# Virgo DQR checks



## In a nutshell

<table>
<tr><th>Task Name</th><th>Purpose</th><th>Allowed states</th><th>Outputs</th><th>Review</th><th>Additional information</th></tr>
<tr><td>bruco</td><td>Brute-force coherence during and around the event</td><td>HIN</td><td>4 web areas: 2 (Standard + ENV) * 2 (during the event + before)</td><td rowspan="3">VirgoDQR machinery extensively tested by running each new version of the package (SVN trunk + SVN tags) on various events<ul><li>In standalone mode: independently from all GraceDB instances</li><li>With upload of the DQR results to GraceDB</li></ul></td><td></td></tr>
<tr><td>data_ref_comparison_INJ</td><td>Data/reference comparison plots for the Injection subsystem</td><td>HIN</td><td>pdf file</td><td>Contents validated by subsystem experts</td></tr>
<tr><td>data_ref_comparison_ISC</td><td>Data/reference comparison plots for the Interferometer Sensing and Control subsystem</td><td>HIN</td><td>pdf file</td><td>Contents validated by subsystem experts</td></tr>
<tr><td>data_ref_comparison_SBE</td><td>Data/reference comparison plots for the Suspended Benches subsystem</td><td>HIN</td><td>pdf file</td><td rowspan="3">DQRs run on specific GPS times corresponding to various states of the detector(s)<ul><li>Full network on</li><li>V1 on, H1 and L1 off</li><li>All 3 detectors off</li><li>Lock lost around the time of the event</li><li>Event during lock acquisition</li><li>Lock acquired a few seconds before the event.</li></ul>In all cases, the DQR outputs are looked at check by check to compare its outputs with the expectation --</td><td>Contents validated by subsystem experts</td></tr>
<tr><td>decode_DMS_snapshots</td><td>Provide the status of the Detector Monitoring System (DMS) around the time of the event</td><td>HIN</td><td>Text file (output of the DMS snapshot process) + 1 link to the DMS playback per processed DMS snapshot.</td><td></td></tr>
<tr><td>dqprint_brmsmon</td><td>Scan DQ flags based on combinations of band-limited RMS computed on various channels (mostly</td><td>Pass<br>HIN</td><td>Table listing all the flags scanned and providing their status: missing, not active (over the whole</td><td></td></tr>
</table>

| | | | | | |
|---|---|---|---|---|---|
| | auxiliary) | | GPS scanned) or active (closest time difference w.r.t. the event time). | when relevant. This is done by<br>• The VirgoDQR package coordinators<br>• The check developpers<br>• The reviewers<br>Feedback and questions are provided<br>• during the weekly DetChar meeting<br>• by e-mail (through the DetChar mailing list or privately)<br>• on the Wiki<br>Each case is followed-up until resolved or understood.<br>Improvements were also done based on the experience gathered during the 2018 two week-long O2 data replays and during ER13 as well (although that run was much less satisfactory than anticipated as there was very little coincident time between the detectors). | |
| dqprint_key_dqflags | Scan important data quality flags | Pass<br>HIN<br>Fail | Table listing all the flags scanned and providing their status: missing, not active (over the whole GPS scanned) or active (closest time difference w.r.t. the event time). | | |
| dqprint_dqflags | Scan all Virgo data quality flags | Pass<br>HIN | Table listing all the flags scanned and providing their status: missing, not active (over the whole GPS scanned) or active (closest time difference w.r.t. the event time). | | |
| generate_dqr_json | Produces the json file expected by the data-quality-report framework | Not relevant | dqr.json<br>dqr.ini | | Tested each time a DQR is run; check results wouldn't be accessible if that code were not working |

| | | | | | |
|---|---|---|---|---|---|
| gps_numerology | Check whether the event GPS time is 'suspicious' | Pass<br>HIN | Only the data-quality-report state + summary | | |
| omicronplot | Glitch distribution in Virgo around the event | HIN | Set of plots showing the glitch distribution one hour around the time of the event | | |
| omicronscanfull2048 | Full Omicron scan over Virgo channels with sampling frequency greater than 2048 Hz | HIN | Full Omicron scan report webpage | | |
| omicronscanfull512 | Full Omicron scan over Virgo channels with sampling frequency lower than 2048 Hz | HIN | Full Omicron scan report webpage | | |
| omicronscanhoftH1 | High-resolution spectogram of H1 h(t) for the event | HIN | h(t) Omicron scan + associated full Omicron scan report | | |
| omicronscanhoftL1 | High-resolution spectogram of L1 h(t) for the event | HIN | h(t) Omicron scan + associated full Omicron scan report | | |
| omicronscanhoftV1 | High-resolution spectogram of V1 h(t) for the event | HIN | h(t) Omicron scan + associated full Omicron scan report | | |
| query_ingv_public_data | Provide an URL allowing to query the INGV public | HIN | Link pointing to a webpage on the INGV website | | |

