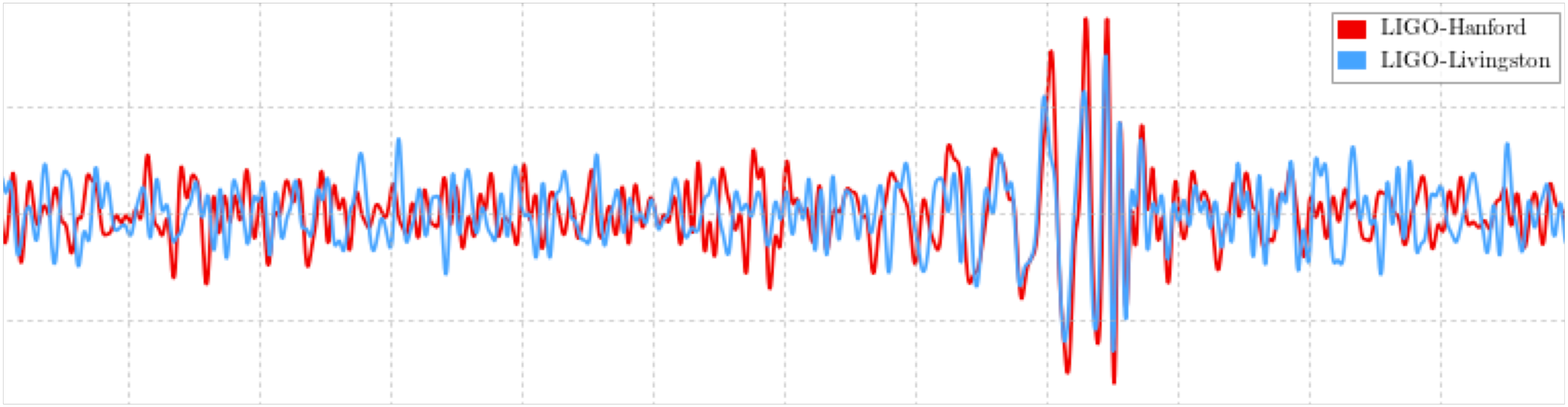
LIGO-Virgo data, data quality, calibration



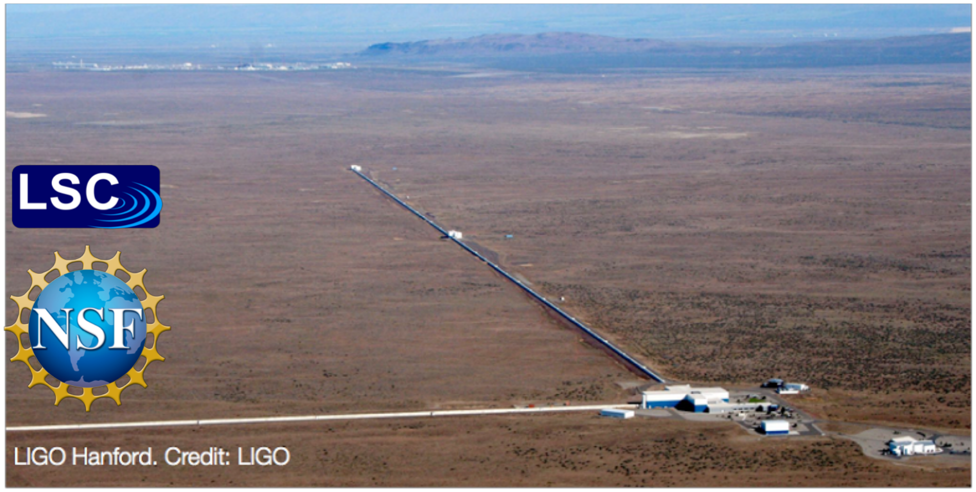
Alan J. Weinstein
LIGO Laboratory, Caltech
On behalf of the
LIGO Scientific & Virgo Collaborations

GWOSC Open Data Workshop, Paris, April 2019
https://dcc.ligo.org/LIGO-G1900XXX/public

LIGO-Virgo-GEO Detector network

LIGO Livingston. Credit: LIGO

Virgo observatory. Credit: Virgo



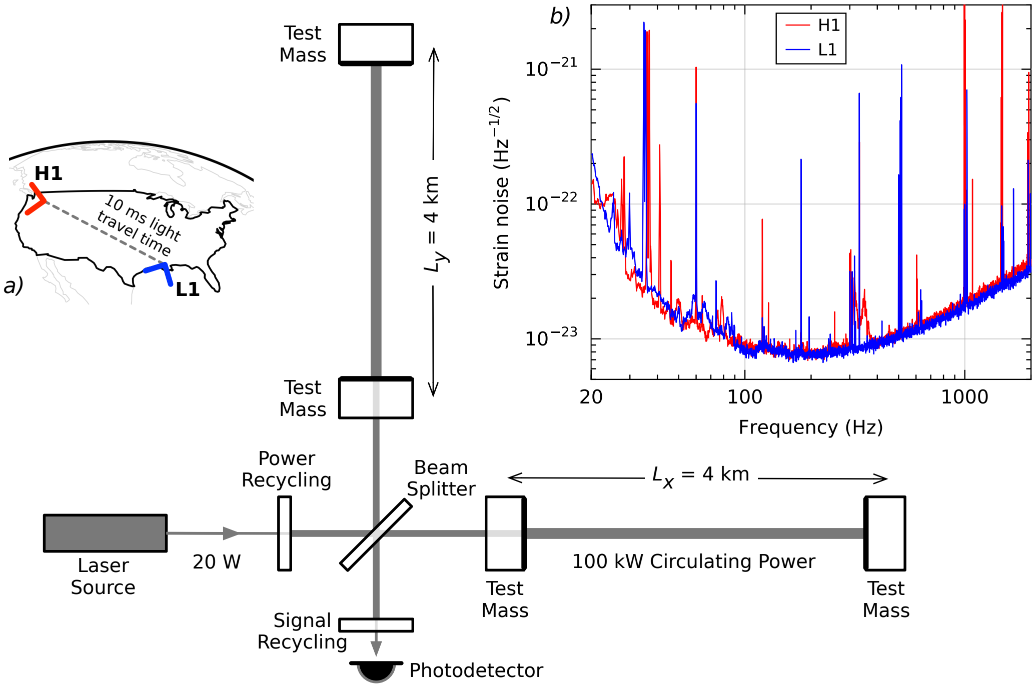
LIGO Hanford. Credit: LIGO

GEO observatory. Credit: GEO

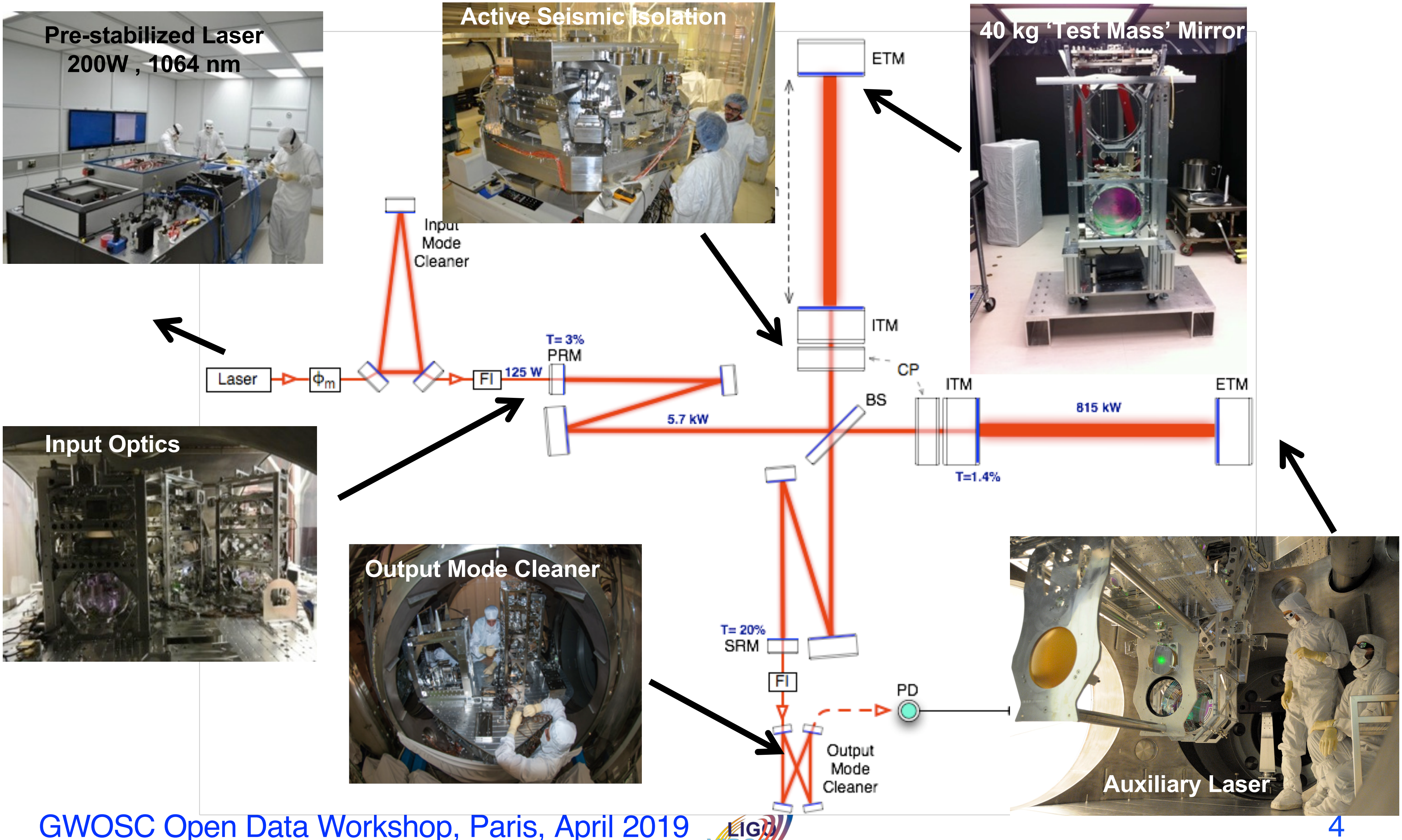
Coming: KAGRA (Japan), LIGO-India

Why are the LIGO & Virgo data so complex?

- Because the detectors are so complex!
- We often show simplified diagrams of our detectors
- But they are orders of magnitude more complex than the simple diagrams suggest!
- Why? Because we are measuring displacements on the order of 1×10^{-20} meters … not so easy!
- Our detectors push many different technologies to (and beyond!) their limits, making use of an enormous range of experimental techniques & tricks.
- aLIGO has around a dozen major sub-systems, and hundreds of smaller sub-systems.



Advanced LIGO Interferometer



Advanced LIGO data channels

- Advanced LIGO logs some **400,000** data channels (!) from *dozens* of sub-systems and thousands of servo control systems, each with multiple sensors and actuators.

- Only ONE is the strain channel h(t), used for GW astrophysical data analysis

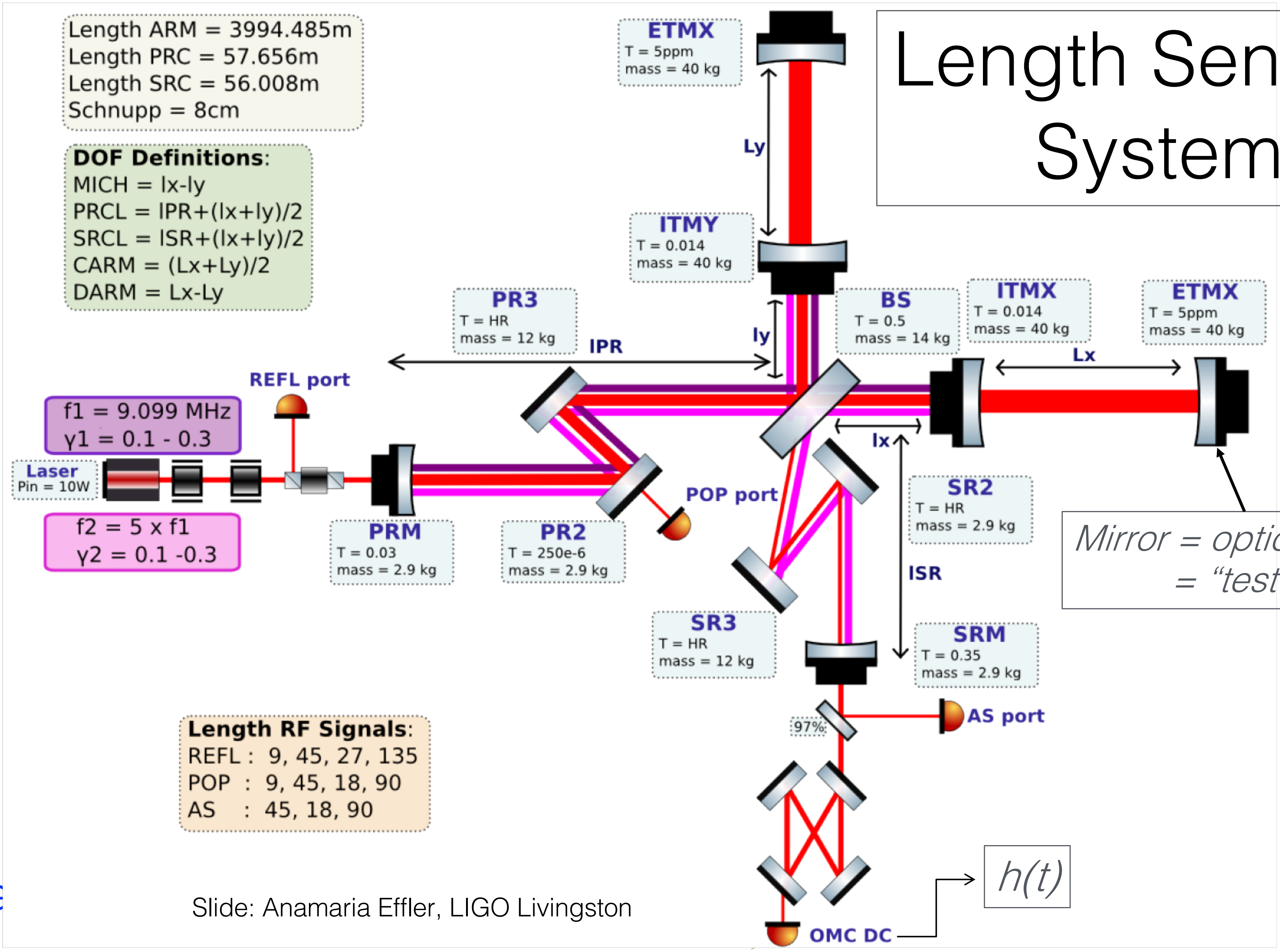
- All of the many engineering techniques and tools that go into our detectors serve one key purpose: to “shunt” environmental and/or instrumental noises to *any* other channel except the strain channel.

- This leaves the strain channel with minimal couplings to any “terrestrial” noises, while still being fully sensitive to GW strain (the stretching of space between the mirrors).

- All those other channels exist to witness the various noises and help in improving the isolation of the strain channel to those noises (i.e., reducing the coupling of terrestrial noises to the GW strain channel).

Advanced LIGO Servo Control

Length Sensing System



Length ARM = 3994.485m
Length PRC = 57.656m
Length SRC = 56.008m
Schnupp = 8cm

DOF Definitions:
MICH = lx-ly
PRCL = lPR+(lx+ly)/2
SRCL = lSR+(lx+ly)/2
CARM = (Lx+Ly)/2
DARM = Lx-Ly

Length RF Signals:
REFL : 9, 45, 27, 135
POP : 9, 45, 18, 90
AS : 45, 18, 90

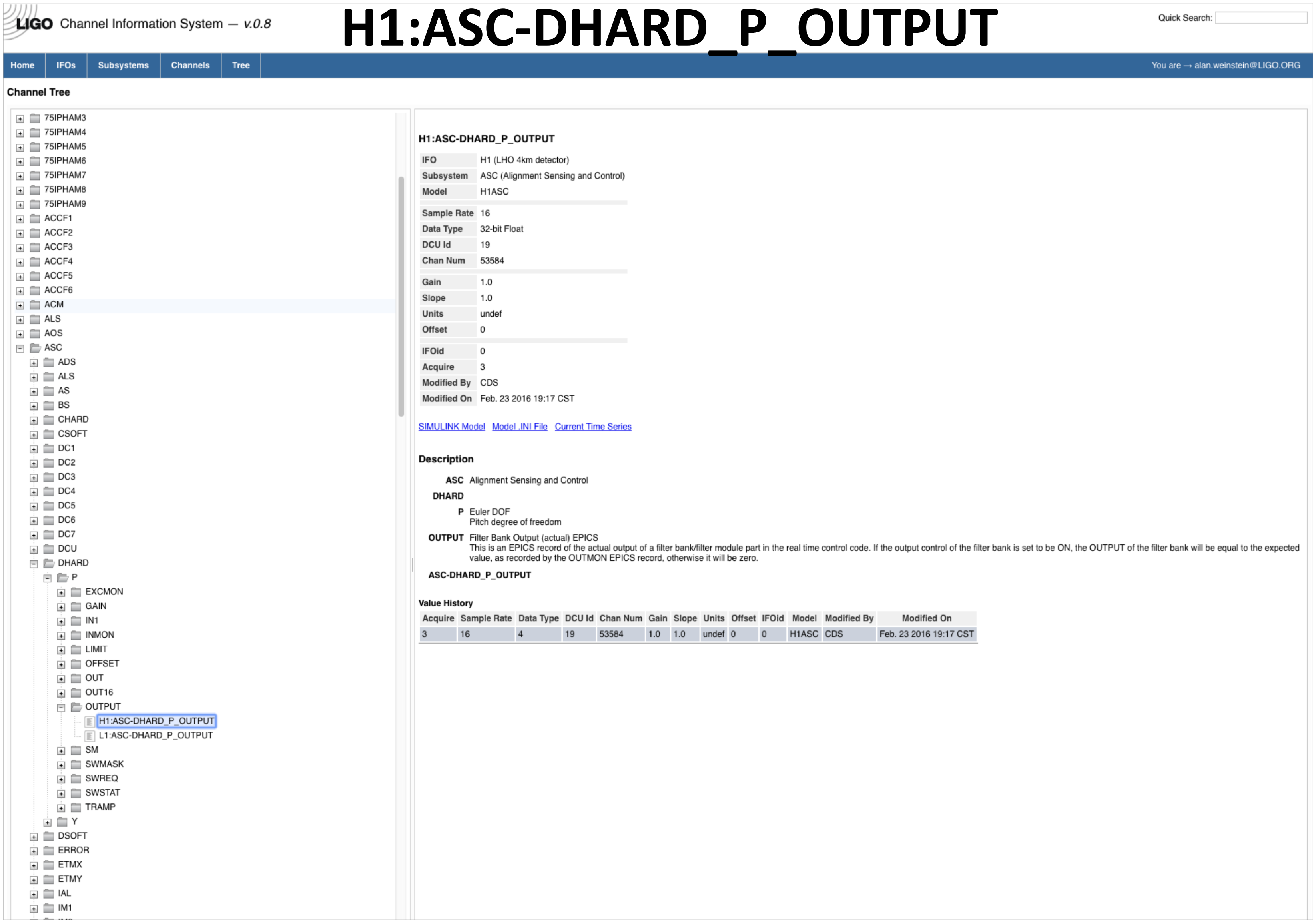
Mirror = optic
= "test mass"

h(t)

Slide: Anamaria Effler, LIGO Livingston

One of the ~400K channels (a rather important one)

H1:ASC-DHARD_P_OUTPUT



H1:ASC-DHARD_P_OUTPUT

IFO	H1 (LHO 4km detector)
Subsystem	ASC (Alignment Sensing and Control)
Model	H1ASC
Sample Rate	16
Data Type	32-bit Float
DCU Id	19
Chan Num	53584
Gain	1.0
Slope	1.0
Units	undef
Offset	0
IFOid	0
Acquire	3
Modified By	CDS
Modified On	Feb. 23 2016 19:17 CST

SIMULINK Model Model .INI File Current Time Series

Description

ASC	Alignment Sensing and Control
DHARD	
P	Euler DOF Pitch degree of freedom
OUTPUT	Filter Bank Output (actual) EPICS This is an EPICS record of the actual output of a filter bank/filter module part in the real time control code. If the output control of the filter bank is set to be ON, the OUTPUT of the filter bank will be equal to the expected value, as recorded by the OUTMON EPICS record, otherwise it will be zero.

