Table of Contents

Diagnostics Information:
- Collecting Data for Troubleshooting ... 2

Troubleshooting:
- Troubleshooting with Browser Developer Tools ... 4
- Accessing MDE column data taking time ... 8
- MDE Publish job appearing hang ... 9
- Find the SQL that causing problem or failing in the MDE job ... 10
- Troubleshoot Connectivity issues ... 12

Monitoring:
- Check Health of the Cluster ... 13
- End-of-day CPD health check ... 16

Preocess:
- Applying Hotfix ... 18

Miscellaneous:
- Action ... 19

Collecting Data for Troubleshooting

CPD platform related problems:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI

Connectivity related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS)
Application logs from pod	oc cp <wdp-connect-connection pod>/logs ./<wdp-connect-connection pod>-logs oc cp <wdp-connect-connector pod>/logs ./<wdp-connect-connector pod>-logs

CPD UI related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI
HAR files	Chrome: More Tools > Developer Tools > Network > Export HAR

Data Class related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS, WKC, AE)
Application logs from swagger	https://<CPD_URL>/v2/data_profiles/api/explorer/#/Hummingbird%20tasks/getHbTaskLogs

Global Search related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS, WKC)
Pod logs	catalog-api
Application logs from swagger	https://<CPD_URL>/v2/cams/explorer/#/Assets/retrieveAssets

Metadata Enrichment related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS, WKC, AE)
Application logs from pod	oc cp <wdp-profiling pod>/logs ./<wdp-profiling pod>-logs oc cp <spark-hb-control-plane pod>/logs ./< spark-hb-control-plane pod>-logs
Job log from UI	Job log for MDE job & Screenshot of the job start time
Application logs from swagger	https://<CPD_URL>/v2/data_profiles/api/explorer/#/Hummingbird%20tasks/getHbTaskLogs

Metadata Enrichment related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS, WKC)

WKC Reporting related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS, WKC)
Reporting status from API	curl -i -k -H "content-type: application/json" -H "Authorization: bearer $Bearer_TOKEN" -X GET "https://$HOSTNAME/v3/reporting/heartbeat" curl -i -k -H "content-type: application/json" -H "Authorization: bearer $Bearer_TOKEN" -X GET "https://$HOSTNAME/v3/reporting/999/register" curl -i -k -H "content-type: application/json" -H "Authorization: bearer $Bearer_TOKEN" -X GET "https://$HOSTNAME/v3/reporting/bistatus?tenant_id=999&table_name=all"

OpenShift related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS)
Must-gather from CLI	cc adm must-gather
Collect data for projects from CLI	oc adm inspect/namespace <CPD namespace> oc adm inspect/namespace <IBM CPD operator namespace>

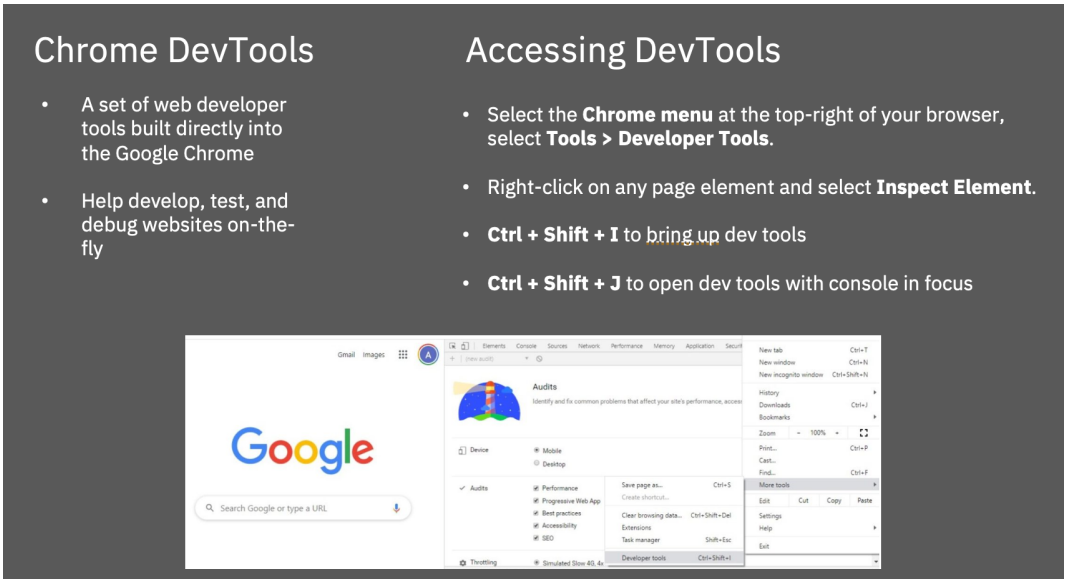
OpenShift Data Fusion related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS)
ODF must-gather from CLI	oc adm must-gather --image=registry.redhat.io/odf4/ocs-must-gather-rhel8:v4.12 --dest-dir=/tmp/odf-must-gather or oc adm must-gather --image=docker-virtual.oneartifactoryci.verizon.com/odf4/ocs-must-gather-rhel8:v4.12

IBM Storage Fusion related issues:

	Data Needs to Capture
Diagnostic job	Gather diagnostics information from CPD UI (CCS)
Collect logs from Fusion console	Fusion Web Console > “Help icon” > Support logs > Collect logs.