| | | | | | |
|---|---|---|---|---|---|
| | database for earthquakes close to Virgo | | | | |
| scan_logfiles | Scan all VPM logfiles around the time of the event and merge all extracted snippets in a single, time-ordered file | HIN | Text file providing the log entries of all the VPM processes during the scanned GPS range of interest. The log entries are sorted by increasing GPS time. | | |
| upv | Run UPV on the past 24 hours before (and including) the event | Pass<br>HIN | UPV scan of the past 24 hours. Table showing coincidences in the GPS range of interest + full UPV report. | | |
| virgo_noise | Test the stationarity of the Virgo noise around the event | Pass<br>HIN | Quantitative statement in the data-quality-report summary + supporting plot showing the Rayleigh spectrum for the GPS range of interest | | |
| virgo_status | Status of the detector during the event | Pass<br>HIN<br>Fail | Detailled script output + supporting plots showing the ITF Mode and the ITF_LOCK state during | | |

| | | | | | |
|---|---|---|---|---|---|
| | | | the GPS range of interest | | |
| weather_sea_ENV | Check weather conditions and sea activity around the event | HIN | Pdf file | | Contents validated by subsystem experts |

(*) A check can also return an Error state if it detects an issue that has been implemented in the code. If, for whatever reason, the output json file is missing whereas the check should be completed, a default json file is produced, with an Error state and indicating that something did not work well for that particular check.

# Detailled description

## BRUte-force COherence

The bruco algorithm is run in 2 configurations: all Virgo auxiliary channels (STANDARD) and only environmental channels (ENVIRONMENT). Moreover, for each configuration the bruco analysis is run twice:

- over 300s including the time of the event
- over 300s before the time of the event

The idea of this check is to compare both analyses and identify a change in the coherence with auxiliary channels.

**Typical latency:** ~1 hour

**How to interpret the test results**

- Open both analysis reports. Identify the frequency rows with the highest coherence values (red rows). Is there any differences?

## Data/Reference comparison plots

The goal of these checks is to produce a set of plots (time series, spectra) that compare data around the event with data from a reference period. There are currently three Virgo systems using this functionality:

- INJ: injection system
- ISC: interferometer sensing and control (locking and alignment)
- SBE: suspended benches

## DMS snapshots

The Detector Monitoring System (aka DMS) takes snapshots of the Virgo detector health every ~10 seconds. These snaphshots, in json format, are labelled according to the GPS time when the snapshot was taken and archived. The script accesses these archives, extracts the snapshots of relevance (script parameters with default values are used to set the boundaries of the GPS range to be scanned), processes them and displays the result in the form of a 'pretty' text file.

That text file displays various information extracted from each snapshot: information from Metatron about the Virgo status, the status of the DMS providers and of the various systems monitored by the DMS, the list of channels muted and shelved and finally the list of all active flags (alarms) at the time of the snapshot. That output is repeated for the different snapshots found in the GPS range to be scanned.

In addition, the script generates links to the DMS playback API that allows one to view the DMS GUI at the time of the selected snapshots.

**Typical latency:** ~1 min

**How to interpret the test results**

- The test returns 'fail' if no snapshot is found -- that should never happen during O3.
- Otherwise it returns 'human input needed'.

## Environment flags (BRMSMon)

Environmental signals from microphones, magnetic, seismometers… are processed by an online application called BRMSMon. Band-limited RMS are computed and large deviations are flagged. This DQR check scans all BRMSMon channels and reports any active flags around the time of the event (+/- 5 s). The check fails if at least one flag is active. In the DQR report, all the flags are listed. The first column indicates the status of each flag:

```
NOT-ACTIVE = the flag is NOT active
+0s        = the flag is active AT the time of the event
+Xs        = the flag is active AFTER the time of the event.
-Xs        = the flag is active BEFORE the time of the event
```

