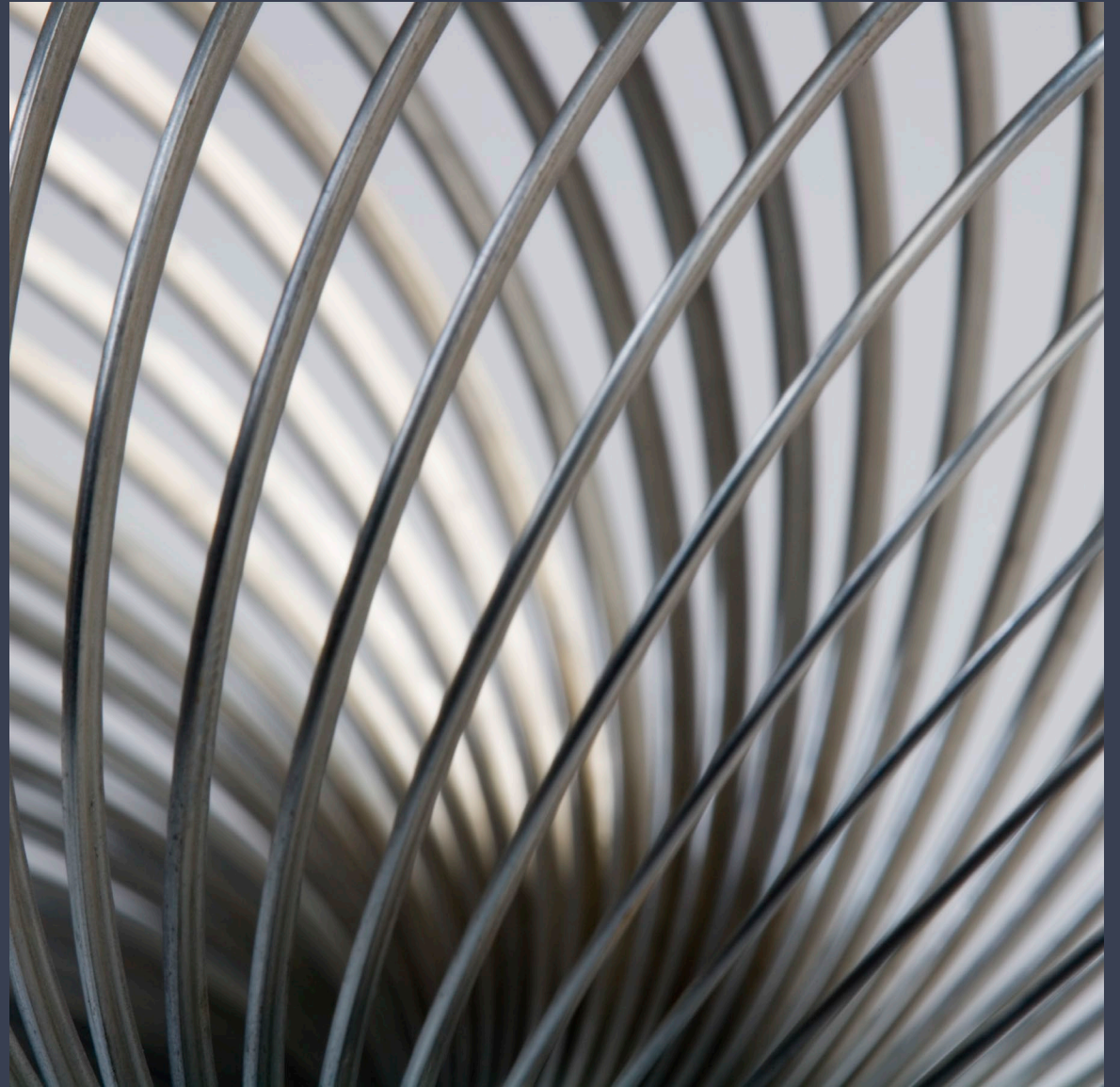


NUGM 2024

ECA BUILDING

Marc Harbeson, NovoROI Systems LLC



AGENDA

- Overview of the ECA.BUILD process screens
- Using HTML in the body of an ECA
- Using a subroutine to generate multiple detail for 1 triggered event (RQH and RQD in our example)
- Discuss different kinds of ECAs
- Discuss the Master Agent interface



