Data Platform Components Monitoring

Estimated time to read: 3 minutes

Overview

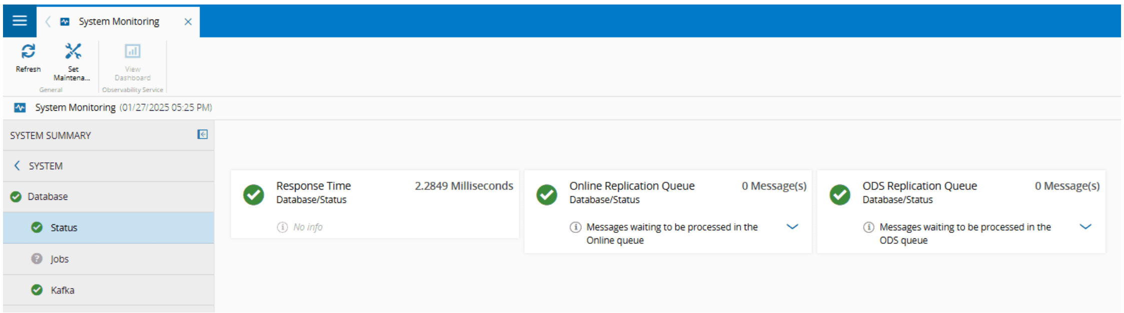
In this tutorial we describe ways to monitor **Data Platforms** components that handle data ingestion, processing, aggregation and replication.

System Monitoring

System Monitoring, in the **Administration** section, contains 6 probes that can be used to monitor **Data Platforms** components.

In the **Status** tab there are two probes:

1. **Online Replication Queue** tells us how many messages are waiting to be published to Kafka, with information regarding what needs to be done for replicating data and building the Canonical Data Model (CDM) events; these messages are produced whenever a transaction is closed in CM MES.
2. **ODS Replication Queue** provides the same information, only for messages created in ODS by the Initial Sync mechanism (which is used to run replication/CDM building for historical data).



System Monitoring (01/27/2025 05:25 PM)

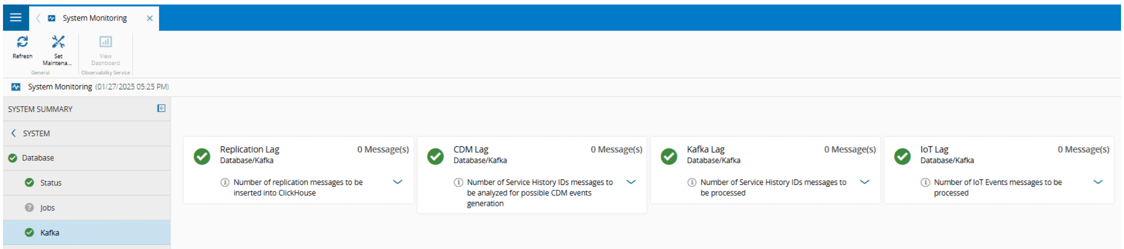
Response Time 2.2849 Milliseconds
Database/Status
No info

Online Replication Queue 0 Message(s)
Database/Status
Messages waiting to be processed in the Online queue

ODS Replication Queue 0 Message(s)
Database/Status
Messages waiting to be processed in the ODS queue

In the **Database -> Kafka** tab there are four more probes:

1. **Kafka Lag** tells us the lag building the replication documents.
2. **CDM Lag** tells us the lag building the CDM documents.
3. **IoT Lag** tells us the lag forwarding IoT events to the destination topics.
4. **Replication Lag** tells us the lag materializing replication documents in the destination database.



System Monitoring (01/27/2025 05:25 PM)

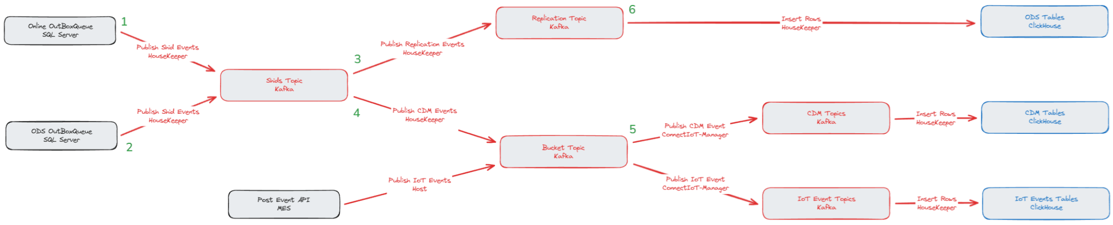
Replication Lag 0 Message(s)
Database/Kafka
Number of replication messages to be inserted into ClickHouse

CDM Lag 0 Message(s)
Database/Kafka
Number of Service History IDs messages to be analyzed for possible CDM events generation

Kafka Lag 0 Message(s)
Database/Kafka
Number of Service History IDs messages to be processed

IoT Lag 0 Message(s)
Database/Kafka
Number of IoT Events messages to be processed

