



**OX2OX Migration Framework Scheduler Technical
Documentation for
2.1.0-rev15**

2022-07-27

Copyright notice

©2022 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

This preview delivery is not for productive usage and not affected by service-level agreements.

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 Preview Delivery 15

1.3 Install Package Repository

This delivery is part of a restricted preview software repository:

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

1.4 Build Dependencies

This delivery was build with following dependencies:

RedHat:RHEL-7,Debian:Stretch,Debian:Buster

2 Shipped Version

2.1 Package open-xchange-omf-orchestrator

OMF Orchestrator CLI to interoperate with the OX2OX Migration Framework.

Version: 2.1.0-15

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: 2.1.0-15

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 [10](#))

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       # The OMF Scheduler HTTP REST API service configuration
7       scheduler-admin-source:
8         url: "${omf.scheduler.url}/omf/scheduler/admin/source/"
9         read-timeout: ${omf.http.read-timeout}
10        connect-timeout: ${omf.http.connect-timeout}
11        ssl:
12          enabled: ${omf.http.ssl.enabled}
13          trust-store:
14            path: ${omf.http.ssl.trust-store.path}
15            password: ${omf.http.ssl.trust-store.password}
16            type: ${omf.http.ssl.trust-store.type}
17
18       scheduler-admin-target:
19         url: "${omf.scheduler.url}/omf/scheduler/admin/target/"
20         read-timeout: ${omf.http.read-timeout}
21         connect-timeout: ${omf.http.connect-timeout}
22         ssl:
23           enabled: ${omf.http.ssl.enabled}
24           trust-store:
25             path: ${omf.http.ssl.trust-store.path}
26             password: ${omf.http.ssl.trust-store.password}
27             type: ${omf.http.ssl.trust-store.type}
28
29       # The OMF Scheduler Migration HTTP REST API service configuration
30       scheduler-migration:
31         url: "${omf.scheduler.url}/omf/scheduler/migration/"
32         read-timeout: ${omf.http.read-timeout}
33         connect-timeout: ${omf.http.connect-timeout}
34         ssl:
35           enabled: ${omf.http.ssl.enabled}
36           trust-store:
37             path: ${omf.http.ssl.trust-store.path}
38             password: ${omf.http.ssl.trust-store.password}
39             type: ${omf.http.ssl.trust-store.type}
40
41       # The OMF Scheduler Migration HTTP REST API service configuration
42       scheduler-userinfo:
43         url: "${omf.scheduler.url}/omf/scheduler/user/"

```

```

44     read-timeout: ${omf.http.read-timeout}
45     connect-timeout: ${omf.http.connect-timeout}
46     ssl:
47         enabled: ${omf.http.ssl.enabled}
48         trust-store:
49             path: ${omf.http.ssl.trust-store.path}
50             password: ${omf.http.ssl.trust-store.password}
51             type: ${omf.http.ssl.trust-store.type}
52
53     # The OMF Scheduler Monitoring HTTP REST API service configuration
54     scheduler-monitoring:
55         url: "${omf.scheduler.url}/ws/omf/scheduler/workers/monitor/"
56         read-timeout: ${omf.http.read-timeout}
57         connect-timeout: ${omf.http.connect-timeout}
58         ssl:
59             enabled: ${omf.http.ssl.enabled}
60             trust-store:
61                 path: ${omf.http.ssl.trust-store.path}
62                 password: ${omf.http.ssl.trust-store.password}
63                 type: ${omf.http.ssl.trust-store.type}
64
65 omf:
66     http:
67         read-timeout: 30s
68         connect-timeout: 10s
69         ssl:
70             enabled: true
71             # If the scheduler does not have a valid public certificate
72             # (e.g. uses a self-signed certificate), then its certificate can be
73             # configured here.
74             trust-store:
75                 path: file:/opt/open-xchange/omf/certs/scheduler.p12
76                 password: secret
77                 type: PKCS12
78     readonly: false
79     shell:
80         start.dir:
81         config:
82             user.dir: ${user.dir}/.omf/config
83             app.dir: /opt/open-xchange/omf/lib/scripts
84     scheduler:
85         # Credentials for the scheduler
86         # On multi-user systems, specifying the password in a configuration file
87         # with proper file system permissions is preferred to specifying it on
88         # the command line, since the command line is visible to all local users.
89         #
90         # Example:
91         # username: admin
92         # password: secret
93
94         # Location of the scheduler. Only the protocol and host name need to be
95         # specified.
96         url: "https://localhost:8443"
97     ui:
98         color: true
99         color.theme: DARK
100        unicode: true
101        expandIds: false
102        tz: UTC
103        showTz: false
104        showAgo: false
105        prettyJson: false
106        highlightJson: false
107        shell:
108            prettyJson: true
109            highlightJson: true
110            fancyPrompt: true
111            rightHandPrompt: true
112        history.file: ${user.dir}/.omf_history
113
114     logger:
115         levels:

```

