



**OX2OX Migration Framework Scheduler Technical
Documentation for
2.1.0-rev25**

2023-07-05

Copyright notice

©2023 by OX Software GmbH. All rights reserved. Open-Xchange and the Open-Xchange logo are trademarks or registered trademarks of OX Software GmbH. All other company and/or product names may be trademarks or registered trademarks of their owners. Information contained in this document is subject to change without notice.

Contents

1	General Information	2
1.1	Warnings	2
1.2	Delivery Comment	2
1.3	Install Package Repository	2
1.4	Build Dependencies	2
2	Shipped Version	2
2.1	Package open-xchange-omf-orchestrator	2
2.1.1	Installation	2
2.1.2	Configuration	2
2.2	Package open-xchange-omf-scheduler	3
2.2.1	Installation	3
2.2.2	Configuration	3
A	Configuration Files	3

1 General Information

1.1 Warnings

Warning

Custom configuration or template files are potentially not updated automatically. After the update, please always check for files with a **.dpkg-new** or **.rpmnew** suffix and merge the changes manually. Configuration file changes are listed in their own respective section below but don't include changes to template files. For details about all the configuration files and templates shipped as part of this delivery, please read the relevant section of each package.

1.2 Delivery Comment

This delivery was requested with following comment:

OMF Scheduler 2.1.0 Maintenance Delivery 25

1.3 Install Package Repository

This delivery is part of a restricted software repository:

<https://software.open-xchange.com/components/omf-scheduler/stable/2.1.0/RHEL7>
<https://software.open-xchange.com/components/omf-scheduler/stable/2.1.0/DebianStretch>
<https://software.open-xchange.com/components/omf-scheduler/stable/2.1.0/DebianBuster>
<https://software.open-xchange.com/components/omf-scheduler/stable/2.1.0/DebianBullseye>

1.4 Build Dependencies

This delivery was build and tested with following dependencies:

RedHat:rhel-7,Debian:Stretch,Debian:Buster,
Debian:Bullseye

2 Shipped Version

2.1 Package open-xchange-omf-orchestrator

OMF Orchestrator CLI to interoperate with the OX2OX Migration Framework.

Version: 2.1.0-25

Type: Other

2.1.1 Installation

Install on nodes with package installer **apt-get** or **yum**:

<package installer> install open-xchange-omf-orchestrator

2.1.2 Configuration

For details, please see appendix [A](#)

/opt/open-xchange/omf/orchestrator/etc/omf-orchestrator.yml (page [5](#))

2.2 Package open-xchange-omf-scheduler

OMF Scheduler OX2OX Migration Framework Scheduler.

Version: 2.1.0-25

Type: Other

2.2.1 Installation

Install on nodes with package installer **apt-get** or **yum**:

```
<package installer> install open-xchange-omf-scheduler
```

2.2.2 Configuration

For details, please see appendix [A](#)

/opt/open-xchange/omf/scheduler/etc/omf-scheduler.yml (page [11](#))

A Configuration Files

File 1 /opt/open-xchange/omf/orchestrator/etc/omf-orchestrator.yml

```

1 micronaut:
2   application:
3     name: omf
4   http:
5     services:
6       scheduler-admin-source:
7         url: "${omf.scheduler.url}/omf/scheduler/admin/source/"
8         read-timeout: ${omf.http.read-timeout}
9         connect-timeout: ${omf.http.connect-timeout}
10        ssl:
11          enabled: ${omf.http.ssl.enabled}
12          trust-store:
13            path: ${omf.http.ssl.trust-store.path}
14            password: ${omf.http.ssl.trust-store.password}
15            type: ${omf.http.ssl.trust-store.type}
16
17       scheduler-admin-target:
18         url: "${omf.scheduler.url}/omf/scheduler/admin/target/"
19         read-timeout: ${omf.http.read-timeout}
20         connect-timeout: ${omf.http.connect-timeout}
21        ssl:
22          enabled: ${omf.http.ssl.enabled}
23          trust-store:
24            path: ${omf.http.ssl.trust-store.path}
25            password: ${omf.http.ssl.trust-store.password}
26            type: ${omf.http.ssl.trust-store.type}
27
28       scheduler-migration:
29         url: "${omf.scheduler.url}/omf/scheduler/migration/"
30         read-timeout: ${omf.http.read-timeout}
31         connect-timeout: ${omf.http.connect-timeout}
32        ssl:
33          enabled: ${omf.http.ssl.enabled}
34          trust-store:
35            path: ${omf.http.ssl.trust-store.path}
36            password: ${omf.http.ssl.trust-store.password}
37            type: ${omf.http.ssl.trust-store.type}
38
39       scheduler-batch:
40         url: "${omf.scheduler.url}/omf/scheduler/batch/"
41         read-timeout: ${omf.http.read-timeout}
42         connect-timeout: ${omf.http.connect-timeout}
43        ssl:

