected. The following fields are visible:

- Component ATP Rule: [Empty field]
- Default Overcompletion Tolerance %: [Empty field]

Fig 3. Default Overcompletion Tolerance on the WIP Parameters form

This is used to determine tolerance levels if the tolerance value is not set up at item level or at the job level (Fig 3).

- For tracking purposes, you can distinguish between an overcompleted job from a normal job.
- Overcompletions are performed in the Assembly Completion window, in the Move Transaction window or in the Pending Move Transaction window, using check boxes. Once the Overcompletion option is enabled, tolerance level validations are performed for the job or schedule item.

The feature of WIP Completions has undergone some enhancements in WIP Release 11i.

- Job and repetitive schedule statuses are automatically updated to Complete when the total number of assemblies completed and scrapped on a job or repetitive schedule is greater than or equal to the required quantity.
- Negative cost, by element by level, can be created by a partial completion method that relieves more cost than incurred. When assemblies are completed as final completions, negative cost in the job is credited to variance. You can capture variances created by negative cost when work in process final completion transactions are transacted. (The test cases that follow shall illustrate this point).

UNDERSTANDING ASSEMBLY COMPLETION IN AVERAGE COSTING

When assemblies being built on WIP jobs are complete, they are moved into a subinventory, called the Completion Subinventory, by means of a completion transaction. You can define the Completion Subinventory of the parent assembly either when defining the Routing (by clicking on the Routing Details button), or later when creating the discrete job, under the Routing tab. Through this completion transaction, the system calculates a unit cost for the completed assemblies and creates accounting entries, using this unit cost.

Since Average Costing calculates the item cost on a perpetual transaction-by-transaction basis, moving the finished assembly into the completion subinventory calls for a recalculation of the unit cost of the completed assembly. This recalculation occurs when the unit cost of the assembly in all subinventories is different from the unit cost being used in the completion transaction.

A completion transaction can be with or without Final Completion. The Final Completion checkbox is found in WIP: Material Transactions > Completion Transactions form. This option is used usually when you complete all remaining assemblies for a job. Then you can check the Final Completion checkbox. This ensures that the current job balance is relieved and spread evenly over the units being completed. Final Completion ensures that no positive residual balance is left in the job after the last assembly has been completed. However, if you choose not to use the Final Completion option, then the default valuation methods defined in the WIP Parameters form come into play. These valuation options are specific to the WIP Accounting Class. The defaults are located at WIP: Setup > Parameters > (T) Costing

