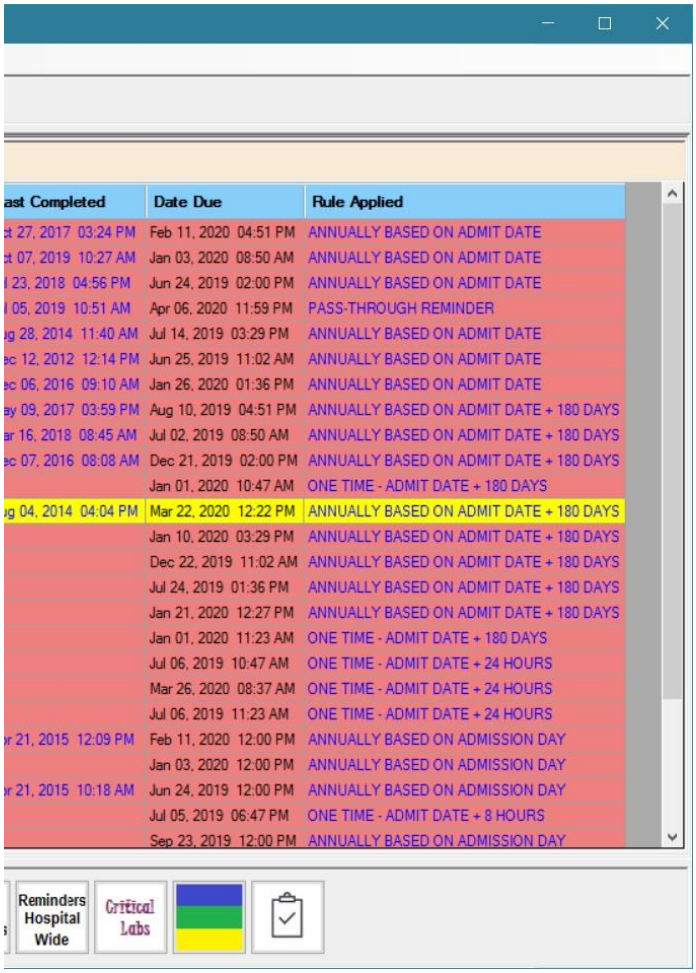
Dashboard Technical Manual



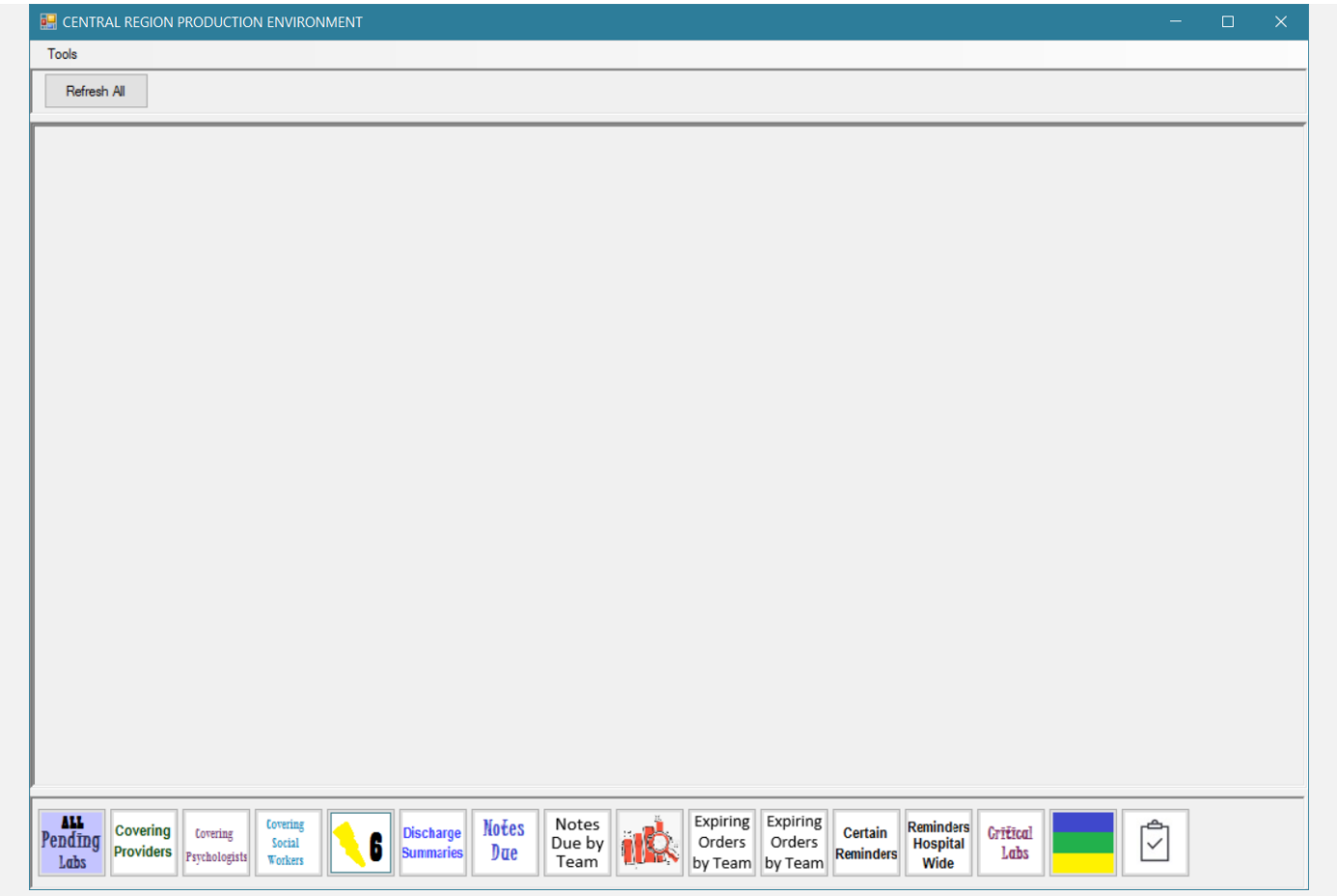
Central Regional Hospital
Authored by: Joe Thurber