```

```

44         enabled: ${omf.http.ssl.enabled}
45     trust-store:
46         path: ${omf.http.ssl.trust-store.path}
47         password: ${omf.http.ssl.trust-store.password}
48         type: ${omf.http.ssl.trust-store.type}
49
50 scheduler-userinfo:
51     url: "${omf.scheduler.url}/omf/scheduler/user/"
52     read-timeout: ${omf.http.read-timeout}
53     connect-timeout: ${omf.http.connect-timeout}
54     ssl:
55         enabled: ${omf.http.ssl.enabled}
56     trust-store:
57         path: ${omf.http.ssl.trust-store.path}
58         password: ${omf.http.ssl.trust-store.password}
59         type: ${omf.http.ssl.trust-store.type}
60
61 scheduler-users:
62     url: "${omf.scheduler.url}/omf/scheduler/users/"
63     read-timeout: ${omf.http.read-timeout}
64     connect-timeout: ${omf.http.connect-timeout}
65     ssl:
66         enabled: ${omf.http.ssl.enabled}
67     trust-store:
68         path: ${omf.http.ssl.trust-store.path}
69         password: ${omf.http.ssl.trust-store.password}
70         type: ${omf.http.ssl.trust-store.type}
71
72 scheduler-scheduler:
73     url: "${omf.scheduler.url}/omf/scheduler/scheduler/"
74     read-timeout: ${omf.http.read-timeout}
75     connect-timeout: ${omf.http.connect-timeout}
76     ssl:
77         enabled: ${omf.http.ssl.enabled}
78     trust-store:
79         path: ${omf.http.ssl.trust-store.path}
80         password: ${omf.http.ssl.trust-store.password}
81         type: ${omf.http.ssl.trust-store.type}
82
83 # The OMF Scheduler Worker Monitoring HTTP REST API service configuration
84 scheduler-worker-monitoring:
85     url: "${omf.scheduler.url}/ws/omf/scheduler/workers/monitor/"
86     read-timeout: ${omf.http.read-timeout}
87     connect-timeout: ${omf.http.connect-timeout}
88     ssl:
89         enabled: ${omf.http.ssl.enabled}
90     trust-store:
91         path: ${omf.http.ssl.trust-store.path}
92         password: ${omf.http.ssl.trust-store.password}
93         type: ${omf.http.ssl.trust-store.type}
94
95 # The OMF Scheduler Window Monitoring HTTP REST API service configuration
96 scheduler-window-monitoring:
97     url: "${omf.scheduler.url}/ws/omf/scheduler/windows/monitor/"
98     read-timeout: ${omf.http.read-timeout}
99     connect-timeout: ${omf.http.connect-timeout}
100    ssl:
101        enabled: ${omf.http.ssl.enabled}
102    trust-store:
103        path: ${omf.http.ssl.trust-store.path}
104        password: ${omf.http.ssl.trust-store.password}
105        type: ${omf.http.ssl.trust-store.type}
106
107 omf:
108     http:
109         read-timeout: 30s
110         connect-timeout: 10s
111     ssl:
112         enabled: true
113         # If the scheduler does not have a valid public certificate
114         # (e.g. uses a self-signed certificate), then its certificate can be
115         # configured here.

```

```

116     trust-store:
117       path: file:/opt/open-xchange/omf/certs/scheduler.p12
118       password: secret
119       type: PKCS12
120 readonly: false
121 shell:
122   start.dir:
123   config:
124     user.dir: ${user.dir}/.omf/config
125     app.dir: /opt/open-xchange/omf/lib/scripts
126 scheduler:
127   # Credentials for the scheduler
128   # On multi-user systems, specifying the password in a configuration file
129   # with proper file system permissions is preferred to specifying it on
130   # the command line, since the command line is visible to all local users.
131   #
132   # Example:
133   # username: admin
134   # password: secret
135
136   # Location of the scheduler. Only the protocol and host name need to be
137   # specified.
138   url: "https://localhost:8443"
139 ui:
140   color: true
141   color.theme: DARK
142   unicode: true
143   expandIds: false
144   tz: UTC
145   showTz: false
146   showAgo: false
147   prettyJson: false
148   highlightJson: false
149   shell:
150     prettyJson: true
151     highlightJson: true
152     fancyPrompt: true
153     rightHandPrompt: true
154   history.file: ${user.dir}/.omf_history
155
156 logger:
157   levels:
158     # change this to TRACE to see a detailed log of the HTTP traffic between the
159     # Orchestrator and the Scheduler
160     io.micronaut.http.client: INFO