**Typical latency:** ~1 min

**How to interpret the test results:**

- This test should complete rather quickly, O(1 min). If the test report is not available after a few minutes, there is a problem.
- Some BRMSMon flags are produced using the DARM signal (LSC_DARM). If there is a GW signal, they are expected to fire. These flags should be ignored.
- If an environmental flag is active at or around the time of the event, it needs to be investigated. For example, you should review the Omicron scan and check the environmental channels

## Data quality flags

Data quality flags are generated online. This check scans a selection of "important" flags and reports active flags around the time of the event (+/- 5 s). The check fails if at least one flag is active. In the

DQR report, all the slected flags are listed. The first column indicates the status of each flag:

```
NOT-ACTIVE = the flag is NOT active
+0s        = the flag is active AT the time of the event
+Xs        = the flag is active AFTER the time of the event.
-Xs        = the flag is active BEFORE the time of the event
```

**Typical latency:** ~1 min

**How to interpret the test results:**

- This test should complete rather quickly, O(1 min). If the test report is not available after a few minutes, there is a problem.
- If a flag is active at or around the time of the event, it needs to be checked-out. For example, you should review the Omicron spectrogram and check that the glitch is actually visible. You should also check the auxiliary channel used to build the flag (Omicron or dataDisplay)

## Data quality key flags

Data quality flags are generated online. This check scans a selection of "important" flags and reports active flags around the time of the event. The check fails if at least one flag is active. In the DQR report, all the selected flags are listed. The first column indicates the status of each flag:

```
NOT-ACTIVE = the flag is NOT active
+0s        = the flag is active AT the time of the event
+Xs        = the flag is active AFTER the time of the event.
-Xs        = the flag is active BEFORE the time of the event
```

**Typical latency:** ~1 min

**How to interpret the test results:**

- This test should complete rather quickly, O(1 min). If the test report is not available after a few minutes, there is a problem.
- If a flag is active at or around the time of the event, it needs to be checked-out. For example, you should review the Omicron spectrogram and check that the glitch is actually visible. You should also check the auxiliary channel used to build the flag (Omicron or dataDisplay)

## DQR json generation

This is not a check but this is a key part of the Virgo DQRs. Based on the list of checks that are going to be run, it generates the json file that is needed by LIGO's data-quality-report framework. With this input, one will be able to navigate between the different tabs and browse the results of each check: its state, the associated summary plus any additional content: text, table, plots.

## GPS numerology

This module tests how 'random' the GPS time of the candidate. It triggers a 'human input' if that GPS time is '(too) close' from the nearest

- half-second
- second
- minute
- quarter of the hour
- hour
- day

Script parameters (with default values) allow one to define what 'close to a particular GPS time mean.

**Typical latency:** ~1 min

**How to interpret the test results**

- If the output is 'pass', the event GPS time is far away from any 'special' GPS time around.
- If the output is 'human input needed', the event GPS is close to at least one of the 'special' GPS times around. The summary message provides the details of the timing coincidences found by the module.

## Glitch distribution around the event

h(t) glitch distributions (omicron triggers) are plotted around the time of the event (1 hour). These plots are useful to detect abnormal glitch behavior.

**Typical latency:** ~30 min

**How to interpret the test results:**

- Unusual glitch rate around the event?
- Presence of high-SNR glitches around the event.
- Identify a high glitch rate at a specific frequency? → usually a hint for a glitch family.

## Omicron scan of high-frequency channels

Omicron is configured to scan ALL high-frequency Virgo auxiliary channels existing at the time of the event. In the DQR report, a link is provided to the full Omicron web report: channels are displayed only if some power excess is detected around the time of the event (SNR>7 in a 1s time window centered on the event).

**Typical latency:** ~2 h

**How to interpret the test results:**

- Open the full scan report and review all the spectrograms.
- If a channel is displayed, look at the signal and compare it with the signal measured in the strain data. Same morphology? Is it exactly coincident in time? in frequency?
- Warning: some channels are "unsafe". It means that the signal is correlated to the dark fringe signal by construction. These channels should be ignored. In doubt, contact the experts.

## Omicron scan of low-frequency channels

Omicron is configured to scan ALL low-frequency Virgo auxiliary channels existing at the time of the event. In the DQR report, a link is provided to the full Omicron web report: channels are displayed only if some power excess is detected around the time of the event (SNR>7 in a 1s time window centered on the event).