Dashboard

What Is It?

Simply stated, the Dashboard is a container used to display data from a mumps database. It is written in C# to run in Microsoft Windows. The data is displayed by object-oriented "modules" meaning that each module is responsible for its own behavior. Modules are defined solely in the mumps database. Up to four modules can be displayed on the Dashboard at any one time.

This is a picture of the Dashboard with no modules displayed:



At the top, you see “Central Region Production Environment”. This text is returned by a remote procedure call (RPC) that returns the first line of text in the INTRO MESSAGE field in the KERNEL SYSTEM PARAMETERS file. We use this to let the user know which of our many VistA environments they are currently accessing.

The “Tools” menu currently has only one option which will allow you to enlarge the display font.

The “Refresh All” button will force a refresh of the data in all modules currently displayed. In this case there are no modules currently displayed.

The large blank section in the middle is the real estate used to display up to four modules at a time. Currently no modules are displayed.

At the bottom are icons representing each module currently defined in VistA. These can be restricted by security key as will be explained later.

There are two possible types of modules:

1) Datagrid – displays data in a grid of rows and columns
2) RichText – displays data in blocks of text

To display a module, double click on or drag the icon into the display area. Up to four modules can be displayed at one time.

Dashboard

Disclaimers (and Diatribe)

License:

This applies to all source code in this release.

Additionally, I am aware that the mumps routines do not fully comply with VA standards. We are not the VA and did not have the time or manpower for this extra requirement. If you want them to comply, please make the changes you need for your site. You will also notice, I'm sure peculiarities in my programming style, both in mumps and C#. For example, I usually use Fileman to write data while using direct reads. I like and respect George Timson as much as anyone; that's just how I choose to code. I make extensive and perhaps unnecessary use of $GET. I use complete $PIECE commands even if I want only the first default piece. Again, this all works at our site and has for many years. If you want to modify the style, that is your prerogative. Some of the routines distributed are very complex and may have overlap where code could have been more streamlined and reusable. This is the direct result of ever-changing specifications and requirements in our dynamic live environment. I'm sure Hardhats reading this will understand. If it pleases you to simplify, go right ahead! In many routines, my dot levels are insane. I know that. Just do your thing for your site! Some of the modules are dependent on proprietary files and code that is not being released at this time, but some supporting RPC's and routines are provided as examples of how this tool may be used. This includes at least the Sticky Notes, Covering Providers, Covering Social Workers, and Covering Psychologists modules.

Dashboard

Supporting Files

There are two main mumps database files that support the dashboard in general. File number 300894, C9C DASHBOARD MODULES and file number 300896, C9C DASHBOARD USER SETTINGS.

C9C DASHBOARD MODULES

^C9C(300894,D0,0)= (#.01) NAME [1F] ^ (#.05) ACTIVE [2S] ^ (#.5) REQUIRED KEY [3P:19.1] ^
^C9C(300894,D0,1)= (#1) BUTTON TEXT [1F] ^
^C9C(300894,D0,2,0)=^300894.02^^ (#2) BUTTON IMAGE
^C9C(300894,D0,2,D1,0)= (#.01) BUTTON IMAGE [1W] ^
^C9C(300894,D0,3)= (#3) MODULE TYPE [1S] ^
^C9C(300894,D0,4,0)=^300894.04^^ (#4) DATAGRID COLUMN HEADERS
^C9C(300894,D0,4,D1,0)= (#.01) DATAGRID COLUMN HEADER [1F] ^ (#.5) COLUMN DATA TYPE [2S] ^
^C9C(300894,D0,4,D1,1)= (#1) ORDER [1N] ^
^C9C(300894,D0,4,D1,2)= (#2) DISPLAY THIS COLUMN [1S] ^
^C9C(300894,D0,4,D1,3)= (#3) CLICK RPC [1P:8994] ^ (#3.2) CLICK RPC PARAMETER COLUMN IENS [2F] ^
^C9C(300894,D0,4,D1,4)= (#4) HINT [1F] ^
^C9C(300894,D0,4,D1,5)= (#2.5) DISPLAY HIDDEN COLUMN LOGIC [E1,245K] ^
^C9C(300894,D0,6)= (#6) DATA POPULATION RPC [1P:8994] ^
^C9C(300894,D0,7)= (#7) POSSIBLE LOCATIONS RPC [1P:8994] ^
^C9C(300894,D0,8)= (#8) LOCATIONS LABEL OVERRIDE TEXT [1F] ^
^C9C(300894,D0,10)= (#10) RPC GET POSSIBLE FILTER VALUES [1P:8994] ^
^C9C(300894,D0,11,0)=^300894.01P^^ (#.06) ACTIVE DIVISIONS
^C9C(300894,D0,11,D1,0)= (#.01) ACTIVE DIVISIONS [1P:4] ^
^C9C(300894,D0,12)= (#12) ACTION BUTTON TEXT [1F] ^
^C9C(300894,D0,13)= (#13) ACTION BUTTON RPC [1P:8994] ^

NAME: Name of the module. Shows as hint when hovering over the icon and shows up in header of the module when displayed.

