LIGO

Streaming Data for the International Gravitational Wave Detector Network

CI Compass Virtual Workshop
Data Management: From Instrument to First Storage
January 2025

Jameson Rollins
LIGO Caltech

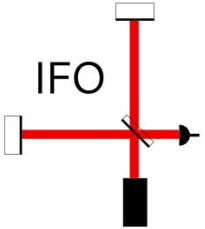
LIGO's controls and data architecture

The LIGO project is undertaking a complete overhaul of its data distribution and stream processing infrastructure.

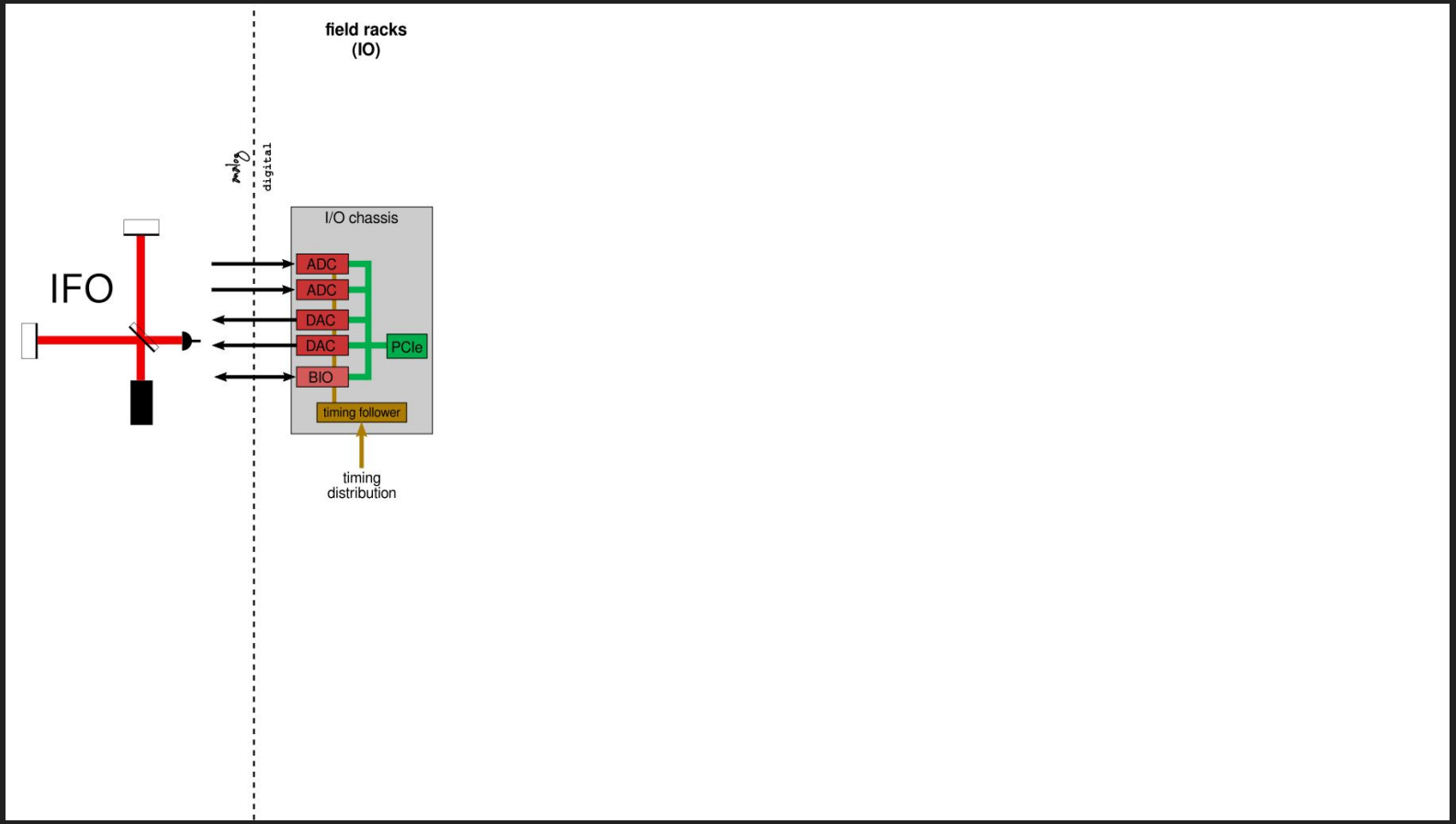
This reflects the continuing evolution of the field's priorities towards *transient event discovery*.

The new effort on this front actually starts just downstream of our "first storage". But let's first take a look at LIGO's existing controls and data acquisition architecture that will feed this new system…

