

Getting the Bad Data Out!

Prepared by:

James R. Schell
Fire Fighter
City of Tukwila
444 Andover Park E
Tukwila, WA 98188
206 575 4404
jschell@ci.tukwila.wa.us

MyFireRules (www.myfirerules.com) is copyrighted by The Rules Guys and as such all rights are reserved.

FireRMS is a registered trademark of Bio-Key International (www.bio-key.com) and as such all rights are reserved.

The Problem...

We live in a very litigious society. Fire departments are now being seen as the next set of deep pockets that can be picked. The ability of the Fire Departments to protect itself through documentation of the incidents that they respond to is critical. Not only in protecting it from law suits, but also in protecting the assets that the department owns. Just look around your department. How many backboards do you have? How many KED devices do you have? How many Hare Traction splints do you have?

Tracking our aid equipment that is used on patients was becoming a dollar loss issue for my department. On average we run 700 auto accidents a year with most but not all patients being placed on a back board. There are those MCI's that occur in everyone's jurisdiction; train wrecks, bus wrecks etc. Furthermore, what about those patients that do not require a back board but one is used for ease of transportation? And what about Fall victims for example, we usually use a backboard on those patients too.

How do we track all this aid equipment? My department found that FAXing a piece of paper to the Main Station (we have 4) was not working. Here is how it was done. E53 would go on an incident of some type and use some type of "hard, reusable, non-disposable aid equipment". They would fill out a half sheet of paper and FAX it to the main station where it would be retrieved as a full piece of paper when someone would get around to taking it out of the fax machine. Then that person would have to transcribe the information onto another piece of paper in a three ring binder – our "Aid Equipment Out Logbook". Then the half piece of paper and the full size piece of paper would be placed into a recycle box.

Well lets see, that comes out to 4 ½ pieces of paper per auto accident that has to be recycled; $4 \times 700 = 5\frac{1}{2}$ ream of paper per year just to track Aid Equipment. This system also relies on humans to write and FAX information, and we all know how stuff falls through the cracks occasionally. I don't know about your department but we can only afford to lose so many pieces of \$300 equipment before our budget starts feeling the impact.

We would send someone to retrieve our aid equipment from the hospitals and there were supposed to be X number of back boards and Y number of splints at certain hospitals. Well our folks were coming back to the department with more aid equipment that what was listed as being sent out – obviously our "system" was not working.

The Solution...

How to solve this documentation problem was the question of the day. Our solution was to create a custom tab to track our aid equipment within our automated incident reporting system. We use BIO-key's FireRMS which allows for the creation of custom tabs and provides a "get the red out" feature to enforce required fields. But this was just an electronic piece of paper replacing a hard piece of paper. We needed a way to require the completion of the aid equipment tab under certain circumstances, in other words make it conditionally required or "sometimes required".

My Fire Rules is a "software toolkit" that provides the enforcement of logical business rules as defined by the local department for comprehensive validation of your incident reports. My Fire Rules currently works its magic on BIO-key's FireRMS only but theoretically could be expanded to other FireRMS products that use SQL (Structured Query Language). You create your rules using the My Fire Rules Toolkit. The rules are transformed into SQL statements and then combined into a SQL stored procedure that resides inside your SQL database waiting to find the "bad data". The user is prevented from saving the incident as "Complete" until all the bad data is resolved.

Creating the Custom Tabs & Related Rules:

Bio-Key's FireRMS™ allows us to create custom tabs. This is a pretty straight forward process. First you must have administrative rights on the program. Then you create the tab by using an assortment of input fields, checkboxes, date time fields etc. Then you save the tab and exit FireRMS. You must open Sql Utilities for FireRMS and lock your database and then reset the permissions on the database. Once this is done then you must exit Sql Utilities and reopen FireRMS and then go back into Tab creator and publish your tab. Once this is all done you have a new Tab to work with.