ACTIVE: Allows you to deactivate modules. If you hold the key called INACTIVE DASHBOARD MODULES, you will see deactivated modules when opening the dashboard. This allows developer access when coding or repairing modules.

REQUIRED KEY: If populated, only users with the designated key will see this module.

BUTTON TEXT: If no image is defined, this text will show up on the icon.

BUTTON IMAGE: Base64 representation of a 60x60 pixels image that will become the icon for the module. Another C# program I wrote, PngToBase64, will be provided with this release to help with this coding.

MODULE TYPE: Datagrid or RichText

DATAGRID COLUMN HEADERS: Multiple to hold datagrid column definitions.

DATAGRID COLUMN HEADER: Text to be used in the datagrid header for this column.

COLUMN DATA TYPE: TEXT, DATE, NUMERIC, ROW COLOR, WEBSITE. The first three are obvious. ROW COLOR is usually set as a column that is not displayed, but will be used in any case to determine what color to use as the background for each row. The mumps return data should return the color for each row as part of it's data. WEBSITE is a url that can be clicked on to get more data about the results as a whole or a specific row. It will show up in blue text similar to a normal hyperlink.

ORDER: Order in which to display this column.

DISPLAY THIS COLUMN: Sounds obvious, but there are valuable uses for hidden columns. One is described above in the row color description. Another is described below in the CLICK RPC PARAMETER COLUMN IENS description.

CLICK RPC: Remote procedure to run when you click on a data cell in this column. Should return a textual value to display back to the user.

CLICK RPC PARAMETER COLUMN IENS: A semicolon separated list of the iens for the column(s) data to send as parameters for the CLICK RPC. Often hidden columns are used for this. For example, the end user is not interested in the patient DFN, but your RPC may need it

to return the correct data. It is key that you use the IEN for the column in the subfile, NOT the order of display for the column(s).

HINT: Text to display when hovering over a cell in this column.

DISPLAY HIDDEN COLUMN LOGIC: Great for developers. You can have your RPC return any data you need to troubleshoot and add mumps logic here that will display it based on having a security key or other criteria. We use a parameter called "VIEW NOTES DUE RULE APPLIED". It can also be used to display highly sensitive data to a subset of users.

DATA POPULATION RPC: This is the main workhorse for returning the data for this module. If the module type is DataGrid, it must return an array of rows each containing the expected number of column values delimited by carets. Data types must also be honored. Empty values are ok, but be careful when using them for parameters in RPC calls. Note: Remember to add these RPC's as well as all click RPC's, get possible filter values, action button, and possible locations RPC's to the C9C DASHBOARD option RPC multiple in file 19.

POSSIBLE LOCATIONS RPC: If populated, this presents a combobox to the user so that a particular location can be chosen, often a ward. The chosen location will be sent to the data population RPC so that results will be specific to that location. This can also be repurposed to allow a user to choose from other types of values, for example a team.

LOCATIONS LABEL OVERRIDE TEXT: If you want the combobox label to say something other than Select Location, enter it here. For example, it may say Select Team.

RPC GET POSSIBLE FILTER VALUES: RPC you write to return a list that can be used to further filter results when retrieving data for this module. A user will be able to choose one or more values from this list to include in the results. I'll describe this more fully later in this document.

ACTIVE DIVISIONS: We are serious about divisionalization at Central Regional and the Dashboard is no exception. This multiple allows us to specify which divisions can see this module.

ACTION BUTTON TEXT: If populated this will cause a button to be displayed that will act on the selected richtext block. Note that this feature is currently available only for richtext modules and needs further development for datagrid use.

ACTION BUTTON RPC: RPC to run when the action button text is clicked. Note that this feature is currently available only for richtext modules and needs further development for datagrid use.

