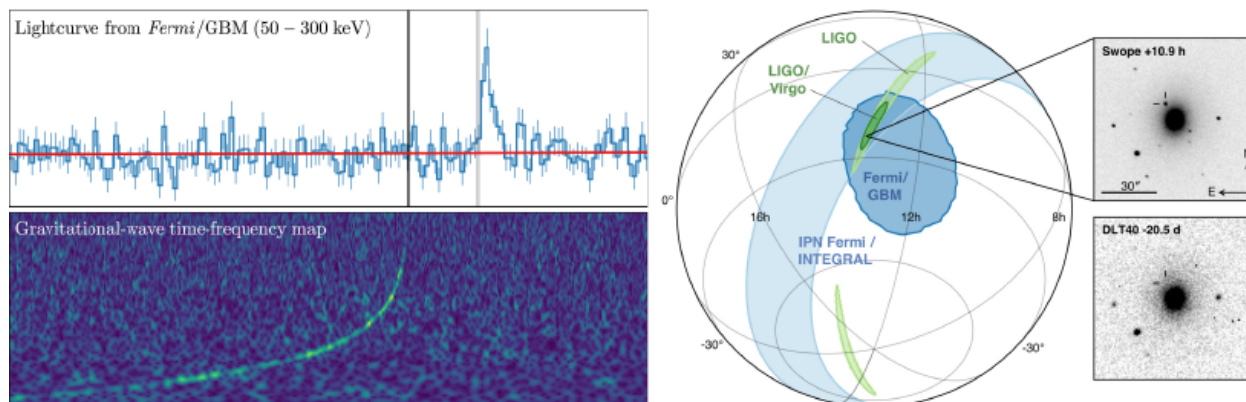

LIGO/Virgo Public Alerts User Guide Documentation

LIGO Scientific Collaboration, Virgo Collaboration

May 12, 2020

Contents

1	Contents	3
1.1	Getting Started Checklist	3
1.2	Observing Capabilities	4
1.3	Procedures	6
1.4	Alert Contents	10
1.5	Sample Code	23
2	Appendix	37
2.1	Change Log	37
2.2	Glossary	38
	Index	39



Welcome to the LIGO/Virgo Public Alerts User Guide! This document is intended for both professional astronomers and science enthusiasts who are interested in receiving alerts and real-time data products related to gravitational-wave (GW) events.

Warning: Some technical details of LIGO/Virgo public alerts may change before the start of Observing Run 3 (O3) in 2019. In particular, details of the alert format and the [GraceDb](#) public portal may evolve. Please check this document regularly for announcements and updates.

Three sites (*LHO*, *LLO*, *Virgo*) together form a global network of ground-based GW detectors. The [LIGO Scientific Collaboration](#) and the [Virgo Collaboration](#) jointly analyze the data in real time to detect and localize transients from compact binary mergers and other sources GW bursts. When a signal candidate is found, an alert is sent to astronomers in order to search for counterparts (electromagnetic waves or neutrinos).

Advanced LIGO and Advanced Virgo are preparing for their third observing run (O3) in early 2019. For the first time, **LIGO/Virgo alerts will be public**. Alerts will be distributed through NASA's Gamma-ray Coordinates Network (*GCN*). There are two types of alerts: human-readable *GCN Circulars* and machine-readable *GCN Notices*. This document provides a brief overview of the procedures for vetting and sending GW alerts, describes their contents and format, and includes instructions and sample code for receiving GCN Notices and decoding GW sky maps.

1.1 Getting Started Checklist

In addition to reading this user guide, we encourage you to sign up for a certain e-mail lists and community resources. Make sure that you have completed the items on this checklist.

1.1.1 1. Read This User Guide

Pay particular attention to the *Alert Contents* section of this user guide to familiarize yourself with the contents of machine-readable LIGO/Virgo *GCN Notices*. Play with the *Sample Code* to receive example GCN Notices and practice working with *localization maps*.

1.1.2 2. Subscribe to GCN Circulars

Subscribe to *GCN Circulars* and review the *instructions for posting GCN Circulars*. A *GCN Circular* is a short, public bulletin to rapidly report an astronomical observation. GCN Circulars are distributed by email and *archived online*.¹ LIGO/Virgo uses GCN Circulars to announce detections, and the astronomy community expects participants to promptly disseminate preliminary reports of follow-up observations of LIGO/Virgo counterparts using GCN Circulars as well.

Important: GCN Circulars can only be posted from registered email addresses. You *must* sign up for GCN Circulars *in advance* in order to post to the list.

¹ GCN Circulars are similar to *Astronomer's Telegrams* (ATels). By longstanding convention, the gamma-ray burst and gravitational-wave astronomy communities use GCNs, whereas the supernova community uses ATels.

1.1.3 3. Join the OpenLVEM Community

Sign up to the OpenLVEM mailing list by following the [OpenLVEM instructions](#). LIGO/Virgo will use this list to make announcements and solicit input. It is also a great place to ask questions or discuss issues related to LIGO/Virgo public alerts. Documents relating to teleconferences and in-person meetings are available at [OpenLVEM wiki](#).

1.1.4 4. Visit GraceDb

Familiarize yourself with [GraceDb](#), LIGO/Virgo's online portal for alerts and real-time results.

Note: The public interface of GraceDb is under development and will be complete some time before O3 starts. Please check back periodically.

1.2 Observing Capabilities

This section summarizes the observing capabilities of the global gravitational-wave detector network as of late 2018. This as a quick reference to the anticipated commissioning and observing schedule, sensitivity to gravitational-wave transients, and sky localization accuracy, as described in the following external documents:

- White Paper¹ on gravitational-wave data analysis and astrophysics
- Living Review² on prospects for observing and localizing gravitational-wave transients with Advanced LIGO, Advanced Virgo, and KAGRA
- Working Plan Towards O3³

1.2.1 Timeline

The gravitational-wave observing schedules is divided into Observing Runs, epochs of months to years of operation at fixed sensitivity; down time for construction and commissioning, and transitional Engineering Runs between commissioning and observing runs. The long-term observing schedule is shown below.

Engineering Run 13 (ER13) is scheduled to start 2018-12-14 16:00 UTC and end 2018-12-18 14:00 UTC. Engineering Run 14 (ER14) is scheduled not before March 2019 for around 4 weeks and will transition without interruption to Observing Run 3 (O3).

During O3, we expect that all three detectors will observe continuously and without significant interruptions for one year. It is possible that the Japanese KAGRA detector may come online and become part of the joint gravitational-wave analysis at some point during O3. The near-term observing schedule is shown below, reproduced from³.

¹ LIGO Scientific Collaboration & Virgo Collaboration 2018, *The LSC-Virgo White Paper on Gravitational Wave Data Analysis and Astrophysics*, LIGO Document T1800058-v1. <https://dcc.ligo.org/LIGO-T1800058/public>

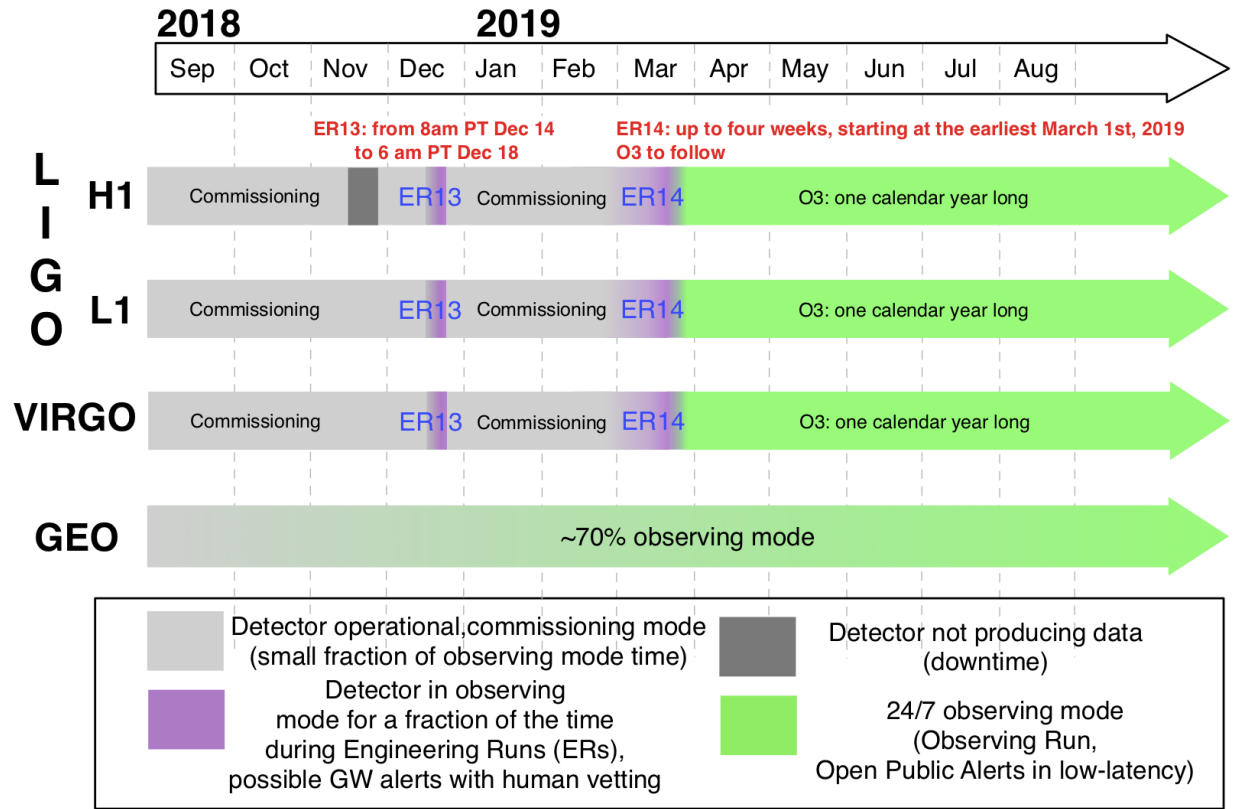
² Abbott, B. P., Abbott, R., Abbott, T. D., et al. 2018, *Living Rev. Rel.*, 21, 3. <https://doi.org/10.1007/s41114-018-0012-9>

³ LIGO Scientific Collaboration & Virgo Collaboration 2018, *LIGO-Virgo Working Plan Towards O3*, LIGO Document G1801056-v4. <https://dcc.ligo.org/LIGO-G1801056/public>

LIGO-VIRGO Joint Run Planning Committee

Working schedule for O3

(Public document G1801056-v4, based on G1800889-v7)



Important: During both Engineering Runs, the detector configuration and analysis software will be changing frequently. We will send public alerts only for highly confident and scientifically significant events and with longer than normal latency. However, at the transition to O3 occurs, we will switch to routine low-latency data analysis and sending public alerts for all significant gravitational-wave events.

1.2.2 Sensitivity and Localization Accuracy

The following O3 projections are adapted from the Living Review² on prospects for observing and localizing gravitational-wave transients with Advanced LIGO, Advanced Virgo, and KAGRA. The range (luminosity distance of detectable sources, averaged over sky position and source orientation) given the detectors' anticipated sensitivity are listed in the table below.

Detector	BNS Range (Mpc)	Burst Range (Mpc)
LIGO	120–170	75–90
Virgo	65–85	40–50
Kagra	8–25	

We expect 1–10 *BNS* events over the course of O3. For BNS events, the median localization accuracy of in terms of the 90% credible area will be 120–180 deg². 12–21% of BNS mergers will be localized to less than 20 deg².

1.2.3 Event Rates

See the [rates presentation](#) from recent Town Hall meeting for LIGO and Virgo's most current estimates of astrophysical rates of compact binary mergers.

1.3 Procedures

In this section we describe the different online searches looking for GW signals. The corresponding triggers from multiple pipelines close enough in time will be considered as originating from the same physical event and will be grouped in a unique *superevent*. See the following pages for technical details.

1.3.1 Online Pipelines

During the third LIGO observation run a number of search pipelines will be running in a low latency, online mode. These can be divided into two groups, modeled and unmodeled. The modeled (CBC) searches specifically look for signals from compact binary mergers of neutron stars and black holes (BNS, NS-BH, and BBH systems). The unmodeled (Burst) searches on the other hand, are capable of detecting signals from a wide variety of astrophysical sources in addition to compact binary mergers: core-collapse of massive stars, magnetar star-quakes, and more speculative sources such as intersecting cosmic strings or as-yet unknown GW sources.

