LB Cluster Solution Configuration Tool User Manual

Overview

This program is command-line user interface to FIXEdge Configuration Service (see LB-Cluster Components). The program is invoked via command `fectl`.

Pre-conditions

1. LB-Cluster solution is deployed
2. OpenJDK 8 installed

Important configuration notes

Configuration tool needs to know about Configuration Service location. There are several ways to do this

1. To use discovery service such as Consul agent. User may install consul agent on the same workstation as for `feclt` tool or by specifying directly address and port in command options. If no agent options are specified configuration tool tries to use local agent.
2. To specify directly address and port of Configuration Service

Common Options

Running fectl without parameters prints help (same as using --help or -h parameters):

Command options

```
~]$ fectl

FixEdge Configuration Tool
Usage: fectl [-dhV] [COMMAND]
  -d, --debug     Enable printing of stacktrace on errors.
  -h, --help      Show this help message and exit.
  -V, --version   Print version information and exit.
Commands:
  add-session                  Adds session to FE cluster
  remove-session               Removes session from FE cluster
  edit-session                 Edit session from FE cluster
  get-all-sessions             Returns all sessions for cluster from
                                 Configuration Storage
  get-session                  Returns session for cluster from Configuration
                                 Storage
  add-schedule                 Adds schedule to FE cluster
  remove-schedule              Removes schedule from FE cluster
  edit-schedule                Edit schedule from FE cluster
  get-all-schedules            Return all schedules from FE cluster
  get-schedule                 Returns schedule for cluster from Configuration
                                 Storage
  get-all-established-sessions  Returns all established session from each
                                 FIXEdge nodes
  get-all-clusters             Returns all clusters from Configuration Storage
  add-cluster                  Creates new cluster instance
  remove-cluster               Removes cluster instance from the storage
```

command option	required	description
--help -h	No	print help message and exit. This option can be passed to `fectl` or its subcommand. In the latter case the help message will describe the selected subcommand.
--cluster=<cluster_name> -c=<cluster_name>	Yes for all subcommands (except get-all-clusters)	specifies cluster name to use.
--discovery=<value>	No	specifies service discovery method to locate the Configuration Service. Valid values: consul (default), nop
--consul-agent=<host>:<port>	No	specifies host and port of the Consul agent to use when service discovery method is set to `consul` or left unspecified. By default local Consul agent is used. default port is 8500
--service-host=<host>	No	specifies host of the Consul agent to use when service discovery method is set to `consul` or left unspecified. By default local Consul agent is used.
`--service-port=<port>`	No	specifies port of the Consul agent to use when service discovery method is set to `consul` or left unspecified. By default local Consul agent is used. If user does not use automatic discovery then --service-port=8787
--verbose	No	write Configuration Service response to standard output.
--debug -d	No	Enable printing of stacktrace on errors.
--version -V	No	Print version information and exit.

Cluster management commands

Get all clusters from Configuration Storage

`Subcommand: get-all-clusters`

Description: command is used to get list of clusters

Example1: to get all clusters run command:

```
fectl get-all-clusters
```

Command output example:

command output

`ClusterName  | ClusterType  |`
`FIXEdge-Java-1  | FIX_EDGE_JAVA |`
`FixEdge1  | FIX_EDGE  |`

Adding new cluster

`Subcommand: add-cluster`

Description: command is used to add new cluster.

command option	required	description
--clusterName=<cluster_name> -c=<cluster_name>a	Yes	Specifies cluster name to use.
--clusterType=<cluster_type> -t=<cluster_type>	Yes	Specifies cluster type to use. Available types: • FIX_EDGE • FIX_EDGE_JAVA

Example1: to add cluster run command:

- For FIX_EDGE type

  ```
  fectl add-cluster --clusterName=FIXEdge2 --clusterType=FIX_EDGE
  ```

- For FIX_EDGE_JAVA type

  ```
  fectl add-cluster --clusterName=FIXEdge-Java-2 --clusterType=FIX_EDGE_JAVA
  ```

Removing existing cluster

`Subcommand: remove-cluster`

Description: command is used to remove cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>a	Yes	specifies cluster name to use.