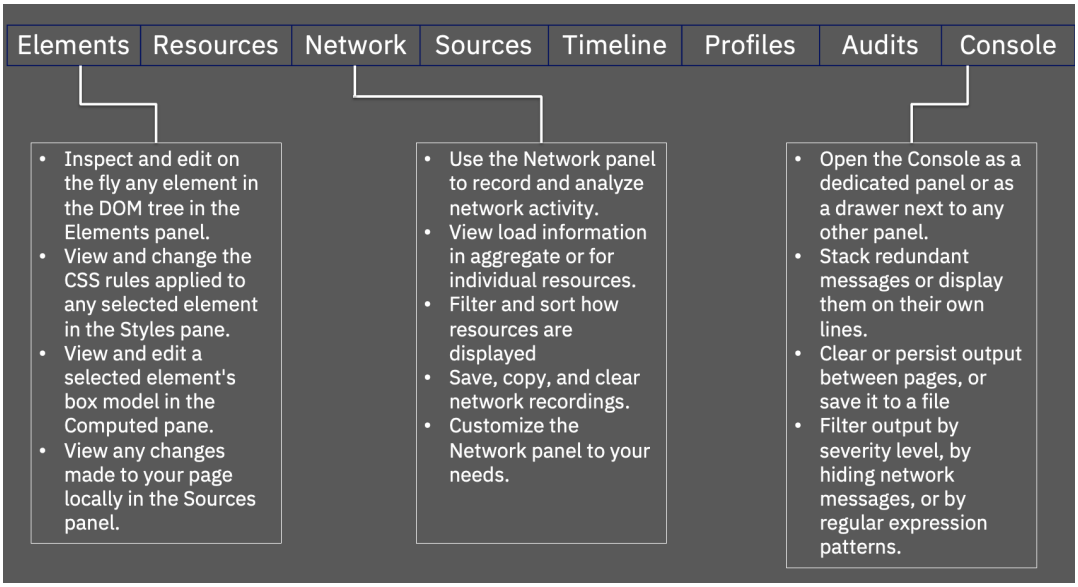
Troubleshooting with Browser Developer Tools



Mainly any CPD UI related problem can be investigate using the browser developer tool.

Each browser has their own way to use their developer tool. In this example we use Chrome as a browser.

A set of web developer tools built directly into the Google Chrome, that you can use for test, and debug issues on-the-fly.

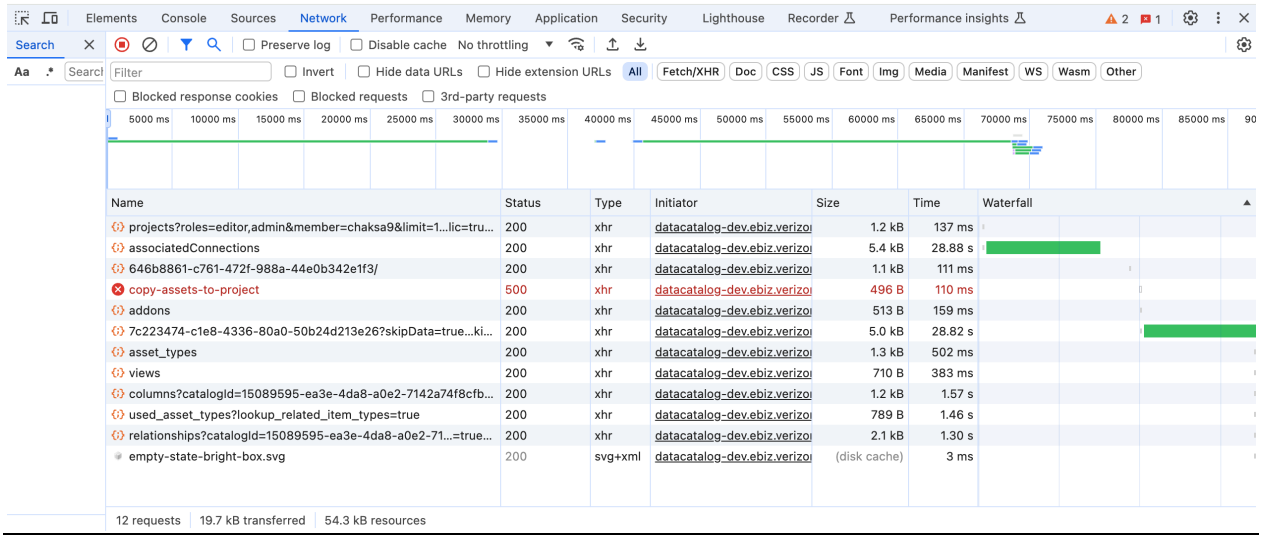


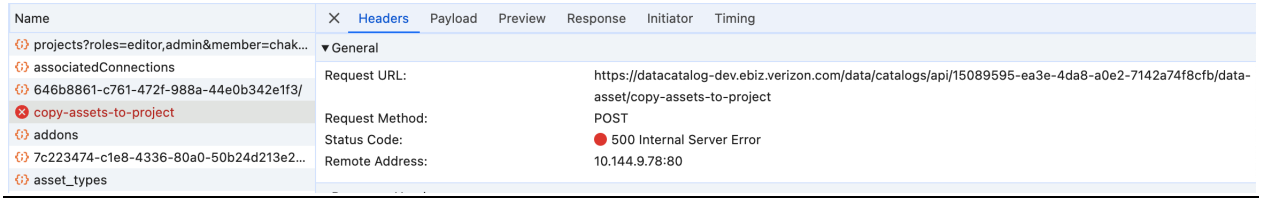
There are different dev tools options to investigate different kind of problems.