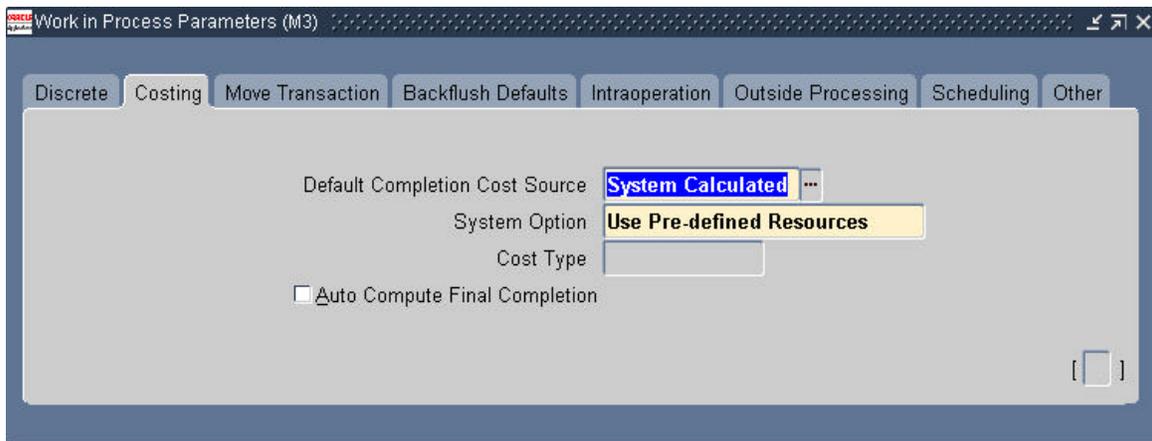
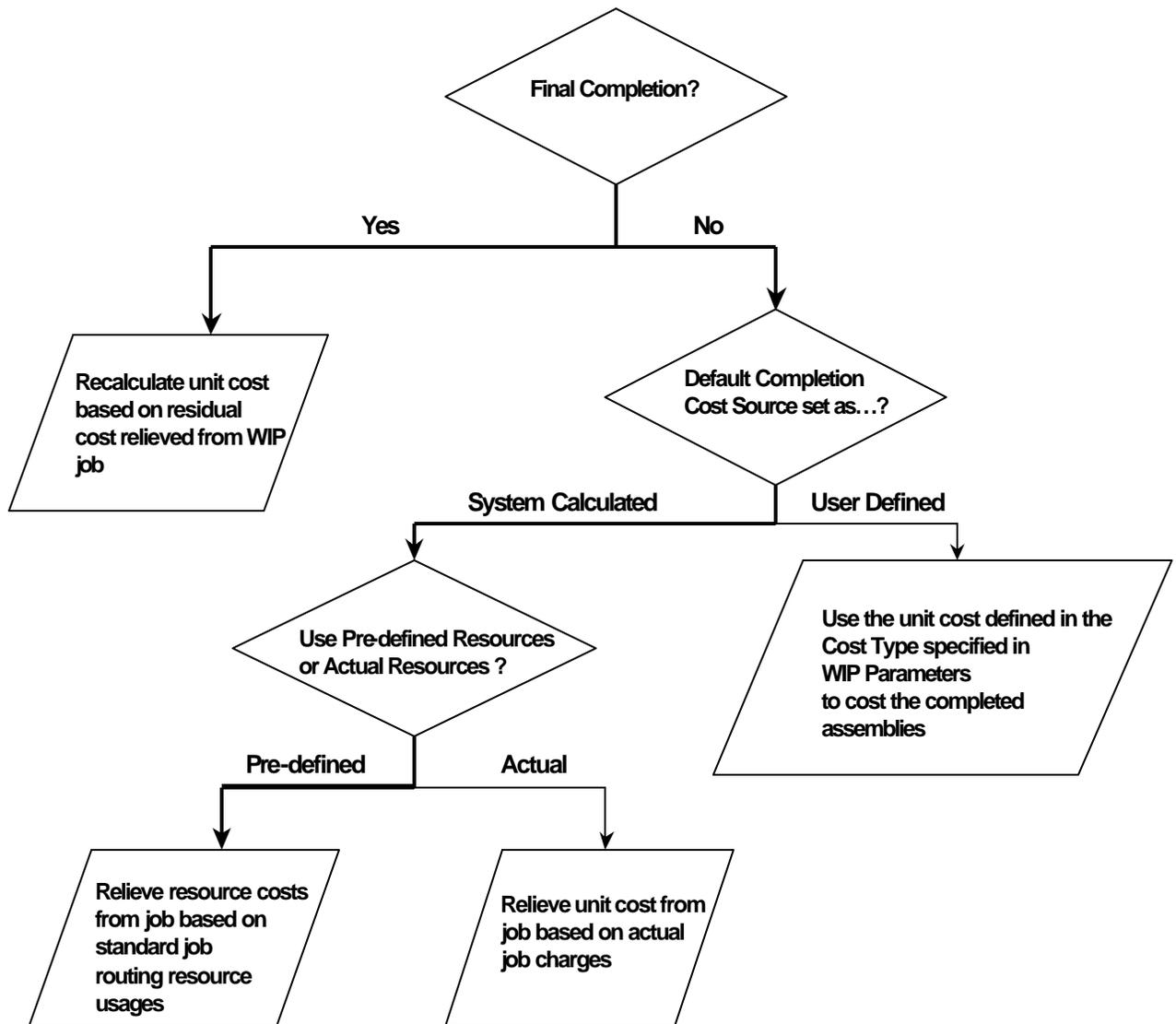


Fig 4. Assembly Completion Cost setup in WIP Parameters

This Costing tab in the WIP Parameters form will not be available in a Standard Costing organization.

How this setup works is explained in the flowchart in Fig 5.



Both the Pre-defined and Actual algorithms use the job Bill of Material (Material Requirements) to calculate cost related to components.

Fig 5. Flow diagram for unit cost calculation of completed assemblies in Average Costing

The investigation performed for this paper will cover cases that comply with the conditions along the “bold lines” in the flowchart above.

SETUP FOR THE TEST CASES

Since we shall be demonstrating the assembly completion options and their costing implications through a series of test cases, we need to create a basic setup for the same.

