Low latency: now and future challenges

Steven Schramm

ET EIB / ET-PP WP8 workshop @ Napoli

July 8, 2024

Introduction

- Sara gave an excellent overview of the current status at the Aachen workshop (03/2023)
    - I will be pulling from her slides [link] to explain what we have now
    - In case of questions on these parts, she surely knows more than me!
- The main focus of this talk will instead be on what we expect to change
    - Some of this comes from discussions on my past EIB Div4 presentations
    - Other points come from the joint OSB/EIB MM meeting in Orsay (11/2023) [link]
    - Thanks also to Stefano for his excellent big picture talk at Maastricht (05/2024) [link]
- My goal is not to provide solutions, but rather to identify use cases/requirements
    - Lots of time between now at the start of ET LL alerts
    - The best architecture/solution for the job is likely to evolve
    - MM science is a young and growing field; even the requirements are likely to evolve!

The purpose

1. **Disseminate public alerts** of transient GW (and MMA involving GW) detections.

2. **Enable the discovery of EM and neutrino counterparts** to GWs (and vice versa) and assist the (common) source characterisation.

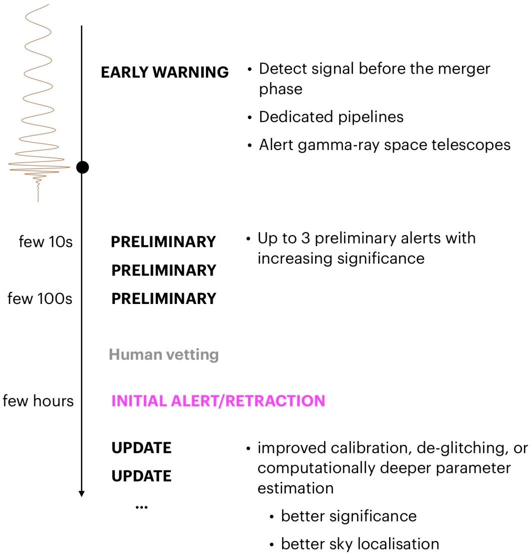
3. **Provide feedback to the instrument** teams by facilitating the diagnosing of detector problems via real-time analyses.

The purpose

- All of these purposes remain relevant, but scope may change, and new topics may emerge
- Recall that the volume of (true) signals is expected to grow enormously
  - Currently $\sim$100/year, less than 1/day
  - Future $10^5 - 10^6$ BNS/year, and $10^5 - 10^6$ BBH/year: every few min – few per min
- This has a huge implication on the demands placed on the low latency infrastructure
  - MM alerts (MMA) changes from rare to semi-constant (more on next slide)

MMA timelines (current)

- Even with the current model, alerts become ∼constant, as multiple stages
  - Preliminary alerts per signal from sub-minute to few minutes
- However, real change is early warning
  - As many events as possible should be detected well in advance of merger
  - Possible in theory: BNS in-band for hours, BBH for minutes
  - The earlier warning the better: gives MM community time to react/ignore
- Infrastructure needs to support a dramatic increase in early warning MMA
- Need to automate whenever possible: human vetting won't scale to ET MMA

