Webinar: Subroutines & Macros

This webinar applies to ProModel & MedModel

ProModel®

Presenters:
Ken Davis, Sr. Consultant & Project Manager
Jennifer Cowden, Director Consulting Services

January 25, 2022

ProModel®
Better Decisions—Faster

Purpose of this webinar

- Why would you use a Macro or Subroutine?
- Why one over the other?
- What types of Macros are there?
- What types of Subroutines are there?

- This is NOT a tutorial of the basics of logic statements, neither Macros nor Subroutines
  - Knowledge of basic ProModel/MedModel is assumed

Topics

- Macros
  - Types
    - Scenario Parameters
    - Text substitutions, Constant values
    - Logic statements (like subroutines)
    - Resource Group
    - Arrival Schedule record
    - Shift Schedule record
- Subroutines (Logic statements)
  - Types
    - None
    - Integer, Real (RETURN a value)
    - Interactive
  - How run
    - “Executed”
    - ACTIVATEd

Which should we use?

- Depends…
  - Functionality!
  - Personal programming preference!!

Why might we use Either Subroutine or Macro

- Multiple lines of Logic
- Logic that will be executed at multiple places
  (Operation, Move, Arrival, any logic window)
  - Need only change the logic in one place

- Therefore, EITHER is OK!

Why <u>Macro</u> or Not

- Depends upon the purpose… Can only be used if:
  - Scenario Parameters
  - Text substitutions, Constant values
  - Resource Groups
  - Arrival Schedule records
  - Shift Schedule records

Why Subroutine or Not

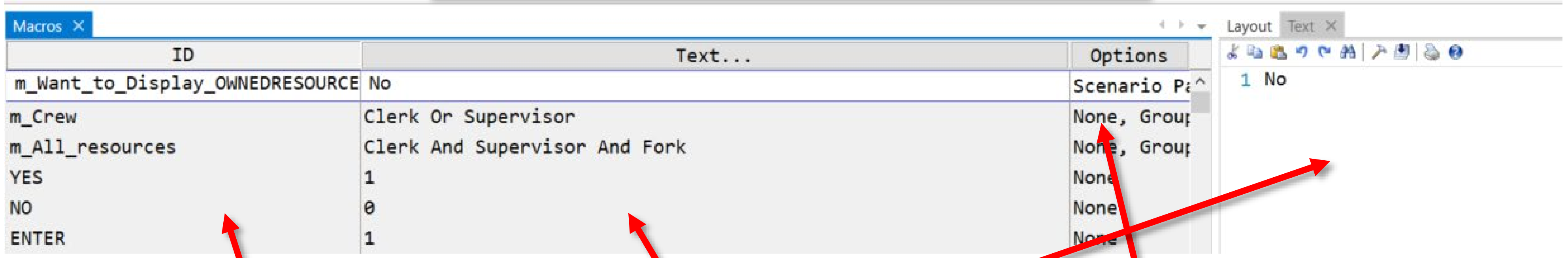
- Depends upon the purpose… Can only be used if:
  - Want to pass a parameter into the logic
    - When one or more items is of interest to control the logic
    - This is the classic form of a subroutine from, say, FORTRAN
    - However, of course…
      Could just use Attributes to pass the information.
      But then it would be less obvious in the sub execution statement
  - Interactive … Want to change something during a simulation run

Run Speed

- It used to be …
  (in a Galaxy far, far away; well, really, back when computers were not as powerful as now)
  that models ran faster using Macros rather than Subroutines
- Now… No big deal! … So, It doesn’t matter.

How do you Build a Macro?

The Macro edit table

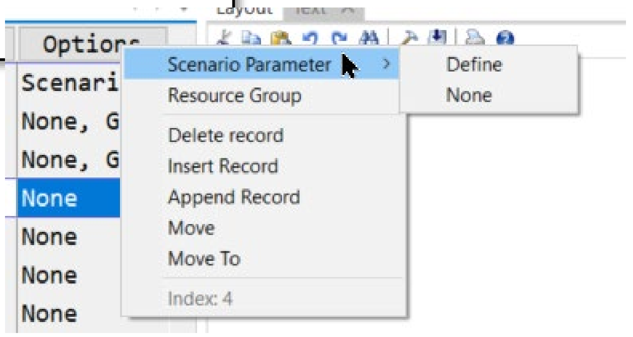


ID	Text...	Options
m_Want_to_Display_OWNEDRESOURCE	No	Scenario Pa
m_Crew	Clerk Or Supervisor	None, Group
m_All_resources	Clerk And Supervisor And Fork	None, Group
YES	1	None
NO	0	None
ENTER	1	None

Enter a name

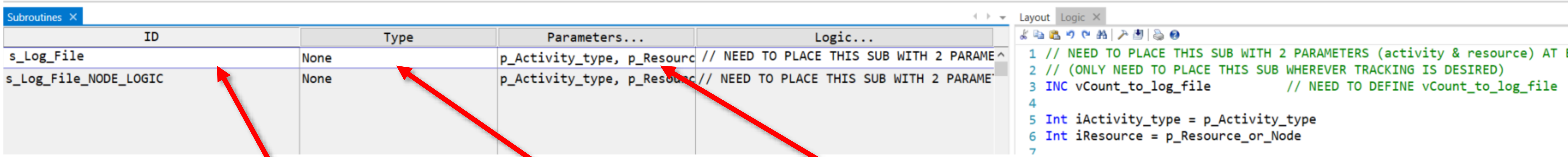
Enter the text of the Macro

Enter the Macro Options…



How do you Build a Subroutine?

The subroutines edit table



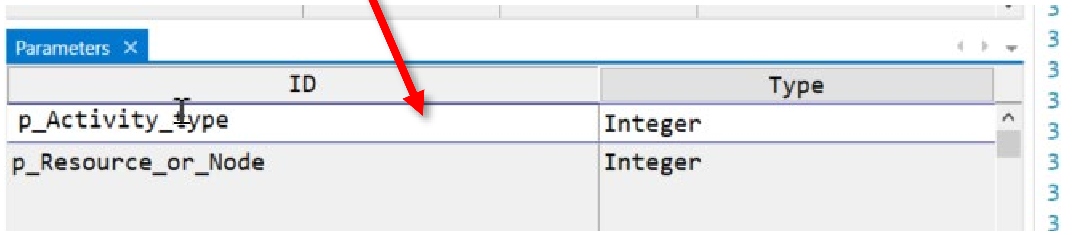
ID	Type	Parameters...	Logic...
s_Log_File	None	p_Activity_type, p_Resourc	// NEED TO PLACE THIS SUB WITH 2 PARAME
s_Log_File_NODE_LOGIC	None	p_Activity_type, p_Resourc	// NEED TO PLACE THIS SUB WITH 2 PARAME

```
// NEED TO PLACE THIS SUB WITH 2 PARAMETERS (activity & resource) AT
// (ONLY NEED TO PLACE THIS SUB WHEREVER TRACKING IS DESIRED)
INC vCount_to_log_file          // NEED TO DEFINE vCount_to_log_file

Int iActivity_type = p_Activity_type
Int iResource = p_Resource_or_Node
```