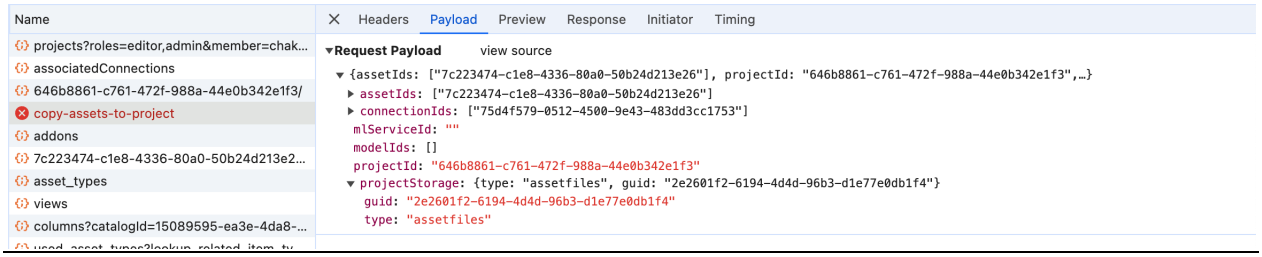
Mainly the “Network” panels are commonly used for investigating network activities, API call failure and error message retuned by a URL.

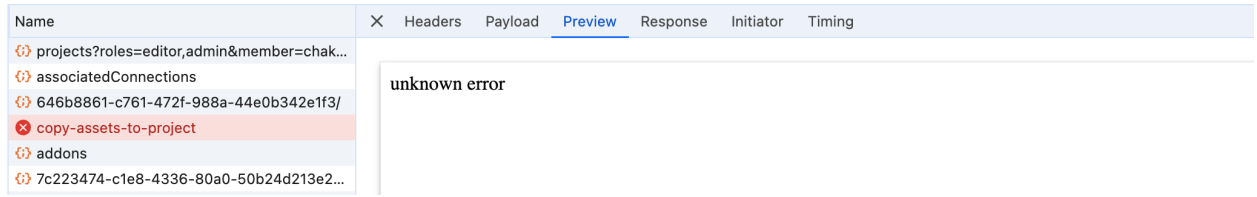
Use the Network panels from developer tool to find out:

- **API call executed**
- **HTTP status code**
- **Call execution time**
- **Error message**





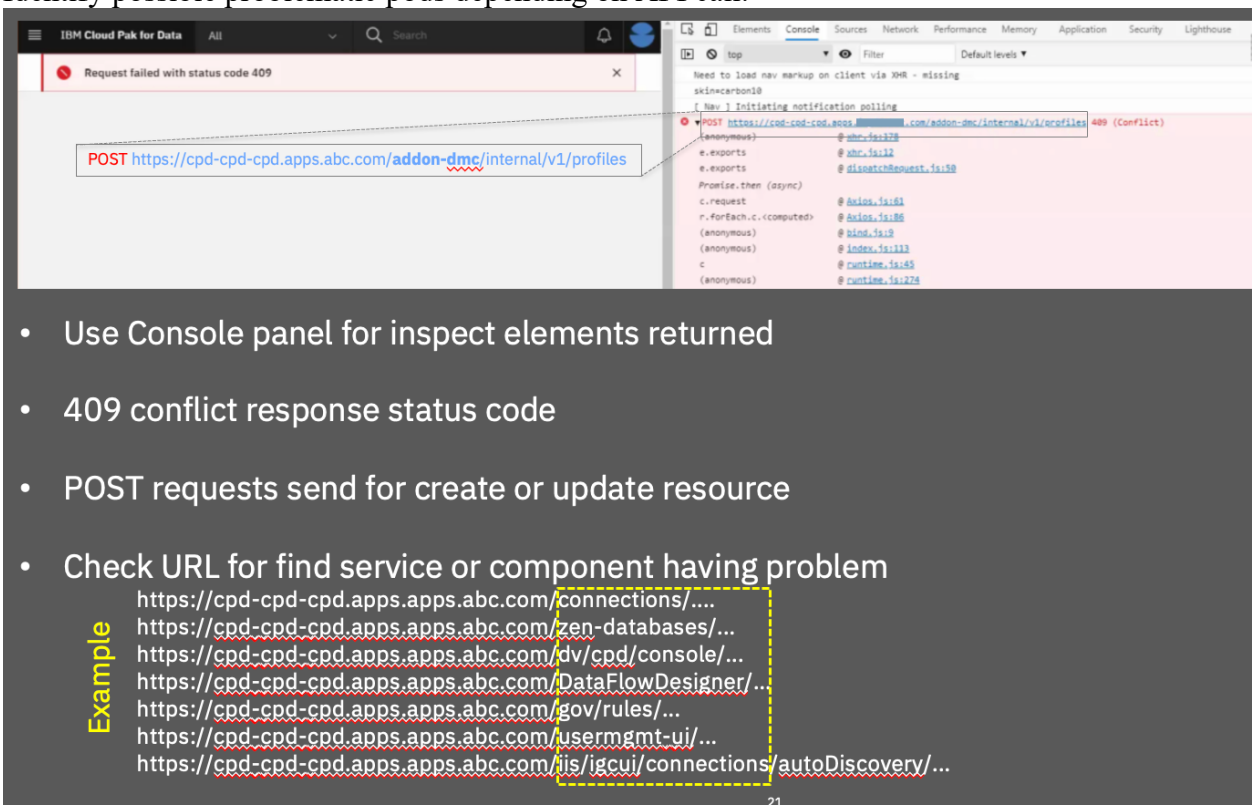




In this case Network panel recorded and analyze network activity. It shows there is one error. The individual API call returns more details about failure. The “Headers”, “Payload” and “Preview” tabs help with the troubleshooting.