Example1: to remove cluster run command:

```
fectl remove-cluster --cluster=Test-Cluster-Name
```

Session Management Commands

ConfiguredName (for FixEdge C++) / *sessionID* (for FixEdge Java) is used as a unique session identity for all session management commands.

Adding New Session

`Subcommand: add-session`

Description: command is used to add new session to Cluster. User have to prepare file in JSON format with all required session parameters.

The minimal set of parameters for acceptor sessions is:

- for FixEdge C++ - SenderCompID, TargetCompID, Role, ConfiguredName, Version, StorageType
- for FixEdge Java - sessionID, senderCompID, targetCompID, fixVersion

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
-f=<file_path> --file=<file_path>	Yes	Session definition file in JSON format, e.g.: session.json

Session definition file template for FixEdge C++ looks as follows:

Session parameters

```
{
  "ActiveConnection": "PRIMARY",
  "AppProtocolBaseVersion": "FIX44",
  "ConfiguredName": "TestFIXAcceptor",
  "CustomVersion": "",
  "ExtraParams": {
    "AggressiveReceiveDelay": 500,
    "AllowMessageWithoutPossDupFlag": false,
    "ConnectAddress": "",
    "ConnectPort": 0,
    "CpuAffinity": 0,
    "CustomSessionType": "GENERIC",
    "CyclicSwitchBackupConnection": false,
    "DeliverAppMessagesOutOfOrder": false,
    "DisableTCPBuffer": false,
    "EnableAutoSwitchToBackupConnection": false,
    "EnableMessageRejecting": false,
    "EncryptMethod": "NONE",
    "FixKey": "",
    "ForceSeqNumReset": "OFF",
    "ForcedReconnect": false,
    "GenerateCheckSum": true,
    "HandleSeqNumAtLogon": false,
    "HiddenLogonCredentials": false,
    "IgnoreSeqNumTooLowAtLogon": false,
    "IntradayLogoutToleranceMode": true,
    "KeepConnectionState": true,
    "LogIncomingMessages": true,
    "LogonMessageSessionQualifierTag": 0,
    "MaskedTags": [
      "554",
      "925"
    ],
    "MaxMessagesAmountInBunch": 0,
    "MessagesStorageSize": 20000,
    "Password": "AcceptorPassword",
```

```
    "PasswordTag": 554,
    "ReconnectInterval": 5000,
    "ReconnectMaxTries": 5,
    "RecvCpuAffinity": 0,
    "ResendRequestBlockSize": -1,
    "SendCpuAffinity": 0,
    "SendLastMsgSeqNumProcessed": false,
    "SenderLocationID": "SLocationId",
    "SenderSubID": "",
    "SocketBusyPollTime": -1,
    "SocketOpPriority": "EVEN",
    "SocketRecvBufSize": -1,
    "SocketSendBufSize": -1,
    "SourceIPaddress": [],
    "StorageRecoveryStrategy": "NONE",
    "StorageType": "Oracle",
    "SuppressDoubleResendRequest": false,
    "TargetLocationID": "TLocationId",
    "TargetSubID": "",
    "Transport": "SOCKETS",
    "UseAsyncConnect": false,
    "UseBlockingSockets": false,
    "Username": "FIXAcceptorUser",
    "UsernameTag": 553,
    "ValidateCheckSum": true,
    "Validation": {
      "AllowZeroNumInGroup": false,
      "CheckRequiredGroupFields": true,
      "IgnoreUnknownFields": false,
      "IsEnabled": false,
      "ProhibitDuplicatedTags": true,
      "ProhibitTagsWithoutValue": true,
      "ProhibitUnknownTags": true,
      "VerifyReperatingGroupBounds": true,
      "VerifyTagsValues": true
    }
  },
  "InSeqNum": 1,
  "IncomingMessagesLimit": 0,
  "IncomingThroughputLimit": 0,
  "OutSeqNum": 1,
  "OutgoingQueueSizeLimit": 0,
  "Protocol": "FIX_TCP",
  "Qualifier": "",
  "RecreateOnLogout": true,
  "Role": "ACCEPTOR",
  "SSLCertificate": "",
  "SSLCheckPrivateKey": false,
  "SSLPrivateKey": "",
  "SSLProtocols": "SSLv2,SSLv3,TLSv1,TLSv1_1,TLSv1_2",
  "SenderCompID": "FIXEDGE",
  "State": "WAIT_FOR_FIRST_LOGON",
  "StorageType": "Oracle",
  "TargetCompID": "FIXCLIENT",
  "TerminateOnLogout": false,
  "Version": "FIX44"
}
```