Enter a name

The type may be:
None
Real
Integer
Interactive

Enter parameters if needed and enter the logic

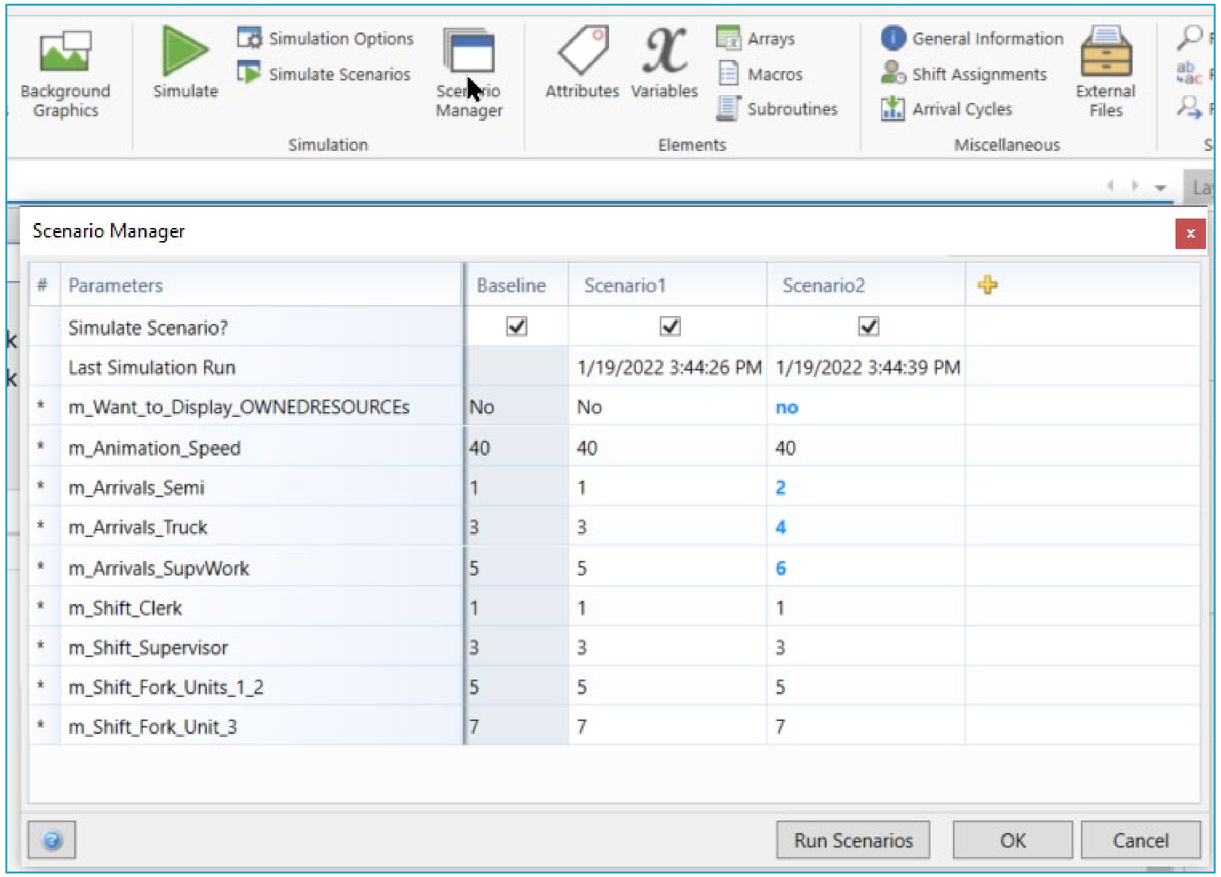


ID	Type
p_Activity_type	Integer
p_Resource_or_Node	Integer

ProModel®
Better Decisions—Faster

Macro Type: Scenario Parameter

- Provide to the Scenario Manager <u>what to change between Scenarios</u>

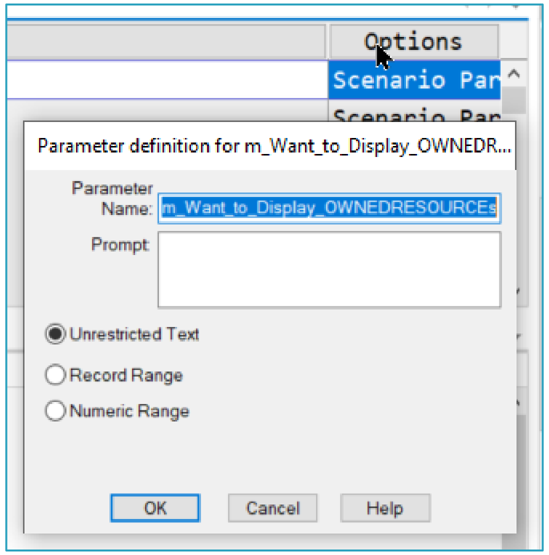


#	Parameters	Baseline	Scenario1	Scenario2
	Simulate Scenario?	✓	✓	✓
	Last Simulation Run		1/19/2022 3:44:26 PM	1/19/2022 3:44:39 PM
*	m_Want_to_Display_OWNEDRESOURCEs	No	No	**no**
*	m_Animation_Speed	40	40	40
*	m_Arrivals_Semi	1	1	**2**
*	m_Arrivals_Truck	3	3	**4**
*	m_Arrivals_SupvWork	5	5	**6**
*	m_Shift_Clerk	1	1	1
*	m_Shift_Supervisor	3	3	3
*	m_Shift_Fork_Units_1_2	5	5	5
*	m_Shift_Fork_Unit_3	7	7	7

ProModel®
Better Decisions—Faster

Macro Type: Scenario Parameter

- Specify how to control the Scenario Parameter



Parameter definition for m_Want_to_Display_OWNEDR...

Parameter Name: m_Want_to_Display_OWNEDRESOURCEs

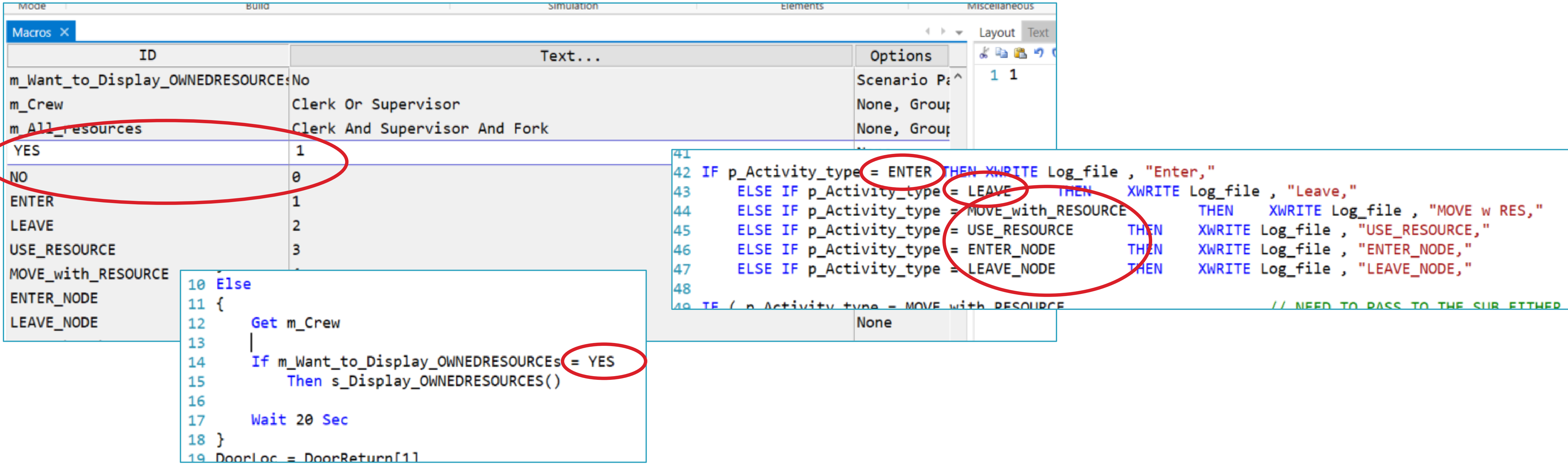
Prompt:

(●) Unrestricted Text
( ) Record Range
( ) Numeric Range

- Example: Animation speed, others

Macro Type: Text substitutions, Constant values

- Allow using <u>real words </u>rather than just numbers in IF/THEN statements and data import (Array Import)
- Whenever there's a List of choices (e.g. Red/White/Blue , NOT 1/2/3)



ID	Text...	Options
m_Want_to_Display_OWNEDRESOURCE:	No	Scenario Pa
m_Crew	Clerk Or Supervisor	None, Group
m_All_resources	Clerk And Supervisor And Fork	None, Group
YES	1	
NO	0	
ENTER	1	
LEAVE	2	
USE_RESOURCE	3	
MOVE_with_RESOURCE		
ENTER_NODE		
LEAVE_NODE		None

```
Else
{
     Get m_Crew
     |
     If m_Want_to_Display_OWNEDRESOURCEs = YES
         Then s_Display_OWNEDRESOURCES()

     Wait 20 Sec
}
DoorLoc = DoorReturn[1]
```

```
IF p_Activity_type = ENTER THEN XWRITE Log_file , "Enter,"
    ELSE IF p_Activity_type = LEAVE     THEN    XWRITE Log_file , "Leave,"
    ELSE IF p_Activity_type = MOVE_with_RESOURCE        THEN    XWRITE Log_file , "MOVE w RES,"
    ELSE IF p_Activity_type = USE_RESOURCE      THEN    XWRITE Log_file , "USE_RESOURCE,"
    ELSE IF p_Activity_type = ENTER_NODE        THEN    XWRITE Log_file , "ENTER_NODE,"
    ELSE IF p_Activity_type = LEAVE_NODE        THEN    XWRITE Log_file , "LEAVE_NODE,"

IF ( p_Activity_type = MOVE_with_RESOURCE                     // NEED TO PASS TO THE SUB EITHER
```

