

GA4GH Schema Registry Specification

Work Stream / Subgroup:


All GA4GH Work Streams

Date (Day/Month): **Friday April 4**

Time (ET): **9:00 - 10:30**

Session chair(s): Nathan Sheffield, Kathy Reinold, Jonathan Fuerth

Session coordinator: Beatrice Amos

Link to slides:  GA4GH-Schema-Registry-Specification.pptx

Link to Sli.do: <https://app.sli.do/event/fCQ49LUkaZaz1pEnVjodq4>

Link to meeting recording:

https://us02web.zoom.us/rec/share/0PHOnNYM92HQcBqDFtUM6RJJa7MDWcQu6f6kSrQt7BLHWJcZGEju_K9XTfBnxJCa-.xXYR4aQBS97hFUqZ

Aim of meeting:

- Align on a new proposal for a cross-GA4GH schema registry standard that defines how different standards should intercommunicate.
- Share and discuss specific use cases to ensure the schema registry addresses broad needs and is feasible for implementation by schema providers and consumers.
- Gather input from participants to assess the current proposal, identify concerns or improvements, and ensure the registry spec meets the needs of the GA4GH community before finalizing the 1.0 version.

Session description:

The primary goal of this session is to come to consensus on a new proposal for a schema registry standard. The schema registry is a cross-GA4GH standard that describes how different standards should intercommunicate. The first 30 minutes of the session will introduce the problem we are trying to solve, share a few specific use cases that would benefit from a schema registry, and describe the current proposal for a new schema registry specification. The final 60 minutes will be structured as an unconference, where attendees can bring up concerns, questions, and related topics. The goal is to collect feedback on the current proposal, determine if we are on the right track, and whether we are addressing broad use cases. This is a critical time: the proposed schema registry spec has taken shape, and there are likely to be 1-2 preliminary implementations, but it's

not set in stone. Feedback from this session on use cases and feasibility from schema providers and consumers can still influence the final 1.0 proposal.

Key Takeaways:



- A **centralised schema registry could improve schema discoverability, governance, and alignment across GA4GH workstreams**, filling a gap not addressed by current GitHub practices.
- Community input supports a **minimal, GitHub-integrated prototype as a first step** - one that includes lifecycle labeling (e.g., experimental vs approved) and namespace clarity.
- **Interoperability and alignment with global efforts** (e.g., EOSC, ELIXIR) are important for long-term success, but GA4GH should start small and focused to prove value.

Agenda:

	Agenda item	Presented by	Time
1.0	Introduction: welcome and overview		
1.1	Brief introduction to the session's goals and structure.	Kathy	5 min
2.0	Presentation: Problem, Use Cases, and Current Proposal		
2.1	- Introduction to the challenge we're solving.	Kathy	5 min
2.2	Overview of the current schema registry proposal.	Jonathan	15 min
3.0	Unconference Discussion		
3.1	- Open floor for attendees to share concerns, questions, and ideas related to the schema registry. - Focus on feedback about use cases, feasibility, and areas for improvement. - Collaborative discussion to refine the proposal and ensure broad applicability.	Nathan	60 min
4.0	Wrap up and Next Steps		
4.1	- Summarise key feedback and outline follow-up actions.		5 min

	- Discussion on the timeline and future steps for the schema registry standard.		
--	---	--	--

Resources and links:

- <https://ga4gh.github.io/schema-registry/>
-  Schema Registry Working Document
-  Schema registry user stories
-

Action items:

- **Develop a prototype of the Schema Registry** using GitHub folder structures and pathing that mimics the proposed API.
- **Define minimal metadata requirements** (e.g., schema status, version, maintainer, namespace) to test in the prototype and guide future governance.
- **Clarify scope and policies for inclusion in the registry** - starting with schemas from the GA4GH workstreams - and distinguish between approved and experimental schemas.

Meeting summary:

The meeting focused on the **Data Modeling and Schema Consensus (DaMaSC)** subgroup, which operates under the broader **TASC group**, dedicated to **fostering interoperability**.

 DaMaSC has two key initiatives:

- a Schema Registry and
- a Best Practices effort for recommending ontologies to help align across GA4GH workstreams and guide new groups.

This session concentrated on the **Schema Registry**, specifically its API, aiming to facilitate schema sharing and reuse in a lightweight, minimally invasive way.

The Schema Registry API is a general-purpose interface that allows schemas to be defined,

discovered, and shared - intended not only for GA4GH but also for broader use. The API emphasises simplicity, with each schema residing in a namespace, having a name and version (following GA4GH GKS semantic versioning guidance). Inspired by GitHub's flexible governance model, the **design allows for both personal and collaboratively governed namespaces**.

Schemas are expected to be in **JSON Schema format to ensure consistency**, although flexibility in creation formats is allowed.

Use cases include consortiums sharing in-progress schemas and public access to resources like dbGaP schemas. The group presented a prototype implementation and invited feedback on whether the API meets community needs, whether it's too complex or too limited, and what kind of metadata should be associated with schemas (e.g., lifecycle stage, maintainer, version, status). The effort **aims to fill a gap** not covered by existing schema tools by **offering a workspace for both finalised and draft schemas** - mirroring GitHub's model of open development with optional governance layers.

