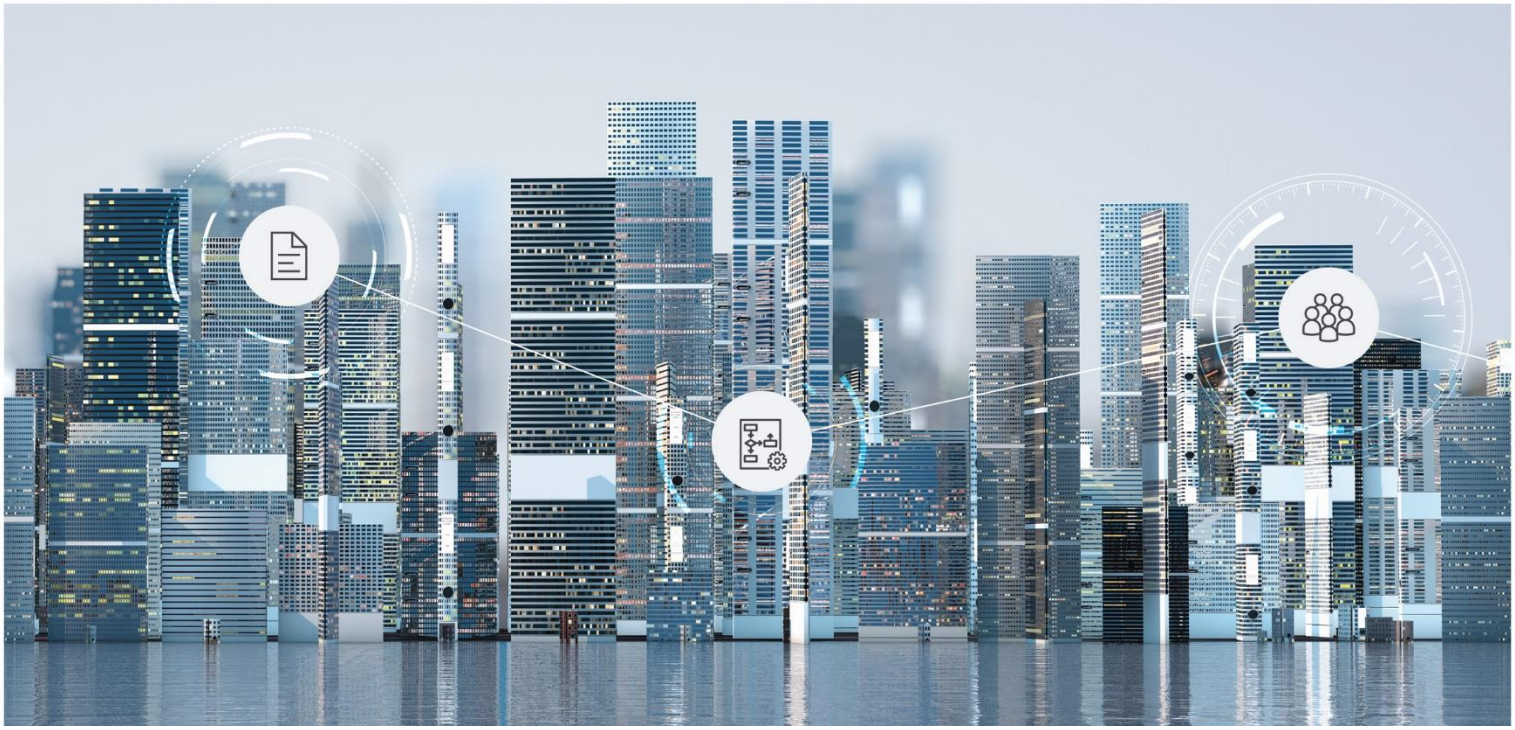
Fabasoft



White Paper

Fabasoft Folio Cache Notification

Fabasoft Folio 2024 Update Rollup 1

Contents

1 Introduction 4

2 Software Requirements 4

3 Functionality in Detail 4

3.1 General 4

3.2 Heartbeat and Sequence Numbers 4

3.3 Event Queue 5

4 Configuration 5

4.1 UDP Multicast 5

4.2 Event Queue 6

4.3 Deployment 7

5 Monitoring and Troubleshooting 8

5.1 Event Log 8

5.2 Network Load Monitoring 9

6 Further Information 9

1 Introduction

Cache notifications improve the Fabasoft Folio Kernel cache behavior. The aim of the protocol is to reduce the communication between server and client, if the object is already placed in the client cache. To achieve this, each Fabasoft Folio COO Service sends event notifications to inform the client about potential object modifications.