ASC-DHARD_P_OUTPUT

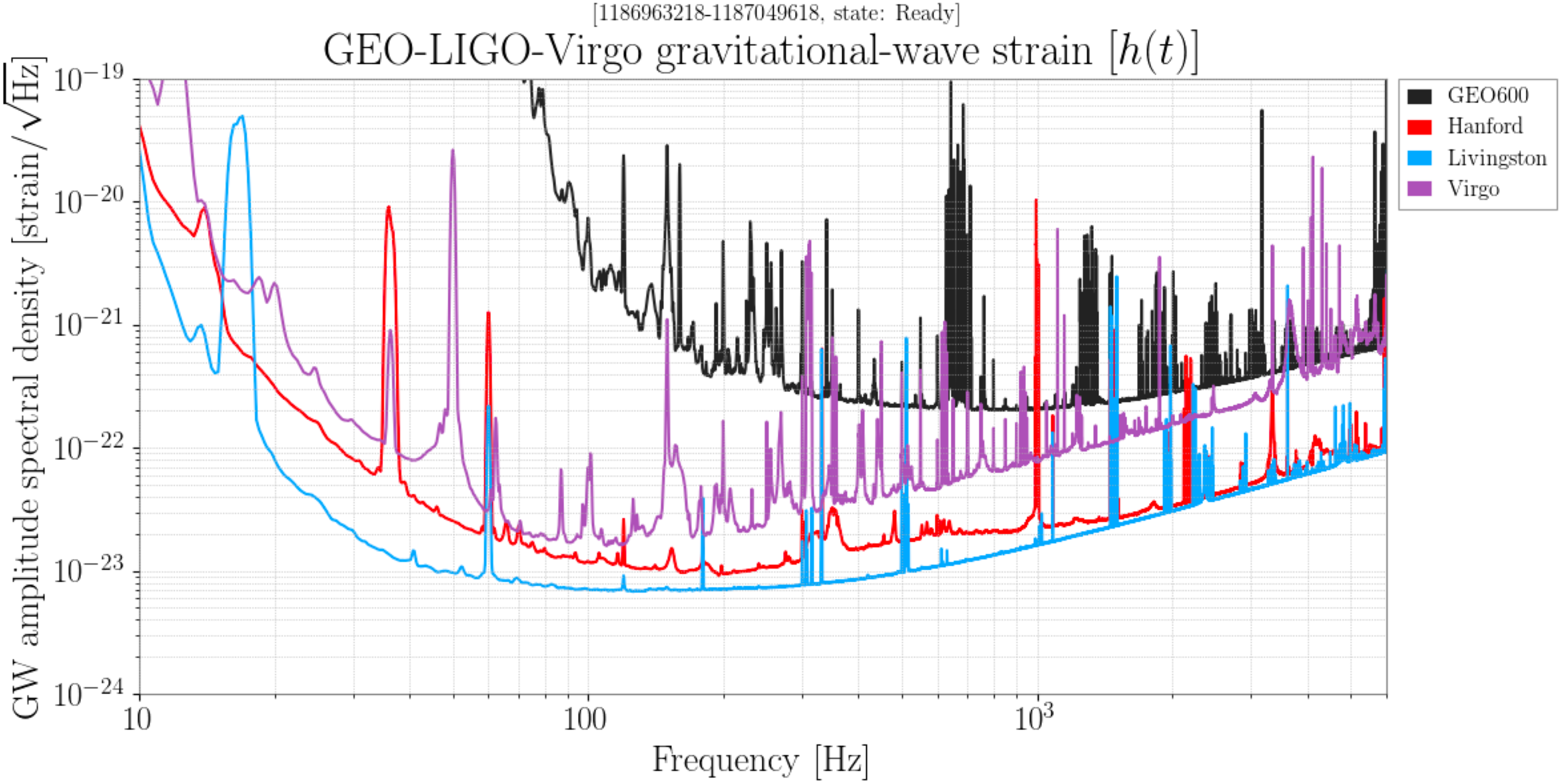
Value History

Acquire	Sample Rate	Data Type	DCU Id	Chan Num	Gain	Slope	Units	Offset	IFOid	Model	Modified By	Modified On
3	16	4	19	53584	1.0	1.0	undef	0	0	H1ASC	CDS	Feb. 23 2016 19:17 CST

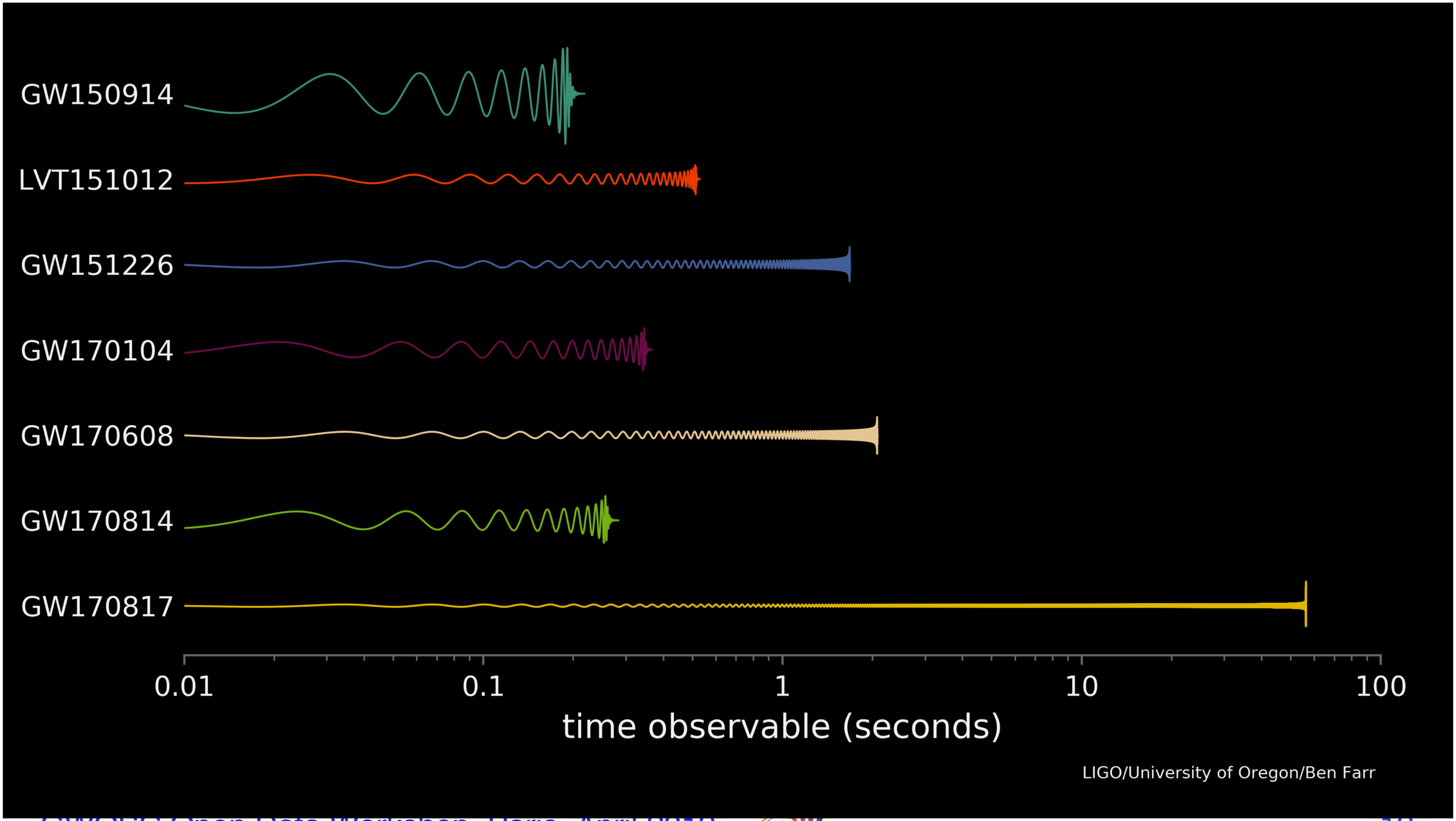
Advanced LIGO data channels

- Those other ~400K channels are digitized at sampling rates of 16 Hz, 256 Hz, 2048 Hz, … 16384 Hz. The strain channel is sampled at f_{samp} = 16384 Hz (2^{14} Hz).
- We use powers of two because we will study the frequency spectrum of the data channels, using digital fast Fourier transforms (DFFTs), which are faster to compute if the data samples are in groups of powers of 2.
- Data channels should have no appreciable frequency content above $f_{Nyquist} = f_{samp}/2$.
- An enormous effort goes into understanding all those other channels, using them to improve detector performance, and then boil them all down to a handful of "Data Quality" (DQ) channels, with 16 Hz sampling (discussed later).
- GWOSC releases the h(t) strain channel and these DQ channels; they are all that is needed by the LIGO-Virgo Collaborations, and GWOSC users, for astrophysical data analysis.

HLVG – four-detector network for GW170817!
"Strain noise amplitude spectral density"

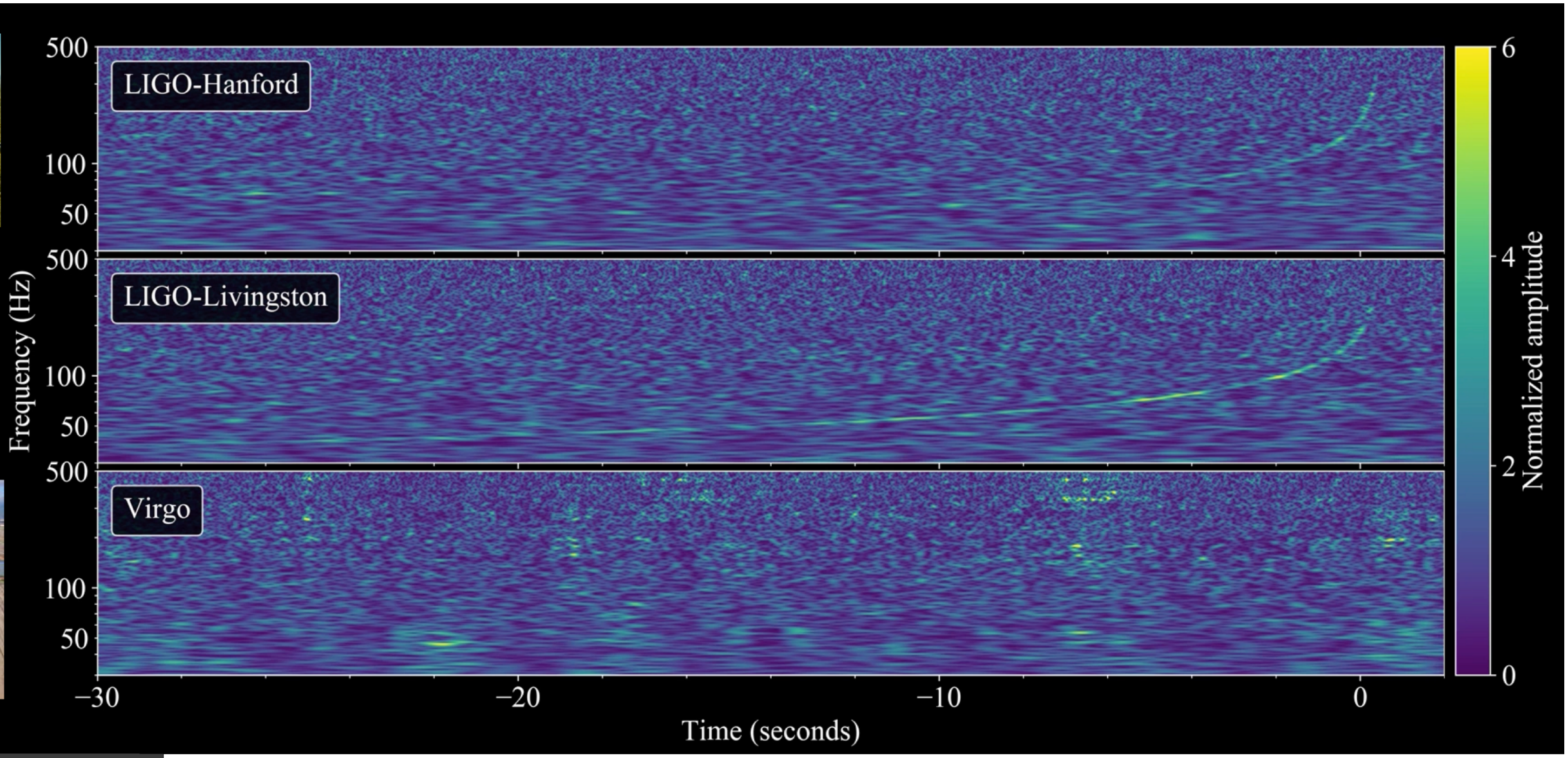


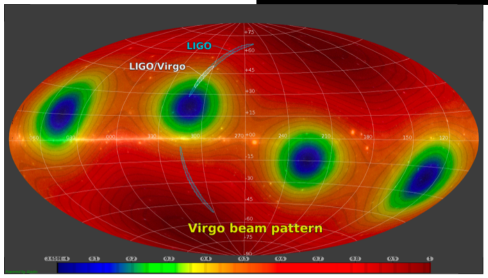
Observed signal durations (above ~30 Hz)



LIGO/University of Oregon/Ben Farr

Time-scales of GW signals from BNS mergers, BBH mergers, CCSNe, CWs, stochastic, …



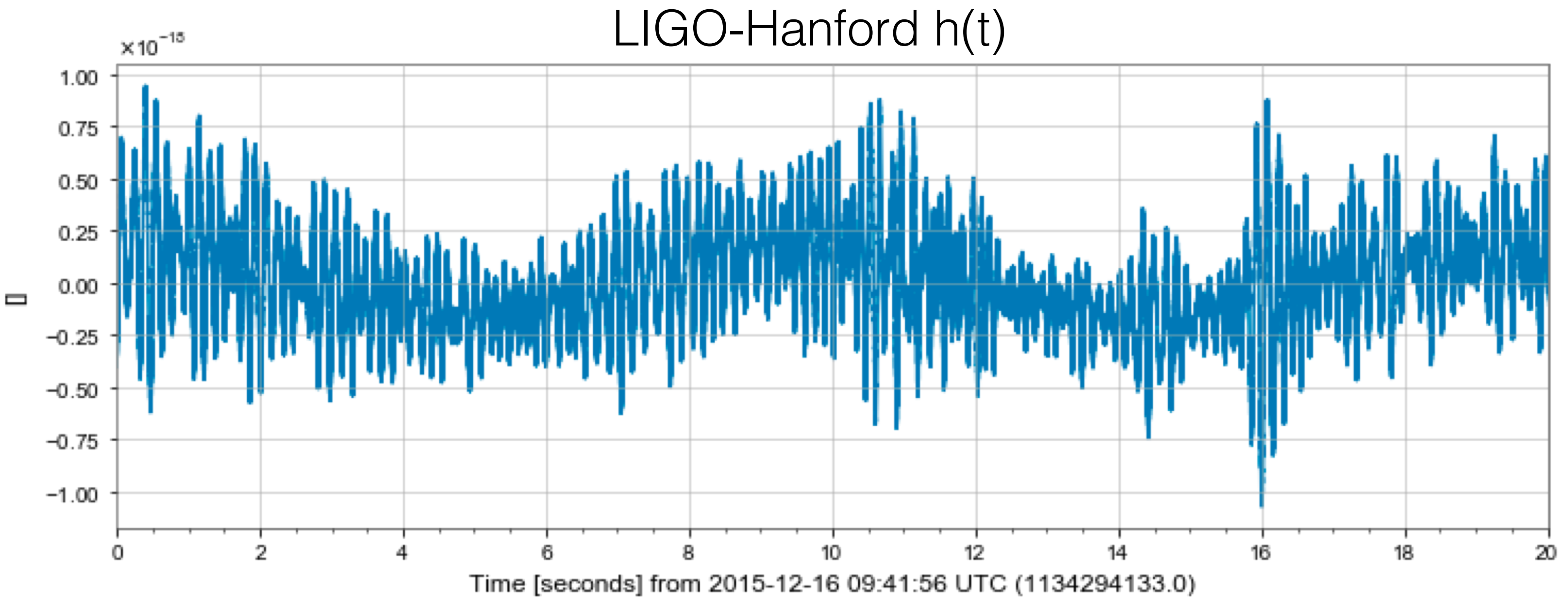


http://ligo.org/detections/GW170817.php

Finding the needle in the messy haystack

- CBC signals are typically < 1 second in-band ([20-3000] Hz). We found 10 of them in O1+O2 = around 6 months of coincident (H+L) ovserving.
  Needle in a haystack!
- It has been shown that, if a suitably precise model of the signal(s) of interest exist, matched filtering, using the model as a template, is "optimal" for identifying weak signals in Gaussian noise (*Extraction of Signals from Noise*: Wainstein and Zubakov, 1962).
- This is true even if unknown model parameters (masses & spins) mean we have not one template, but 100,000's or more. (CW searches: ~10^{13} templates!)
- The templates must be accurate over (potentially) very many cycles in-band!
- But aLIGO data are non-Gaussian and non-stationary – a big challenge!
- CBC detection pipelines used in LIGO & Virgo:
  `pycbc`, `gstlal`, `MBTA`, `SPIIR`, …

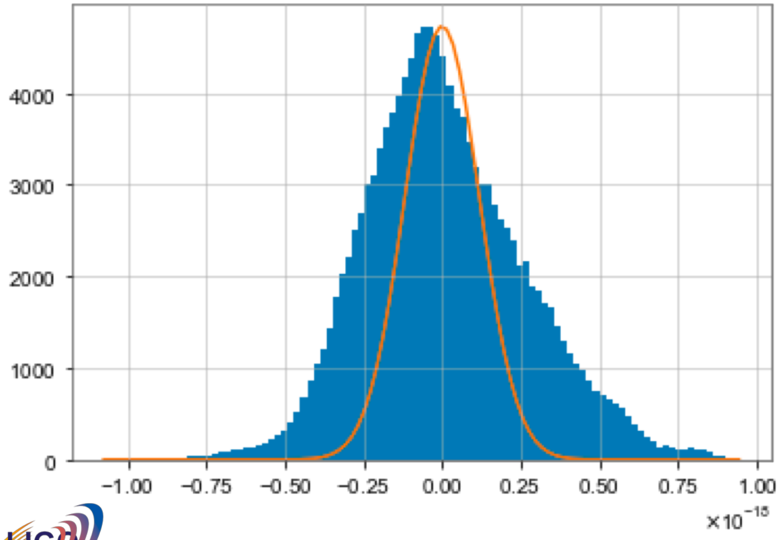
What do LIGO data look like?



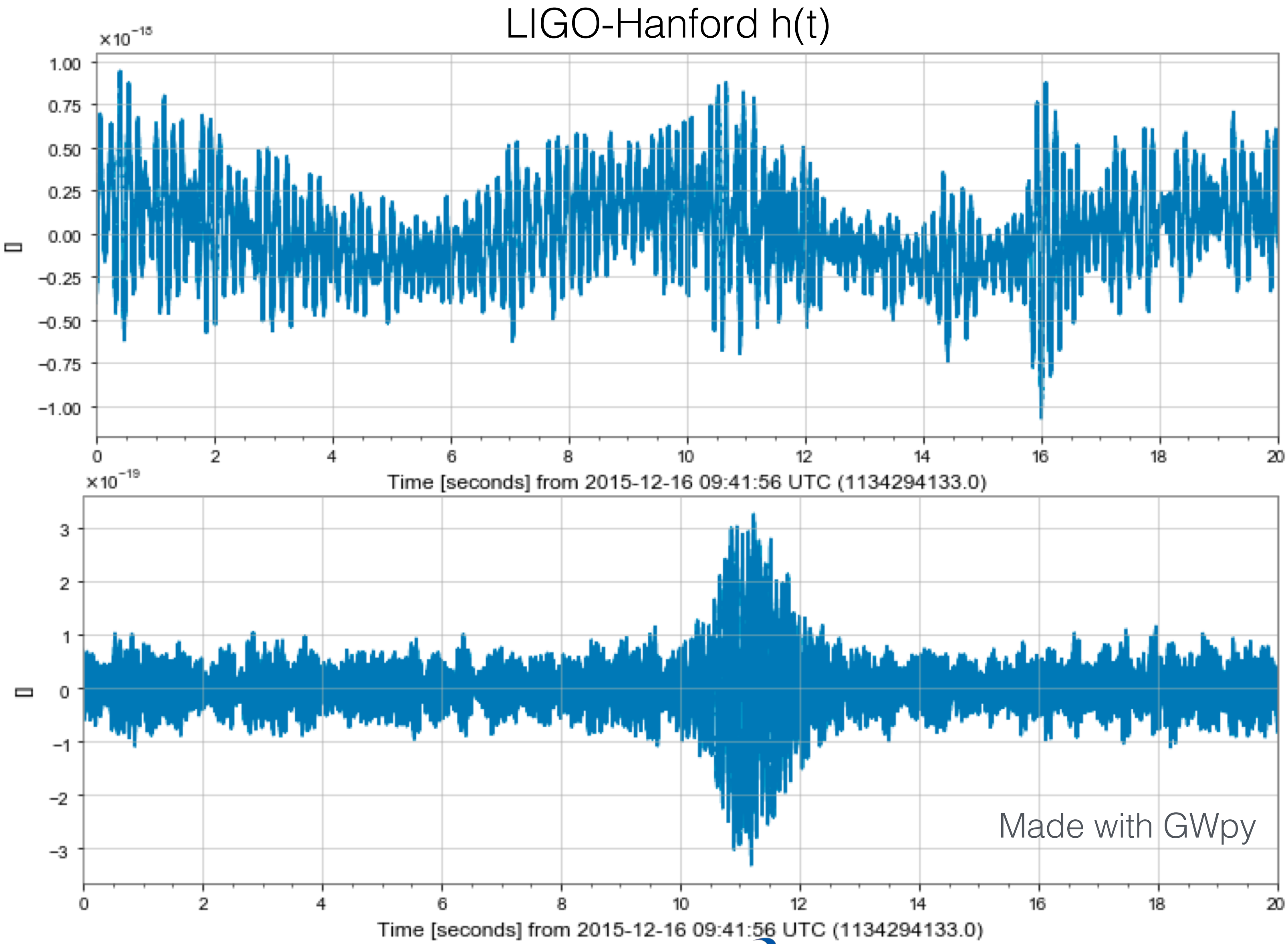
```
from gwpy.timeseries import TimeSeries
t0 = 1134294133
hdata = TimeSeries.fetch_open_data('H1', t0,t0+20)
hdata.plot()
```

The data are far from Gaussian!

- The noise is colored – much wider Gaussian for low and high frequencies.
- There are glitches (signal??) in there!



What do LIGO data look like, after bandpass?


LIGO-Hanford h(t)

Made with GWpy

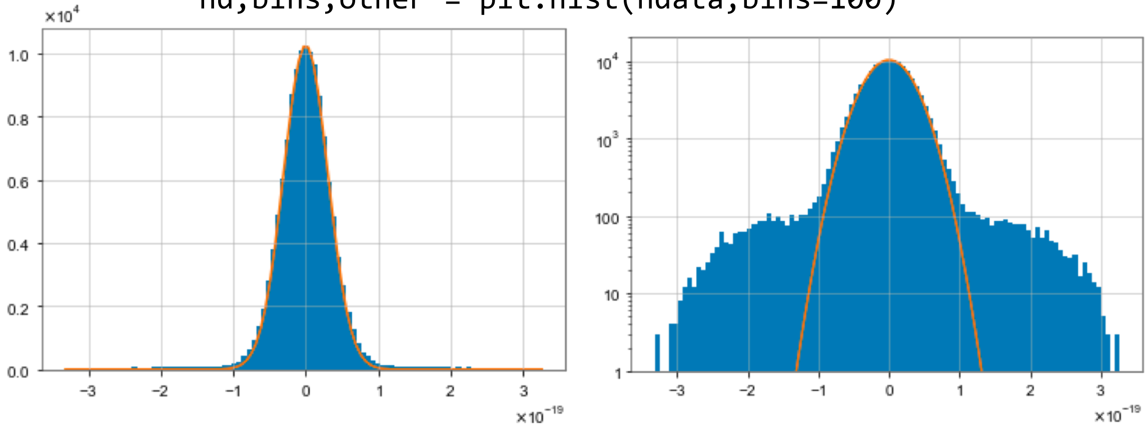
After bandpass, [10,1000] Hz, notches at 60*n:

- Example gwpy code for bandpassing:

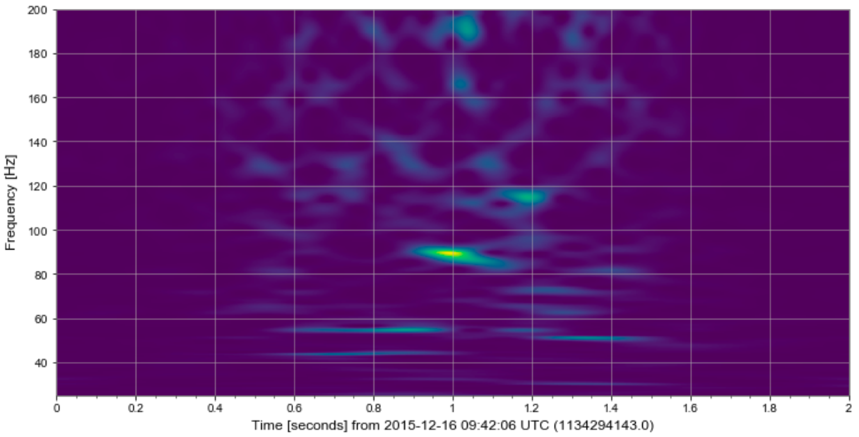
```
from gwpy.signal import filter_design
bp = filter_design.bandpass(10, 1000, data.sample_rate)
notches = [filter_design.notch(line, data.sample_rate) for
       line in (60, 120, 180)]
zpk = filter_design.concatenate_zpks(bp, *notches)
hfilt = hdata.filter(zpk, filtfilt=True)
```

- It’s looking more Gaussian…
- Ah, but there are long non-Gaussian tails, due to the glitch (signal?)
- All plots made by gwpy, pretty easy!