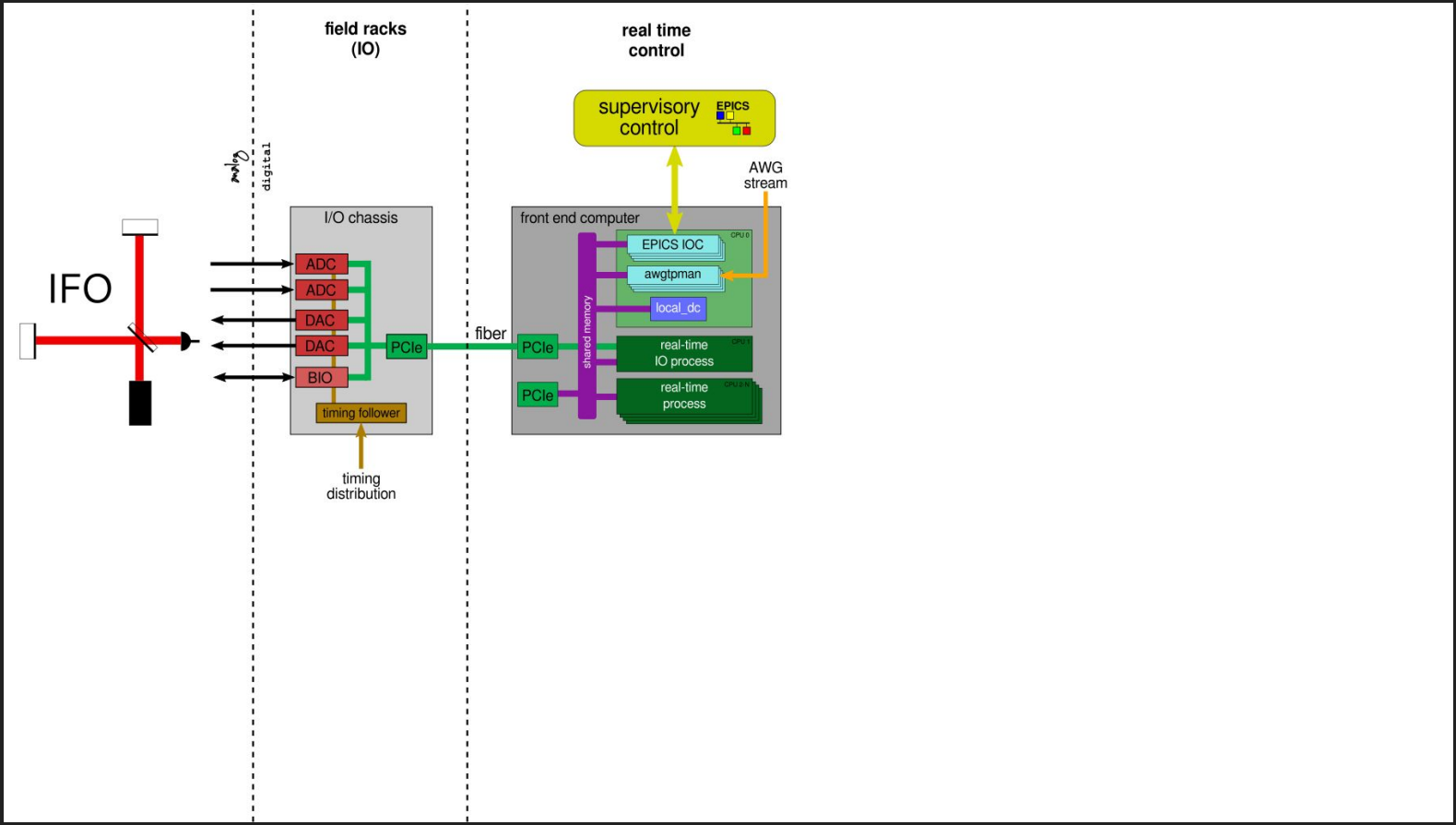
Instrument to First Storage



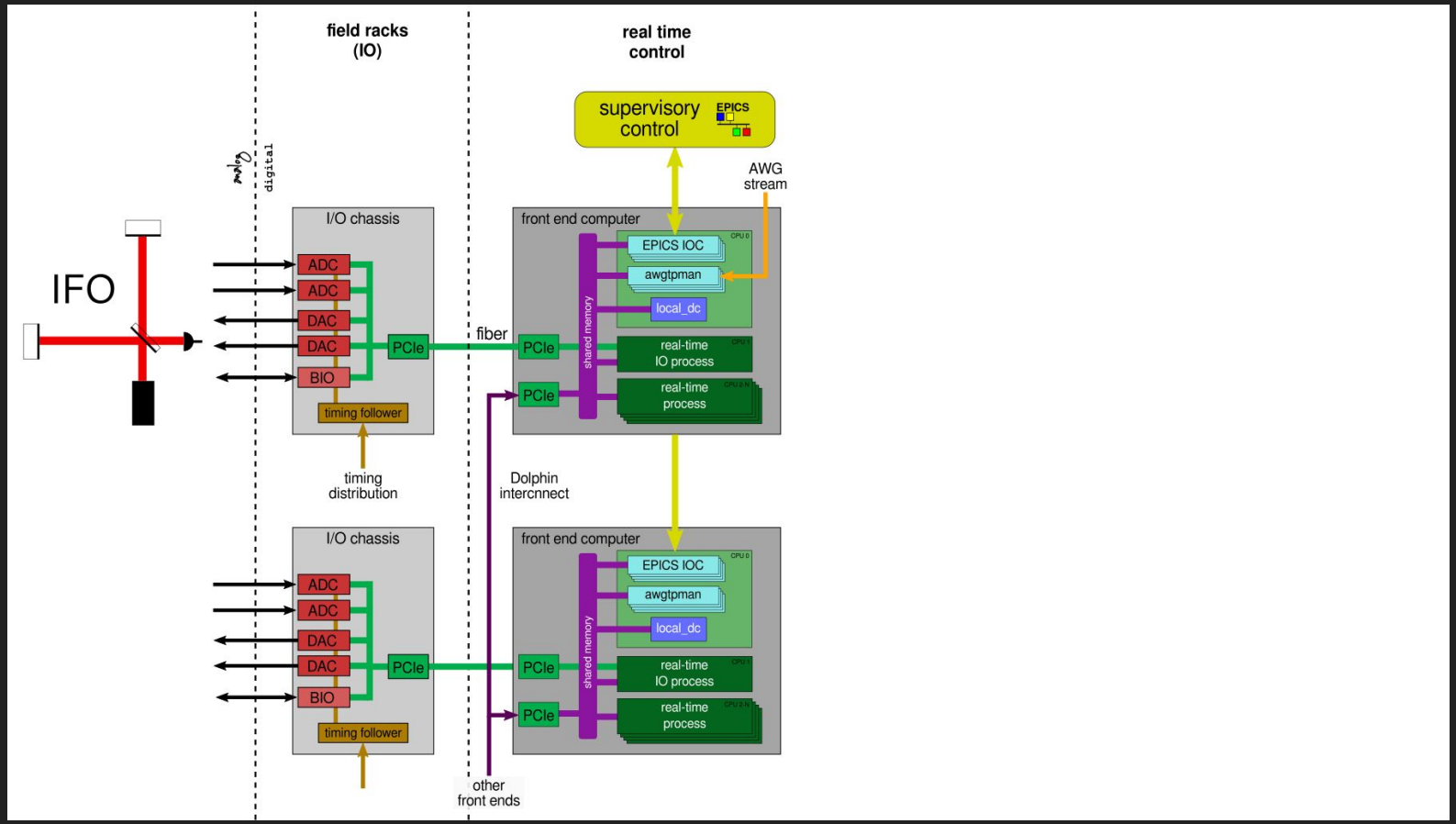
Instrument to First Storage



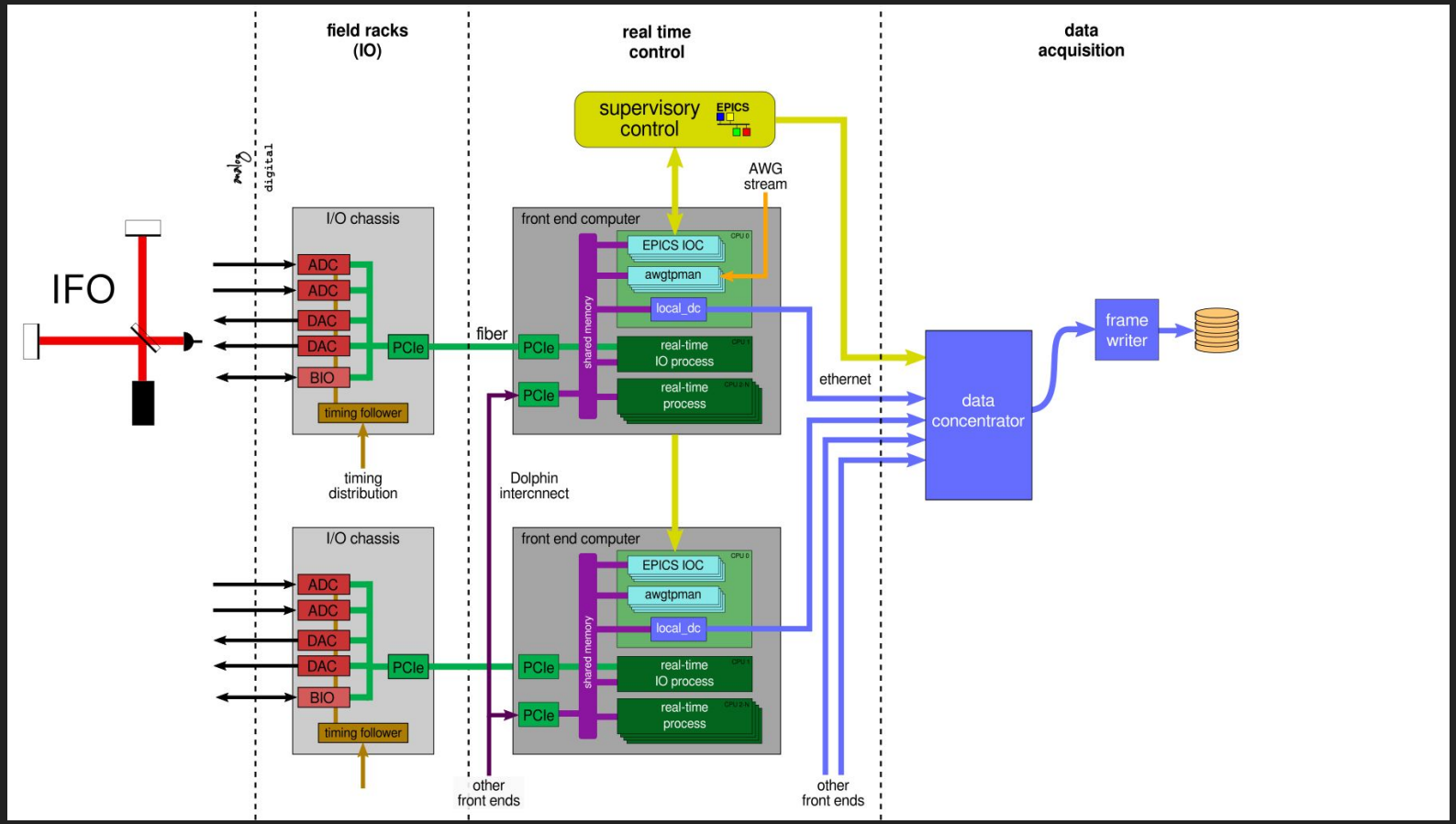
Instrument to First Storage



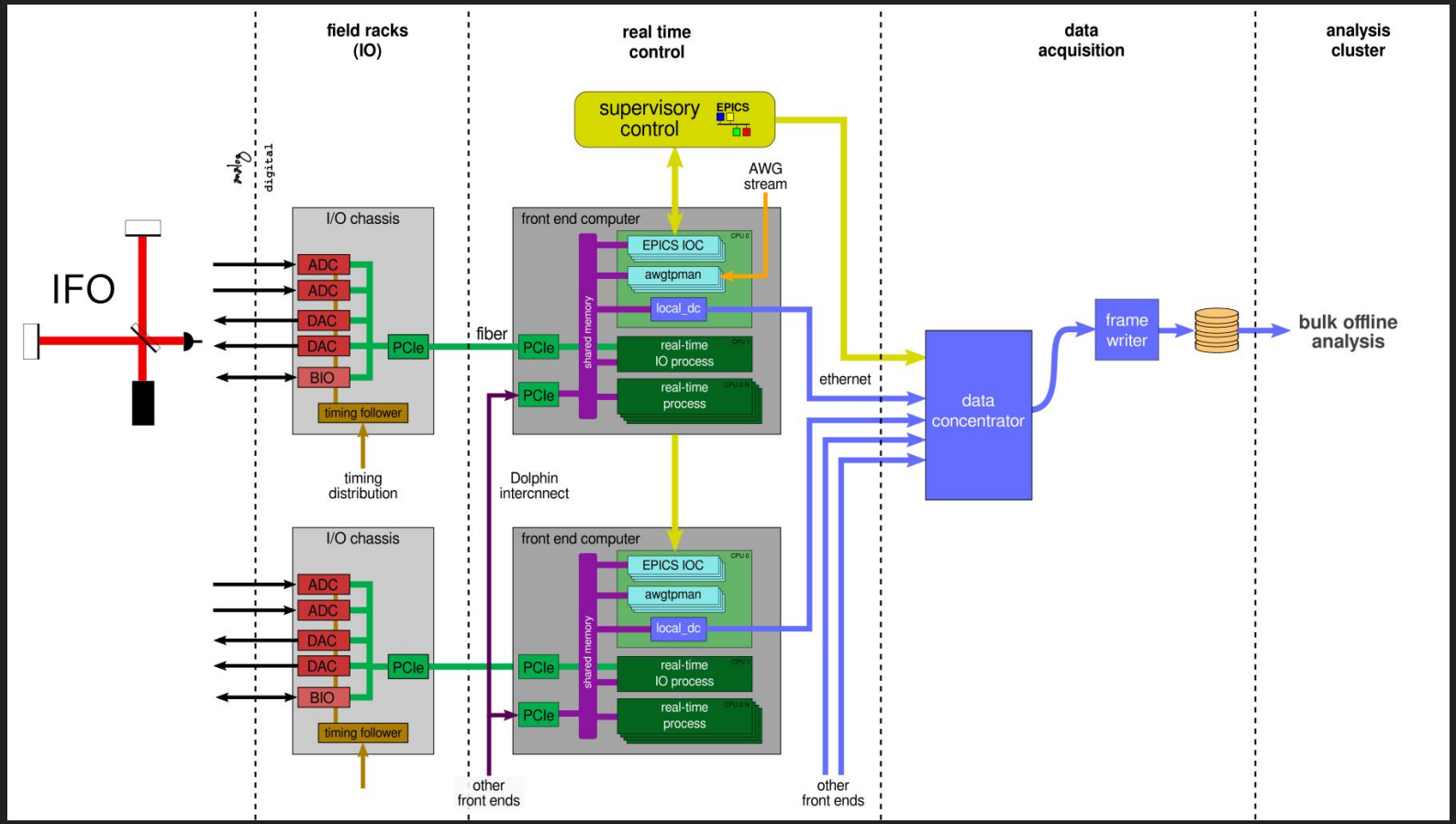