The cache notifications can be enabled or disabled in the *Current Domain* (default: enabled).

2 Software Requirements

System environment: All information contained in this document implicitly assumes a Microsoft Windows environment or Linux environment.

Supported platforms: For detailed information on supported operating systems and software see the software product information on the Fabasoft distribution media.

3 Functionality in Detail

3.1 General

If an object gets locked (e.g. because of editing) it is refreshed by the `FORCE` option. By means of this option in each case an RPC (remote procedure call) to the Fabasoft Folio COO Service is performed to ensure that the object is up to date. If necessary, the update is ensured by another RPC.

If an object that is already placed in the Fabasoft Folio Kernel cache is accessed without the `FORCE` option (e.g. reading access), no RPC to the Fabasoft Folio COO Services is necessary in case that no modification notice is received from the Fabasoft Folio COO Service. This RPC reduction improves the performance.

If an object is modified by means of a transaction, the appropriate Fabasoft Folio COO Service sends a modification notice. Each Fabasoft Folio Kernel receives this modification notice and marks the object in its Fabasoft Folio Kernel cache as modified. As a consequence, the object needs to be updated by Fabasoft Folio COO Service at the next access.

The modification notice only indicates that the object has been modified but the update of the object is not performed until the next access.

3.2 Heartbeat and Sequence Numbers

Because there is no guarantee that notifications reach their recipient, it may happen that the Fabasoft Folio Kernel spuriously treats an object to be up to date.

To avoid this each Fabasoft Folio COO Service sends a so-called heartbeat periodically (every 5 seconds) where each message contains a sequence number.

An object in Fabasoft Folio Kernel cache is only up to date if the following conditions are fulfilled:

- Since the last update all notices have been received in the cache (ensured by sequence numbers),
- no received notice contains a modification of this object and

- the last received notice is not older than the specified heartbeat plus a tolerance value (about 7 seconds) for the duration of reception.

3.3 Event Queue

The event queue is an async queue used for sending notifications from COO services to kernels as well as from kernels to kernels.

It is using RabbitMQ as scalable async queue without a single point of failure.

4 Configuration

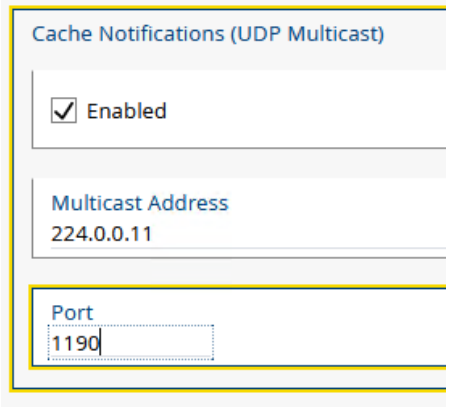
For the notifications there are two transport layers available that can be configured independently. These settings can be made in the current domain object.

If the settings are modified, it takes a short moment to ensure that the modifications are distributed to other Fabasoft Folio COO Services (synchronization threads run in periodical intervals).

After a restart of Fabasoft Folio COO Services the new settings are active. Starting the Fabasoft Folio COO Services the used settings are written to the event log (see chapter 5.1 “Event Log”).

4.1 UDP Multicast

For UDP multicast you need to configure the multicast address and the port number.



Performing a Fabasoft Folio update or a reinstallation, the “UDP multicast transport” is activated by default and the multicast address and the appropriate port are initialized with default values. These default values are dependent on the installed Fabasoft Folio Domain. The assigned multicast address stays between 224.0.0.10 and 224.0.0.250.