- Inventory: Items > Master Items
Define an item called Fountain Pen. Use the Finished Good template for this item. This is a Make item that will constitute our final assembly. Under the Work In Process tab, set the Overcompletion Tolerance Type as Percent and the Tolerance Value as 200. Save your work. Assign this item to organization Dallas Manufacturing, hereafter referred to as M3. Define Buy items, Body and Nib. Use the Purchased Item template for each of these. Assign these items to M3.
- Ensure that the Cost Manager and the Move transaction Manager are active.
Inventory: Setup > Transactions > Interface Managers
If any of the managers are inactive, they need to be launched from Tools > Launch Manager.
- Inventory: Transactions > Miscellaneous Transaction
Perform a Miscellaneous Receipt of 100 units each of Body and Nib
@ \$10/unit for Body, and
@ \$12/unit for Nib
into the Stores subinventory.
The system will register these values as the Material unit costs for these items under the Cost Type **Average**. You can verify this from
Inventory > Costs > Item Costs
- Bills Of Materials > Bills > Bills
Create the Bill of Material as shown below :

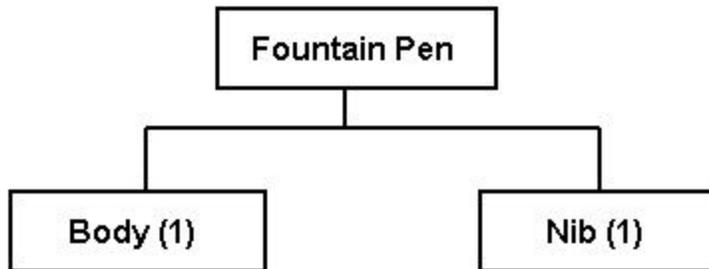


Fig 6. Bill of Material of Fountain Pen

Under the Material Control tab, set the Supply Type to Push for both components and enter the Subinventory as Stores.

- Bills Of Materials > Routings > Departments
Define a department called Stationery.
- Bills Of Materials > Routings > Resources
Define two resources – Grinder and Fitter. The resource Grinder ensures a proper finishing on the metal body of the pen. The resource Fitter then fits the nib into the body to complete the Fountain Pen product.
For both of these resources, UOM = HR, Type = Machine,
Charge Type = WIP Move, Basis = Item

Note: The UOM you define here, should either be the same as the UOM specified in the profile option BOM: Hour UOM, or should at least belong to the same class as that of the UOM specified in the profile option.

Costed = checked

Since we are costing these resources, we will need to enter an Absorption Account.

This is the account that gets credited when the resource is charged to a job.

Standard = checked

Since these resources are to be costed at standard rate, enter a Variance Account.

Bills Of Materials > Routings > Resources > (B) Rates

Cost Type = **First Quarter** (say). We have defined **First Quarter** as the rates cost type for the first quarter.

Rate = \$2/HR (defining the same rate for both Grinder and Fitter)

- Bills Of Materials > Routings > Departments
Query the Stationery department.
Bills Of Materials > Routings > Departments > (B) Resources
Attach the two resources defined above to this department.
Units = 1, for both the resources.
- Bills Of Materials > Routings > Routings
Define a routing for Fountain Pen as shown in Fig 7.

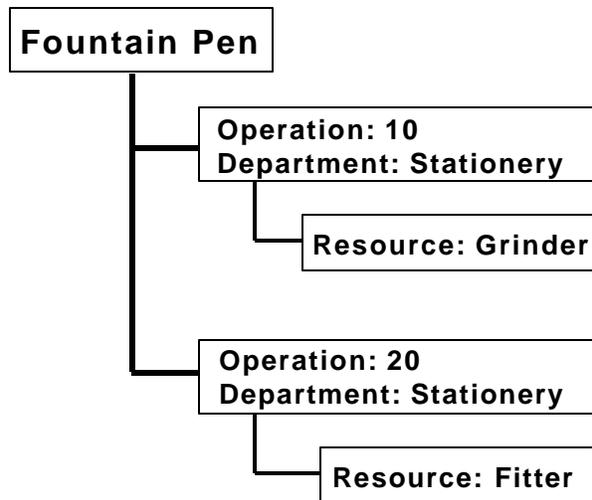


Fig 7. Routing of Fountain Pen

Bills Of Materials > Routings > Routings > (B) Routing Details
Enter **FGI** as the Completion Subinventory.

For each resource, **Usage = 1**.
Both operations have Count Point and Autocharge enabled.

- Bill Of Materials > Bills > Bills
Requery the BOM for Fountain Pen.
Attach Operation 10 to the component Body, and Operation 20 to the component Nib.

This completes the setup we will need to carry out our tests.

TEST CASES FOR COMPLETION / OVERCOMPLETION TRANSACTIONS

We shall demonstrate four test cases. Each case is uniquely identified by a specific set of parameters. Based on these parameters, the cases are delineated as shown in the table below :

	Supply Type of Components	In the final transaction...	
		Final Completion	Overcompletion
Case 1	Push	No	Yes
Case 2	Push	Yes	Yes
Case 3	Pull	Yes	No
Case 4	Pull	Yes	Yes

Fig 8. Test Case definitions

For each of these test cases, we shall constantly refer to the WIP Value Summary form to evaluate the costs incurred by and relieved from the WIP job at each stage.

