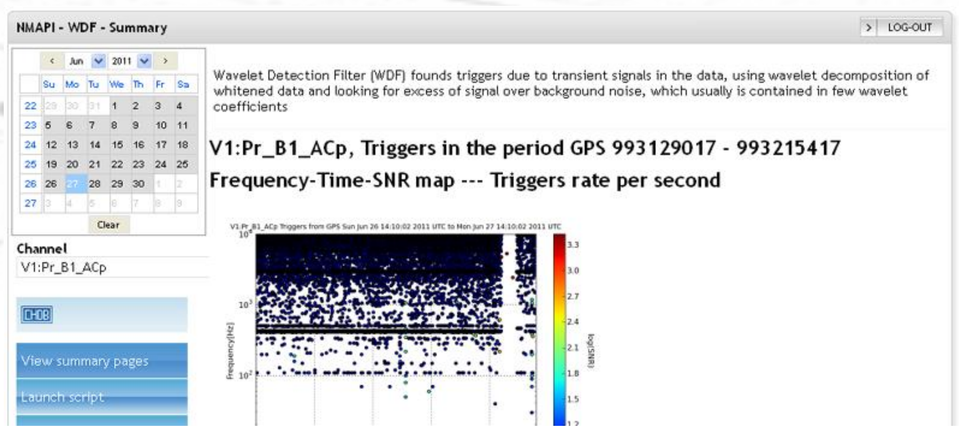
Noise monitor tools & their application to Virgo data

E. Cuoco*, A. Colla^ & G. Hemming*, for the Virgo Collaboration
*EGO, ^Università Sapienza, Roma

The understanding of noise for interferometric gravitational detectors is fundamental both in terms of enabling prompt reactions when attempting to mitigate noise disturbances and when setting up appropriate veto strategies. Monitor tools, which perform on-line and off-line noise analysis, such as transient signal detection, line identification algorithms, and coherence searches, were used to characterise Virgo detector noise. Here, we describe the framework in which these tools are integrated and show examples of their application.

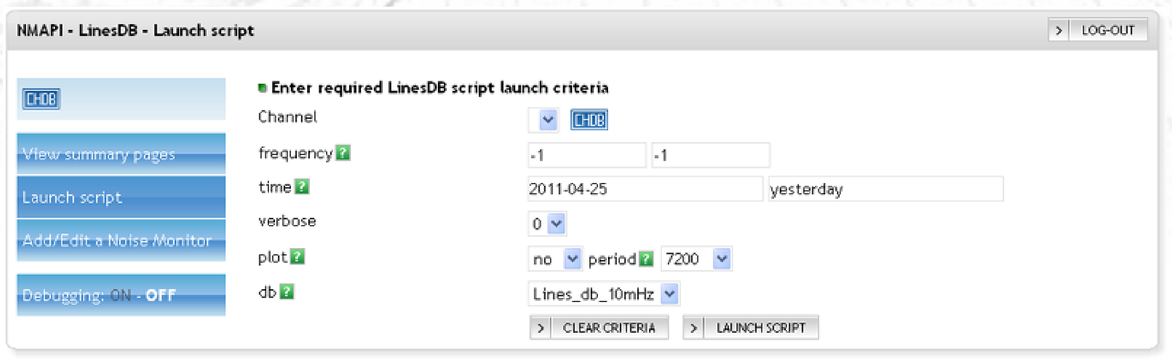
The idea was to have tools (Noise Monitors) processing either on- or off-line data, the results of which could then be archived in MySQL databases or similar storage locations. From these sources, users would be able to produce plots based on available data, including that which had been recently acquired.
The aim was to give, to both scientists on shift and the commissioning crew, instruments that are able to easily provide a snap-shot of noise behaviour at any given moment. The following fields were monitored: **Glitchiness** (several transient signal detection tools, e.g. *WDF*, which looks at energy excesses in a wavelet map); **Stationarity** (RMS in band); **Coherence** with auxiliary channels; and **NoEMi** for line identification and catalogue purposes (see A. Colla's poster).