Modeled Search

GstLAL, **MBTAOnline**, **PyCBC Live** and **SPIIR** are matched-filtering based analysis pipelines that rapidly identify compact binary merger events, with sub-minute to 1 minute latencies. They use discrete banks of waveform templates to cover the target parameter space of compact binaries, with all pipelines covering the mass ranges corresponding to BNS, NS-BH, and stellar mass BBH systems. However, GstLAL and PyCBC Live and SPIIR also include intermediate-mass BBH systems and the O2 banks differ in detail from pipeline to pipeline.

A coincident analysis is performed by GstLAL, PyCBC Live, and MBTAOnline, where candidate events are extracted separately at each detector via matched-filtering and later combined across detectors. SPIIR extract candidates of each detector via matched-filtering and look for coherent responses in other detectors that a localization of the source can be constructed. Of the four pipelines, GstLAL and MBTAOnline use several matched filters to cover the detector bandwidth i.e., the matched filter is split across multiple frequency bands. All pipelines also implement different kinds of signal-based vetoes to reject instrumental transients which cause large SNR values but can otherwise be easily distinguished from compact binary coalescence signals

GSTLAL¹ is a matched-filter pipeline designed to find gravitational-waves from compact binaries in low-latency. It uses the likelihood-ratio, which increases monotonically with signal probability, to rank candidates, and then uses Monte Carlo sampling methods to estimate the distribution of likelihood-ratios in noise. This distribution can then be used to compute a false alarm rate and p-value.

SPIIR^{2,3} applies zero-latency SPIIR filters to approximate matched-filtering results. It selects high-SNR events from each detector and find coherent responses from other detectors. It constructs background by time-shifting detector data one hundred times over a week to form a background statistic distribution used to evaluate foreground candidate significance.

MBTA⁴ constructs its background by making every possible coincidence from single detector triggers over few hours of recent data. It then folds in the probability of a pair of triggers passing the time coincidence test.

¹ Messick, C., Blackburn, K., Brady, P., et al. 2017, *Phys. Rev. D*, 95, 042001. <https://doi.org/10.1103/PhysRevD.95.042001>

² Hooper, S., Chung, S. K., Luan, J., et al. 2012, *Phys. Rev. D*, 86, 024012. <https://doi.org/10.1103/PhysRevD.86.024012>

³ Chu, Q. 2017, Ph.D. Thesis. <https://api.research-repository.uwa.edu.au/portalfiles/portal/18509751>

⁴ Adams, T., Buskulic, D., Germain, V., et al. 2016, *Class. Quantum Grav.*, 33, 175012. <http://doi.org/10.1088/0264-9381/33/17/175012>

PyCBC Live⁵⁶ estimates the noise background by performing time-shifted analyses using triggers from a few hours of recent data. Single-detector triggers from one of the LIGO detectors are time shifted by every possible multiple of 100 ms, thus any resulting coincidence must be unphysical given the 10 ms light travel time between detectors. All such coincidences are recorded and assigned a ranking statistic; the false alarm rate is then estimated by counting accidental coincidences louder than a given candidate, i.e. with a higher statistic value.

Unmodeled Search

cWB⁷ is a power excess algorithm focused to identify gravitational-like signals with short time duration. It uses a wavelet transformation to identify time-frequency pixels which can be grouped in a single cluster if they satisfy neighboring conditions. A tuned version for compact-binary coalescences chooses the time-frequency pixels if they mainly follow a frequency increasing pattern. A maximum-likelihood-statistics calculated over the cluster is used to identify the proper parameter of the event, in particular the probability of the source direction and the coherent network signal-to-noise ratio. The last one is used to assign detection significance to the found events.

oLIB⁸ uses Q transform to decompose GW strain data into several time-frequency planes of constant quality factors Q , where $Q \sim \tau f_0$. The pipeline flags data segments containing excess power and searches for clusters of these segments with identical f_0 and Q spaced within 100 ms of each other. Coincidences among the detector network of clusters with a time-of-flight window up to 10 ms are then analyzed with a coherent (i.e., correlated across the detector network) signal model to identify possible GW candidate events.

Coincident with External Trigger Search

RAVEN⁹ In addition, we will operate the Rapid On-Source VOEvent Coincidence Monitor (RAVEN), a fast search for coincidence between GW online pipeline events and gamma-ray bursts or galactic supernova notifications. A similar approach is under development for high energy neutrinos.

1.3.2 Localization and Parameter Estimation

Immediately after one of the *search pipelines* reports an event, localization and parameter estimation analyses begin. These analyses all use Bayesian inference to calculate the posterior probability distribution over the parameters (sky location, distance, and/or intrinsic properties of the source) given the observed gravitational-wave signal.

There are different parameter estimation methods for modeled (CBC) and unmodeled (Burst) events. However, in both cases there is a rapid analysis that estimates only the localization, and is ready in seconds, and a refined analysis that explores a larger parameter space and completes up to hours or a day later.

Modeled Events

BAYESTAR¹ is the rapid CBC localization algorithm. It reads in the matched-filter time series from the *search pipeline* and calculates the posterior probability distribution over the sky location and distance of the source by coherently modeling the response of the gravitational-wave detector network. It explores the parameter space using Gaussian quadrature, lookup tables, and sampling on an adaptively refined HEALPix grid. The localization takes tens of seconds and is included in the preliminary alert.

LALInference² is the full CBC parameter estimation algorithm. It explores a greatly expanded parameter space

⁵ Nitz, A. H., Dent, T., Dal Canton, T., Fairhurst, S., & Brown, D. A. 2017, *Astrophys. J.*, 849, 118. <https://doi.org/10.3847/1538-4357/aa8f50>

⁶ Dal Canton, T., & Harry, I. W. 2017. <https://arxiv.org/abs/1705.01845>

⁷ Klimentenko, S., Vedovato, G., Drago, M., et al. 2016, *Phys. Rev. D*, 93, 042004. <https://doi.org/10.1103/PhysRevD.93.042004>

⁸ Lynch, R., Vitale, S., Essick, R., Katsavounidis, E., & Robinet, F. 2017, *Phys. Rev. D*, 95, 104046. <https://doi.org/10.1103/PhysRevD.95.104046>

⁹ Urban, A. L. 2016, Ph.D. Thesis. <http://adsabs.harvard.edu/abs/2016PhDT.....8U>

¹ Singer, L. P., & Price, L. R. 2016, *Phys. Rev. D*, 93, 024013. <https://doi.org/10.1103/PhysRevD.93.024013>

² Veitch, J., Raymond, V., Farr, B., et al. 2015, *Phys. Rev. D*, 91, 042003. <https://doi.org/10.1103/PhysRevD.91.042003>

including sky location, distance, masses, and spins, and performs full forward modeling of the gravitational-wave signal and the strain calibration of the gravitational-wave detectors. It explores the parameter space using Markov chain Monte Carlo and nested sampling. For all events, there is an automated LALInference analysis that uses the least expensive CBC waveform models and completes within hours and may be included in a subsequent alert. More time-consuming analyses with more sophisticated waveform models are started at the discretion of human analysts, and complete days or weeks later.

Unmodeled Events

cWB, the burst *search pipeline*, also performs a rapid localization based on its coherent reconstruction of the gravitational-wave signal using a wavelet basis and the response of the gravitational-wave detector network³. The cWB localization is included in the preliminary alert.

Refined localizations for unmodeled bursts are provided by two algorithms that use the same MCMC and nested sampling methodology as LALInference. **LALInference Burst (LIB)**⁴ models the signal as a single sinusoidally modulated Gaussian. **BayesWave**⁵ models the signal as a superposition of wavelets and jointly models the background with both a stationary noise component and glitches composed of wavelets that are present in individual detectors.

1.3.3 Superevents

Superevents are a new abstraction to unify gravitational-wave candidates from multiple search pipelines. Each superevent is intended to represent a single astrophysical event.

A superevent consists of one or more event candidates, possibly from *different pipelines*, that are neighbors in time. At any given time, one event belonging to the superevent is identified as the *preferred event*. The superevent inherits properties from the preferred event such as time, significance, localization, and classification.

The preferred event may change after a preliminary alert has been sent, but the name of the superevent will stay the same. The naming scheme is described in the *alert contents* section.

Selection of the Preferred Event

When multiple online searches report events at the same time, the preferred event is decided by applying the following rules, in order:

1. Events that are detected in multiple interferometers are preferred over an events from a single interferometer.
2. Events from modeled *CBC* searches are preferred over events from unmodeled Burst searches (see *Searches* for details on search pipelines).
3. In the case of multiple CBC events, the event with the highest signal to noise ratio (SNR) is preferred. In the case of multiple Burst events, the event with the lowest false alarm rate (FAR) is preferred.

Note: A Preliminary GCN is automatically issued for superevents when the false alarm rate is lower than a threshold value.

Note: In case of an *offline* trigger upload from a pipeline, no preliminary GCN will be sent.

³ Klimenko, S., Vedovato, G., Drago, M., et al. 2011, *Phys. Rev. D*, 83, 102001. <https://doi.org/10.1103/PhysRevD.83.102001>

⁴ Lynch, R., Vitale, S., Essick, R., Katsavounidis, E., & Robinet, F. 2017, *Phys. Rev. D*, 95, 104046. <https://doi.org/10.1103/PhysRevD.95.104046>

⁵ Cornish, N. J., & Littenberg, T. B. 2015, *Class. Quantum Grav.*, 32, 135012. <https://doi.org/10.1088/0264-9381/32/13/135012>

1.3.4 Inference

For *CBC* events, we calculate a number of quantities that are inferred from the signal. In preliminary alerts, these quantities are based on the candidate significance and the matched-filter estimates of the source parameters. Once parameter estimation has been completed, updated values will be provided based on samples drawn from the posterior probability distribution.

Classification

The classification consists of four numbers, summing to unity, that give the probability that the source is a *BNS*, *NSBH*, *BBH* merger, or terrestrial (i.e. a background fluctuation or a glitch).

This assumes that terrestrial and astrophysical events occur as independent Poisson processes. A source-dependent weighting of matched-filter templates is used to compute the mean values of expected counts associated with each of these four categories. The mean values are updated weekly based on observed matched-filter count rates. They are then used to predict the category for new triggers uploaded by *search pipelines*.

Properties

The source properties consist of a set of numbers, each between zero and unity, that give the probabilities that the source satisfies certain conditions. These conditions are:

HasNS: At least one of the compact objects in the binary (that is, the less massive or *secondary* compact object) has a mass that is consistent with a neutron star. Specifically, we define this as the probability that the secondary mass satisfies $m_2 \leq 3M_\odot$.

HasRemnant: The probability that the source ejected a nonzero mass outside the final remnant compact object. This is calculated using the disk mass fitting formula from [Foucart et al. 2018](#) (Equation 4).

For preliminary estimates based on the matched-filter pipeline results, the source properties are calculated using a supervised learning technique; mass-dependent rates are the same as those used for the classification. For parameter estimation, updated properties are calculated from posterior samples.

1.3.5 Candidate Vetting

GWCelery orchestrates and supervises performing basic data quality and detector state checks, grouping of events from individual pipelines into *superevents*, initiating automated *localization and parameter estimation*, inferring *classification and source properties*, and sending alerts.

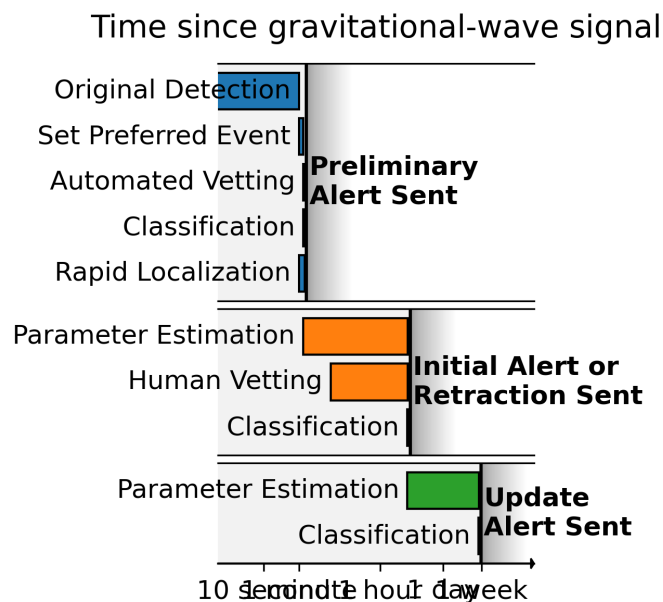
A Data Quality Report is prepared that consists of a semi-automated detector characterization and data quality investigation for each event. It provides a variety of metrics based on auxiliary instrumental and environmental sensors that are used by a rapid response team in order to make a decision of whether to confirm or retract a candidate.