Case 1

- (1) WIP: Discrete > Discrete Jobs
Create a job for Fountain Pen.
Start Quantity = 4.
- (2) WIP: Material Transactions > WIP Material Transactions
For the job defined above, perform a WIP component issue for 4 units of Body and 4 units of Nib from the Stores subinventory. (Recall: The Supply Type of the components is set to **Push** in the BOM).

WIP: Discrete > WIP Value Summary
Review Value Summary for the above job.
We find :

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	0	0	88
Resource	0	0	0	0
Total	88	0	0	88

Reason:

Cost Incurred by the job = 4 units of Body + 4 units of Nib = (4*\$10) + (4*\$12) = \$88.

Net Activity reflects the value of Cost Incurred minus Cost Relieved.

The accounting distributions would show:

WIP Valuation Material Account	Dr	88	
Subinventory (Stores) Material Account	Cr		88

- (3) WIP: Move Transactions > Move Transactions
Perform a Move Transaction for Qty = 2.
From Op 10 Queue to Op 20 To move.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	0	0	88
Resource	8	0	0	8
Total	96	0	0	96

Reason:

Cost Incurred = Resource charge = Resource Rate * Resource Usage * Qty.

For 2 units of Body through Grinder + 2 units of Nib (and Body) through Fitter,

Resource charge = (\$2/HR*1HR/unit*2units) + (\$2/HR*1HR/unit*2units) = \$8.

The accounting distributions show:

WIP Valuation Resource Account	Dr	8	
Resource Absorption Account	Cr		8

- (4) WIP: Material Transactions > Completion Transaction
Perform a Completion transaction for Qty = 2.
Final Completion = No (box unchecked).

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	44	0	44
Resource	8	8	0	0
Total	96	52	0	44

Reason:

For this completion transaction, Material cost relieved from the job amounts to the raw material (components) worth 2 units of Fountain Pen, i.e. 2 units of Body (\$20) + 2 units of Nib (\$24). Hence \$44 is relieved.

Resource was charged strictly in proportion to the units completed.

Hence the entire \$8 of resource charge is relieved from the job.

The accounting distributions shows:

Subinventory (FGI) Material Account	Dr	44	
Subinventory (FGI) Resource Account	Dr	8	
WIP Valuation Material Account	Cr		44
WIP Valuation Resource Account	Cr		8

Note:

2 units of Fountain Pen have been completed into Inventory. The system will create Material and Resource unit costs under the Material and Resource Cost Elements respectively, for Fountain Pen, with Cost Type = Average.

The unit costs at this stage are:

Material = Material Cost relieved from WIP job into Inventory/assembly units completed
= \$44/assembly units completed = \$44/2 = \$22

Resource = Resource Cost relieved/assembly units completed = \$8/2 = \$4

Therefore, at this point our Fountain Pen has an average unit cost of : \$22 + \$4 = \$26.

Verify this from either of the two:

Inventory: Costs > Item Costs

Cost: Item Costs > Item Costs

(5) WIP: Move Transactions > Move Transactions

Perform a Move Transaction for Qty = 2.

From Op 10 Queue to Op 20 To move.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	44	0	44
Resource	16	8	0	8
Total	104	52	0	52

Reason:

For this move transaction,

Resource charge = 2 units moved through Operation 10 + 2 units moved through Operation 20
= (\$2/HR*1HR/unit*2units) + (\$2/HR*1HR/unit*2units) = \$8.

Hence, resource Cost Incurred increases to \$16.

(6) WIP: Material Transactions > Completion Transactions

Perform a Completion transaction for Qty = 3.

Overcompletion = Yes

Final Completion = No

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	110	0	-22
Resource	20	20	0	0
Total	108	130	0	-22

The job status would now change to **Complete** as the job Start Quantity has been exceeded.

Reason:

3 units of Fountain Pen have been completed and moved into the FGI subinventory. So the Material Cost Relieved increases by 3*(Body + Nib) = 3*(\$10 + \$12) = \$66. Also, 2 units of Fountain Pen had been completed earlier. So the Material Cost Relieved stands at \$110. The Resource Cost Incurred has increased from \$16 to \$20. This is to account for the excess 1 unit of completed assembly that needs to be charged through Operation 10 and Operation 20.

When overcompletion is performed and you move more than the available quantity into To move, any extra quantities will be moved through each of the previous operations. The WIP job would be charged (and relieved) accordingly.

Whatever resource charge had been incurred (\$20) has been relieved following the 'completion' transaction.