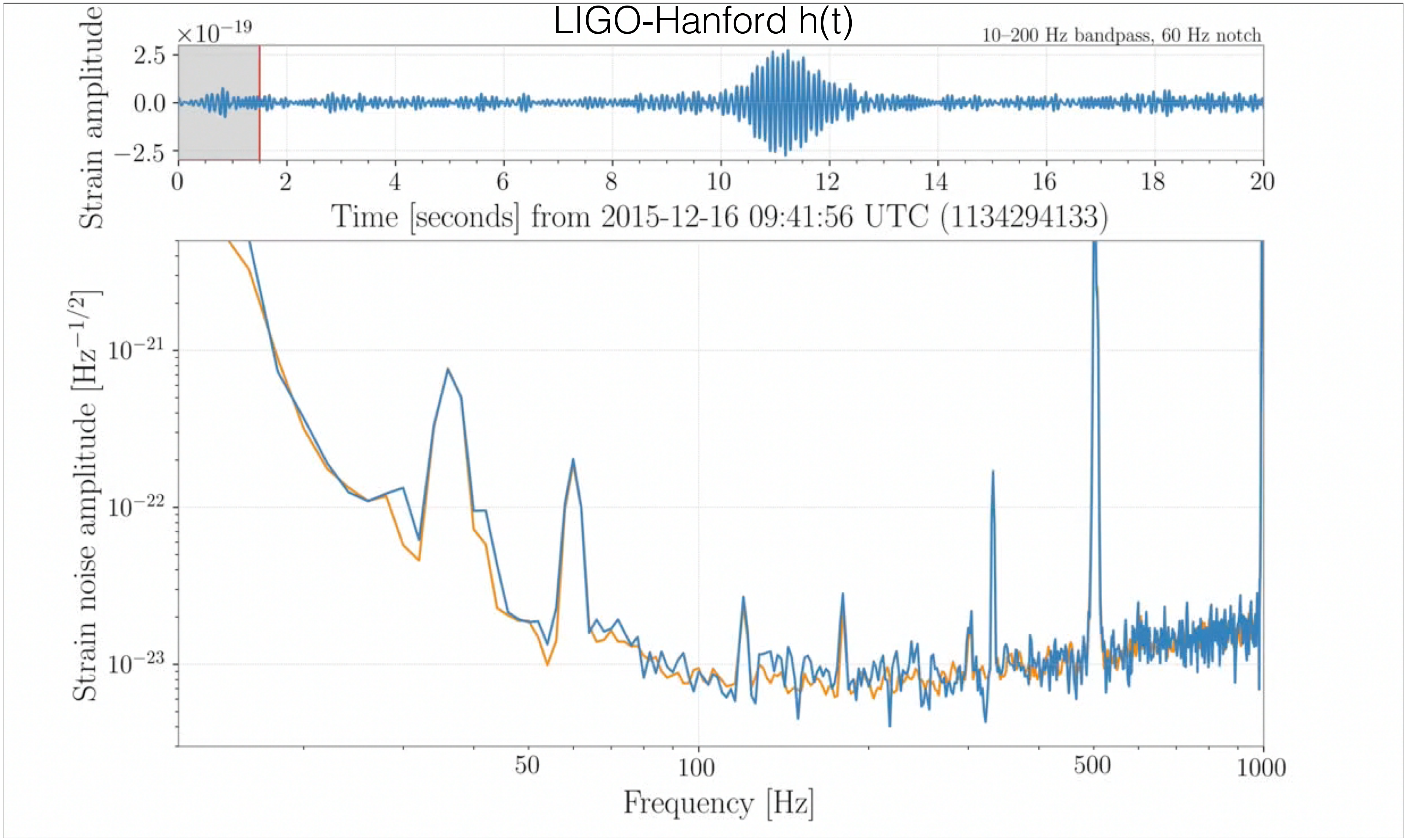
`hd,bins,other = plt.hist(hdata,bins=100)`



`hdata.q_transform(frange=(25,200)).plot()`



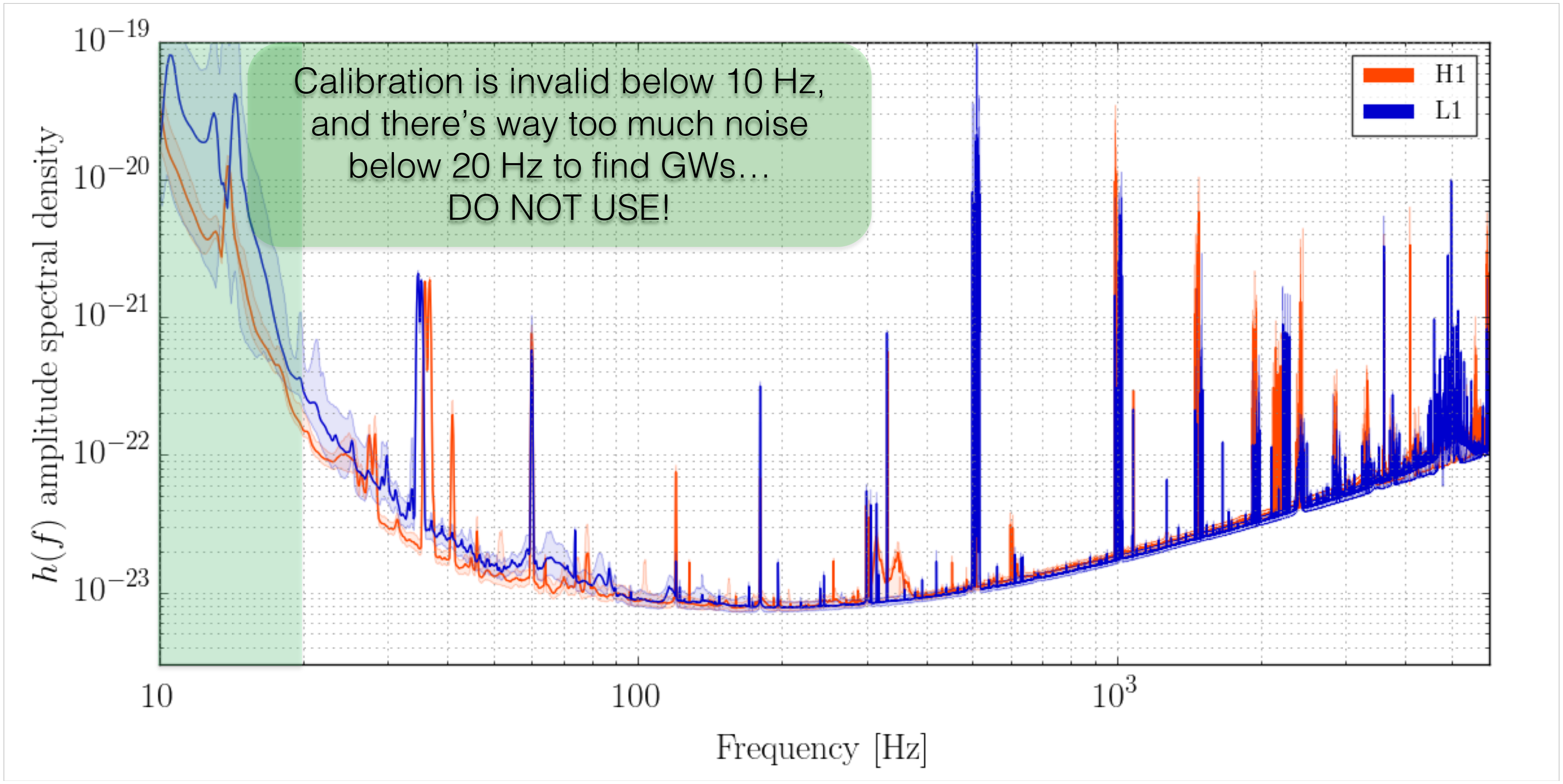
LIGO data in the frequency domain



Made with GWpy by Duncan Macleod. Code: https://git.io/gwpy-ligo-scattering-animation

0.5 second FFT; 5 averages covering 1.5 seconds; 50% overlap

Calibrated Strain noise – in the frequency domain

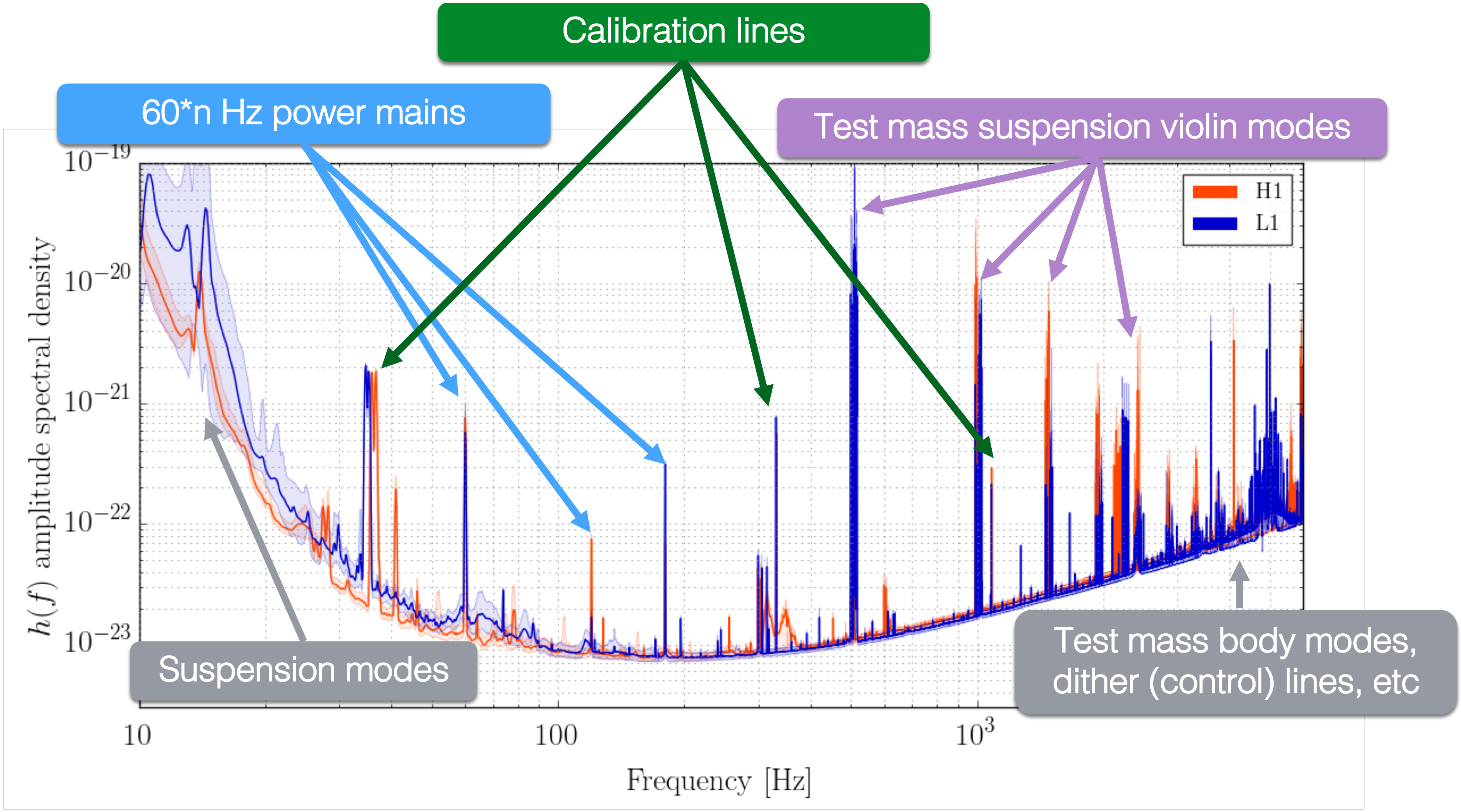


Source: https://losc.ligo.org/events/GW150914/

computed within ~5 s and broadcast to computing clusters

received by analysis pipelines to rapidly search for signals

Calibrated Strain noise spectral lines


Calibration lines

60*n Hz power mains

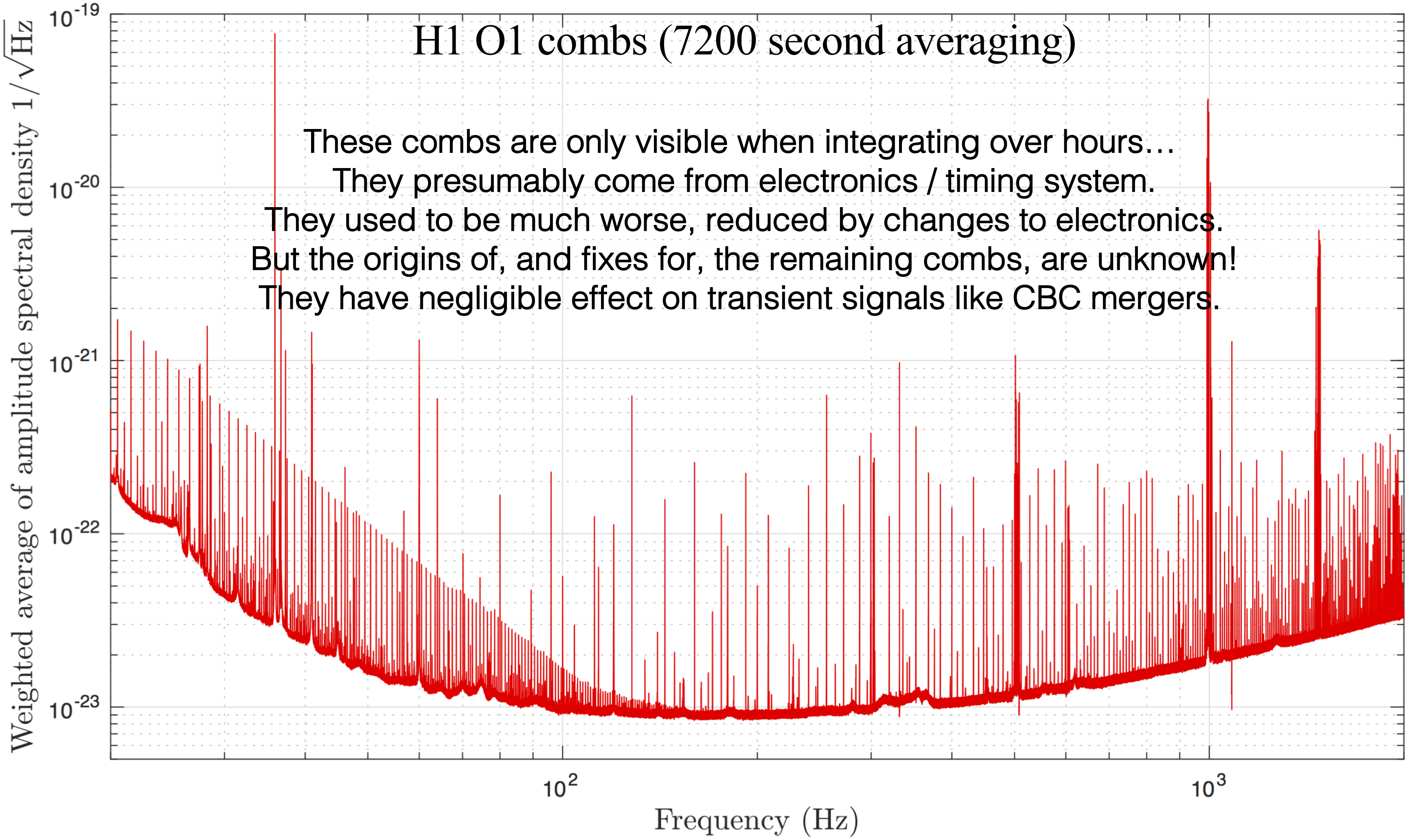
Test mass suspension violin modes

Suspension modes

Test mass body modes, dither (control) lines, etc

Source: https://losc.ligo.org/events/GW150914/

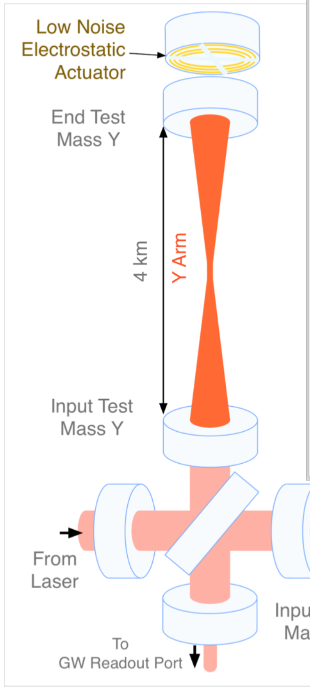
Combs of lines in LIGO data

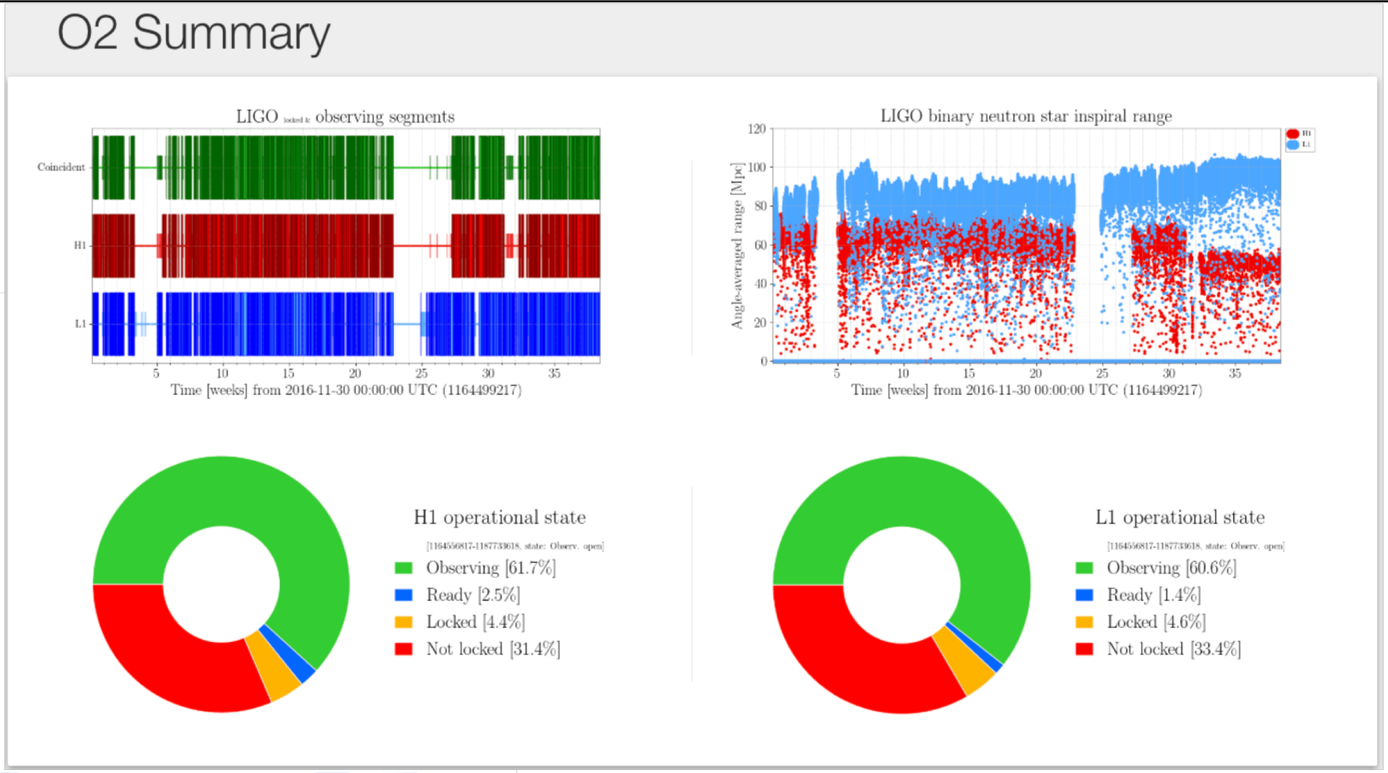


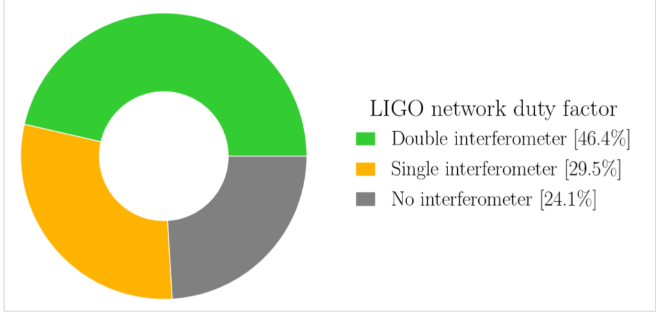
These combs are only visible when integrating over hours…
They presumably come from electronics / timing system.
They used to be much worse, reduced by changes to electronics.
But the origins of, and fixes for, the remaining combs, are unknown!
They have negligible effect on transient signals like CBC mergers.