The rapid response team consists of commissioning, computing, and calibration experts from each of the detector sites, pipeline experts, detector characterization experts, and follow-up advocates.

The timeline for distribution of alerts is described below.

1.3.6 Alert Timeline

Here, we described the sequence of the GW alert distributed the Gamma-ray Coordinates Network (*GCN*) via notices and circulars (*Alert content* and *Technical*). Alerts should contain all of the information that is useful for searching for a counterpart.



Within 1–10 minutes after GW trigger time, the first *preliminary notice* will be sent fully autonomously (not necessarily attached to localization). The trigger will be immediately visible under the LIGO/Virgo GW database *GraceDb*. As soon as the localization area is available, a second preliminary notice is sent. The procedure is fully automatic and some preliminary alerts may be retracted after human inspection for data quality, instrumental conditions, and pipeline behavior.

Within 24 hours after the GW trigger time (possibly within 4 hours for *BNS* or *NSBH* sources, to be decided), the *Initial notices and circulars* will be distributed with an update for the sky localization area and the source classification. They are vetted by human instrument scientists and analysts. In case of a binary coalescence including a neutron star or a burst trigger, the initial circular can be labeled as **retracted** (data are unsuitable) or **confirmed**. Note that the initial circular is considered the first LIGO/Virgo publication of a GW candidate, appropriate to cite in publications.

Within a day, black hole mergers will be fully vetted by experts and retraction or confirmation status will be reported.

Update notice and circulars are sent whenever the sky localization area or significance accuracy improves (e.g. as a result of improved calibration, de-glitching, or computationally deeper parameter estimation). Updates will be sent up until the position is determined more accurately by public announcement of an unambiguous counterpart. At which point they will stop until publication of the event.

At any time, we can **promote** an extraordinary candidate that does not pass our public alert thresholds if it is compellingly associated with a multimessenger signal (e.g. GRB, core-collapse SN). In this case, *Initial notices and circulars* will be distributed.

1.4 Alert Contents

Public LIGO/Virgo alerts are distributed using NASA's Gamma-ray Coordinates Network (*GCN*). There are two types of alerts:

GCN Notices are machine-readable packets. They are available as *VOEvent* XML and several other formats. We strongly recommend receiving LIGO/Virgo alerts in the *VOEvent* XML format.

GCN Circulars are short human-readable astronomical bulletins. They are written in a certain well-established format and style. You can subscribe to *GCN Circulars* to receive and post them by email, or you can view them in the public *GCN Circulars* archive.

1.4.1 Notice Types

For each event, there are up to four kinds of GCN Notices:

A **Preliminary GCN Notice** is issued automatically within minutes after a gravitational-wave candidate is detected. The candidate must have passed some automated data quality checks, but it may later be *retracted* after human vetting. There is no accompanying GCN Circular at this stage.

An **Initial GCN Notice** is issued after human vetting (see *Candidate Vetting*). If the signal does not pass human vetting (i.e., it is a glitch), then instead of an initial alert there will be a *retraction*. The initial alert is also accompanied by a GCN Circular which should be considered as the first formal publication of the candidate and can be cited as such.

An **Update GCN Notice** is issued whenever further analysis leads to improved estimates of the source localization, significance, or classification. There may be multiple updates for a given event, and updates may be issued hours, days, or even weeks after the event.

Lastly, a **Retraction GCN Notice** is issued if the candidate is rejected as a result of vetting by human instrument scientists and data analysts. A retraction indicates that the candidate has been withdrawn because it is probably not astrophysical.

All types of GCN Notices *except for Retraction notices* contain the following information, which are described in further detail below:

- *Name*: a unique identifier for the candidate
- *Significance*: estimated false alarm rate
- *Localization*: inferred sky position and (*CBC* candidates only) distance
- *Inference*: inferred source classification and properties (*CBC* candidates only)

All types of GCN Notices *except for Preliminary notices* are accompanied by human-readable GCN Circulars, which restates all of the above information as well as a *data quality assessment*.

1.4.2 Notice Contents

The table below is a diagrammatic representation of the contents of a LIGO/Virgo GCN Notice.

Root		
IVORN	ivo://nasa.gsfc.gcn/LVC#[<i>{T,M}</i>] <i>SYMMDDabc</i> - <i>{1,2,3}</i> - <i>{Preliminary, Initial, Update, Preliminary-Retractio</i>	
Role	<i>{observation, test}</i>	
Who		
Date	Time sent (UTC, ISO-8601), e.g. 2018-11-01T22:34:49	
Author	LIGO Scientific Collaboration and Virgo Collaboration	
WhereWhen	Time of signal (UTC, ISO-8601), e.g. 2018-11-01T22:22:46.654437	
What		
GraceID	GraceDb ID: [<i>{T,M}</i>] <i>SYMMDDabc</i> . Example: MS181101abc	
Packet Type	GCN Notice type: <i>{Preliminary, Initial, Update}</i>	
Notice Type	Numerical equivalent of GCN Notice type: <i>{150, 151, 152}</i>	
FAR	Estimated false alarm rate in Hz	
Sky Map	URL of HEALPix FITS localization file	
Group	CBC	Burst
Pipeline	<i>{Gstlal, MBTAOnline, PyCBC, SPIIR}</i>	<i>{cWB, oLIB}</i>
Central-Freq	N/A	Central frequency in Hz
Duration		Duration of burst in s
Fluence		Gravitational-wave fluence in erg cm^{-2}
BNS, NSBH, BBH, Noise	Probability that the source is a <i>BNS</i> , <i>NSBH</i> , <i>NSBH</i> merger, or terrestrial (i.e., noise) respectively	N/A
HasNS, Has-Remnant	Probability, under the assumption that the source is not noise, that at least one of the compact objects was a neutron star, and that the system ejected a nonzero amount of neutron star matter, respectively.	

Name

The name of an event is its *GraceDb* ID, a uniquely assigned identifier such as MS181101abc. A GraceDb ID has three parts:

- Prefix: S for normal candidates and MS or TS for mock or test events respectively. The S stands for *superevent*.
- Date: The six-digit UTC date of the event consisting of a two-digit year, month, and day of month.
- Suffix: A lowercase alphabetic string that is incremented automatically (a, b, ..., z, aa, ab, ... az, aaa, etc.) whenever a candidate on a given date is added to GraceDb.

Significance

The significance of the event is quantified by its false alarm rate (FAR): the expected rate of events from the pipeline that produced the preferred event with equal or greater significance in the absence of any astrophysical signals.

Localization

The localization consists of the posterior probability distribution of the source's sky position and (for *CBC* events only) luminosity distance. The GCN Notice and Circular will provide a URL for the localization file stored in GraceDb. The

localization is saved in a [FITS](#) file as a [HEALPix](#) all-sky image. See our [sample code](#) for instructions on working with localization files.

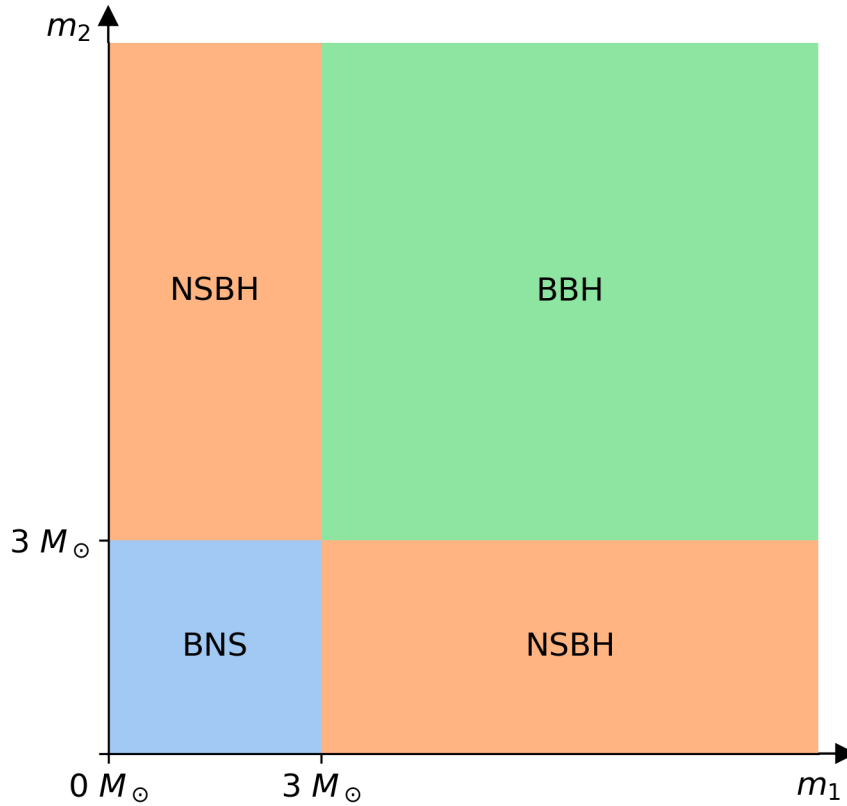
Inference

The inference section is present for [CBC](#) events *only*. It has two parts:

Classification: Four numbers, summing to unity, giving probability that the source belongs to the following four categories:

- [BNS](#) merger
- [NSBH](#) merger
- [BBH](#) merger
- Terrestrial (i.e., a chance background fluctuation or a glitch)

The figure below shows the extent of the three astrophysical categories (BNS, NSBH, and BBH) in terms of the component masses m_1 and m_2 .



Properties: Probabilities that the source has each of the following properties, *assuming that it is not noise* (e.g., assuming that it is a BNS, NSBH, or BBH merger):

- **HasNS:** The mass of one or more of the binary's two companion compact objects is consistent with a neutron star.

- **HasRemnant:** A nonzero amount of neutron star material remained outside the final remnant compact object (a necessary but not sufficient condition to produce certain kinds of electromagnetic emission such as a short GRB or a kilonova).

All of the quantities in the Classification and Properties sections are model dependent to some extent: the Classification section takes into consideration prior knowledge of astrophysical compact binary merger rates from previous LIGO/Virgo observations, and both the Classification and Properties sections depend on details of neutron star physics (e.g. maximum NS mass, equation of state). See the earlier *procedures* for implementation details.

1.4.3 Circular Contents

The following information will be present in the human-readable GCN Circulars.

Data Quality Assessment

Circulars may contain concise descriptions of any instrument or data quality issues that may affect the significance estimates or the GW parameter inferences. Unresolved data quality issues could mean that localization estimates may shift after they have been mitigated, but does not mean that they will. This is to be considered as advisory information.

Localization Ellipse

Generally, GW localizations are irregularly shaped. However, for particularly accurately localized events, the localization region can be well described by an ellipse. For well-localized events, the GCN Circulars will include a 90% containment ellipse in the format of a [DS9 region string](#) (right ascension, declination, semi-major axis, semi-minor axis, position angle of the semi-minor axis), for instance:

```
icrs; ellipse(03h08m25s, -45d08m14s, 9d, 3d, 112d)
```

1.4.4 Not Included in Alerts

The alerts will not contain quantitative estimates of intrinsic properties such as masses and spins, nor contain information on the GW strain or reconstructed waveforms. After final analysis, those data products are released through the Gravitational Wave Open Science Center.

1.4.5 Examples

Below are some sample VOEvents to illustrate the formatting of the GCN Notices.