ECA.BUILD SCREEN 1

ECA.BUILD - (S1) General Control Parameters				"ADD" Mode		
Code	RQN.SCAN.TEST			04 Active	Y	
02 Desc	RQN Scan Event Test ECA			05 Date In	05-09-24	
03 Comments	0 (for Documentation Purposes only)			06 Date Out		
.01)				07 Txn Hist	14	
.02)				08 Err Hist		
09 File	10 Filename	1	12 Functions	0	11 Flag	INCLUDE
.01)	RQH	RQH	.01)			
.02)			.02)			
13 Tol	14 Factor	0	15 CM	16 Classes	0	
.01)				.01)		
17 Options	0	18 Source User				
.01)						
.02)						
19 Scan	RQH	20 Schedule		21 Max		
22 With	1	23 24 Value	25 Ctv	26 27 Sorted by	0	
.01)	ID	EQ 1000		.01)		

- In this example, we are building a RQN notification ECA
 - Since this is a demo we will use the SCAN type ECA so we can trigger it manually.
- 02 – Describe your ECA
- 03 – Internal comments – Document your ECA
- 04 – Active or Not
- 05 / 06 – Effect Dates In / Out
- 07 – How long to retain ECAT records
 - ECA.AGENT will purge them
- 08 – How long to retain ECAT records in PURGE and ERROR



ECA.BUILD SCREEN 1

ECA.BUILD - (S1) General Control Parameters "ADD" Mode

Code	RQN.SCAN.TEST			04 Active	Y
02 Desc	RQN Scan Event Test ECA			05 Date In	05-09-24
03 Comments	0 (for Documentation Purposes only)			06 Date Out	
.01)				07 Txn Hist	14
.02)				08 Err Hist	
09 File	10 Filename	1	12 Functions	0	11 Flag
.01)	RQH	RQH	.01)		INCLUDE
.02)			.02)		
13 Tol	14 Factor	0	15 CM	16 Classes	0
.01)				.01)	
17 Options	0		18 Source User		
.01)					
.02)					
19 Scan	RQH	20 Schedule		21 Max	
22 With	1	23 24 Value	25 Ctv	26 27 Sorted by	0
.01)	ID	EQ	1000	.01)	

- 09 – Files that will be referenced in this ECA
 - Important that calling ECA.UPDATE also references these files in the event triggering ECA
- 10 – How you want to reference in ECA the file in 09
- 11/12 – You can include or exclude functions that are allowed to trigger this ECA
 - Blank is wildcard
- 13/14/15 – Set tolerance to enable this ECA
 - For example RQN changed by more than 10%
- 19/21/22/23/24/25/26/27 – Used with ECA Scan type to filter to only certain events
- 20 – How often will ECA.AGENT perform this scan
 - (blank = manual user scan)



ECA.BUILD SCREEN 1

ECA.BUILD - (S1) General Control Parameters *ADD* Mode

Code	RQN.SCAN.TEST	04 Active	Y
02 Desc	RQN Scan Event Test ECA	05 Date In	05-09-24
03 Comments	0 (for Documentation Purposes only)	06 Date Out	
.01)		07 Txn Hist	14
.02)		08 Err Hist	

09 File	10 Filename	1	12 Functions	0	11 Flag	INCLUDE
.01)	RQH	RQH	.01)			
.02)			.02)			

13 Tol	14 Factor	0	15 CM	16 Classes	0
.01)				.01)	

17 Options	0	18 Source User
.01)		
.02)		

19 Scan	RQH	20 Schedule		21 Max	
22 With	1	23 24 Value	25 Ctv	26 27 Sorted by	0
.01)	ID	EQ 1000		.01)	

- 17 – Enable different debug trigger types for interactive user debug of the ECA as triggered
 - Use this when you are not getting the ECA triggered and want to see what M2K is evaluating the conditions with
- 18 – Filters the event to specific users
 - Hint – you want your user ID here