NMAPI (*Noise Monitor Application Programming Interface*) [1]



NMAPI provides a framework into which different Noise Monitors can be plugged and then configured, enabling users with little knowledge of web development or web programming languages and standards, to interface their scripts to an Internet audience. The application enables authenticated users to not only access information, but also to produce results, e.g. HTML pages or text files, based upon bespoke search criteria specified by them via the user interface.
This mode of producing results dynamically means that, not only are users able to produce highly-configurable results, but storage requirements are reduced to a minimum, given that files are created on-the-fly.

LinesDB [2]



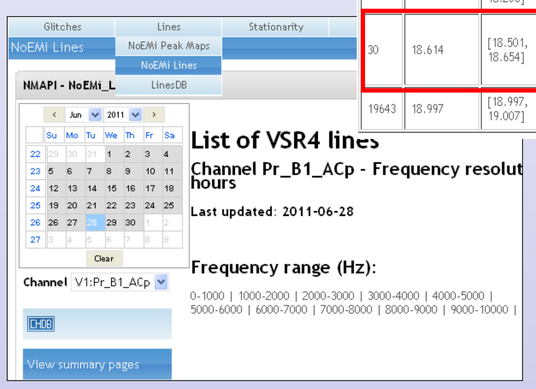
LinesDB provides a web interface to an archive of identified line events stored in a MySQL database.
Users can query data to produce lists of lines that meet specified criteria (frequency, time, location in which the line was measured, etc.) and can also add metadata (type of instrument used in measurement, general information relating to the line, etc.) to a line, rendering it as 'known' in the process.
This functionality enables the history of a line to be recorded and evaluated over time. Providing not only a single point of reference for new line information, but also a useful resource in tracking lines that may appear and disappear over time.

How does NMAPI work?

Using the *NoEMi Lines* and *NoEMi PeakMaps* Noise Monitors, for example:

1. A user examines the *NoEMi Lines* summary page, which contains a list of all of the lines archived in *LinesDB*.
2. An inspection of the results shows a line of particular interest (for example the 18.6Hz which presumably seems to be due to the motor of an air conditioning machine in the Virgo Central building and which enters the data via the seismic path).
3. A look at the plot related to the line confirms that the line is worth examining in further detail.
4. The user studies the history of the disturbances in the Dark Fringe and the coincident auxiliary channels versus time, using the calendar tool available to the *NoEMi PeakMaps* summary pages, in order to choose a day of interest.
5. Via the NM script interface, the user launches the steering script, using the criteria detailed in point #2, providing a detailed peak-map or line plot for a specific channel.
6. The user confirms the results by looking at the *Coherence* tool
7. From here it is possible, using *LinesDB*, to search for, or insert, related metadata, describing the line and providing qualitative information regarding, for example, where and how the line has been measured.