Preliminary

```
<?xml version="1.0" ?>
<voe:VOEvent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:voe="http://www.ivoa.net/xml/VOEvent/v2.0"
xsi:schemaLocation="http://www.ivoa.net/xml/VOEvent/v2.0 http://www.ivoa.net/xml/
→VOEvent/VOEvent-v2.0.xsd"
version="2.0" role="test" ivorn="ivo://gwnet/gcn_sender#MS181101ab-1-Preliminary">
  <Who>
    <Date>2018-11-01T22:34:49</Date>
    <Author>
      <contactName>LIGO Scientific Collaboration and Virgo Collaboration</
→contactName>
```

(continues on next page)

(continued from previous page)

```

    </Author>
  </Who>
  <What>
    <Param name="internal" dataType="int" value="0">
      <Description>Indicates whether this event should be distributed to LSC/
      ↪Virgo members only</Description>
    </Param>
    <Param name="Packet_Type" dataType="int" value="150">
      <Description>The Notice Type number is assigned/used within GCN, eg
      ↪type=150 is an LVC_PRELIMINARY notice</Description>
    </Param>
    <Param name="Pkt_Ser_Num" dataType="string" value="1"/>
    <Param name="GraceID" dataType="string" value="MS181101ab" ucd="meta.id">
      <Description>Identifier in GraceDB</Description>
    </Param>
    <Param name="AlertType" dataType="string" value="Preliminary" ucd="meta.
    ↪version">
      <Description>VOEvent alert type</Description>
    </Param>
    <Param name="Retraction" dataType="int" value="0">
      <Description>Set to 1 if the event is retracted.</Description>
    </Param>
    <Param name="HardwareInj" dataType="int" value="0" ucd="meta.number">
      <Description>Indicates that this event is a hardware injection if 1, no
      ↪if 0</Description>
    </Param>
    <Param name="OpenAlert" dataType="int" value="1" ucd="meta.number">
      <Description>Indicates that this event is an open alert if 1, no if 0</
    ↪Description>
    </Param>
    <Param name="EventPage" dataType="string" value="https://example.org/
    ↪superevents/MS181101abc/view/" ucd="meta.ref.url">
      <Description>Web page for evolving status of this GW candidate</
    ↪Description>
    </Param>
    <Param name="Instruments" dataType="string" value="H1,L1" ucd="meta.code">
      <Description>List of instruments used in analysis to identify this event</
    ↪Description>
    </Param>
    <Param name="FAR" dataType="float" value="9.11069936486e-14" ucd="arith.rate;
    ↪stat.falsealarm" unit="Hz">
      <Description>False alarm rate for GW candidates with this strength or
      ↪greater</Description>
    </Param>
    <Param name="Group" dataType="string" value="CBC" ucd="meta.code">
      <Description>Data analysis working group</Description>
    </Param>
    <Param name="Pipeline" dataType="string" value="gstlal" ucd="meta.code">
      <Description>Low-latency data analysis pipeline</Description>
    </Param>
    <Param name="Search" dataType="string" value="MDC" ucd="meta.code">
      <Description>Specific low-latency search</Description>
    </Param>
    <Group type="GW_SKYMAP" name="bayestar">
      <Param name="skymap_fits" dataType="string" value="https://emfollow.docs.
      ↪ligo.org/userguide/_static/bayestar.fits.gz" ucd="meta.ref.url">
        <Description>Sky Map FITS</Description>

```

(continues on next page)

(continued from previous page)

```

        </Param>
        <Param name="skymap_png" dataType="string" value="https://emfollow.docs.
↪ligo.org/userguide/_static/bayestar.png" ucd="meta.ref.url">
        <Description>Sky Map image</Description>
    </Param>
</Group>
<Group type="Classification">
    <Description>
        Source classification: binary neutron star (BNS), neutron star-black
↪hole (NSBH), binary black hole (BBH), or terrestrial (noise)
    </Description>
    <Param name="BNS" dataType="float" value="0.95" ucd="stat.probability">
        <Description>Probability that the source is a binary neutron star
↪merger</Description>
    </Param>
    <Param name="NSBH" dataType="float" value="0.01" ucd="stat.probability">
        <Description>Probability that the source is a neutron star - black
↪hole merger</Description>
    </Param>
    <Param name="BBH" dataType="float" value="0.03" ucd="stat.probability">
        <Description>Probability that the source is a binary black hole merger
↪</Description>
    </Param>
    <Param name="Terrestrial" dataType="float" value="0.01" ucd="stat.
↪probability">
        <Description>Probability that the source is terrestrial (i.e., a
↪background noise fluctuation or a glitch)</Description>
    </Param>
</Group>
<Group type="Properties">
    <Description>
        Qualitative properties of the source, conditioned on the assumption
↪that the signal is an astrophysical compact binary merger
    </Description>
    <Param name="HasNS" dataType="float" value="0.95" ucd="stat.probability">
        <Description>Probability that at least one object in the binary has a
↪mass that is less than 3 solar masses</Description>
    </Param>
    <Param name="HasRemnant" dataType="float" value="0.91" ucd="stat.
↪probability">
        <Description>Probability that a nonzero mass was ejected outside the
↪central remnant object</Description>
    </Param>
</Group>
</What>
<WhereWhen>
    <ObsDataLocation>
        <ObservatoryLocation id="LIGO Virgo"/>
        <ObservationLocation>
            <AstroCoordSystem id="UTC-FK5-GEO"/>
            <AstroCoords coord_system_id="UTC-FK5-GEO">
                <Time>
                    <TimeInstant>
                        <ISOTime>2018-11-01T22:22:46.654437</ISOTime>
                    </TimeInstant>
                </Time>
            </AstroCoords>

```

(continues on next page)

(continued from previous page)

```

        </ObservationLocation>
    </ObsDataLocation>
</WhereWhen>
<How>
    <Description>Candidate gravitational wave event identified by low-latency_
↪analysis</Description>
    <Description>H1: LIGO Hanford 4 km gravitational wave detector</Description>
    <Description>L1: LIGO Livingston 4 km gravitational wave detector</
↪Description>
</How>
    <Description>Report of a candidate gravitational wave event</Description>
</voe:VOEvent>

```

Initial

```

<?xml version="1.0" ?>
<voe:VOEvent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:voe="http://www.ivoa.net/xml/VOEvent/v2.0"
xsi:schemaLocation="http://www.ivoa.net/xml/VOEvent/v2.0 http://www.ivoa.net/xml/
↪VOEvent/VOEvent-v2.0.xsd"
version="2.0" role="test" ivorn="ivo://gwnet/gcn_sender#MS181101ab-2-Initial">
    <Who>
        <Date>2018-11-01T22:36:25</Date>
        <Author>
            <contactName>LIGO Scientific Collaboration and Virgo Collaboration</
↪contactName>
        </Author>
    </Who>
    <What>
        <Param name="internal" dataType="int" value="0">
            <Description>Indicates whether this event should be distributed to LSC/
↪Virgo members only</Description>
        </Param>
        <Param name="Packet_Type" dataType="int" value="151">
            <Description>The Notice Type number is assigned/used within GCN, eg_
↪type=151 is an LVC_INITIAL notice</Description>
        </Param>
        <Param name="Pkt_Ser_Num" dataType="string" value="2"/>
        <Param name="GraceID" dataType="string" value="MS181101ab" ucd="meta.id">
            <Description>Identifier in GraceDB</Description>
        </Param>
        <Param name="AlertType" dataType="string" value="Initial" ucd="meta.version">
            <Description>VOEvent alert type</Description>
        </Param>
        <Param name="Retraction" dataType="int" value="0">
            <Description>Set to 1 if the event is retracted.</Description>
        </Param>
        <Param name="HardwareInj" dataType="int" value="0" ucd="meta.number">
            <Description>Indicates that this event is a hardware injection if 1, no_
↪if 0</Description>
        </Param>
        <Param name="OpenAlert" dataType="int" value="1" ucd="meta.number">
            <Description>Indicates that this event is an open alert if 1, no if 0</
↪Description>
        </Param>
        <Param name="EventPage" dataType="string" value="https://example.org/
↪superevents/MS181101abc/view/" ucd="meta.ref.url">

```

(continues on next page)

(continued from previous page)

```

        <Description>Web page for evolving status of this GW candidate</
↪Description>
        </Param>
        <Param name="Instruments" dataType="string" value="H1,L1" ucd="meta.code">
        <Description>List of instruments used in analysis to identify this event</
↪Description>
        </Param>
        <Param name="FAR" dataType="float" value="9.11069936486e-14" ucd="arith.rate;
↪stat.falsealarm" unit="Hz">
        <Description>False alarm rate for GW candidates with this strength or
↪greater</Description>
        </Param>
        <Param name="Group" dataType="string" value="CBC" ucd="meta.code">
        <Description>Data analysis working group</Description>
        </Param>
        <Param name="Pipeline" dataType="string" value="gstlal" ucd="meta.code">
        <Description>Low-latency data analysis pipeline</Description>
        </Param>
        <Param name="Search" dataType="string" value="MDC" ucd="meta.code">
        <Description>Specific low-latency search</Description>
        </Param>
        <Group type="GW_SKYMAP" name="bayestar">
        <Param name="skymap_fits" dataType="string" value="https://emfollow.docs.
↪ligo.org/userguide/_static/bayestar.fits.gz" ucd="meta.ref.url">
        <Description>Sky Map FITS</Description>
        </Param>
        <Param name="skymap_png" dataType="string" value="https://emfollow.docs.
↪ligo.org/userguide/_static/bayestar.png" ucd="meta.ref.url">
        <Description>Sky Map image</Description>
        </Param>
        </Group>
        <Group type="Classification">
        <Description>
        Source classification: binary neutron star (BNS), neutron star-black
↪hole (NSBH), binary black hole (BBH), or terrestrial (noise)
        </Description>
        <Param name="BNS" dataType="float" value="0.95" ucd="stat.probability">
        <Description>Probability that the source is a binary neutron star
↪merger</Description>
        </Param>
        <Param name="NSBH" dataType="float" value="0.01" ucd="stat.probability">
        <Description>Probability that the source is a neutron star - black
↪hole merger</Description>
        </Param>
        <Param name="BBH" dataType="float" value="0.03" ucd="stat.probability">
        <Description>Probability that the source is a binary black hole merger
↪</Description>
        </Param>
        <Param name="Terrestrial" dataType="float" value="0.01" ucd="stat.
↪probability">
        <Description>Probability that the source is terrestrial (i.e., a
↪background noise fluctuation or a glitch)</Description>
        </Param>
        </Group>
        <Group type="Properties">
        <Description>
        Qualitative properties of the source, conditioned on the assumption
↪that the signal is an astrophysical compact binary merger

```

(continues on next page)

(continued from previous page)

```

</Description>
<Param name="HasNS" dataType="float" value="0.95" ucd="stat.probability">
  <Description>Probability that at least one object in the binary has a
↳mass that is less than 3 solar masses</Description>
</Param>
<Param name="HasRemnant" dataType="float" value="0.91" ucd="stat.
↳probability">
  <Description>Probability that a nonzero mass was ejected outside the
↳central remnant object</Description>
</Param>
</Group>
</What>
<WhereWhen>
  <ObsDataLocation>
    <ObservatoryLocation id="LIGO Virgo"/>
    <ObservationLocation>
      <AstroCoordSystem id="UTC-FK5-GEO"/>
      <AstroCoords coord_system_id="UTC-FK5-GEO">
        <Time>
          <TimeInstant>
            <ISOTime>2018-11-01T22:22:46.654437</ISOTime>
          </TimeInstant>
        </Time>
      </AstroCoords>
    </ObservationLocation>
  </ObsDataLocation>
</WhereWhen>
<How>
  <Description>Candidate gravitational wave event identified by low-latency
↳analysis</Description>
  <Description>H1: LIGO Hanford 4 km gravitational wave detector</Description>
  <Description>L1: LIGO Livingston 4 km gravitational wave detector</
↳Description>
</How>
  <Description>Report of a candidate gravitational wave event</Description>
</voe:VOEvent>

```

Update

```

<?xml version="1.0" ?>
<voe:VOEvent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:voe="http://www.ivoa.net/xml/VOEvent/v2.0"
  xsi:schemaLocation="http://www.ivoa.net/xml/VOEvent/v2.0 http://www.ivoa.net/xml/
↳VOEvent/VOEvent-v2.0.xsd"
  version="2.0" role="test" ivorn="ivo://gwnet/gcn_sender#MS181101ab-2-Initial">
  <Who>
    <Date>2018-11-01T22:36:25</Date>
    <Author>
      <contactName>LIGO Scientific Collaboration and Virgo Collaboration</
↳contactName>
    </Author>
  </Who>
  <What>
    <Param name="internal" dataType="int" value="0">
      <Description>Indicates whether this event should be distributed to LSC/
↳Virgo members only</Description>
    </Param>

```

(continues on next page)