Net Activity shows -\$22. How or why did this happen? Let us sum up the accounting distributions to figure this out.

From (2),

WIP Valuation Material Account	Dr	88	
Subinventory (Stores) Material Account	Cr		88

From (4),

Subinventory (FGI) Material Account	Dr	44	
WIP Valuation Material Account	Cr		44

From (6),

Subinventory (FGI) Material Account	Dr	66	
WIP Valuation Material Account	Cr		66

Therefore, WIP Valuation Material Account has a net credit of \$22.

You can check at the table level for cost incurred and relieved by cost element through the following query:

```
SELECT pl_material_in Matl_CI, pl_material_out Matl_CR,
tl_resource_in Res_CI, tl_resource_out Res_CR, acct_period_id Acct_Pd
FROM WIP_PERIOD_BALANCES
WHERE pl_material_in <> 0
AND wip_entity_id IN (SELECT wip_entity_id
FROM WIP_ENTITIES
WHERE wip_entity_name = '&discrete_job'
AND organization_id = &org_id);
```

MATL_CI	MATL_CR	RES_CI	RES_CR	ACCT_PD
88	110	20	20	2226

Each row in the WIP_PERIOD_BALANCES table stores this level and previous level cost element summary values for a job or schedule in a given accounting period.

Note:

Check the item cost for Fountain Pen.

Recalculated unit cost = [(Qty on Hand * Average unit cost) + (Completion cost for new units)]/(Qty on Hand + new units completed)

As per the transactions so far,

Material Unit Cost = [(2*\$22) + \$66]/(2+3) = \$22

Resource Unit Cost = [(2*\$4) + \$12]/(2+3) = \$4

Therefore unit cost = \$26.

Verify this from either of the two:

Inventory: Costs > Item Costs

Cost: Item Costs > Item Costs

Also, every time you perform a completion transaction, you may verify the unit cost for the completed assembly including the prior cost and the new cost from the table MTL_MATERIAL_TRANSACTIONS, by using the following query :

```

SELECT primary_quantity PQ, transaction_quantity TQ, actual_cost AC, prior_cost PC,
new_cost NC, final_completion_flag F
FROM MTL_MATERIAL_TRANSACTIONS
WHERE transaction_source_id IN(SELECT wip_entity_id
                               FROM WIP_ENTITIES
                               WHERE wip_entity_name = '&discrete_job'
                               AND organization_id = &org_id)
AND transaction_action_id = 31 /* Assembly Completion */
AND organization_id = &org_id;

```

PQ	TQ	AC	PC	NC F
2	2	26	0	26 N
3	3	26	26	26 N

The final_completion_flag shows if the Final Completion box was checked for the transaction.

Overcompletion also adds to the Material Requirements of the components to match the actual quantity of parent assembly completed. This can be viewed from WIP: Job/Schedule Details > View Material Requirements

You can also verify this information from the table WIP_REQUIREMENT_OPERATIONS. The following query demonstrates this:

```

SELECT substr(MSI.segment1,0,10) Item, WRO.quantity_per_assembly PER_ASSY,
WRO.required_quantity REQD,
WRO.quantity_issued ISSUED, WRO.required_quantity-WRO.quantity_issued OPEN
FROM mtl_system_items MSI, wip_requirement_operations WRO
WHERE MSI.inventory_item_id = WRO.inventory_item_id
AND MSI.organization_id = WRO.organization_id
AND MSI.segment1 IN('&component1','&component2')
AND WRO.organization_id = &org_id
AND WRO.wip_entity_id IN(SELECT wip_entity_id
                           FROM WIP_ENTITIES
                           WHERE wip_entity_name = '&discrete_job'
                           AND organization_id = &org_id);

```

ITEM	PER_ASSY	REQD	ISSUED	OPEN
Body	1	5	4	1
Nib	1	5	4	1

Check: Although the job Start Quantity is 4, the value in the required_quantity field for the components has increased to accommodate the overcompleted quantity of 5 units.

- (7) WIP: Discrete > Close Discrete Jobs > Close Discrete Jobs (Form)
 Query the job. Tools > Close
 This will launch the job close request.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	110	-22	0
Resource	20	20	0	0
Total	108	130	-22	0

The residual negative balance on the job is relieved as variance upon job closure.

Case 2

- (1) WIP: Discrete > Discrete Jobs
 Create a job for Fountain Pen.
 Start Quantity = 4.
- (2) WIP: Material Transactions > WIP Material Transactions
 For the job defined above, perform WIP component issue for 4 units of Body and 4 units of Nib from the Stores subinventory. (Recall: The Supply Type of the components is set to **Push** in the BOM).