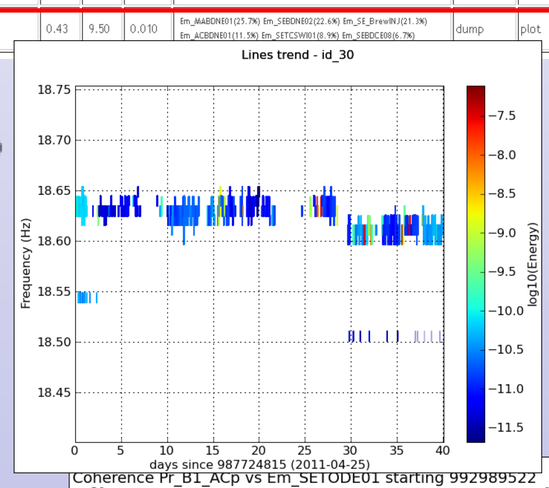
1



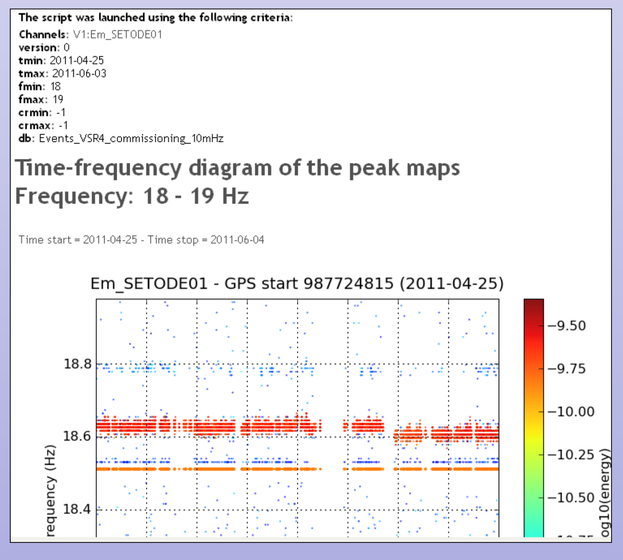
2

Id	Mean Frequency (Hz)	Frequency range (Hz)	First seen	Presence (periods)	Mean pers	Mean CR	Mean sigma (Hz)	Coincident auxiliary channels	Verbose dump	Plot
63081	17.996	[17.996, 18.015]	2011-05-24_16	133	0.37	9.27	0.010	Em_ACBDWE01(30.7%) Em_MABDCE01(13.3%) Em_SEBDNE02(5.3%)	dump	plot
7707	18.090	[18.082, 18.206]	2011-04-25_06	149	0.71	9.57	0.010	Em_ACBDWE01(22.0%)	dump	plot
30	18.614	[18.501, 18.654]	2011-04-25	574	0.53	22.34	0.018	Em_SETODE01(7.8%) Em_SE_EIB_04(7.7%) Em_SEDBDL03(7.7%) Em_SE_CryoT(7.6%) Em_SE_BrewINJ(7.6%) Em_SETCSW101(7.6%) Em_SEBDCE01(7.6%) Em_AC_EIB(7.6%) Em_SETCSNI02(7.5%) Em_SEBDCE08(7.4%) Em_ACBDCE01(6.8%) Em_SE_LB_04(6.7%)	dump	plot
19643	18.997	[18.997, 19.007]	2011-05-20_16	156	0.43	9.50	0.010	Em_MABDNE01(25.7%) Em_SEBDNE02(22.6%) Em_SE_BrewINJ(21.3%) Em_ACBDNE01(11.5%) Em_SETCSW101(8.9%) Em_SEBDCE08(6.7%)	dump	plot

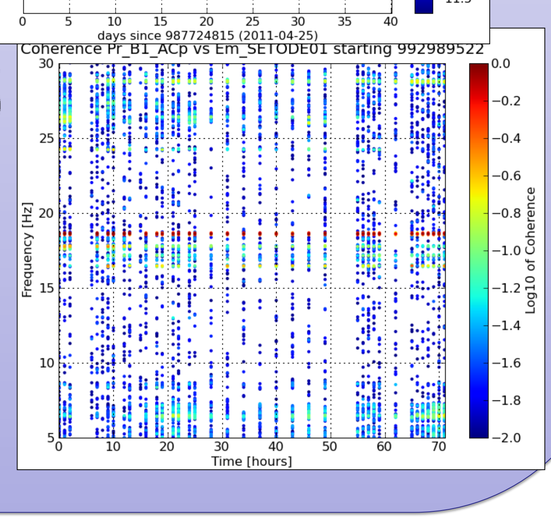
3



5



6



Next steps for the Noise Monitors and NMAPI

• Increase linkages with other applications in the Virgo PHP-MySQL environment (Hardware Inventory & Integration Database, Virgo Logbook, Detector Monitoring System, etc.). Linkages to the Channels Database already exist, but these will be explored more fully via the implementation of a Connections Database, enriching available information and helping to ensure healthy cross-application communication.
• Integration with data and results produced by other similar detectors to facilitate cross-correlation of advanced detector results.

References

• F.Carbognani, A.Colla, E.Cuoco, G.Hemming, *Lines Database Web Interface Software Requirements* (VIR-0227A-11), https://tds.ego-gw.it/ql/?c=8281, 2011.
• A.Colla, E.Cuoco, G.Hemming, *Noise Monitor Application Programming Interface (NMAPI) Software Requirements*, (VIR-0226B-11), https://tds.ego-gw.it/ql/?c=8280, 2011.