(continued from previous page)

```

    <Param name="Packet_Type" dataType="int" value="152">
      <Description>The Notice Type number is assigned/used within GCN, eg_
↪type=151 is an LVC_INITIAL notice</Description>
    </Param>
    <Param name="Pkt_Ser_Num" dataType="string" value="3"/>
    <Param name="GraceID" dataType="string" value="MS181101ab" ucd="meta.id">
      <Description>Identifier in GraceDB</Description>
    </Param>
    <Param name="AlertType" dataType="string" value="Update" ucd="meta.version">
      <Description>VOEvent alert type</Description>
    </Param>
    <Param name="Retraction" dataType="int" value="0">
      <Description>Set to 1 if the event is retracted.</Description>
    </Param>
    <Param name="HardwareInj" dataType="int" value="0" ucd="meta.number">
      <Description>Indicates that this event is a hardware injection if 1, no_
↪if 0</Description>
    </Param>
    <Param name="OpenAlert" dataType="int" value="1" ucd="meta.number">
      <Description>Indicates that this event is an open alert if 1, no if 0</
↪Description>
    </Param>
    <Param name="EventPage" dataType="string" value="https://example.org/
↪superevents/MS181101abc/view/" ucd="meta.ref.url">
      <Description>Web page for evolving status of this GW candidate</
↪Description>
    </Param>
    <Param name="Instruments" dataType="string" value="H1,L1" ucd="meta.code">
      <Description>List of instruments used in analysis to identify this event</
↪Description>
    </Param>
    <Param name="FAR" dataType="float" value="9.11069936486e-14" ucd="arith.rate;
↪stat.falsealarm" unit="Hz">
      <Description>False alarm rate for GW candidates with this strength or_
↪greater</Description>
    </Param>
    <Param name="Group" dataType="string" value="CBC" ucd="meta.code">
      <Description>Data analysis working group</Description>
    </Param>
    <Param name="Pipeline" dataType="string" value="gstlal" ucd="meta.code">
      <Description>Low-latency data analysis pipeline</Description>
    </Param>
    <Param name="Search" dataType="string" value="MDC" ucd="meta.code">
      <Description>Specific low-latency search</Description>
    </Param>
    <Group type="GW_SKYMAP" name="bayestar">
      <Param name="skymap_fits" dataType="string" value="https://emfollow.docs.
↪ligo.org/userguide/_static/bayestar.fits.gz" ucd="meta.ref.url">
        <Description>Sky Map FITS</Description>
      </Param>
      <Param name="skymap_png" dataType="string" value="https://emfollow.docs.
↪ligo.org/userguide/_static/bayestar.png" ucd="meta.ref.url">
        <Description>Sky Map image</Description>
      </Param>
    </Group>
    <Group type="Classification">
      <Description>

```

(continues on next page)

(continued from previous page)

```

Source classification: binary neutron star (BNS), neutron star-black
hole (NSBH), binary black hole (BBH), or terrestrial (noise)
    </Description>
    <Param name="BNS" dataType="float" value="0.95" ucd="stat.probability">
    <Description>Probability that the source is a binary neutron star
merger</Description>
    </Param>
    <Param name="NSBH" dataType="float" value="0.01" ucd="stat.probability">
    <Description>Probability that the source is a neutron star - black
hole merger</Description>
    </Param>
    <Param name="BBH" dataType="float" value="0.03" ucd="stat.probability">
    <Description>Probability that the source is a binary black hole merger
</Description>
    </Param>
    <Param name="Terrestrial" dataType="float" value="0.01" ucd="stat.
probability">
    <Description>Probability that the source is terrestrial (i.e., a
background noise fluctuation or a glitch)</Description>
    </Param>
</Group>
<Group type="Properties">
    <Description>
        Qualitative properties of the source, conditioned on the assumption
that the signal is an astrophysical compact binary merger
    </Description>
    <Param name="HasNS" dataType="float" value="0.95" ucd="stat.probability">
    <Description>Probability that at least one object in the binary has a
mass that is less than 3 solar masses</Description>
    </Param>
    <Param name="HasRemnant" dataType="float" value="0.91" ucd="stat.
probability">
    <Description>Probability that a nonzero mass was ejected outside the
central remnant object</Description>
    </Param>
</Group>
</What>
<WhereWhen>
    <ObsDataLocation>
    <ObservatoryLocation id="LIGO Virgo"/>
    <ObservationLocation>
    <AstroCoordSystem id="UTC-FK5-GEO"/>
    <AstroCoords coord_system_id="UTC-FK5-GEO">
    <Time>
    <TimeInstant>
    <ISOTime>2018-11-01T22:22:46.654437</ISOTime>
    </TimeInstant>
    </Time>
    </AstroCoords>
    </ObservationLocation>
    </ObsDataLocation>
</WhereWhen>
<How>
    <Description>Candidate gravitational wave event identified by low-latency
analysis</Description>
    <Description>H1: LIGO Hanford 4 km gravitational wave detector</Description>
    <Description>L1: LIGO Livingston 4 km gravitational wave detector</
Description>

```

(continues on next page)

(continued from previous page)

```

</How>
<Description>Report of a candidate gravitational wave event</Description>
</voe:VOEvent>

```

Retraction

```

<?xml version="1.0" ?>
<voe:VOEvent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:voe="http://www.ivoa.net/xml/VOEvent/v2.0"
xsi:schemaLocation="http://www.ivoa.net/xml/VOEvent/v2.0 http://www.ivoa.net/xml/
VOEvent/VOEvent-v2.0.xsd"
version="2.0" role="test" ivorn="ivo://gwnet/gcn_sender#MS181101ab-4-Preliminary-
Retraction">
  <Who>
    <Date>2018-11-01T23:36:23</Date>
    <Author>
      <contactName>LIGO Scientific Collaboration and Virgo Collaboration</
contactName>
      </Author>
    </Who>
    <What>
      <Param name="internal" dataType="int" value="0">
        <Description>Indicates whether this event should be distributed to LSC/
Virgo members only</Description>
      </Param>
      <Param name="Packet_Type" dataType="int" value="150">
        <Description>The Notice Type number is assigned/used within GCN, eg
type=150 is an LVC_PRELIMINARY notice</Description>
      </Param>
      <Param name="Pkt_Ser_Num" dataType="string" value="4"/>
      <Param name="GraceID" dataType="string" value="MS181101ab" ucd="meta.id">
        <Description>Identifier in GraceDB</Description>
      </Param>
      <Param name="AlertType" dataType="string" value="Preliminary" ucd="meta.
version">
        <Description>VOEvent alert type</Description>
      </Param>
      <Param name="Retraction" dataType="int" value="1">
        <Description>Set to 1 if the event is retracted.</Description>
      </Param>
      <Param name="HardwareInj" dataType="int" value="0" ucd="meta.number">
        <Description>Indicates that this event is a hardware injection if 1, no
if 0</Description>
      </Param>
      <Param name="OpenAlert" dataType="int" value="1" ucd="meta.number">
        <Description>Indicates that this event is an open alert if 1, no if 0</
Description>
      </Param>
      <Param name="EventPage" dataType="string" value="https://example.org/
superevents/MS181101abc/view/" ucd="meta.ref.url">
        <Description>Web page for evolving status of this GW candidate</
Description>
      </Param>
    </What>
    <WhereWhen>
      <ObsDataLocation>
        <ObservatoryLocation id="LIGO Virgo"/>

```

(continues on next page)

(continued from previous page)

```

    <ObservationLocation>
      <AstroCoordSystem id="UTC-FK5-GEO"/>
      <AstroCoords coord_system_id="UTC-FK5-GEO">
        <Time>
          <TimeInstant>
            <ISOTime>2018-11-01T22:22:46.654437</ISOTime>
          </TimeInstant>
        </Time>
      </AstroCoords>
    </ObservationLocation>
  </ObsDataLocation>
</WhereWhen>
</voe:VOEvent>

```

1.5 Sample Code

This section provides Python sample code for receiving and interacting with *GCN Notices*. GCN Notices are available over several different protocols and in several different formats. LIGO/Virgo strongly recommends using the VOEvent Transport Protocol (*VTP*) to receive notices in *VOEvent* XML format because it is anonymous, configuration-free, and easy to parse.

This tutorial will walk you through writing a Python script to receive and process example LIGO/Virgo GCN notice that are sent every hour (starting around 2018-12-14 16:00 UTC). The tutorial is broken into the following subsections:

1.5.1 Prerequisites

In order to run this sample code, you will need Python ≥ 3.5 on a Unix-like operating system (Linux or macOS) and a few third-party Python packages:

- **PyGCN** for connecting to GCN (alternatives: **comet**)
- **lxml** for parsing *VOEvent* XML packets (see also the **voevent-parse** and **VOEventLib** helper libraries, both of which are based on **lxml**)
- **Healpy** for decoding *HEALPix* coordinates (alternatives: **astropy-healpix**, the official C/C++/Fortran/Java/IDL *HEALPix* bindings for **HEALPix**, **DS9**, **Aladin**)
- **astropy** for astronomical coordinate transformations, observability, etc.
- **numpy** and **matplotlib**, popular math and plotting packages for Python

If you are on a Mac and use the **MacPorts** package manager, you can install all of the above with the following command:

```
$ sudo port install py37-gcn py37-healpy
```

Otherwise, the fastest way to install the dependencies is with **pip**, a package manager that comes with most Python distributions. To install these packages with **pip**, run the following command:

```
$ pip install pygcn healpy
```

Imports

Now we'll write a GCN handler script. First, some imports:

```
import gcn
import gcn.handlers
import gcn.notice_types
import healpy as hp
import numpy as np
```

1.5.2 Receiving GCNs

Next, we'll write a GCN handler function that we want PyGCN to call every time it receives a GCN notice. We decorate the handler with `@gcn.handlers.include_notice_types` to specify that we only want to process certain GCN notice types (`LVC_PRELIMINARY`, `LVC_INITIAL`, and `LVC_UPDATE`).

Events come in two very general flavors: 'CBC' or compact binary coalescence candidates detected by matched filtering, and generic 'Burst' candidates detected by model-independent methods. Your handler can take different actions based on this. The example below will handle only 'CBC' events.

Important: Note that mock or 'test' observations are denoted by the `role="test"` VOEvent attribute. Alerts resulting from real LIGO/Virgo science data will always have `role="observation"`. The sample code below will respond **only** to 'test' events. When preparing for actual observations, you **must remember to switch to 'observation' events**.

Note: Observe in the example below that we do not have to explicitly download the FITS file because the `hp.read_map()` function works with either URLs or filenames. However, you could download and save the FITS file in order to save it locally using `astropy.utils.data.download_file()`, `requests.get()`, `urllib.request.urlopen()`, or even `curl`.

The following basic handler function will parse out the URL of the FITS file, download it, and extract the probability sky map:

```
# Function to call every time a GCN is received.
# Run only for notices of type
# LVC_PRELIMINARY, LVC_INITIAL, or LVC_UPDATE.
@gcn.handlers.include_notice_types(
    gcn.notice_types.LVC_PRELIMINARY,
    gcn.notice_types.LVC_INITIAL,
    gcn.notice_types.LVC_UPDATE)
def process_gcn(payload, root):
    # Respond only to 'test' events.
    # VERY IMPORTANT! Replace with the following code
    # to respond to only real 'observation' events.
    # if root.attrib['role'] != 'observation':
    #     return
    if root.attrib['role'] != 'test':
        return

    # Read all of the VOEvent parameters from the "What" section.
    params = {elem.attrib['name']:
               elem.attrib['value']}
    for elem in root.iterfind('.//Param'):

    # Respond only to 'CBC' events. Change 'CBC' to "Burst"
    # to respond to only unmodeled burst events.
```

(continues on next page)

(continued from previous page)

```

if params['Group'] != 'CBC':
    return

# Print all parameters.
for key, value in params.items():
    print(key, '=', value)

# Read the HEALPix sky map and the FITS header.
skymap, header = hp.read_map(params['skymap_fits'],
                             h=True, verbose=False)

header = dict(header)

# Print some values from the FITS header.
print('Distance =', header['DISTMEAN'], '+/-', header['DISTSTD'])

```

Listen for GCNs

Now, we will start the VOEvent client to listen for GCNs using the `gcn.listen` function. By default, this will connect to the anonymous, public GCN server. You just need to tell `gcn.listen` what function to call whenever it receives an GCN; in this example, that is the `process_gcn` handler that we defined above.

```

# Listen for GCNs until the program is interrupted
# (killed or interrupted with control-C).
gcn.listen(handler=process_gcn)

```

When you run this script, you should receive a sample LIGO/Virgo GCN Notice every hour.

Note: `gcn.listen` will try to automatically reconnect if the network connection is ever broken.