Session definition file template for FixEdge Java looks as follows:

Session parameters

```
{
  "sessionID": "SessionJava1",
  "senderCompID": "sender1",
  "targetCompID": "target1",
  "fixVersion": "FIX.4.4",
  "inSeqNumsForNextConnect": 0,
  "outSeqNumsForNextConnect": 0,
  "username": "user",
  "password": "pass",
  "lastSeqNumResetTimestamp": 0,
  "heartbeatInterval": 30,
  "sessionType": "acceptor",
  "groups": "A, B, C"
}
```

The JSON object contains parameter-value pairs in format `"parameter name" : "parameter value"`. Parameter-value pairs are separated by comma.

The fields of the JSON object reflect the properties of the session definition.

The `null` value of a field or missing field means use of the default value for the field.

Example1: to add a new session to Cluster run command:

- For FIX_EDGE type

  `fectl add-session --cluster=FIXEdge1 --file=session3.json`

  session name is contained in session.json file ("ConfiguredName" field)

  Successful adding of session:

Command and output

```
user@lbc:~/work/test$ fectl add-session -c=FixEdge1 -f=session3.json
Response: OK 200
[["SessionParams are saved to the storage"],[{"Content":[{"Status":200,"Response":"null","Action":"create
session","Node":"10.3.0.2:8903"},{"Status":200,"Response":"null",
"Action":"create session","Node":"10.3.0.3:8903"}],"Service":"FixEdge"},{"Content":[{"Status":200,"Action":"
createSession","Node":"10.3.0.12:8686"},
{"Status":200,"Action":"createSession","Node":"10.3.0.16:8686"}],"Service":"SchedulerService"}]]
```

- For FIX_EDGE_JAVA type

  `fectl add-session --cluster=FIXEdge-Java-1 --file=session3.json`

  session name is contained in session.json file ("sessionID" field)

  Successful adding of session:

Command and output

```
user@lbc:~/work/test$ fectl add-session -c=FIXEdge-Java-1 -f=session3.json
Response: OK 200
[["FejSessionParams are saved to the storage"],[{"Content":[{"Status":201,"Response":"{\"password\":\"pass\",
\"storagefactory\":\"com.epam.fixengine.fo.fixaj.cluster.storage.ClusterStorageFactory\",\"
serveracceptorstrategy\":\"com.epam.fej.server.fix.FejSessionAcceptorStrategyHandler\",
\"username\":\"user\",\"state\":\"WAITING_FOR_CONNECTION\",\"autostart.acceptor.admin.storagefactory\":\"com.
epam.fixengine.storage.FilesystemStorageFactory\",
\"autostart.acceptor.admin.login\":\"admin\",\"inmemoryqueue\":\"true\",\"port\":\"0\",\"queuethresholdsize\":\"
0\",\"senderCompID\":\"sender1\",\"validation\":\"false\",
\"autostart.acceptor.admin.password\":\"admin\",\"sessionID\":\"SessionJava1\",\"lastSeqNumResetTimestamp\":\"
0\",\"forceseqnumreset\":\"Never\",
\"fo.multicast.address\":\"224.0.3.1\",\"sessionType\":\"acceptor\",\"autostart.acceptor.targetids\":\"admin\",
\"autostart.acceptor.admin.ip\":\"*\",\"autostart.acceptor.admin.fixserverlistener\":\"com.epam.admintool.
AdminTool\",
\"targetCompID\":\"target1\",\"groups\":\"A,B,C\",\"fo.multicast.port\":\"5001\",\"outSeqNumsForNextConnect\":\"
0\",\"fixVersion\":\"FIX.4.4\",\"inSeqNumsForNextConnect\":\"0\"}",
"Action":"create session","Node":"10.3.0.13:9010"},{"Status":201,"Response":"{\"password\":\"pass\",\"
storagefactory\":\"com.epam.fixengine.fo.fixaj.cluster.storage.ClusterStorageFactory\",
\"serveracceptorstrategy\":\"com.epam.fej.server.fix.FejSessionAcceptorStrategyHandler\",\"username\":\"user\",
\"state\":\"WAITING_FOR_CONNECTION\",
\"autostart.acceptor.admin.storagefactory\":\"com.epam.fixengine.storage.FilesystemStorageFactory\",\"autostart.
acceptor.admin.login\":\"admin\",\"inmemoryqueue\":\"true\",\"port\":\"0\",
\"queuethresholdsize\":\"0\",\"senderCompID\":\"sender1\",\"validation\":\"false\",\"autostart.acceptor.admin.
password\":\"admin\",\"sessionID\":\"SessionJava1\",
\"lastSeqNumResetTimestamp\":\"0\",\"forceseqnumreset\":\"Never\",\"fo.multicast.address\":\"224.0.3.1\",\"
sessionType\":\"acceptor\",\"autostart.acceptor.targetids\":\"admin\",
\"autostart.acceptor.admin.ip\":\"*\",\"autostart.acceptor.admin.fixserverlistener\":\"com.epam.admintool.
AdminTool\",\"targetCompID\":\"target1\",\"groups\":\"A,B,C\",
\"fo.multicast.port\":\"5001\",\"outSeqNumsForNextConnect\":\"0\",\"fixVersion\":\"FIX.4.4\",\"
inSeqNumsForNextConnect\":\"0\"}","Action":"create session","Node":"10.3.0.14:9010"}],
"Service":"FixEdge"},{"Content":[],"Service":"SchedulerService"}]]
```