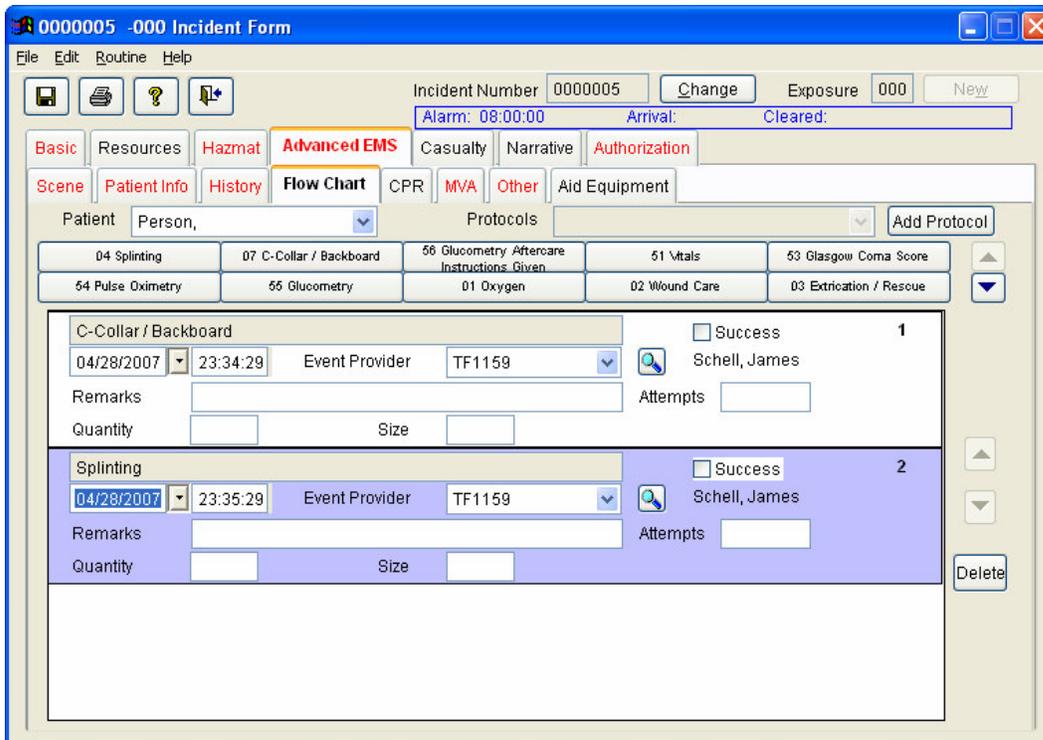
Here is the downfall of custom tab creation. If you use a checkbox, they cannot be made to be a mandatory or recommended field. Well most people equate a checkbox list as an efficient means of listing what was done or used. How do you enforce when a checkbox should and should not be marked? MyFireRules to the rescue.

We have a series of checkboxes and one text field on the custom tab that we created. This tab is designed to track our aid equipment sent out and to let us know what has not been returned to the department. There is also a text field specifically for items not found in the checkbox list. We made provision for using mutual aid company's equipment as well as the private ambulance's equipment. We even addressed the usage of disposable splints such as a cardboard splint or a finger splint which we would not want to retrieve. While this may not be a complete list based on your department, you can create your own and make it customized to your department.

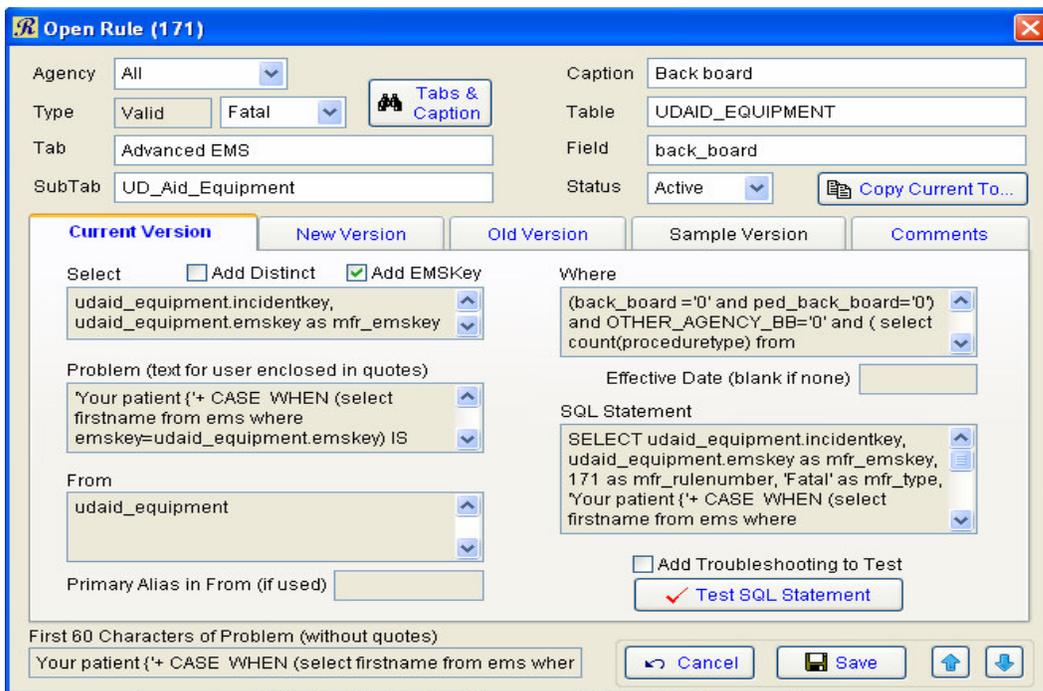
The screenshot shows a software window titled "0000005 -000 Incident Form". The menu bar includes "File", "Edit", "Routine", and "Help". The main area has several tabs: "Basic", "Resources", "Hazmat", "Advanced EMS" (highlighted), "Casualty", "Narrative", and "Authorization". Below these are sub-tabs: "Scene", "Patient Info", "History", "Flow Chart", "CPR", "MVA", "Other", and "Aid Equipment" (highlighted). The "Patient" dropdown menu shows "Person,". The "Aid Equipment" section contains three columns of checkboxes: "Select Our Aid Equipment Used", "Other Agency Equipment", and "Aid Gear Returned". The "Select Our Aid Equipment Used" column includes: Adult Back Board, Ped Back Board, Large Vac Splint, Med Vac Splint, Small Vac Splint, Adult Hare Traction Splint, Pediatric Hare Traction Splint, KED Splint, and Other Aid Equipment (with a text input field). The "Other Agency Equipment" column includes: Other Agency Back Board and Other Agency Splint. The "Aid Gear Returned" column includes: Aid Equipment Returned. A "Disposable Splint" checkbox is located at the bottom left of the equipment list.