For each sample notice, you should see output that looks like this:

```

internal = 0
Packet_Type = 150
Pkt_Ser_Num = 1
GraceID = MS181101abc
AlertType = Preliminary
Retraction = false
HardwareInj = 0
Vetted = 0
OpenAlert = 1
EventPage = https://example.org/superevents/MS181101abc/view/
Instruments = H1,L1
FAR = 9.11069936486e-14
Group = CBC
Pipeline = gstlal
Search = MDC
skymap_fits = https://emfollow.docs.ligo.org/userguide/_static/bayestar.fits.gz
skymap_png = https://emfollow.docs.ligo.org/userguide/_static/bayestar.png
BNS = 0.95
NSBH = 0.01
BBH = 0.03
Noise = 0.01
HasNS = 0.95

```

(continues on next page)

(continued from previous page)

```
HasRemnant = 0.91
Distance = 141.1453950128411 +/- 39.09548411497191
```

Offline Testing

Sometimes it is convenient to be able to explicitly call the GCN handler with a sample input, rather than waiting for the next broadcast of a sample alert. You can download the [example GCN notices](#) from this documentation and pass it into your GCN handler at any time. First, download the sample GCN notice using curl:

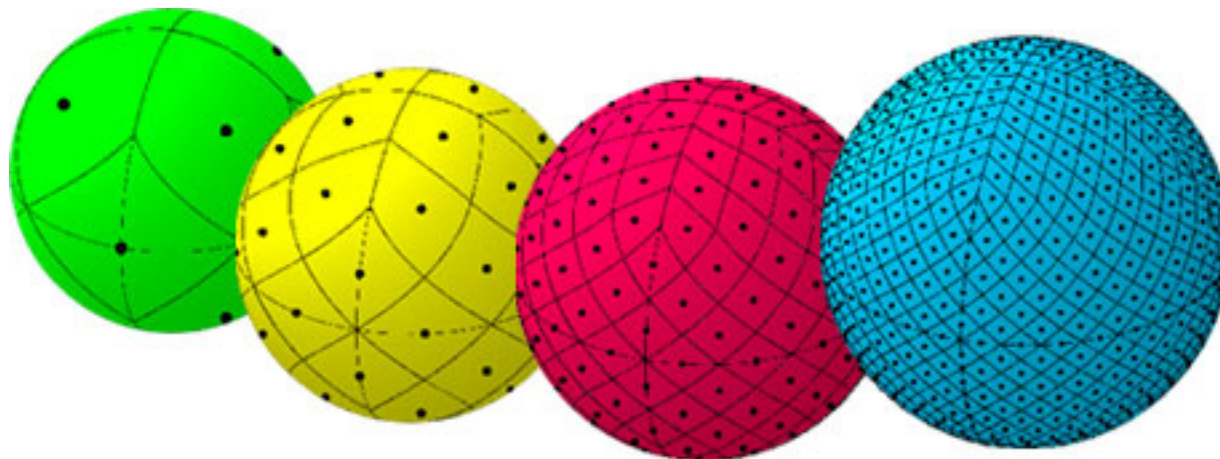
```
curl -O https://emfollow.docs.ligo.org/userguide/_static/MS181101ab-1-Preliminary.xml
```

Then you can manually invoke your GCN handler using this Python code:

```
import lxml.etree
payload = open('MS181101abc-1-Preliminary.xml', 'rb').read()
root = lxml.etree.fromstring(payload)
process_gcn(payload, root)
```

1.5.3 Working with Sky Maps

Let's take a look at what is inside one of the LIGO/Virgo probability sky maps. They are FITS image files and can be manipulated and viewed with many commonplace FITS tools. However, they are a little unusual in two regards. First, since they are all-sky images, they are stored in the [HEALPix](#) projection, a format that is used for [Planck](#) all-sky CMB maps and by Aladin for hierarchical, progressively refined, all-sky survey images ([HiPS](#)). Second, the value stored at each pixel is the probability that the gravitational-wave source is within that pixel.



Let's download an example FITS file with curl:

```
$ curl -O https://emfollow.docs.ligo.org/userguide/_static/bayestar.fits.gz
```

We can look at the metadata inside the FITS file by printing its header with tools like [funhead](#) from [Funtools](#), [imhead](#) from [WCSTools](#), or [fitsheader](#) from [Astropy](#):

```
$ fitsheader bayestar.fits.gz
# HDU 0 in bayestar.fits.gz:
SIMPLE = T / conforms to FITS standard
BITPIX = 8 / array data type
```

(continues on next page)

(continued from previous page)

```

NAXIS      =          0 / number of array dimensions
EXTEND     =          T

# HDU 1 in bayestar.fits.gz:
XTENSION= 'BINTABLE'      / binary table extension
BITPIX    =          8 / array data type
NAXIS     =          2 / number of array dimensions
NAXIS1    =         32 / length of dimension 1
NAXIS2    =       786432 / length of dimension 2
PCOUNT    =          0 / number of group parameters
GCOUNT    =          1 / number of groups
TFIELDS   =          4 / number of table fields
TTYPE1    = 'PROB      '
TFORM1    = 'D        '
TUNIT1    = 'pix-1    '
TTYPE2    = 'DSTMU    '
TFORM2    = 'D        '
TUNIT2    = 'Mpc      '
TTYPE3    = 'DISTSIGMA'
TFORM3    = 'D        '
TUNIT3    = 'Mpc      '
TTYPE4    = 'DISTNORM'
TFORM4    = 'D        '
TUNIT4    = 'Mpc-2    '
PIXTYPE   = 'HEALPIX  ' / HEALPIX pixelisation
ORDERING= 'NESTED    ' / Pixel ordering scheme: RING, NESTED, or NUNIQ
COORDSYS= 'C         ' / Ecliptic, Galactic or Celestial (equatorial)
NSIDE     =         256 / Resolution parameter of HEALPIX
INDXSCHM= 'IMPLICIT  ' / Indexing: IMPLICIT or EXPLICIT
OBJECT    = 'M2052    ' / Unique identifier for this event
REFERENC= 'https://gracedb-playground.ligo.org/events/M2052' / URL of this event
INSTRUME= 'H1,L1     ' / Instruments that triggered this event
DATE-OBS= '2018-11-01T22:22:46.654438' / UTC date of the observation
MJD-OBS   =      58423.93248442614 / modified Julian date of the observation
DATE      = '2018-11-01T22:34:37.000000' / UTC date of file creation
CREATOR   = 'BAYESTAR' / Program that created this file
ORIGIN    = 'LIGO/Virgo' / Organization responsible for this FITS file
RUNTIME   =         11.0 / Runtime in seconds of the CREATOR program
DISTMEAN=  141.1453950128411 / Posterior mean distance (Mpc)
DISTSTD   =  39.09548411497191 / Posterior standard deviation of distance (Mpc)
LOGBCI    =  7.793862946657789 / Log Bayes factor: coherent vs. incoherent
LOGBSN    =  47.28194676827084 / Log Bayes factor: signal vs. noise
VCSVERS   = 'ligo.skymap 0.0.17' / Software version
VCSREV    = 'cb59e5fd04d41c5181ae9e41fe59de232877ddd2' / Software revision (Git)
DATE-BLD= '2018-10-24T20:50:55' / Software build date
HISTORY
HISTORY Generated by calling the following Python function:
HISTORY ligo.skymap.bayestar.localize(event=..., waveform='o2-uberbank', f_low=3
HISTORY 0.0, min_distance=None, max_distance=None, prior_distance_power=None, co
HISTORY smology=False, mcmc=False, chain_dump=None, enable_snr_series=True, f_hi
HISTORY gh_truncate=0.95)
HISTORY
HISTORY This was the command line that started the program:
HISTORY gwcelery worker -l info -n gwcelery-openmp-worker -Q openmp -c 1

```

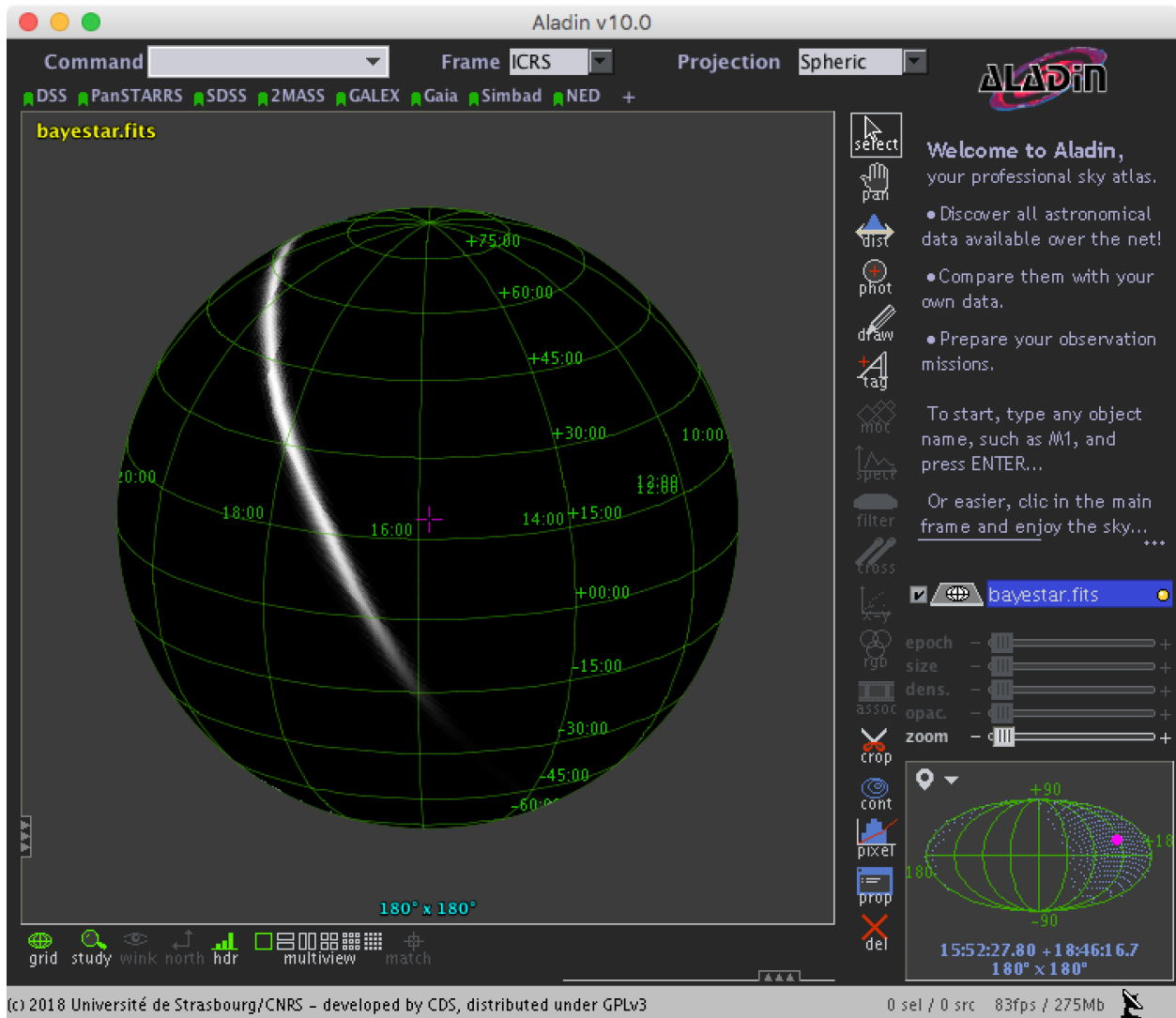
There are several useful pieces of information here:

- COORDSYS=C, telling you that the *HEALPix* projection is in the Celestial (equatorial, ICRS) frame, as all