ProModel®
Better Decisions—Faster

CONFIDENTIAL

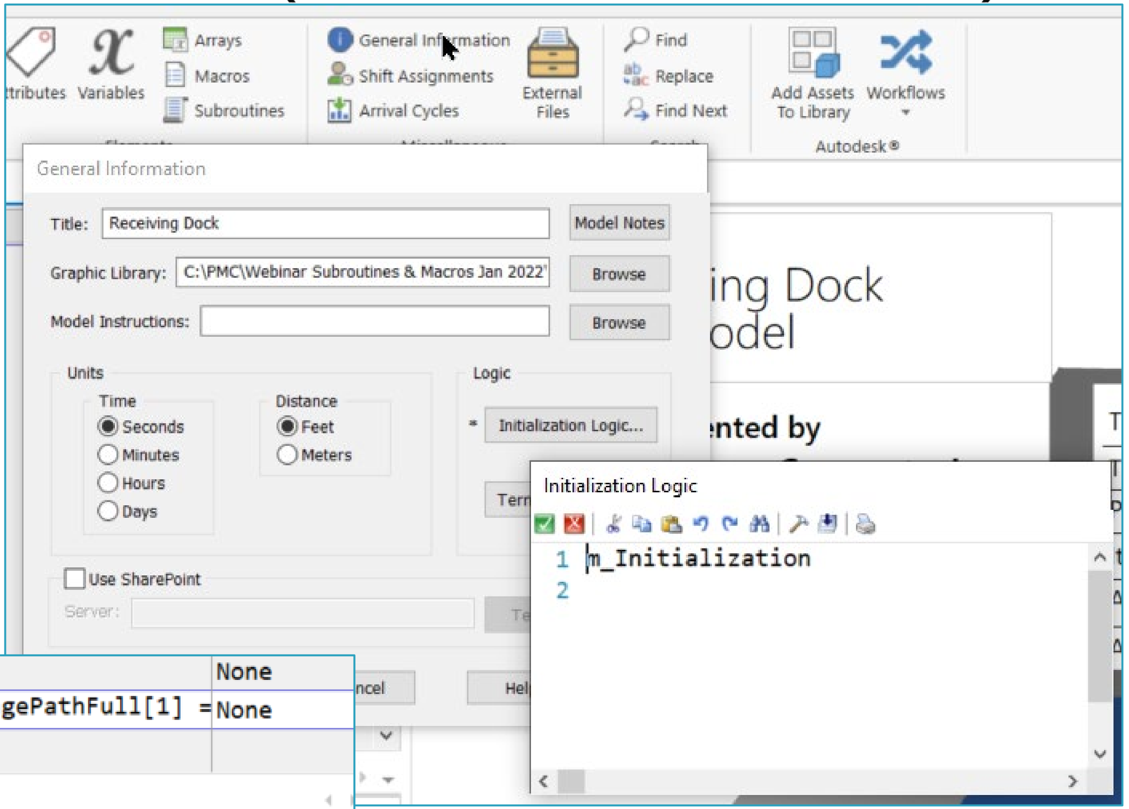
Macro Type: Text substitutions, Constant values

- Great anywhere there's a <u>List of Stuff</u>
  - e.g. …
    - Part number
    - Truck route destinations
- Remember: Standard ProModel expression naming must be done
  - Cannot begin with a numeral (e.g. 123 → PN_123)
  - Cannot be a reserved word (e.g. EXIT → LEAVE)

- Example: Yes/No, production Grade, others

Macro Type: Logic statements (like subroutines)

- Presentation Script



General Information

Title: Receiving Dock

Graphic Library: C:\PMC\Webinar Subroutines & Macros Jan 2022

Model Instructions:

Units — Time: [✓ Seconds] Minutes Hours Days

Distance: [✓ Feet] Meters

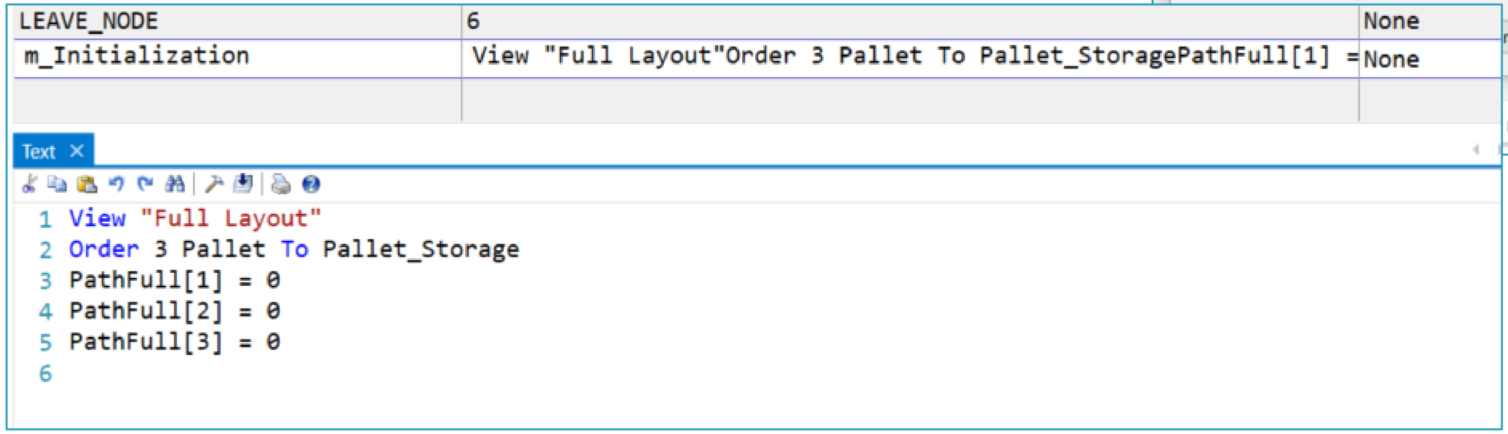
Logic: Initialization Logic...

Use SharePoint

Server:

Initialization Logic

```
m_Initialization
```



LEAVE_NODE	6	None
m_Initialization	View "Full Layout"Order 3 Pallet To Pallet_StoragePathFull[1] =	None

Text

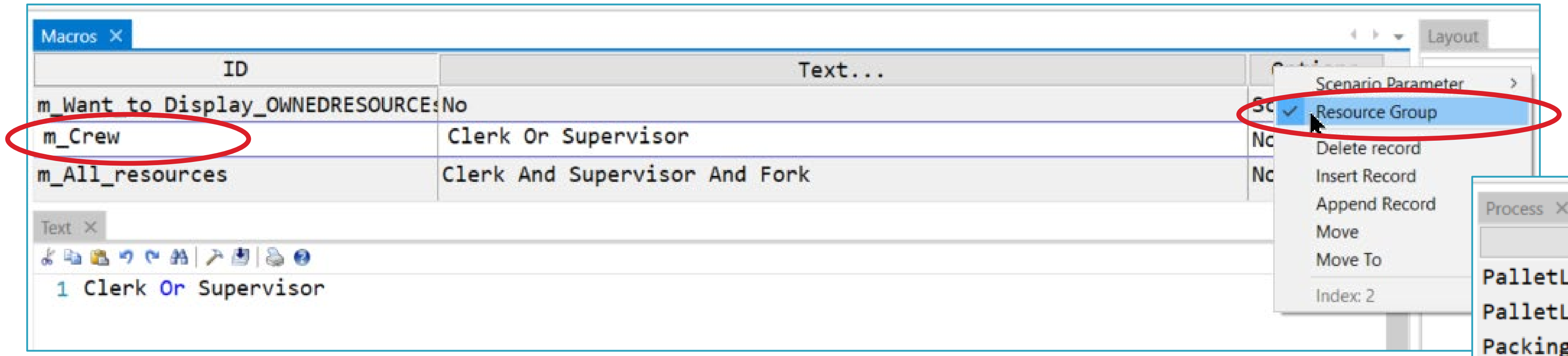
```
View "Full Layout"
Order 3 Pallet To Pallet_Storage
PathFull[1] = 0
PathFull[2] = 0
PathFull[3] = 0
```