Option and Description

- Audit.Read.Message.Notify Audit (READ mode) Message Notification
- Audit.Read.Screen.Notify Audit (READ mode) Screen Notification
- Audit.Update.Message.Notify Audit (UPDATE mode) Message Notification
- Audit.Update.Screen.Notify Audit (UPDATE mode) Screen Notification
- Disable.Scan.Loqqing Disable Scan Loqqing to ECA.AGENT.CTLS
- Prompt.Read.Preview Prompt (READ mode) for Preview (PC only)
- Prompt.Read.SendToList Prompt (READ mode) for Send To List (PC on
- Prompt.Read.TruthResults Prompt (READ mode) for Truth Results (PC o
- Prompt.Read.UserRemarks Prompt (UPDATE mode) for User Remarks (PC
- Prompt.Update.Preview Prompt (UPDATE mode) for Preview (PC only)
- Prompt.Update.SendToList Prompt (UPDATE mode) for Send To List (PC
- Prompt.Update.TruthResults Prompt (UPDATE mode) for Truth Results (PC
- Prompt.Update.UserRemarks Prompt (UPDATE mode) for User Remarks (PC



ECA.BUILD SCREEN 2

	03 Text	04	05	06	07	08	09	
	Keyword	3	File	Field/Value/Compute	Conv	Frmt	File	Field
.01)	RQNNBR		NEW/RQH	ID				
.02)	REQBY		NEW/RQH	Req_By				
.03)	REASON		NEW/RQH	Reason				
.04)								
.05)								

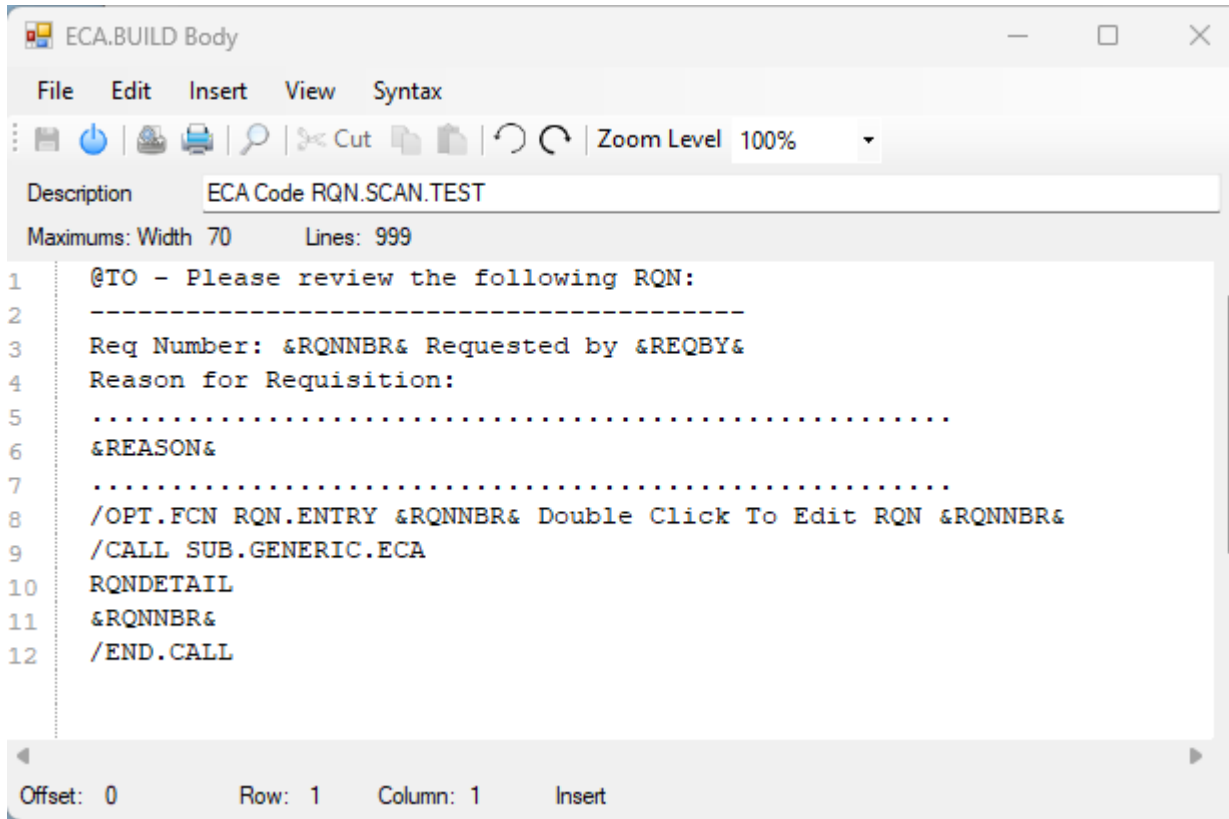
```
10 From @SYSUSER
11 Subj RQN Entered in system &RQNNBR& - please review
```

```
14 ECL 15 Uniquery ECL Command 0
.01) Change Which Field .....
```