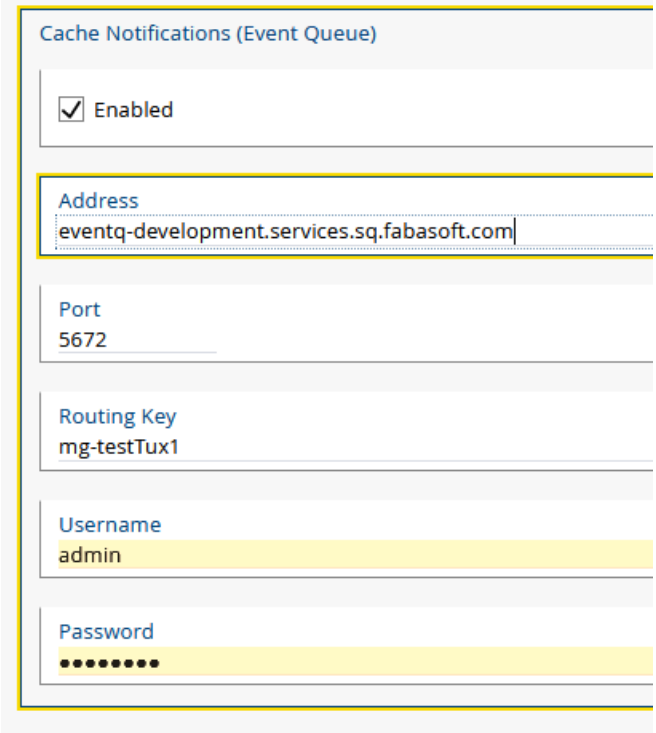
Note:

- The address range between 224.0.0.0 and 224.0.0.255 is assigned to low level protocols. Datagrams that are sent to addresses within this range are not routed by a multicast capable router.
- The transport works only if the UDP packets sent by the server can be received by the client. Ensure an appropriate network configuration. On Linux systems, the following command can be executed to add the appropriate route:
  `route add -net 224.0.0.0 netmask 240.0.0.0 dev <device>`

- If the “UDP multicast transport” does not work (error message in the event log), it is possible that another Fabasoft Folio Domain with the same license (same domain ID) exists in the network. In this case change the automatically calculated default values to prevent that several domains use the same multicast address and the same port.
- If you are using Linux and no default gateway is set you have to set a route entry for the Multicast address-range.

4.2 Event Queue

The event queue transport requires a RabbitMQ broker be installed within the network. In addition, the broker address or host name, port number, routing key, username, and password can be configured.



The event queue transport is disabled by default. For the use of containers, the use of RabbitMQ is recommended as UDP might not be available.

Event Queue Technical Settings

The event queue has some technical settings like timeouts set.

The values are:

- Message lifetime: 10 minutes
  This applies if a message is not consumed by the customer which is the kernel.
- Queue lifetime: 1 hour without consumers
  This applies if a queue has no consumers for the given time. That can happen when OpenShift terminates connections because of certain reasons like the licenses has to be renewed or a node failure occurs.

4.3 Deployment

It is possible to have a configuration where both the UDP multicast and the event queue transport are enabled. In this case the Fabasoft Folio COO Service generates both types of events.

In this case an override should be used in the process hosting the Fabasoft Folio Kernel to enable just one of the transports. To configure the event listener set the environment variables `EVENTQUEUELISTENERENABLED` to 0 or 1 and `MULTICASTLISTENERENABLED` to 0 or 1 to override the "Enabled" setting in the corresponding configuration.

Note: If both protocols are enabled, the Fabasoft Folio Kernel will receive notifications twice that may slow down the system.

5 Monitoring and Troubleshooting

5.1 Event Log

At the startup of the Fabasoft Folio COO Service an event log entry is generated, which shows the status of the multicast server.

In the following example the "UDP multicast transport" is enabled, the used multicast address is 224.0.0.11 and the multicast port is 1190.

Example

```
ReportEvent: FSCRMP-00006: RMP::MulticastServer: startup
  fscrmp.channel: RMP::MulticastServer
  fscrmp.address: 224.0.0.11
  fscrmp.port: 1190
  fscrmp.service: COO Service 1 (COOSVC1)
  fscrmp.sequence: 7053609
  fscrmp.domain: 1.506 (mg-testtux1)
  fscrmp.serviceaccount: fscsrv
```

In the following example the "event queue transport" is enabled, the used RabbitMQ broker hostname is `eventq-development.services.sq.fabasoft.com` and the port is 5672.