ProModel® Better Decisions—Faster

CONFIDENTIAL

Macro Type: Resource Group

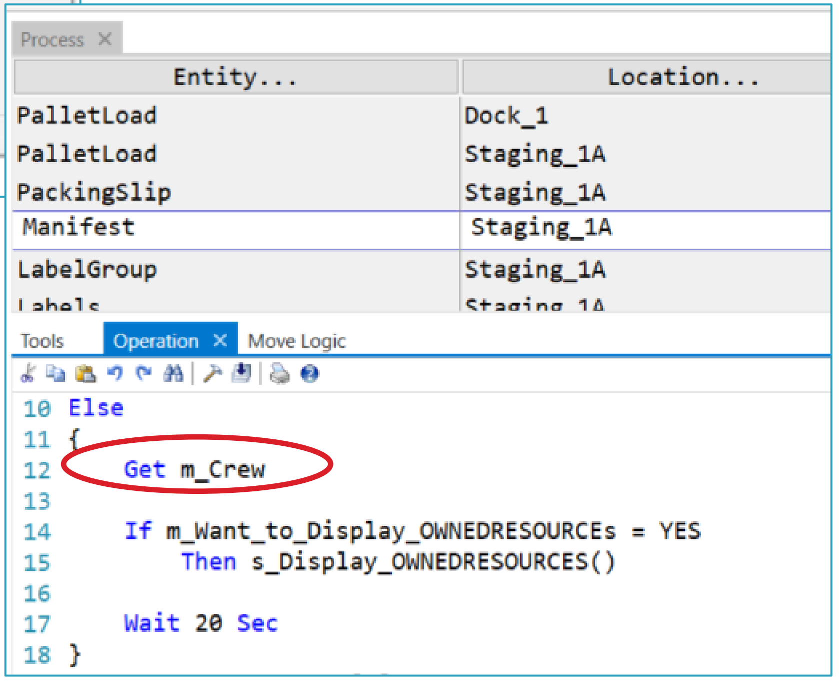
- Easily list multiple Resources including Qty & ANDs & ORs
  - Can create multiple grouped choices of GET & USE for IF/THENs



ID	Text...	
m_Want_to_Display_OWNEDRESOURCES	No	Sc
m_Crew	Clerk Or Supervisor	Nc
m_All_resources	Clerk And Supervisor And Fork	Nc

```
Clerk Or Supervisor
```

- Example: GET, just for aggregate reporting



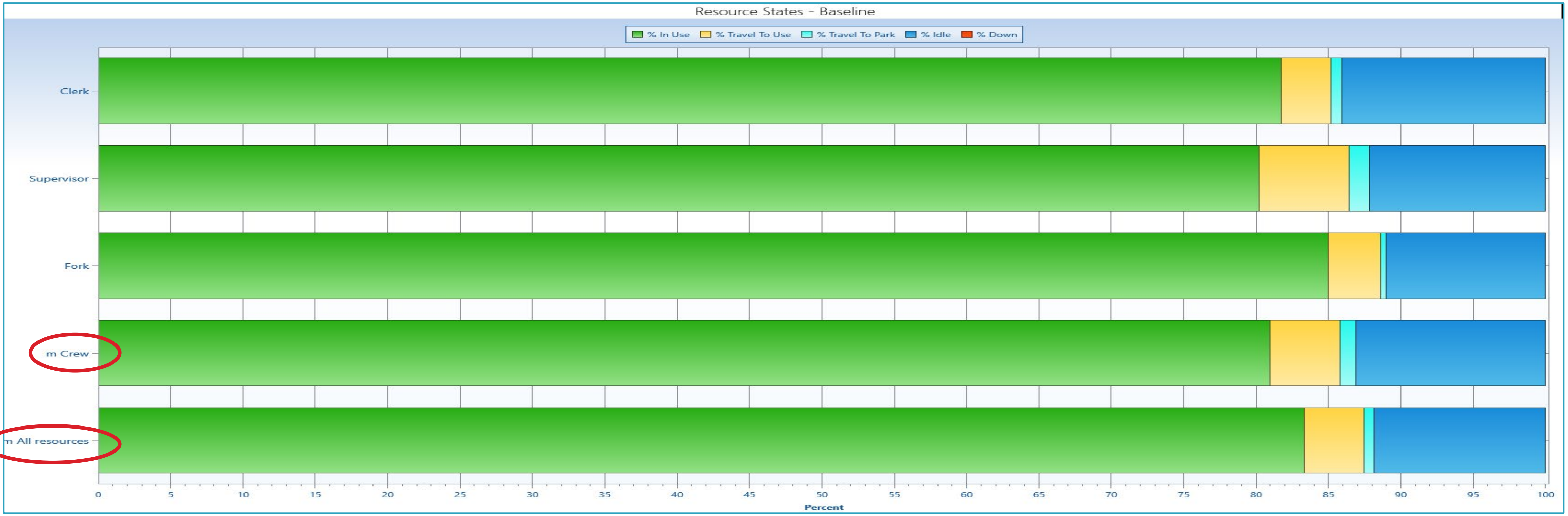
Entity...	Location...
PalletLoad	Dock_1
PalletLoad	Staging_1A
PackingSlip	Staging_1A
Manifest	Staging_1A
LabelGroup	Staging_1A
Labels	Staging_1A

```
Else
{
    Get m_Crew

    If m_Want_to_Display_OWNEDRESOURCEs = YES
        Then s_Display_OWNEDRESOURCES()

    Wait 20 Sec
}
```

ProModel®
Better Decisions—Faster

Macro Type: Resource Group

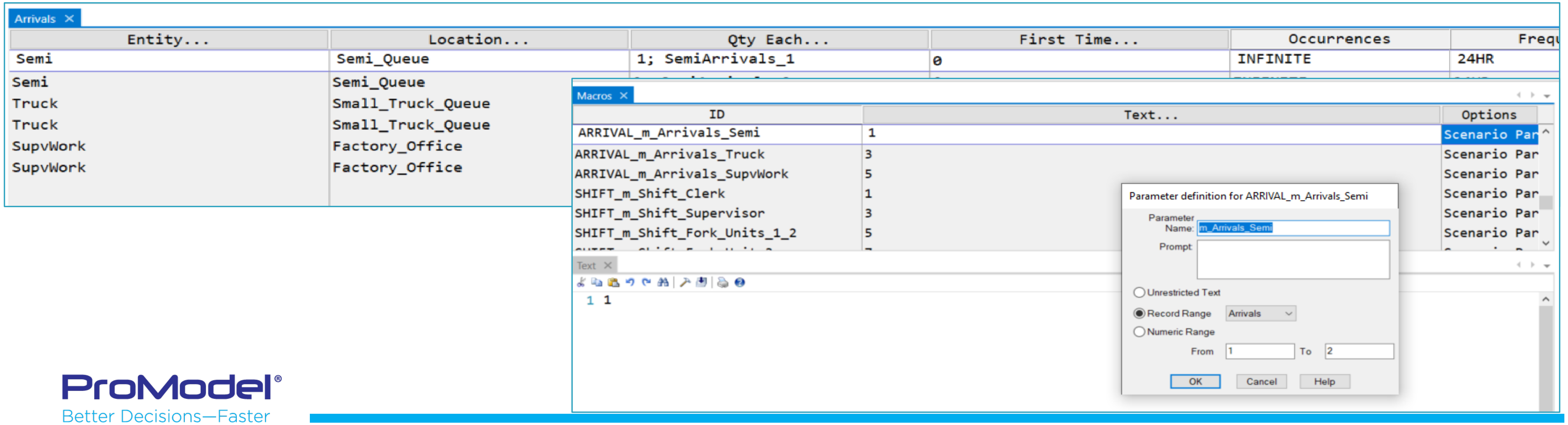
- Extra benefit: Output Viewer shows Utilization & State of the GROUP in addition to each individual Resource
  - Therefore, Can use this purely to show the aggregate statistics of multiple Resources



ProModel®
Better Decisions—Faster

Macro Type: Arrival Schedule record

- Use the Scenario Manager to specify which Arrival Records to run
  - Within a list of Arrival Records … Which record do we use?
  - Specify a default in the Text field for the Macro.
  - ALL Arrival Records are used if just Simulate!!! (unless are DISABLEd)
  - If Disabled, cannot use in Scenario
- "Arrival_" is automatically added to the beginning of the Macro name



Arrivals

Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequ
Semi	Semi_Queue	1; SemiArrivals_1	0	INFINITE	24HR
Semi	Semi_Queue				
Truck	Small_Truck_Queue				
Truck	Small_Truck_Queue				
SupvWork	Factory_Office				
SupvWork	Factory_Office				

