# NMAPI and NoEMi for Advanced Detectors

- Interest in running NoEMI tool on LIGO data has been raised by LIGO people
- From the aLIGO detchar plans -> aLIGO software requirements:

> - Line tracking and finding tools
>   - Baseline: Noemi (port from Virgo) and fscan
>   - people: Nelson Christenson

- NoEMI is one of the Noise Monitors tools and is integrated in the NMAPI framework.

- We take the opportunity and propose to work on a **common interface** for Noise Monitors for the Advanced Detectors.

NoeMi workflow
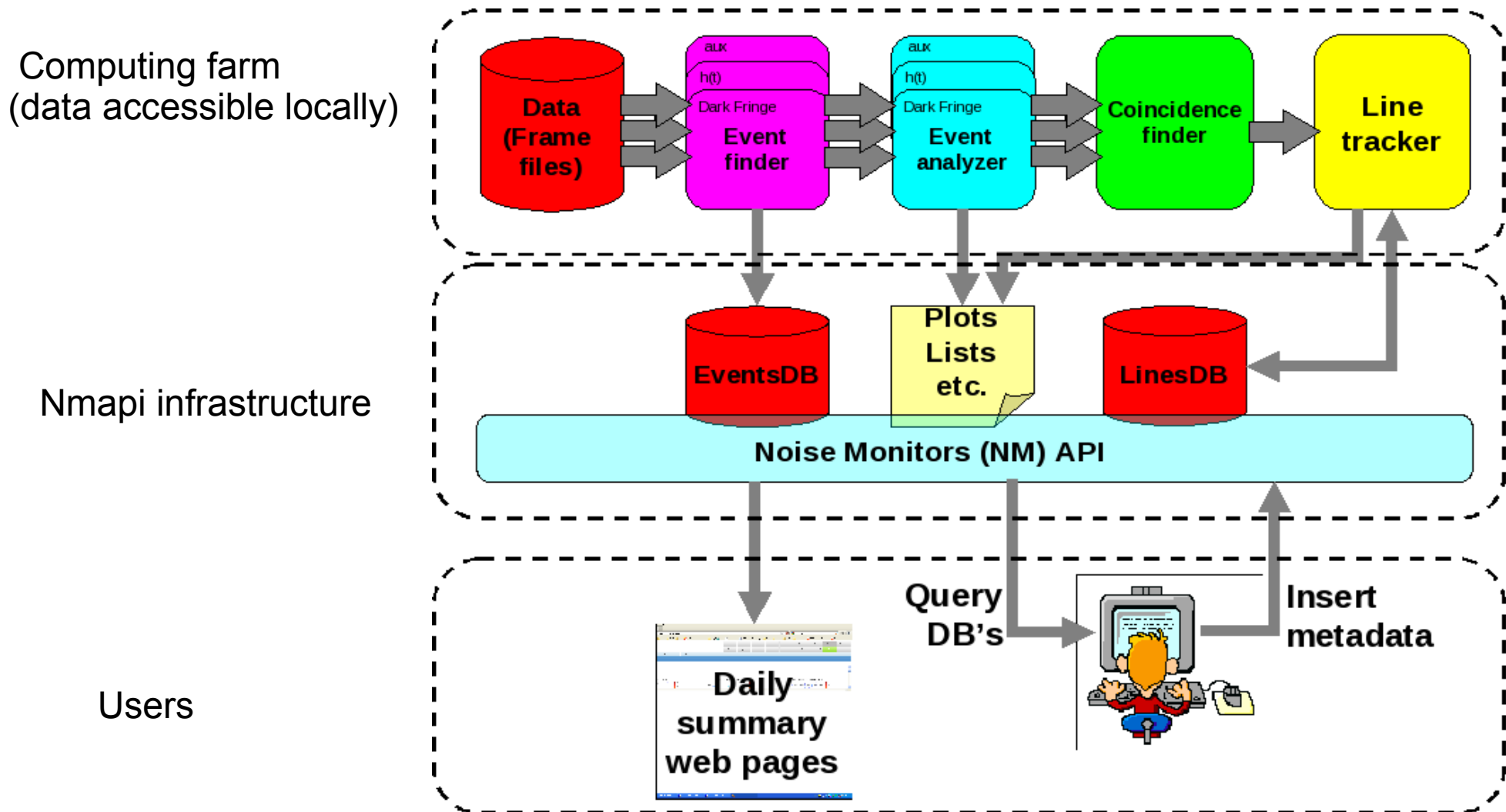
How NoEMi worked so far

- NoEMi analyzed VSR3 and VSR4 data in-time

- Run at Cascina, on the on-line farm

  - ➡ see Computing requirement slides

- Events and Lines databases set on a MySQL server at Cascina (olserver35)

  - ➡ see Storage requirement slides

- Summary pages and NMAPI on-the-fly pages published on the Cascina web server

Run on LIGO data

Two/three??? possible solutions (and two/three different additional requirements):

- Run NoEMi on a LIGO farm (e.g. Caltech) and fill database in Cascina

    - Requires NoEMi installed and tested on LIGO farm

    - **MySQL port open for incoming & outgoing traffic**

    - Possibility to send locally created plots & web pages to Cascina web server (e.g. via LDR?)