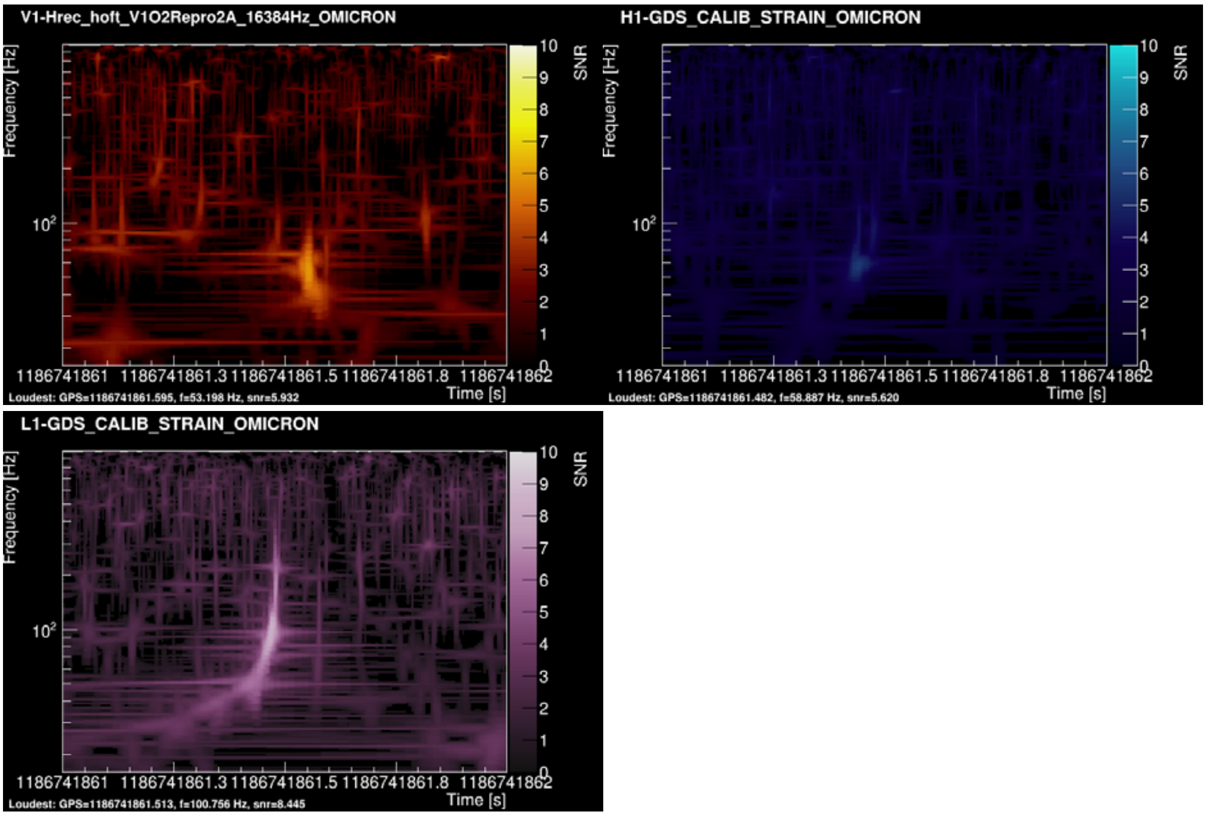
**Typical latency:** ~5 min

**How to interpret the test results:**

- Open the full scan report and review all the spectrograms.
- If a channel is displayed, look at the signal and compare it with the signal measured in the strain data. Same morphology? Is it exactly coincident in time? in frequency?
- Warning: some channels are "unsafe". It means that the signal is correlated to the dark fringe signal by construction. These channels should be ignored. In doubt, contact the experts.

## High-resolution spectrograms

High resolution spectrograms of h(t) signals are produced for the LIGO and Virgo detectors. h(t) signals are processed with Omicron configured to analyze the data with a high time-frequency resolution. This check is used to visualize the (possible) GW signals in the 3 detectors. For example:




In the DQR report, a link is provided to the full Omicron web report.

**Typical latency:** ~5 min

**How to interpret the test results:**

- Do you recognize the signal morphology? A CBC typical signal? A known glitch morphology?

Something unknown?
- Go to the full Omicron web report. Several spectrogram views are available. Play with the different views and try to better isolate the signal.

## INGV close earthquake DB query

INGV, the Italian equivalent of the US Geological survey (USGS) provides a public DB listing all earthquakes detected in Italy and around, that is close to Virgo. Earthquakes with a low magnitude may be a concern if they are very close from the site but they are usually not reported by the USGS. Therefore, querying the INGV public DB is complementary to the seismon software which provides earthquake early warnings using as input an USGS private DB. The issue is that the latency of the public DB is not know as entries are added by humans. The scripts provides almost immediately the URL to use to query the DB but a given earthquake may only appear in the DB few tens of minutes after its occurrence. Therefore, one should check regularly that URL to see if new earthquakes appear. If the page is empty, that means that no earthquake is listed in the DB (yet).