Macros

ID	Text...	Options
ARRIVAL_m_Arrivals_Semi	1	Scenario Par
ARRIVAL_m_Arrivals_Truck	3	Scenario Par
ARRIVAL_m_Arrivals_SupvWork	5	Scenario Par
SHIFT_m_Shift_Clerk	1	Scenario Par
SHIFT_m_Shift_Supervisor	3	Scenario Par
SHIFT_m_Shift_Fork_Units_1_2	5	Scenario Par

Text

```
1
```

Parameter definition for ARRIVAL_m_Arrivals_Semi

Parameter Name: m_Arrivals_Semi

Prompt:

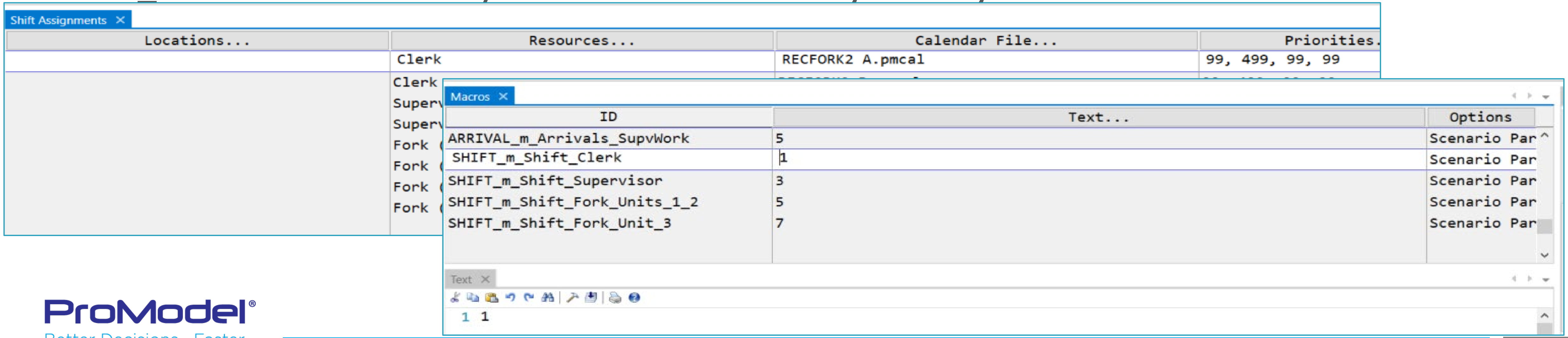
Unrestricted Text [✓ Record Range] Arrivals Numeric Range

From 1 To 2

OK Cancel Help

Macro Type: Shift Schedule record

- Use the Scenario Manager to specify which Shift Assignment Records to run
  - Within a list of Shift Assignment Records … Which record do we use?
  - Specify a default in the Text field for the Macro.
  - ALL Shift Assignment Records are used if just Simulate!!! (unless are DISABLEd)
    - But can be very confusing… Overlapping shift specifications!?
  - If Disabled, cannot use in Scenario
- "Shift_" is automatically added to the beginning of the Macro name



Locations...	Resources...	Calendar File...	Priorities.
	Clerk	RECFORK2 A.pmcal	99, 499, 99, 99
	Clerk		
	Superv		
	Superv		
	Fork (		
	Fork (		
	Fork (		
	Fork (		

ID	Text...	Options
ARRIVAL_m_Arrivals_SupvWork	5	Scenario Par
SHIFT_m_Shift_Clerk	1	Scenario Par
SHIFT_m_Shift_Supervisor	3	Scenario Par
SHIFT_m_Shift_Fork_Units_1_2	5	Scenario Par
SHIFT_m_Shift_Fork_Unit_3	7	Scenario Par

```
1
```

ProModel®
Better Decisions—Faster

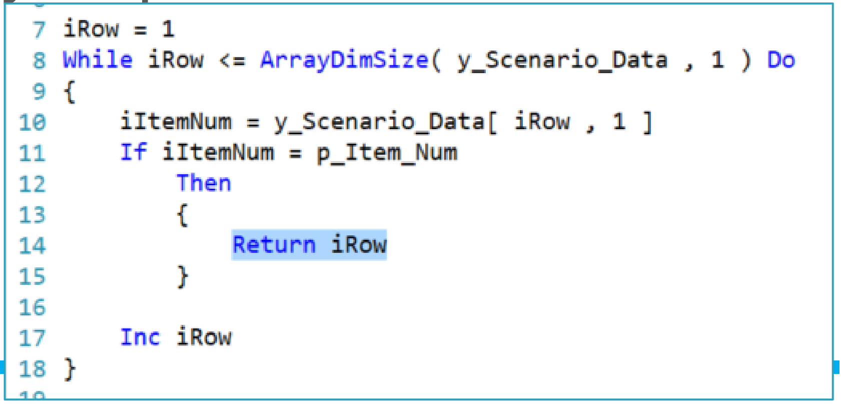
Subroutine: NONE

- Any set of logic statements can be put into a NONE subroutine to be Executed or ACTIVATEd anywhere in your model
- Might, but don’t need to, have input PARAMETERS
  - With no Parameters, it’s just like a Macro with logic statements
- Example: Display OWNEDRESOURCEs

Subroutine: INTEGER or REAL

- The purpose of REAL/INTEGER Subroutine is to RETURN a value (say, for what is the pass/fail status of an entity)
- But you could just assign an entity attribute for the desired value
- However, sometimes we'd like to determine a non-entity-related item
  - e.g. Which row of an array has the data of interest
  - e.g. Calculation of process time based upon LOTS OF STUFF

- Example: Complex lookup of which Excel array import row to use
  - s_Det_Scenario_input_row_from_Item_Num ...

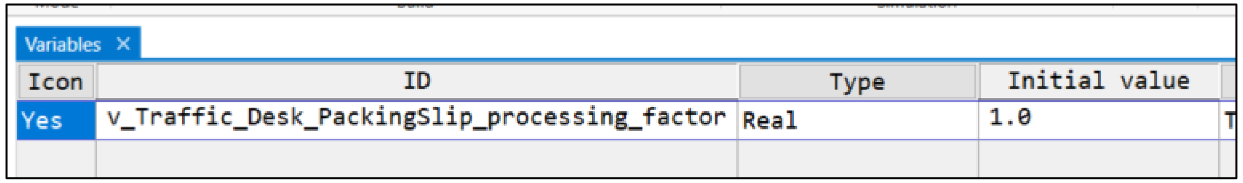


```
iRow = 1
While iRow <= ArrayDimSize( y_Scenario_Data , 1 ) Do
{
    iItemNum = y_Scenario_Data[ iRow , 1 ]
    If iItemNum = p_Item_Num
        Then
        {
            Return iRow
        }

    Inc iRow
}
```

ProModel® Better Decisions—Faster

Interactive Subroutine

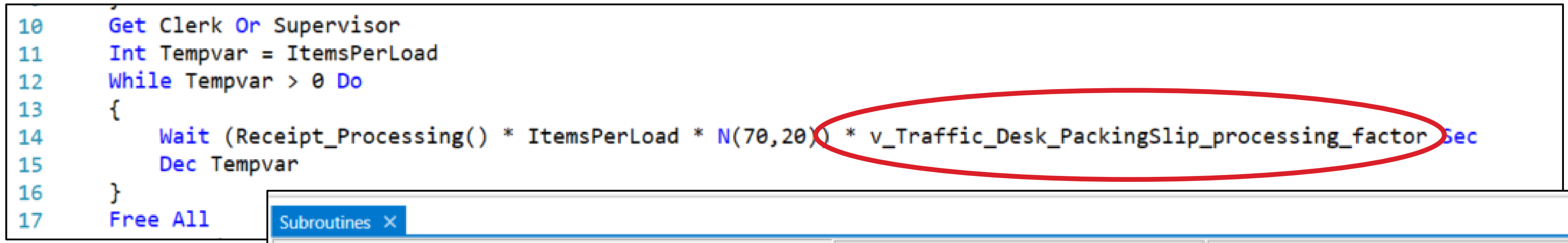
- Useful for ad-hoc runtime changes
  - Experiments, What-Ifs
  - Academic exercises