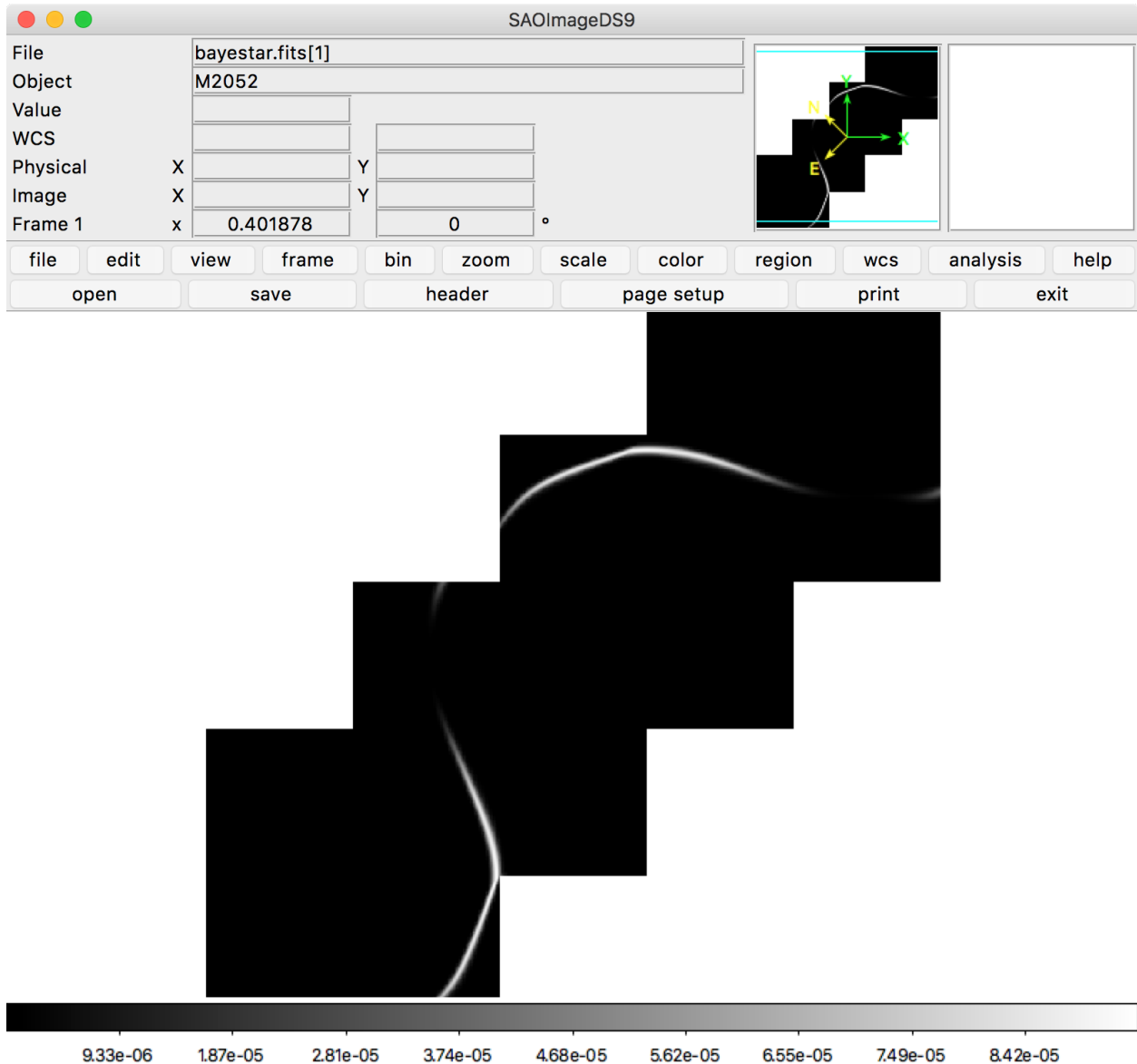
LIGO/Virgo probability sky maps will be.

- **OBJECT**, the unique LIGO/Virgo identifier for the event.
- **REFERENC**, a link to the candidate page in *GraceDb*.
- **INSTRUME**, a list of gravitational-wave sites that triggered on the event: *H1* for LIGO Hanford, *L1* for LIGO Livingston, and *V1* for Virgo.
- **DATE-OBS**, the UTC time of the event. In the case of a compact binary coalescence candidate, this is the time that the signal from the merger passed through the geocenter.
- **MJD-OBS**, same as *DATE-OBS*, but given as a modified Julian day.

You can view the sky map in many common FITS image viewers such as *Aladin*:



or *DS9* (although *DS9* shows HEALPix sky maps in an unusual orientation; see Figure 4 of Calabretta & Roukema (2007) for more information).



Now, let's go through some examples of manipulating HEALPix sky maps programmatically. The [HEALPix](#) project provides official libraries for many languages, including C, C++, Fortran, IDL, and Java. However, since this is a Python tutorial, we are going to demonstrate how to manipulate HEALPix maps with the official Python library, [Healpy](#).

Reading Sky Maps

First, if you have not already downloaded an example sky map, you can do so now by having Python call *curl* on the command line:

```
$ curl -O https://emfollow.docs.ligo.org/userguide/_static/bayestar.fits.gz
```

Next, we need to read in the file in Python with Healpy:

```
>>> hpx = hp.read_map('bayestar.fits.gz')
NSIDE = 256
```

(continues on next page)

(continued from previous page)

```
ORDERING = NESTED in fits file
INDXSCHM = IMPLICIT
Ordering converted to RING
```

You can suppress printing informational messages while loading the file by passing the keyword argument `verbose=False`. You can read both the HEALPix image data and the FITS header by passing the `h=True` keyword argument:

```
>>> hpx, header = hp.read_map('bayestar.fits.gz', h=True, verbose=False)
```

Manipulating HEALPix Coordinates

The image data is a 1D array of values:

```
>>> hpx
array([6.22405744e-25, 1.46981290e-25, 1.94449365e-25, ...,
       2.33147793e-20, 6.78207416e-21, 3.07118068e-22])
```

Healpy has several useful plotting routines including `hp.mollview` for plotting a Mollweide-projection all-sky map:

```
>>> hp.mollview(hpx)
```

Each entry in the array represents the probability contained within a quadrilateral pixel whose position on the sky is uniquely specified by the index in the array and the array's length. Because HEALPix pixels are equal area, we can find the number of pixels per square degree just from the length of the HEALPix array:

```
>>> npix = len(hpx)
>>> sky_area = 4 * 180**2 / np.pi
>>> sky_area / npix
0.052455852825697924
```

The function `hp.pix2ang` converts from pixel index to spherical polar coordinates; the function `hp.ang2pix` does the reverse.

Both `hp.pix2ang` and `hp.ang2pix` take, as their first argument, `nside`, the lateral resolution for the HEALPix map. You can find `nside` from the length of the image array by calling `hp.npix2nside`:

```
>>> nside = hp.npix2nside(npix)
>>> nside
256
```

Let's look up the right ascension and declination of pixel number 123. We'll call `hp.pix2ang` to get the spherical polar coordinates (θ, ϕ) in radians, and then use `np.rad2deg` to convert these to right ascension and declination in degrees.

```
>>> ipix = 123
>>> theta, phi = hp.pix2ang(nside, ipix)
>>> ra = np.rad2deg(phi)
>>> dec = np.rad2deg(0.5 * np.pi - theta)
>>> ra, dec
(129.375, 88.5380288373519)
```

Let's find which pixel contains the point RA=194.95, Dec=27.98.

```
>>> ra = 194.95
>>> dec = 27.98
>>> theta = 0.5 * np.pi - np.deg2rad(dec)
>>> phi = np.deg2rad(ra)
>>> ipix = hp.ang2pix(nside, theta, phi)
>>> ipix
208938
```

Most Probable Sky Location

Let's find the highest probability pixel. What is the probability inside it?

```
>>> ipix_max = np.argmax(hpx)
>>> hpx[ipix_max]
9.35702310989353e-05
```

Where is the highest probability pixel on the sky? Use `hp.pix2ang`.

```
>>> theta, phi = hp.pix2ang(nside, ipix_max)
>>> ra = np.rad2deg(phi)
>>> dec = np.rad2deg(0.5 * np.pi - theta)
>>> ra, dec
(90.87890625, -40.620185190672686)
```

Integrated Probability in a Circle

How do we find the probability that the source is contained within a circle on the sky? First we find the pixels that are contained within the circle using `hp.query_disc`. Note that this function takes as its arguments the Cartesian coordinates of the center of the circle, and its radius in radians. Then, we sum the values of the HEALPix image array contained at those pixels.

First, we define the RA, Dec, and radius of circle in degrees:

```
>>> ra = 213.22
>>> dec = -37.45
>>> radius = 3.1
```

Then we convert to spherical polar coordinates and radius of circle in radians:

```
>>> theta = 0.5 * np.pi - np.deg2rad(dec)
>>> phi = np.deg2rad(ra)
>>> radius = np.deg2rad(radius)
```

Then we calculate the Cartesian coordinates of the center of circle:

```
>>> xyz = hp.ang2vec(theta, phi)
```

We call `hp.query_disc`, which returns an array of the indices of the pixels that are inside the circle:

```
>>> ipix_disc = hp.query_disc(nside, xyz, radius)
```

Finally, we sum the probability in all of the matching pixels:

```
>>> hpx[ipix_disc].sum()
9.522375325439142e-06
```

Integrated Probability in a Polygon

Similarly, we can use the `hp.query_polygon` function to look up the indices of the pixels within a polygon (defined by the Cartesian coordinates of its vertices), and then compute the probability that the source is inside that polygon by summing the values of the pixels.

```
>>> xyz = [[-0.69601758, -0.41315628, -0.58724902],
...        [-0.68590811, -0.40679797, -0.60336181],
...        [-0.69106913, -0.39820114, -0.60320752],
...        [-0.7011786 , -0.40455945, -0.58709473]]
>>> ipix_poly = hp.query_polygon(nside, xyz)
>>> hpx[ipix_poly].sum()
3.935524328237466e-11
```

These are all of the HEALPix functions from `Healpy` that we will need for the remainder of this tutorial.

Other useful `Healpy` functions include `hp.ud_grade` for upsampling or downsampling a sky map and `hp.get_interp_val` for performing bilinear interpolation between pixels. See the [Healpy tutorial](#) for other useful operations.

1.5.4 Basic Observability Calculations

Now we are going to teach our GCN handler how to determine whether a gravitational-wave event is observable. We are going to use the `astropy.coordinates`. (See also the [Astropy example on observation planning in Python](#).) First, we will need to import a few extra Python modules:

```
import astropy.coordinates
import astropy.time
import astropy.units as u
```

The LIGO/Virgo probability sky maps are always in equatorial coordinates. Once we have looked up the coordinates of the HEALPix pixels, we will use `Astropy` to transform those coordinates to an alt/az frame for a particular site on the Earth at a particular time. Then we can quickly determine which pixels are visible from that site at that time, and integrate (sum) the probability contained in those pixels.

Note: You may want to do something more sophisticated like determine how much of the probability is visible for at least a certain length of time. This example will illustrate one key function of HEALPix (looking up coordinates of the grid with `hp.pix2ang`) and some of the key positional astronomy functions with `Astropy`. For more advanced functionality, we recommend the [astroplan](#) package.

```
def prob_observable(m, header):
    """
    Determine the integrated probability contained in a gravitational-wave
    sky map that is observable from a particular ground-based site at a
    particular time.

    Bonus: make a plot of probability versus UTC time!
    """

    # Determine resolution of sky map
    npix = len(m)
    nside = hp.npix2nside(npix)

    # Get time now
```

(continues on next page)

(continued from previous page)

```

time = astropy.time.Time.now()
# Or at the time of the gravitational-wave event...
# time = astropy.time.Time(header['MJD-OBS'], format='mjd')
# Or at a particular time...
# time = astropy.time.Time('2015-03-01 13:55:27')

# Geodetic coordinates of observatory (example here: Mount Wilson)
observatory = astropy.coordinates.EarthLocation(
    lat=34.2247*u.deg, lon=-118.0572*u.deg, height=1742*u.m)

# Alt/az reference frame at observatory, now
frame = astropy.coordinates.AltAz(obstime=time, location=observatory)

# Look up (celestial) spherical polar coordinates of HEALPix grid.
theta, phi = hp.pix2ang(inside, np.arange(npix))
# Convert to RA, Dec.
radecs = astropy.coordinates.SkyCoord(
    ra=phi*u.rad, dec=(0.5*np.pi - theta)*u.rad)

# Transform grid to alt/az coordinates at observatory, now
altaz = radecs.transform_to(frame)

# Where is the sun, now?
sun_altaz = astropy.coordinates.get_sun(time).transform_to(altaz)

# How likely is it that the (true, unknown) location of the source
# is within the area that is visible, now? Demand that sun is at
# least 18 degrees below the horizon and that the airmass
# (secant of zenith angle approximation) is at most 2.5.
prob = m[(sun_altaz.alt <= -18*u.deg) & (altaz.secz <= 2.5)].sum()

# Done!
return prob

```

Finally, we need to update our GCN handler to call this function:

```

@gcn.handlers.include_notice_types(
    gcn.notice_types.LVC_PRELIMINARY,
    gcn.notice_types.LVC_INITIAL,
    gcn.notice_types.LVC_UPDATE)
def process_gcn(payload, root):
    # Respond only to 'test' events.
    # VERY IMPORTANT! Replace with the following line of code
    # to respond to only real 'observation' events.
    # if root.attrib['role'] != 'observation':
    #     return
    if root.attrib['role'] != 'test':
        return

    # Respond only to 'CBC' events. Change 'CBC' to 'Burst'
    # to respond to only unmodeled burst events.
    if root.find("./Param[@name='Group']").attrib['value'] != 'CBC':
        return

    skymap_url = root.find("./Param[@name='skymap_fits']").attrib['value']

    skymap, header = hp.read_map(skymap_url, h=True, verbose=False)

```

(continues on next page)

(continued from previous page)

```
prob = prob_observable(skymap, header)
print('Source has a {:d}% chance of being observable now'.format(
    int(round(100 * prob))))
if prob > 0.5:
    pass # FIXME: perform some action
```

Let's run the new GCN handler now...