C9C DASHBOARD USER SETTINGS

^C9C(300896,D0,0)= (#.01) USER [1P:200] ^
^C9C(300896,D0,10,0)=^300896.02P^^ (#.5) DIVISION
^C9C(300896,D0,10,D1,0)= (#.01) DIVISION [1P:4] ^
^C9C(300896,D0,10,D1,1,0)=^300896.21P^^ (#1) MODULE
^C9C(300896,D0,10,D1,1,D2,0)= (#.01) MODULE [1P:300894] ^
^C9C(300896,D0,10,D1,1,D2,1,0)=^300896.212^^ (#2) MODULE INSTANCE
^C9C(300896,D0,10,D1,1,D2,1,D3,0)= (#.01) MODULE INSTANCE [1N] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,1,0)=^300896.2121^^ (#1) FILTER ITEMS SHOWN
^C9C(300896,D0,10,D1,1,D2,1,D3,1,D4,0)= (#.01) FILTER ITEM SHOWN IEN [1N] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,1,D4,1)= (#1) FILTER ITEM SHOWN NAME [1F] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,2)= (#2) NOW SHOWING [1S] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,3)= (#3) ORDER SHOWN [1S] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,4)= (#4) LOCATION SELECTION IEN [1N] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,5)= (#5) SORT COLUMN [1N] ^
^C9C(300896,D0,10,D1,1,D2,1,D3,6)= (#6) SORT DIRECTION [1S] ^
^C9C(300896,D0,10,D1,2)= (#2) OVERALL WINDOW WIDTH [1N] ^
^C9C(300896,D0,10,D1,3)= (#3) OVERALL WINDOW HEIGHT [1N] ^
^C9C(300896,D0,10,D1,4)= (#4) FONT SIZE [1S] ^
^C9C(300896,D0,10,D1,5)= (#5) SPLITTER LEFT HORIZONTAL POSITION [1N] ^
^C9C(300896,D0,10,D1,6)= (#6) SPLITTER RIGHT HORIZONTAL POSITION [1N] ^
^C9C(300896,D0,10,D1,7)= (#7) SPLITTER VERTICAL POSITION [1N] ^
^C9C(300896,D0,10,D1,8)= (#8) WINDOW LEFT [1N] ^
^C9C(300896,D0,10,D1,9)= (#9) WINDOW TOP [1N] ^

I'm not going to describe each field in the user settings file. Basically, it saves and returns division specific settings for each user. It keeps up with modules that are open, filters in place, and visual settings. These are saved each time a user closes the application and retrieved when opened.

Dashboard

Getting Started

I have provided KIDS builds for the two files above and include the module definitions we have developed at Central Regional along with the remote procedures for each one. Our VistA version is highly customized and I feel certain that these modules will not work for you without some changes. One certain example is that the pointers in the C9C DASHBOARD MODULES file will likely all have to be updated to point to the correct remote procedures. Ah well, if I only knew KIDS better, perhaps I could save you this step. I will provide results of a Fileman print of the RPC's for each module; that should be of help. We also have more than a few custom fields in standard VA files, some of which may affect these modules. They are not provided at this time.

One of our more complex module groups is the Notes Due, Notes Due By Team, and Notes Due Forecast set of modules. I say set because they are very similar to each other and all make use of the same basic logic. They also take advantage of another GUI that we are releasing here called the Rules Engine. There are other documents describing the Rules Engine. However, all the code and data that we use is provided. The note titles we use are not provided. Keep in mind that this was evolutionary and that is reflected in the rather complex code.

The GUI applications can reside on a network. Just make sure that you include in the directory Medsphere's two dll files: Medsphere.OpenVista.Remoting.dll and Medsphere.OpenVista.Shared.dll. Thank you Medsphere! You'll also need a shortcut set up like you would a shortcut to CPRS. It will look something like this:
N:\WorldVistA\Dashboard\Dashboard.exe S=192.168.33.39 P=9004
These are fictitious ip address and port. Use the same ones you'd use for CPRS.

You'll have to create in file 19 the C9C DASHBOARD option and register all the RPC's to it. You will have to add this option as a secondary menu to any user who requires access. Our current listing looks like this (I have removed those RPC's that are related to our proprietary login code which has been removed from this release):

NUMBER: 11468 NAME: C9C DASHBOARD
MENU TEXT: C9C DASHBOARD TYPE: Broker (Client/Server)
CREATOR: THURBER,JOSEPH H
DESCRIPTION: Dashboard to display custom modules
TIMESTAMP OF PRIMARY MENU: 65462,38399
RPC: C9C MODULE GET NOTES DUE
RPC: XUS INTRO MSG
RPC: XUS GET USER INFO
RPC: XUS DIVISION GET
RPC: XUS DIVISION SET
RPC: ORWU HASKEY
RPC: XOBV TEST PING
RPC: C9C MODULE DEFINITIONS
RPC: C9C DATAGRID COLUMN HEADERS
RPC: C9C BUTTON IMAGE BYTE STRING
RPC: C9C NOTES DUE ALL FILTERS
RPC: C9C DASHBOARD GET USER FILTERS
RPC: C9C DASHBOARD PUT USER FILTERS
RPC: C9C DB MODULE NOTE TEXT
RPC: C9C SAVE DASHBOARD PREFERENCES
RPC: C9C GET DASHBOARD PREFERENCES
RPC: C9C GET DIVISIONS
RPC: C9C WARDS AND UNITS
RPC: C9C GET CRITICAL LAB RESULTS
RPC: C9C DASHBOARD STICKY NOTES
RPC: C9C GET PENDING LABS
RPC: C9C GET WARDNAMES
RPC: C9C FILTER STICKY NOTES
RPC: C9C NOTES DUE BY TEAM
RPC: C9C ND GET TEAMS
RPC: ORQOR DETAIL
RPC: C9C LAB DRAWS OTHER THAN SIX
RPC: C9C UNVERIFIED FOR DASHBOARD
RPC: C9C EXPIRING FOR DASHBOARD
RPC: C9C FORECAST NOTES DUE
RPC: C9C FORCE STICKY NOTE EXP
RPC: C9C MISSING DSCG SUMMARIES
RPC: C9C DASHBOARD REMINDER MODULE
RPC: C9C C AND R FILTERS
RPC: C9C COVERING PROVIDERS MODULE
RPC: C9C GET ALL APPROPRIATE TEAMS
RPC: C9C ALL PATIENTS REMINDERS
RPC: C9C COVERING PSYCHOL MODULE
RPC: C9C COVERING SW MODULE
RPC: C9C SHOW OVERRIDES