Instrument to First Storage



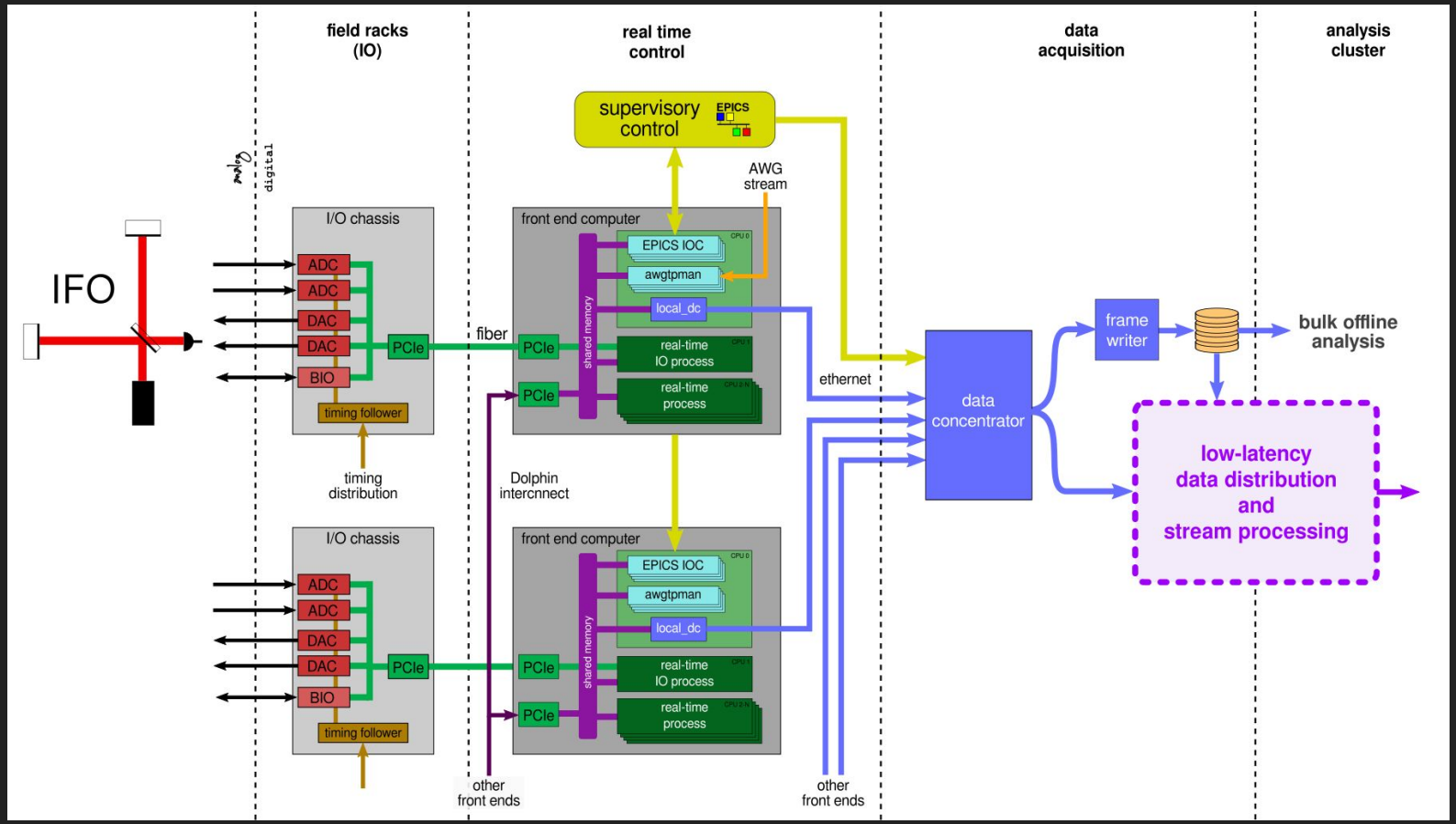
Instrument to First Storage



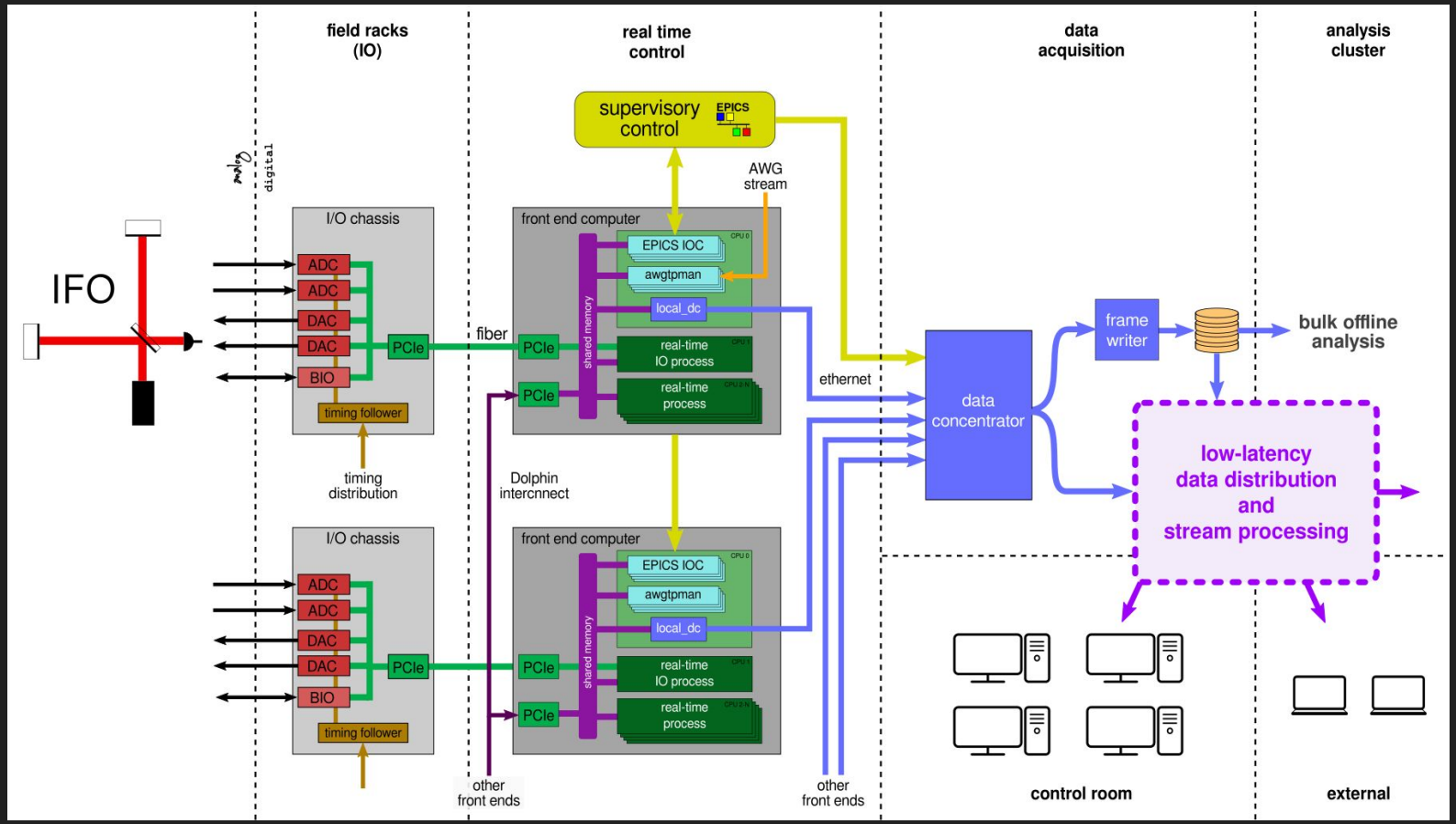
LIGO Instrument to First Storage



Instrument to First Storage



Instrument to First Storage



From offline to online