Larry from the Broad Institute raised a concern familiar to those working in the GKS (Genomic Knowledge Standards) work stream: **schemas and standards are often developed in a vacuum and then struggle to gain adoption.** 💡 He suggested flipping the model to start with implementers' needs and mentioned the potential benefit of having a centralised GA4GH-approved registry of schemas, especially with the growing number of schemas (30 - 40 across GKS products). Kathy responded by saying that the Experiments Metadata group has been eager for a starting point to develop schemas and has already joined the discussions, highlighting the usefulness of this effort for other GA4GH groups who want to know what's already being done elsewhere.

Ian Fore chimed in with a similar experience from working on the refget sequence collection standard. He emphasised **the need for a place to put and find schemas across GA4GH**, noting that current practices - like scattering JSON Schema drafts in GitHub repositories - aren't discoverable or coordinated. He appreciated the idea of structured registries and clear namespaces.

Larry, building on this, emphasised the **need for a well-delineated lifecycle system**: an incubator-like area for in-progress or experimental schemas vs a registry of reviewed and approved schemas. Sasha supported this, saying it's **crucial to clearly label the maturity of schemas** - "approved," "experimental," or "use at your own risk" - and thought the proposed structure addressed that well.

? Kyle then posed a foundational question: **is all this actually necessary?** Could the whole registry idea just be handled using GitHub conventions and a central list of trusted repositories, with no need for an additional specification or protocol? Jonathan clarified that yes, technically you could implement the proposed Schema Registry as static files in a GitHub repo and expose an API from that. **But the value comes in the interoperability and governance - the ability to find all GA4GH schemas, know their versions and maturity, and ensure consistency across registries.**

The group agreed that **the goal isn't to reinvent infrastructure, but to assess whether this registry API adds enough value beyond GitHub alone.** They also want to test the API further to see if it genuinely facilitates discoverability and governance.

There was a question raised about other potential registries beyond GA4GH - if interoperability is a goal, who else might be maintaining schema registries, and how would they connect?

The discussion focused on **refining the scope and approach** for a proposed GA4GH schema registry, with participants expressing **concerns and suggestions around boundaries, maintenance, utility, and technical implementation.** The need to clearly define the version 1 boundaries was emphasised - it could start as a registry limited to schemas from the nine existing workstreams. This could allow **internal testing of utility and maintenance feasibility** before proposing integration into the broader GA4GH product development approval process.

A key concern raised was ensuring **schema versions are kept up to date**, to avoid outdated schemas causing confusion, drawing parallels to the current state of the service registry, which many view as obsolete and underutilised.

There was also broader reflection on the user experience for newcomers, with examples of discoverability challenges when trying to navigate GA4GH's existing tools and services.

It was suggested that addressing these challenges - such as through updated or simplified registries - could **help new contributors more effectively engage.** Sveinung from Norway brought up the **importance of machine-operability and semantic web compatibility**, highlighting the need for the registry to support automatic schema extraction and interaction.

The conversation then shifted to whether the group should focus on the abstract question of “should we do this?” or instead test out a proposed minimal implementation, such as a GitHub repository with a folder structure that mimics the API’s intended pathing. This could help surface real issues and clarify policy needs. Participants noted **differences between the proposed registry and existing efforts like the GKS meta-schema**, suggesting that while a minimal schema registry API is attractive technically, it will require accompanying policy and governance to be usable and trustworthy.

Technically, concerns were raised about the **maturity of JSON Schema**, especially for representing complex data structures involving multiple tables, and its sufficiency for GA4GH use cases. Some argued for supporting multiple schema formats (e.g., LinkML), or at least allowing schemas to be accessible in formats beyond JSON Schema, perhaps by query parameter. The proposal, however, was clarified to **require a JSON Schema format as a minimum, not as an exclusive format**, thus leaving room for broader representation.

Overall, the conversation highlighted a classic GA4GH scenario: balancing minimal viable infrastructure with the diverse needs of users, and ensuring thoughtful policy accompanies any technical solution.

Andy Yates (EMBL-EBI) raised questions about the **schema registry’s function, particularly whether it should serve as a pointer to schemas that live elsewhere**, such as in the VRS repository. He emphasised the **need for a clear documentation landing page** and clarified that the endpoint being discussed seems to resolve to a location with more information about the schema rather than simply returning a JSON Schema file. He questioned how to handle serialisation formats like XML and protobuf, pointing out that different formats may not represent the same schema due to their inherent structural differences - especially noting that the protobuf version of VRS may not be semantically equivalent to the JSON version.

Karen agreed with Andy’s concerns and brought up the distinction between where schemas are actually developed and documented (e.g., within individual product repos like VRS or Phenopackets) vs what should be hosted in the schema registry. She asked whether the registry should just hold approved JSON Schemas with metadata pointing to their development locations or something more extensive.

Larry responded by highlighting a solution they adopted for namespace management using W3ID resolvable URLs for schemas. He suggested the GA4GH schema registry should serve as the definitive, centralised location for versioned and approved schemas only, not a catch-all repository for anything and everything. He stressed keeping it constrained to approved schemas to avoid chaos.

Another person supported Larry's view and emphasised that the schema registry should be purpose-built to support GA4GH product developers - answering questions like how a data element (e.g., an "Identifier") is represented within GA4GH standards, not across all domains.

Ian noted the importance of avoiding digressions into broad debates like identifiers - encouraging a separate task force to handle that issue instead.

💡 Sven broadened the discussion by pointing to **related efforts in Europe**, including the European Open Science Cloud (EOSC), RDA, and ELIXIR, where schema and metadata registries and crosswalks are also under active development. He encouraged **alignment with global initiatives and avoiding reinvention or obvious missteps**, even while keeping things simple and focused for GA4GH's immediate needs.