Marquez - Community Meetings & Calendar

- Marquez Workflow Group Calendar Overview
  - View Instructions on How to Subscribe to LF AI Group Calendars
  - For detailed information on LF AI meeting management processes view this page: LF AI Foundation - Community Meetings and Calendars

- Marquez Meetings List
- Marquez Group Calendar

Marquez Monthly Community Meeting

The Marquez Community Meeting occurs on the fourth Thursday of each month. Meetings are held on Zoom.

Next meeting: March 28, 2024

Tentative agenda:

1. Announcements
2. New WIP Tagging Feature: data governance and other relevant use cases [Willy Lulciuc]
3. New WIP Tagging Feature: UI implementation [Peter Hicks]
4. open discussion

February 22, 2024

Agenda:

1. announcements
2. discussion: docs and landing pages updates
3. discussion: learnings about performance issues and possible solutions
4. open discussion

Attendance:

- TSC:
  - Willy Lulciuc, Co-creator of Marquez
  - Peter Hicks, Senior Engineer, Astronomer
  - Michael Robinson, Community Team, Astronomer
  - Julien Le Dem, Project Lead, OpenLineage
- And:
  - Harsh Loomba, Upgrade
  - David Goss, Staff Software Engineer, Matillion
  - David Sharp, ANZ

January 25, 2024

Tentative agenda:

1. announcements
2. recent release
3. 2024 Roadmap
4. demos of UI tag support for datasets and a NEW UI redesign for operational lineage and column-level lineage

December 7, 2023

Tentative agenda:

1. announcements
2. 2023 recap
3. 2024 roadmap discussion
4. open discussion

October 26, 2023

Tentative agenda:

1. announcements
2. recent releases
3. I/O tab addition
4. hover-over tooltip for dataset tags demo by @David Sharp
5. static lineage progress update
6. open discussion

September 28, 2023

Agenda:

1. Announcements
2. Recent releases
3. Recent API changes
4. Recent Web UI changes
5. Static lineage and streaming support update
6. Discussion

Attendance:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Peter Hicks, Senior Engineer, Astronomer
    - Michael Robinson, Community Team, Astronomer
    - Julien Le Dem, Project Lead, OpenLineage
    - John Lukenoff, Software Engineer, Asana
- And:
    - Harel Shein, Director of Engineering, Astronomer
    - Pawe Leszczyski, Data Engineer, GetInData
    - David Goss, Staff Software Engineer, Matillion
    - David Sharp, ANZ

August 24, 2023

Agenda:

1. Announcements
2. Recent releases
3. Applying data retention for Marquez at Astronomer
4. Static lineage support update
5. Open discussion

Attendance:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Peter Hicks, Senior Engineer, Astronomer
    - Michael Robinson, Community Team, Astronomer
    - Julien Le Dem, Project Lead, OpenLineage
- And:
    - David Goss, Matillion
    - Harsh Loomba, Upgrade
    - Harel Shein, Engineering Director, Astronomer

Notes:

- Announcements [Willy]
    - Upcoming event: Airflow Summit, September 19-21 in Toronto
        - Airflow talks and OpenLineage meetup
        - Julien and Willy will be attending
    - Upcoming meetup: the first Marquez meetup will be happening in San Francisco in October
    - LF AI Graduation presentation coming up on September 7th
- Recent releases [Michael R.]
    - 0.38.0
    - 0.39.0
    - 0.40.0
- Applying data retention for Marquez at Astronomer [Willy]
- Static lineage support update [Willy]
- Open discussion

July 27, 2023

Agenda:

1. Announcements
2. Recent releases
3. Applying data retention for Marquez at Astronomer
4. Open discussion

Attendance:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Peter Hicks, Senior Engineer, Astronomer
    - Michael Robinson, Community Team, Astronomer

June 22, 2023

Agenda:

1. Announcements
2. Recent releases
3. Lineage graph cycling fix in 0.35.0
4. Datasets pagination in 0.35.0
5. Open discussion

Meeting:

Attendance:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Peter Hicks, Senior Engineer, Astronomer
    - Michael Robinson, Community Team, Astronomer
    - John Lukenoff, Software Engineer, Asana

May 25, 2023

Agenda:

1. Announcements
2. Updates
3. Recent releases
4. Db retention change
5. inputFacets and outputFacets change
6. Discussion items
    a. compaction to fix unnecessary duplication
7. Open discussion

April 27, 2023

Agenda:

1. Roadmap discussion
2. Open discussion

Meeting:

Attendance:

- TSC:
  - Willy Lulciuc, Co-creator of Marquez
  - Peter Hicks, Senior Engineer, Astronomer
  - Michael Robinson, Community Team, Astronomer
- And:
  - Harel Shein, Director of Engineering, Astronomer

Notes:

- Roadmap and OpenLineage v2 spec discussion [Willy]
  - Feedback on OpenLineage 2 discussions will be important because the evolution of the spec will require work on Marquez that will need to be aligned with the release
  - One issue in the discussions: whether to introduce a breaking change in OpenLineage, or support static lineage in a backwards-compatible way
    - Why would we want to be backwards-compatible?
      - It's still a young spec, but there have been deep integrations with Airflow and others. We've talked to stakeholders who have said they will stay on version 1 for a while rather than migrate in the case of a breaking change.
    - What additional work will we need to do on the backend to make static lineage work?
      - Marquez has always been focused on the job run specifically: it extracts inputs and outputs as events are coming in. The logic includes creating a job version when this happens. We version the output dataset. If we don't have that run anymore, what will we need to change?
      - The job and entity represented in the run event would no longer be there, so you would not have a job name, but maybe there would be facets associated with the job itself.
      - Example: Snowflake. They don't necessarily have a job, but they have query logs. These are an example of when something runs in the background is a job but it's just there to facilitate the query.
        - Query log: good way to say there was a run but there are no job details. But we do know the inputs /outputs and the tables, giving us more flexibility to capture lineage details for cases where jobs don't exist.
        - Do we implement dummy job names?
      - Marquez has been very opinionated with APIs in the model itself. Is it now loosening up in order to adapt to the changing OpenLineage standard?
        - I think it's a natural expansion of the model where you don't always have the discrete entities running SQL or code. A lot of these things can be ephemeral, which we're trying to address with the changes to the spec – one-time instances, etc.
    - Peter, you mentioned at the meetup last night that this spec change could have some implications for the UI and the user experience.
      - Peter: it's a bit of a challenge because what will, I think, be generated out of this will be a lot of these very small graphs. You have an entity, whatever. We're going to call that a jobless entity and its output. But you'll be generating a lot of that kind of small graph, and I think the UI is not really constructed around that case. So we'll have to probably think about how that exists, how we can maybe aggregate the same jobless entities together and have some heuristic based on that.
      - Facets might be optional.
      - Maybe we just base it off the run information we get – database logs where we have some timestamps available to us. The best example we have is the Snowflake adapter that's querying historical query logs. We could funnel that information into Marquez and see what we could do to generate a job name that would be consistent across different cases where job information doesn't exist.
    - There would be work needed to support new types that would make events unserializable despite being valid OpenLineage events.
      - Using the OpenLineage server models, which are more flexible, while keeping the initial high-level fields, would be one option.
        - Marquez always had a path to create a dataset (one of the proposed new types)
        - We're coming full-circle because Marquez had static lineage support at one time. We had registered datasets for jobs, then behind the scenes Marquez would create a default new state for a job that hadn't run yet.
        - At the time, the feedback was that it is a lot of work to maintain all these API calls, and it would be nice if everything was provided in the payload.
        - OpenLineage came out, and we defined everything that's specific to a run, including the job and inputs and outputs, and made a simple call.
        - Now what we're hearing from the community is that in some cases I don't have a run and I just want to register static lineage.
    - How would you see the migration working?
      - The endpoint would handle any of the types. There would be different types, but this wouldn't change the run. Everything remains the same through the contract. It's just you have additional event types, which naturally happens when you see more usage and use cases in the wild.
      - The API might need to do a little split. Static lineage will be quicker to retrieve.
    - What is the expected timeline?
      - There's no timeline at the moment but for sure this year or this quarter, or next quarter.

March 23, 2023

Agenda:

1. Announcements
2. Recent releases

3. A recent jobs symlinks fix
4. Discussion items:
    a. BI support
    b. Implications of design lineage support in the OL spec
5. Open discussion

Meeting:

Slides:

Attendance:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Peter Hicks, Senior Engineer, Astronomer
    - Julien Le Dem, Chief Architect, Astronomer
- And:
    - Michael Robinson, Software Engineer, Dev. Rel., Astronomer
    - Minkyu Park, Senior Engineer, Astronomer
    - John Thomas, Software Engineer, Dev. Rel., Astronomer
    - Benji Lampel, Product Manager, Astronomer
    - Bruno Cavestro, BI engineer curious about potentiality of the tool

Notes:

- **Announcements [Michael R.]**
    - Mar. 30: Julien (@julienledem) speaking at Data Council Austin
    - Mar. 30: OpenLineage Meetup at Data Council Austin
    - Mar. 28-30: Ross (@rossturk), Pawel (@pawel-big-lebowski) and Maciej (@mobuchowski) speaking at Big Data Tech Warsaw 2023
- **Recent Release 0.32.0 [Michael R.]**
    - **Fixed**
        - API: improve dataset facets access #2407 @pawel-big-lebowski
        - Chart: fix communication between the UI and the API #2430 @thomas-delrue
        - UI: always render MqCode #2454 @JDarDagran
    - **Removed**
        - API: remove job context #2373 @JDarDagran
        - API: remove jobs_fqn table and move FQN into jobs directly #2448 @collado-mike
    - **Release:** https://github.com/MarquezProject/marquez/releases/tag/0.32.0
    - **Changelog:** https://github.com/MarquezProject/marquez/blob/0.32.0/CHANGELOG.md
    - **Commit history:** https://github.com/MarquezProject/marquez/compare/0.31.0...0.32.0
    - **Maven:** https://oss.sonatype.org/#nexus-search;quick~marquez
    - **PyPI:** https://pypi.org/project/marquez-python/
- **Discussion Items**
    - **BI Support [Bruno]**
        - What does BI mean in connection to Marquez?
        - BI dashboard not yet part of the model, but static lineage might be relevant to BI [Julien]
        - Key metrics often part of BI, and these must be connected to lineage; a BI facet has been discussed, but introducing it presents a challenge [Willy]
        - How do machine learning models come into play? These might not be traditional BI tools but are avenues to explore. [Benji]
        - Basic, ideal situation: aggregation and documentation of where KPI is used in a chart, etc.
        - Problems include: BI tools generate internal data models or BI dashboards touch a company's main production database, making integrating OpenLineage difficult
    - **RunEvent-less Metadata Support in OpenLineage [Willy]**
        - Motivation: make community aware of RunEvent-less metadata emission proposal
        - We need to define a "v 2" of OpenLineage to define events outside of the run context. How would Marquez define lineage outside a run?
        - The run is critical to versioning in Marquez
        - Static lineage would mean focusing on the relationship between job and dataset, rather than also on the job and run
            - Julien: IMO, the only thing excluded from the current model would be the run
                - The dataset <> job version layer would be updated
            - Mike: we could uncouple the relationship between inputs/outputs and job versions
                - I would love to see a CI/CD integration
                - The only version info it would need would be the version of the code
                - Versioning info could also come from integrations, e.g., Iceberg, and be supplemented as needed.
            - Willy: another approach: implicit and explicit versioning
        - REST API > lineage event reprioritization
        - Do we need to define new events, e.g. a dataset event?
        - What are the implications for Marquez of events without run IDs?
        - Proposed model:
            - JSON schema would persist for datasets and jobs
            - inputs and outputs would be predefined
        - Recommended: review the document and give input on where the standard should go.
        - Timeline for feedback is not firm, but the discussion is accelerating.

February 23, 2023

Attendees

- TSC:
  - Willy Lulciuc, Co-creator of Marquez
  - Peter Hicks, Senior Engineer, Astronomer
  - Julien Le Dem, Chief Architect, Astronomer
- And:
  - Michael Robinson, Software Engineer, Developer Relations, Astronomer
  - Minkyu Park, Senior Engineer, Astronomer
  - John Thomas, Software Engineer, Dev. Rel., Astronomer
  - Prachi Mishra, Senior Engineer, Astronomer
  - Ross Turk, Senior Director of Community, Astronomer
  - Benji Lampel, Product Manager, Astronomer
  - Bruno Cavestro, BI engineer curious about potentiality of the tool

Agenda

- Announcements
- Recent releases
- LFAI progress update
- UI improvements demo
- Open discussion

Meeting

Slides:

Notes

- **Announcements [Willy]**
  - Feb. 3: Willy voted Marquez Project Technical Lead. Congrats, Willy!
  - Feb. 9: LFAI&Data conducted its annual review of the project
- **LFAI&Data Progress Update [Willy]**
  - Our annual review took place on February 9th
  - Project representative: Willy
  - LFAI program stages: Sandbox, Incubation, Graduation
  - Our current stage: Incubation
  - Presentation topics:
    - Project description
    - Project history
    - Key milestone: the creation of OpenLineage
    - Adoption trends
    - Graduation progress
    - Top contributor data points and trends
    - Commits, LOC by organization
    - Notable collaborations
    - OpenSSF badge (silver)
    - Governance, procedure and TSC documentation
    - LFAI collaborations
- **Recent releases 0.30.0, 0.31.0 [Michael R.]**
  - 0.30.0 (Important: please read the migration plan before upgrading)
    - API: OL facets PR #1: create and write new events to new tables while not reading them #2350 @wslulciuc @pawel-big-lebowski
    - API: OL facets PR #2: read facets from views based on lineage_events table #2355 @pawel-big-lebowski
    - API: OL facets PR #3: migrate data to facet tables #2359 @pawel-big-lebowski
    - UI: Display column lineage of a dataset #2293 @pawel-big-lebowski @tito12
    - UI: Add soft delete option to UI #2343 @tito12
    - Docker: add new script for stopping Docker #2380 @rossturk
    - **Migration plan:** https://github.com/MarquezProject/marquez/blob/main/api/src/main/resources/marquez/db/migration/V57__readme.md
    - **Release:** https://github.com/MarquezProject/marquez/releases/tag/0.30.0
    - **Changelog:** https://github.com/MarquezProject/marquez/blob/0.30.0/CHANGELOG.md
    - **Commit history:** https://github.com/MarquezProject/marquez/compare/0.29.0...0.30.0
    - **Maven:** https://oss.sonatype.org/#nexus-search;quick~marquez
    - **PyPI:** https://pypi.org/project/marquez-python/
  - 0.31.0
    - UI: add facet view enhancements #2336 @tito12
    - UI: highlight selected path on graph and display status of jobs and datasets based on last 14 runs or latest quality facets #2384 @tito12
    - UI: enable auto-accessibility feature on graph nodes #2388 @merobi-hub
    - **Release:** https://github.com/MarquezProject/marquez/releases/tag/0.31.0
    - **Changelog:** https://github.com/MarquezProject/marquez/blob/0.31.0/CHANGELOG.md
    - **Commit history:** https://github.com/MarquezProject/marquez/compare/0.30.0...0.31.0
    - **Maven:** https://oss.sonatype.org/#nexus-search;quick~marquez
    - **PyPI:** https://pypi.org/project/marquez-python/
- **UI improvements [Peter]**

        - Check out the new Gitpod feature, an easy way to try out Marquez
        - Shout out to Vlad for doing much of the recent work on the UI
        - Critical path highlighting
            - recursive path-chasing for currently selected graph node
        - Replaced the JSON snippet view
            - wanted something that wouldn't overwhelm users with hundreds of lines of data
            - search adds convenience
            - implemented across the board in the project wherever we display JSON
            - plan to enhance this in the future
        - Added a lineage events viewer
            - paginated
        - Added delete functionality
        - Q & A
            - Benji: Is this the only place where you can delete (as opposed to on the graph or the sidebar)?
                - Currently, the dataset and run views, in addition to the API, are the only vehicles for the soft delete feature
            - Willy: it would be nice to aggregate functions including soft deletion in the main, sidebar-accessible page for datasets and jobs
- **Discussion**
    - Benji: on the subject of aggregation, is locating deletion anywhere else in the UI on the roadmap?
        - Willy: yes, we're taking baby steps
        - Julien: example:
            - delete is useful when M is unaware a DAG has been renamed;
            - it's a flag, so M keeps the history – nothing is actually removed from the db
        - Willy: there's also a namespace deletion feature that is also flag-based; maybe a force delete could be added in the future
    - Michael R.: noteworthy new issue is #2428, Proposal for improving the visibility of lineage
        - Willy: agree – there are great ideas here
    - Bruno: an experience BI engineer connecting from Europe
        - introduces himself, attending because looking for a lineage tool
        - almost no tool is able to do the job for BI
        - what is the philosophy?
        - Julien: we plan to discuss how to incorporate support for BI tooling in the OpenLineage spec in upcoming meetups
        - considering an idea to implement code from lineage with a parser using Marquez
        - Willy: for us, we wanted a tool for deriving the schema at runtime
        - Julien: observability is central: we instrument running transformations
            - we discovered that the only way to boil the ocean is through a standard format
        - what I like about Marquez is the simplicity of it
        - why did you start from the big data world with Spark and so on?
            - the market for big data tooling and the impact you can have based on this
        - Willy: it would be great to discuss support for BI tools at future Marquez and OpenLineage meetings
        - what are facets meant to represent?
            - Julien: a flexible mechanism for extending/specializing the spec
            - for example: versioning (you can add the version of Git used), schema, column-level lineage, logical plan
            - see the docs site for more information

January 26, 2022

Attendees

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Michael Collado, Staff Software Engineer, Astronomer
    - Peter Hicks, Senior Engineer, Astronomer
- And:
    - Howard Yoo, Staff Product Manager, Astronomer
    - Michael Robinson, Software Engineer, Developer Relations, Astronomer
    - Minkyu Park, Senior Engineer, Astronomer
    - Maciej Obuchowski, OpenLineage Committer and Software Engineer, GetInData
    - John Thomas, Software Engineer, Dev. Rel., Astronomer
    - Prachi Mishra, Senior Engineer, Astronomer
    - Pawe Leszczyski, Data Engineer, GetInData
    - Sam Holmberg, Senior Engineer, Astronomer
    - Yannick Libert, Lead Data Engineer, Decathlon France
    - Henoc Mukadi, Prodigy Finance
    - Bramha Naidu Aelem, Big Data/ML/AI Cloud Architect, Tiger Analytics

Meeting

Agenda

- Announcements [Willy]
- Recent release 29.0 [Michael R.]
- Column lineage overview and demo [Pawel]
- Soft delete UI feature overview [Howard]
- New OpenLineage facets, migration process [Willy]
- "2023 in Marquez" roadmap discussion [Willy]

Notes

- **Announcements [Willy]**
  - Marquez will be applying for Graduation status with the LFAI&Data Foundation soon! Stay tuned for updates.
- **Recent releases [Michael R.]**
  - **0.29.0**
    - Added
      - Add point-in-time requests support to column-lineage endpoints #2265 @pawel-big-lebowski
      - Add column lineage point-in-time Java client methods #2269 @pawel-big-lebowski
      - Add raw event viewer to UI #2249 @tito12
      - Update events page with styling synchronization #2324 @phixMe
      - Update helm Ingress template to be cross-compatible with recent k8s versions #2275 @jlukenoff
      - Add delete namespace endpoint doc to OpenAPI docs #2295 @mobuchowski
      - Add i18next and language switcher for i18n of UI #2254 @merobi-hub @phixMe
      - Add indexed `created_at` column to lineage events table #2299 @prachim-collab
    - Thanks to all our contributors, including new contributors @jlukenoff and @tito12. For more details and the many bug fixes included in the release see:
      Release: https://github.com/MarquezProject/marquez/releases/tag/0.29.0
      Changelog: https://github.com/MarquezProject/marquez/blob/0.29.0/CHANGELOG.md
      Commit history: https://github.com/MarquezProject/marquez/compare/0.28.0...0.29.0
      Maven: https://oss.sonatype.org/#nexus-search;quick~marquez
      PyPI: https://pypi.org/project/marquez-python/
- **Column lineage UI feature overview & demo [Pawe]**
  - now it's possible to see in the UI whether or not column lineage has been created
  - a new tab in `datasetInfo` displays column lineage JSON
  - not a "big" feature but important for testing
  - point-in-time lineage support currently available via the API will hopefully be added to this and other features in the UI
  - note: column-level lineage has recently been added to the seed data [Ross]
  - hopefully this will inspire someone to do UI work [Willy]
    - great "seed" for someone to build on!
  - where can you find the seed data? [Howard]
    - Ross: I created it manually; you'll find it in the output datasets in the seed data
- **Soft delete UI feature [Howard]**
  - frequently requested feature
  - API for deleting datasets, etc. exists
  - now there is a delete button in the UI for jobs and datasets
    - logical deletion routine
    - takes you back to the dataset list because the graph cannot be rendered
  - doesn't delete the dependencies upstream or downstream
  - what happens if you delete something in the middle of the graph? [Peter]
    - splits the graph
  - improved notifications are desired
  - namespace deletion also desirable
  - driven through a flag, because retaining historical metadata is necessary [Willy]
- **OpenLineage facets [Willy]**
  - overview and status update
  - proposal by Pawe and Willy viewable by anyone
  - `lineage_events` table has been the source for facets, but these tables can be very large, creating performance issues and out of memory exceptions
  - proposed solution: three new separate tables for facets
  - will allow for adding of new functionality and features
  - three PRs in process (2359, 2355, 2350)
  - those running PSQL 12 will require `sudo`
  - feedback on this is desired [Mike C.]
    - is anyone running Postgres 12?
    - is using a super user a burden?
  - scheduled for Marquez 0.30.0, expected next week
- **Project roadmap [Willy]**
  - extensibility
    - search
      - in-memory search
      - backend search service with common interface
    - auth layer
      - need of large orgs
      - considering the best platform for this
      - more community involvement desirable
      - more important as features such as soft delete are added
    - notifications
      - exploring methods for plugging in Slack, PagerDuty notifications
  - scalability
    - API performance testing
    - data retention
    - data modeling
  - usability
    - UI
      - lineage graph extension
    - website
      - blog post or other resource on how orgs are deploying Marquez?
        - how do users scale their architecture as their deployments grow?
  - live roadmap always available on GitHub
- **Open discussion**

        - Question following up on the following post in the OpenLineage Slack:
            - First activity : Making HTTP Call to pull the lookup data and store it in datalake.
              Second Activity : After the completion of first activity, invoking Azure databricks to use the lookup file and generate the output tables. What are all the steps should be follow to refer databricks generated tables facets as an output in the current activity & input to the subsequent activities in the pipeline.
              When I configure spark generated tables as output to the current activity the existing spark metadata is not showing up. How can this be achievable.
            - recommendation: use dataset symlinks feature due to the multiple inputs, if this is possible using Spark [Pawe]
            - examples go a long way with questions like this [Willy]

November 17, 2022

Attendees:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Michael Collado, Staff Software Engineer, Astronomer
    - Peter Hicks, Senior Engineer, Astronomer
    - Julien Le Dem, Chief Architect, Astronomer
- And:
    - Howard Yoo, Staff Product Manager, Astronomer
    - Michael Robinson, Software Engineer, Developer Relations, Astronomer
    - Minkyu Park, Senior Engineer, Astronomer
    - John Thomas, Software Engineer, Dev. Rel., Astronomer
    - Prachi Mishra, Senior Engineer, Astronomer
    - Ross Turk, Senior Director of Community, Astronomer

Agenda:

- Announcements
- LFAI & Data progress update
- Under-documented topics
- Review of major architectural decisions
- Open discussion

Meeting:

Notes:

- Announcements [Willy]
    - Marquez 0.28.0 is coming soon, featuring:
        - new optimized current runs query
        - new governance docs
        - ability to soft-delete namespaces
    - The next Marquez meeting will be on January 26th
- LFAI & Data progress update [Michael R.]
    - LFAI & Data structure
        - under umbrella of the LF
        - hosted projects need approval of TAC and Governing Board
    - Marquez one of many open-source projects hosted by the LFAI
    - Current status (since December 2019): Incubation
    - Next milestone: Graduation
    - To dos/outstanding:
        - one unassociated significant contribution (e.g., integration)
        - CII Silver Badge (96%)
        - CII Gold Badge (83%)
        - appointment of technical lead
        - approving votes of LFAI TAC and Governing Board
- Under-documented topics [Willy]
- Review of major architectural decisions [Willy]
- Open discussion
    - should Marquez have a social media account in addition to the Twitter account?
        - Mastodon a good candidate
        - an unofficial OpenLineage account already exists there
    - how can the project be internationalized to meet the expectations of the LFAI?
        - a tool such as React's i18next would make this task less daunting

October 27, 2022

Attendees:

- **TSC:**
    - Willy Lulciuc, Co-creator of Marquez
    - Michael Collado, Staff Software Engineer, Astronomer

- **And:**
    - Michael Robinson, Software Engineer, Developer Relations, Astronomer
    - Ross Turk, Senior Director of Community, Astronomer
    - Pawe Leszczyski, Data Engineer, GetInData
    - Minkyu Park, Senior Engineer, Astronomer
    - John Thomas, Software Engineer, Dev. Rel., Astronomer
    - Arek Osinski, Senior Data Engineer, Allegro Group
    - Prachi Mishra, Senior Engineer, Astronomer

Agenda:

1. Announcements
2. Recent release 0.27.0
3. Dataset symlinks feature demo [Pawel]
4. Node color changes to reflect run state in UI demo [Willy]
5. UI improvements roadmap review [Willy]

Meeting:

Notes:

- Announcements
    - Marquez 0.27.0 was released on October 24th
    - FYI, today, October 27th, is the CFP deadline for Data Council Austin 2023
- Recent release 0.27.0
    - New dataset symlinks feature:
        - Implement dataset symlink feature #2066 @pawel-big-lebowski
        - Provide dataset_symlinks table for SymlinkDatasetFacet #2087 @pawel-big-lebowski
    - New column lineage feature:
        - Model and store column lineage in Marquez #2096 @mzareba382 @pawel-big-lebowski
        - Add a lineage graph endpoint for column lineage #2124 @pawel-big-lebowski
        - Enrich returned dataset resource with column lineage information #2113 @pawel-big-lebowski
        - Add downstream column lineage #2159 @pawel-big-lebowski
        - Include column lineage in dataset resource #2148 @pawel-big-lebowski
        - Implement column lineage within Marquez Java client #2163 @pawel-big-lebowski
        - Add endpoint to get column lineage by a job #2204 @pawel-big-lebowski
        - Add column lineage methods to Python client #2209 @pawel-big-lebowski
        - Fix column lineage returning multiple entries for job run multiple times #2176 @pawel-big-lebowski
        - Increase size of column-lineage.description column #2205 @pawel-big-lebowski
        - Fix downstream recursion #2181 @pawel-big-lebowski
    - Lineage graph changes:
        - Display current run state for job node in lineage graph #2146 @wslulciuc
    - API changes:
        - Add indices on the job table #2161 @phixMe
        - Update insert job function to avoid joining on symlinks for jobs with no symlinks #2144 @collado-mike
        - Add support for parentRun facet as reported by older Airflow OpenLineage versions #2130 @collado-mike
        - Add fix and tests for handling Airflow DAGs with dots and task groups #2126 @collado-mike @wslulciuc
        - Fix bug that caused a single run event to create multiple jobs #2162 @collado-mike
        - Fix API spec issues #2178 @phixMe
        - Update jobs_current_version_uuid_index and jobs_symlink_target_uuid_index to ignore NULL values #2186 @collado-mike
    - Release: https://github.com/MarquezProject/marquez/releases/tag/0.27.0
    - Changelog: https://github.com/MarquezProject/marquez/blob/0.27.0/CHANGELOG.md
    - Commit history: https://github.com/MarquezProject/marquez/compare/0.26.0...0.27.0
    - Maven: https://oss.sonatype.org/#nexus-search;quick~marquez
    - PyPI: https://pypi.org/project/marquez-python/
- Dataset symlinks feature demo [Pawel]
    - This workshop is available, including all installation steps, in the `openlineage/workshops` repository on GitHub
    - Scenario: datasets are sometimes known by different names
    - This can lead to broken lineage
    - An extra facet makes the dataset symlinks feature possible
    - The facet is used to create lineage edges over an alternate name
    - Workshop notes:
        - involves starting a Spark cluster and using the Spark OpenLineage connector, accessing a Hive metastore
        - when verifying the event using the Marquez events API endpoint, one can see the different name in the output
        - when trying to locate the same dataset using both names, one gets the same dataset back
        - when accessing a lineage graph, one can see two tables represented
- Node color changes to reflect run state in UI demo [Willy]
    - The changes are part of a recent PR completed with the help of Peter Hicks
    - The work was spurred by a discussion in the #random channel in the Marquez Slack
    - Colors are now used to indicate run state in the UI
    - Now available, and feedback is welcome
    - Question: does the run state come from Airflow?
        - The Airflow integration does support it
        - But it is supported globally, as well
        - More information about using Airflow with Marquez is available in `marquez/examples/airflow` on GitHub
- UI improvements roadmap review [Willy]
    - The roadmap is publicly available on GitHub Projects
    - The roadmap is filterable by label (e.g., "web")
    - Potential contributors are welcome to pick up any of the good first issues there

- Howard Yoo does a lot of work on the roadmap
- You should start seeing more of these features in future releases:
    - raw event viewer to make use of the new events endpoint
        - will make event stats and JSON payloads available in the UI
    - search enhancements
        - recently proposed: use Elastic Search, instead of matched text searching, for search
    - facet viewer
        - will take advantage of OpenLineage facets, make them interactive in the UI (expandable, collapsable, etc.)
    - time range-based query
        - will provide an API for retrieving historical data, make former versions of datasets viewable and comparable
        - dataset versions, job versions, run IDs make point-in-time snapshots possible
        - discussions about how to proceed are ongoing
    - lineage graph display mode
        - will make job status visible in the UI (e.g., "failed")
    - soft delete
        - ability to delete metadata
        - Marquez should be the source of truth, so deletion should not be permanent
        - "deleted" datasets will be available on the backend but not visible on the lineage graph
        - under discussion: should all users have the ability to delete?
- Feedback on all open issues is welcome!
- Big thanks to Howard Yoo for his work on these issues!

September 22, 2022

Attendees:

TSC:

- Willy Lulciuc, Co-creator of Marquez
- Peter Hicks, Senior Engineer, Astronomer
- Julien Le Dem, Chief Architect, Astronomer
- Michael Collado, Staff Software Engineer, Astronomer

And:

- Pawe Leszczyski, Data Engineer, GetInData
- Harel Shein, Director of Engineering, Astronomer
- Ross Turk, Senior Director of Community, Astronomer
- Howard Yoo, Staff Product Manager, Astronomer
- Michael Robinson, Software Engineer, Developer Relations, Astronomer
- Ryan Hatter, Customer Reliability Engineer, Astronomer
- Minkyu Park, Senior Engineer, Astronomer
- Maciej Obuchowski, OpenLineage Committer and Software Engineer, GetInData
- John Thomas, Software Engineer, Dev. Rel., Astronomer
- Herrick Muhlestein, Software Engineer, Ancestry
- Amay Kadre, Senior Software Engineer, Ancestry
- Dayle Woolston, Principal Software Engineer, Ancestry

Agenda:

Announcements

Recent release 0.26.0

Recent work on versioning

New and in-process APIs

Discussion topics:

How to improve the Marquez UI?

New/in-process APIs

Enhancing search to include schema field names and facets

Meeting:

Slides: https://docs.google.com/presentation/d/160WuwGB0hQSpfMRq_4_R0xls6VXkFYxtpQvj57_SIw0/edit?usp=sharing

Notes:

Announcements

- Marquez stickers are still available: https://www.astronomer.io/datakin-swag
- Recent talk:

    - Willy at LinuxCon: https://www.youtube.com/watch?v=sN7j5mZcUQA
- LFAI & Data progress update:
    - External contributions needed

- Marquez 0.26.0

    - ADDED
        - Add possibility to soft-delete datasets and jobs #2032 #2099 #2101 @mobuchowski
        - Add raw OpenLineage events API #2070 @mobuchowski
        - Update FlywayFactory to support an argument to customize the schema programmatically #2055 @collado-mike
        - Add --metadata option & metadata cmd #2082 #2091 @wslulciuc
        - Create column lineage endpoint proposal #2077 @julienledem @pawel-big-lebowski
        - Add steps on proposing changes to Marquez #2065 @wslulciuc
        - Improve documentation on nodeId in the spec #2084 @howardyoo

- CHANGED
    - Update lineage query to only look at jobs with inputs or outputs #2068 @collado-mike
    - Persist OpenLineage event before updating Marquez model #2069 @fm100
    - Drop requirement to provide marquez.yml for seed cmd #2094 @wslulciuc
- FIXED
    - Fix/rewrite jobs fqn locks #2067 @collado-mike
    - Fix enum string types in the OpenAPI spec #2086 @studiosciences
    - Fix incorrect PostgresSQL version #2089 @jabbera
    - Update OpenLineageDao to handle Airflow run UUID conflicts #2097 @collado-mike
- **Release:** https://github.com/MarquezProject/marquez/releases/tag/0.26.0
- **Changelog:** https://github.com/MarquezProject/marquez/blob/0.26.0/CHANGELOG.md
- **Commit history:** https://github.com/MarquezProject/marquez/compare/0.25.0...0.26.0
- **Maven:** https://oss.sonatype.org/#nexus-search;quick~marquez
- **PyPI:** https://pypi.org/project/marquez-python/

Recent work on versioning [Ryan]

- Open issues regarding dataset versioning: 1977, 1883
- Confusing: schema for dataset version contains UUID and version field
- 2071: tries to resolve confusion by removing the version field and replacing it with an external version
    - supports different tools that might have a dataset version baked in
    - Mqz users can use this as a version field
- These improvements are welcome [Willy]
    - The project's approach to versioning has remained unchanged since we began
    - Mqz is opinionated about versioning, but other systems and dbs have their own versioning
    - This change hasn't been on the roadmap, but we've known for a long time that it was needed
    - This will add the flexibility that OpenLineage offers

- New and in-process APIs [Maciej]

    - Row event API
        - Future work needed: make it possible to get the data via namespaces
        - Challenge
    - Delete APIs
        - soft delete approach
        - future work: make it possible to clear an entire namespace
        - planned: "real" deletion
    - Q & A
        - is it possible to undelete datasets and jobs?
        - not yet
- Discussion topics

    - updating the UI [Howard]
        - list of possible improvements:

Item	Importance	Description	Target Date
Facet View Enhancement	?	When viewing job or dataset, UI should be able to pull in all the facet information, and present it in an easy to Navigate way. Also convert URLs or links to working links.	?
Raw Event Viewer	?	Marquez should be able to provide the UI that displays the raw event API result (should also be paginated for ease of use and performance)	?
Soft Delete	?	Display option to delete jobs and datasets using soft delete button. Also, feature to search and 'undelete' soft deletes if necessary. Might be required to delete 'group' of datasets or jobs - so multi-deletion should be available. Also, option to delete single instance of it, or all the past instances of it.	?
Time range based query	?	Current visualization does NOT have features to view lineage graph in point-of-time, or in ranges of time. This feature will enable users to be able to track and view the historical changes that happened in their lineage over period of time.	?
Graph display mode enhancements	?	Display graphs in various style (star, linear, circle, etc.), as well as highlight various status of the jobs / datasets. For example, display job-centric or dataset-centric graph - option to turn on/off things like data quality, errors, recency (or the events), density (how frequent events are received), etc.	?
Search Enhancements	?	Dataset Column names and descriptions Job Code (e.g. SQL queries) Job facet property names and values within JSON Job descriptions	?
Other search ideas	?	Ability to limit search downstream from a specific job or dataset display job with status color in lineage view to quickly find failed jobs add a "last status" column in jobs list view to quickly find failed jobs data consumer awareness such as a dashboard custom dataset icons (kafka, api, etc.)	?

- Enhancing search [Herrick]
  - Current state of search in Mqz: helpful if looking for job or dataset
  - However, more data is available
  - First question from our data governance team: can we look up a column?
  - Possible enhancements:
    - search for column names and descriptions
    - job codes (e.g., SQL queries)
    - job facet property names and values within JSON
    - job descriptions
  - Potential benefits:
    - easy to find where data comes from given a column or keyword
    - what job transformed a column in a dataset
    - reverse lookup from metadata such a SQL query or S3 bucket to find a related job
  - Other ideas:
    - limit search downstream from a specific job or dataset
    - display job status color in lineage view to quickly find failed jobs
    - add a "last status" column in the Jobs list view to quickly find failed jobs
    - data "consumer" awareness such as a dashboard (OpenLineage dependency)
    - custom dataset icons (Kafka, API), to help visualize where things are coming from
  - Some of these ideas could be implemented in conjunction with existing ongoing projects, such as column-level lineage [Mike C.]
  - We would be happy to help you be successful in whichever parts of these you would want to build [Julien]
  - These are small changes but very impactful for usability [Willy]

                - search has never been very sophisticated because we're not using a true search engine
                - start by creating issues!
            - Some of these are low-hanging fruit, but it would be helpful to have them prioritized [Peter]
            - A UI hack day might be all we need to knock many of these out [Willy]
            - Publicize some of these by creating issues labeled as good first issues [Minkyu]

August 25, 2022

Attendees:

- TSC**:**
    - Michael Collado, Staff Software Engineer, Astronomer
    - Julien Le Dem, Chief Architect, Astronomer
    - Willy Lulciuc, Co-creator of Marquez
- And:
    - Minkyu Park, Senior Engineer, Astronomer
    - Nikhil Koli, Software Engineer, Moody's
    - Harel Shein, Director of Engineering, Astronomer
    - Michael Robinson, Software Engineer, Developer Relations, Astronomer
    - Ryan Hatter, Customer Reliability Engineer, Astronomer
    - Howard Yoo, Staff Product Manager, Astronomer

Agenda:

- Announcements [Willy]
- Recent release 0.25.0 [Michael R.]
- Column-level lineage proposal [Julien]
- Lineage optimization of `getLineage()` [Michael C.]
- New proposal process [Willy]
- Optimization of query performance for facets [Willy]
- Runs API removal/migration [Willy]

Notes:

- Announcements [Willy]
- Recent release 0.25.0 [Michael R.]
    - Fixed
        - Fix `py` module release #2057 @wslulciuc
        - Use /bin/sh in web/docker/entrypoint.sh #2059 @wslulciuc
- Column-level lineage proposal [Julien]
    - Main use case: compliance (GDPR, CCPA, etc.)
        - private information especially
        - banking regulations
    - Point in time lineage
        - retrievable from Marquez: version of database in the past
        - makes it possible to identify exactly where a breakdown in protocols happened
    - New facet in the OpenLineage spec
        - for each col in output, you can specify where the data came from
        - can also identify whether data is masked or not
    - Collection of column-level lineage currently automatic in the Spark integration
    - More to come! also: can be added to custom extractors
    - Proposal
        - add 3 endpoints
            - column lineage as first-class in the lineage endpoint
            - column lineage specific endpoint
            - point in time lineage endpoint
        - currently up for review
        - describes use cases, proposed new endpoints
        - most complicated: point in time lineage
            - API requires dataset version ID (UUID)
    - Next steps
        - adding detail, use cases to the docs
    - Q&A:
        - Nikhil: possible to add point in time for jobs?
            - JLD: possible at the run level (Marquez captures lineage for each run)
            - runs point to specific versions of jobs
            - see blog post on OpenLineage site for more info
- Lineage optimization of `getLineage()` [Michael C.]
    - lineage query that uses temp tables to calculate inputs and outputs of every job
    - uses left join to select only the current version of the job
    - we were noticing that this query was taking several minutes to return due to the number of jobs (as many as 300k) in the database
    - most popular operators have no inputs or outputs (e.g., bash and python operators)
    - change: map from `job_versions_io_mapping` table
    - reduced execution time to a few seconds
    - this a "hack" because eventually we want to cover Python and bash operators
    - Q&A

        - Julien: will there be a similar query for point in time lineage?
        - MC: a different solution will be needed there
- New proposal process [Willy]
    - 4-step process
        1. open an issue (please follow the template)
        2. it will be either accepted or declined
        3. we'll add the issue to our backlog if it's accepted
        4. then we'll pin it to a milestone
    - Check out the contributing guide when working on your PR
- Optimization of query performance for facets [Willy]
    - events can get very large
    - proposal
        - raw events have to be accessed every time for facets
        - new separate tables will be used instead – e.g., `dataset_version_facets`
        - look for this change in 0.26.0
    - Q&A:
        - Nikhil: possible to search for dataset versions using the search box?
            - WL: search API currently very simple, but this could make for an interesting proposal
            - Take a look at the data model (see link in proposal)
- Runs API removal/migration [Willy]
    - we've switched over to using OpenLineage events from the Runs API
    - try it out using the `seed` command and pass in a file containing OpenLineage events
    - facets and runs displayed in the UI

July 28, 2022

Attendees:

- TSC:
    - Willy Lulciuc, Co-creator of Marquez
    - Michael Collado, Staff Software Engineer, Astronomer
- And:
    - Michael Robinson, Software Engineer, Dev. Rel., Astronomer
    - Minkyu Park, Senior Engineer, Astronomer
    - John Thomas, Software Engineer, Dev. Rel., Astronomer
    - Ross Turk, Senior Director of Community, Astronomer
    - Ryan Hatter, Customer Reliability Engineer, Astronomer
    - Howard Yoo, Staff Product Manager, Astronomer

Agenda:

1. Announcements
2. Introducing the Marquez blog
3. Architecture review: the lineage graph
4. Discussion
    a. Marquez issue #2048

Meeting:

Notes:

1. **Announcements [Willy]**
2. **Introducing the Marquez Blog [Michael R. and Ross]**
    a. new blog can be found at marquezproject.ai/blog
    b. designed and built by Ross
    c. to contribute a blog post on GitHub:
        i. write post in Markdown, place it in new directory in OpenLineage/website/contents/blog
        ii. OR: open an issue first to suggest a topic or get feedback on your idea
        iii. artwork: Ross happy to make the images; tag him
        iv. Ross also happy to document the artwork creation process for others
3. **Architecture review: the lineage graph [Willy]**
    a. What is Marquez doing in the background to surface lineage metadata at the run level during execution?
    b. What is a current lineage graph?
        i. bigraph with nodes for jobs and datasets
        ii. run-level lineage is collected from OpenLineage events
        iii. representation of job is based on datasets and the inputs and outputs they produce
        iv. datasets stitched together using OpenLineage `ID` (global and unique)
        v. versioning of jobs enabled by OpenLineage `JobVersion`
            1. Marquez keeps track of changes to code and datasets behind the scenes
    c. Marquez data model
        i. Marquez keeps track of:
            1. job versions

        2. runs of each version
        3. sources
    ii. each node represents the latest, or current, version of the job's lineage
    iii. `Job` is `ID` and arrays representing input and output datasets
d. Demo
    i. UI defaults to latest/current graph
    ii. prior versions accessible via `version history` tab
    iii. selecting a version makes another job node/datasets visible
    iv. makes "time travel" possible in your pipeline
    v. all of this possible thanks to the OpenLineage spec
e. Q & A
    i. If a job has not completed, will you not see metadata? [Howard]
        - no – a job has to complete in order for versioning logic to be applied
    ii. Is a job version associated with the code that produced it? [Ryan]
        - yes – if the code is provided as a source location facet
        - Marquez will determine if the code has changed
        - changes to schema also monitored using dataset versioning; this tied to job version

4. **Discussion**

a. Howard: issue 2048:
    i. There is an edge case (using a custom extractor) where the TaskMetadata's given input or output dataset would NOT have the fields populated (`dataset.fields = []`).
    ii. Having this type of metadata makes Marquez overwrite the existing version of the dataset with empty fields
    iii. Proposal: Marquez should try to reuse the dataset instead of rewriting
b. Agreed; question remains about how to do it [Willy]
    i. behavior reflects versioning logic
    ii. possible solution: use `null` value in OL spec rather than empty array
    iii. challenge: we want to avoid making assumptions

June 23, 2022

Attendees:

- **TSC**
    - Willy Lulciuc, Co-creator of Marquez
    - Julien Le Dem, Chief Architect, Astronomer
- **And**
    - Martin Fiser, Head of Professional Services, Keboola
    - Michael Robinson, Software Engineer, Dev. Rel., Astronomer
    - Minkyu Park, Senior Engineer, Astronomer
    - John Thomas, Support Engineer, Astronomer
    - Ross Turk, Senior Director of Community, Astronomer

Agenda:

- Announcements
- Recent release: 0.23.0
- User story by Martin Fiser (Keboola)
- Open discussion

Meeting:

Notes:

- **Announcements [Willy]**
    - Mqz/OL swag is still available!
    - Willy talked Mqz at OS Summit (LinuxCon)
- **Recent Release 0.23.0 [Michael R.]**
    - Added
        - Update docker-compose.yml: Randomly map postgres db port (#2000, @RNHTTR)
        - Job parent hierarchy (#1935 #1980 #1992, @collado-mike)
    - Changed
        - Set default limit for listing datasets and jobs in UI from 2000 to 25 (#2018, @wslulciuc)
    - Fixed
        - Return the tag for postgresql to 12.1.0 (#2015, @rossturk)
- **Keboola Use Case [Martin]**
    - Topic: OL integration with the Keboola platform
        - Overview of platform
            - modern data experience: data stack as a service
            - all-in-one service
            - writers/reverse ETL through component framework
            - enables version control, governance, etc., in workspaces
            - much metadata produced and collected, permitting visibility across entire pipeline
                - pipeline jobs
                - storage events

        - data loads/unloads
        - user-generated metadata
    - Purpose of OL integration
        - data governance to support users' feeding data to external tools
        - OL a "language" for speaking to various tools
        - offer API for OL information
        - native Keboola component
            - feeds OL information to an endpoint (e.g., Marquez)
            - can be orchestrated on customizable interval
            - supports SSH
            - exports full job information to the endpoint
    - Demo
        - users have multiple projects on the platform
        - a few hundred components are offered to users out of the box (e.g., Google Drive, SQL, Python, Google Sheets)
        - metadata manually pushable to OpenLineage endpoint
        - orchestrator could benefit from parent/job support
    - Challenges
        - need: richer metadata
            - component config
            - info about tables
        - lighter UI
            - reflects feedback about legibility
            - icon customizability
        - namespaces
            - connectivity between projects
        - more integrations
        - rounded logo
    - Q & A
        - Are you interested in contributing? [Julien]
            - would like to; possibly in the future
        - Would you like to open issues? (custom facets, UI) [Willy]
            - not currently able to
        - Are you using any integrations? java or python [Willy]
            - component can be anything in the docker container
            - multiple languages used in development
        - Customers using it already? [Conor]
            - some testing is going on
            - not in production yet
            - no plans to offer Marquez to customers
        - Does it work for every connector? [Conor]
            - each will produce at least a job
        - Auth model [Willy]
            - problem: slippery slope [Martin]
            - recommended at ingress level [Willy]
            - not a focus at the moment
            - contributions to related issues welcome
        - Is data discovery offered? [Naga]
            - built in with API
            - additional tools can be added if integration would be seamless

May 26, 2022

Attendees:

TSC:

- Willy Lulciuc, Co-creator of Marquez
- Peter Hicks, Senior Engineer, Astronomer

And:

- Ross Turk, Senior Director of Community, Astronomer
- Minkyu Park, Senior Engineer, Astronomer
- John Thomas, Support Engineer, Astronomer
- Michael Robinson, Developer Relations Engineer, Astronomer
- Joshua Wankowski, Associate Data Engineer, Northwestern Mutual
- Sam Holmberg, Software Engineer, Astronomer
- Dako Dakov, R&D Manager, VMware
- Agita Jaunzeme, Community Manager, VMware
- Radmila Radovanvic, Senior Data Engineer, Northwestern Mutual
- Gage Russell, Data Engineer, Q2
- Rae Green, Developer, Q2ebanking
- Dimira Petrova, Supervisor of Data Analytics, VMware
- Martin Fiser, Head of Professional Services, Keboola
- Antoni Ivanov, Staff Engineer, VMware

Agenda:

- Announcements
- Use cases from Northwestern Mutual and VMware
- New feature: linking job runs and datasets

Meeting:

- Recording
- Password: WMz0&@Gm

Notes:

- **Announcements [Willy]**
    - Marquez stickers are now available: https://www.astronomer.io/datakin-swag
    - Michael C. is presenting today at Airflow Summit @ 7 pm PT: https://airflowsummit.org/program/
    - Willy will be talking Mqz at Open Source Summit in June: https://sched.co/11NgS
- **Northwestern Mutual Use Case [Joshua]**
    - Big-picture role of Mqz at NWM
        - Mqz used to track data usage as a whole
        - Mqz critical at NWM to data ops, has special future here
    - Company background
        - Massive insurance co. with investment management arm
        - 150+ history with many customer touch points
        - Massive data with lots of users
    - Rationale for adoption
        - OL is where I spend most of my time
        - These tools will be the industry standards for dataset usage going forward
        - We desired one data standard, not random internal standards
    - Breakdown of use case
        - We track the HOW of usage from initial consumption to end usage
        - We record data product usage over time
        - Bonus: improved security
            - can see how/which users are actually using data
            - allows comparison to security frameworks, double-checking of work
        - Visualization is key
            - helps in building reports and modeling huge data systems
            - we can check the entire platform stack from ingest to updates, normalization, end-usage
    - Personal perspective
        - Mqz is data ops for data processing
        - Will we have a data ops center in the future like we have currently with NOCs?
        - The visual language is the key strength of the tool
        - This is the future of data
    - Q & A
        - Are screenshots available? Do you use Spark? [Naga]
            - Can't share due to proprietary concerns
        - How much data? [Naga]
            - Can't be specific, but it's a lot!
        - It's exciting to see others excited about the project. Are you using any custom integrations? [Willy]
            - Yes, custom integrations support streaming and ingestions across the platform
- **VMware use case [Antoni]**
    - Demo of VDK
    - Our motivation
        - Verification problems
        - OLMqz was the solution
        - The common standard provided by OL is essential
    - Why Mqz?
        - It's helpful in debugging complex jobs, troubleshooting
        - It's key to understanding usage for maintenance – e.g., enabling removal of irrelevant datasets, jobs
        - The shared metadata is useful
    - Diagram of architecture
    - Code demo
    - Suggestions
        - Add visualization of parent/child relationships [note: see PR 1935]
        - Make output searchable by metadata (e.g., make it possible to find all late jobs)
    - Our stack
        - Postgres, Presto, Snowflake, Greenplum db, Trino
    - Q & A
        - How many integrations in use? [Gage]
            - 100 teams, 1000s of tables
        - Are you using the Python client? [Willy]
            - Yes
        - It's amazing to get this feedback [Willy]
        - The grouping of jobs is hard, but we're addressing this
        - Feel free to open issues and contribute
- **New feature linking job runs to datasets [Peter]**
    - Recently added to jobs: created_by available on dataset views
    - Dataset versions also now available on version history tab

        - Allows for historical introspection in case of an issue
            - Allows for seeing if the code changed, for example
- **Open discussion**
    - Is anyone using the Python client for OL? [Gage]
        - Based on today's discussion, the answer is yes
    - Projects, docs are coming [Willy]
        - You can also use the Airflow integration for insight into the Python client
    - Column-level lineage has been added to OL [Willy]
        - We worked with Microsoft on the spec
        - Look for this in the API in the next few months
        - Feedback on this appreciated
    - What's in the roadmap for multi-tenancy? How can this be used in Mqz? [Naga]
        - For every event, route it through Kafka – we're working with a company to help us document this a bit more [Willy]
        - Alternate approach: use a namespace to add metadata
        - Issue with this: access control (see the project roadmap for more info)

April 28, 2022

Attendees:

TSC:

- Willy Lulciuc, Co-creator of Marquez
- Michael Collado, Staff Software Engineer, Astronomer
- Julien Le Dem, Chief Architect, Astronomer

And:

- Ross Turk, Senior Director of Community, Astronomer
- Minkyu Park, Senior Engineer, Astronomer
- John Thomas, Support Engineer, Astronomer
- Michael Robinson, Developer Relations Engineer, Astronomer
- Gage Russell, Data Engineer, Q2
- Pawe Leszczyski, Data Engineer, GetInData
- Joshua Wankowski, Associate Data Engineer, Northwest Mutual
- Dillon Stadther

Agenda:

- 0.22.0 preview [Willy]
- lifecycleStateChange support [Pawel]
- Updates to job renaming and symlinking [Michael C.]

Meeting:

Notes:

- Announcements [Willy]:
    - Cool swag is available! https://www.astronomer.io/datakin-swag
    - Willy has two talks about Marquez upcoming:
    - Airflow Summit: https://airflowsummit.org/program/
    - Open Source Summit: https://sched.co/11NgS
- 0.22.0 Preview [Willy]:
    - lifecycleStateChange support will offer visibility into dataset lifecycle changes, including deleting of tables
    - Pawel:
        - change motivated by desire for more information about datasets
        - approach started out with the Spark integration
        - still more information about lifecycle changes is possible/desirable
        - additional feature idea: notification console friendly to backend developers
    - Additional possibility: grayed out nodes on graph for deleted datasets, logging to show lifecycle history
    - Pawel: panel on website could display changes to dataset over X days
        - Agreed. Create an issue and we can build on that idea.
    - Helm chart addition
        - allows annotations, e.g. Prometheus metrics
    - Support for renaming and redirection
        - introducing job hierarchy
        - symlink will permit visibility into name changes to datasets
- Updates to job renaming and symlinking [Michael C.]
    - stemmed from desire to tie linked jobs together, e.g., jobs called by DAGs, even in cases where identical code is part of different chains
    - challenge: linking old jobs to fully qualified version
    - motivating factor: changes to job names results in junk nodes on graph
    - there was no way to remove the old job names from the graph
    - but there is frequently a need to keep track of old job names

    - hence the idea of symlinking a job
    - currently there's no API to do this
    - updating must be done manually currently
        - add the UUID of the new job to the db
        - from that point on, the job history will redirect to the new job (with a 301)
    - future: API will make this possible programmatically
    - Willy: is documentation needed for this?
        - Yes, I will post a change to the README
        - We want to do the same thing for datasets
- **Open discussion**
    - Gage: is a home repo coming?
        - Willy: Minkyu has looked into this
        - Willy: we want to add the Helm chart to the new website
        - Willy: this is on our radar
    - New release coming soon!

March 31, 2022

Attendees:

TSC:

- Willy Lulciuc, Co-creator of Marquez
- Michael Collado, Staff Engineer, Astronomer
- Julien Le Dem, Chief Architect, Astronomer
- Peter Hicks, Senior Engineer, Astronomer

And:

- Ross Turk, Sr. Director of Community, Astronomer
- Minkyu Park, Senior Engineer, Astronomer
- John Thomas, Support Engineer, Astronomer
- Michael Robinson, Developer Relations Engineer, Astronomer
- Howard Yoo, Staff Product Manager, Astronomer

Agenda:

- Website update
- Backlog and roadmap discussion
- Open discussion

Meeting:

Slides

Notes:

Announcements [Michael R.]