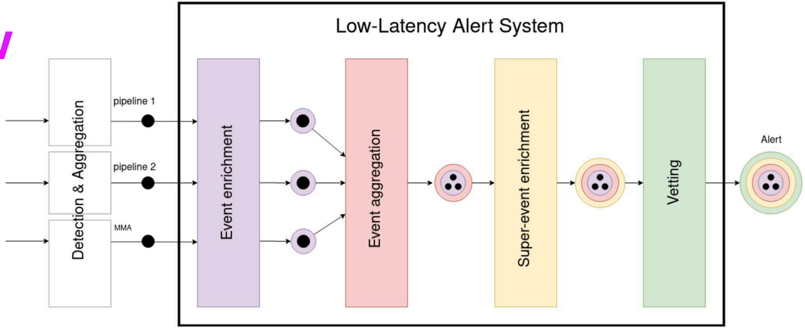


Overlapping signals

- Beyond early warnings, the other major change is that signals are no longer isolated
- For offline users, impact is mitigated: focus on the merger phase of the waveform
  - Even if there are many signals in-band, the merger phase is very short
  - Not impossible to have two overlapping merging signals, but much less common
  - Three or more all merging at the "same time" should be extremely rare
- All of this breaks in the context of early warning MMA, and can break dramatically
  - Example: consider two signals, s_1 and s_2, merging at t_1 and t_2, with $\Delta(t_1, t_2) = \Delta t$
  - Let t_2^A be the time we want to alert about s_2
  - Let the two signals be of similar SNR, and assume there are only two signals (simplification!)
  - $t_1 < t_2^A < t_2$: not many issues, as s_1 ringdown is quickly done, and only s_2 is in-band
  - $t_1 \approx t_2^A < t_2$: unlikely to see s_2 under the merger peak of t_1, so alert is delayed/hidden
  - $t_2^A < t_1 < t_2$: the signal from s_1 should be larger than s_2; may prevent early warning of s_2
  - Cannot let t_1 be a barrier for MMA s_2; need earlier pre-merger warnings for MM community
- Given BNS are in-band so long, early warnings reliant upon overlapping signal handling
  - Algorithmic challenge, but has infrastructure implications
  - Likely implies significant increase in computing requirements for LL MMA early warnings

Workflow

- GW data
- EM data
- Neutrino data



- **All-sky searches:**
  - no assumption on sky location or time of a transient
  - Compact Binary Coalescence: modelled, matched-filtering
  - Bursts: no assumption on signal morphology, time-frequency analysis
  - no information from non GW sources
  - also low significance alerts for early warning alerts (pre-merger)
- **Multi-messenger searches:**
  - integrate information from EM or neutrino external triggers
  - can be targeted to a region of the sky or a time identified by the external trigger (medium-latency)

- **Event aggregation:**
  - grouping of events related to the same astrophysical (or not) cause into a *superevent*
  - combining triggers across multiple pipelines but also from a given pipeline

- **Superevent enrichment:**
  - source classification
  - sky localisation
  - data quality
  - ...
- **Vetting:**
  - human or automatic decision concerning the publication or retraction of the GW detection
  - based on the enriched information available for the *superevent*