O1 and O2 noise lines paper: Covas et al. (2017) arXiv 1801.07204
Instrumental lines catalog for LIGO-Hanford and LIGO Livingston: losc.ligo.org/o1speclines

Detectors' duty cycle, coincidence, non-stationarity



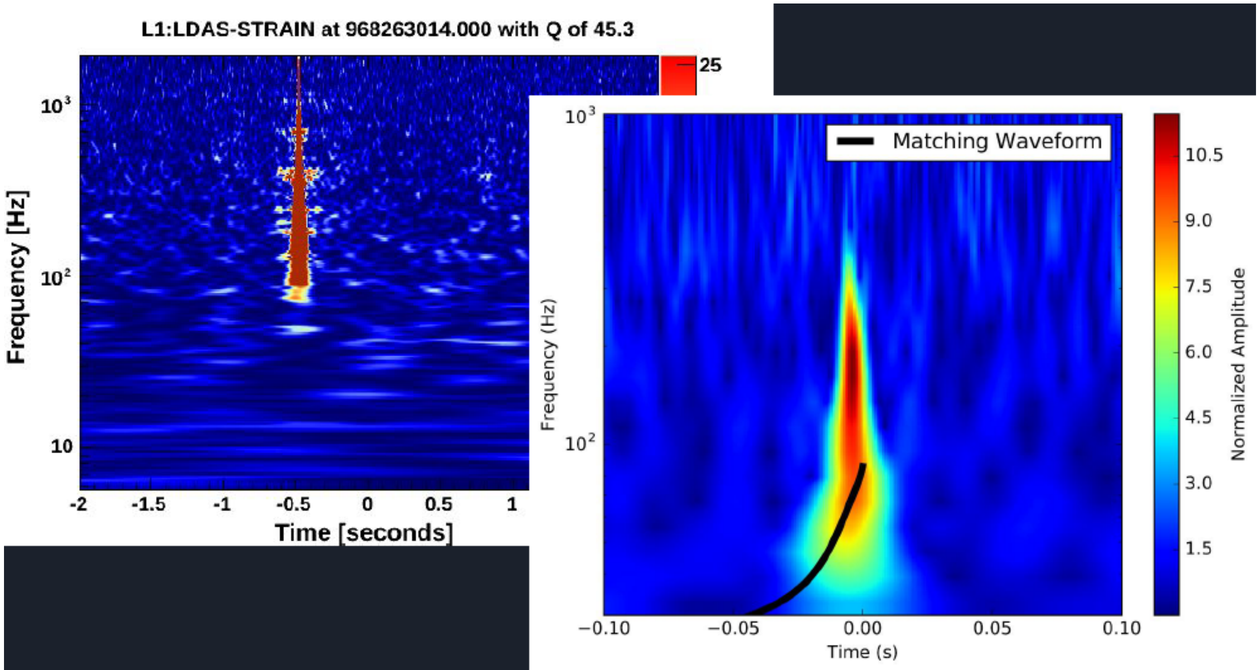

O2 Summary



Glitches in LIGO data

- Template-based matched filtering works “optimally” to find weak signals in stationary, Gaussian data…
- But real LIGO data is NOT exactly Gaussian or stationary ☹
- Glitches can occur due to a variety of different influences, such as:

  environmental disturbances; RFI; scattered light; control system instabilities; …
- Glitch sources that have been identified, have been fixed / eliminated / reduced.
- The remaining glitches have been more resistant to diagnosis / fixing; constant effort continues!
- Transient non-Gaussian noise fluctuations (glitches) can “ring up” a template filter!



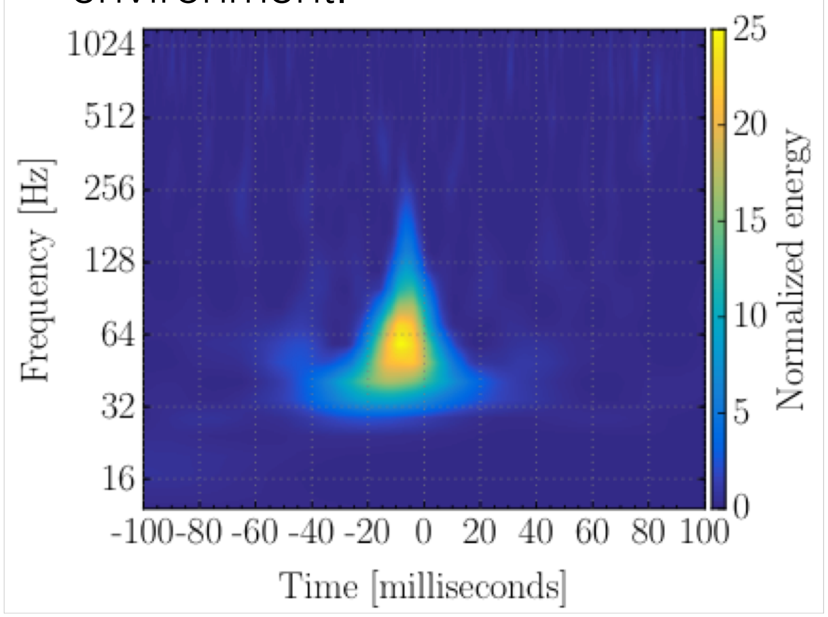
[redacted]

[redacted]

LIGO data are non-stationary!

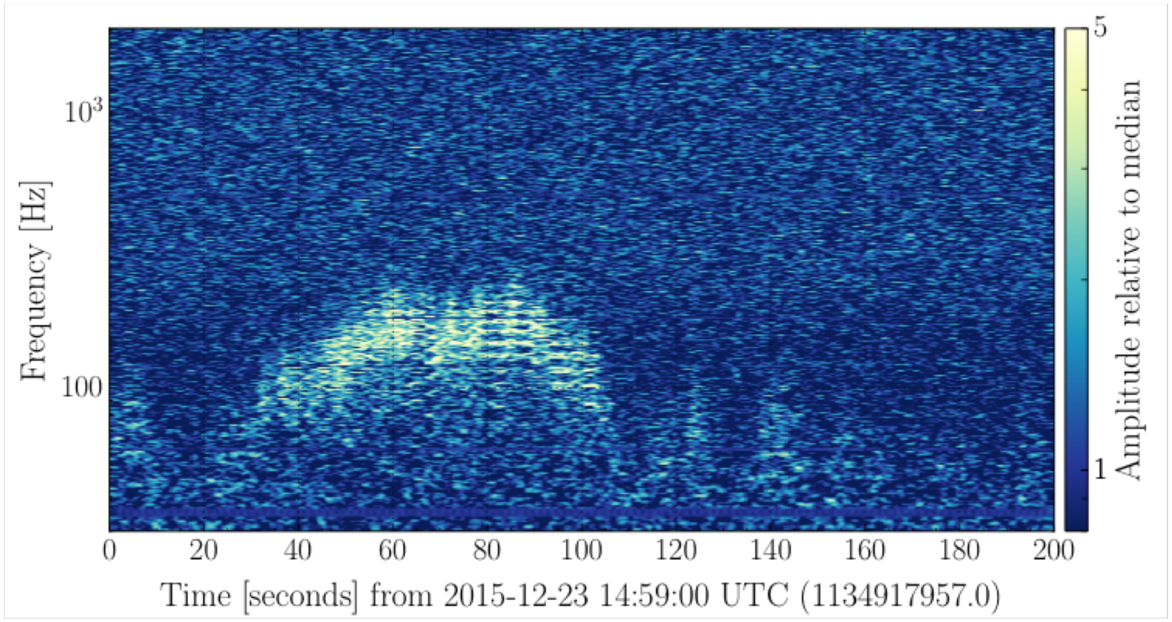
Blip glitches

- The biggest contributor to the transient GW search backgrounds
- Seen in both LIGO detectors (non-coincident) ~1x/hour
- No known correlation with instrument behavior or environment.



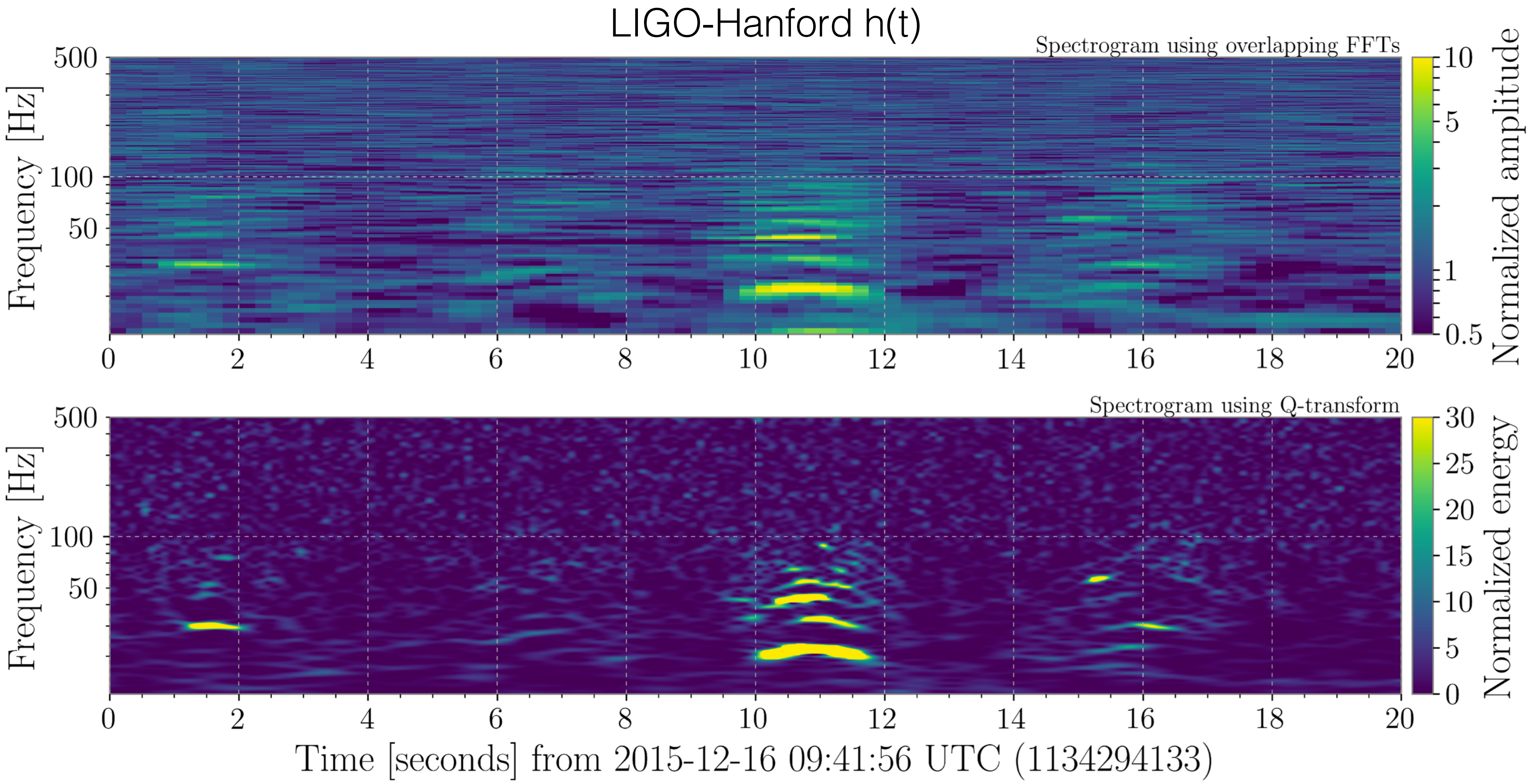
60-200 Hz non-stationary noise

- Pollutes LIGO-Livingston data in a critical frequency range (~50-500Hz)
- Longer duration (10s or 100s of seconds)
- Major contributor to CBC and burst backgrounds



arXiv 1710.02185

Visualizing glitches:
Time-frequency spectrograms

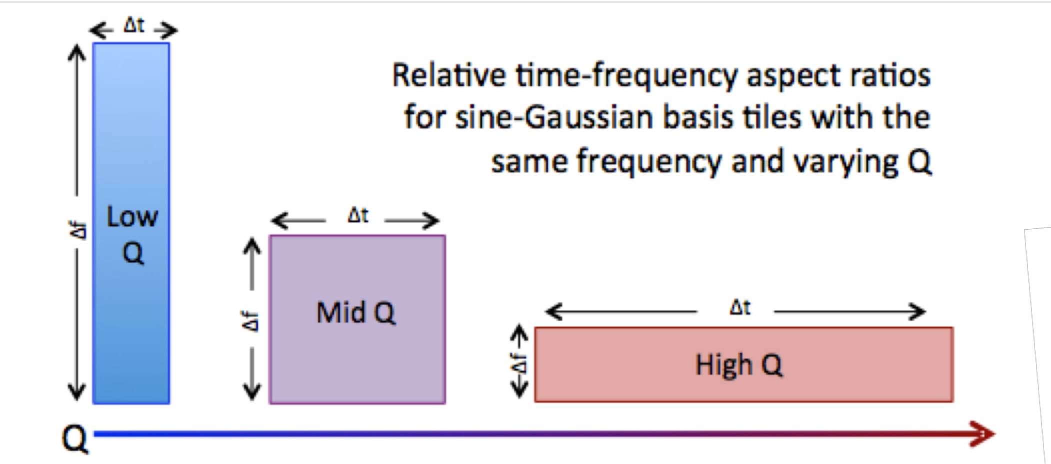


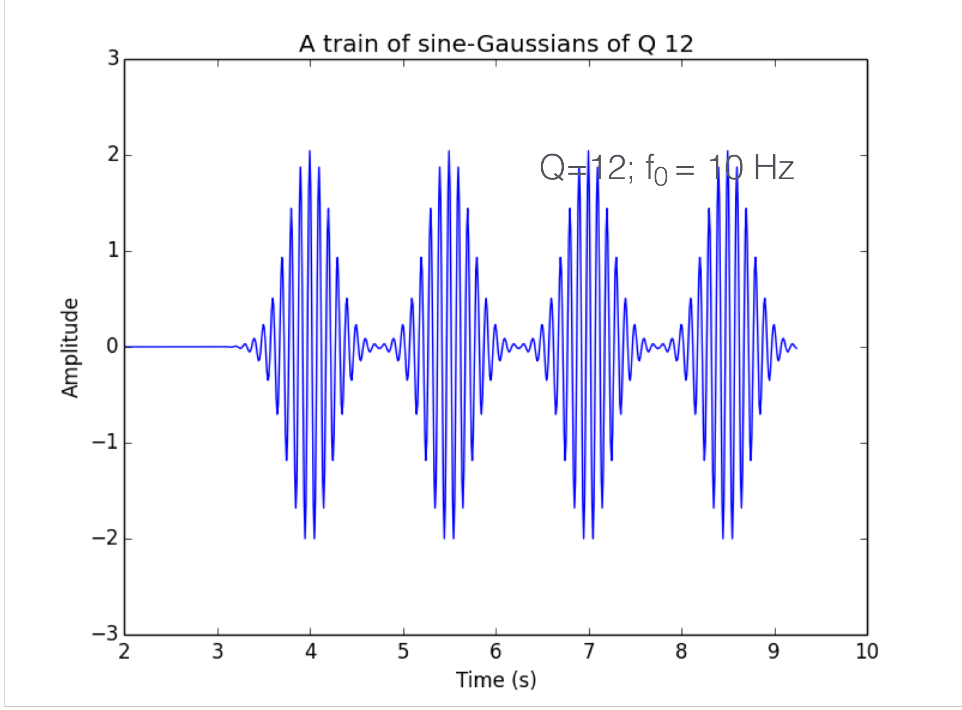
Made with GWpy by Duncan Macleod

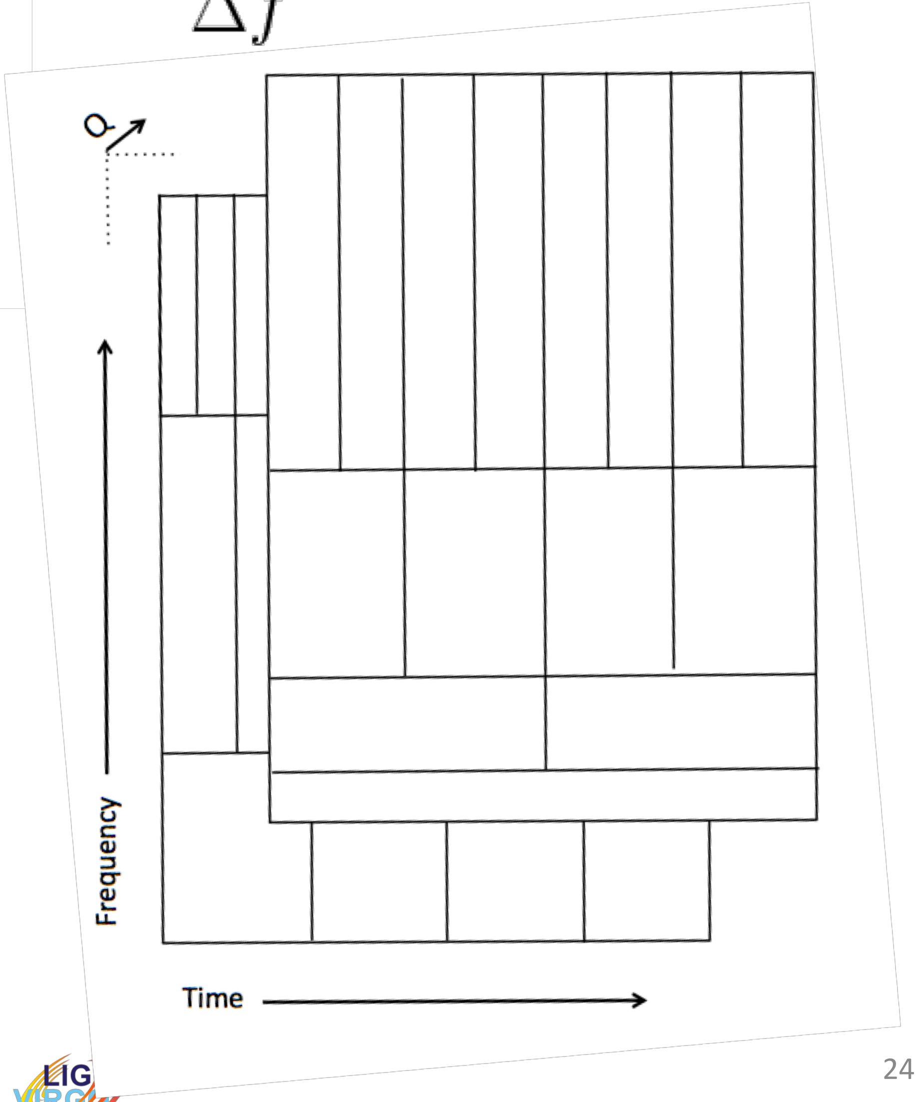
The Q transform

S. Chatterji et al. CQG (2010)
Images: McIver

$$Q = \frac{f_0}{\Delta f}$$


Relative time-frequency aspect ratios for sine-Gaussian basis tiles with the same frequency and varying Q

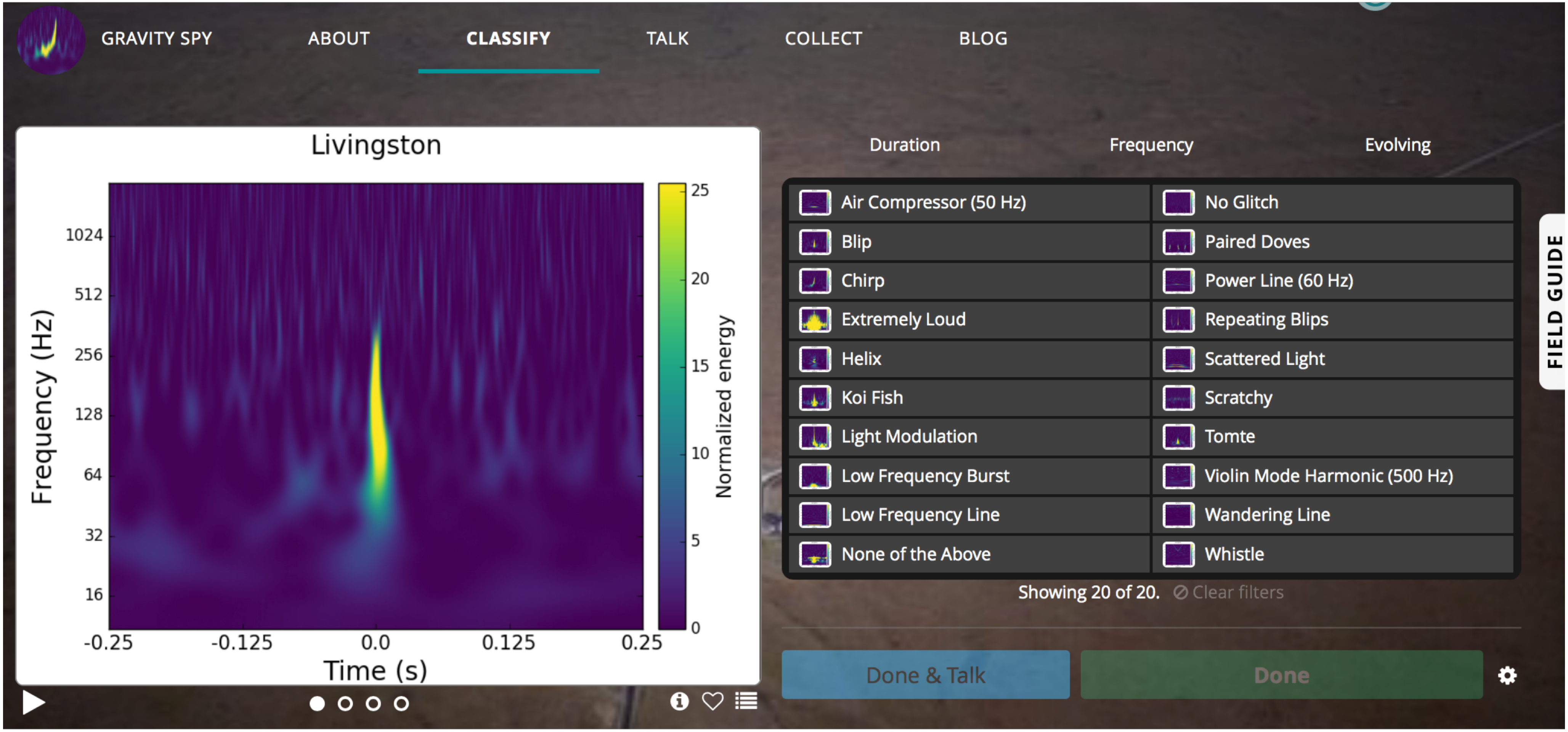

A train of sine-Gaussians of Q 12



A menagerie of common glitch types

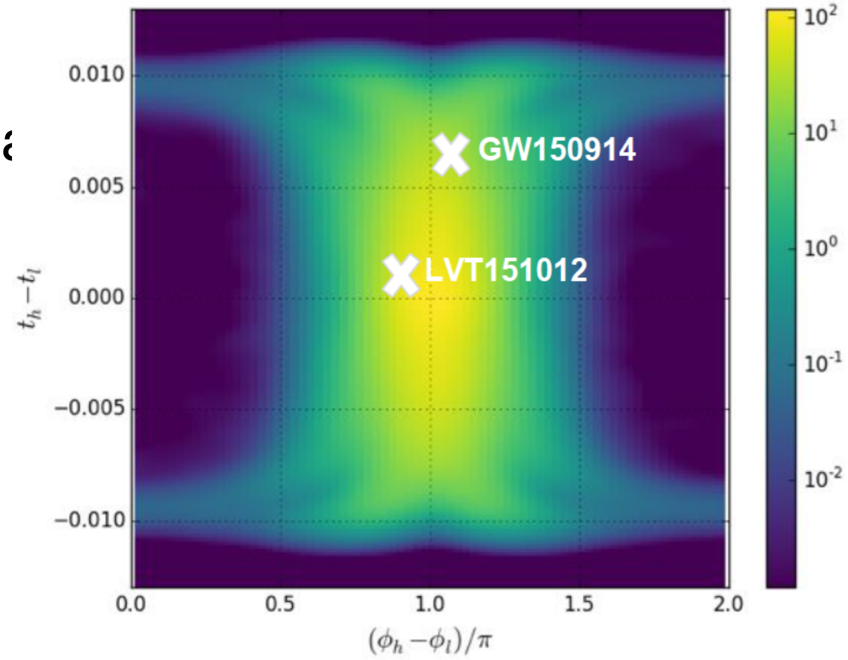
gravityspy.org Zevin et al, 2017, CQG

Citizen science: help us identify & classify glitch morphologies, train machine-learning algorithms to recognize them



How to mitigate glitches

- Check auxiliary monitoring channels for environmental and instrumental causes, and veto the effected data during such times (DQ vetoes).
- Require coincidence between multiple detectors, at observatories separated by thousands of km:
  - Δt - within light travel time between the observatories (±10 ms between LHO and LLO). Add ~5 ms for timing errors in match between emplate and signal in noisy data
  - ΔA, Δϕ - relative amplitude and phase of observed signal at 2 detectors consistent with astrophysical source
  - Δm - Signal morphology (governed by masses) is consistent between detectors – eg, require exact same template.