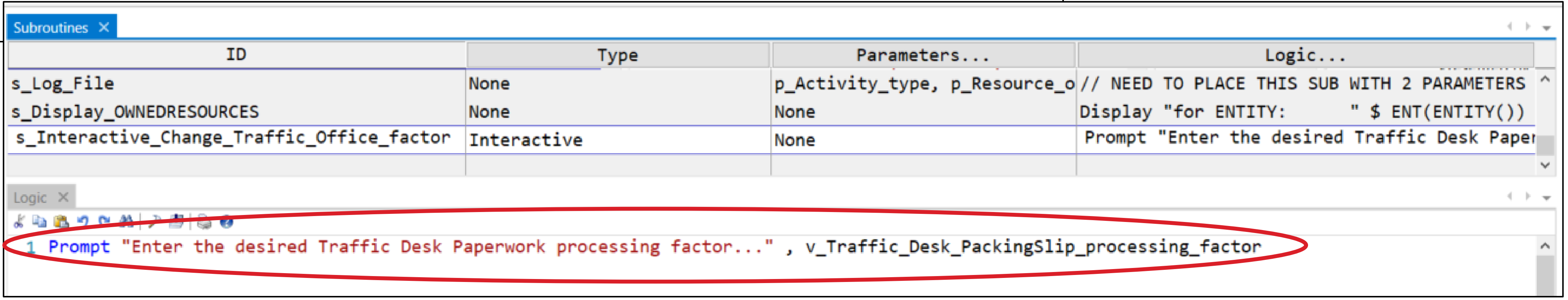
Variables

Icon	ID	Type	Initial value
Yes	v_Traffic_Desk_PackingSlip_processing_factor	Real	1.0

- Example … Change a Variable process time multiplicative factor



```
    Get Clerk Or Supervisor
    Int Tempvar = ItemsPerLoad
    While Tempvar > 0 Do
    {
        Wait (Receipt_Processing() * ItemsPerLoad * N(70,20)) * v_Traffic_Desk_PackingSlip_processing_factor Sec
        Dec Tempvar
    }
    Free All
```



Subroutines

ID	Type	Parameters...	Logic...
s_Log_File	None	p_Activity_type, p_Resource_o	// NEED TO PLACE THIS SUB WITH 2 PARAMETERS
s_Display_OWNEDRESOURCES	None	None	Display "for ENTITY: " $ ENT(ENTITY())
s_Interactive_Change_Traffic_Office_factor	Interactive	None	Prompt "Enter the desired Traffic Desk Paper

Logic

```
Prompt "Enter the desired Traffic Desk Paperwork processing factor..." , v_Traffic_Desk_PackingSlip_processing_factor
```

ProModel®
Better Decisions—Faster

Subroutines… ACTIVATEd vs. “Executed”

- An <u>ACTIVATEd</u> subroutine is run in the “background” “parallel” with entities being process through the Processing/Routing

- The logic in an ACTIVATEd sub is done <u>independently</u> of the entity that caused it to start
  - Therefore, the entity logic carries on regardless of the happenings in the ACTIVATEd sub
  - Examples:
    - For scheduling, an activated subroutine can be used to periodically examine inventory status (and reorder entities).
    - For animation, an activated subroutine can be used to choreograph a model presentation script with animation speeds and saved views.

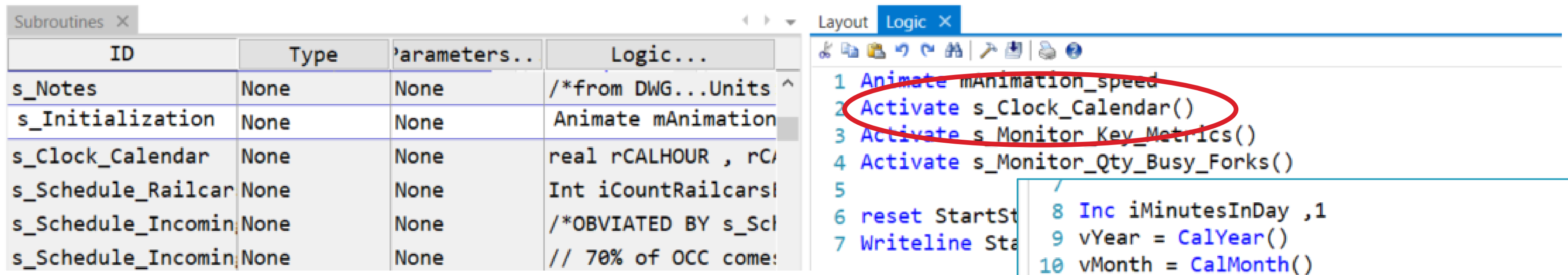
Subroutines… ACTIVATEd vs. “Executed”

- An ACTIVATEd subroutine CANNOT contain ENTITY ATTRIBUTEs

- The logic in an “Executed” sub must be fully completed before the logic can be continued!
    - If the “Executed” sub does not finish, then the entity that executed it is delayed!

Activated Subroutine

- Use the ACTIVATE statement

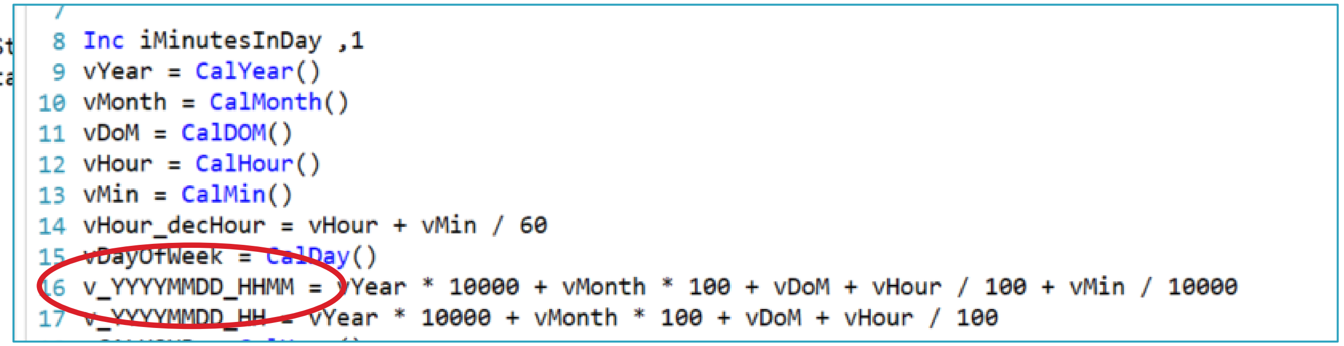


Subroutines

ID	Type	Parameters..	Logic...
s_Notes	None	None	/*from DWG...Units
s_Initialization	None	None	Animate mAnimation
s_Clock_Calendar	None	None	real rCALHOUR , rC
s_Schedule_Railcar	None	None	Int iCountRailcars
s_Schedule_Incomin	None	None	/*OBVIATED BY s_Sc
s_Schedule_Incomin	None	None	// 70% of OCC come:

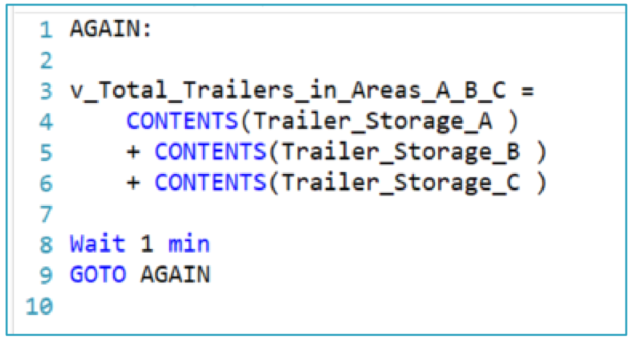
Layout Logic

```
Animate mAnimation_speed
Activate s_Clock_Calendar()
Activate s_Monitor_Key_Metrics()
Activate s_Monitor_Qty_Busy_Forks()

reset StartSt
Writeline Sta
```



```
Inc iMinutesInDay ,1
vYear = CalYear()
vMonth = CalMonth()
vDoM = CalDOM()
vHour = CalHour()
vMin = CalMin()
vHour_decHour = vHour + vMin / 60
vDayOfWeek = CalDay()
v_YYYYMMDD_HHMM = vYear * 10000 + vMonth * 100 + vDoM + vHour / 100 + vMin / 10000
v_YYYYMMDD_HH = vYear * 10000 + vMonth * 100 + vDoM + vHour / 100
```

- Examples:
  - ClockCalendar
    - In other logic: WAIT UNTIL v_YYYYMMDD_HHMM >= a_Start_Time
  - Monitor Key Metrics
  - Presentation script



```
AGAIN:

v_Total_Trailers_in_Areas_A_B_C =
    CONTENTS(Trailer_Storage_A )
    + CONTENTS(Trailer_Storage_B )
    + CONTENTS(Trailer_Storage_C )

Wait 1 min
GOTO AGAIN
```