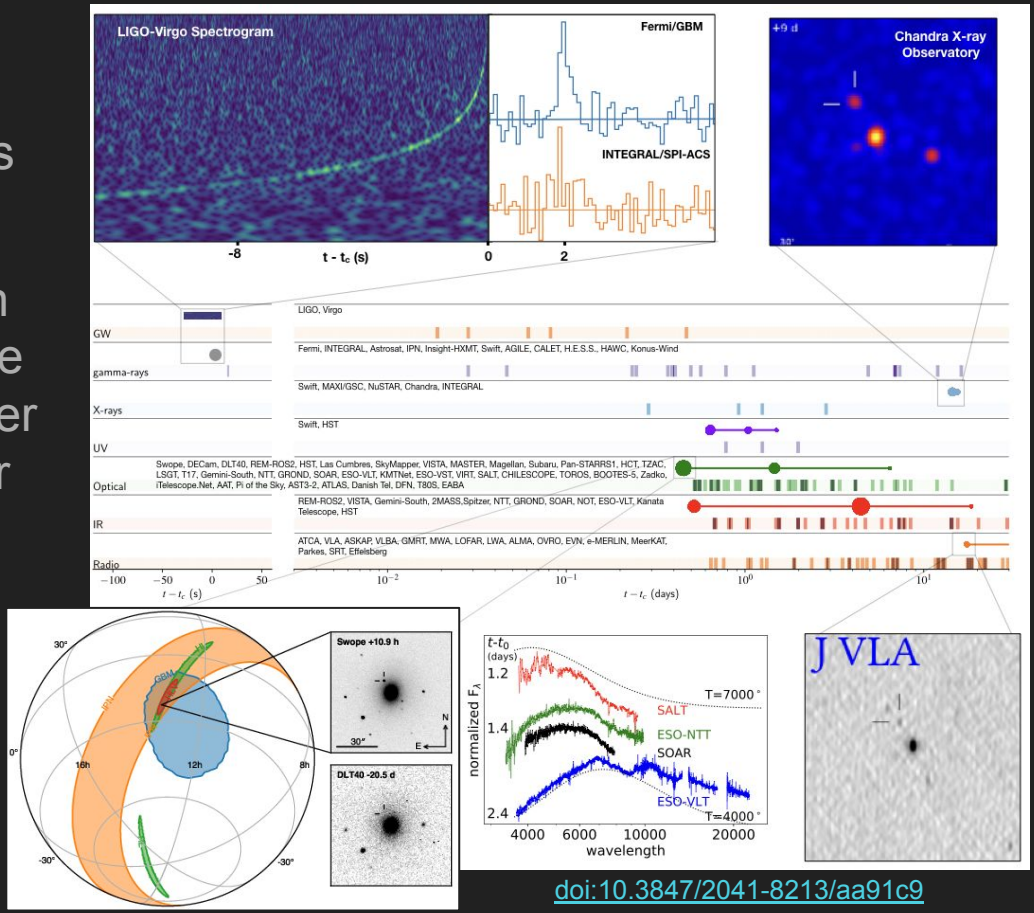
So why the (relatively) new focus on low latency data processing?

LIGO A multi-messenger gold mine

The first “low latency” searches were deployed in 2009 during Initial LIGO’s sixth science run (S6).

But the effort really came to fruition in Advanced LIGO (2015), leading to the groundbreaking 2017 multi-messenger observation of the binary neutron star merger **GW170817** →

Informing astronomers of where to look for exceptional transient events soon became one of our most important science priorities.



doi:10.3847/2041-8213/aa91c9

Many challenges for low-latency alerts

But getting out high quality alerts to the astronomical community is *difficult*.

In particular, it requires good coordination between all observatories:

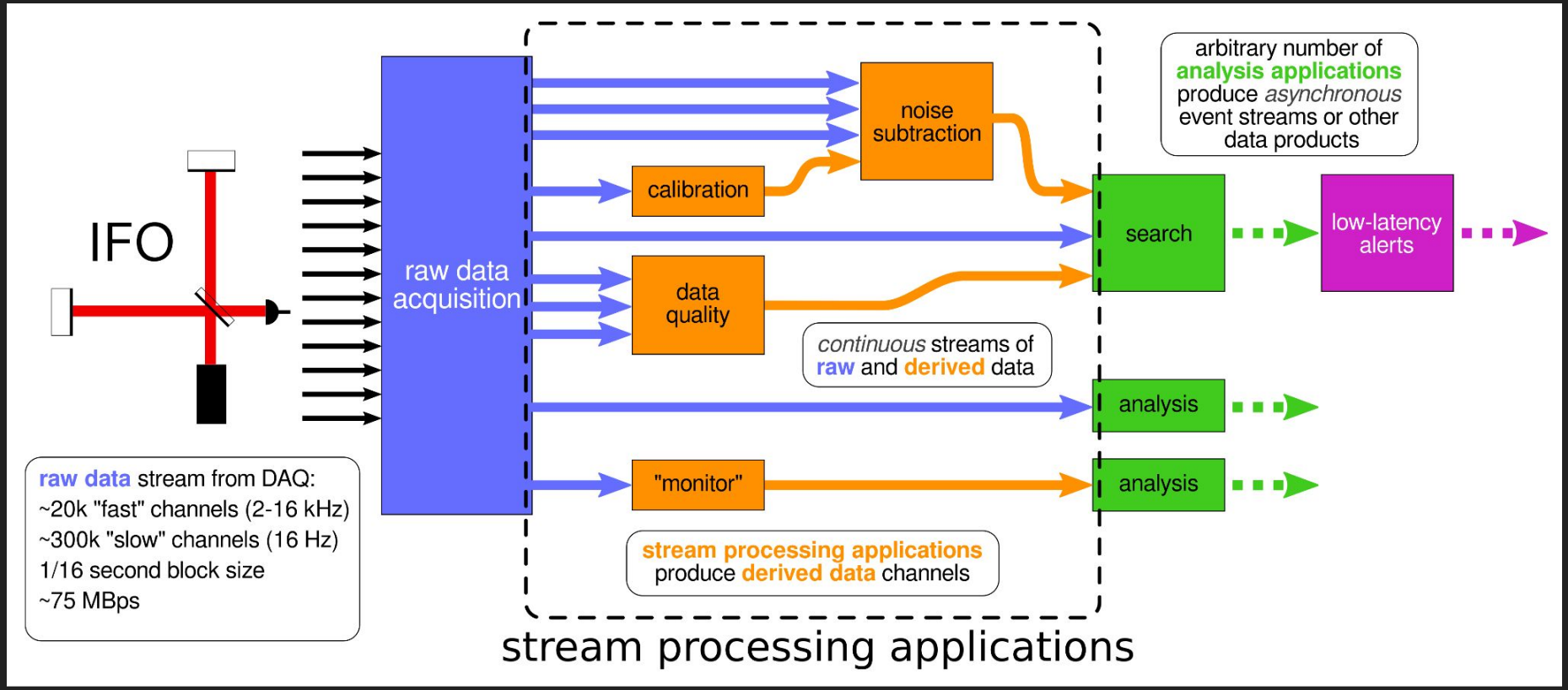
- *Localization improves with more detectors.*

Latency is also a major factor:

- *Direct correlation between the speed with which we can issue alerts and the quality of the science.*

Each observatory must also do a lot of work to calibrate, clean, annotate, etc. their data before distributing it to the searches.

Low latency pipeline conceptual overview



Next generation data delivery for LIGO

LIGO is overhauling of its stream processing and data distribution infrastructure, with the following goals:

- Focus on stream processing as primary for science goals.
- Lower latency of data delivery.
- Modernize everything: use industry standard tools and protocols
- Unify data access methods: online/offline access with the same interface.
- Publish “derived” data streams that are immediately discoverable and accessible.
- Auto discovery of data, current and past, across all domains (federation).
- Lower barrier to entry: enable broader development of stream processing applications.

Arrakis architecture and development

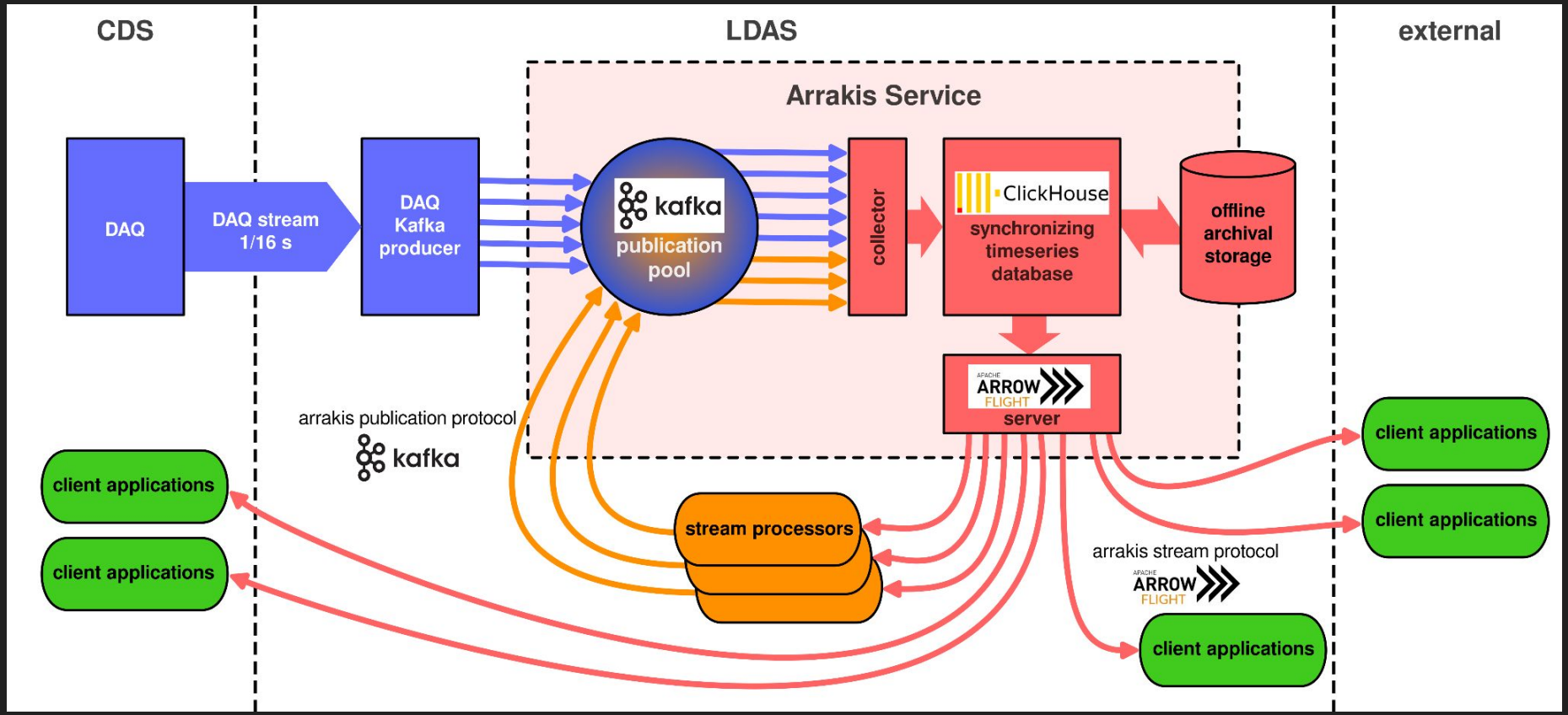
New **Arrakis** architecture overview:

- Arrow Flight (HTTP, gRPC) network protocol for both online and offline data access/streaming.
- Kafka-based derived channel publication.
- Phase 2: Kafka-based distribution of primary strain and state info channels directly to low-latency search pipelines.

Development status:

- Prototype services have been deployed at the sites, consuming all raw channels from the detectors.
- Reference client libraries for Python and Rust.
- Currently prototyping derived channel publication.

Arrakis service overview



International Gravitational Wave Network

The world-wide network of ground-based gravitational wave detection projects (LIGO, Virgo, KAGRA, plus 3G efforts) is in the process of reconstituting a new international scientific collaboration to improve coordination: ***IGWN***

So how can Arrakis help improve coordination between facilities in strengthening international network?

Data Mesh: unifying disparate domains

Data Mesh is a relevant data management and distribution concept that seems to have some traction in the tech world. The basic tenants are:

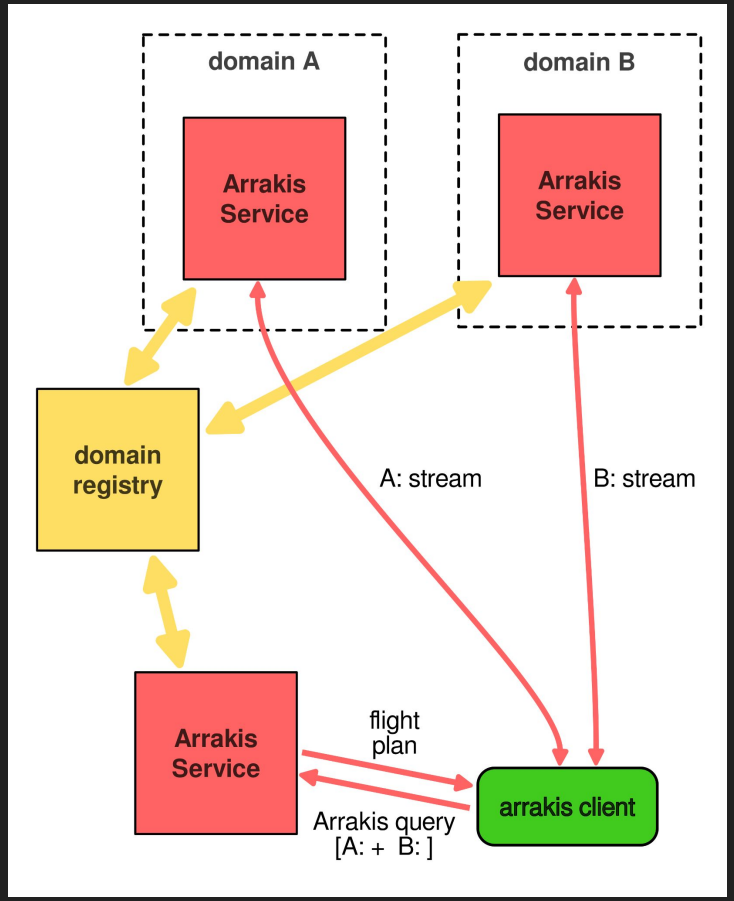
- *Distributed domain-driven architecture*
  - No centralized data store, different domains manage their own data products.
- *Data as a product*
  - Data is *discoverable, addressable, trustworthy, self-describing*, etc.
- *Self-serve data infrastructure*
  - Individual domains serve their own data.
- *Federated data governance*
  - Global standards for protocols and APIs, centralized registry of data products, etc.

Arrakis as a Data Mesh

How can we apply these principles to our services?

- Each “domain” (observatory, search pipeline, working group, etc.) would define and manage their own data products.
- Common shared protocol, API, and interfaces for access.
- Standardized data types.
- Distributed registry of available domains.
- Data discoverable across domains.

Arrow Flight protocol enables much of this →



Arrakis as a Data Mesh for IGWN

How can the Arrakis project enable this data sharing for the next generation of the world-wide network?

- Arrakis defines common standards:
  - on-the-wire protocols
  - APIs
  - data types
- Arrakis provides reference client libraries and server implementations.
  - Reference clients already allows for streaming from multiple sources.
  - Reference server already provides plugable interface for multiple data backends.
- **Maintaining a distributed registry of domains/channels/data sources is maybe the biggest challenge.**