```

File 2 /opt/open-xchange/omf/scheduler/etc/omf-scheduler.yml

```

1  # https://docs.micronaut.io/latest/guide/config.html#configurationProperties
2  ---
3  micronaut:
4    # SSL configuration
5    # Required for production environments.
6    # See https://docs.micronaut.io/latest/guide/index.html#https for details.
7    ssl:
8      enabled: true
9      port: 8443
10     key-store:
11       path: file:/opt/open-xchange/omf/certs/keystore.p12
12       type: PKCS12
13       password: secret
14     server:
15       dual-protocol: true
16       port: 8080
17
18     http:
19       services:
20         # The omf-source service is use to collect health status and metrics from the Source

```

```

21 # OMF nodes. The HTTP client can be configured (ex: ssl) here by referencing
22 # https://docs.micronaut.io/latest/guide/configurationreference.html#io.micronaut.
    http.client.ServiceHttpClientConfiguration
23 # and the subsequent sections related to micronaut.http.services.*
24 omf-source:
25     connect-timeout: 30s
26     read-timeout: 120s
27     # Example SSL configuration in case a source uses a private certificate
28     # ssl:
29     #     trust-store:
30     #         path: file:/opt/open-xchange/omf/certs/source.p12
31     #         type: PKCS12
32     #         password: secret
33 omf-target:
34     connect-timeout: 30s
35     read-timeout: 120s
36 application:
37     name: omf-scheduler
38 # Configure security including basic auth: https://micronaut-projects.github.io/
    micronaut-security/latest/guide/#basicAuth
39 # Must be set to true or the Source Controller is not secure
40 security:
41     enabled: true
42     # Change the security of the open api views to anonymous so that they can be viewed
        without credentials
43 intercept-url-map:
44     - pattern: /swagger/**
45       access:
46         - isAnonymous()
47     - pattern: /swagger-ui/**
48       access:
49         - isAnonymous()
50     - pattern: /rapidoc/**
51       access:
52         - isAnonymous()
53     - pattern: /redoc/**
54       access:
55         - isAnonymous()
56 # https://docs.micronaut.io/latest/guide/index.html#_configuring_caches
57 #caches:
58     #example:
59     #    charset: UTF-8
60     #expire-after-access: 1h
61 metrics:
62     enabled: true
63 export:
64     # Creates an endpoint like http://host/prometheus - uses basic auth from
65     # credentials under scheduler.http.admin
66     prometheus:
67         enabled: true
68         step: PT1M
69         descriptions: true
70 router:
71     # Adds api versioning: https://docs.micronaut.io/latest/guide/index.html#apiVersioning
72     versioning:
73         enabled: true
74         parameter:
75             enabled: true
76             names: 'v'
77         header:
78             enabled: true
79             names: 'X-API-VERSION'
80 # Allows the openapi views to be seen
81 static-resources:
82     swagger:
83         paths: classpath:META-INF/swagger
84         mapping: /swagger/**
85     redoc:
86         paths: classpath:META-INF/swagger/views/redoc
87         mapping: /redoc/**
88     rapidoc:
89         paths: classpath:META-INF/swagger/views/rapidoc