RPC: C9C GERD DRUG FILTER
RPC: C9C ON GERD DRUGS

And if you use the Rules Engine, you will need to create an option for it in file 19 and register these RPC's to it:

NUMBER: 11467 NAME: C9C RULES ENGINE
MENU TEXT: C9C RULES ENGINE TYPE: protocol
CREATOR: THURBER,JOSEPH H TIMESTAMP OF PRIMARY MENU: 65289,41013
RPC: C9C GET DIVISIONS
RPC: XUS DIVISION SET
RPC: XUS GET USER INFO
RPC: XUS DIVISION GET
RPC: C9C GET TITLES
RPC: XOBV PING
RPC: ORQPT WARDS
RPC: C9C WARDS AND UNITS
RPC: C9C GET FORMULAS
RPC: C9C SAVE RULES ENGINE WARDS
RPC: C9C SAVE RULES ENGINE UNITS
RPC: C9C GET RULE NAMES
RPC: C9C ADD RULE TO FORMULA
RPC: C9C RULES ENGINE ORD ITEMS
RPC: C9C DELETE SINGLE RULE
RPC: C9C SAVE NEW FORMULA
RPC: C9C DELETE FORMULA
RPC: XUS INTRO MSG
RPC: C9C GET DASHBOARD REMINDERS
RPC: C9C GET COHORT FOR TITLE
RPC: C9C SET COHORT FOR TITLE
RPC: C9C GET PASS-THROUGH REMINDERS
RPC: XOBV TEST PING

Each module that you want to use will have to be edited to be sure that the RPC pointers are correct and that your Division is in the multiple.

In this release, simple access code/verify code are used for authentication. They are hashed using the standard released algorithm.

If you have never logged into this application before, and you made all the required changes to get this to run in your environment, the application will open with a default module and settings. Simply drag modules into the center window or double-click on

them. There are splitters and scrollbars that you can use to change individual module window sizes.

Dashboard

Step By Step – Build A Module

Let's create a very simple example module from start to finish. Suppose we want to see on the dashboard every patient that has an active order for drugs used to treat GERD. Further, we want to be able to choose the ward to see and to have different row colors for males and females. Here is how I would approach it:

First, I would write a routine to retrieve the desired data. In looking at the orderable items file, there is an AOI index that allows you to see which patients have an active order for specified orderable items. We'll use this.

This is the relatively untested routine I put together quickly for this:

```
C9CMODCD ;CRH JHT 2020 - show patients with orders for particular drugs
  Q ;no entry from top
RPCFILT(C9CRET) ;RPC: C9C GERD DRUG FILTER
 ;C9CRET - return array
 ;find orderable item ien's for drugs we're interested in
 ;unfortunately in our environment, lot's of drug names have a trailing space
 ;so we need to check for that
 N OIIEN,CNT
 S CNT=0
 N DN
 F DN="FAMOTIDINE TAB","FAMOTIDINE TAB ","OMEPRAZOLE CAP,EC","OMEPRAZOLE CAP,EC ",
"OMEPRAZOLE/SODIUM BICARBONATE ","PANTOPRAZOLE TAB,EC","PANTOPRAZOLE TAB,EC ",
"PANTOPRAZOLE INJ,PWDR","PANTOPRAZOLE INJ,PWDR ","RANITIDINE TAB","RANITIDINE TAB ",
"RANITIDINE INJ","RANITIDINE INJ ","ESOMEPRAZOLE CAP,EC","ESOMEPRAZOLE CAP,EC ",
"ESOMEPRAZOLE CAP,SA","ESOMEPRAZOLE CAP,SA ","LANSOPRAZOLE CAP,EC",
"LANSOPRAZOLE CAP,EC ","LANSOPRAZOLE TAB,ORAL DISINTEG","RABEPRAZOLE TAB,EC",
"RABEPRAZOLE TAB,EC ","DEXLANSOPRAZOLE CAP,EC","DEXLANSOPRAZOLE CAP,EC " D
 .S OIIEN=$O(^ORD(101.43,"B",DN,0))
 .I OIIEN>0 S C9CRET($I(CNT))=OIIEN_"^"_$$TRIM^XLFSTR(DN)
 Q
RPC(C9CRET,WARD,FILTER) ;RPC: C9C ON GERD DRUGS
 ;C9CRET - return array
 ;WARD - ward name from ward/location file
 ;FILTER - optional Filter Values As Set By User
 Q:$G(WARD)=""
 I '$D(FILTER) D ;if no filter, populate with all possible values
 .D RPCFILT(.FILTER)
```

```
Q:'$D(FILTER)
;order status we are interested in - from status file 100.01
N AX,STATIEN,STAT
F AX="ACTIVE","PENDING","RENEWED","DELAYED","SCHEDULED" D
.S STATIEN=$O(^ORD(100.01,"B",AX,0))
.I STATIEN>0 S STAT(STATIEN)=AX
Q:'$D(STAT)
N CNT S CNT=0
N CX
S CX="" F  S CX=$O(FILTER(CX)) Q:CX=""  D ;filters come in from C# in zero based array
.N TD ;orderable item ien
.S TD=$P(FILTER(CX),"^",1)
.N PAT ;AOI index has global reference like "4167;DPT("
.S PAT="" F  S PAT=$O(^OR(100,"AOI",TD,PAT)) Q:PAT=""  D
..N IDT ;inverse date/time, patient may have multiple orders
..S IDT=0 F  S IDT=$O(^OR(100,"AOI",TD,PAT,IDT)) Q:IDT'>0  D
...N ORNO ;order number in file 100
...S ORNO=0 F  S ORNO=$O(^OR(100,"AOI",TD,PAT,IDT,ORNO)) Q:ORNO'>0  D
....N STATUS
....S STATUS=$P($G(^OR(100,ORNO,3)),"^",3)
....Q:'$D(STAT(STATUS))
....N PATDFN I $P(PAT,";",2)="DPT(" S PATDFN=$P(PAT,";",1)
....Q:$G(PATDFN)'>0
....N PATWARD
....S PATWARD=$P($G(^DPT(PATDFN,.1)),"^",1)
....I PATWARD=WARD D
.....N EDATE ;external date
.....S EDATE=$$FMTE^XLFDT(9999999-IDT)
.....N ROWCOLOR S ROWCOLOR="" ;Default
.....N MF ;male,female, etc.
.....S MF=$P($G(^DPT(PATDFN,0)),"^",2)
.....I MF="M" S ROWCOLOR="LightGreen"
.....I MF="F" S ROWCOLOR="LightCoral"
.....N PATNAME S PATNAME=$P($G(^DPT(PATDFN,0)),"^",1)
.....N DRUGNAME S DRUGNAME=$P(FILTER(CX),"^",2)
.....S C9CRET($I(CNT))=PATDFN_"^"_PATNAME_"^"_WARD_"^"_DRUGNAME_"^"_EDATE_"^"_ORNO_
"^"_STAT(STATUS)_"^"_ROWCOLOR
Q
```