Reference: https://developer.chrome.com/docs/devtools/network/reference

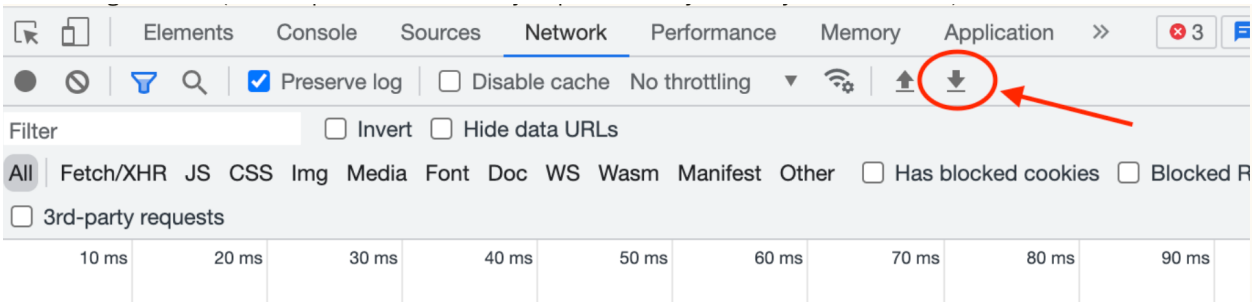
Identify possible problematic pods depending on API call.



Find Details using DevTools

- Use Network panel for inspect elements returned
- POST requests send for create or update resource
- Check URL for find service or component having problem. It is not always true you can find out service or component from URL. But in some cased it might help. Check pod reference table[A6] to find pods relation with service.

In this example, the issue is something to do with Data Management Console (DMC).



A HAR (HTTP ARchive) file is a way to preserve the interactions between a web browser and a web page. The captured performance data and other factors can be a great help during troubleshooting. It contains information like HTTP status codes for all transactions that go on behind the scenes to load a web page.

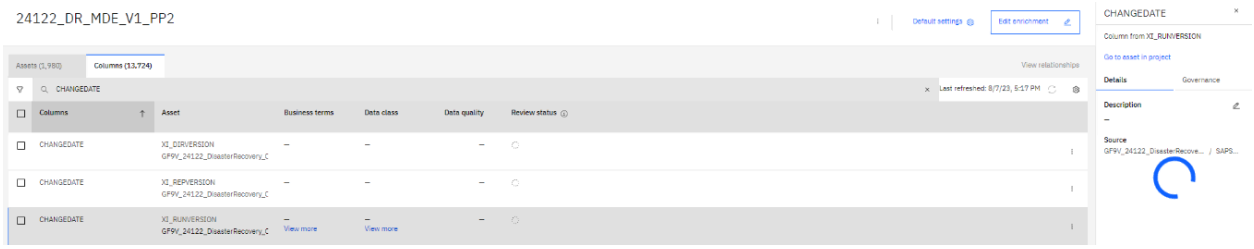
Use HAR Files to troubleshoot web pages that are failing to fully load.

- Track web browser requests
- Include response headers
- The body content
- Page load time.

Depending on the nature of the request, a HAR file can include sensitive details such as passwords, payment information, and private keys. There are open tools available to sanitize HAR file.

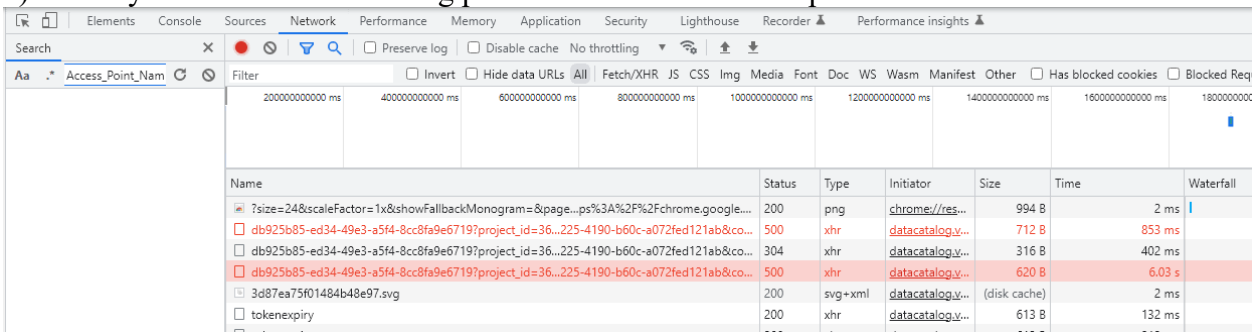
If you look at a .har file in a text editor, you will see that it's just a JSON document, containing your request and the associated response. Search through the entire archive for any and all data that may be sensitive.