```



```

90     mapping: /rapidoc/**
91     swagger-ui:
92       paths: classpath:META-INF/swagger/views/swagger-ui
93       mapping: /swagger-ui/**
94 ---
95 scheduler:
96   # Must be unique for each Scheduler instance.
97   # Be very careful when changing this value, as it is also used to determine the
98   # transactional ID for writing
99   # batches into Kafka topics for each Scheduler node.
100   id: 'scheduler-0'
101   hostname: ''
102   window:
103     # If more than this many batches are created as part of a new Window, don't return the
104     # list of
105     # Batch IDs as part of the Window creation result DTO, as they would just be too many
106     # to display
107     # in the first place (also affects the debug logging):
108     batch.id.threshold: 50
109     # If more than this many context IDs are assigned as part of a new Window, don't
110     # return the list of
111     # context IDs as part of the Window creation result DTO, as they would just be too
112     # many to display
113     # in the first place:
114     context.id.threshold: 50
115     monitor:
116       recalculateStats: true
117       defaultInitialRate: ''
118       defaultErrorRateThreshold: ''
119     finishOnClose: true
120   batch:
121     defaultPriority: 0
122     presync:
123       # Max number of contexts in a batch
124       size: 10
125       # Strategy to use when creating batches.
126       # Current supported strategies:
127       #   - fill-first: create batches up to the batch size then create the next batch
128       #   - fill-equal: create batches of equal size
129       strategy: fill-equal
130     cutover:
131       size: 10
132       strategy: fill-equal
133     preprovisioning:
134       size: 10
135       strategy: fill-equal
136   kafka:
137     enabled: true
138     wait: false
139     queues:
140       batch: "omf-batch"
141       response: "omf-response"
142     resize:
143       batch: true
144       response: true
145     record.header.enhance: true
146     topic:
147       list:
148         timeout: -1s
149       describe:
150         timeout: -1s
151       retry:
152         attempts: 3
153         wait: 3s
154     create:
155       # whether topics should be created when a source is created or synced (true) or
156       # whether we make use of auto-creation instead (old behaviour prior to 2.1.0-6):
157       enabled: true
158       timeout: -1s
159       partitions: 2
160       replication.factor: 0
161       config:

```

```

157         retention.ms: 432000000
158     rest:
159         lastNextBatches: 10
160     workers:
161         allow:
162             # allow Worker shutdown via the REST API
163             shutdown: false
164             # allow overriding the Sources Workers subscribe to via the REST API
165             changeSources: false
166     http:
167         users:
168             # Users defined here can have roles OMF_ADMIN or OMF_USER. Users without a role
169             # automatically have role OMF_USER
170             # assigned. Users with the role OMF_ADMIN have access to every REST API method.
171             # OMF_USER role is restricted to
172             # a subset of the REST API.
173             admin:
174                 password: secret
175                 role: OMF_ADMIN
176             omfuser:
177                 password: secret
178                 role: OMF_USER
179         workers:
180             # a list of valid tokens for workers to use
181             # can generate some with `pwgen -n1 64 1`
182             - 'bei90hchie8nai5em5asee9wohz6uu0ahshaigh0bia1isi4liKi0iwo8bu2niey'
183             - 'eeW5moi6eleik0ziw7ivaen3phoi6oolae9aht2ox9uY0ebiVaht0gashoof1rai'
184             - 'oogheePhaeB5iezairu6ongee8Ee6faePashi9thietahG0bieghiixeivahroco'
185         admin.controller.path: /omf/scheduler/admin
186         migration.controller.path: /omf/scheduler/migration
187         user.controller.path: /omf/scheduler/user
188         batch.controller.path: /omf/scheduler/batch
189         scheduler.controller.path: /omf/scheduler/scheduler
190     metrics:
191         migration:
192             enabled: true
193         leadership:
194             enabled: true
195         batchresponse:
196             enabled: true
197             percentiles: true
198         scheduler:
199             enabled: true
200         schedulers:
201             enabled: true
202         source:
203             enabled: true
204         target:
205             enabled: true
206         monitor:
207             windows:
208                 enabled: true
209                 interval: 5m
210                 delay: 30s
211             batches:
212                 enabled: true
213                 interval: 5m
214                 delay: 30s
215             sources:
216                 enabled: true
217                 interval: 5m
218                 delay: 30s
219             targets:
220                 enabled: true
221                 interval: 5m
222                 delay: 30s
223         contextmappings:
224             enabled: true
225             interval: 5m
226             delay: 30s
227         usermappings:
228             enabled: true