How are data quality segments defined?

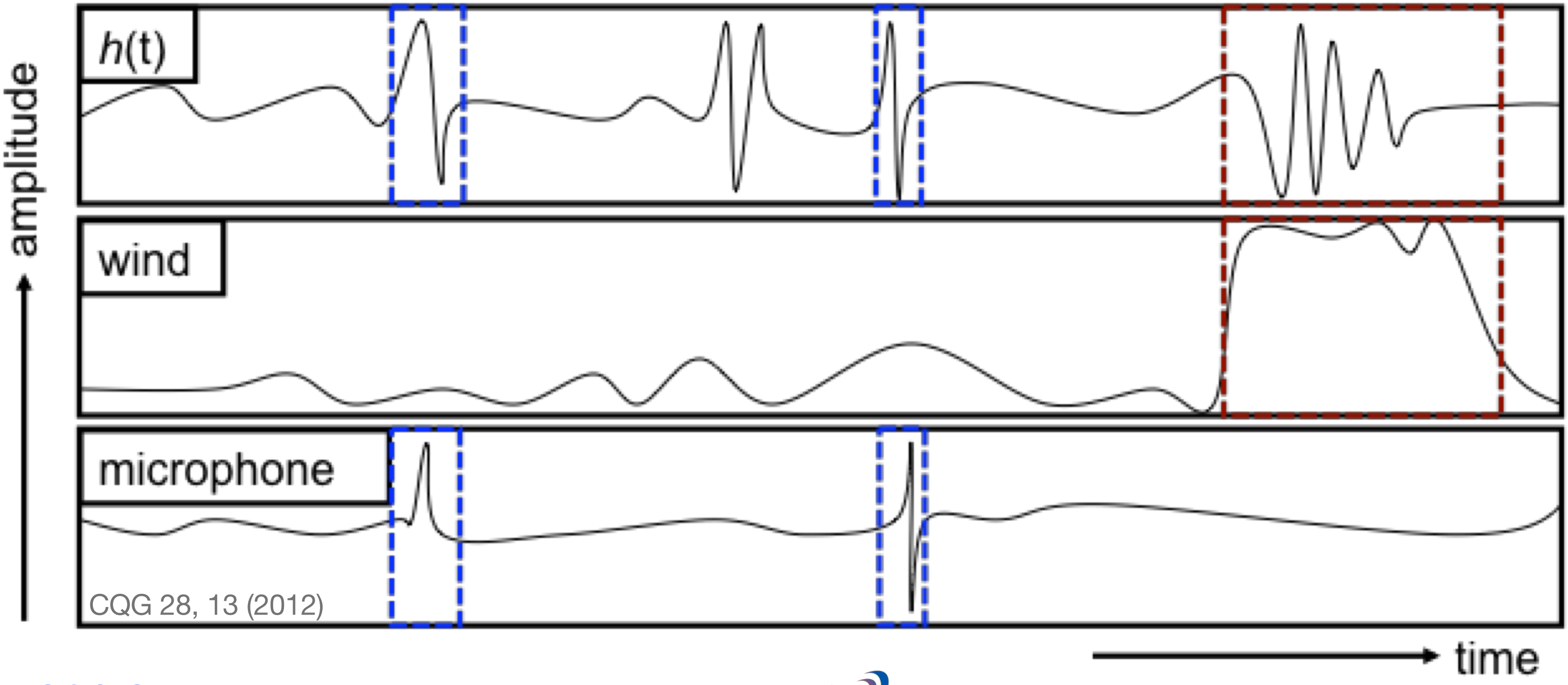
Data quality vetoes require an ***auxiliary witness***

- That auxiliary witness is required to be ***safe***; to (demonstrably) not be sensitive to changes in spacetime strain
- Veto segments were defined based on noise sources *known* to couple to h(t)
- Veto categories were determined for each type of search independently depending on noise contributors to that search's background
    - There are differences between CAT2 and CAT3 definitions between the burst and CBC searches

Auxiliary channels

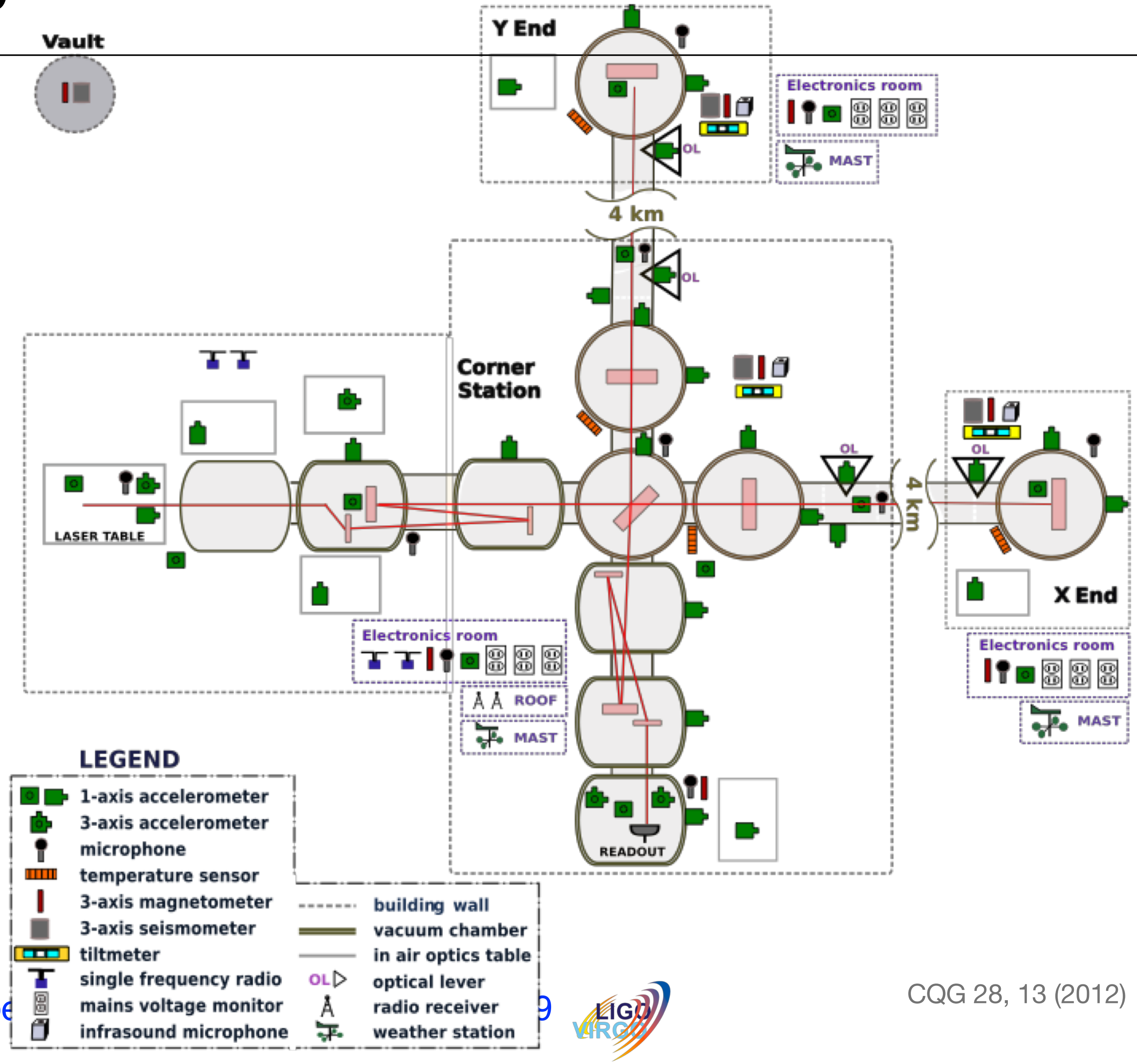
We record **over 200,000 channels per detector** that monitor the environment and detector behavior.

We can use these to **help trace the instrumental causes of glitches** that pollute the search backgrounds.



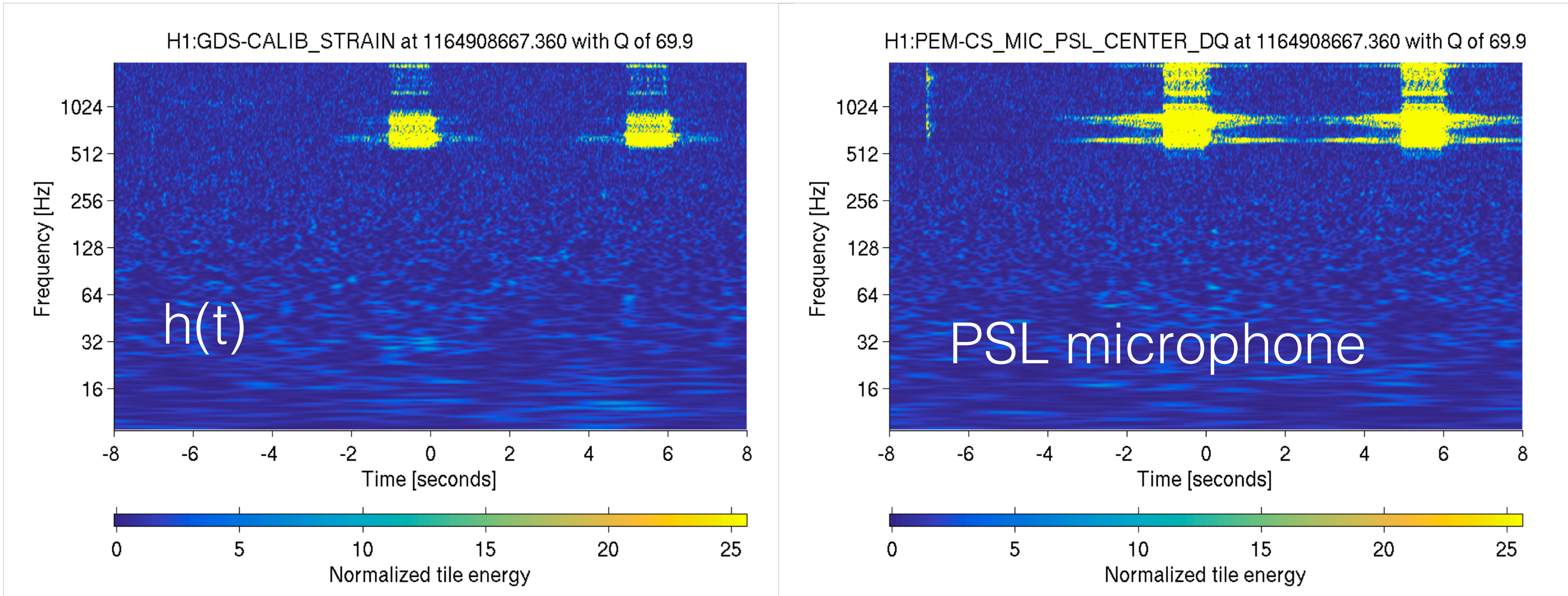
CQG 28, 13 (2012)

Physical environment channels



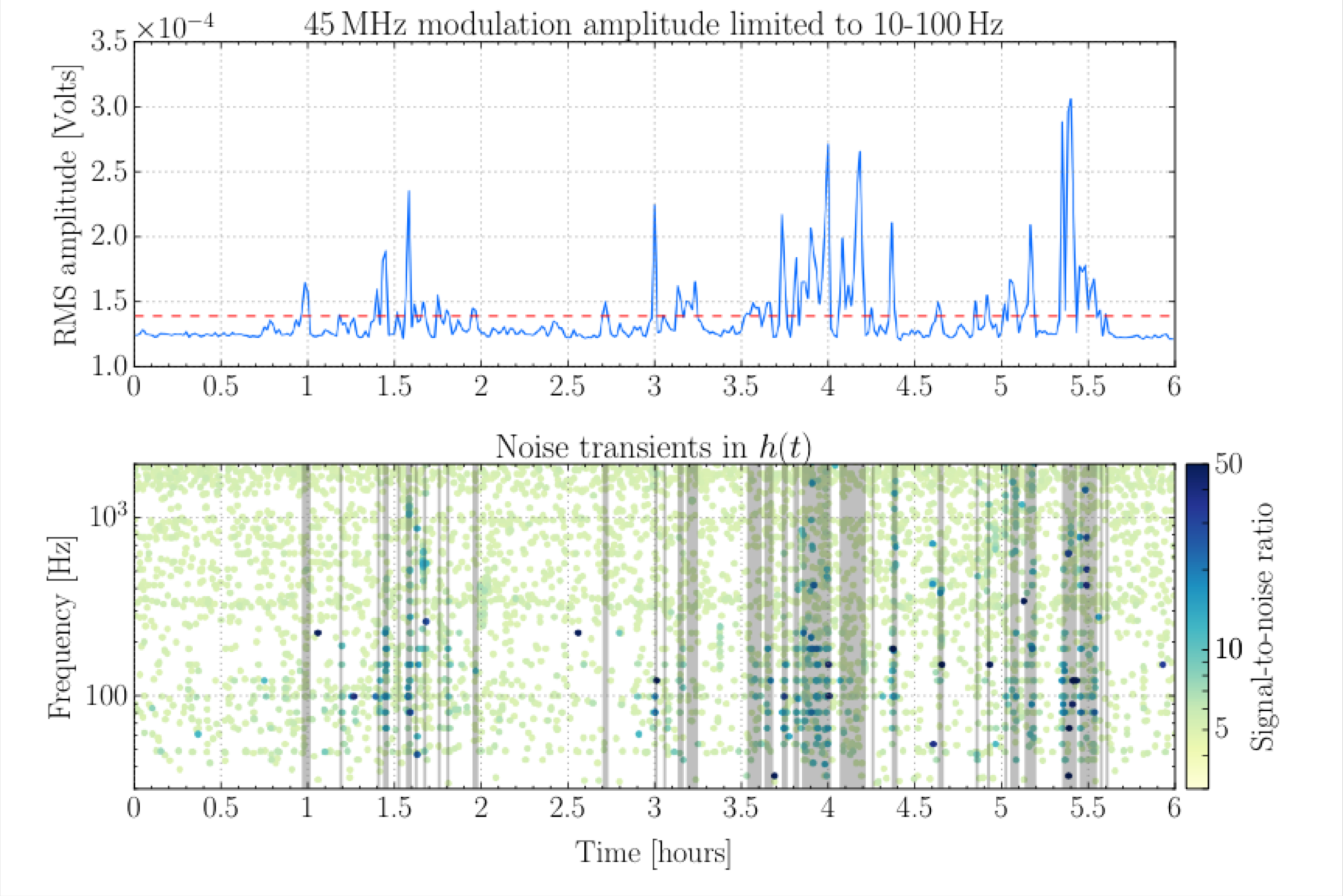
CQG 28, 13 (2012)

Laser glitches - h(t) vs. microphones


H1:GDS-CALIB_STRAIN at 1164908667.360 with Q of 69.9

H1:PEM-CS_MIC_PSL_CENTER_DQ at 1164908667.360 with Q of 69.9

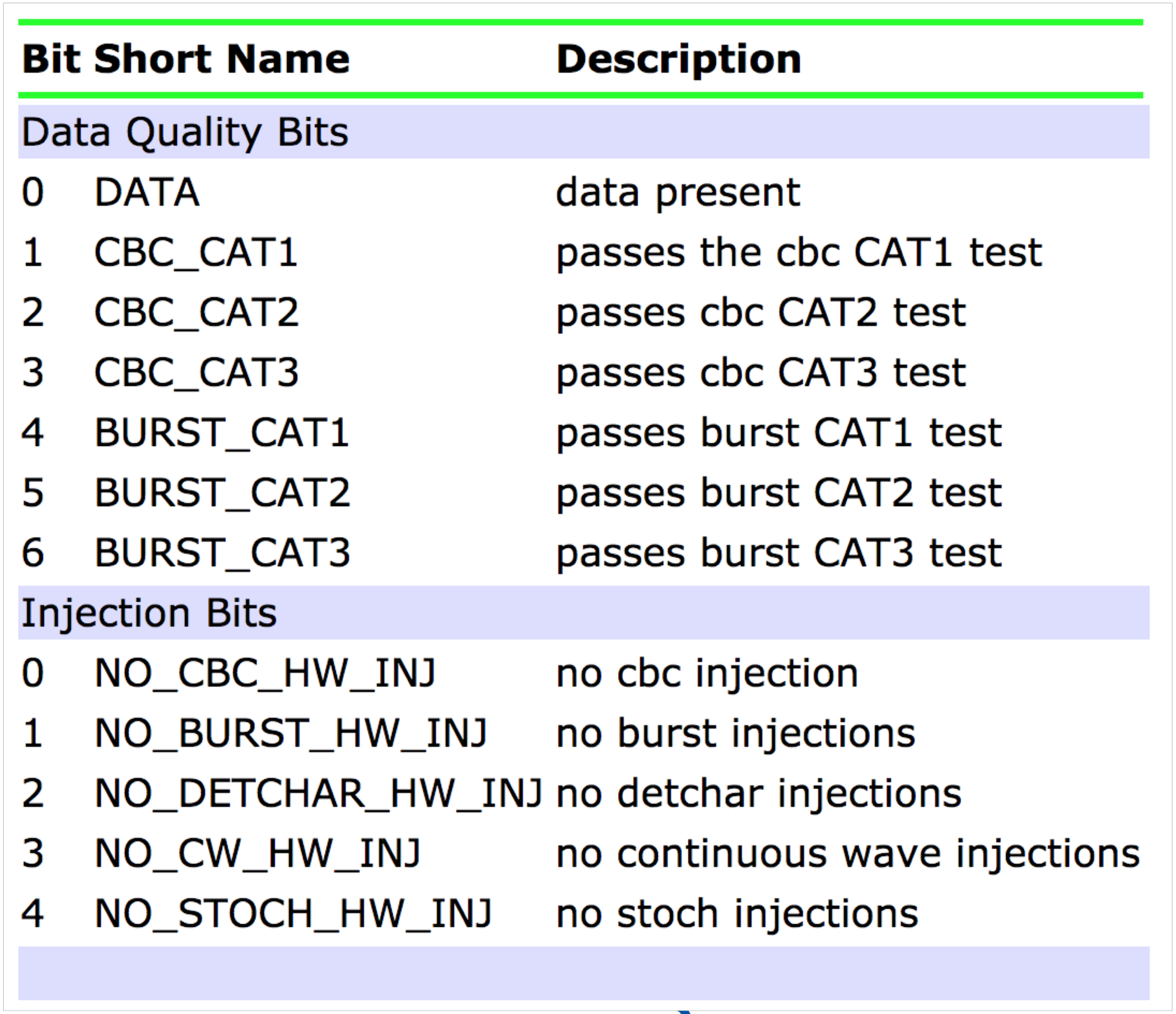
Example of a data quality veto in O1



LIGO-Virgo collaboration (2016) - arXiv 1602.03844

LIGO data quality information

Available with the h(t) strain data via the GWOSC



Bit	Short Name	Description
Data Quality Bits		
0	DATA	data present
1	CBC_CAT1	passes the cbc CAT1 test
2	CBC_CAT2	passes cbc CAT2 test
3	CBC_CAT3	passes cbc CAT3 test
4	BURST_CAT1	passes burst CAT1 test
5	BURST_CAT2	passes burst CAT2 test
6	BURST_CAT3	passes burst CAT3 test
Injection Bits		
0	NO_CBC_HW_INJ	no cbc injection
1	NO_BURST_HW_INJ	no burst injections
2	NO_DETCHAR_HW_INJ	no detchar injections
3	NO_CW_HW_INJ	no continuous wave injections
4	NO_STOCH_HW_INJ	no stoch injections

Data quality information

DATA (Data Available): Failing this level indicates that LIGO data are not publicly available because the instruments or data calibration were not operating in an acceptable condition.

CAT1 (Category 1): Failing a data quality check at this category indicates **a critical issue with a key detector component not operating in its nominal configuration**.