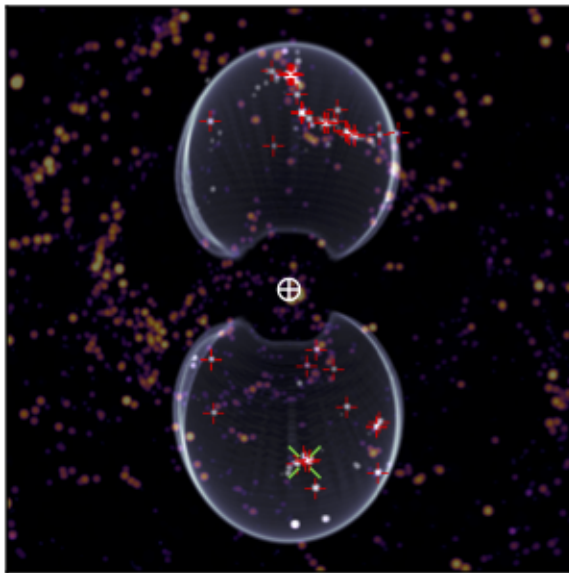
```
# Listen for GCNs until the program is interrupted
# (killed or interrupted with control-C).
gcn.listen(handler=process_gcn)
```

When you run this script, each time you receive a sample LIGO/Virgo GCN Notice, it will print something like the following (note that probability will change as a function of time):

Source has a 76% chance of being observable now

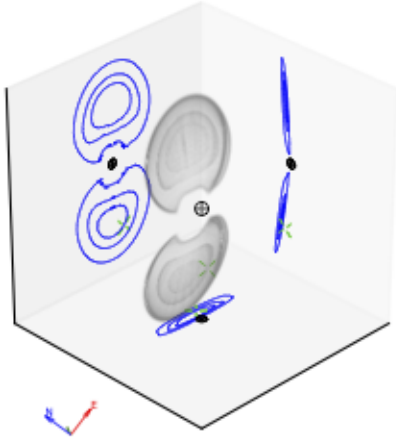
1.5.5 Distance and 3D Sky Maps

All localization FITS files for CBC events are three dimensional: they include both the sky probability map and a directionally dependent distance estimate. This can be useful for identifying possible host galaxies using a galaxy redshift catalog. For further details see¹ and for sample code see².



¹ Singer, L. P., Chen, H.-Y., Holz, D. E., et al. 2016, *Astropys. J. Lett.*, 829, L15. <https://doi.org/10.3847/2041-8205/829/1/L15>

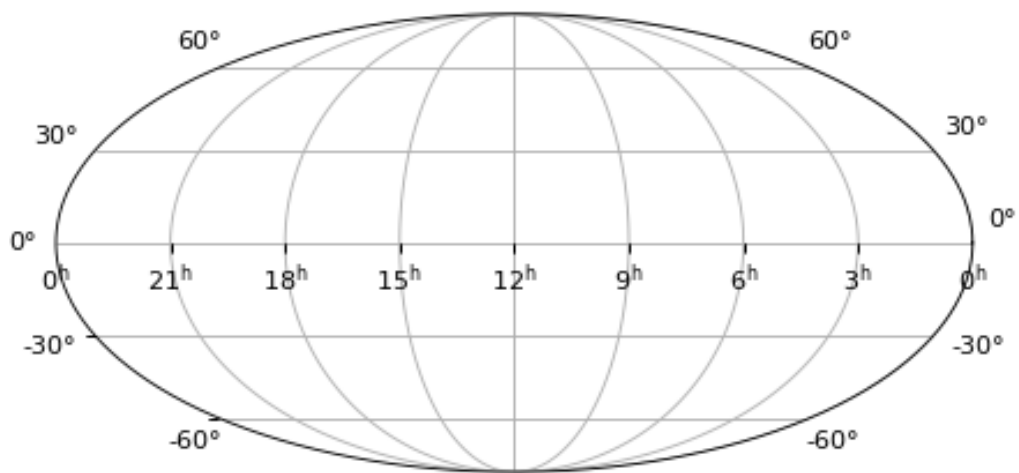
² Singer, L. P., Chen, H.-Y., Holz, D. E., et al. 2016, *Astropys. J. Supp.*, 226, 10. <https://doi.org/10.3847/0067-0049/226/1/10>



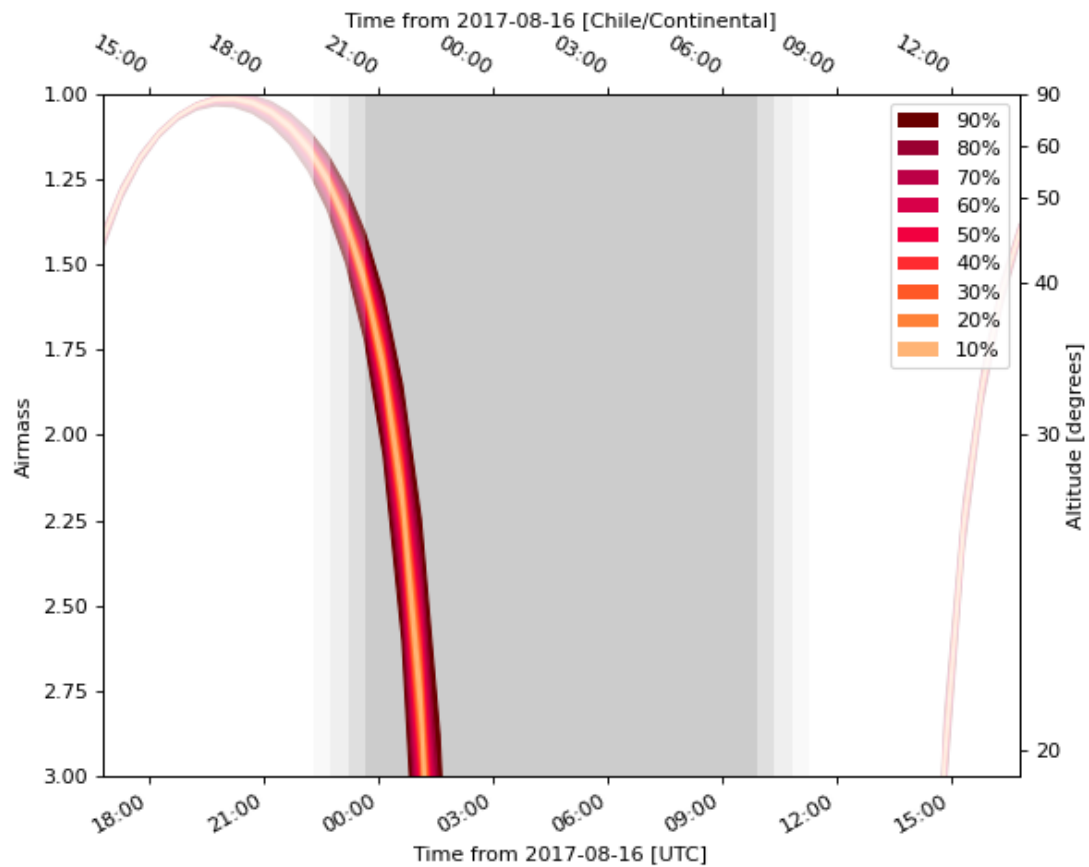
1.5.6 Additional Tools

The `ligo.skymap` package includes a number of advanced tools for working with GW probability sky maps.

- Publication-quality astronomical mapmaking built on Astropy (`ligo.skymap.plot.allsky`)



- Functions for manipulating distance posteriors (`ligo.skymap.distance`)
- Probabilistic airmass plots (`ligo-skymap-plot-airmass`)



- The rapid localization code [BAYESTAR](#), which is used to produce the initial sky maps for CBC events but can also be used to create [simulated localizations](#).
- The postprocessing tool that creates updated sky maps from MCMC samples ([ligo-skymap-from-samples](#))

2.1 Change Log

2.1.1 Version 2 (2018-12-13)

Alert Contents

- Removed the `Vetted` parameter from GCN Notices. It was intended to indicate whether the event had passed human vetting. However, it was redundant because by definition Preliminary events have not been vetted and all Initial and Update alerts have been vetted.
- The type of the `Retraction` parameter in the GCN Notices was changed from a string (`false` or `true`) to an integer (0 or 1) for consistency with other flag-like parameters.
- Remove the `units` attribute from parameters that are not numbers.

Sample Code

- GCN has now begun publicly broadcasting sample LIGO/Virgo GCN Notices. Updated the sample code accordingly with instructions for receiving live sample notices.

2.1.2 Version 1 (2018-11-27)

Getting Started Checklist

- Updated instructions for joining the OpenLVEM Community.

Observing Capabilities

- Changed the expected number of BNS events in O3 from 1-50, as stated in the latest version of the Living Review, to 1-10 events, as stated in the more recent rates presentation.

Alert Contents

- In the example VOEvents, moved the Classification and Inference quantities from the <Why> section to the <What> section so that they validate against the VOEvent 2.0 schema.

2.2 Glossary

BBH Binary black hole system, a binary system composed of two black holes.

BNS Binary neutron star, a binary system composed of two neutron stars.

CBC Compact binary coalescence.

FITS Flexible Image Transport System, a format for astronomical tables, images, and multidimensional data sets.

LHO LIGO Hanford Observatory (see [LHO observatory home page](#)), site of a 4 km gravitational-wave detector in Hanford, Washington, USA.

LLO LIGO Livingston Observatory (see [LLO observatory home page](#)), site of a 4 km gravitational-wave detector in Livingston, Louisiana, USA.

GCN The Gamma-ray Coordinates Network (<https://gcn.gsfc.nasa.gov>), a portal for discoveries and observations of astronomical transients. Historically, GCN has served high-energy satellites but now also other electromagnetic wavelengths and also gravitational-wave, cosmic ray, and neutrino facilities.

GCN Circular A human-readable astronomical bulletin distributed through [GCN](#).

GCN Notice A machine-readable alert distributed through [GCN](#).

Gracedb Gravitational Wave Candidate Event Database (<https://gracedb.ligo.org>), the official public marshal portal for LIGO/Virgo candidates.

HEALPix Hierarchical Equal Area isoLatitude Pixelation, a scheme for indexing positions on the unit sphere.

NSBH Neutron star black hole, a binary system composed of one neutron star and one black hole.

Virgo Virgo Observatory (see [Virgo observatory home page](#)), site of a 3 km gravitational-wave detector in Cascina, Italy.

VOEvent An XML format for describing astronomical transients. For the specification, see the official [VOEvent IVOA Recommendation](#).

VTP VOEvent Transport Protocol, a simple TCP-based protocol for sending and receiving VOEvents, used by [GCN](#). For the specification, see the official [VTP IVOA recommendation](#).

- [Change Log](#)
- [Glossary](#)
- [search](#)
- [Report issues](#)

B

BBH, [38](#)
BNS, [38](#)

C

CBC, [38](#)

F

FITS, [38](#)

G

GCN, [38](#)
GCN Circular, [38](#)
GCN Notice, [38](#)
GraceDb, [38](#)

H

HEALPix, [38](#)

L

LHO, [38](#)
LLO, [38](#)

N

NSBH, [38](#)

V

Virgo, [38](#)
VOEvent, [38](#)
VTP, [38](#)