```

```

227         interval: 60m
228         delay: 5m
229     migrationevents:
230         enabled: true
231         interval: 5m
232         delay: 30s
233     workers:
234         enabled: true
235         idle.since: [5m, 10m, 30m]
236     orphan-check:
237         context.batch.size: 50
238     database:
239         migration:
240             allowRead: false
241             allowWrite: false
242         scheduler:
243             allowRead: false
244             allowWrite: false
245             useSkipLocked: false
246     batchSkipList:
247         reap:
248             windows: true
249             batches: true
250 ---
251     jackson:
252         bean-introspection-module: true
253         serialization:
254             indent-output: true
255             writeDatesAsTimestamps: false
256 ---
257     datasources:
258         # Used to persist scheduling data
259         scheduler:
260             # url should use createDatabaseIfNotExist=true if the database will not
261             # already exist: https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-reference-
262             # configuration-properties.html
263             url: jdbc:mysql://localhost:3306/scheduler?createDatabaseIfNotExist=true
264             username: root
265             password: my-secret-pw
266             driverClassName: org.mariadb.jdbc.Driver
267             dialect: MYSQL
268             maximumPoolSize: 10
269             maxLifetime: 180000
270         # Used to create/drop databases for sources. This is not really the "default" data
271         # source
272         # but we need to use default because of bug https://github.com/micronaut-projects/
273         # micronaut-data/issues/598s
274         default:
275             url: jdbc:mysql://localhost:3306/
276             username: root
277             password: my-secret-pw
278             driverClassName: org.mariadb.jdbc.Driver
279             dialect: MYSQL
280             maximumPoolSize: 5
281             maxLifetime: 180000
282 ---
283     endpoints:
284         loggers:
285             enabled: true
286             sensitive: true
287         health:
288             discovery-client:
289                 enabled: false
290             sources:
291                 enabled: false
292             targets:
293                 enabled: false
294         prometheus:
295             enabled: true
296             sensitive: false
297     metrics:
298         enabled: true

```

```

296     sensitive: false
297 liquibase:
298     # fails with missing transition, might be fixed in later Micronaut releases
299     enabled: false
300 info:
301     enabled: true
302     sensitive: true
303     sourceCodeOrigin:
304         enabled: true
305         location: file:/opt/open-xchange/omf/scheduler/share/SourceCodeOrigin.txt
306 ---
307 zookeeper:
308     server: zookeeper:2181
309     blockUntilConnected: true
310     maxConnectedWaitTime: 30s
311     sessionTimeout: 1m
312     connectionTimeout: 15s
313     maxCloseWait: 15s
314     waitForShutdownTimeout: 15s
315     connectionRetry:
316         baseSleepTime: 5s
317         maxSleepTime: 30s
318         maxRetries: 50
319 ---
320 kafka:
321     bootstrap:
322         servers: kafka-1:9092, kafka-2:9092, kafka-3:9092
323     producers:
324         batch-producer:
325             enable.idempotence: true
326             # This enables transactions for the Batch Producer
327             # The value must be unique per application, but should
328             # not change for the same app after a crash, etc.
329             # Note that if you only want to allow a single Scheduler instance to be capable of
330             # writing Batches into the Kafka topics, then change this to be the same value
331             # across
332             # all Scheduler instances as Kafka will fence them (see PRODUCER_FENCED).
333             # But if you want all Scheduler instances to be able to write batches into Kafka
334             # topics,
335             # use a unique value for each Scheduler instance:
336             transactional.id: ${scheduler.id}
337     #consumers:
338     #response-consumer:
339 ---
340 mail:
341     # whether to send emails
342     enabled: false
343     window:
344         # whether to send emails when a Window succeeds:
345         success: false
346         # whether to send emails when a Window fails:
347         failure: true
348         # whether to send emails when a Window is canceled:
349         canceled: true
350     # mandatory, must be set to be able to send emails and it
351     # must be a valid email address in the form localpart@domain, or sending will fail:
352     from: ${scheduler.id}@example.com
353     # whom to send those mails to (can be a comma separated list):
354     to:
355     cc:
356     bcc:
357     # text to include in the subject line, wrapped in []:
358     subject.id:
359     smtp:
360         host: localhost
361         port: 25
362         # leave empty for no authentication:
363         username:
364         password:
365         # SMTP, SMTPS or SMTP_TLS (SMTP with mandatory StartTLS):
366         transport: SMTP
367         # whether to allow SMTP without StartTLS:

```

```
366     smtp.plain: true
367     tls:
368         # whether to trust all SMTP server keys
369         trustall: false
370         # whether to verify SMTP server keys
371         verify: true
372 ---
373 liquibase:
374     datasources:
375         scheduler:
376             change-log: 'classpath:liquibase/scheduler/liquibase-changelog.xml'
377 ---
378 logger:
379     levels:
380         ROOT: INFO
381         com.openxchange: INFO
382         omf: INFO
383         omf.scheduler.admin.AuthenticationProviderUserPassword: WARN
384         org.apache.kafka.clients.consumer.ConsumerConfig: WARN
385         omf.scheduler.security: INFO
386 ---
```