- These times are identical for each data analysis group.
- *Times that fail CAT1 flags are not available as LIGO open data.*

CAT2 (Category 2): Failing a data quality check at this category indicates times when there is a **known, understood physical coupling to the gravitational wave channel**. For example, high seismic activity.

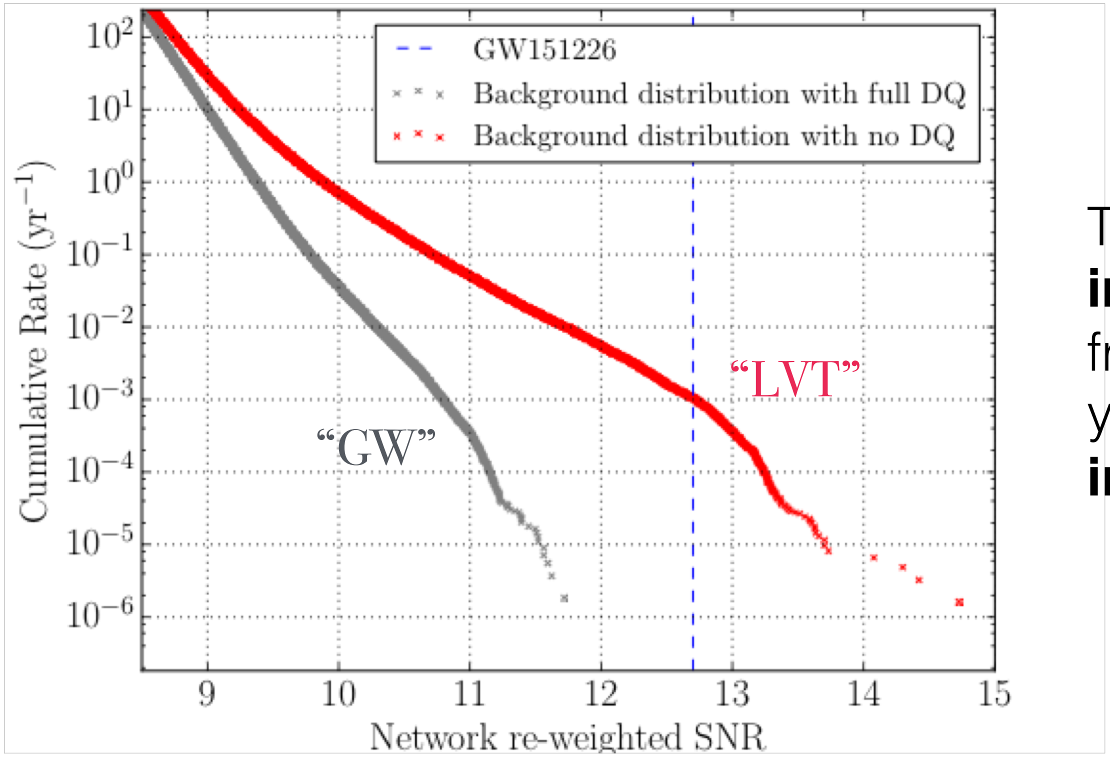
CAT3 (Category 3): Failing a data quality check at this category indicates times when there is **statistical coupling to the gravitational wave channel** which is not fully understood.

Data quality levels are defined in a cumulative way: a time which fails a given category automatically fails all higher categories.

Data quality categories are defined independently for different analysis groups: if something fails at CAT2_BURST, it could pass CAT2_CBC.

The impact of data quality vetoes

GW151226 analysis



LIGO-Virgo collaboration (2017) - arXiv 1710.02185

The false alarm rate of GW151226 **improves by a factor of 567**, from 1 in 320 years to 1 in 183000 years, **with detector data quality information!**

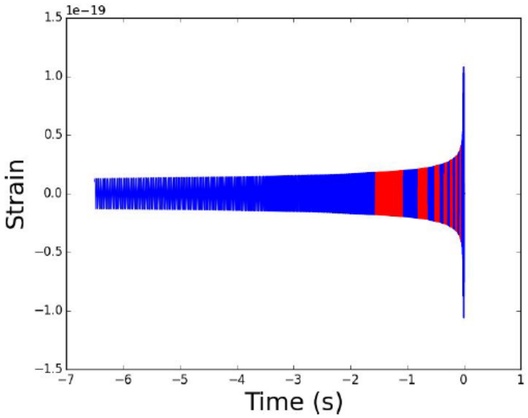
More glitch rejection tools:

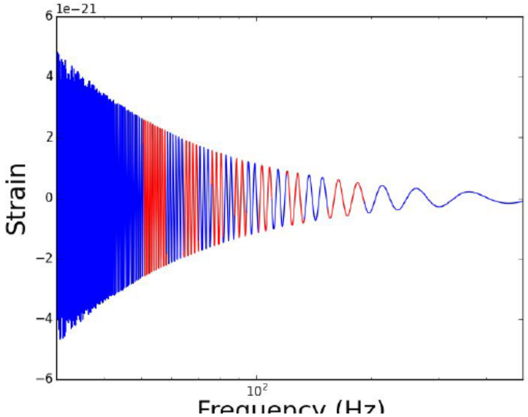
Require consistency between signal model and observed data in various dimensions

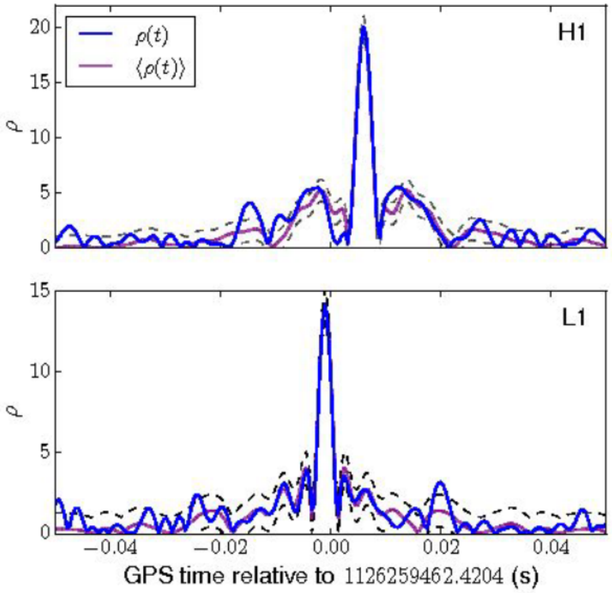
- SNR contribution by frequency band – time-frequency “Allen” χ^2
  divide the template into frequency bands of equal expected power.

- Template autocorrelation function χ^2
  Check for expected fall-off of SNR as template is shifted away from signal

-

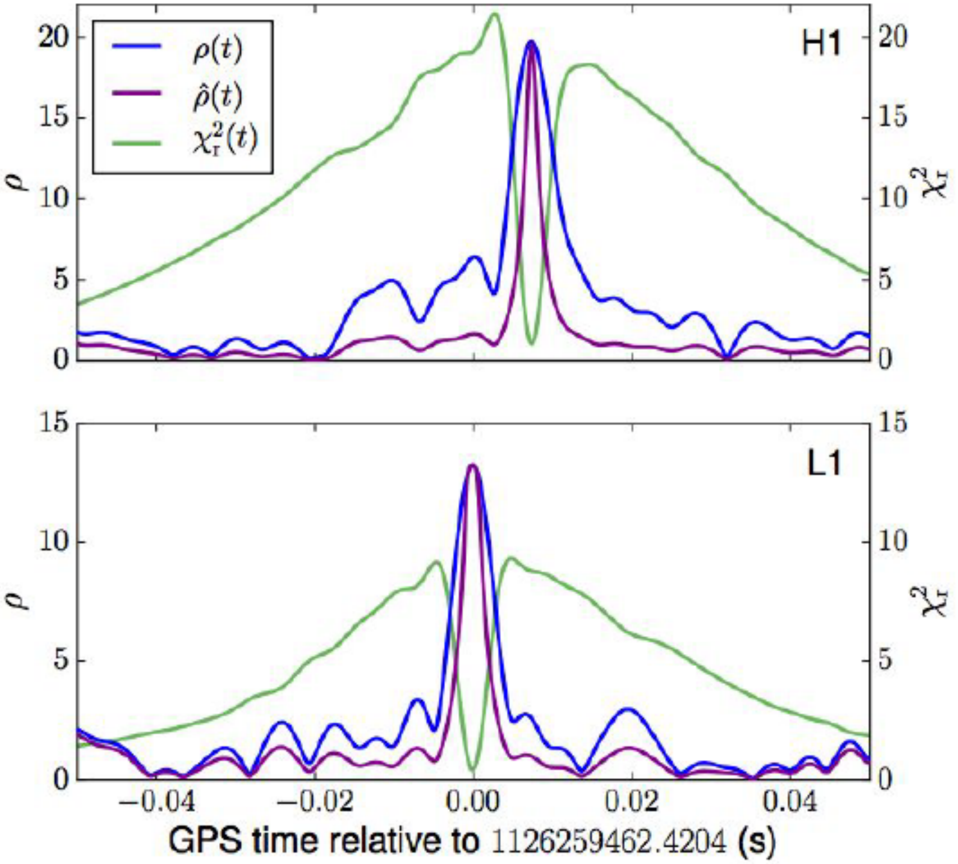




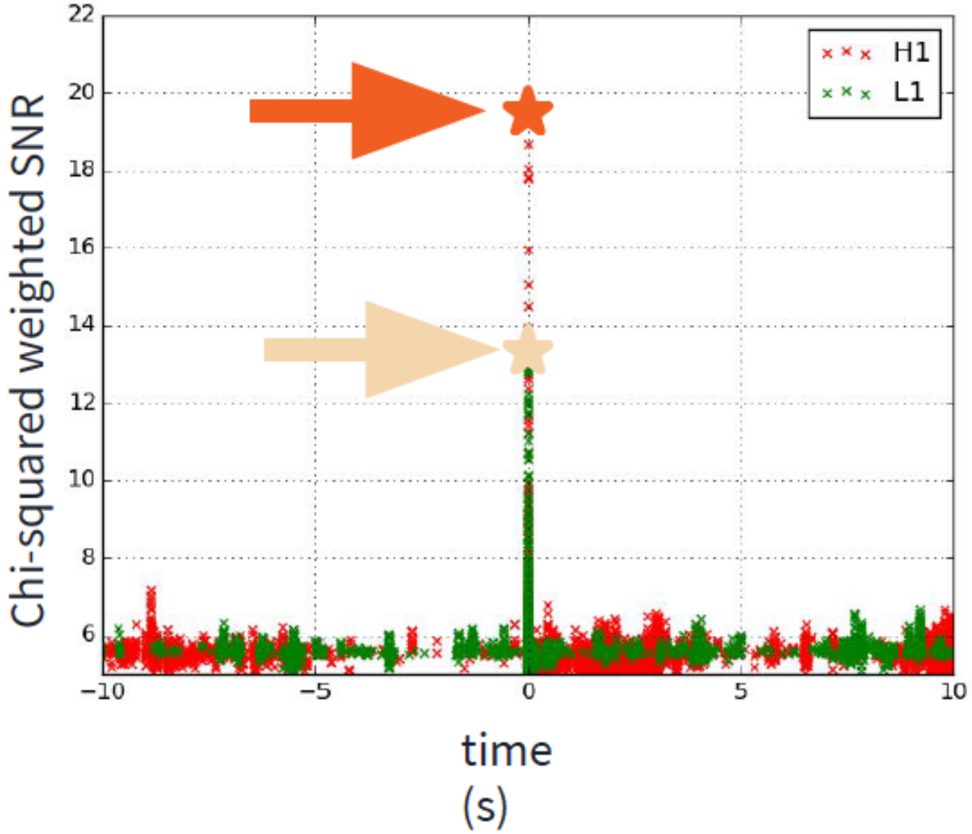


$$\chi^2 \propto \sum(\rho_l - \rho/N_{bins})^2$$

GW150914 SNR, χ^2, ρ_{eff}

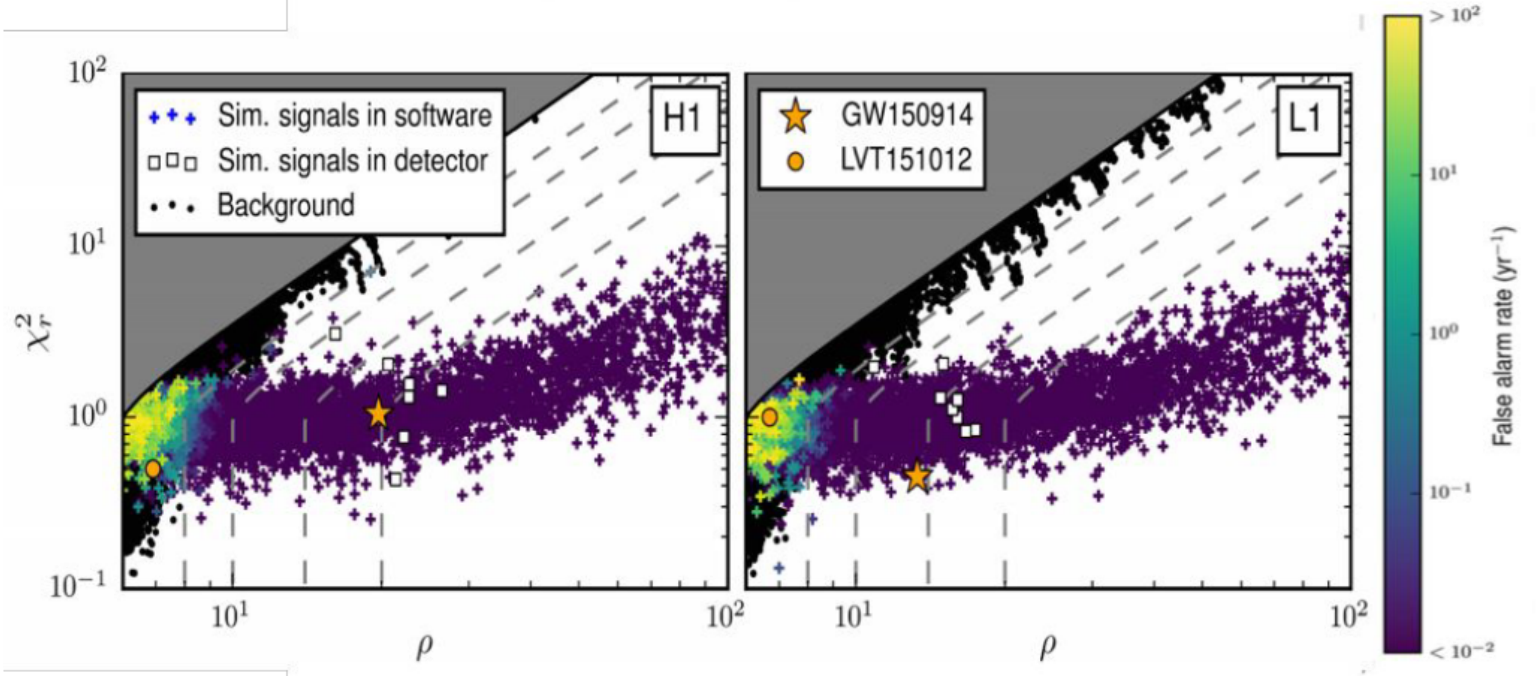


Abbot et al. 2016 arXiv:1602.03839



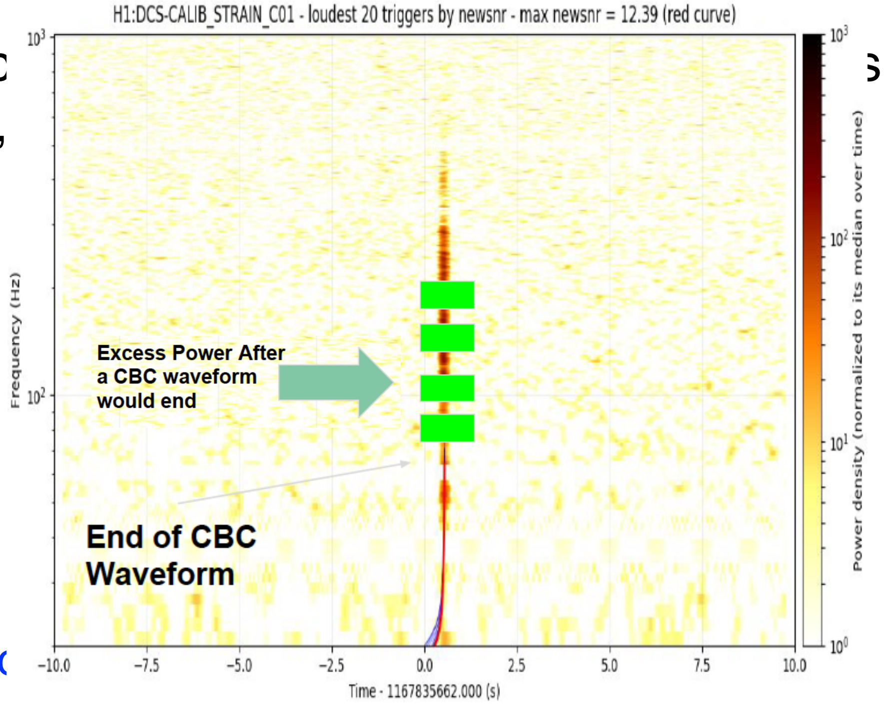
Generalized detection statistic

- Re-weight the SNR (aka ρ) by time frequency χ^2 - form $\hat{\rho}$ (ρ, χ^2) – a new detection statistic to replace SNR
- Normalized so that if χ^2 /ndof is ≤ 1, $\hat{\rho} = \rho$
- Suppresses detection statistic for glitches, keeps it as a measure of "loudness" for real signals, to better distinguish between them.



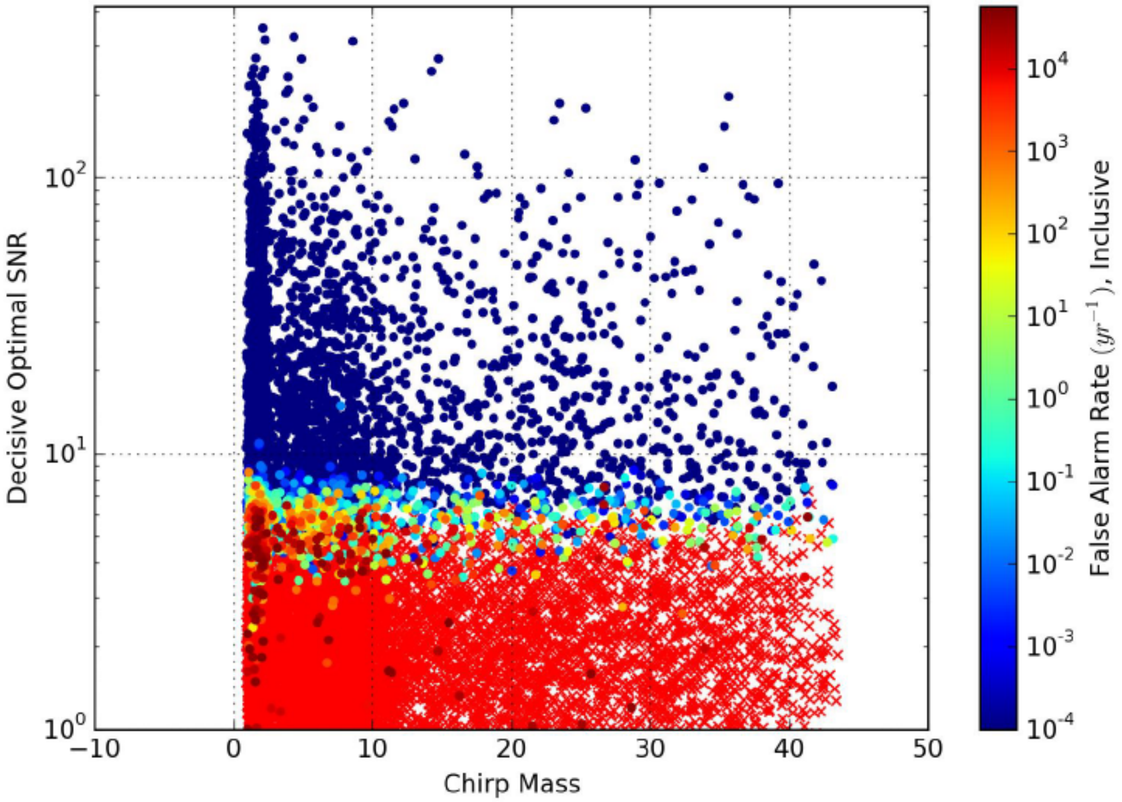
O2 Blip-glitch killer

- Check for excess power at frequencies above those expected from template
- For each trigger, look for excess power at frequencies beyond where the template waveform should have ended.
- Tiles of Sine-Gaussians with configurable Q and central frequencies
- Define a new χ^2 b s
(A. Nitz, CQG 35,

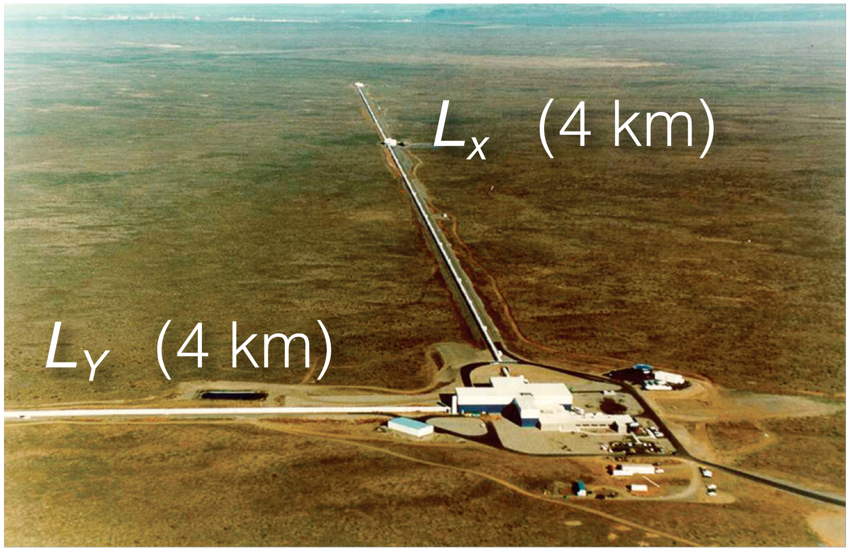


Veto safety – we should not be vetoing the signals we are looking for!

- Inject simulated signals (with wide range of SNRs, distances, masses, etc) into real LIGO data (assumed to be all noise, no signal!) over the full observation time.
- Run through detection pipeline, determine "loudness".
- Compare with background distribution, estimate False Alarm Probability (FAP) and False Alarm Rat (FAR).
- Events with low FAR (or high Inverse FAR, IFAR) are "detected".
- From this, measure detection efficiency vs distance
- From this, compute sensitive volume.
- For N detected events, astrophysical rate is R = N/VT.
- We can compute this for different source classes (BNS, NSBH, BBH, …).



