# Workload Optimization Manager 3.3.5 REST API Changes

June 17, 2022

This document describes issues that result in changes to the REST API. It is a running list of changes that accumulates until the next release of the API Programming Guide. When we release a new version of the programming guide, it will include these changes. At that time, this list will start with new changes to the API.

Because the API is a full reflection of the Workload Optimization Manager product, some changes to the product necessarily cause changes in the API. This can include bug fixes and improvements to the product. For example, we might add new commodities that you can get for a given entity type. Or a bug fix might change the data that is returned by an API request. We strive to make these changes backward-compatible whenever possible.

In some instances, we must make changes that are not backward-compatible. In this case, we deprecate the change for a period of time that should be sufficient for you to understand the change and implement an alternative in your scripts.

- For a list of upcoming deprecations, see Appendix: API Deprecation Contract *(on page 3)*.

## REST API Changes

- **Improvement:**

  Fix Version: 8.5.5

  **When deleting a reservation, you can now specify whether to permanently delete that reservation.**

  The API now includes the new `reservation_force_delete` parameter that you can pass along with a `DELETE` call to a specific reservation. When `reservation_force_delete = true`, the system removes the reservation permanently, no matter what state it is in. Before this release, a reservation that you have deleted via the API or user interface continues to display for deletion and is only permanently deleted after 48 hours.

- **Improvement:**

  Fix Version: 8.5.1

  **The API includes a new LABEL commodity type.**

  The `commodityType` enum includes a new `LABEL` commodity type.

- **Improvement:**

  Fix Version: 8.5.4

  **For API developers, when a reservation you created fails, the returned DTO includes more data about the failure.**

  When you create a reservation, and the reservation fails, the API returns a DTO to explain the cause of the failure. This DTO includes new fields:

  - `timespan`: Indicates whether the reservation calculation was based on realtime or historical resource utilization data. Can be one of:
    - `Current`: The calculations were based on realtime utilization data
    - `Historical`: The calculations were based on historical utilization
  - `cluster`: The name of the cluster that came the closest to being able to provide the resources that the reservation needed.
  - `hrefCluster`: A link to stats for the closest-seller cluster.
  - `hrefEntity`: A link to stats for the closest-seller entity that is a member of the closest-seller cluster.

  For example, the DTO can include data similar to the following for a placement failure:

```
"placements": {
    "failureInfos": [
        {
            "resource": "StorageAmount",
            "timespan": "Historical",
            "closestSeller": {
                "uuid": "74167780552264",
                "displayName": "QSGRID01:DC6DS2",
                "className": "Storage",
                "cluster": "StorageCluster-A",
                "hrefEntity": "/reservations/stats/74167780552264",
                "hrefCluster": "/reservations/stats/74167780552265"
            },
            "maxQuantityAvailable": 248294.59024047852,
            "quantityRequested": 273184.5625,
            "units": "MB"
        }
    ]
}
```

- **Improvement:**

  Fix Version: 8.5.4

  **When you examine a failed reservation via the API, the data now returns a reason for the failire.**

  After you create a reservation, you can GET the `reservations` endpoint for data about the current reservations in your environment. The resulting data includes a `status`. If the status is `PLACEMENT_FAILED`, Workload Optimization Manager could not place all the workloads in the reservation for some reason. Starting with version 3.3.4, the `placements` field in the data includes `invalidInfo` to give a reason for the placement failure. The reason can be:

  - `MARKET_CONNECTIVITY_ERROR`: The reservation process could not connect to the Market component for some reason.
  - `MARKET_NOT_READY`: The market does not have data from the Headroom plans for the given reservation scope.
  - `INVALID_CONSTRAINTS`: The reservation policies include conflicting or invalid constraints.

  For example, the data for a single reservation could include the following data:

```
"demandEntities": [
    {
        "template": {
            "uuid": "215167980658768",
            "displayName": "MyVmTemplate",
            "className": "VirtualMachineProfile"
        },
        "placements": {
            "invalidInfo": {
                "invalidReason": "MARKET_CONNECTIVITY_ERROR"
            }
        }
    }
]
```

# API Deprecation Contract

The following endpoints, requests, DTOs, and parameters are deprecated in the Workload Optimization Manager API. The End of Life (EOL) gives the product version in which we will finalize these changes, and will no longer offer the deprecated items.

**NOTE:**
See the instructions for alternative usage for deprecated items with a defined EOL.

## EOL: Not Determined

We deprecate the following features, but we have not determined which Workload Optimization Manager version marks their End of Life.

| | |
|---|---|
| Endpoints | None |
| Requests | None |
| Request Parameters | None |
| DTOs | ▪ DTO: `ScenarioChangeApiDTO`<br>Instruction: Use the `loadChanges` or `configChanges` parameters in the `ScenarioApiDTO`. |
| DTO Parameters | ▪ DTO: `ScenarioApiDTO`<br>Parameter Name: `changes`<br>Instruction: Use the `loadChanges` or `configChanges` parameters. |

## EOL: 8.6.1

**Behavior Changes:**

- Pagination of Returned Data from `/search` requests:

  For searches, the API paginates the returned data. If you do not specify a pagination limit, then the API returns pages set to the default limit. Starting with version 8.6.1, the default and max pagination limits for search data will be 500. You can set pagination to less than 500, but if your search results include more than 500 entries, the results will be paginated. Your scripts should always check for a pagination cursor when processing search results.

