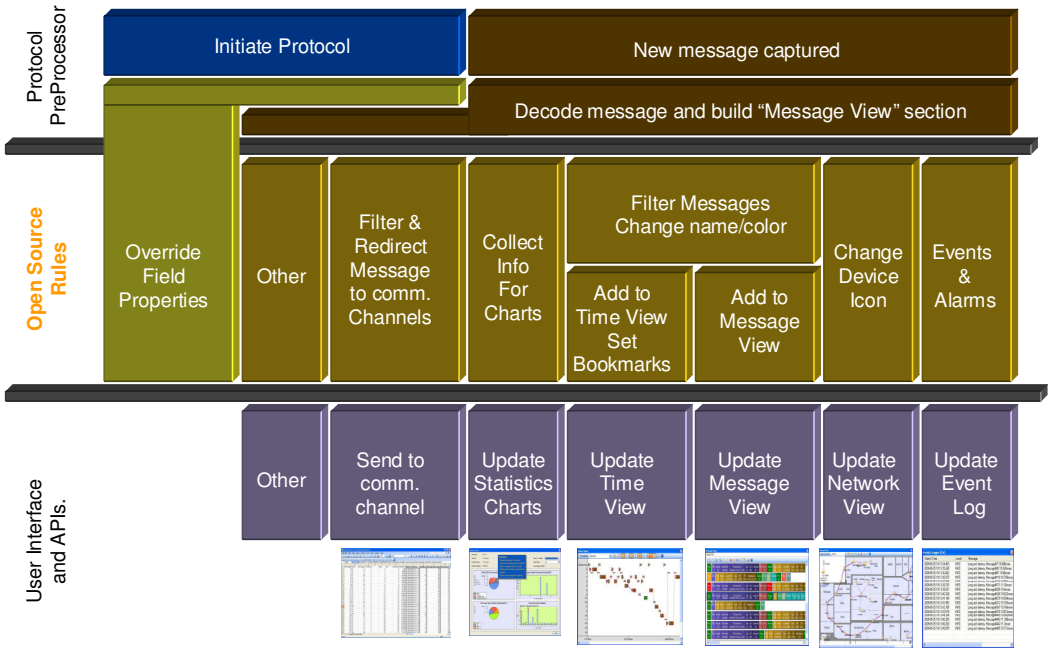
Perytons

# Perytons Open Source Rules

Rules are practically small pieces of code that enable the user (or Perytons support) to add features to the analyzer, or to modify existing analysis, without the need to add these features into the Perytons Analyzer core. Such features are aimed to serve specific needs or requirements by the user of the standard Perytons Analyzer platform. After they have been written they can be incorporated as an additional layer of the analysis process and into the different Peryton Analyzer active licenses of choice.

The open source rules enhance capabilities of the Perytons Analyzers' environment in different dimensions like monitoring of live-networks; change the look and feel of the existing views (time, message, network, etc.), definition of customizable statistic charts, interconnection to external applications through numerous communication ports and protocols, e-mail alerts, and more.

Rules are run whenever a new message is analyzed or a new message is displayed as explained within the following diagram:



The various Perytons Analyzers basic models already include some rules examples (the list of basic rules included in the standard products may change from time to time without prior notice).

These example rules are briefly described in the following list:

- Choose messages that will be automatically displayed in time-view
- Choose messages that will be automatically added into message view, and automatically expand some of their fields
- Set message ‘name’ and color
- Set the visible fields for message view, modify field name and hint
- Set the device icon that will be displayed in the network view
- Generate events and alarms
- Build data into customized charts

New rules can be written by any user who has the Peryton-Monitor add on license. Building, debugging and running rules is done within the analyzer GUI (user Interface) with no need for any additional development environment.

These rules can be used by any analyzer license; therefore Perytons Analyzer users can share rules between them.

When building a rule, all message properties decoded by the analyzer as well as device properties are available for the rule usage. Unlike with XML coding, with the Open Source Rules, all C sharp coding options are available so any condition can be written for the relevant rule operations.

In addition, being a piece of code, the rule can access other resources of the PC (such as communication sockets), to add a variety of features. For example, sending ZigBee NWL payload of received messages to another user application over UDP can be easily done by an appropriate rule. Such a rule would check if the message includes NWL payload, and if so, will build the relevant fields, e.g. timestamp, RSSI, etc., into a structure and send it over UDP to a user defined address.

Another example is related to users that have their proprietary encryption which they can't or don’t want to share. In this case “cryptographic experts” can build the encryption decoding engine within a rule that will override some of the message fields with the decoded/clear data. Since the resulting rule can run on the other Perytons Analyzer active installations, it is now just a matter of sending the resulting rule code to the different machines installed with analyzers and the encryption engine becomes an integrated part of all the Peryton Analyzers' user group within the specific company.