- ECA Keywords are used in message construction, condition evaluation and action selection as well
 - Keywords are building blocks
- When sending messages, this determines the from and subject
 - Keywords and System Variables can be used here
 - For a full list of system variables – see the prompt help on prompt 12
- This can be used to put a LIST statement in the body, or used to generate Contact IDs for notification on next ACTION screen



ECA.BUILD SCREEN 2



The screenshot shows a window titled "ECA.BUILD Body" with a menu bar (File, Edit, Insert, View, Syntax) and a toolbar. The description field contains "ECA Code RQN.SCAN.TEST" and the maximums are set to Width 70 and Lines 999. The code editor contains the following text:

```
1 @TO - Please review the following RQN:
2 -----
3 Req Number: &RQNNBR& Requested by &REQBY&
4 Reason for Requisition:
5 .....
6 &REASON&
7 .....
8 /OPT.FCN RQN.ENTRY &RQNNBR& Double Click To Edit RQN &RQNNBR&
9 /CALL SUB.GENERIC.ECA
10 RQNDetail
11 &RQNNBR&
12 /END.CALL
```

At the bottom, the status bar shows "Offset: 0", "Row: 1", "Column: 1", and "Insert" mode.

- The body can use Keywords
- The body can also use System Variables (See Doc)
- You can also attach functions (See Doc for multiple options)
- You can CALL a subroutine to construct more body
 - In our example HTML formatted RQD lines
- CALL is also used to engage Master Agent (see prompt doc)