WIP: Discrete > WIP Value Summary
 Review Value Summary for the above job.
 We find :

Cost Element	Cost Incurred	Cost Relieved	Variance Relieved	Net Activity
Material	88	0	0	88
Resource	0	0	0	0
Total	88	0	0	88

- (3) WIP: Move Transactions > Move Transactions
 Perform a Move Transaction for Qty = 2.
 From Op 10 Queue to Op 20 To move.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	0	0	88
Resource	8	0	0	8
Total	96	0	0	96

Reason:

Cost Incurred = Resource charge = Resource Rate * Resource Usage * Qty.
 For 2 units of Body through Grinder + 2 units of Nib (and Body) through Fitter,
 Resource charge = (\$2/HR*1HR/unit*2units) + (\$2/HR*1HR/unit*2units) = \$8.

- (4) WIP: Material Transactions > Completion Transaction
 Perform a final completion transaction for Qty = 2.
 Final Completion = Yes.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	88	0	0
Resource	8	8	0	0
Total	96	96	0	0

Reason:

Final completion relieves the **entire** cost incurred by the job. As such the residual balance remaining on the job is zero.

Note:

Contrast this with **Case 1**, step (4). In case of an easy completion (using the Move Transactions form), or using the Completion Transactions form with the Final Completion flag **unchecked**, only cost proportional to the completed quantity will be relieved from the job, even if that leaves a residual balance on the job. Final Completion on the other hand, will ensure that no residual balance remains on the job.

Let us review the unit cost of Fountain Pen at this stage.

Material Unit Cost = $[(5 * \$22) + \$88] / (5 + 2) = \$28.285714285714285714$

Resource Unit Cost = $[(5 * \$4) + \$8] / (5 + 2) = \$4$

Therefore unit cost = $\$32.285714285714285714$

Verify this from either of the two:

Inventory: Costs > Item Costs

Cost: Item Costs > Item Costs

- (5) WIP: Move Transactions > Move Transactions
 Perform a Move Transaction for Qty = 2.
 From Op 10 Queue to Op 20 To move.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	88	0	0
Resource	16	8	0	8
Total	104	96	0	8

Reason:

Only resource charge is incurred for 2 units processed through Operations 10 and 20.

- (6) WIP: Material Transactions > Completion Transactions
 Perform a completion transaction for Qty = 3.
 Overcompletion = Yes
 Final Completion = Yes

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	88	0	0
Resource	20	20	0	0
Total	108	108	0	0

Reason:

Only resource charge is incurred for the additional 1 unit of Fountain Pen, assuming it went through Operations 10 and 20.

Resource charge = Resource Rate * Resource Usage * Qty.

For 1 unit of Body through Grinder + 1 unit of Nib (and Body) through Fitter,

Resource charge = (\$2/HR*1HR/unit*1unit) + (\$2/HR*1HR/unit*1unit) = \$4.

Note:

When the components in the job have Supply Type as **Push** and these are already issued to the job, and you perform an assembly completion transaction with Overcompletion **On** and Final Completion **On** in the last completion transaction of the job, then Material cost is neither incurred nor relieved for the additional completed units. However, resources are charged for all the operations and for all the completed units.

This is where **Case 2** differs from **Case 1**. When you do **not** perform a Final Completion when doing an Overcompletion (as in **Case 1**), the Material Cost (equivalent to the additional assembly units being completed) is relieved from the job, and this cost over-relief is posted as variance when the job is closed.

Let us review the unit cost of Fountain Pen at this stage.

Material Unit Cost = $[(7 * \$28.285714285714285714) + 0] / (7 + 3) = \19.8

Resource Unit Cost = $[(7 * \$4) + \$12] / (7 + 3) = \$4$

Therefore unit cost = \$23.8

Verify this from either of the two:

Inventory: Costs > Item Costs

Cost: Item Costs > Item Costs

A Special Note on Case 2: If in **Case 2**, step (6), we had set Final Completion = No, then the outcome would have reflected in the WIP Value Summary as shown below :

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	88	154	0	-66
Resource	20	20	0	0
Total	108	174	0	-66

A Final Completion followed by an Overcompletion would induce the system to relieve excess Material Cost for the entire Overcompleted quantity of Fountain Pen (= 3 units = \$66). The negative residual balance on the job of \$66 would again be posted as variance upon job closure. **Internal bug 2710904 has been logged to track this issue in Case 2.**

Case 3

- (1) WIP: Discrete > Discrete Jobs
Create a Discrete job for Fountain Pen.
Start Quantity = 4.
- (2) Job/Schedule Details > Material Requirements
Query the job created in (1). Under the Supply tab, set the Supply Type of the components as follows:
Body = Operation Pull
Nib = Assembly Pull
Save your work.
- (3) WIP: Move Transactions > Move Transactions
Perform a Move Transaction for Qty = 5.
From Op 10 Queue to Op 20 To move.
Overcomplete = Yes

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	50	0	0	50
Resource	20	0	0	20
Total	70	0	0	70

Reason:

Material charge = 5 units of Body backflushed = 5*\$10 = \$50

Note: Component Nib having Supply Type as Assembly Pull will be backflushed only upon assembly completion.