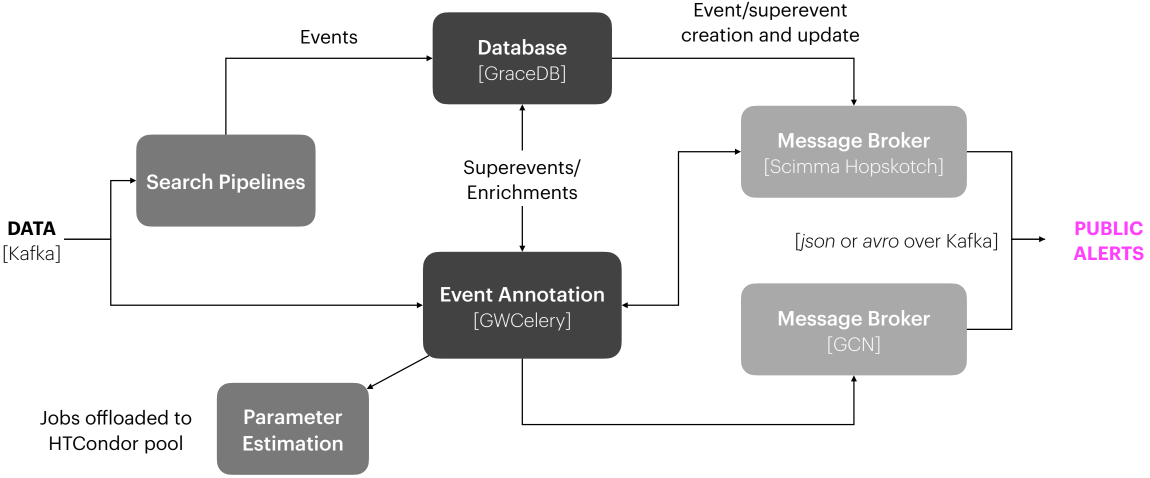
Reacting to external alerts

- To first order, reacting to external alerts seems to no longer be relevant
  - If you pre-warn all signals, then what would you need to react to?
- However, given sufficient pre-merger reliability, a new scenario could be envisioned
  - An optical telescope may detect that some system is approaching a critical state
  - Expensive to use a dedicated optical telescope time to keep watching for the event
  - In contrast, GW observatories have full-sky sensitivity (to first order)
  - External facilities could thus "trigger" GW to implement a sky-localised pipeline for days/weeks/months, and resume collecting data following that targeted GW MMA
  - Has implications on MMA ingress interfaces, supported methods for reacting to external triggers, LL pre-merger sensitivity "guarantees", observatory duty cycle, and more
- Moreover, the existing case remains: there can always be sub-threshold signals
  - External triggers can thus still help to identify/refine GW MMA in some cases

Natively multimodal analyses in LL

- Beyond external triggers, we may also consider natively multimodal analyses
  - Triggering is the first step down the path to common pipelines
  - Motivations could include sub-threshold signal detection, guiding MM community follow-ups (depending on observed transient properties), and direct science measurements
- Before such natively multimodal analyses exist in ET LL, requires significant thought
  - Data access paradigms / MoUs between collaborations?
    - What if the other RI has public data access model?
  - Streaming raw data between RIs, or some higher-level LL data?
  - How are new algorithms integrated into the LL system?
    - Proposals to be reviewed by some committee? Is the code also reviewed? How are computing resources provided/accounted? What if new architecture changes are needed to support it?

High-level architecture

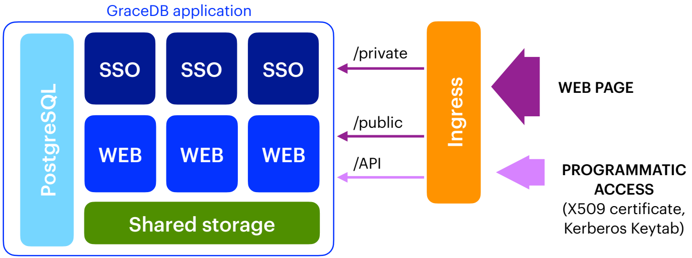


Overall architecture

- The current architecture works, but is already sub-optimal
  - Rare events $\implies$ requires the ability to handle very short spikes in computing demands
    - In the future, this will become $\sim$constant; the infrastructure and architecture need to scale
  - Database is being used both as source of truth (FSM) and public-facing interface
    - Introduces latency: enrichment is a reaction to database updates, not direct process
    - Inefficient for DB to trigger job queue, which then updates database
  - Services are deployed as HTCondor jobs on a single machine
  - System essentially spends $10-15$ second latency just waiting for pieces to interact
- There is a clear need for an effort to design a new architecture already now
  - Already tried for O3 to O4 but not possible, maybe trying for O5
  - Could have a significant impact on ET if this defines a new standard for IGWN
  - Need to make sure that new architecture can scale well to ET (requirement definitions), acknowledging the implementation may evolve as new industry standards are developed

GraceDB: the database

Primary Authentication (Shibboleth):
- managing federated identities and providing a single sign-on (SSO) portal
- uses a metadata provider to collect user attributes from an attribute authority and put them into the user's session



Web application:
- **Django application:** GraceDB is written in Python and is constructed around the Django web framework.
- **Backend Webserver (Gunicorn):** Gunicorn is a lightweight Python webserver which interfaces directly with the Django service via the WSGI protocol.
- **Frontend Webserver (Apache):** used in concert with Gunicorn as an interface with Shibboleth. It is configured as a reverse proxy which gets authentication information from Shibboleth, sets that information in the headers, and then passes it on to Gunicorn.
- **Igwn-alert Overseer:** registers new events in Scimma

Shared storage:
- save event enrichment files (i.e. skymaps)
- relies on Amazon Elastic Block Store

Deployed with Docker Swarm on AWS.