Calibrating GW detectors



$$\Delta L_{\mathrm{free}} = \Delta L_x(t) - \Delta L_y(t)$$

$$h(t) = \frac{\Delta L_{\mathrm{free}}(t)}{L}$$

The “differential arm length change” (DARM_ERR) is the uncalibrated strain:

$$\text{DARM\_ERR(t)} = R * Lh(t) + n(t) = R * [Lh(t) + n_h(t)]$$

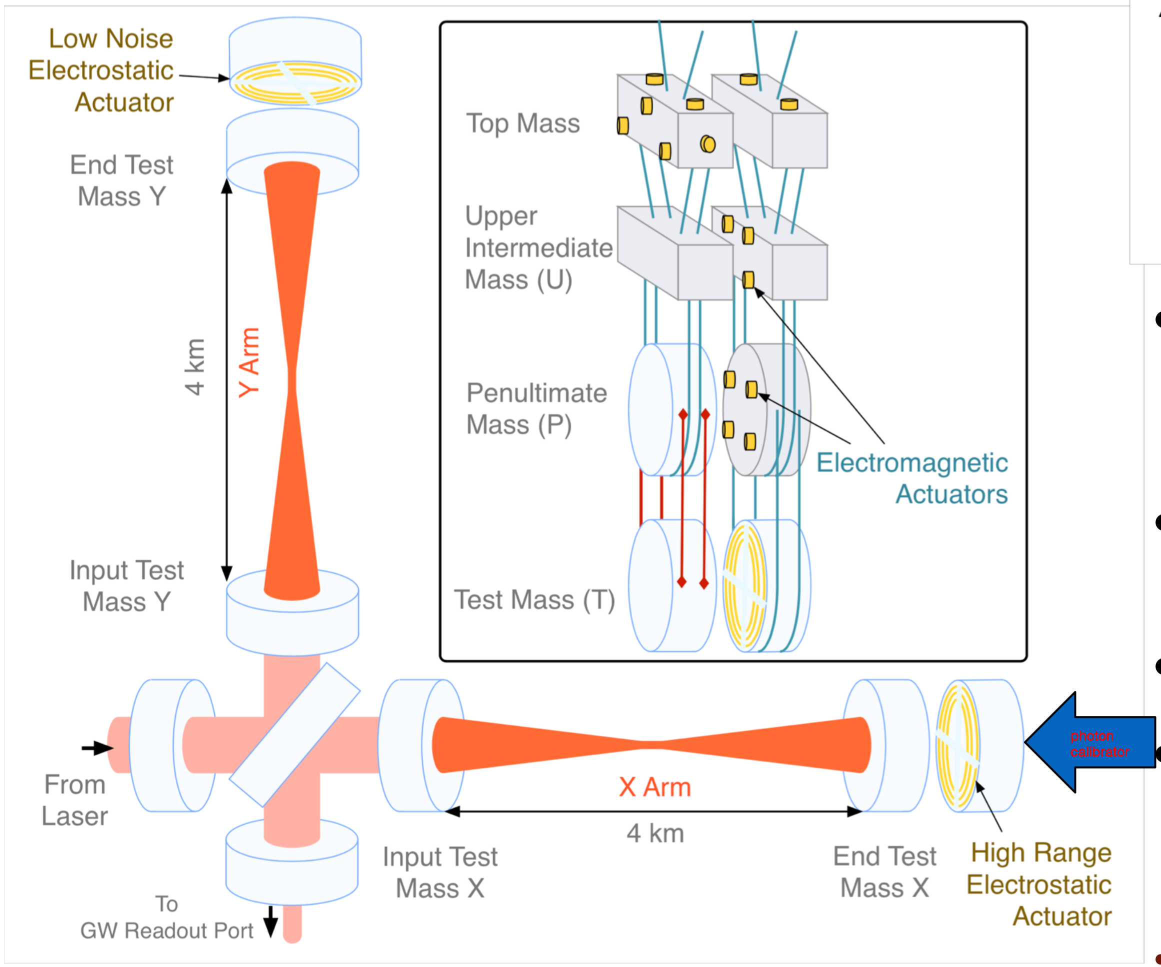
R is the “detector response” to strain. n_h are all the noises, *referenced to strain*.
GW detectors are engineered to be (ideally) Linear and Time Invariant (LTI),
So that R is most naturally a constant function of frequency, not time.

$$\text{DARM\_ERR(f)} = R * L\tilde{h}(f) + \tilde{n}(f) = R(f) \times [Lh(f) + \widetilde{n_h}(f)]$$

Calibration is about turning DARM_ERR(t) into h(t) using R(f) (and ignoring n(t)!).
Its about the response to signal, it’s not about noise at all (that’s DQ).

What is the DARM servo loop?

Differential arm (DARM) control system



C. Cahillane et al. 2017 PRD 96, 102001

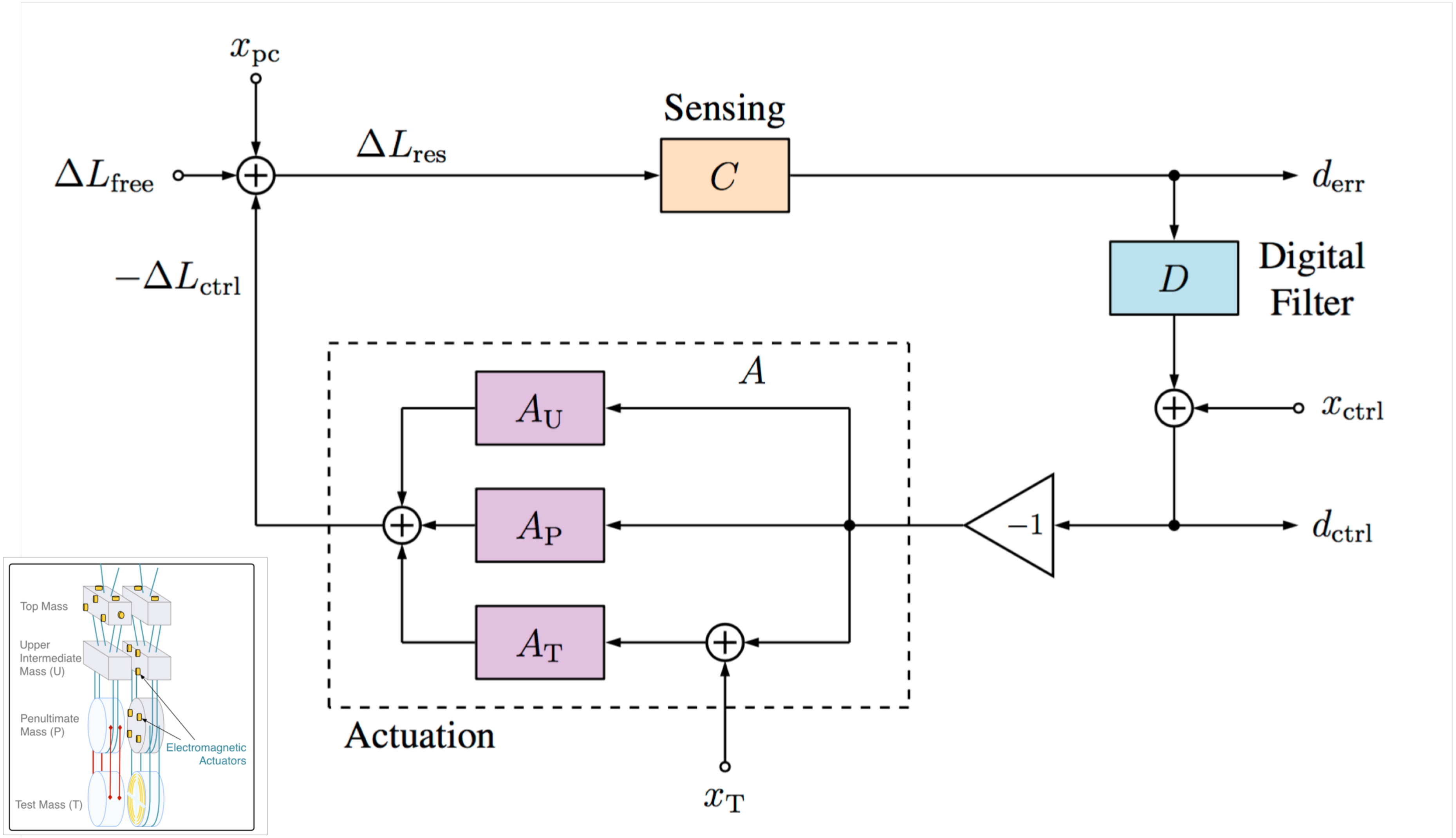
DARM displacement:

$$\Delta L_{\text{free}} = \Delta L_x(t) - \Delta L_y(t)$$

$$h(t) = \frac{\Delta L_{\text{free}}(t)}{L}$$

- Need to hold the optical cavities (XARM, YARM, MICH, PRC, SRC) on resonance
- Quadruple pendulum suspension systems
- Active seismic isolation systems
- Not enough – DARM displacement must be further controlled!
- “Active null instrument”

The DARM Control Loop

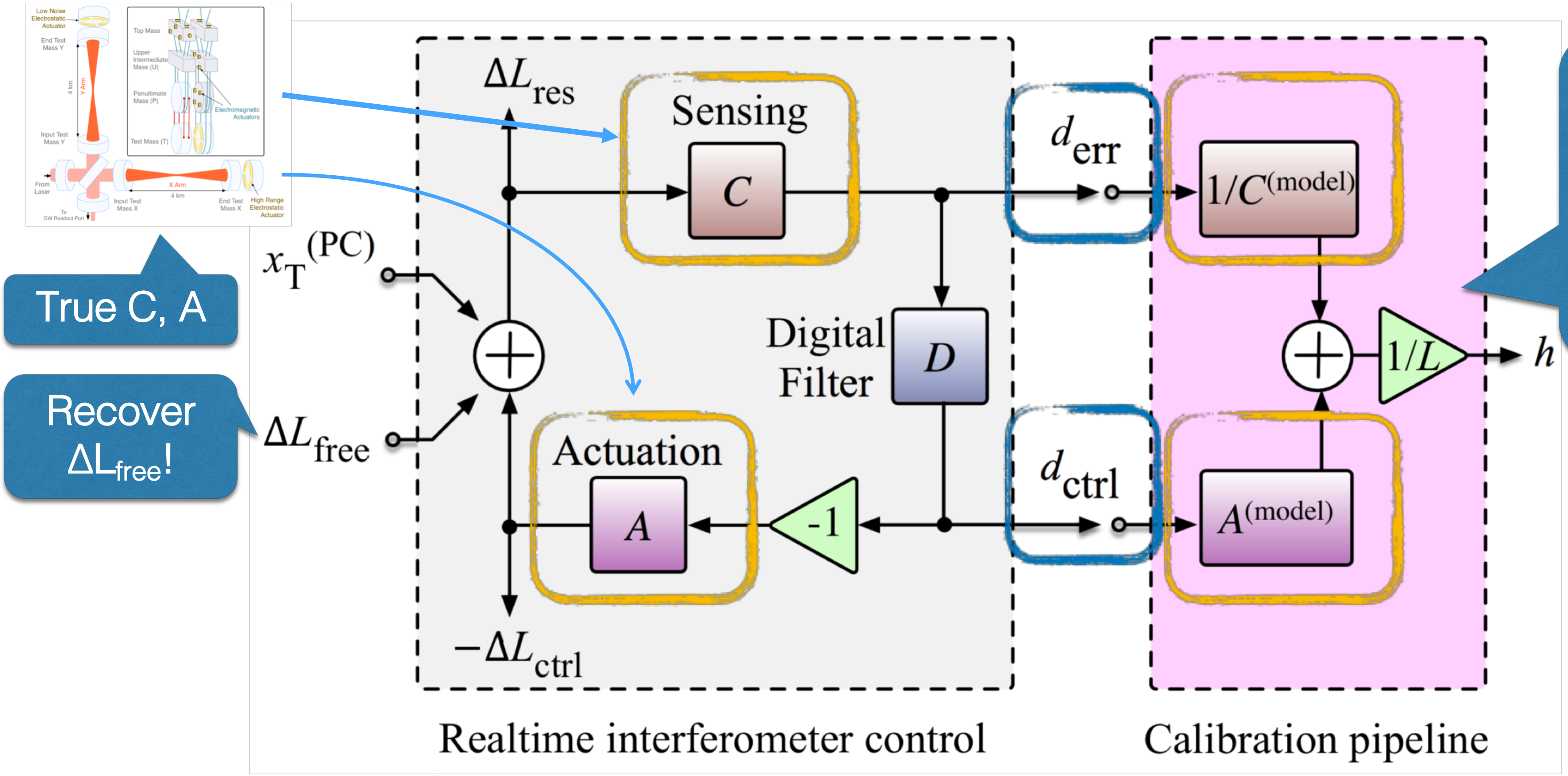


Viets et al., arXiv 1710.09973 , Class. Quantum Gravity, 35, 095015 (2018).

What does calibration do?

Differential arm (DARM) control loop

B. P. Abbott, et al. 2017 PRD 95 062003



True C, A

Recover $\Delta L_{\rm free}$!

Model the response (both C and A) as best we can!

$$\Delta L_{\rm free}(f) = \frac{1}{C(f)} d_{\rm err}(f) + A(f) d_{\rm ctrl}(f)$$

$$\Delta L_{\rm free}(t) = \frac{1}{\mathcal{C}(t)} * d_{\rm err}(t) + \mathcal{A}(t) * d_{\rm ctrl}(t)$$

$$h(t) = \frac{\Delta L_{\rm free}(t)}{L}$$

Time dependent correction factors (TDCFs)

- If Sensing (C) and Actuation (A) models are perfectly known at any given time, there's no systematic error (there's still statistical uncertainty).

$$\Delta L_{\mathrm{free}} = \frac{1}{C_{\mathrm{model}}} d_{\mathrm{err}} + A_{\mathrm{model}} d_{\mathrm{ctrl}}$$

- But the model parameters are imperfectly modeled, and also changing slightly – due to charge accumulation around the test/reaction mass, optical alignment drifts in the arm cavities, etc.

- Use high SNR calibration lines, track temporal variations in DARM loop model parameters, and correct them

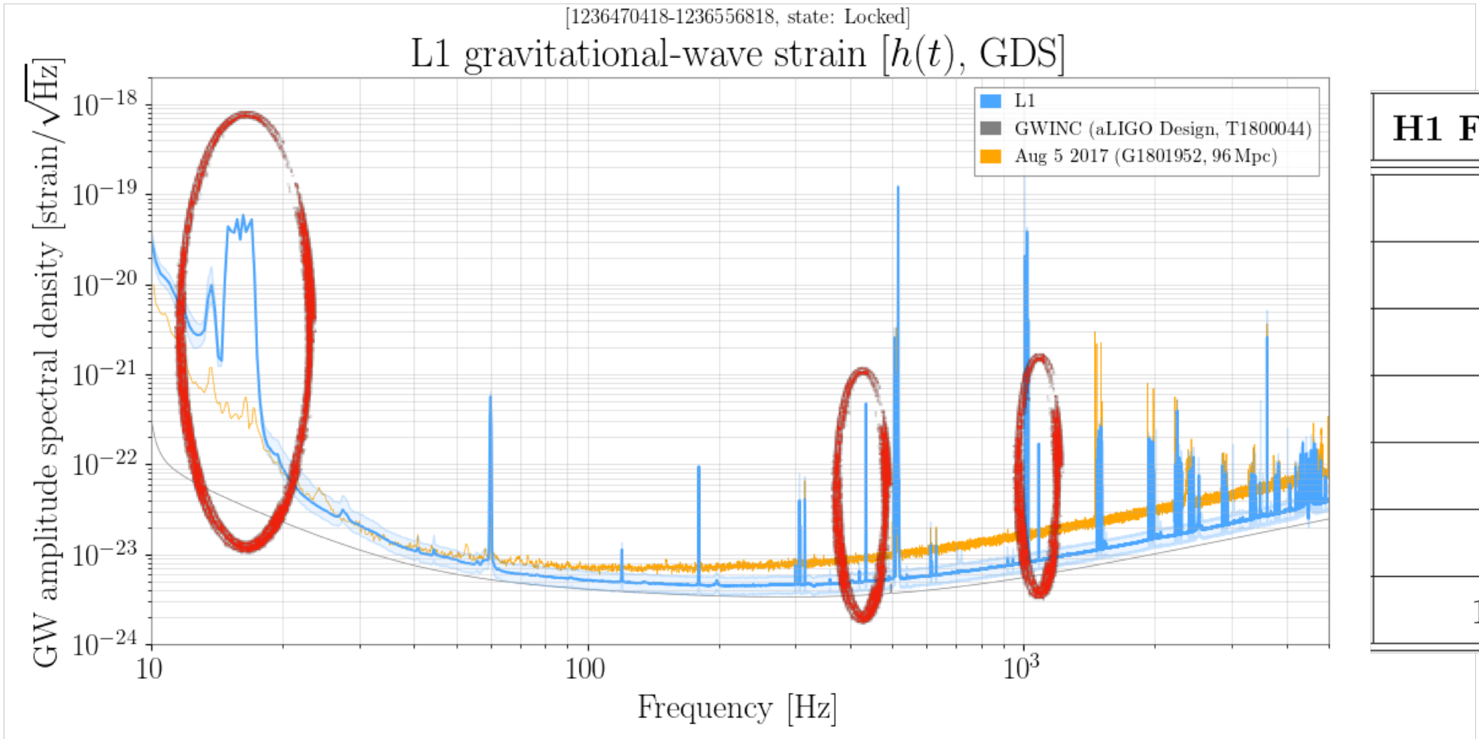
$$\Delta L_{\mathrm{free}} = \frac{1}{\kappa_C(t) C_{\mathrm{model}}} d_{\mathrm{err}} + \kappa_A(t) A_{\mathrm{model}} d_{\mathrm{ctrl}}$$

– See Viets et al. 2018 (arXiv:1710.09973), T1700106 for details

TDCFs

- Including factors for 3 actuation stages

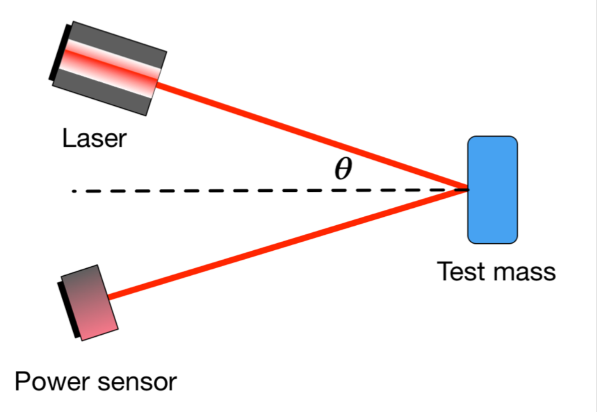
Calibration lines – always in the data …



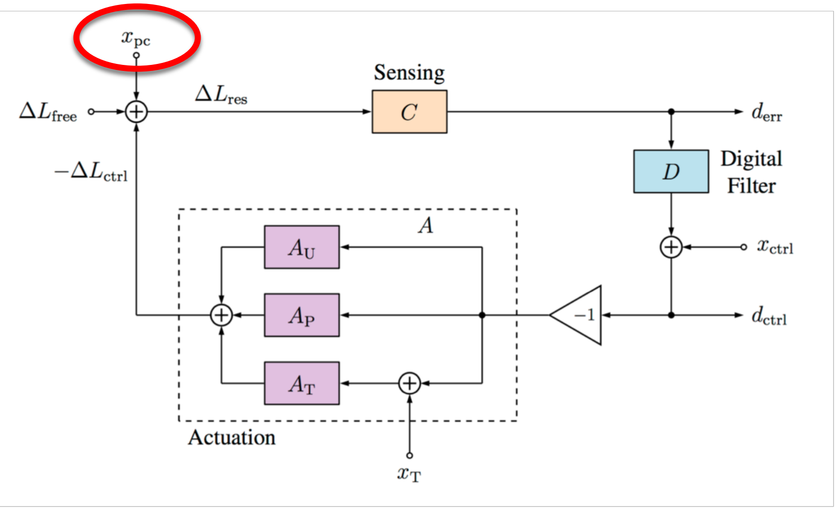
H1 Freq. (Hz)	**L1 Freq. (Hz)**
7.93	N/A
16.7	15.1
17.3	15.7
17.9	16.3
19.1	16.9
433.7	434.9
1083.7	1083.1

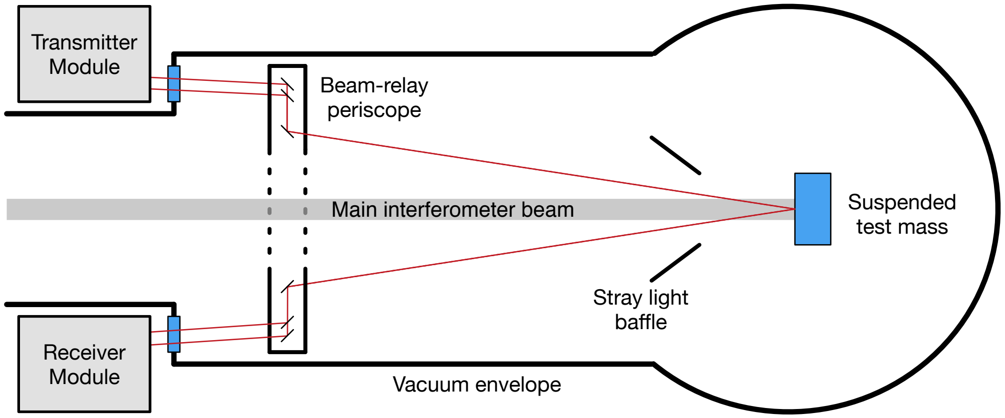
- Better characterize the response – constantly measuring C(f), A(f), and TDCFs.
- Minimize footprint in the most sensitive, astrophysically interesting band of the detectors.
- Can be removed in post-processing – “clean” data