Resource charge = 5 units through Operations 10 and 20

$$= (\$2/\text{HR} * 1\text{HR}/\text{unit} * 5\text{units}) + (\$2/\text{HR} * 1\text{HR}/\text{unit} * 5\text{unit}) = \$20$$

- (4) WIP: Material Transactions > Completion Transactions
Perform an assembly completion for Qty = 5.
Final Completion = Yes.

Note: Although we are completing a quantity greater than the job Start Quantity, there is no need to check the Overcompletion option, because we have already declared that option in step (3) above.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	110	110	0	0
Resource	20	20	0	0
Total	130	130	0	0

Reason:

Material charge = 5 units of Nib backflushed on assembly completion= 5*\$12 = \$60
 Therefore, the Material charge increases to \$110.
 This being a Final Completion, all the cost incurred is relieved.

Note:

Review the unit cost of Fountain Pen.
 Material Unit Cost = $[(10 \times \$19.8) + \$110] / (10 + 5) = \$20.533333333333333$
 Resource Unit Cost = $[(10 \times \$4) + \$20] / (10 + 5) = \$4$
 Therefore unit cost = $\$24.533333333333333$
 Verify this from either of the two:
 Inventory: Costs > Item Costs
 Cost: Item Costs > Item Costs

This was a fairly straightforward case of assembly completion where the components had supply type as **Pull** and so were available for backflushing just when they were needed. The complexity was further reduced by not doing a last transaction overcompletion.

Case 4

This case is similar to **Case 3**. However, as in the first two cases, here too we shall attempt an overcompletion in the last transaction and check the results.

- (1) WIP: Discrete > Discrete Jobs
 Create a Discrete job for Fountain Pen.
 Start Quantity = 4.
- (2) WIP: Job/Schedule Details > Material Requirements
 Query the job created in (1). Under the Supply tab, set the Supply Type of the components as follows:
 Body = Operation Pull
 Nib = Assembly Pull
 Save your work.
- (3) WIP: Move Transactions > Move Transactions
 Perform a Move Transaction for Qty = 5.
 From Op 10 Queue to Op 20 To move.
 Overcomplete = Yes

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	50	0	0	50
Resource	20	0	0	20
Total	70	0	0	70

Reason:

Material charge = 5 units of Body backflushed = $5 * \$10 = \50

Resource charge = 5 units through Operations 10 and 20

$$= (\$2/\text{HR} * 1\text{HR}/\text{unit} * 5\text{units}) + (\$2/\text{HR} * 1\text{HR}/\text{unit} * 5\text{unit}) = \$20$$

- (4) WIP: Material Transactions > Completion Transactions

Perform an assembly completion for Qty = 6.

Overcompletion = Yes

Final Completion = Yes.

Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	132	122	0	10
Resource	24	24	0	0
Total	156	146	0	10

Reason:

Material charge = 1 unit of Body backflushed (to account for the excess 1 unit completed in the last transaction) + 6 units of Nib backflushed = $(1 * \$10) + (6 * \$12) = \$82$

Resource charge = 1 additional unit through Operations 10 and 20

$$= (\$2/\text{HR} * 1\text{HR}/\text{unit} * 1\text{unit}) + (\$2/\text{HR} * 1\text{HR}/\text{unit} * 1\text{unit}) = \$4$$

Note:

Due to the last transaction being a Final Completion transaction, all the cost incurred in the job is relieved on assembly completion. But there is an exception! The Material Cost Incurred is not completely relieved.

Cost Relieved = 5 units of Body + 6 units of Nib = $[(5 * \$10) + (6 * \$12)] = \$122$.

The cost of the Operation Pull component is **not** relieved for the excess 1 unit of assembly overcompleted in the last transaction. Under the given conditions, this behaviour has been observed to occur consistently in Release 11i.

Internal Bug 2706581 had been logged for tracking this issue. The fix will be included in Release 11.5.9.