Ingress:
- Traefik
- Redirects to different application components (Linux containers) according to url path

Database evolution

- GraceDB works right now, but unclear if it's the right solution for ET
  - Preliminary feeling of experts I have spoken with seems to point towards GraceDB not scaling
  - Need to go through a requirement-gathering phase to define critical ET uses
- At least the public-facing database and FSM should be split, as the previous slide

- Looking forward, we will need more interactive databases
  - RI-specific databases vs central MMA databases for cross-correlation results
  - Ideally, different instances of the same database implementation: improve harmonisation, communication, maintenance, etc
    - Each RI retains public/private division of course, central database is just the public-facing part
  - Alternatively, there should at least be a common interface and standard alert format
- Database may also need resource monitoring/similar entries
  - For GW: keep track of allocated resources per natively multimodal pipeline, etc
  - For EM: keep track of reactions to targets of opportunity (what did they interrupt to do so, what alert motivated them to do so, on whose time share was the time charged, etc)

GWCelery: the event annotation service

https://rtd.igwn.org/projects/gwcelery/en/latest/index.html

- package for annotating and orchestrating LIGO/Virgo alerts
- built on the Celery distributed task queue and widely used opensource components
- different deployments (tiers), connected to corresponding GraceDB instances
- uses Redis to route and distribute Celery task messages and to store task results for later retrieval
- runs on dedicated VMs that also host Redis and offloads computing intensive tasks to a HTCondor cluster

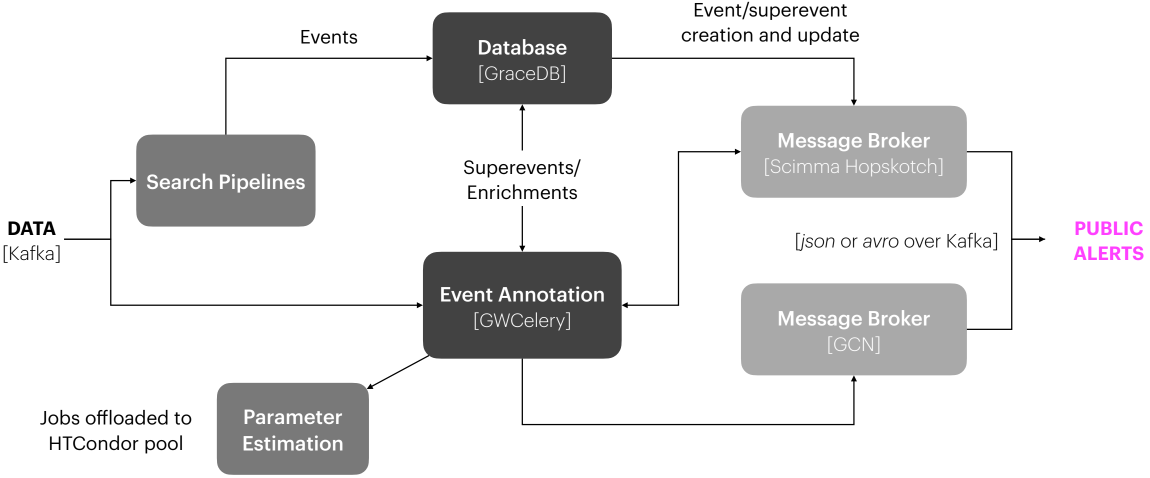
- **GWCelery's responsibilities include:**
  - merging related candidates from multiple online LIGO/Virgo transient searches into *superevents*
  - correlating LIGO/Virgo events with gamma-ray bursts, neutrinos, and supernovae
  - launching automated follow-up analyses including data quality checks, rapid sky localisation, automated parameter estimation, and source classification
  - generating and sending preliminary and updated public Notices (machine readable)
  - automatically composing public Circulars (human readable)