Accessing MDE column data taking time

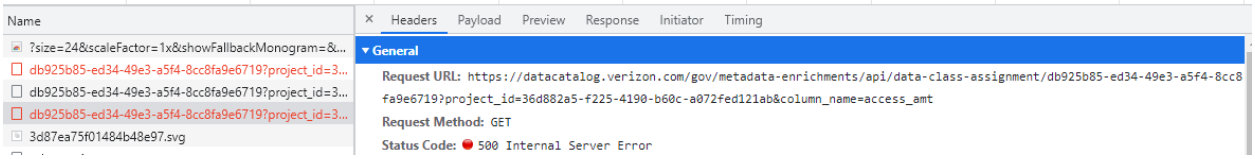


Troubleshooting:

1) For any CPD UI issue a starting point is the browser developer tool.



- Mainly the "Network" and "Console" panels are commonly used for investigating network activities, API call failure and error message retuned by a URL.
- Focused on the HTTP return/status code and elapsed time to find the problematic API. Compare elapsed time with previous successful run.



- The URL in the "Network > Headers" tab can help to identify the CPD service or component having problem.

Request URL: /gov/metadata-enrichments/api/data-class-assignment/

- Service: gov/WKC/IKC
- Component: metadata-enrichments/data-class-assignment

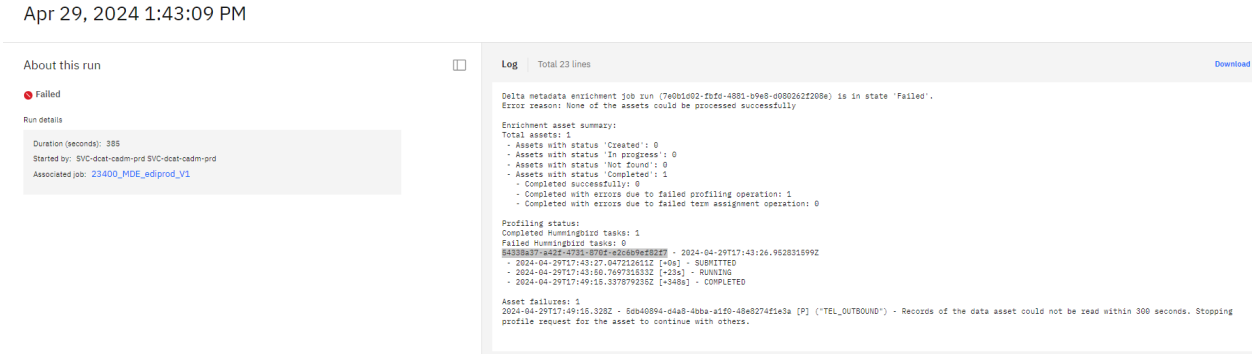
2) Check pod reference table[A6] to find pods associated to the Service/Component.
   - Are all pods healthy? [A1]
   - Any recent warning/fail event associated to the pod? [A2]
   - Has pod restarted recently? [A3]
   - Any error in the pod log during the problem? [A4]
   - What is memory, CPU consumption of pod? [A5]

MDE Publish job appearing hang

1. Check the job log for any error.
2. Monitor the job log for any progress in statistical values.
3. Check pod reference table[A6] to find pods associated to publishing
    a. Are all pods healthy? [A1]
    b. Any recent warning/fail event associated to the pod? [A2]
    c. Has pod restarted recently? [A3]
    d. Any error in the pod log during the problem? [A4]
    e. What is memory, CPU consumption of pod? [A5]

Find the SQL that causing problem or failing in the MDE job

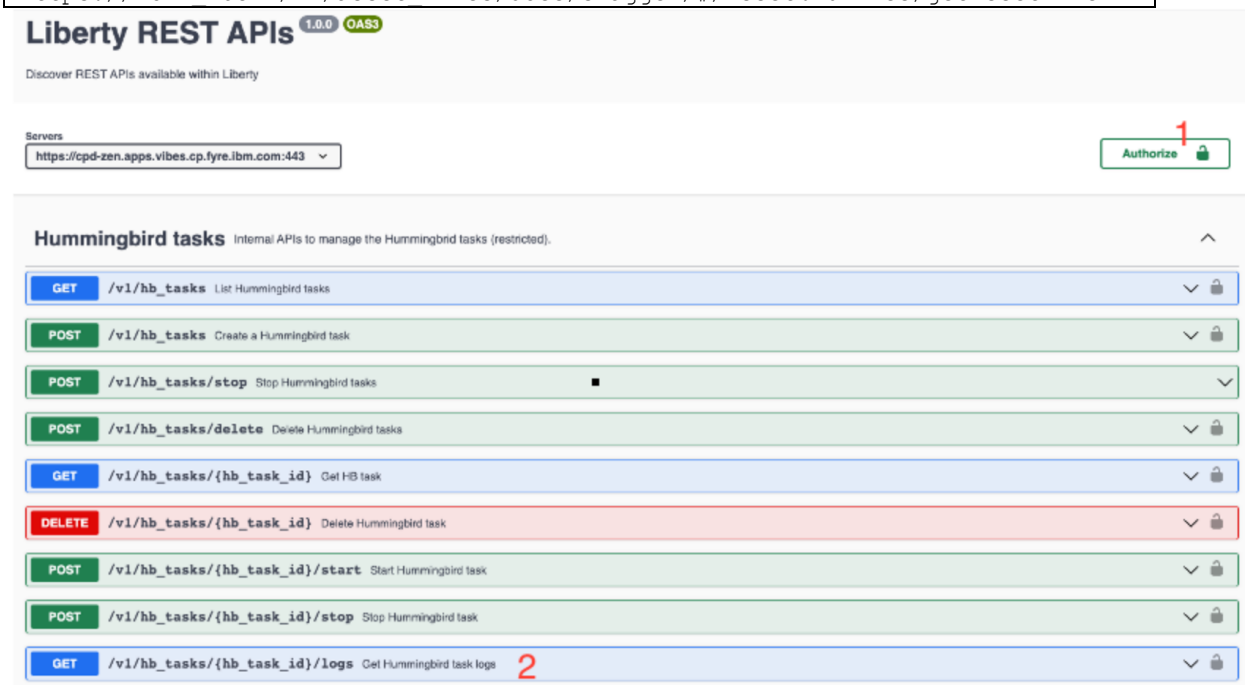
For example, an MDE job log reported profiling on the table “TLE_OUTBOUND” taking longer than 300 seconds and it skipped. You like to know the SQL used in this MDE job.