Let us review the final unit cost of the Fountain Pen.

$$\begin{aligned} \text{Material Unit Cost} &= [(15 * \$20.533333333333333) + \$122] / (15 + 6) \\ &= \$20.476190476190476190 \end{aligned}$$

$$\text{Resource Unit Cost} = [(15 * \$4) + \$24] / (15 + 6) = \$4$$

Therefore unit cost = $\$24.476190476190476190$

Verify this from either of the two:

Inventory: Costs > Item Costs

Cost: Item Costs > Item Costs

- (5) WIP: Discrete > Close Discrete Jobs > Close Discrete Jobs (Form)
 Query the job. Tools > Close
 This will launch the job close request.
 Review Value Summary for the job.

Cost Element	Costs Incurred	Costs Relieved	Variances Relieved	Net Activity
Material	132	122	10	0
Resource	24	24	0	0
Total	156	146	10	0

The residual positive balance on the job is relieved as variance upon job closure.

SUMMARY OF FINDINGS

The above tests help us arrive at an understanding on the following points :

- (1) Whenever a completion transaction is performed, cost will be relieved from the WIP job.
 This implies that the completion subinventory elemental accounts will be debited and the corresponding WIP Valuation accounts will be credited.
- (2) The cost per unit of the completed assembly is **recalculated** using the formula

$$\frac{[(\text{Qty on Hand} * \text{Average unit cost}) + (\text{Completion cost of new units})]}{(\text{Qty on Hand} + \text{new units completed})}$$

$$= \frac{[(\text{Qty on Hand} * \text{Average unit cost}) + (\text{Cost relieved since last completion transaction on this job})]}{(\text{Qty on Hand} + \text{new units completed})}$$
These calculations are by element by level.
- (3) If all the **Push** components have already been issued to the job in their required quantities, then for an **Easy Completion** i.e., a completion through the Move Transaction form or a completion through the Completion Transactions form with the Final Completion box unchecked – only the cost incurred by the completed quantity will be relieved from the job even if that leaves a positive residual balance on the job. Final Completion, on the other hand will ensure that no residual balance remains on the job.
- (4) When the last assembly completion is performed with Overcompletion **On** and Final Completion **Off**, then Material Cost equivalent to the additional assembly units being completed is relieved from the job. However, if the previous completion transaction had Overcompletion **Off** and Final Completion **On** and was followed by a Overcompletion **On** and Final Completion **Off** transaction, then the Material Cost relieved would exceed the Material Cost incurred by the exact number of assembly units overcompleted in that Transaction (not just the excess units completed). If the Push components had already been issued to the job, then this would result in a negative balance remaining with the job. This residual amount is posted as variance when the job is closed.

- (5) When we have Overcompletion **On** and Final Completion **On** for the last completion transaction, then
- for components with Supply Type as **Push**, Material Cost is neither incurred nor relieved for the excess quantity overcompleted. This behaviour has been consistently observed in Release 11i and Internal bug 2710904 has been logged for tracking this issue.
 - for components with Supply Type as **Assembly Pull**, Material Cost is incurred as well as relieved for the excess quantity overcompleted.
 - for components with Supply Type as **Operation Pull**, Material Cost is incurred but **not** relieved for the excess quantity overcompleted. This behaviour has been consistently observed in Release 11i and Internal bug 2706581 had been logged for tracking this issue. The fix for this issue will now be incorporated in Release 11.5.9.
 - The Resource Costs (for resources with charge Type as WIP Move) are always incurred and relieved for the overcompleted quantity.
- (6) Based on the results of the test case we simulated, we find the following :

	Supply Type of Components	In the final transaction...		Final outcome
		Final Completion	Overcompletion	Costs Incurred (CI) versus Costs Relieved (CR)
Case 1	Push	No	Yes	CI < CR
Case 2	Push	Yes	Yes	CI = CR
Case 3	Pull	Yes	No	CI = CR
Case 4	Pull	Yes	Yes	CI > CR

CONCLUSION

A completion transaction will always relieve cost from the job. A Final Completion will relieve the entire cost incurred by the job. The system allows a Final Completion to be performed at any stage during the processing of a job. The costing implications of performing a Final Completion must determine its timing. The additional cost relieved since the last completion (against the given job), coupled with the number of units completed in that transaction will determine the unit cost of the completion for that transaction.

ABOUT THE AUTHOR

Saumit Mandal, CPIM is a Senior Support Analyst and a BDE for Core Manufacturing at Oracle's India Support Center, Bangalore.



White Paper: 11i WIP Assembly Completion & Overcompletion Transactions in Average Costing
December, 2002

Author: Saumit Mandal, CPIM

Contributing Authors: N/A

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Oracle is a registered trademark of Oracle Corporation. Various product and service names referenced herein may be trademarks of Oracle Corporation. All other product and service names mentioned may be trademarks of their respective owners.

Copyright © 2002 Oracle Corporation
All rights reserved.