Example

```
ReportEvent: FSCRMP-00006: RMP::EventQueueServer: startup
  fscrmp.channel: RMP::EventQueueServer
  fscrmp.address: eventq-development.services.sq.fabasoft.com
  fscrmp.port: 5672
  fscrmp.service: COO Service 1 (COOSVC1)
  fscrmp.sequence: 7053609
  fscrmp.domain: 1.506 (mg-testtux1)
  fscrmp.serviceaccount: fscsrv
```

If the cache notification is disabled, an entry in the event log is created when starting the Fabasoft Folio COO Service.

Example

```
ReportEvent: COOSTD-00803: Multicast and event queue based cache invalidation is
disabled
  coostd.category: RMP
  coostd.domain: 1.506 (mg-testtux1)
  coostd.service: COO Service 1 (COOSVC1)
  coostd.serviceaccount: fscsrv
  source: Fabasoft Folio Server
  processid: 52390
  processname: COOService_1_506_1
  threadid: 140434003068672
  version: 21.11.0.7
```

The client (Fabasoft Folio Kernel) writes an entry in the event log after a successful start.

Example

```
ReportEvent: FSCRMP-00006: RMP::MulticastClient: startup
  fscrmp.channel: RMP::MulticastClient
  fscrmp.address: 224.0.0.11
  fscrmp.port: 1190
  fscrmp.interface: 10.10.67.215
```

UDP package losses (modification notices and heartbeats) are written as warning in the event log.

Note: A package loss does not lead to a data loss or inconsistency. It is simply impossible to perform optimized cache behavior.

```
FSCRMP: Server lost: unexpected message sequence number 757177
Server: 286586004570114
Sequence: 757176
```

Or

Example

```
ReportEvent: FSCRMP-00108: RMP::MulticastClient: server lost: heartbeat missing
  fscrmp.channel: RMP::MulticastClient
  fscrmp.address: 224.0.0.11
  fscrmp.port: 1190
  fscrmp.interface: 10.10.67.215
```

5.2 Network Load Monitoring

For analyzing the network traffic, a network sniffing tool like "Wireshark" (http://www.wireshark.org) may be used.

With an appropriate filter all UDP packages of a certain address (e.g. IP address of the server) can be filtered in a certain area.

Example: `ip.addr == 192.168.100.83 and udp.port == 1190`

This filter restricts to UDP packages with the sender IP address 192.168.100.83 and the destination port 1190.

Filter: ip.addr == 192.168.100.83 and udp.port == 1190 | Expression... | Clear | Apply

No.	Time	Source	Destination	Protocol	Info
322	14.712154	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
440	16.773242	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190
590	19.390818	192.168.100.83	224.0.0.11	UDP	Source port: 3050 Destination port: 1190
675	19.712896	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
707	21.773607	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190
848	24.392957	192.168.100.83	224.0.0.11	UDP	Source port: 3050 Destination port: 1190
853	24.713534	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
963	26.774018	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190
1103	29.393390	192.168.100.83	224.0.0.11	UDP	Source port: 3050 Destination port: 1190
1108	29.713688	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
1141	31.774407	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190
1207	34.393760	192.168.100.83	224.0.0.11	UDP	Source port: 3050 Destination port: 1190
1211	34.714093	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
1272	36.774812	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190
1342	39.394154	192.168.100.83	224.0.0.11	UDP	Source port: 3050 Destination port: 1190
1348	39.714497	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
1411	41.775205	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190
1531	43.964443	192.168.100.83	224.0.0.11	UDP	Source port: 3032 Destination port: 1190
1556	44.394573	192.168.100.83	224.0.0.11	UDP	Source port: 3050 Destination port: 1190
1599	46.775658	192.168.100.83	224.0.0.11	UDP	Source port: 3041 Destination port: 1190

6 Further Information

Topic	Link
UDP (RFC 768)	https://tools.ietf.org/html/rfc768
Host Extensions for IP Multicasting (RFC 1112)	https://tools.ietf.org/html/rfc1112

Transmission of IPv6 Packets over Ethernet Networks (RFC 2464)	https://tools.ietf.org/html/rfc2464
Internet Protocol Version 6 (IPv6) Addressing Architecture (RFC 3513)	https://tools.ietf.org/html/rfc3513
Wireshark	https://www.wireshark.org/
RabbitMQ	https://www.rabbitmq.com