We will make the following deprecation changes in the 8.6.1 release. These deprecated features will be removed from the API at that time.

| | |
|---|---|
| Endpoints | None |
| Requests | None |

| | |
|---|---|
| Request Parameters | None |
| DTOs | ▪ `ReportScheduleApiDTO`<br>This release will remove `ReportScheduleApiDTO`. |
| DTO Parameters | ▪ DTO: `ActionApiDTO`<br>Parameter Name: `CurrentLocation`<br>Instruction: The object in this parameter will only contain the following fields:<br>– `links`<br>– `uuid`<br>– `className`<br>– `displayName`<br>– `discoveredBy`<br>– `environmentType`<br>– `vendorIDs`<br>▪ DTO: `ActionApiDTO`<br>Parameter Name: `NewLocation`<br>Instruction: The object in this parameter will only contain the following fields:<br>– `links`<br>– `uuid`<br>– `className`<br>– `displayName`<br>– `discoveredBy`<br>– `environmentType`<br>– `vendorIDs`<br>▪ DTO: `ProductVersionDTO`<br>Parameter Name: `buildUser`<br>Instruction: Deprecated with no replacement. This parameter was not set in the 8.x product family.<br>▪ DTO: `GroupApiDTO`<br>Parameter Name: `remoteId`<br>Instruction: Deprecated with no replacement. This parameter was not set in the 8.x product family.<br>▪ DTO: `UserApiDTO`<br>Parameter Names:<br>– `roleUuid`<br>– `roleName`<br>Instruction: Deprecated. Use the `roles` parameter in `UserApiDTO`. |

## EOL: 8.7.1

We will make the following deprecation changes in the 8.7.1 release. These deprecated features will be removed from the API at that time.

| | |
|---|---|
| Endpoints | None |
| Requests | ▪ `DELETE /entities/<entityId>/tags/<tagKey>`<br>Instead, use `DELETE /entities/<entityId>/tags?key=<tagKey>`<br>▪ `DELETE /groups/<groupId>/tags/<tagKey>`<br>Instead, use `DELETE /groups/<groupId>/tags?key=<tagKey>` |
| Request Parameters | None |
| DTOs | ▪ `LicenseApiInputDTO`<br>Instruction: Use `LicenseApiDTO`. |
| DTO Parameters | ▪ DTO: `HttpProxyDTO`<br>Parameter Name: `portNumber`<br>Instruction: Use `proxyPortNumber`.<br>▪ DTO: `LogEntryApiDTO`<br>Parameter Name: `reasonCommodity`<br>Instruction: Use the array, `reasonCommodities`. |

# API Deprecation History

The following endpoints, requests, DTOs, and/or parameters have been deprecated in the Workload Optimization Manager API. The End of Life (EOL) gives the product version in which we finalized these changes, and no longer support the deprecated items.

## EOL: 8.5.1

**Behavior Changes:**

- Pagination of Returned Data

  The default pagination will change for the following requests:

  – `GET /groups`

  – `GET /groups/<groupId>/members`

  – `GET /groups/<groupId>/entities`

  – `GET /targets/<targetId>/entities`

  – `GET /tags`

  – `GET /tags/{tagKey}/entities`

  To paginate returned data, the `limit`, `orderBy`, and `cursor` parameters control how the APIs paginate their data. Currently if you specify any one of these parameters, the API paginates the returned data. If you do not specify any of these parameters, the API does not paginate the output, and it returns an unlimited set of data.

  Starting with version 8.5.1, if you do not specify a `limit` for these requests, the API will paginate the data to a default limit. If your scripts make these requests without setting a limit, you should update them to check for a pagination cursor in the returned data.

We made the following deprecation changes in the 8.5.1 release. These deprecated features were removed from the API at that time.

| | |
|---|---|
| Endpoints | None |
| Requests | ▪ Request: GET /users/onetimepassword<br>Instruction: Deprecated with no replacement. This is not required for any Workload Optimization Manager activities. |
| Request Parameters | None |
| DTOs | None |
| DTO Parameters | None |

## EOL: 8.4.3

We made the following deprecation changes in the 8.4.3 release. These deprecated features were removed from the API at that time.

| | |
|---|---|
| Endpoints | None |
| Requests | None |
| Request Parameters | ▪ Request: GET /search<br>Parameter Name: regex<br>Instruction: Use the queryType parameter. |
| DTOs | ▪ DTO: StatPercentileApiDTO<br>Instruction: Use StatHistUtilizationApiDTO. |
| DTO Parameters | ▪ DTO: ActionApiDTO<br>Parameter Name: resizeToValue<br>Instruction: Use the newValue parameter.<br>▪ DTO: ActionScopesApiDTO<br>Parameter Name: limitEntities<br>Instruction: Use the limit parameter.<br>▪ DTO: BillingFamilyApiDTO<br>Parameter Name: uuidToNameMap<br>Instruction: Use the businessUnitApiDTOList parameter.<br>▪ DTO: BusinessUnitApiDTO<br>Parameter Name: hasRelatedTarget<br>Instruction: Use the associatedTargetId parameter.<br>▪ DTO: ScopeUuidsApiInputDTO<br>Parameter Name: topologyContextId<br>Instruction: Use the marketId parameter.<br>▪ DTO: StatApiDTO<br>Parameter Name: percentile<br>Instruction: Use the histUtilizations parameter.<br>▪ DTO: TargetApiDTO<br>Parameter Name: currentBusinessAccount<br>Instruction: Use the associatedTargetId parameter of the BusinessUnitApiDTO. |