While you can make fields mandatory and recommended within the design mode of FireRMS, you really do not want to make the text field above mandatory. Because this tab is linked to a patient, the program will create a child record in the table that was created when you created the custom tab, and making it “mandatory” would require you to enter data in the text box every time you have an Advanced EMS call for every patient. All custom tabs have a table created to store the data that is associated to the tab you created associated with either the Incident or the Patient.

This is where the SQL statement comes into play. Because you have a table with data in it, you can evaluate it. For example the tab pictured below shows two procedures entered. These procedures are specific to the patient displayed. Because we have a procedure inserted into the EMSVitalsAndProcedures table we can evaluate what the procedure was and check it against the aid equipment tab. While the SQL statement is a little complex it is really straight forward. If **this** is true and **that** is false then give me this message.



For example here is the rule created in the My Fire Rules Administrator's Kit. This rule checks for a missing back board checkbox on the aid equipment tab. It looks for the value of the procedure specifically procedure number 07(located on the top row of buttons second from the left in the preceding image) and looks to see if the end user has marked either adult back board, pediatric back board or other agency back board.



The rule is made up of several parts. The SELECT portion, the PROBLEM text, the FROM portion and the WHERE clause. So what we are doing here is selecting or choosing if you will, the fields we need to create our rule and then what text we want the end user to see as the problem or error message. Next we are deciding where we want our data to come from and finally, when to show the error message i.e. the where clause part.

My Fire Rules only allows SQL select statements. If you try to use a restricted word such as insert update or delete then you will receive a warning. You cannot make the rule active until you remove the restricted keyword(s) and the rule “Tests Successfully”.

So what we do with My Fire Rules is create an error message that tells the end user that they used a procedure code for a back board but did not mark a back board check box on the aid equipment tab. Within the error message we tell them to either remove the errant procedure code or to enable a check box on the aid equipment tab.

The same process applies to the splinting procedure code.

I hear some of you saying, what if they enable a check box on the aid equipment tab and don't have the corresponding procedure in the flow chart. If you change the logic a little on the SQL statement you can do just that.

With two rules you can check for the presence of the procedure with no corresponding aid equipment checked out or for the presence of equipment checked out with no corresponding procedure. I have a total of four rules for the aid equipment tab specific to the back board procedure and the splinting procedure. I have created a crystal report that now returns both details and summary information about the aid equipment checked out.

Regarding the text box for “Other Aid Equipment”, this is a free form 50 character field that allows our end users to enter equipment that is not listed as a check box. For example they left a large bp cuff on the patient when the patient went to the hospital. I know this never happens in your department, but it does on occasion in mine and this gives us a way to track that on my crystal report.

The crystal report is based on a start and end date that the end user specifies when they run the report. The end user can then perform the aid equipment retrieval and go back into FireRMS and go to each specific incident where there was equipment sent out and mark the Equipment returned check box. The next time the report is run then any equipment that has been marked as returned will not show up in the report.