The location of the probes is represented in this simplified flow chart, which can be used to help diagnose potential issues with **Data Platform** components:


Online OutBoxQueue SQL Server
1
Publish Shid Events HouseKeeper
ODS OutBoxQueue SQL Server
2
Publish Shid Events HouseKeeper
Shids Topic Kafka
3
Publish Replication Events HouseKeeper
4
Publish CDM Events HouseKeeper
Replication Topic Kafka
6
Insert Rows HouseKeeper
ODS Tables ClickHouse
Post Event API MES
Publish IoT Events Host
Bucket Topic Kafka
5
Publish CDM Event ConnectIoT-Manager
CDM Topics Kafka
Insert Rows HouseKeeper
CDM Tables ClickHouse
Publish IoT Event ConnectIoT-Manager
IoT Event Topics Kafka
Insert Rows HouseKeeper
IoT Events Tables ClickHouse

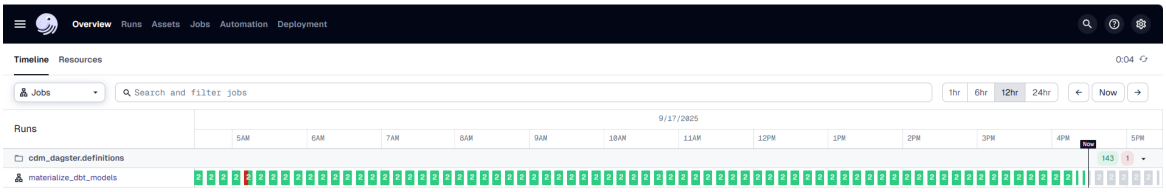
A quick way to look at it:

1. If **Online Replication Queue** shows a lot of messages waiting to be published, then it is possible that there is an issue with either **Kafka** or Data Platform component **HouseKeeper**.
2. If **ODS Replication Queue** shows a lot of messages waiting to be published, then the diagnosis is the same, i.e., a potential issue with either **Kafka** or **HouseKeeper**.
3. If **Kafka Lag** is too high, then there might be an issue with **HouseKeeper**, but it is more likely that either the MES **SQL Server** database is too slow, or more **HouseKeeper** instances are needed to keep up with the workload being generated by CM MES.
4. If **CDM Lag** is too high, then the diagnosis is similar, i.e., an issue with **HouseKeeper**, a slow **SQL Server**, or the need for more **HouseKeeper** instances.
5. If **IoT Lag** is too high, then there might be an issue with the component **ConnectIoT-Manager**.
6. If **Replication Lag** is too high, then the mostly likely diagnosis is an issue with **ClickHouse** or **HouseKeeper**.

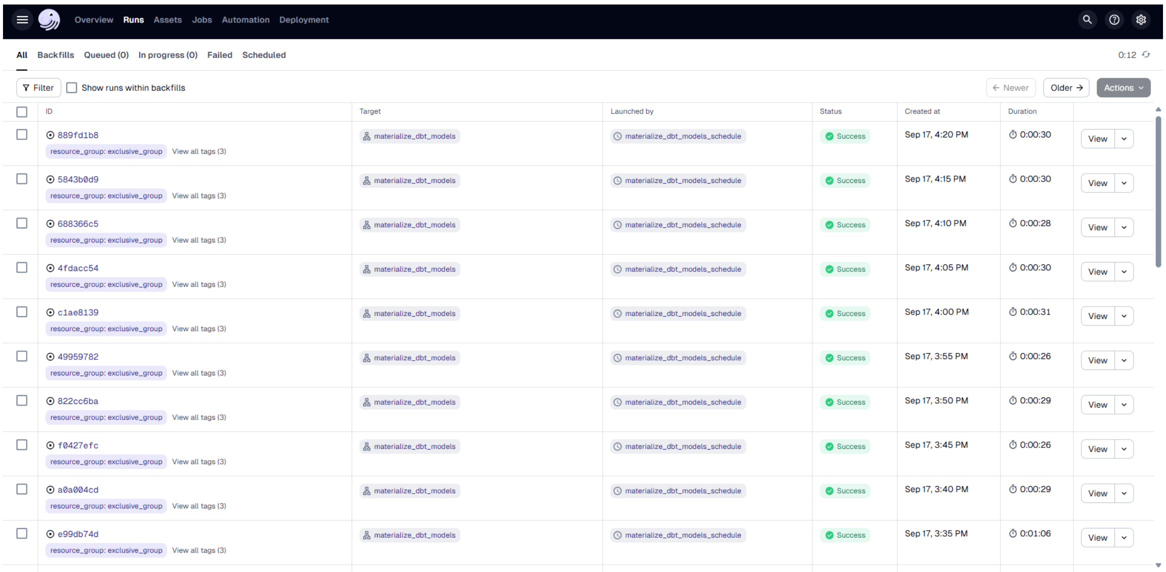
By looking at the status of these probes it should be easy to determine if the CDM building and the ODS replication are working as expected and without issues.