- Marquez stickers are now available: https://www.astronomer.io/datakin-swag
- Willy and Julien gave a talk on OpenLineage, Airflow and Marquez at Data Council Austin on March 23
- The project's Github star count stands at 983. Have you starred the project yet?
- 1k stars are a requirement for graduation status from the LFAI. The project is nearing completion of all requirements, so formal application will be possible soon.

Website [Ross]

- The project now has a new website.
- Appropriately, it's an open-source project; PRs are welcome.
- Tech: Gatsby, Github Projects
- Dev: run `yarn deploy` to work on it
- Plans: blog page. Proposals for posts welcome – post them in Slack or open a PR if you prefer.

Backlog and roadmap [Willy]

- Issue: currently, PRs are driven by a small team (e.g., Peter's view for dataset versions, Pawel's lifecycle PR)
- How to get the broader community involved? Want people to have more input/control over the issues we take up.
- Solution: Github's Roadmap feature. Milestones and releases visible there. Choose Marquez on the Projects tab.
- Process: review issues on monthly basis, move to roadmap, then release.
- Question from Howard about how to propose new features
- Follow-up work: discussion of how to prioritize issues; documentation needed about how to label new issues (e.g., as "features")
- Comment from Michael C.: it's possible to add new columns to the roadmap, in addition to new issues.

Open discussion

- Michael C.: please note issue #1928: supporting job grouping and hierarchy.

    - Problem: the project does not track parent/child job relationships, despite this nomenclature being used in OpenLineage to describe related jobs.
    - Proposal: a `parent_job_id` column should be added to the jobs table and to the runs table, both being uuids.
- Michael R.: please note that the meeting typically takes place on the 4th Thursday of each month.

February 24, 2022

Attendees:

TSC:

- Willy Lulciuc, Co-creator of Marquez
- Michael Collado, Staff Engineer, Datakin

And:

- Minkyu Park, Senior Engineer, Datakin
- Michael Robinson, Developer Relations Engineer, Datakin
- Ross Turk, VP of Marketing, Datakin

Agenda:

- Review of integrations to create runs and associate metadata with runs (replaced with OpenLineage)
- Demo: How to collect OpenLineage events with the lineage API to send metadata to Marquez
- Demo: OL Java client
- Dataset lifecycle management
- Open discussion

Meeting:

Slides

Notes:

- Announcements [Willy]
    - Release date of 0.21.0 is now 2/28
    - Confusion in the community about which Java client to use is being addressed in OpenLineage PR #480
        - We hope to have this merged for the next OL release
- Integrations and OL demo [Willy]
    - OL integration
        - Available at openlineage.io/integration/, where you can also find instructions for installing and configuring it
        - Requirements.txt needs to install airflow
        - Set OpenLineage URL to local instance of Marquez
        - Marquez is moving towards using a task listener to pull metadata in real time
        - For now use the OL Airflow DAG
        - You can still use the OL backend; there are limitations there, however
    - Spark integration
        - When doing the Spark submit command you need to provide configuration - specify the extra listener (thanks to Michael C for his work on this)
        - Point the host to your deployment
        - See the OL website for more details (openlineage.io/integration/spark-spark)
    - Upcoming: Flink and Kafka
    - Your feedback on these integrations appreciated
    - There are many connections you can use in your platform by switching over to OL to collect metadata
- OL Java client demo [Willy]
    - The Java client employs a workflow with interface
    - Definition of run method required
    - Instance of database required
    - This ex: simpleworkflow with database via newDatabase method
    - Relies on a Job class
    - In Marquez you can see the calls
    - For the code see https://github.com/DatakinHQ/demo/tree/main/custom/java/simple
- Dataset lifecycle management [Willy]
    - Marquez can now capture changes to dataset names
    - Community voiced desire for this feature
    - Marquez now supports soft deletes of datasets
    - See PR #1847
    - Support of lifecycle now more concrete: can see the phases datasets go through
- Open discussion
    - Julien and Willy will be speaking in-person at the Data Council conference in Austin next month (March 23-24)
    - Michael C. will be presenting virtually at the Subsurface LIVE conference (March 2-3); topic: Spark

January 27, 2022

Attendees:

TSC:

- Willy Lulciuc, Co-creator of Marquez
- Julien Le Dem, CTO of Datakin
- Michael Collado, Staff Engineer, Datakin
- Peter Hicks, Senior Engineer, Datakin
- Kevin Mellott, Assistant Director of Data Engineering, Northwestern Mutual

And:

- Ross Turk, VP of Marketing, Datakin
- Minkyu Park, Senior Engineer, Datakin
- John Thomas, Support Engineer, Datakin
- Michael Robinson, Developer Relations Engineer, Datakin

Agenda:

- Marquez recent releases overview [Willy]
    - Marquez release 0.21.0 overview
        - Upgrade to Java17
- Migrating integrations to OpenLineage [Willy]
- Cloud-based development instance of Marquez via Gitpod [Peter]
- Open discussion

Meeting:

Slides

Notes:

- 0.21.0 overview [Willy]
    - Features:
        - Bug fixes
        - Removal of excess code
        - Upgrade to Java17
            - API image migrated
            - Eclipse Temurin integrated
            - All CI deployment updated to support Java17
    - Discussion [Kevin, Willy, Michael C.]:
        - Support for Java client possible in lower version
        - Proposed: schedule separate meeting about this
- Migrating integrations to OpenLineage [Willy]
    - Spark library in Marquez now deprecated
    - Use of OpenLineage Spark integration recommended going forward
        - review the docs about how to configure your instance
        - remember to add underscore to marquez_airflow
    - OpenLineage integration allows task listener
        - workaround: import DAG from OpenLineage
    - See the changelog: environment variables for the Airflow instance have changed
- Cloud-based development instance of Marquez [Peter]
    - Enabled by integration of Gitpod
    - Docker image in the cloud with Marquez and UI
    - Ideal for those not ready to install everything locally or who are having issues with their OS
    - Fast (30 seconds), eliminates risk
    - API also available
    - Can be made private or public
    - Big advantage: shareable within organizations via URL
    - Supports everything one could do locally in VS Code or similar IDE
    - Discussion [Willy, Peter, Kevin, Julien]:
        - common use case: potential users want to see metadata from their org and share the tool
        - potential side-effect: increase in Docker pulls
        - availability of metrics unknown
        - email address required
- Open Discussion
    - Advantages of possible move from CircleCI to Github Actions
        - CircleCI downsides: outages, billing issues [Willy]
        - Julien proposed: moving to Github actions eventually after running both in parallel
        - Kevin asked to experiment with Github Actions and report back
    - Issue #1800: add support for table operations reported from OpenLineage
        - Formal solution needed [Willy]
        - Willy proposed: deploy in two modes and use flags (Julien agreed)

- NodeID
  - An easy win: add a field that returns a nodeID [Willy]
  - Willy proposed: prioritize in next release

Marquez Workflow Group Calendar Overview

Effective March 22, 2019: Group calendars are managed within LF AI Foundation Groups.io subgroups (mail lists); with each sub-group (mail list) having a unique group calendar. Meeting invites from these group calendars are sent to the applicable sub-group (mail list). In order to see the various group calendars you must:

- Be logged into LF AI Foundation Groups.io
- Be subscribed to the sub-group(mail-list) you're interested in
- Thereafter, you will see all the calendars for the sub-groups you subscribe to under your LF AI Foundation Group Calendar via Groups.io OR
- You can also view a specific group calendar via the Wiki (if the group has created a Wiki group calendar) whether you are a member of the sub-group (mail list) or not
  - Example: LF AI TAC Group Calendar (tac-general@lists...) via Wiki

View **Instructions on How to Subscribe to LF AI Group Calendars**

For detailed information on LF AI meeting management processes view this page: LF AI Foundation - Community Meetings and Calendars

---

Marquez Meetings List

Schedule	Title	Owner	Subgroup (mail list)	Purpose	Dial In Link
Day of Week (frequency) 00:00 AM/PM - 00:00 AM/PM (timezone)	Meeting Title (Zoom Account Used)	Meeting Owner /Moderator	marquez-mail-list@lists.lfai.foundation	Meeting Purpose	Zoom Name: https://zoom.us/...

Marquez Group Calendar

Team Calendars