ECA SUBROUTINE EXAMPLE

```
SUBROUTINE SUB.GENERIC.ECA(PROPS,ECA.INFO,RESULTS)*
COMMAND = TRIM(PROPS<1>)
RESULTS = ''
BEGIN CASE
  CASE COMMAND = 'RQNDetail'
    RQH.KEY = TRIM(PROPS<2>)
    RQH.REC = XLATE("RQH",RQH.KEY,-1,"X")
    VEN.ID = RQH.REC<9>
    VEN.REC = XLATE("VEN",VEN.ID,-1,"X")
    VEN.NAME = VEN.ID : ' ' : VEN.REC<2,1>
    RESULTS<1,-1> = '<STYLE TYPE="text/css">'
    RESULTS<1,-1> = '{font-family: Arial; font-size: 10pt;}'
    RESULTS<1,-1> = '</STYLE>'
    RESULTS<1,-1> = '<TABLE BORDER="1" font size=1>'
    RESULTS<1,-1> = '<CAPTION>RQN Header Info</CAPTION>'
    RESULTS<1,-1> = '<TR><TD><B>Vendor Name</B></TD><TD>':VEN.NAME:'</TD></TR>'
    RESULTS<1,-1> = '<TR><TD><B>Rqn Total</B></TD><TD>':OCONV(RQH.REC<29>,"MD2$,"'):'</TD></TR>'
    RESULTS<1,-1> = '<TR><TD><B>Special Instructions</B></TD><TD>': RQH.REC<22> : '</TD></TR>'
    RESULTS<1,-1> = '</TABLE>'
    *
    RESULTS<1,-1> = '<TABLE BORDER="1" font size=1>'
    RESULTS<1,-1> = '<CAPTION>RQN Details</CAPTION>'
    RESULTS<1,-1> = '<TR><TD><B>Line</B></TD>'
    RESULTS<1,-1> = '<TD><B>Item</B></TD>'
    RESULTS<1,-1> = '<TD><B>Desc</B></TD>'
    RESULTS<1,-1> = '<TD><B>Qty</B></TD>'
    RESULTS<1,-1> = '<TD><B>UM</B></TD>'
    RESULTS<1,-1> = '<TD><B>Unit Price</B></TD>'
    RESULTS<1,-1> = '<TD><B>Extd Price</B></TD>'
    RESULTS<1,-1> = '<TD><B>Disp Dest</B></TD></TR>'
    *
    RQD.LINES = RQH.REC<27>
    FOR L = 1 TO DCOUNT(RQD.LINES,@VM)
      RQD.KEY = RQH.KEY : "*" : RQD.LINES<1,L>
      RQD.REC = XLATE("RQD",RQD.KEY,-1,"X")
      QTY = RQD.REC<6>
      PRICE = RQD.REC<7>
      *** CALC EXT.PRICE = QTY * PRICE :QTY=0,PRICE=4,EXT.PRICE=4<4> ;*** Calc Source Follows (03-22-15)
      EXT.PRICE=(INT(((QTY/10)*PRICE)*10))
      ***
      IF RQD.REC = '' THEN CONTINUE
      RESULTS<1,-1> = '<TR><TD>' : RQD.LINES<1,L> : '</TD>'
      RESULTS<1,-1> = '<TD>' : RQD.REC<3> : '</TD>'
      RESULTS<1,-1> = '<TD>' : RQD.REC<4> : '</TD>'
      RESULTS<1,-1> = '<TD>' : QTY : '</TD>'
      RESULTS<1,-1> = '<TD>' : RQD.REC<9> : '</TD>'
      RESULTS<1,-1> = '<TD>' : OCONV(PRICE,"MD4,$") : '</TD>'
      RESULTS<1,-1> = '<TD>' : OCONV(EXT.PRICE,"MD4,$") : '</TD>'
      RESULTS<1,-1> = '<TD>' : RQD.REC<54> : ' ' : RQD.REC<55> : '</TD></TR>'
    NEXT L
    RESULTS<1,-1> = '</TABLE>'
  END CASE
RETURN
```



ECA.BUILD SCREEN 3

	03 EVENTS	04 Type	05 File	06 Field	07 Tol	08 Conditions	09 Actions	10 SU	11 more=>
.01)	MyScan	SCAN	RQH			IsMyRQN	TellMe	N	Y
.02)									
.03)									

	12 CONDITIONS	13 IF	14 File	15 Field/Value	16 Op	17 File	18 Field/Value
.01)	IsMyRQN	IF	NEW/RQH	ID	EQ	VALUE	1000
.02)							
.03)							
.04)							
.05)							

	19 ACTIONS	20 Type	21 Select	22 Parameter	23 File	24 Field/Key/Value	25 Conditions
.01)	TellMe	Notify	EcaKwd	RQNKWD			IsMyRQN
.02)							
.03)							

Change Which Field

- Top section defines the event which will trigger the ECA (The E)
- The middle section is where you define the condition (The C)
 - Conditions can be multiple to multiple in both Events and Actions
 - And / Or logic is also available
- The bottom section defines the Action (The A)
 - There are multiple ways to do actions
 - Actions can be multiple to multiple as well
- To make debugging easier, remember the KISS method: (Keep It Simple Stupid)
 - Don't do multiple things in one ECA



ECA.BUILD SCREEN 3

	03 EVENTS	04 Type	05 File	06 Field	07 Tol	08 Conditions	09 Actions	10 SU	11 more=>
.01)	MyScan	SCAN	RQH			IsMyRQN	TellMe	N	Y
.02)									
.03)									

Option and Description

CREATED Record CREATED
 DELETED Record DELETED
 READ Record READ
 ROUTING Routing
 SCAN File SCAN
 VAL.ADD Record CHANGED - Field Value ADDED
 VAL.ANY Record CHANGED - Field Value ANYTHING
 VAL.CHG Record CHANGED - Field Value CHANGED
 VAL.DEC Record CHANGED - Field Value DECREASED
 VAL.DEL Record CHANGED - Field Value DELETED
 VAL.INC Record CHANGED - Field Value INCREASED
 VAL.NUL Record CHANGED - Field Value NULL

03	Event	MyScan
04	Type	SCAN
05	File	RQH
06	Field	
07	Description	RQN Scanned
08	Tolerance	
09	Conditions	IsMyRQN
10	Actions	TellMe
11	Signup	N
12	Notify Text	
13	List Name	
14	Nag Buffer	0
15	Active	Y
16	Date In	05-09-24
17	Date Out	
18	Return->	Y

- Event Types – These are the kinds of Events you can capture
 - Depending on how this is answered depends on how prompts 5-6 are answered
 - Prompt 7 references the tolerance defined on screen 1
 - Prompt 8 / 9 are a bit of chicken and egg
 - First time through leave blank
- A note about the Nag Buffer
 - Use this when your transaction may get called multiple times within several seconds
 - For example, SOH gets written multiple times when saving a Sales Order
 - Think – once per SOD, and other calls as well



ECA.BUILD SCREEN 3

	12	13	14	15	16	17	18
	CONDITIONS	IF	File	Field/Value	Op	File	Field/Value
.01)	IsMyRQN	IF	NEW/RQH	ID	EQ	VALUE	1000
.02)							
.03)							
.04)							
.05)							

- A single condition can be on multiple lines
 - AND / OR logic applies using Prompt 13 on additional lines
- Multiple conditions (with different IDs in Prompt 12) can also be defined
- Old / New values can be compared to VALUE (Static coded value), File References (Old or New values), SYS.CTL values, etc.
- See Prompt Doc – again ECA prompt documentation is really good



ECA.BUILD SCREEN 3

	19 ACTIONS	20 Type	21 Select	22 Parameter	23 File	24 Field/Key/Value	25 Conditions
01)	TellMe	Notify	EcaKwd	RQNKWD			IsMyRQN
02)							
03)							

Change Which Field

Option and Description

Action1 via Action1 (System Users only)
 Alarm via Alarm (System Users only)
 Message via Message Box (Source System Users only)
 Notify via Contact Methods (Email/Fax/Printer/EcaTxn)
 Publish via newsfeed publishing
 Routing via Contact Method (EcaTxn only)
 Screen via Screen (Source System Users only)

Option and Description

CntKwd Select Contact(s) for Contact Keyword
 CntOrig Select Contact(s) originating transaction/routing
 Contact Select Contact for specific ID
 EcaKwd Select Contact(s) for Contact ECA Keyword
 EclRef Select Contact(s) via ECL Reference Command
 SrcUser Select Contact(s) for Source User
 SysRef Select Contact(s) for Contact ECA System Ref

- Same here – Multiple actions can be defined
- Multiple outputs from same action are also allowed by using the same Action ID
- Types determine where the action goes
 - Screen Msg, Notification, News Feed Routing, etc.
- Selection defines how a destination is selected (User / Contact) via direct reference (Contact ID's or User ID's) or indirect reference such as Keywords that select multiple Contacts



CONTACTS ECA KEYWORD EXAMPLE

Contact	First Name	MI	Last Name	Nickname
<input type="text" value="MYSELF"/>	<input type="text" value="Marc"/>	<input type="text"/>	<input type="text" value="Harbeson"/>	<input type="text"/>

06 Eca Default Method		07 Eca Keywords		08 ECA Print Dest	
.01)	<input type="text" value="Normal"/>	.01)	<input type="text" value="RQNKWD RQN Test"/>	.01)	<input type="text"/>
.02)	<input type="text"/>	.02)	<input type="text"/>	.02)	<input type="text"/>
.03)	<input type="text"/>	.03)	<input type="text"/>	.03)	<input type="text"/>
.04)	<input type="text"/>	.04)	<input type="text"/>	.04)	<input type="text"/>
Entries	1	Entries	1	Entries	0

- By using Keywords instead of Contact IDs, as users are added and removed from the system, the ECA requires no modifications to keep up with the ever changing list of contacts in the system



ECA OUTPUT FROM SAMPLE

ECA.TXNS

File Edit View Help

Back Forward Stop Refresh Home Short Cuts Print

Selection Go

MANAGE 2000[®] **ECA Transaction Maintenance**

NEW Audit Record 1 of 1 5 Default

Menu Change/Forward Inbox Save Purge Fri 05-24-24

From: ROI Sent: Thu 05-09 at 01:42PM
 To: Harbeson, Marc Packet: n/a
 Subject: RQN Entered in system 1000 - please review

Harbeson, Marc - Please review the following RQN:

 Req Number: 1000 Requested by MAH
 Reason for Requisition:

[Double Click To Edit RQN 1000](#)

RQN Header Info

Vendor Name	1010 US Wire and Steel
Rqn Total	\$26,675.00
Special Instructions	

RQN Details

Line	Item	Desc	Qty	UM	Unit Price	Extd Price	Disp Dest
1	200 01	Processor Pentium 400MHz	100	EA	\$26.5500	\$2,655.0000	S 200 01
2	301 01	Chassis, Full	50	EA	\$0.2500	\$12.5000	S 301 01

Menu Change/Forward Inbox Save Purge Fri 05-24-24

Ref: #100164 RQN.SCAN.TEST/MyScan [C:\ROMAH.MAIN] Sent: Thu 05-09-24 at 01:42PM

[top] [subject] [close window]

Complete C:\Users\mah\AppData\Local\Temp\Eca49355-1001.HTML



ECA OUTPUT AS MASTER AGENT INPUT

ECA Subroutine: WRITE.ECA.MA.REQUEST(Properties, EcaInfoResults, Results)

This subroutine was specifically designed to be used with the /CALL command. It will generate a record for the MA_JOB_CTL file so ECA triggered information can be exported to the Master Agent subsystem to load a queue of choice including swap variables. Optionally, selected label(s) within a queue can be loaded with the default being all labels.



All properties must be carefully entered in ECA.BUILD since no validations are done when ending out. All validations are done in Master Agent at runtime and any errors are reported via email to Master Agent Error Administrators (configured in MA.CONSTANTS). You should set yourself up accordingly.

```
/CALL WRITE.ECA.MA.REQUEST
Queue=queueid
Label=labelid
{taskuser}=@SYSUSER
{custnbr}=&custnbr&
{sortopt}=B
/END.CALL
```

The property 'Queue=queueid' is required and must occur as the first property. Only one queue can be declared for each call, but many individual calls can be scripted. The reference 'queueid' must be a valid Master Agent Queue (MA.QUEUES) with a Queue Type of 'E' (ECA). This queue is a template and all outstanding job requests pointing at it will be honored.

The property 'Label=labelid' is optional and can be located anywhere after the first property. The reference 'labelid' must be a valid Master Agent Queue Label (MA.QUEUES). This property can be listed multiple times declaring many labels, but remember, once you start declaring labels, you must include all the ones desired. If no labels are declared, then all labels are selected.

Master Agent Swap Variable(s), not to be confused with ECA Variables, are optional and can be located anywhere after the first property. This is a very powerful concept to load Master Agent queues tailored to the specifics of an ECA trigger. The Master Agent Swap Variable is on the left side (=) of the property. The right side of the property is the actual value that will be swapped (e.g. literal, ECA Text Keyword, ECA Variable, etc). The reference '{taskuser}' above is an example of Master Agent Swap Variable using the ECA Variable '@SYSUSER' as the actual value that will be loaded into the Master Agent Queue at runtime (e.g. queue script line Task CUST.LIST {taskuser}').



NUGM 2024

THANK YOU

Marc Harbeson

NovoROI Systems LLC