Data enrichment

- Enrichment will likely become a constant task, with large resource demands
  - Separating all of the overlapping events and running PE for pre-merge pipelines
  - As merger approaches, likely more detailed PE, refining the sky localisation
  - More precise PE also needed beyond sky localisation
    - Reasonably precise eccentricity measures or other PE results may be key to having specific events stand out for follow-up observation

- Exactly what we make public is beyond the scope of this talk
  - Started the discussion at the joint OSB/EIB meeting, and the answer is very unclear
- However, we have to be prepared to support different scenarios
  - If we can make full PE public (highly unlikely), less relevant
  - If not, we may need to be able to flag events for specific RIs, following their requirements

High-level architecture



MMA brokers and MMA cross-correlations

- Right now, GW MMA are rare, thus little need for a variety of brokers
- In the future, this may completely change
  - Vera Rubin is already making this clear: a variety of brokers with different objectives
- Moreover, as MMA becomes common, the community will need more guidance
  - Following-up on every MMA is unfeasible (excluding possible MMA-dedicated facilities)
  - Instead, most users will want filtering of MMA via key parameters, cross-correlation, etc
- Brokers and cross-correlations
  - There are already several brokers for a single observatory (Vera Rubin) as it will produce so many transient alerts that brokers generally target only a subset of science cases
  - Add ET and other future MM facilities, and broker multiplicity may explode
  - Cross-correlations of different facilities is also likely to become an expectation
    - Probably this will have to be done "centrally" by RIs sharing/pleding resources, rather than by independent brokers each re-calculating the correlation sky maps etc
  - Brokers could then filter that correlation info to support different communities
- Cross-correlations and filtering would have to be LL to support community MMA reactions

Low latency computing infrastructure

- Many of the points I mentioned involved a significant scaling up of LL resources
  - More transient signals, handling overlapping signals, more parameter estimation, etc
- This brings up the question of what resources will run LL jobs
  - One site per RI? One site for global GW? One site for global MM? Distributed?
  - If one site per RI, is it at/close to the RI, or external?
    - External adds latency, but if early warnings have a sufficient lead time, should be negligible
  - In the future, LL should be a “constant” resource consumer; maybe implies switch to dedicated resources (reminder: right now it’s dynamic, growing in response to a signal)
    - Does not have to be RI-owned resources, rather something that is reserved instead of dynamic
- What hardware will be needed for LL jobs?
  - Right now, CPU is king, but there is a strong (external) push to leave that paradigm behind
  - Could switch to GPU/FPGA/etc implementations of existing algorithms
  - Could also switch to AI/ML approaches, and GPU/FPGA/TPU/... as appropriate in 2035+

Summary

- The low latency system requirements will evolve considerable from right now to ET
- Those requirements will likely continue to evolve during ET
  - Different external MM facilities and user communities
  - Specific types of transient signals may become more or less relevant with time
- Already now, the low latency system could benefit from a redesign
  - Important to define the requirements, for now and for ET, before redesigning
  - Then we can see how to design a new system that can handle the requirements, and which can evolve with the requirements and with industry standards/technology updates
- LL is not only MMA (also prompt feedback on detector issues), but MMA is a major part
  - It is thus important to include the MM community in such planning: other RIs, users, etc
    - We need to provide a public-facing interface that is both usable and useful
  - Ideally, MM community would develop common database/interfaces/alert format/standards
- Everything changes with time: we must be prepared to adapt to the unforeseen