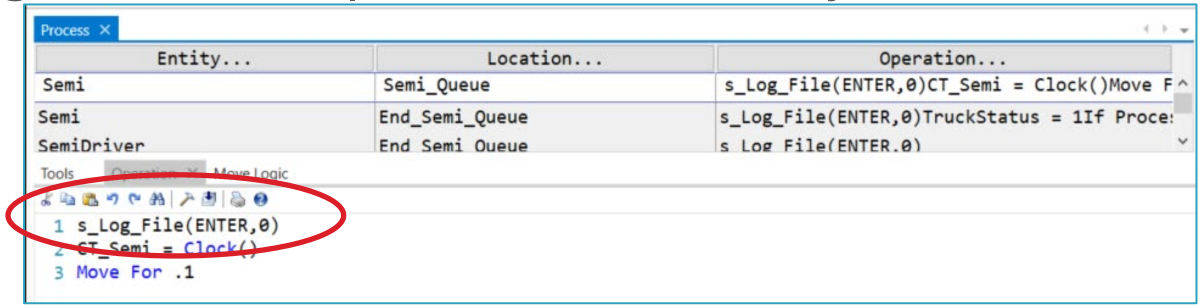
Animate 10
Wait 2 hr
View "Full"
Animate 50
Wait 5 hr
Animate 20

ProModel®
Better Decisions—Faster

“Executed” Subroutine

- Simply name the subroutine in a logic statement

- Example: Log File CSV creation
  - Pass Parameters …
    ENTER/LEAVE (in Receiving.Mod … all PRs)
    USE_RESOURCE (in Receiving.Mod … PR#4)
    MOVE_with_RESOURCE (in Receiving.Mod … PR#6)
  - Use of text substitution Macros (words, not numbers)
  - Refer to ProModel Corporation Website
    Solutions Café Webinar “Using Excel to Improve Model Analysis Webinar”



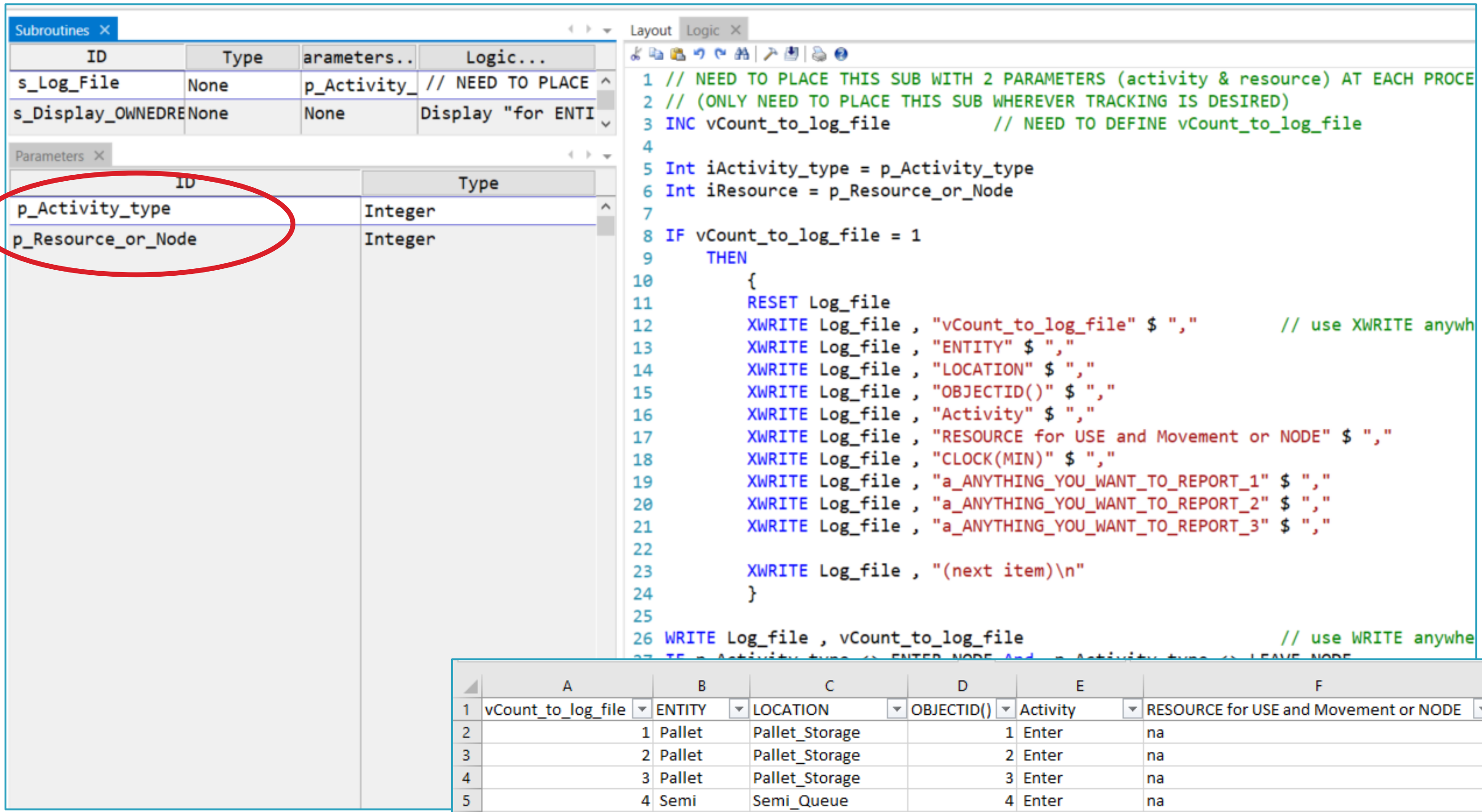
Entity...	Location...	Operation...
Semi	Semi_Queue	s_Log_File(ENTER,0)CT_Semi = Clock()Move F
Semi	End_Semi_Queue	s_Log_File(ENTER,0)TruckStatus = 1If Proce:
SemiDriver	End_Semi_Queue	s_Log_File(ENTER,0)

```
s_Log_File(ENTER,0)
CT_Semi = Clock()
Move For .1
```

ProModel®
Better Decisions—Faster

"Executed" Subroutine

- Example: Log File CSV creation



Subroutines

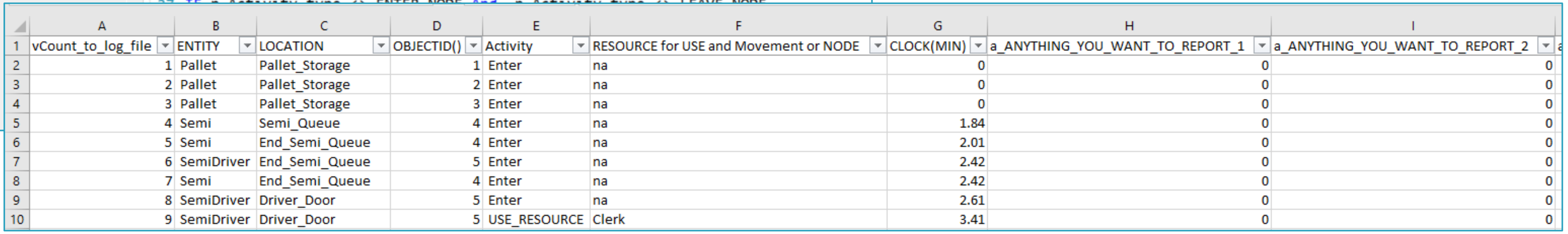
ID	Type	arameters..	Logic...
s_Log_File	None	p_Activity_	// NEED TO PLACE
s_Display_OWNEDRE	None	None	Display "for ENTI

Parameters

ID	Type
p_Activity_type	Integer
p_Resource_or_Node	Integer

Layout Logic

```
// NEED TO PLACE THIS SUB WITH 2 PARAMETERS (activity & resource) AT EACH PROCE
// (ONLY NEED TO PLACE THIS SUB WHEREVER TRACKING IS DESIRED)
INC vCount_to_log_file          // NEED TO DEFINE vCount_to_log_file

Int iActivity_type = p_Activity_type
Int iResource = p_Resource_or_Node

IF vCount_to_log_file = 1
    THEN
        {
        RESET Log_file
        XWRITE Log_file , "vCount_to_log_file" $ ","        // use XWRITE anywh
        XWRITE Log_file , "ENTITY" $ ","
        XWRITE Log_file , "LOCATION" $ ","
        XWRITE Log_file , "OBJECTID()" $ ","
        XWRITE Log_file , "Activity" $ ","
        XWRITE Log_file , "RESOURCE for USE and Movement or NODE" $ ","
        XWRITE Log_file , "CLOCK(MIN)" $ ","
        XWRITE Log_file , "a_ANYTHING_YOU_WANT_TO_REPORT_1" $ ","
        XWRITE Log_file , "a_ANYTHING_YOU_WANT_TO_REPORT_2" $ ","
        XWRITE Log_file , "a_ANYTHING_YOU_WANT_TO_REPORT_3" $ ","

        XWRITE Log_file , "(next item)\n"
        }

WRITE Log_file , vCount_to_log_file                         // use WRITE anywhe
```