```

116 # change this to TRACE to see a detailed log of the HTTP traffic between the
117 # Orchestrator and the Scheduler
118 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.
23         # http.client.ServiceHttpClientConfiguration
24         # and the subsequent sections related to micronaut.http.services.*
25         omf-source:
26           connect-timeout: 30s
27           read-timeout: 120s
28           # Example SSL configuration in case a source uses a private certificate
29           # ssl:
30           #   trust-store:
31           #     path: file:/opt/open-xchange/omf/certs/source.p12
32           #     type: PKCS12
33           #     password: secret
34         omf-target:
35           connect-timeout: 30s
36           read-timeout: 120s
37
38     application:
39       name: omf-scheduler
40     # Configure security including basic auth: https://micronaut-projects.github.io/
41     # micronaut-security/latest/guide/#basicAuth
42     # Must be set to true or the Source Controller is not secure
43     security:
44       enabled: true
45       # Change the security of the open api views to anonymous so that they can be viewed
46       # without credentials
47       intercept-url-map:
48         - pattern: /swagger/**
49           access:
50             - isAnonymous()
51         - pattern: /swagger-ui/**
52           access:
53             - isAnonymous()
54         - pattern: /rapidoc/**
55           access:
56             - isAnonymous()
57         - pattern: /redoc/**
58           access:
59             - isAnonymous()
60
61     # https://docs.micronaut.io/latest/guide/index.html#_configuring_caches
62     #caches:
63       #example:
64         #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   batch:
116     presync:
117       # Max number of contexts in a batch
118       size: 10
119       # Strategy to use when creating batches.
120       # Current supported strategies:
121       #   - fill-first: create batches up to the batch size then create the next batch
122       #   - fill-equal: create batches of equal size
123       strategy: fill-equal
124     cutover:
125       size: 10
126       strategy: fill-equal
127     preprovisioning:
128       size: 10
129       strategy: fill-equal
130   kafka:
131     wait: false

```



```

127     queues:
128         batch: "omf-batch"
129         response: "omf-response"
130     resize:
131         batch: true
132         response: true
133     record.header.enhance: true
134     topic:
135         list:
136             timeout: -1s
137     describe:
138         timeout: -1s
139     retry:
140         attempts: 3
141         wait: 3s
142     create:
143         # whether topics should be created when a source is created or synced (true) or
144         # whether we make use of auto-creation instead (old behaviour prior to 2.1.0-6):
145         enabled: true
146         timeout: -1s
147         partitions: 2
148         replication.factor: 0
149         config:
150             retention.ms: 432000000
151     workers:
152         allow:
153             # allow Worker shutdown via the REST API
154             shutdown: false
155             # allow overriding the Sources Workers subscribe to via the REST API
156             changeSources: false
157     http:
158         users:
159             # Users defined here can have roles OMF_ADMIN or OMF_USER. Users without a role
160             # automatically have role OMF_USER
161             # assigned. Users with the role OMF_ADMIN have access to every REST API method.
162             # OMF_USER role is restricted to
163             # a subset of the REST API.
164             admin:
165                 password: secret
166                 role: OMF_ADMIN
167             omfuser:
168                 password: secret
169                 role: OMF_USER
170             admin.controller.path: /omf/scheduler/admin
171             migration.controller.path: /omf/scheduler/migration
172             user.controller.path: /omf/scheduler/user
173     metrics:
174         migration:
175             enabled: true
176         leadership:
177             enabled: true
178         batchresponse:
179             enabled: true
180             percentiles: true
181         schedulers:
182             enabled: true
183         source:
184             enabled: true
185         target:
186             enabled: true
187         monitor:
188             windows:
189                 enabled: true
190                 interval: 5m
191                 delay: 30s
192             batches:
193                 enabled: true
194                 interval: 5m
195                 delay: 30s
196             sources:
197                 enabled: true
198                 interval: 5m

```

