

Academy Diaspora Integration

Gulesh Shukla
Master Program in Computer Science



AGENDA

1. Introduction
2. What is Diaspora?
3. What is Python logging?
4. Why do we need the integration?
5. Live Demo
6. Installation Steps
7. Next Steps
8. Acknowledgement
9. Q&A and Feedbacks



Introduction

- Pursuing MS in Computer Science.
- BS in CS from Haverford College
- 3 years as Software Engineer at Dassault Systèmes before this program
- Core interest: low-level, performance-critical systems — C/C++ at the hardware-software boundary
- Goal: real distributed systems work — observability is where any performance analysis has to start



What is Diaspora?

- kafka-based event fabric; implements Event Driven Architecture
- Uses Globus auth for authentication
- For each login:
 - Namespace
 - Used as prefix for topics create by that user
 - Access to small python client to create/clear/delete topics
 - Access to KafkaProducer / KafkaConsumer to read/write events
- Two operation crucial for this use case:
 - `producer.send(topic, event)`
 - `get_diaspora_events(topic_name=..., time_horizon=...)` (Not Diaspora)



Python Logging

- Logging module – separate what gets logs from where it goes
- Provides Loggers – a named channel with its own logging level
- Handlers – destination attached to a logger, for Diaspora it is kafka
- Formatter - turns a LogRecord into text
- Note- a single logger.info can fan out to multiple handlers without the calling code knowing or caring about where it lands
- Academy use case, add a new Handler which writes the logs to Diaspora



Why this Integration?

- Academy program are distributed by design
- Consequently, the logs stay local for each entity (Manager/Agent/Executor)
- “What happened across the whole run?” – Collect logs from N places
- This integration – collects logs in one place and Diaspora allows filtering to analysis the logs and understand bottlenecks in the whole process (next step).



Integration

- Three modules - diaspora_logger.py, diaspora_context.py, diaspora.py
- diaspora_logger.py -
 - **DiasporaHandler** — a logging.Handler subclass; publishes every log record as a structured JSON event to a Kafka topic. Works with any logger — yours, or Academy's own "academy" logger
 - **DiasporaLogConfig** — implements Academy's LogConfig interface; log_config=DiasporaLogConfig(kafka_topic). Currently works with a local executor (ThreadPoolExecutor + LocalExchangeFactory). Making it work with GCExecutor + HttpExchangeFactory requires deeper integration with Academy's logging system — that's a concrete next step.
- diaspora_context.py – provides **get_diaspora_events** method to consume the logs from a kafka topic
- diaspora.py – management CLI: **setup** registers Globus Auth credentials, **context** replays a topic's recent events, **clear** wipes a topic

Why use this integration than local Diaspora ?

- Diaspora does not know anything about academy
- Require writing your own logging Handler around KafkaProducer
- Write a topic-naming convention themselves
- Write their own offset-seeking Kafka consumer to replay history.
- With this integration, **the only change to an existing Academy program is the `log_config=` argument** — agent code is untouched,
- Distributed multi-agent debugging becomes "tail or replay one Kafka topic" instead of "SSH into N machines and grep N log files."





Live Demo

Installation

- git clone the repo
- Create/activate a python virtual envrinment
- pip install -e .
- Get Globus credentials: Globus Compute → Settings → Developers → Advanced Registration
- Copy DIASPORA_SDK_CLIENT_ID + DIASPORA_SDK_CLIENT_SECRET into .env
- One-time setup: python -m diaspora setup



Next Steps

- Performance analysis of distributed logging overhead
- Integrate DiasporaLogConfig with Academy's logging system so it works with Globus Compute by moving DiasporaLogConfig into Academy's logging configs
- Integrate with Yadu's Agent Dashboard for a live-tailing interface
- Rich filtering when consuming from a Diaspora topic (by agent, action, log level)



Acknowledgement

- Kyle Chard
- Ben Clifford
- Alok Kamatar
- Haochen Pan



Q&A And Feedback

- Try it on your own Academy program — install steps are in the README
- Hit friction anywhere in the Globus Auth / credential flow? That feedback is exactly what I need
- Which academy.* log fields would you most want to filter on?