1. Find out the Hummingbird task ID from the MDE job log where the asset enrichment attempted. In this case it is “5433a37-a42f-4731-870f-e2c6b9ef82f7”. If there are multiple Hummingbird tasks, you need to consider the asset enriched time and the Hummingbird tasks start time to find the correct one.

2. Go to Hummingbird tasks explorer (Swagger) from web browser and capture the HB task log. You need to authenticate yourself on the Swagger. Secondly, make sure you have access to the respective project where MDE executed.

```
https://<CPD_HOST>/v2/asset_files/docs/swagger/#/Asset%20Files/getAssetFile
```



3. Download the HB task log and search the table to find out the SQL. For example:

```
    % grep -i TEL_OUTBOUND hb_task_log.txt
```

```
INFO: CDICO0004I: Interaction properties: {query_timeout=300,
row_limit=10001, schema_name=EDIMON, table_name=TEL_OUTBOUND}.

INFO: CDICO2019I: The specified table name is: TEL_OUTBOUND

INFO: CDICO2020I: The connector will run the statement: SELECT
"FILE_NAME", "HEADER", "CONTENT" FROM "EDIMON"."TEL_OUTBOUND" FETCH FIRST
10001 ROWS ONLY
```

Troubleshoot Connectivity issues

Example:

1. Check data source host and port accessible from bastion node:
   - `curl -v <database hostname>:<database port>`

2. Verify same data source host and port accessible from inside a wpd-connect-connector pod:
   - `oc rsh <name of wdp-connect-connector pod> bash`
   - `curl -v <database hostname>:<database port>`

3. Are all connectivity related pods healthy?
   - Primary connectivity pods: wdp-connect-connection, wdp-connect-connector
   - Other pods: zen-data-sorcerer, zen-core-api, ibm-nginx

4. Check the HAR file form browser developer tool.
   - Review network panels to find out:
     - API call executed
     - HTTP status code
     - Call execution time
     - Error message

5. Capture all wdp-connect-connector pods log
   - Search all wdp-connect-connector pod logs for some unique details from the failure to find out the “trace_ID”. For example: error message, error code, IP address etc. The “trace_ID” can help to find out all messages to a single connection attempt.

6. Capture Trace log from all wdp-connect-connector pods:
   - `oc cp < wdp-connect-connector>:/logs/trace.log ./<wdp-connect-connector>.log`
   - Find the “trace_ID” in all trace log. Pay attention on the time when problem occurred.

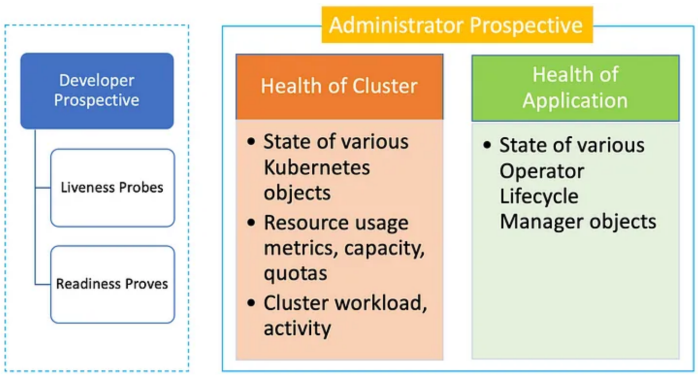
7. Capture CPD diagnostics data.

Check Health of the Cluster

Why should you run the health check?

- Administrator needs to maintain a sense of the CPD cluster's health in order to ensure they're offering **application availability and timely technical solution to end users**.
- It provides a **snapshot of the current status** and helps to identify any risk factor. If a CPD component does not work as expected that can cause larger problems down the line.
- Health check provides valuable information that allows the **administrator to make decisions about what solution** is needed for a potential problem.

What to check?

• OpenShift cluster health • Health of application	

What are you currently checking?

- The health check tool (https://github.com/IBM-ICP4D/cpd-health-check-v4)
- Running it as a cron job.