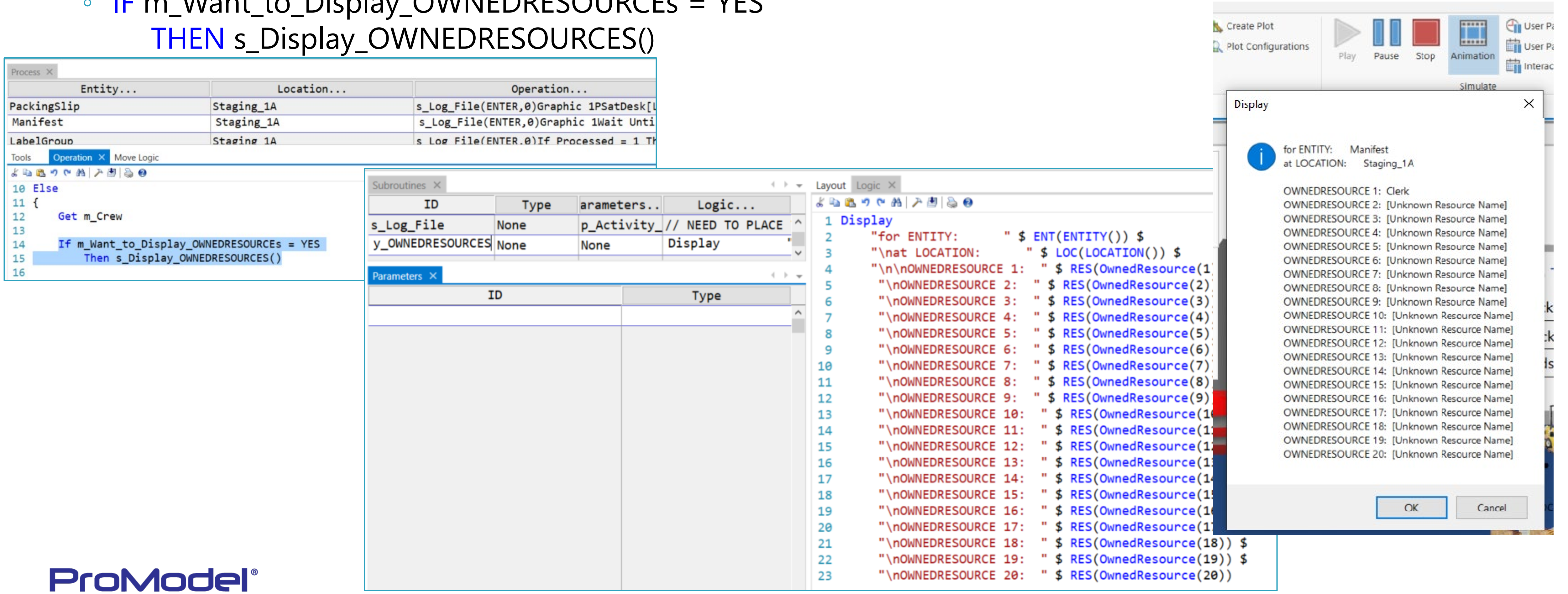


	A	B	C	D	E	F	G	H	I
1	vCount_to_log_file	ENTITY	LOCATION	OBJECTID()	Activity	RESOURCE for USE and Movement or NODE	CLOCK(MIN)	a_ANYTHING_YOU_WANT_TO_REPORT_1	a_ANYTHING_YOU_WANT_TO_REPORT_2
2	1	Pallet	Pallet_Storage	1	Enter	na	0	0	0
3	2	Pallet	Pallet_Storage	2	Enter	na	0	0	0
4	3	Pallet	Pallet_Storage	3	Enter	na	0	0	0
5	4	Semi	Semi_Queue	4	Enter	na	1.84	0	0
6	5	Semi	End_Semi_Queue	4	Enter	na	2.01	0	0
7	6	SemiDriver	End_Semi_Queue	5	Enter	na	2.42	0	0
8	7	Semi	End_Semi_Queue	4	Enter	na	2.42	0	0
9	8	SemiDriver	Driver_Door	5	Enter	na	2.61	0	0
10	9	SemiDriver	Driver_Door	5	USE_RESOURCE	Clerk	3.41	0	0

ProModel®
Better Decisions—Faster

“Executed” Subroutine

- Example: If want to display what resources are owned by an entity…
  - IF m_Want_to_Display_OWNEDRESOURCEs = YES
    THEN s_Display_OWNEDRESOURCES()



Entity...	Location...	Operation...
PackingSlip	Staging_1A	s_Log_File(ENTER,0)Graphic 1PSatDesk[
Manifest	Staging_1A	s_Log_File(ENTER,0)Graphic 1Wait Unti
LabelGroup	Staging_1A	s_Log_File(ENTER,0)If Processed = 1 Th

```
Else
{
     Get m_Crew

     If m_Want_to_Display_OWNEDRESOURCES = YES
        Then s_Display_OWNEDRESOURCES()

```

ID	Type	arameters...	Logic...
s_Log_File	None	p_Activity_	// NEED TO PLACE
y_OWNEDRESOURCES	None	None	Display

ID	Type

```
Display
    "for ENTITY:        " $ ENT(ENTITY()) $
    "\nat LOCATION:       " $ LOC(LOCATION()) $
    "\n\nOWNEDRESOURCE 1:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 2:  " $ RES(OwnedResource(2)
     "\nOWNEDRESOURCE 3:  " $ RES(OwnedResource(3)
     "\nOWNEDRESOURCE 4:  " $ RES(OwnedResource(4)
     "\nOWNEDRESOURCE 5:  " $ RES(OwnedResource(5)
     "\nOWNEDRESOURCE 6:  " $ RES(OwnedResource(6)
     "\nOWNEDRESOURCE 7:  " $ RES(OwnedResource(7)
     "\nOWNEDRESOURCE 8:  " $ RES(OwnedResource(8)
     "\nOWNEDRESOURCE 9:  " $ RES(OwnedResource(9)
     "\nOWNEDRESOURCE 10:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 11:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 12:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 13:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 14:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 15:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 16:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 17:  " $ RES(OwnedResource(1
     "\nOWNEDRESOURCE 18:  " $ RES(OwnedResource(18)) $
     "\nOWNEDRESOURCE 19:  " $ RES(OwnedResource(19)) $
     "\nOWNEDRESOURCE 20:  " $ RES(OwnedResource(20))
```

Display

```
for ENTITY:     Manifest
at LOCATION:    Staging_1A

OWNEDRESOURCE 1:  Clerk
OWNEDRESOURCE 2:  [Unknown Resource Name]
OWNEDRESOURCE 3:  [Unknown Resource Name]
OWNEDRESOURCE 4:  [Unknown Resource Name]
OWNEDRESOURCE 5:  [Unknown Resource Name]
OWNEDRESOURCE 6:  [Unknown Resource Name]
OWNEDRESOURCE 7:  [Unknown Resource Name]
OWNEDRESOURCE 8:  [Unknown Resource Name]
OWNEDRESOURCE 9:  [Unknown Resource Name]
OWNEDRESOURCE 10:  [Unknown Resource Name]
OWNEDRESOURCE 11:  [Unknown Resource Name]
OWNEDRESOURCE 12:  [Unknown Resource Name]
OWNEDRESOURCE 13:  [Unknown Resource Name]
OWNEDRESOURCE 14:  [Unknown Resource Name]
OWNEDRESOURCE 15:  [Unknown Resource Name]
OWNEDRESOURCE 16:  [Unknown Resource Name]
OWNEDRESOURCE 17:  [Unknown Resource Name]
OWNEDRESOURCE 18:  [Unknown Resource Name]
OWNEDRESOURCE 19:  [Unknown Resource Name]
OWNEDRESOURCE 20:  [Unknown Resource Name]
```

ProModel®
Better Decisions—Faster

Questions???

- Contact ProModel Technical Support
  - Support@ProModel.com
  - 888-PROMODEL
- Look at the Webinars in the ProModel Solutions Café