```

197         delay: 30s
198     targets:
199         enabled: true
200         interval: 5m
201         delay: 30s
202     contextmappings:
203         enabled: true
204         interval: 5m
205         delay: 30s
206     usermappings:
207         enabled: true
208         interval: 60m
209         delay: 5m
210     migrationevents:
211         enabled: true
212         interval: 5m
213         delay: 30s
214     workers:
215         enabled: true
216         idle.since: [5m, 10m, 30m]
217     orphan-check:
218         context.batch.size: 50
219     database:
220         migration:
221             allowRead: false
222             allowWrite: false
223         scheduler:
224             allowRead: false
225             allowWrite: false
226 ---
227 jackson:
228     bean-introspection-module: true
229     serialization:
230         indent-output: true
231         writeDatesAsTimestamps: false
232 ---
233 datasources:
234     # Used to persist scheduling data
235     scheduler:
236         # url should use createDatabaseIfNotExist=true if the database will not
237         # already exist: https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-reference-
238         # configuration-properties.html
239         url: jdbc:mysql://localhost:3306/scheduler?createDatabaseIfNotExist=true
240         username: root
241         password: my-secret-pw
242         driverClassName: org.mariadb.jdbc.Driver
243         dialect: MYSQL
244         maximumPoolSize: 10
245         maxLifetime: 180000
246     # Used to create/drop databases for sources. This is not really the "default" data
247     # source
248     # but we need to use default because of bug https://github.com/micronaut-projects/
249     # micronaut-data/issues/598s
250     default:
251         url: jdbc:mysql://localhost:3306/
252         username: root
253         password: my-secret-pw
254         driverClassName: org.mariadb.jdbc.Driver
255         dialect: MYSQL
256         maximumPoolSize: 5
257         maxLifetime: 180000
258 ---
259 endpoints:
260     loggers:
261         enabled: true
262         sensitive: true
263     health:
264         discovery-client:
265             enabled: false
266         sources:
267             enabled: false
268         targets:

```

```

266         enabled: false
267 liquibase:
268     # fails with missing transition, might be fixed in later Micronaut releases
269     enabled: false
270 info:
271     enabled: true
272     sensitive: true
273     sourceCodeOrigin:
274         enabled: true
275         location: file:/opt/open-xchange/omf/scheduler/share/SourceCodeOrigin.txt
276 ---
277 zookeeper:
278     server: zookeeper:2181
279     blockUntilConnected: true
280     maxConnectedWaitTime: 30s
281     sessionTimeout: 1m
282     connectionTimeout: 15s
283     maxCloseWait: 15s
284     waitForShutdownTimeout: 15s
285     connectionRetry:
286         baseSleepTime: 5s
287         maxSleepTime: 30s
288         maxRetries: 50
289 ---
290 kafka:
291     bootstrap:
292         servers: kafka-1:9092, kafka-2:9092, kafka-3:9092
293     producers:
294         batch-producer:
295             enable.idempotence: true
296             # This enables transactions for the Batch Producer
297             # The value must be unique per application, but should
298             # not change for the same app after a crash, etc.
299             # Note that if you only want to allow a single Scheduler instance to be capable of
300             # writing Batches into the Kafka topics, then change this to be the same value
301             # across
302             # all Scheduler instances as Kafka will fence them (see PRODUCER_FENCED).
303             # But if you want all Scheduler instances to be able to write batches into Kafka
304             # topics,
305             # use a unique value for each Scheduler instance:
306             transactional.id: ${scheduler.id}
307     #consumers:
308     #response-consumer:
309 ---
310 mail:
311     # whether to send emails
312     enabled: false
313     window:
314         # whether to send emails when a Window succeeds:
315         success: false
316         # whether to send emails when a Window fails:
317         failure: true
318     # mandatory, must be set to be able to send emails and it
319     # must be a valid email address in the form localpart@domain, or sending will fail:
320     from: ${scheduler.id}@example.com
321     # whom to send those mails to (can be a comma separated list):
322     to:
323     cc:
324     bcc:
325     # text to include in the subject line, wrapped in []:
326     subject.id:
327     smtp:
328         host: localhost
329         port: 25
330         # leave empty for no authentication:
331         username:
332         password:
333         # SMTP, SMTPS or SMTP_TLS (SMTP with mandatory StartTLS):
334         transport: SMTP
335         # whether to allow SMTP without StartTLS:
336         smtp.plain: true
337         tls:

```

```
336      # whether to trust all SMTP server keys
337      trustall: false
338      # whether to verify SMTP server keys
339      verify: true
340  ---
341  liquibase:
342    datasources:
343      scheduler:
344        change-log: 'classpath:liquibase/scheduler/liquibase-changelog.xml'
345  ---
346  logger:
347    levels:
348      ROOT: INFO
349      com.openxchange: INFO
350      omf: INFO
351      omf.scheduler.admin.AuthenticationProviderUserPassword: WARN
352      org.apache.kafka.clients.consumer.ConsumerConfig: WARN
353  ---
```