Validation	Details
Nodes status	Validate nodes in ready state
Nodes CPU utilization	Flag nodes where CPU usage higher than 80%
Nodes memory utilization	Flag nodes where memory usage higher than 80%
Nodes memory status	Identify nodes with memory pressure
Nodes disk status	Identify nodes with disk pressure
Nodes pid status	Identify nodes with PID pressure
Deployments status	Validate deployments are healthy
Statefulset status	Validate statefulsts are healthy
Replicasets status	Validate all replicasets available
Daemonsets status	Validate all daemonsets available
Routes status	CPD and Openshift console routes accessible
Openshift certificates signing status	Validate certificate signing requests in approve state
Openshift ETCD status	All ETCD members are available
Persistent volume status	Validate PVs in bound state
Persistent volume claims status	Validate PVCs in bound state
Pods status	Validate PODs in running state
High CPU consuming pods	List top 15 CPU consumed pods
High memory consuming pods	List top 15 memory consumed pods
High numner of restarted pods	List top 15 pods that restarted
External TLS Certificate	Verify TLS Certificate active
Internal TLS Certificate	Verify TLS Certificate active

New Health check tool to check health of your Red Hat OpenShift cluster and the Cloud Pak for Data platform.

- Introduced with CPD 4.8
- cpd-cli 13.1.5 command-line utility
- `cpd-cli health`
  - cluster, nodes, operators, and operands.
  - storage-validation, storage-performance

When should you use health command?

- Before you install CPD
- After CPD installed
- Before upgrade
- After upgrade

Example:

- `cpd-cli health cluster`
- `cpd-cli health cluster --verbose`
- `cpd-cli health cluster --verbose –save`
- `cpd-cli health cluster --verbose --save --log-level=trace`

```
# cpd-cli health cluster
################################################################################
CLUSTER RESOURCES
################################################################################
Health Check Report

Cluster Version Check
[SUCCESS...]

Connectivity Test
[SUCCESS...]

Machine Config Pools Healthcheck
[SUCCESS...]

Certificate Signing Request Healthcheck
[SUCCESS...]

Cluster Operator Healthcheck
[SUCCESS...]

ETCD Healthcheck
[SUCCESS...]

Cluster healthcheck info gathered successfully!
```

End-of-day CPD health check

Process of identifying potential issue:

1) What are you going to run over night?
    a. MDI/MDE/Publishing jobs

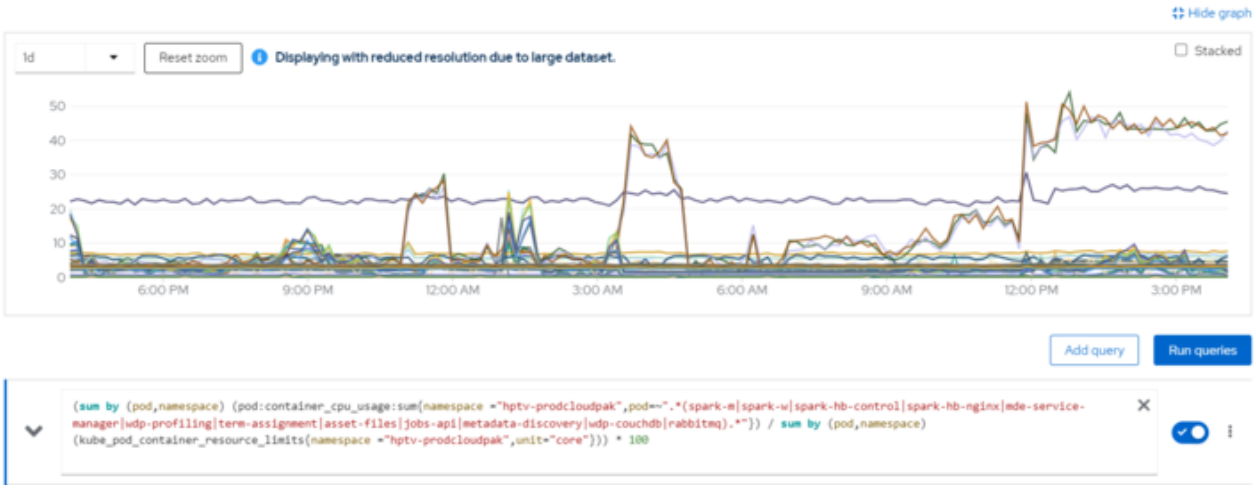
2) What are pods associated to these operations?
    a. Check pod reference table[A6] to find pods associated to the Service/Component.

3) Find status of pods using command line or CPD monitor console.
    a. Are all pods healthy? [A1]
    b. Any pods restarted recently within last 24 hours? If yes, investigate why in restart and take corrective action. [A3]

```
# oc get pods | egrep 'spark-m|spark-w|spark-hb-control|spark-hb-nginx|mde-service-manager|wdp-
profiling|term-assignment|asset-files|jobs-api|metadata-discovery|wdp-couchdb|rabbitmq'

asset-files-api-7b74f7d4cf-82hzb                                  1/1     Running     1 (4d4h ago)     4d23h
asset-files-api-7b74f7d4cf-b4xjj                                  1/1     Running     2 (3d10h ago)    4d23h
asset-files-api-7b74f7d4cf-dgbmz                                  1/1     Running     0                4d23h
asset-files-api-7b74f7d4cf-hxtpq                                  1/1     Running     2 (38h ago)      4d23h
asset-files-api-7b74f7d4cf-lr2t8                                  1/1     Running     1 (4d6h ago)     4d23h
asset-files-api-7b74f7d4cf-z5nqc                                  1/1     Running     2 (27h ago)      4d23h
jobs-api-6484667d79-5xjv8                                         1/1     Running     0                5d23h
metadata-discovery-68bfc986dd-gd8zr                               1/1     Running     0                5d19h
metadata-discovery-68bfc986dd-rxt7d                               1/1     Running     0                5d19h
metadata-discovery-68bfc986dd-w67fn                               1/1     Running     0                5d19h
rabbitmq-ha-0                                                     1/1     Running     0                5d3h
rabbitmq-ha-1                                                     1/1     Running     0                5d3h
rabbitmq-ha-2                                                     1/1     Running     0                5d3h
rabbitmq-ha-3                                                     1/1     Running     0                5d3h
spark-hb-control-plane-59759fcdb6-89678                           2/2     Running     0                8d
spark-hb-nginx-85f7c6995d-6hjbj                                   1/1     Running     0                5d18h
wdp-couchdb-0                                                     2/2     Running     0                5d21h
wdp-couchdb-1                                                     2/2     Running     0                5d21h
wdp-couchdb-2                                                     2/2     Running     0                5d21h
wdp-profiling-745bd5897b-4vhs4                                    1/1     Running     0                4d23h
wdp-profiling-745bd5897b-bj262                                    1/1     Running     0                4d23h
```