- Run NoEMi at Cascina on transferred LIGO Data

    - Requires to send LIGO RDS at Cascina

    - Computing and storage requirements at Cascina increase (more dedicated online nodes, solve concurrent data access problems ...)

- "Hybrid" solution (????)

    - Run event finder on a LIGO farm

    - Transfer partial results (peak maps, ASCII files) to Cascina (via LDR)?

    - Run the rest of the analysis at Cascina

In any case: solve firewall problem which prevents access to Virgo web pages from outside INFN network!

Test Setup for NoEMi

- We plan to perform tests to identify the best solutions

NoEMi Computing requirements

- Analysis of 1 single channel requires < 2 GB RAM and one CPU

    - Line tracker requires 8 GB RAM

- Data (frame files) must be accessible locally

    ➡ E.g. at Cascina data disks are NFS mounted

An important point to be solved is to have a fast and reliable access to the data by the online nodes. During the past Virgo runs the concurrent access to the data disks by the noise monitors caused slow-downs and, from time to time, "Stale NFS" problems. For this reason we were forced to run only one channel in parallel on each machine, which slowed down the daily analysis.

    ➡ A solution may be to FrCopy on the local disk only the channels analyzed by each node

- Noise Monitor & database server available at Cascina

- Python 2.5.2 + some modules (matplotlib, numpy, MySQLdb, markup)

NoEMi Storage requirements

What	Disk size / 1 day	Disk size / 1 year
Log Files - DF and auxiliary channels (1) (2)	100 MB	40 GB
Log Files – h(t) (3)	700 MB	260 GB
Log files - DF + h(t) + 27 auxiliary channels (1)	3.5 GB	1.3 TB
Events database - DF and auxiliary channels	50 MB	20 GB
Events database - h(t)	200 MB	70 GB
EVF database - DF+h(t)+27 auxiliary channels	1.6 GB	600 GB
Lists of lines & plots for the daily pages	0.4 GB	150 GB
Lines database	**2 MB**	**0.7 GB**
Plots for Lines table web page (4)	100 MB	100 MB
(1) Log files can be deleted/zipped after the analysis		
(2) the sizes refer to the 20 KHz sampled channels; the channels sampled at lower frequencies are smaller.		
(3) h(t) analysis is done with a lower CR thresholds, therefore its size is bigger		
(4) Lines table is rewritten everyday and its size is practically constant		

The numbers listed in the table refer to the analysis of the Virgo data in one single configuration (e.g. 1 mHz frequency resolution, daily updates).

During VSR3 and VSR4 NoEMi was run in 2 different configurations (the above and 10 mHz resolution, 2 hour updates). Therefore the storage effectively used was roughly 2 times the numbers listed in the table.

NMAPI upgrade

NMAPI currently sits on one single pc.

It sits and waits for requests, passed via HTTPS.

Upon request receipt, the information is passed to the dedicated Noise Monitor *Steering Script*, which deals with retrieving requested results from data source.

Distributed-NMAPI architecture
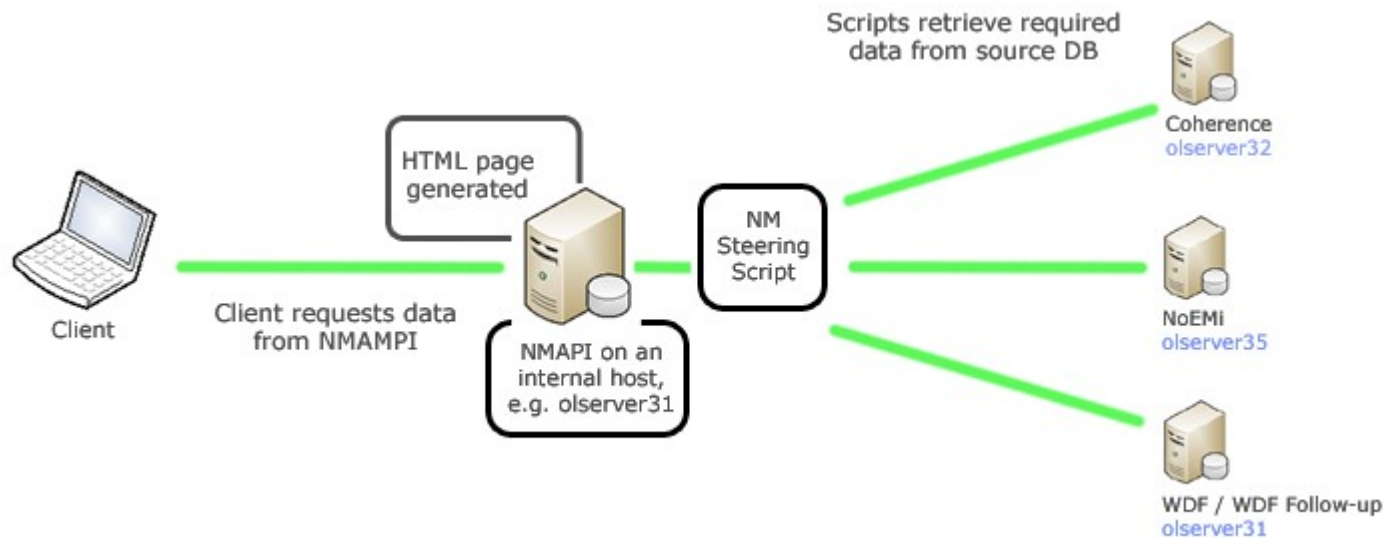
- We propose to make NMAPI working on more detectors data.