**Typical latency:** ~1 min to provide the query DB URL; wait 30 minutes-1 hour to make sure that nearby earthquakes are all listed; come back a few hours later to make sure that the list is indeed complete.

**How to interpret the test results**

- The state of theck is always 'human input needed'.
- If the URL leads to an empty page, that means that no earthquake exist in the scanned time range.

## Logfiles scan

Logfiles from all the known online processes are scanned by this Python module, in order to extract their contents in the GPS range scanned around the GPS time of the candidate (script parameters, with default values, are used to set the boundaries of this GPS range). The logged messages are all gathered in a text file after some formating. The text file contains four columns

1. the log message GPS time (log entries are sorted by increasing GPS time)
2. the logfile where it comes from (that filename contains the name of the process which wrote that message)
3. the severity of the message
4. the message itself.

**Typical latency:** ~10-20 min

**How to interpret the test results**

- If this check completes, its state is always 'human input needed' as experts should look at the logfile contents to see whether there is some issue to be reported.

## UPV event time test

The data of the last 24h are processed with UPV. The UPV algorithm measures the correlation between

Omicron triggers in an auxiliary channel and DARM Omicron triggers. If the correlation is high enough a list of veto segments is produced. With this check, the time of the event is compared with the resulting veto segments. The check fails if one of the veto is found to be active in a 10 ms time window around the event time. In addition, the efficiency of the veto is reported.

**Typical latency:** ~30 min

**How to interpret the test results:**

- The test fails if a coincidence is found with an auxiliary channels veto.
- A high-efficiency number indicates that the glitch population is large.

## Noise stationarity around the event

The Rayleigh spectrum is computed for h(t) over the data 600 seconds around the time of the event and for the frequency band 20-500 Hz.
Two quantities are computed: RMAX (max value of the Rayleigh spectrum over the frequency band) and RMEAN (mean value of the Rayleigh spectrum) .

**Typical latency:** ~10 min

**How to interpret the test results:**

- For stationary data, the Rayleigh spectrum is 0.5 over all the frequency band.
- h(t) is considered non-stationary (and human check is needed) as soon as RMAX>1.0 and RMEAN>0.7

## Virgo status

This module checks the status of the Virgo detector around the time of the event. It mainly checks two things:

- the detector Mode, that should be SCIENCE when we are taking good physics data;
- the state of the top-level automation node, called Metatron, that provides the actual status of the detector. For O2 and until now (beginning of ER14), the only state that allows the SCIENCE mode to be triggered is LOW_NOISE_3 (state index = 160). Soon (before O3 starts), a new state will share this property: LOW_NOISE_3_SQZ (name to be confirmed), with state index = 170 (value to be confirmed). The new state will imply that squeezing is up and working well, whereas the old LOW_NOISE_3 state mean no squeezing. Obviously the O3 goal is to be in LOW_NOISE_3_SQZ state + SCIENCE mode but should there be an issue with the squeezing, one would stick to LOW_NOISE_3 + SCIENCE configuration.

Plots are produced to visually show the evolution of the detector state and Mode during the GPS segment of relevance. In addition, a text file provides various additional information that could allow to characterize better the status of Virgo at the time of the event.

**Typical latency:** ~1 min

**How to interpret the test results**

- If Virgo is in LOW_NOISE_3(_SQZ) + SCIENCE, the check obviously returns 'pass'.
- If Virgo is not in SCIENCE Mode and that its state index is lower than the one of LOW_NOISE_3, the check returns 'fail'.
- If Virgo is not in SCIENCE Mode but that the detector is state is consistently LOW_NOISE_3(_SQZ), the check returns 'human_input_needed'. One would have to assess whether the detector was able to take good physics data at the time of the event although the SCIENCE mode button had not been hit yet.
- Otherwise (in particular if there is a transition LOW_NOISE_3 <→ LOW_NOISE_3_SQZ during the GPS range of interest), the check returns 'fail'.

## Weather conditions and sea activity

This reports a summary of the weather conditions and of the sea activity around the time of the event.

-- Main.Narnaud - 24 Feb 2019

This topic: DetChar > DetCharDQRChecks
Topic revision: