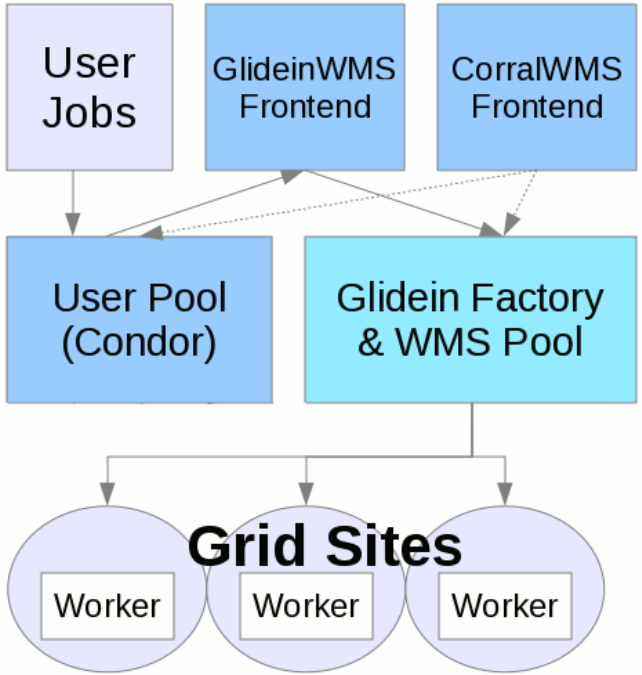
Tracing Service for Tracing-Driven Glidein Optimizations

Namratha Urs, John Tyndall, Ralph Ortiz, Marco Mambelli
Fermi National Accelerator Laboratory
Throughput Computing 2023
14 July 2023, Madison, WI

FERMILAB-SLIDES-23-167-CSAID

GlideinWMS (GWMS)

- Simplifies resource provisioning for distributed high throughput computing (dHTC)
  - From heterogenous resources to uniform virtual clusters
  - Leverages HTCondor for scheduling and job control
- Three components:
  1. **Frontend:** look for user jobs and request the Factory to provide glideins
  2. **Factory:** self-advertise, listen for requests from Frontend and submit glideins
  3. **Glideins:** provide a customized execution environment for user jobs



Animation Credits: GWMS Project Documentation

Fermilab

Glideins (aka Pilot Jobs)

- Provide tailored execution environments for user jobs to run on diverse, complex resources in a distributed setting
  - Scout for resources and validate the worker node
  - Customize the worker node (environment, libraries, containers, etc.)
  - Starts and monitors the execution of the job

Provides a tested and customized execute node to HTCondor

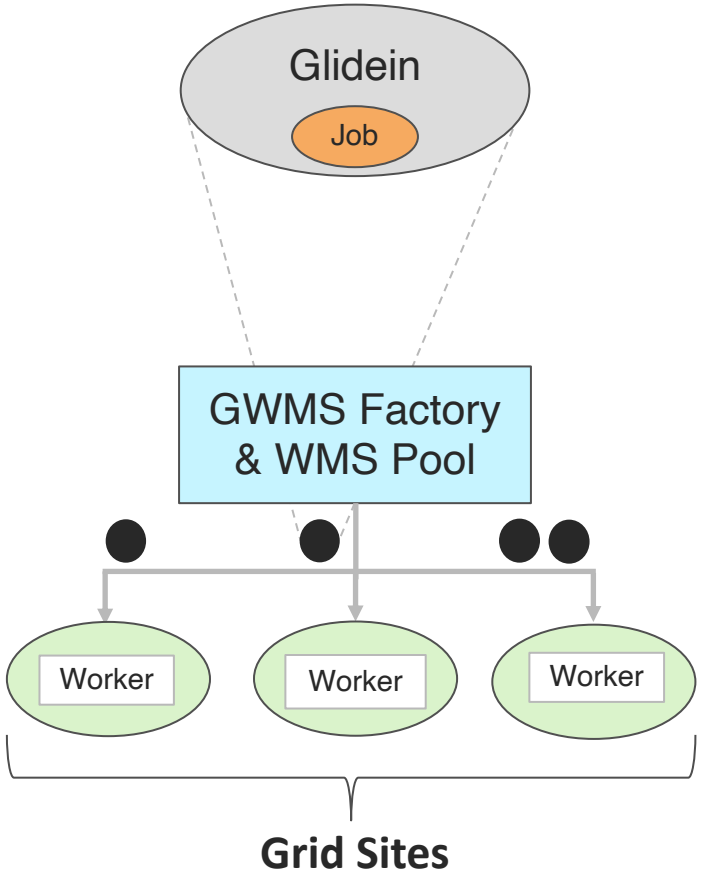
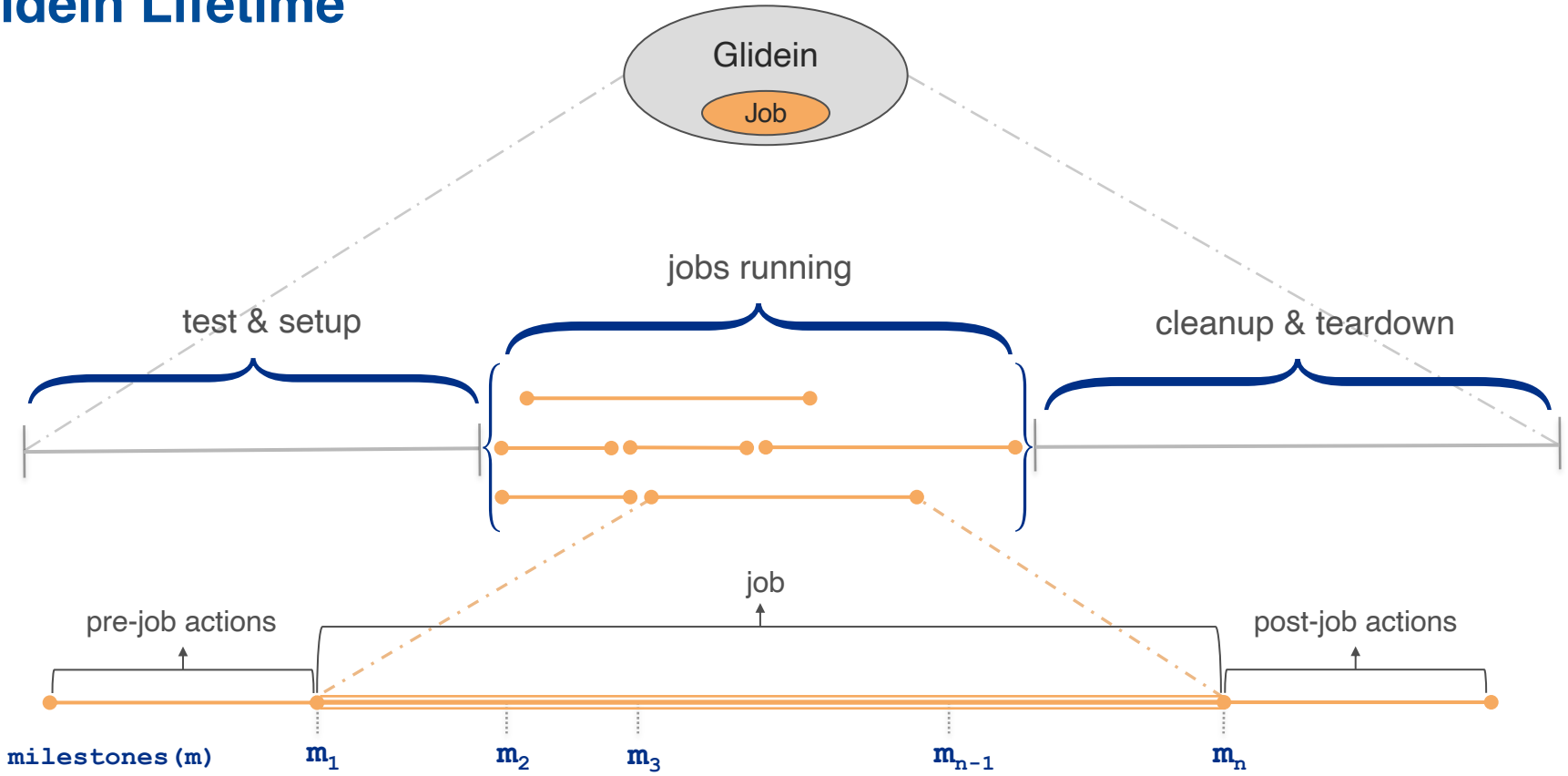


Illustration adapted from here

Fermilab

Glidein Lifetime



Fermilab

Current State of Telemetry in GWMS

- **Telemetry**: data emitted from a system about its behavior in the form of logs, metrics or traces[1]
- Different logging and monitoring capabilities exist
- Telemetry in a distributed, heterogenous ecosystem is essential
  - Cater to multi-location, multi-language, multi-channel aspects
  - Maintain security
  - Diversify paths

Examples

1. Glidein logging to standard output and error; report back to the Factory
2. A remote logging API to generate shards that coalesce to a single log sent to a web server
3. Packing HTCondor logs as a base64 encoded zip file
4. GlideinMonitor that filters, archives and unpacks glidein logs
5. Feed of logs to ElasticSearch
6. Logs fed to remote syslog by OSG

[1] https://opentelemetry.io/docs/concepts/observability-primer/#reliability--metrics

Motivation

- Existing heuristics useful for Glidein optimizations
  - E.g., how many to request, how long to wait for new jobs, the wait time for termination etc.
- Awareness of resources utilized and time exhausted can be invaluable
  - Failure and resubmission details of Glideins may help in their optimization
- Distributed high throughput computing + production setting → CHALLENGE!
  - Difficulty in understanding the cause of performance problems/degradation
  - Lack of visibility directly proportional to our inability to reproduce issues locally

Observability in distributed computing ecosystems can make a difference!

Proposed Solution

- Use of distributed tracing from two perspectives:
  1. **Generate traces for Glideins** [**internal-facing**]
     - **Goal:** Insights on end-to-end Glidein lifecycle for operators and developers
     - Find answers for optimization questions
  2. **Provide a tracing service for user jobs** [**user-facing**]
     - **Goal:** *Why is my job not running yet?*
     - Motivate users to run jobs efficiently

(Distributed) Tracing

UNITED STATES POSTAL SERVICE®

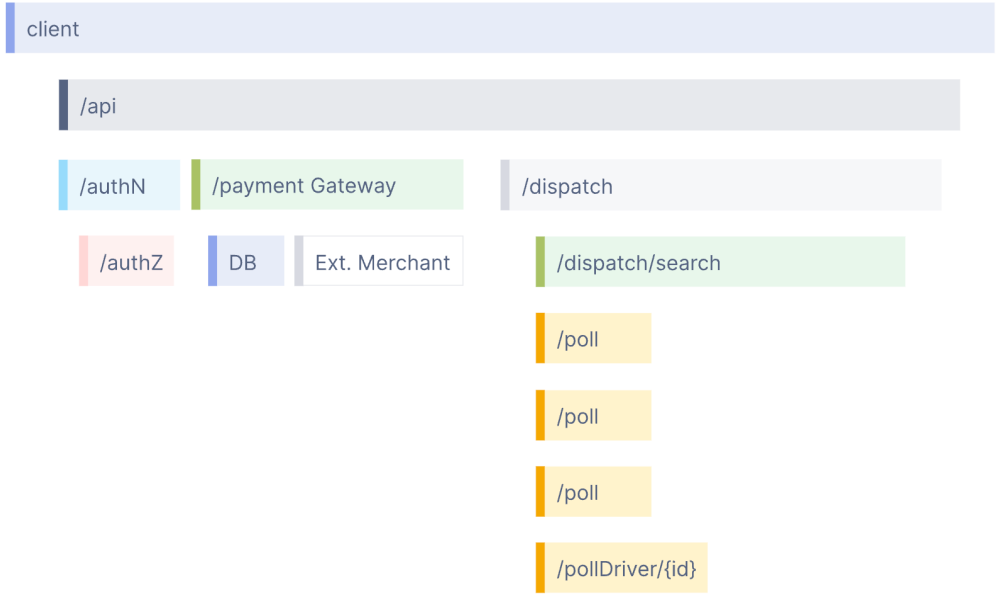
USPS Tracking Plus™ Statement
As of January 25, 2021

Tracking Number: 9205590100042400136447

Date & Time	Status of Item	Location
January 2, 2021 12:12 pm	Delivered, Front Door/Porch	SILVER SPRING, MD 20901
January 2, 2021 6:10 am	Out for Delivery	SILVER SPRING, MD 20901
January 2, 2021 5:00 am	Arrived at Post Office	SILVER SPRING, MD 20904
January 1, 2021	In Transit to Next Facility	
December 31, 2020 10:51 am	Arrived at USPS Regional Destination Facility	WASHINGTON DC NETWORK DISTRIBUTION CENTER
December 29, 2020 4:33 pm	Arrived at USPS Facility	HYATTSVILLE, MD 20785
December 29, 2020 2:50 pm	Departed USPS Regional Facility	BALTIMORE MD DISTRIBUTION CENTER
December 27, 2020 9:50 pm	Arrived at USPS Regional Destination Facility	BALTIMORE MD DISTRIBUTION CENTER
December 22, 2020 1:53 am	Departed USPS Regional Origin Facility	FORT WORTH TX DISTRIBUTION CENTER
December 19, 2020 8:16 pm	Arrived at USPS Regional Origin Facility	FORT WORTH TX DISTRIBUTION CENTER
December 19, 2020 7:01 pm	Accepted at USPS Regional Origin Facility	FORT WORTH TX DISTRIBUTION CENTER

If the USPS Tracking Plus™ Statement was prepared before the item reached its final destination, the tracking history will continue to refresh and populate status updates as the item travels through the USPS® network. Return to USPS.com to prepare a new USPS Tracking Plus™ Statement.

Graphic Credits: USPS



Graphic Credits: OpenTelemetry Docs

Fermilab

(Distributed) Tracing

- One of the three pillars of observability
- Captures “big picture” information about the system when an event flows from start to finish
- Facilitates debugging and diagnosis of common problems with software
- Useful for understanding the behavior of distributed systems
- Not the same as monitoring[2]
  - Monitoring → something is wrong
  - Observability → what is wrong and why it happened using telemetry data

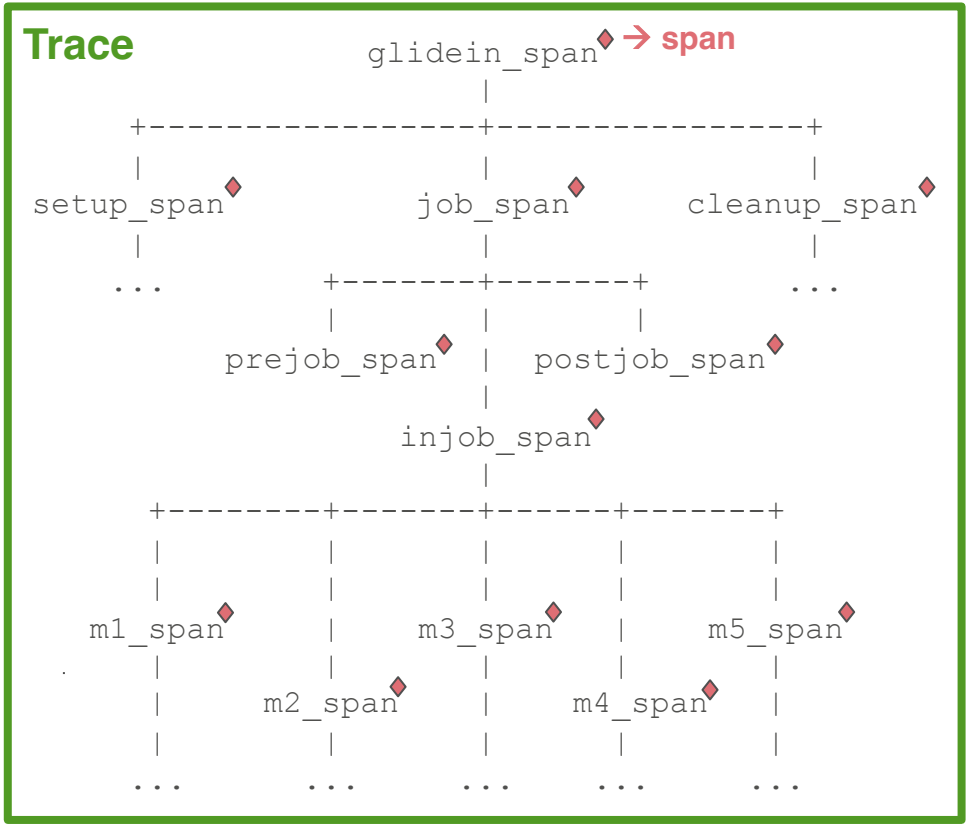


Illustration inspired from OpenTelemetry Docs

[2] https://www.crowdstrike.com/cybersecurity-101/observability/observability-vs-monitoring/

Fermilab

Generating and Processing Telemetry

1. System instrumentation

Focus: generate, collect, manage, export traces

- OpenCensus
- OpenTracing
- OpenTelemetry

2. Do something with it (aka the back-end)

Focus: storage, visualize, analyze traces

- Zipkin
- Prometheus
- Jaeger
- Tempo
- Honeycomb
- Datadog
- DynaTrace …

OpenTelemetry[3] (OTel)

- Open-source, vendor-agnostic and tool-agnostic framework
  - Specification
  - Protocol
  - Semantic conventions
  - APIs, SDKs and much more
- Data owned by YOU!
- Single set of APIs and conventions
- Standardized data format
- Easy integrations, extensible, native support across vendors

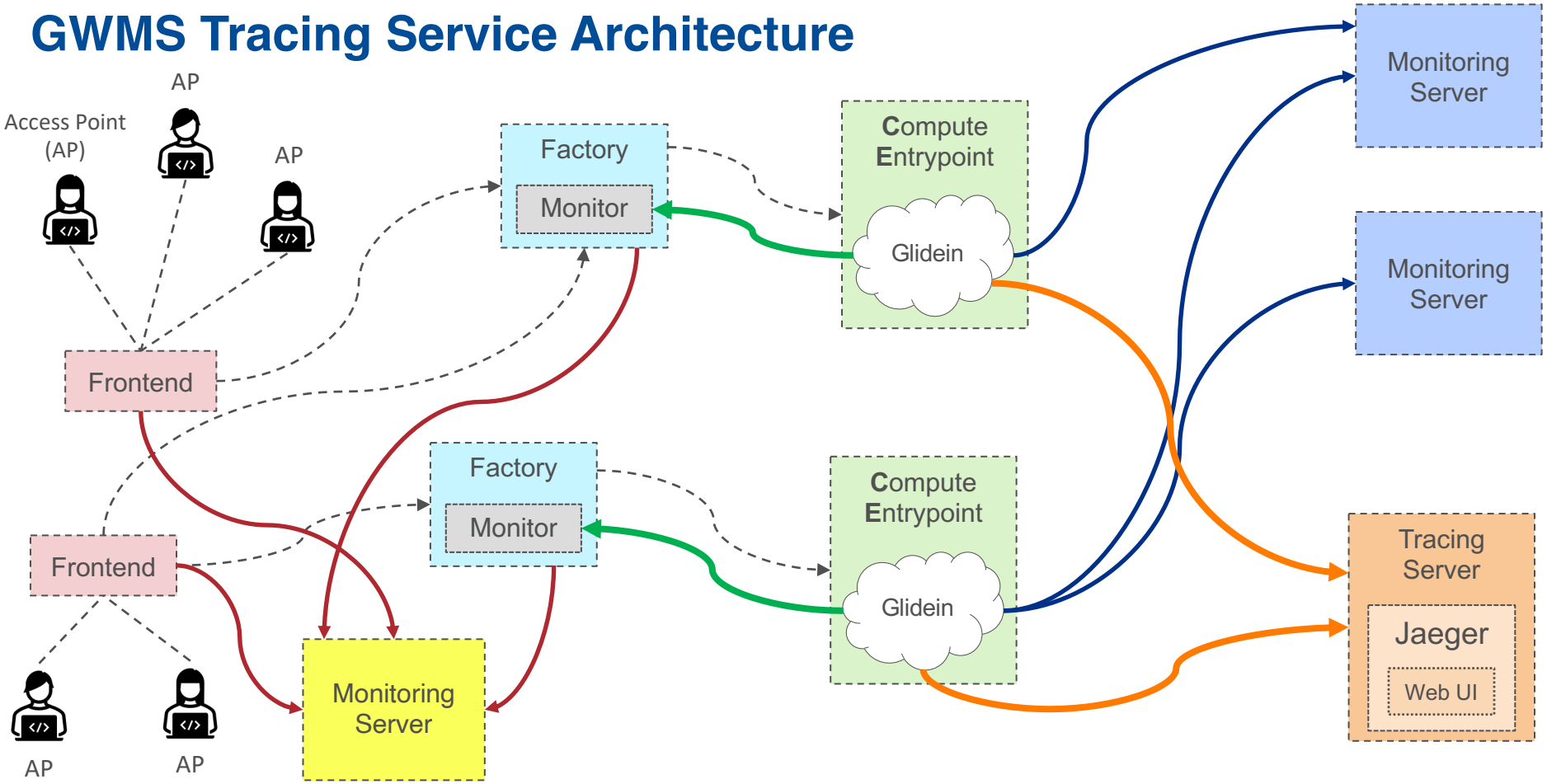
Jaeger[4]