This may seem like a little overkill, but hey, at least I can track what piece of equipment went with what patient. If I find I am short on splints that are supposed to be at a hospital, all I have to is track down those patients. Sometimes our Ambulances get diverted to another hospital and we do not know it. The report gives me an incident number, the patient's name, the equipment used on the patient and to what hospital they were sent. I

can simply call the ambulance company and find out which patient was diverted and now I would know where my equipment went.

You can even use this to track equipment left at a patient's residence simply by filling in the text box with that information. Again I know this never happens in your department...

Here is our aid equipment out report:

**AID EQUIPMENT SENT TO HOSPITALS
Between 4/1/2006 and 4/29/2007**

Auburn General Hospital	
Incident # / Date / Patient Name	0603696 10/14/2006 citizen, Jo
Equipment Used: Adult Backboard, Medium Vacuum Splint, KED Splint. See the Equipment left/sent with patient section for details about an item that was left/sent with the patient	
Backboards: 1	Pediatric BB: 0 Lg Vac Splint: 0 Md Vac Splint: 1 Sm Vac Splint: 0
Adult Hare Traction: 0	Pediatric Hare Traction: 0 KED SPLINT: 1

Group Health / Eastside	
Incident # / Date / Patient Name	0603696 10/14/2006 citizen, Jo
Equipment Used: Adult Backboard, Pediatric Backboard, Adult Hare Traction, Pediatric Hare Traction, Large Vacuum Splint, Medium Vacuum Splint, Small Vacuum Splint, KED Splint. See the Equipment left/sent with patient section for details about an item that was left/sent with the patient	
Backboards: 1	Pediatric BB: 1 Lg Vac Splint: 1 Md Vac Splint: 1 Sm Vac Splint: 1
Adult Hare Traction: 1	Pediatric Hare Traction: 1 KED SPLINT: 1

Highline Community Hospital / Riverton	
Incident # / Date / Patient Name	0603687 10/14/2006 citizen, Jo
Equipment Used: Medium Vacuum Splint.	
0603691 10/14/2006 citizen, Jo	Equipment Used: Adult Backboard,
Backboards: 1	Pediatric BB: 0 Lg Vac Splint: 0 Md Vac Splint: 1 Sm Vac Splint: 0
Adult Hare Traction: 0	Pediatric Hare Traction: 0 KED SPLINT: 0

Between 4/1/2006 and 4/29/2007

TOTALS:	
Backboards: 5	Pediatric BB: 1 Lg Vac Splint: 1 Md Vac Splint: 3 Sm Vac Splint: 1
Adult Hare Traction: 1	Pediatric Hare Traction: 1 KED SPLINT: 2

Other Equipment Left/Sent With Patient	
Auburn General Hospital	
PATIENT	citizen, Jo
INCIDENT ADDRESS	301 Baker Blvd
APARTMENT	
CITY	TUKWILA
Phone Number	(123)456-7890
INCIDENT NUMBER	0603696
ALARM DATE	10/14/06
OTHER EQUIP	large bp cuff left with patient
Group Health / Eastside	
PATIENT	citizen, Jo
INCIDENT ADDRESS	301 Baker Blvd
APARTMENT	
CITY	TUKWILA
Phone Number	(123)456-7890
INCIDENT NUMBER	0603696
ALARM DATE	10/14/06
OTHER EQUIP	pulse ox sent with pt

Expanding My Fire Rules...

The Tukwila Fire Department has 3 shifts of 19 personnel per shift, all filling out NFIRS and Medical Reports. We run on an average of 4500 to 5000 calls for service per year. Review that many reports for the multiple data entry problems (like bad zip codes or the wrong census track or the street name is misspelled etc.) and still keep one's sanity is a time consuming and never-ending job. This is where MyFireRules from the The Rules Guys comes into play.

Yes there are the NFIRS rules that all FireRMS software has to try to enforce, and the NFIRS Validation Tool helps with that, but NFIRS does not care if you put 12345 in the zip code field when your zip codes all begin with 98***.

You could put that every call occurs in a 419 (single family home) and the NFIRS Validation Tool would not care. It only wants to see a value in the field that is in the "List". So if we put bad data in guess what comes out? That's right... bad data. It is the GIGO principle garbage in garbage out.