- Since for Advanced Detectors the computational needs will be higher, we think about a distributed architecture

- The Web interface will be the same for all the detectors

- The framework is extensible thanks to its "plug-in" architecture: new NMs can be added in the future
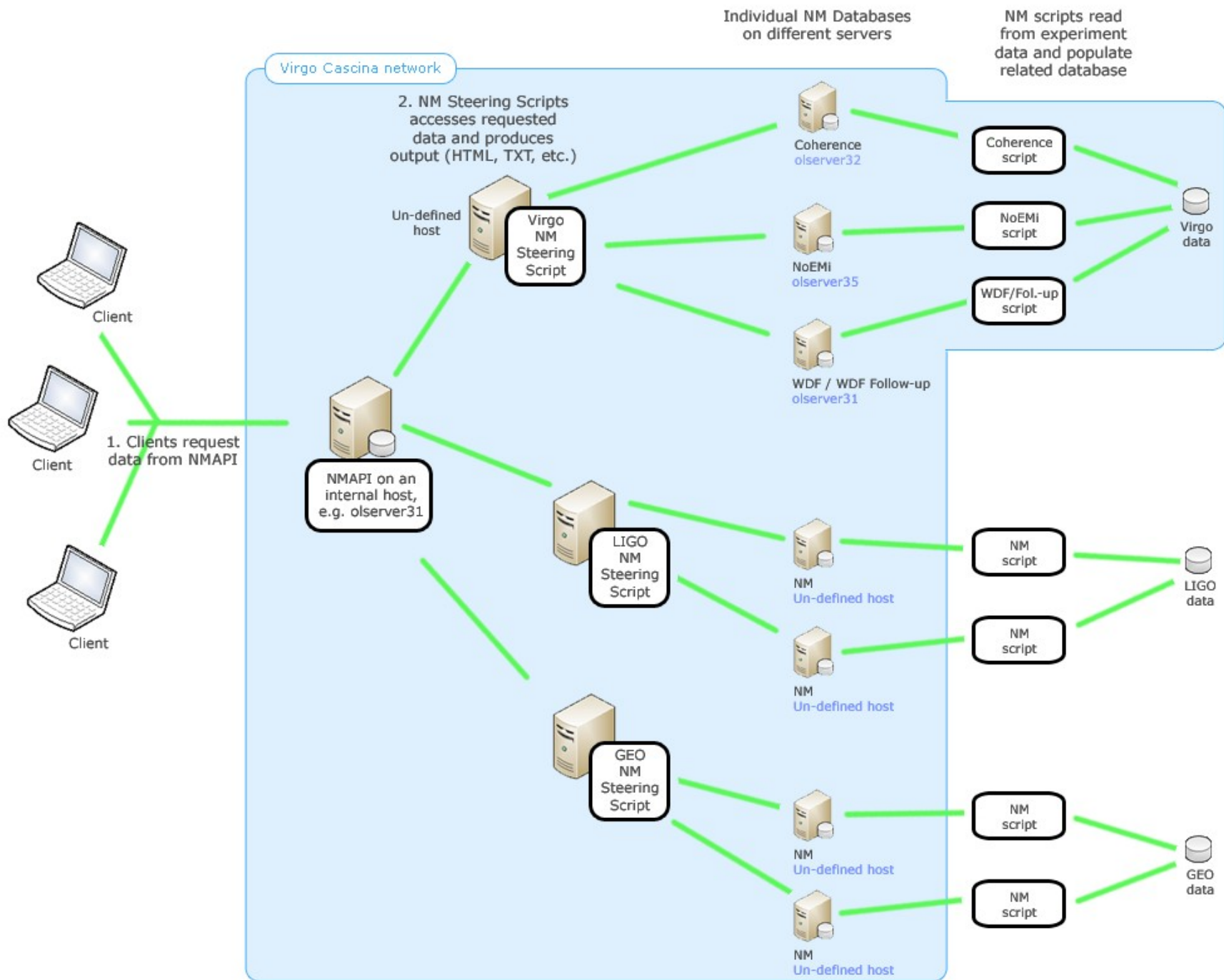
D-NMAPI: common interface?

- In the aLIGO page
  https://wiki.ligo.org/foswiki/bin/view/DetChar/ALIGODetcharPlan

under the section "More neboulous brainstorming area..." there are
  ideas about implementing common summary pages GEO-like,etc..

- It could be worth to try working together LIGO-GEO-Virgo for a
  common DetChar Monitoring interface.