- Open-source distributed tracing system
- Supports
  - OpenTelemetry SDKs
  - Storage backends
    - Cassandra
    - ElasticSearch
    - In-memory
    - …
  - Sampling techniques
- Minimal deployment (all-in-one binary)
- Scalable as needed

[3] https://opentelemetry.io/docs/what-is-opentelemetry/

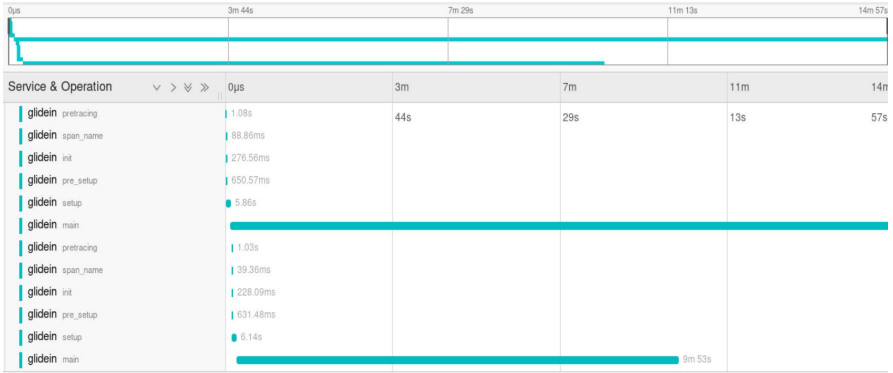
[4] https://github.com/jaegertracing/jaeger