For example, a rule could be defined to prevent the station field from being left blank. The resulting SQL statement would be:

:

```
SELECT Station
FROM Incident
WHERE station = ''
```

This statement simply looks in the incident table for any record where the station field is blank. These rules can be very simple or they can be very complex such as this.

```
SELECT EMSAdvanced.incidentkey, EMSAdvanced.emskey as mfr_emskey,
34 as mfr_rulenummer, 'Fatal' as mfr_type, 'Receiving Facility cannot be Blank,
if NO Transport THEN code [000] must be chosen for Patient= {'+
CASE WHEN (select firstname from ems where emskey=emsAdvanced.emskey)
IS NULL THEN (select top 1 RTRIM(lastname)+' , blank firstname' from ems
where emskey=emsAdvanced.emskey) ELSE (select top 1 RTRIM(lastname)+' ,
'+RTRIM(firstname) from ems where emskey=emsAdvanced.emskey)
END +'}' as mfr_problem, 'Advanced EMS-Other' as mfr_tab,
'Receiving Facility: Facility' as mfr_caption
FROM EMSAdvanced
WHERE receivingfacility IS NULL
```

This statement looks at the value of the field that tells where a patient was transported to, i.e. your local hospital. It is a required field for us. But not every patient gets transported. This tells the end user that they must fill in this field. Yes you can make the drop down box a required field when designing the custom tab for FireRMS but this only works if your "Hospital List" contains a "No Transport" code. There are other instances where

making a field “Mandatory” is not feasible because it is only required “IF...” which does not lend itself to the “mandatory” functionality.

In other words the tab with a “sometimes required” field on it does not turn red indicating that there are fields that need to be filled in. So the end user thinks they are done because they are in the “if it’s black don’t go back” mode and mark the report complete. Then you do a NFIRS or Local EMS export and those records and data fail to export because there were required fields that did not get filled in.

How do we overcome these limitations within our FireRMS? MyFireRules allows me send an error message to the end user via a crystal report for everything I determine that is “bad data”. The end user gets to correct the problem before I even see it. There are two levels of error messages: Warning and Fatal. The error level is chosen in the MyFireRules Administrators’ Tool Kit when you create each “rule”.

Fatal errors are just that fatal. The end user cannot save the incident marked as complete unless they fix what ever the error was. Warnings are just procedural things. In King County Washington, after a CPR event there are certain things that we must do for data documentation. We need to call the ROC (Resuscitation OutCome) hotline, we need to login to the county website and upload our electronic data from our defib machines, we need to etc etc... You get the picture. A rule was created to list the procedural things that the end user must do to comply with the County directives and displays this to the firefighter when they say they have completed their report (in the computer). This rule is activated by the data we put into the report. Specifically in this case, inserting a CPR procedure.

What about repeat false alarms at the same address? My Chief wanted a way to count the number of false alarms at the same address within a calendar year. If the number of calls was greater than two per year we now “throw a warning message” that instructs the officer filling out the report to send a copy to the Fire Marshal’s office for any needed follow up.

Any field that is connected to an incident can be evaluated. MyFireRules has taken my QA time and cut it in half right from the start when we turned it on and went “Live”. Over the next few months, I had managed to tighten up the rules we enforce to make our data even better. My last NFIRS Export of 1000 plus records went out error free to the State. I got zero errors back from the State compared to another agency I know of who had over 100 critical errors in just over 100 reports exported. Needless to say, they had to spend a lot of time to correct those reports and then export again and start the process all over.

Prior to using My Fire Rules I was spending a minimum of 7 to 8 hours per week fixing reports and getting data cleaned up for export; now I spend less that 3. I had to do this after hours or when ever I could sneak it in during the regular work day as I am on shift and have other duties that take priority such as running calls, training, inspections etc.

Another jurisdiction near me has reclaimed $\frac{3}{4}$ time from a full time employee who was tasked with reviewing incident reports by using My Fire Rules. Let's see, $\frac{3}{4}$ time from a \$40,000 dollar employee is \$30,000 in increased productivity carried forward year after year.

I have even found a way to validate my addresses against my occupancy module using MyFireRules. Yes I know there is the validate address button and the occupancy button on the BasicLocation tab. I have the street pick list filled in and yet I have users who insist upon filling it in by hand which leads to mistakes. Heck, I was even missing errors. There is just no way I could evaluate every report and find every mistake made. Currently our reports are evaluated against 160 rules; utilizing MyFireRules allows me to check for all these errors at the point of incident completion.

Conclusion:

While Bio-Key's FireRMS is extensible, via custom tab creation, it does limit us to what we can do as far as making certain fields required. While there is no one panacea piece of software that can do everything we need or want, we can make use of some tools to help our main piece of software do more. The combination of being able to Create a Custom Tab (and a User Defined table) AND the ability to enforce the use of that Tab through My Fire Rules gives us the true functionality that we need.

If you have any questions about creating custom tabs or about My Fire Rules please contact me at :

Jim.schell@myfirerules.com
www.myfirerules.com