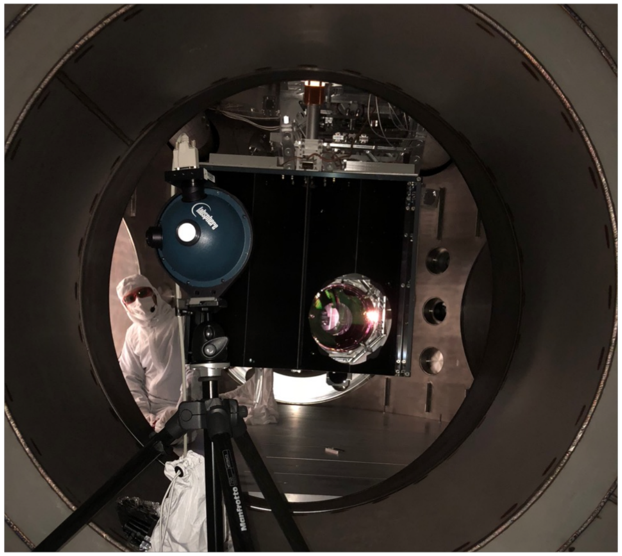
Photon calibrator – ~1% precision



$$x(f) = \frac{2\cos(\theta)}{c} P(f) \frac{1}{M(2\pi f)^2} \mathcal{R}\ \mathcal{G}(f)$$





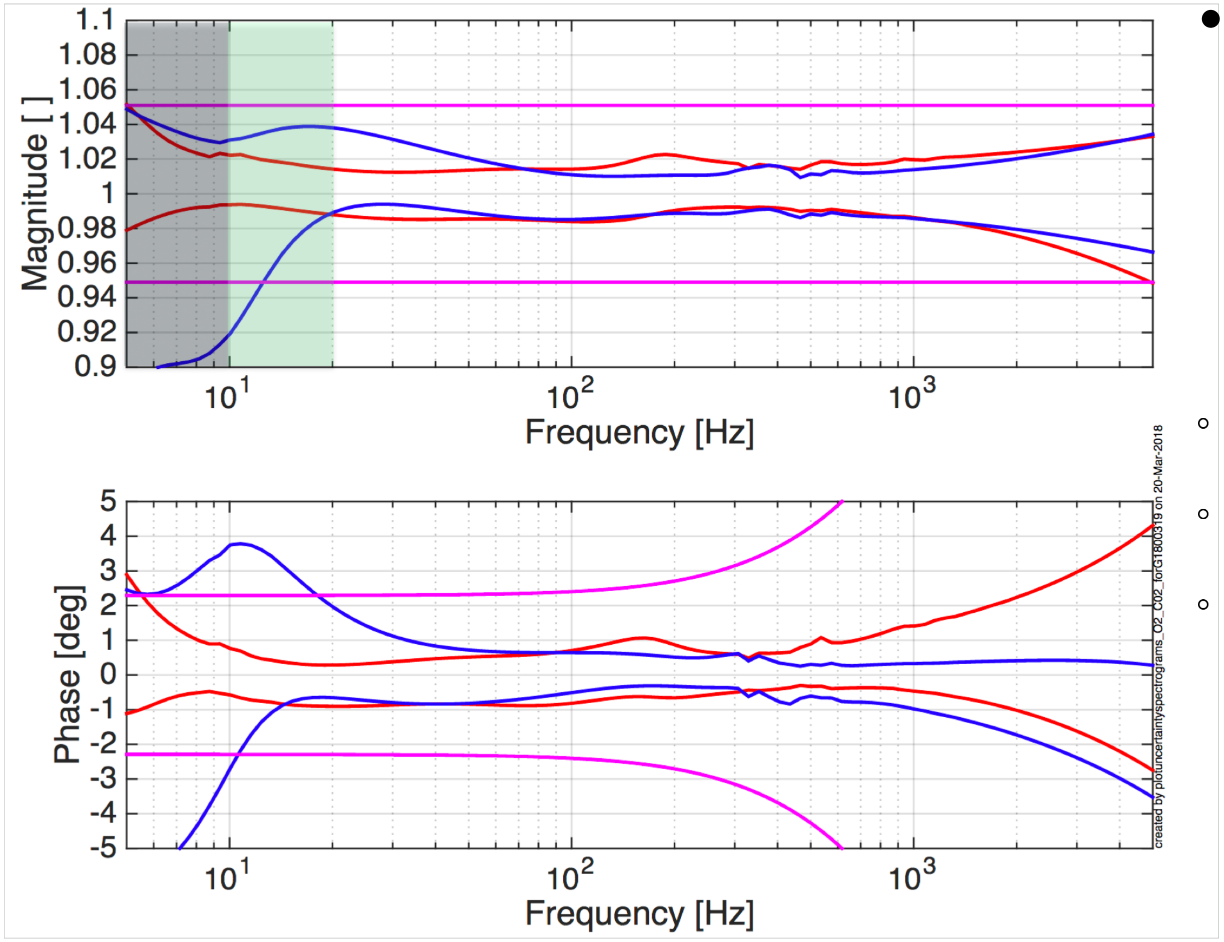


Calibration uncertainty estimation – $R_{true}(f) / R_{model}(f)$

H

L

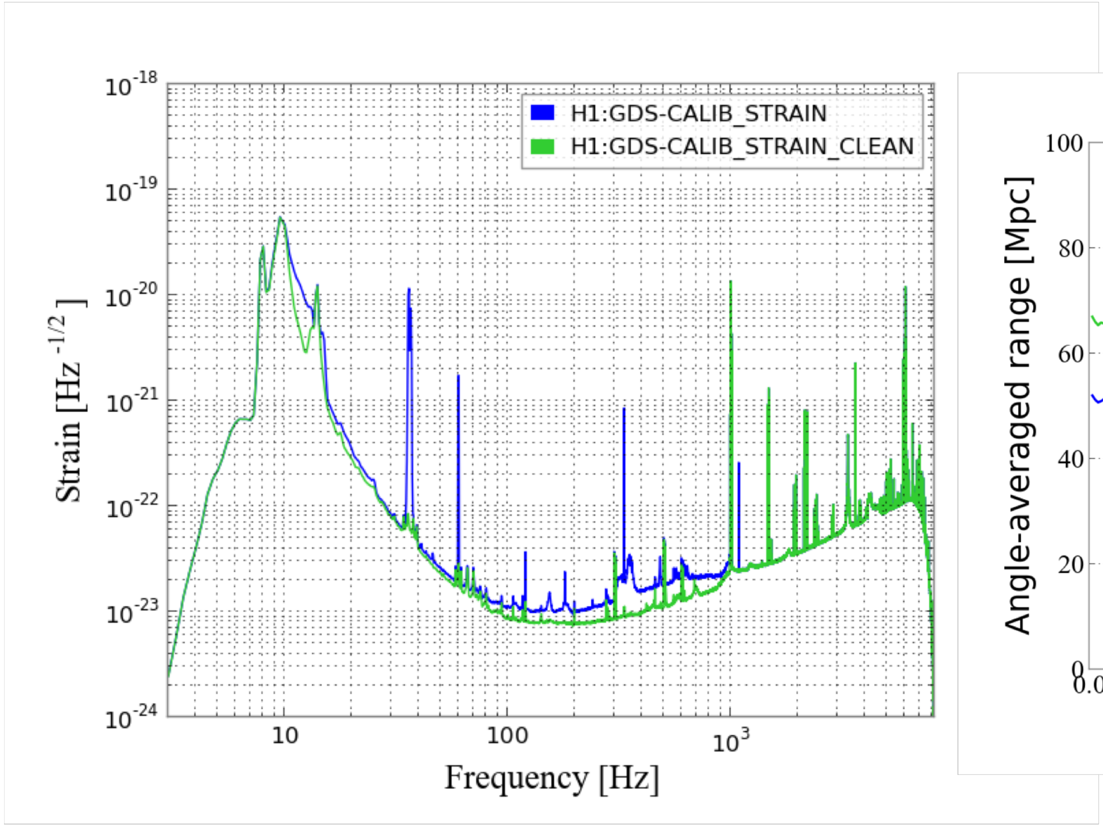
V

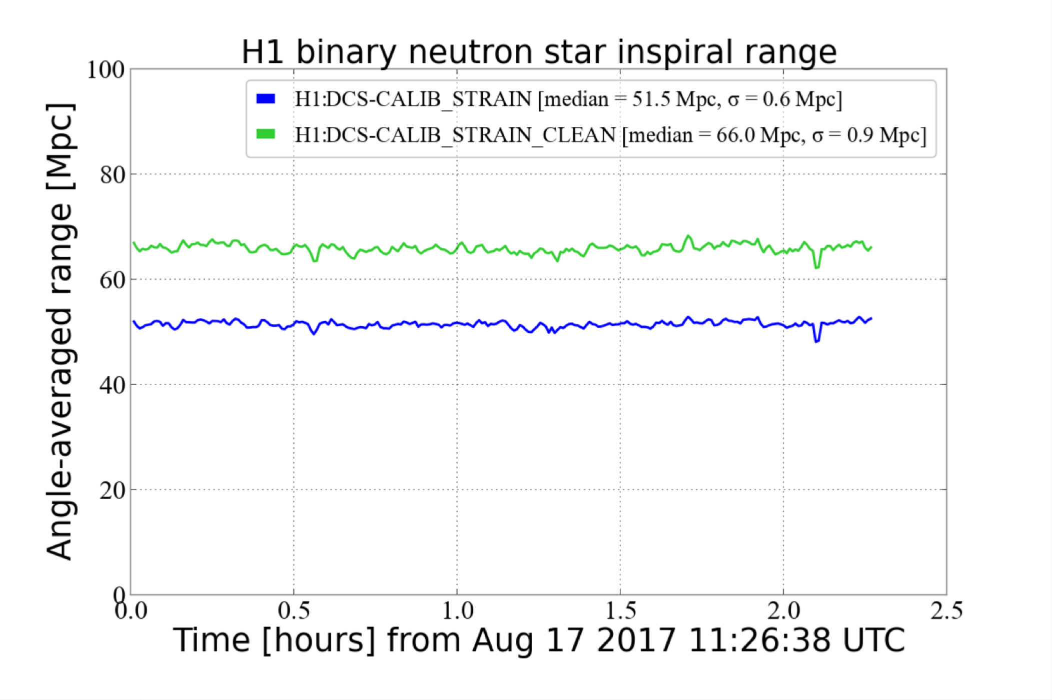


- Maximum excursion of the 1-sigma limits of statistical uncertainty and systematic error from unity magnitude/zero phase (20–1024 Hz):
  - Magnitude error: ~2–3 %
  - Phase error: ~2–3 deg
  - ~1% (mag) and ~1 deg (phase) in the most sensitive region (~100 Hz)

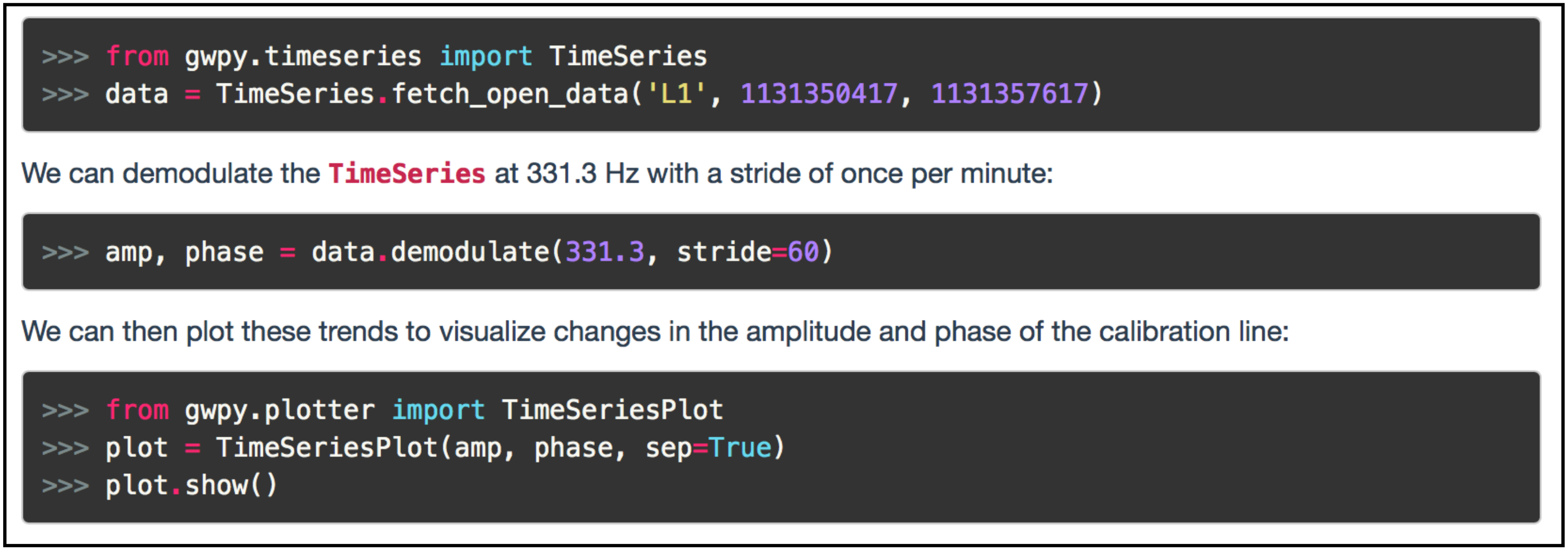
- Best results achieved in offline calibration (3 months after O2 ended) – time-dependent variations (TDCFs) were corrected

Low-latency Noise Subtraction – "CLEAN" data





Monitoring calibration lines



```
>>> from gwpy.timeseries import TimeSeries
>>> data = TimeSeries.fetch_open_data('L1', 1131350417, 1131357617)
```

We can demodulate the `TimeSeries` at 331.3 Hz with a stride of once per minute:

```
>>> amp, phase = data.demodulate(331.3, stride=60)
```

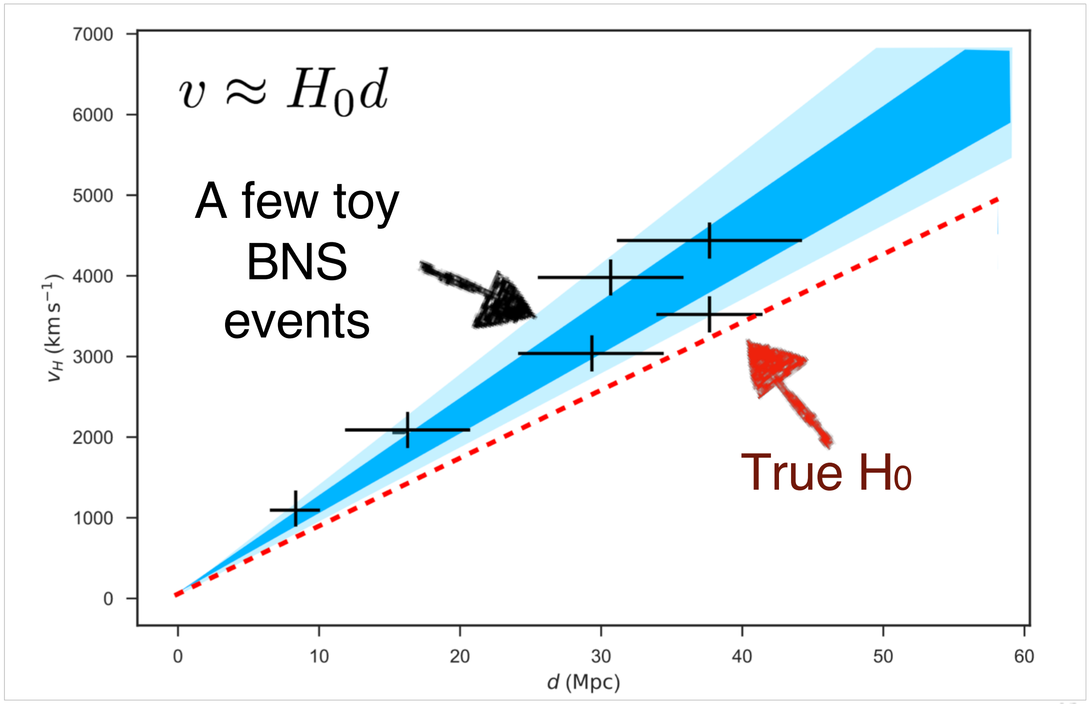
We can then plot these trends to visualize changes in the amplitude and phase of the calibration line:

```
>>> from gwpy.plotter import TimeSeriesPlot
>>> plot = TimeSeriesPlot(amp, phase, sep=True)
>>> plot.show()
```

Support for calibration line tests is
now a feature in GWPy
(see Duncan Macleod’s tutorial)

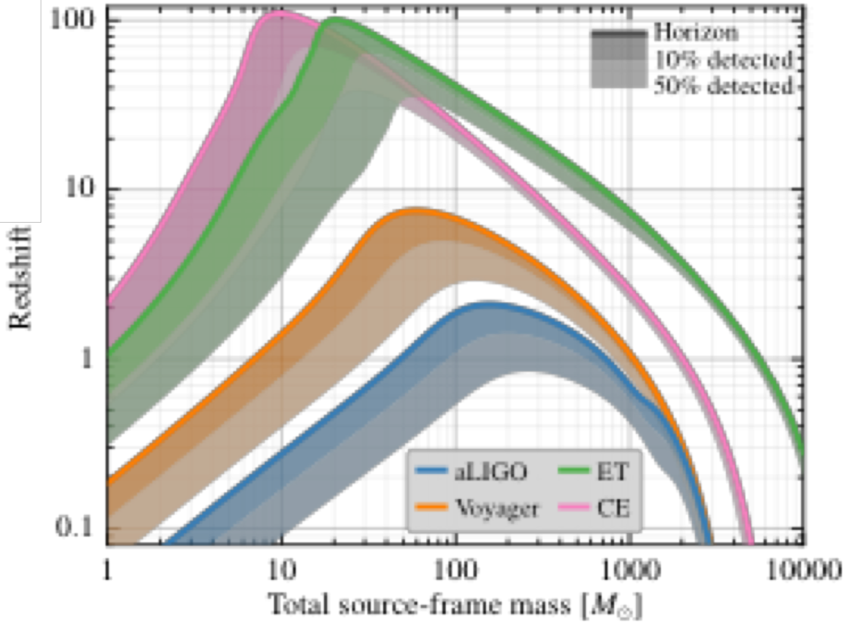
Why do we need precise calibration? Sensitive volume / merger rates, Cosmology

Credit: W. Farr



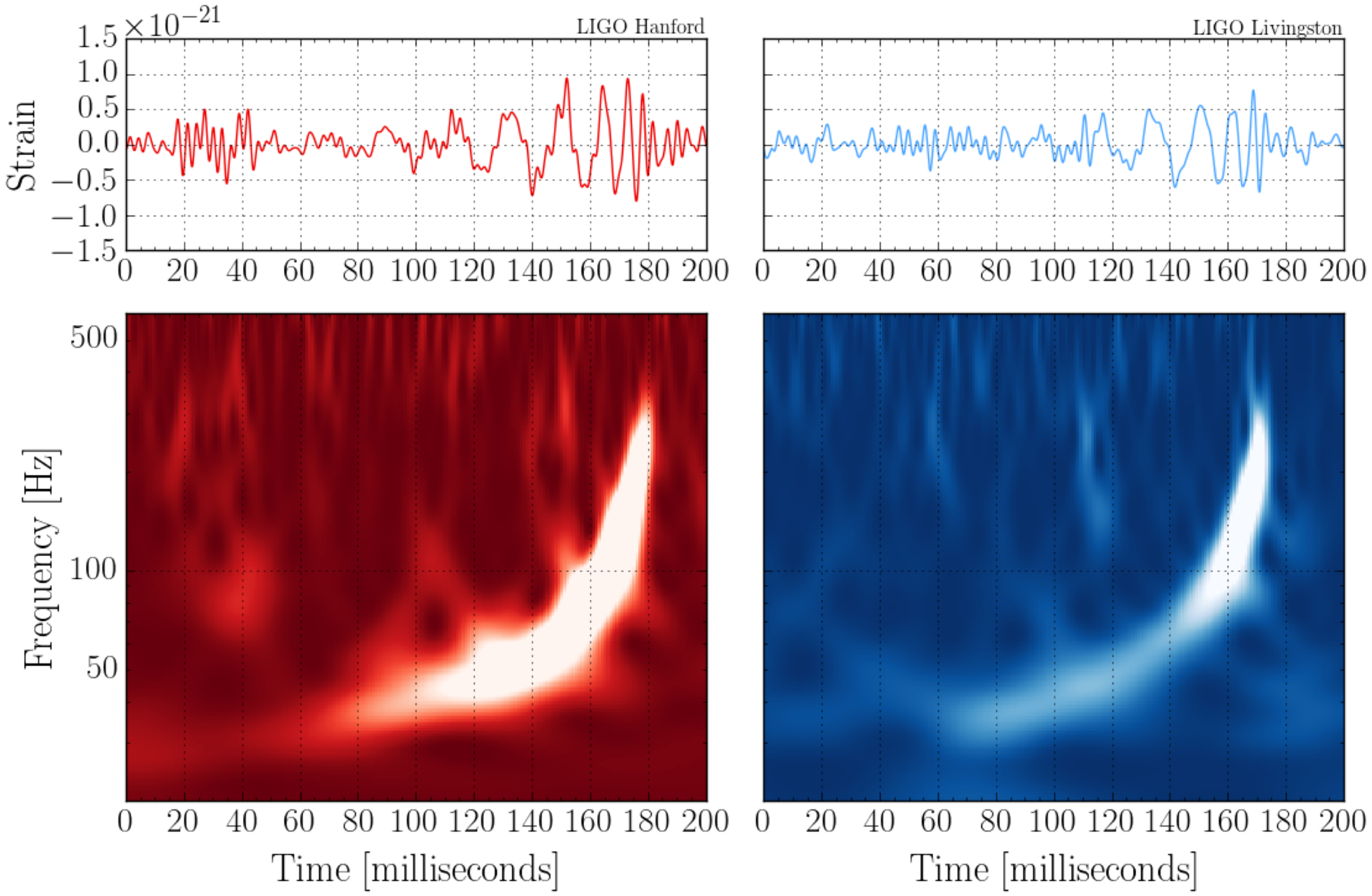
- Correlated calibration systematic errors impact H0 measurement

Credit: E. Hall



- Calibration systematic errors and statistical uncertainties impact the horizon distance, CBC rates, especially at high redshift

Precision calibration allows us to compare the data with predictions from General Relativity and look for deviations …



see also Abbot et al., Phys Rev Lett **116**, 061102 (2016)

Next up:

- Accessing the LIGO & Virgo data from GWOSC
- Signal processing with GWpy
- Finding GWs from compact binary coalescence (CBC)
- Using pyCBC to find the signals in noisy data
- Bayesian estimation of the parameters of the signal (masses, spins, etc)
- Localizing the signal source on the sky,
  alerting EM astronomers