GWMS Tracing Service Architecture



Fermilab

Next Steps and Beyond…



TraceID:SpanID	Parent Span	Entry	Client	Start Time (UTC)	Duration(µs)
6b934decf61abafeb052bb65697cf4f1:36014dffdb54eae3	53d43bdbe6b9d56b	None	None	2022-08-12 18:01:15	1171248
6b934decf61abafeb052bb65697cf4f1:2f7ea231a8d5afdc	53d43bdbe6b9d56b	None	None	2022-08-12 18:01:16	239999
6b934decf61abafeb052bb65697cf4f1:14c2abd67242d923	53d43bdbe6b9d56b	None	None	2022-08-12 18:01:16	419110
6b934decf61abafeb052bb65697cf4f1:36416b792c098bb4	53d43bdbe6b9d56b	None	None	2022-08-12 18:01:16	772407
6b934decf61abafeb052bb65697cf4f1:6bb19407bb8740d7	53d43bdbe6b9d56b	None	None	2022-08-12 18:01:16	6012456
6b934decf61abafeb052bb65697cf4f1:594ffbac394cc0e9	53d43bdbe6b9d56b	None	None	2022-08-12 18:02:12	862291
6b934decf61abafeb052bb65697cf4f1:66707a82a6bdd6c5	53d43bdbe6b9d56b	None	None	2022-08-12 18:02:12	870534
6b934decf61abafeb052bb65697cf4f1:3e448569b04fd938	53d43bdbe6b9d56b	None	None	2022-08-12 18:02:12	1106334
6b934decf61abafeb052bb65697cf4f1:271b4b12d8e0570f	53d43bdbe6b9d56b	None	None	2022-08-12 18:02:13	499740
6b934decf61abafeb052bb65697cf4f1:788e7380a8c0c9e6	53d43bdbe6b9d56b	None	None	2022-08-12 18:02:13	6065959
6b934decf61abafeb052bb65697cf4f1:1ff59c29a87a23e7	53d43bdbe6b9d56b	None	None	2022-08-12 18:01:22	821973386
6b934decf61abafeb052bb65697cf4f1:20848a139df63c89	53d43bdbe6b9d56b	None	None	2022-08-12 18:02:19	1402045341
6b934decf61abafeb052bb65697cf4f1:5adfb159176406659	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:24	283256
6b934decf61abafeb052bb65697cf4f1:5ea1fc6f007dc179	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:24	294266
6b934decf61abafeb052bb65697cf4f1:0debc7452802200c	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:24	484565
6b934decf61abafeb052bb65697cf4f1:68cf943181b5aeb9	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:24	856750
6b934decf61abafeb052bb65697cf4f1:030892546&de090c	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:24	6334235
6b934decf61abafeb052bb65697cf4f1:42b3e57958269bbd	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:32	753733
6b934decf61abafeb052bb65697cf4f1:5279b8f3858b893d	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:32	761589
6b934decf61abafeb052bb65697cf4f1:74564269e5fb7c43	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:32	950314
6b934decf61abafeb052bb65697cf4f1:26b0b991e1cc4ae4	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:32	1322935
6b934decf61abafeb052bb65697cf4f1:2961dcb6897946a7	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:33	6033886
6b934decf61abafeb052bb65697cf4f1:0a34d653e125d69d	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:39	610622377
6b934decf61abafeb052bb65697cf4f1:502697b5a867b95e	53d43bdbe6b9d56b	None	None	2022-08-12 19:41:30	624622497

1. Working prototype
   - Glidein instrumentation
   - Job instrumentation via user-facing API
   - Investigate generated traces using Jaeger UI
2. Integration with GlideinWMS
   - Knobs to trigger trace generation
   - REST interfaces to query and visualize stored trace data
3. Instrumentation for other GWMS components
4. Migrate to Grafana Tempo (long-term)

* Work in progress icon from here

Acknowledgements

- Special shout-out: John Tyndall and Ralph Ortiz!
- DOE Omni Technology Allliance Internship Program
- Kevin Retzke (FNAL) for being our go-to person for Observability best practices

Summary

- Make use of existing heuristics on Glidein behavior
- Enable Glideins to generate traces as they progress through their lifecycle
- Provide custom API to generate traces during user job execution
- Make collected trace data available via a UI and REST interface
- With distributed tracing:
  - Drive optimizations of Glideins (and eventually, other GWMS components)
  - Corroborate efficiency improvements using statistics on heuristics
  - Increase understanding of GWMS (as a framework) and in real-time dHTC settings
  - Enable observability in dHTC ecosystems

Questions and/or suggestions? Reach us at glideinwms-support@fnal.gov