Now we'll need to create the remote procedures to access the above routine. These are what I created:

```
NUMBER: 3174                    NAME: C9C GERD DRUG FILTER          TAG: RPCFILT
 ROUTINE: C9CMODCD                  RETURN VALUE TYPE: ARRAY
 AVAILABILITY: PUBLIC
RETURN PARAMETER DESCRIPTION:
IEN^DRUGNAME
 M-CODE LINE (c): RPCFILT^C9CMODCD(C9CRET) ;RPC: C9C GERD DRUG FILTER
```

```
NUMBER: 3175                    NAME: C9C ON GERD DRUGS            TAG: RPC
  ROUTINE: C9CMODCD                 RETURN VALUE TYPE: ARRAY
  AVAILABILITY: PUBLIC
 DESCRIPTION:
 Returns Dashboard data for patients on GERD drugs specified in same
 routine
INPUT PARAMETER: WARD               PARAMETER TYPE: LITERAL
  MAXIMUM DATA LENGTH: 50            REQUIRED: YES
  SEQUENCE NUMBER: 1
 DESCRIPTION:
 Ward Name
INPUT PARAMETER: FILTER             PARAMETER TYPE: LITERAL
  MAXIMUM DATA LENGTH: 50            REQUIRED: NO
  SEQUENCE NUMBER: 2
 DESCRIPTION:
 Array of IEN^DRUGNAME
 RETURN PARAMETER DESCRIPTION:
 PATDFN_"^"_PATNAME_"^"_WARD_"^"_DRUGNAME_"^"_EDATE_"^"_ORNO_"^"_STAT(STATU
S)_"^"_ROWCOLOR
  M-CODE LINE (c): RPC^C9CMODCD(C9CRET,WARD,FILTER) ;RPC: C9C ON GERD DRUGS
```

Add these new RPC's to the rpc multiple of file 19.

```
Select OPTION NAME: C9C DASHBOARD       C9C DASHBOARD
Select RPC: C9C SHOW OVERRIDES// C9C GERD DRUG FILTER
  Are you adding 'C9C GERD DRUG FILTER' as a new RPC (the 42ND for this OPTION)
? No// y  (Yes)

Select RPC: C9C ON GERD DRUGS
  Are you adding 'C9C ON GERD DRUGS' as a new RPC (the 43RD for this OPTION)? N
o// y  (Yes)
```

Now let's create a picture for the icon. First create a .png that is 60x60 pixels. Using Microsoft Paint, this is a simple one I created:

Using my PngToBase64.exe, convert to Base64 and copy to the Windows clipboard.

Then define the module in the C9C DASHBOARD MODULES file. This is how my Definition looks. There are a couple of RPC's being used that already existed, one

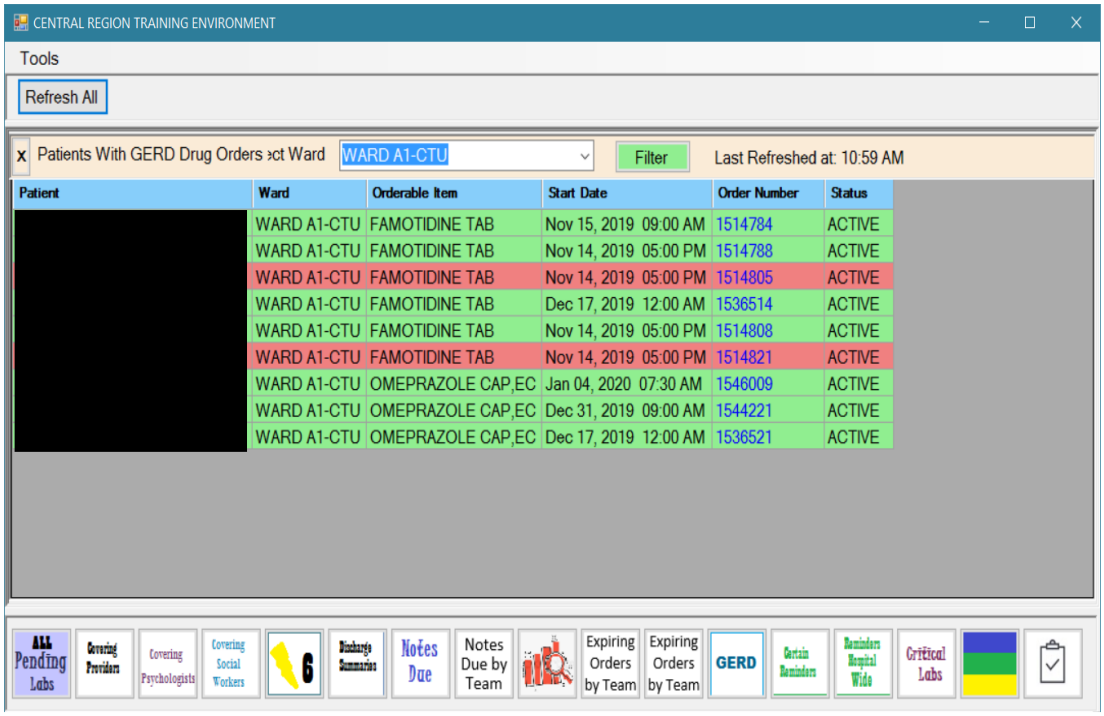
from CRH, one standard VA. They have already been registered in file 19 as well.

NUMBER: 17 NAME: Patients With GERD Drug Orders ACTIVE: YES
BUTTON TEXT: GD
BUTTON IMAGE:
137^80^78^71^13^10^26^10^0^0^0^13^73^72^68^82^0^0^0^60^0^0^0^60^8^2^0^0^0^
181^158^78^37^0^0^0^1^115^82^71^66^0^174^206^28^233^0^0^0^4^103^65^77^65^0
^0^177^143^11^252^97^5^0^0^0^9^112^72^89^115^0^0^18^116^0^0^18^116^1^222^1
02^31^120^0^0^1^128^73^68^65^84^104^67^237^211^193^113^194^48^16^133^97^21
5^69^61^12^117^80^65^134^59^55^247^96^106^224^70^1^46^0^46^212^64^20^73^20
0^210^111^73^198^147^108^130^50^251^102^47^203^130^248^108^175^187^174^191
^181^87^236^155^40^246^77^20^251^38^138^125^19^133^254^241^150^1^82^209^98
^1^82^209^98^1^82^209^98^1^82^209^98^1^82^209^98^1^82^209^98^1^82^209^98^1
^82^209^98^1^82^209^98^1^242^21^244^184^219^126^116^81^109^134^187^159^92^
207^155^116^228^106^119^49^51^254^202^149^29^229^167^211^177^179^0^185^132
^46^176^186^227^88^153^86^208^221^182^63^92^139^211^146^27^200^58^250^126^
216^251^227^102^119^104^56^153^46^160^221^53^36^121^126^115^127^118^179^21
1^209^253^48^189^36^63^13^199^186^75^98^128^172^162^107^38^155^31^67^63^19
8^161^143^166^12^144^85^244^101^112^7^249^167^246^108^109^217^91^146^95^15
^251^16^166^155^151^22^110^237^12^157^221^16^32^127^21^29^129^136^230^127^
165^1^114^253^122^248^211^83^244^194^122^132^11^8^43^91^188^211^223^94^143
^204^139^24^62^89^133^142^158^146^87^2^157^30^59^11^144^117^116^105^1^76^8
5^214^195^61^226^18^43^125^17^211^202^238^134^9^144^75^232^175^240^15^166^
163^87^160^227^87^194^44^125^6^93^18^155^0^249^10^250^239^3^164^162^197^2^
164^162^197^2^164^162^197^2^164^162^197^2^164^162^197^2^164^162^197^2^164^
162^197^2^164^162^197^2^164^162^197^2^164^162^197^2^228^191^64^183^81^236^
155^40^246^77^20^251^247^175^254^246^9^110^84^37^61^85^100^215^18^0^0^0^0^
73^69^78^68^174^66^96^130^
MODULE TYPE: DataGrid
DATAGRID COLUMN HEADER: DFN COLUMN DATA TYPE: TEXT ORDER: 1
DISPLAY THIS COLUMN: NO HINT: DFN
DATAGRID COLUMN HEADER: Patient COLUMN DATA TYPE: TEXT ORDER: 2

Type <Enter> to continue or '^' to exit:
DISPLAY THIS COLUMN: YES HINT: Patient Name
DATAGRID COLUMN HEADER: Ward COLUMN DATA TYPE: TEXT ORDER: 3
DISPLAY THIS COLUMN: YES HINT: Patient Ward
DATAGRID COLUMN HEADER: Orderable Item COLUMN DATA TYPE: TEXT ORDER: 4
DISPLAY THIS COLUMN: YES HINT: Orderable Item
DATAGRID COLUMN HEADER: Start Date COLUMN DATA TYPE: DATE ORDER: 5
DISPLAY THIS COLUMN: YES HINT: Start Date
DATAGRID COLUMN HEADER: Order Number COLUMN DATA TYPE: NUMERIC ORDER: 6
DISPLAY THIS COLUMN: YES CLICK RPC: ORQOR DETAIL
CLICK RPC PARAMETER COLUMN IENS: 6;1 HINT: Order Number
DATAGRID COLUMN HEADER: Status COLUMN DATA TYPE: TEXT ORDER: 7
DISPLAY THIS COLUMN: YES HINT: Order Status

DATAGRID COLUMN HEADER: Row Color COLUMN DATA TYPE: ROW COLOR ORDER: 8
DISPLAY THIS COLUMN: NO
DATA POPULATION RPC: C9C ON GERD DRUGS
POSSIBLE LOCATIONS RPC: C9C WARDS AND UNITS
LOCATIONS LABEL OVERRIDE TEXT: Select Ward
RPC GET POSSIBLE FILTER VALUES: C9C GERD DRUG FILTER
ACTIVE DIVISIONS: CENTRAL REGIONAL HOSPITAL
ACTIVE DIVISIONS: WHITAKER
ACTIVE DIVISIONS: BLACKLEY

So now if you run the Dashboard, you should see the additional icon. Once you drag it into the center window or double-click, be sure to click on the filter button since only the first drug will be selected by default. After that, the program will remember your selections. The data will refresh when you choose a different ward or filter, when you click on Refresh All, or every fifteen minutes. You can sort on any column. This is how mine looks:



Dashboard

Released Modules RPC Table

Module Name	DATA POPULATION RPC	POSSIBLE LOCATIONS RPC	RPC GET POSSIBLE FILTER VALUES	ACTION BUTTON RPC	DATAGRID COLUMN HEADER	CLICK RPC
All Pending Labs T-10 To T+5	C9C GET PENDING LABS	C9C WARDS AND UNITS			Order Number	ORQOR DETAIL
Covering Providers	C9C COVERING PROVIDERS MODULE					
Covering Psychologists	C9C COVERING PSYCHOL MODULE					
Covering Social Workers	C9C COVERING SW MODULE					
Lab Draws Today Not At 6AM	C9C LAB DRAWS OTHER THAN SIX	C9C WARDS AND UNITS			Order Number	ORQOR DETAIL
Missing Discharge Summaries	C9C MISSING DSCG SUMMARIES					
Notes Due	C9C MODULE GET NOTES DUE	C9C WARDS AND UNITS	C9C NOTES DUE ALL FILTERS		Last Completed	C9C DB MODULE NOTE TEXT
Notes Due	C9C MODULE GET NOTES DUE	C9C WARDS AND UNITS	C9C NOTES DUE ALL FILTERS		Rule Applied	C9C SHOW OVERRIDES
Notes Due By Team	C9C NOTES DUE BY TEAM	C9C GET ALL APPROPRIATE TEAMS	C9C NOTES DUE ALL FILTERS		Last Completed	C9C DB MODULE NOTE TEXT
Notes Due By Team	C9C NOTES DUE BY TEAM	C9C GET ALL APPROPRIATE TEAMS	C9C NOTES DUE ALL FILTERS		Rule Applied	C9C SHOW OVERRIDES
Notes Due Forecast	C9C FORECAST NOTES DUE	C9C ND GET TEAMS	C9C NOTES DUE ALL FILTERS		Last Completed	C9C DB MODULE NOTE TEXT
Notes Due Forecast	C9C FORECAST NOTES DUE	C9C ND GET TEAMS	C9C NOTES DUE ALL FILTERS		Rule Applied	C9C SHOW OVERRIDES
Orders Expiring Next 10 Days By Team	C9C EXPIRING FOR DASHBOARD	C9C ND GET TEAMS			Order Number	ORQOR DETAIL
Orders Expiring Next 5 Days By Team	C9C EXPIRING FOR DASHBOARD	C9C ND GET TEAMS			Order Number	ORQOR DETAIL
Patients With GERD Drug Orders	C9C ON GERD DRUGS	C9C WARDS AND UNITS	C9C GERD DRUG FILTER		Order Number	ORQOR DETAIL
Reminders	C9C DASHBOARD REMINDER MODULE	C9C WARDS AND UNITS	C9C C AND R FILTERS			
Reminders Hospital Wide	C9C ALL PATIENTS REMINDERS		C9C C AND R FILTERS			
Show Critical Chem Lab Results Last 7 Days	C9C GET CRITICAL LAB RESULTS					
Sticky Notes	C9C DASHBOARD STICKY NOTES	C9C WARDS AND UNITS	C9C FILTER STICKY NOTES	C9C FORCE STICKY NOTE EXP		
Unverified orders > T-30	C9C UNVERIFIED FOR DASHBOARD	C9C WARDS AND UNITS			Order Number	ORQOR DETAIL