    c. What is memory, CPU consumption of pod over last 24 hours? Measure the pods resource usage from OpenShift console using metrics query. Current resource usage below 40% will be ideal.


Hide graph
1d
Reset zoom
Displaying with reduced resolution due to large dataset.
Stacked
50
40
30
20
10
0
6:00 PM
9:00 PM
12:00 AM
3:00 AM
6:00 AM
9:00 AM
12:00 PM
3:00 PM
Add query
Run queries
(sum by (pod,namespace) (pod:container_cpu_usage:sum{namespace ="hptv-prodcloudpak",pod=~".*(spark-m|spark-w|spark-hb-control|spark-hb-nginx|mde-service-manager|wdp-profiling|term-assignment|asset-files|jobs-api|metadata-discovery|wdp-couchdb|rabbitmq).*"}) / sum by (pod,namespace)
(kube_pod_container_resource_limits{namespace ="hptv-prodcloudpak",unit="core"})) * 100

Applying Hotfix

- Two step operations

WKC 4.6.5 Hotfix 1 March 2024 for Verizon Installation Instructions

Follow **Step 1** and the sub-steps to download and copy the images to a local private registry for an air-gapped environment.

Follow **Step 2** and the sub-steps to go through applying the patch using the online IBM entitled registry, or to apply the hotfix using the images downloaded to the local private registry from Step 1.

- Platform team will copy images to the private registry (artifactory)
- Identify Custom Resources (CR) impacted by the hotfix.
  - Common CRs: CCS, ZenService, WKC, AE

In Step 2

b) Run the following command to apply the patch to the Common Core Services custom resource (ccs-cr):

```
oc patch ccs ccs-cr -n ${PROJECT_CPD_INSTANCE} --type=merge -p
'{"spec":{"catalog_api_image":{"name":"catalog_master@sha256","tag":"a5f2b44fbe532b
9fecd4f67b00c937becde897e1030b7aa48087cbc2c8505707","tag_metadata":"2.0.0-
20240311173712-babe16ea94"}}}'
```

c) Run the following command to apply the patch to the WKC custom resource (wkc-cr):

```
oc patch wkc wkc-cr -n ${PROJECT_CPD_INSTANCE} --type=merge -p
'{"spec":{"metadata_discovery_image":{"name":"metadata-
discovery@sha256","tag":"02a1923656678cd74f32329ff18bfe0f1b7e7716eae5a9cb25203fcfd2
3fcc35","tag_metadata":"4.6.519"},"wkc_term_assignment_image":{"name":"term-
assignment-
service@sha256","tag":"80df5ba17fe08be48da4089d165bc29205009c79dde7f3ae3832db2edb7c
```

- Do you need downtime?
  - Pod needs to restart to load hotfix image.
  - Pod restarts can cause unexpected termination of respective applications/processes.
  - It's safe to schedule a downtime or stop using respective applications/processes.
- How much downtime do you need to schedule?
  - Step 1 can be performed ahead of time and don't need downtime.
  - Step 2 depends on CR involved in the hotfix
    - Production – CPD 4.6.5
    - CCS – around 2 hours
    - ZenService, WKC, AE – around 30 minutes
    - Downtime multiplies if CR status is in maintenance mode
- Work with project management
  - Getting approval on change request.
  - Scheduling downtime for hotfix - Step 2.
- Platform team will apply hotfix – Step 2.
- Work with testing team to validate the fix.

Action

A1. Are all pods healthy? If not:
- Investigate pod issues
- Try restart the pod

A2. Any recent warning/fail event associated to the pod? If yes:
- Get an understanding of the event and take corrective action.

A3. Has pod restarted recently? If yes:
- Need to investigate if pod restarted multiple times.
- Check pod description to find the reason of restart.
- Capture the previous pod log. Look for any error message at the end of log to figure out the cause of restart.
- Search exiting Jira for similar error message.
- If need help open support ticket with IBM along with all necessary diagnostics data.

A4. Any error in the pod log during the problem? If yes:
- Search exiting Jira for similar error message.
- If need help open support ticket with IBM along with all necessary diagnostics data.

A5. What is memory, CPU consumption of pod?
- Check the pod description for current resource configured.
- Check the pod description for restart due to lack of resource.
- Using OpenShift console monitor pod resource usage for a period.
- Check with IBM support if pod restarted due to lack of resource.

A6. Pod reference table
- https://github.com/sanjitc/Cloud-Pak-for-Data/blob/main/wkc/troubleshooting/Pods_Making_CPD.md