Data Orchestrator

Another **Data Platform** process that can be monitored in MES is the aggregation of the Data Warehouse data sets. This can be done by navigating to the **Data Orchestrator**, in the **Administration** section:

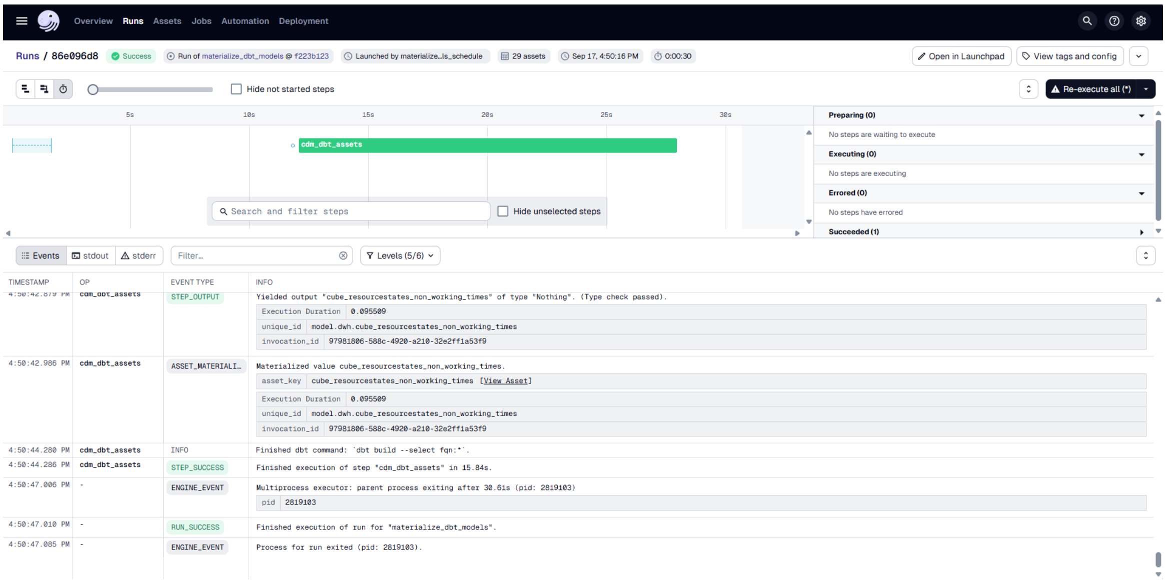


The **Data Orchestrator** is **Data Platforms** job runner. If you navigate to the **Runs** tab you can see the status of the job that aggregates the data in the Data Warehouse data sets, which goes by the name `materialize_dbt_models` and runs every 5 minutes:



☐	ID	Target	Launched by	Status	Created at	Duration	
☐	889fd1b8 resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 4:20 PM	0:00:30	View
☐	5843b0d9 resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 4:15 PM	0:00:30	View
☐	688366c5 resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 4:10 PM	0:00:28	View
☐	4fdacc54 resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 4:05 PM	0:00:30	View
☐	c1ae8139 resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 4:00 PM	0:00:31	View
☐	49959782 resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 3:55 PM	0:00:26	View
☐	822cc6ba resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 3:50 PM	0:00:29	View
☐	f0427efc resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 3:45 PM	0:00:26	View
☐	a0a004cd resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 3:40 PM	0:00:29	View
☐	e99db74d resource_group: exclusive_group View all tags (3)	materialize_dbt_models	materialize_dbt_models_schedule	Success	Sep 17, 3:35 PM	0:01:06	View

If you select **View** for a specific run, you can get details on the aggregations that were performed:



```
TIMESTAMP       OP              EVENT TYPE        INFO
4:50:42.879 PM  cdm_dbt_assets  STEP_OUTPUT       Yielded output "cube_resourcestates_non_working_times" of type "Nothing". (Type check passed).
                                                  Execution Duration  0.095509
                                                  unique_id  model.dwh.cube_resourcestates_non_working_times
                                                  invocation_id  97981806-588c-4920-a210-32e2ff1a53f9

4:50:42.986 PM  cdm_dbt_assets  ASSET_MATERIALI...  Materialized value cube_resourcestates_non_working_times.
                                                  asset_key  cube_resourcestates_non_working_times  [View Asset]
                                                  Execution Duration  0.095509
                                                  unique_id  model.dwh.cube_resourcestates_non_working_times
                                                  invocation_id  97981806-588c-4920-a210-32e2ff1a53f9

4:50:44.280 PM  cdm_dbt_assets  INFO              Finished dbt command: `dbt build --select fqn:*`.
4:50:44.286 PM  cdm_dbt_assets  STEP_SUCCESS      Finished execution of step "cdm_dbt_assets" in 15.84s.
4:50:47.006 PM  -               ENGINE_EVENT      Multiprocess executor: parent process exiting after 30.61s (pid: 2819103)
                                                  pid  2819103

4:50:47.010 PM  -               RUN_SUCCESS       Finished execution of run for "materialize_dbt_models".
4:50:47.085 PM  -               ENGINE_EVENT      Process for run exited (pid: 2819103).
```