Example2: to add a new session to FIXEdge1 Cluster with Configuration Service specified directly (running on host 1.2.3.4 at port 8787):

```
fectl add-session --service-host=1.2.3.4 --service-port=8787 --cluster=FIXEdge1 --file=session.json
```

Example3: to add a new session to FIXEdge1 Cluster using connection via remote Consul agent. Consul agent on host 5.6.7.8 (and default port 8500) is used to locate the Configuration Service.:

```
fectl add-session --consul-agent=5.6.7.8 --cluster=FIXEdge1 --file=session.json
```

Removing Existing Session

Subcommand: remove-session

Description: command is used to remove a session that already exists in Cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
-s=<session_name> --session=<session_name>	Yes	Session name for session to be removed ("ConfiguredName" (or "sesionID" for FixEdge Java) field in JSON session configuration file)

Example1: to remove the specified session from Cluster run command:

- For FIX_EDGE type

```
fectl remove-session --cluster=FIXEdge1 --session=TestFIXAcceptor3
```

Successful removing of session:

Command and output

```
user@lbc:~/work/test$ fectl remove-session -c=FixEdge1 -s=TestFIXAcceptor3
Response: OK 200
[{"Status":200,"Response":"null","Action":"remove session","Node":"10.3.0.2:8903","Parameter":"
TestFIXAcceptor3"},{"Status":200,"Response":"null","Action":"remove session","Node":"10.3.0.3:8903","
Parameter":"TestFIXAcceptor3"}]
```

- For FIX_EDGE_JAVA type

```
fectl remove-session --cluster=FIXEdge-Java-1 --session=SessionJava1
```

Successful removing of session:

Command and output

```
user@lbc:~/work/test$ fectl remove-session -c=FixEdge1 -s=SessionJava1
Response: OK 200
[["FejSessionParamswith id = SessionJava1 are deleted from the storage"],[{"Content":[{"Status":200,"Action":"
remove session","Node":"10.3.0.13:9010","Parameter":"SessionJava1"},
{"Status":200,"Action":"remove session","Node":"10.3.0.14:9010","Parameter":"SessionJava1"}],"Service":"
FixEdge"},{"Content":[],"Service":"SchedulerService"}]]
```

All common options to locate the Configuration Service can be used as shown in examples in section AddingNewSession.

Modifying Existing Session

Subcommand: edit-session

Description: command is used to modify parameters of a session that already exists in Cluster. To modify session user has to prepare a file in JSON format containing session parameters to be changed. Command apply these parameters to specified session.

command option	required	description
--cluster<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
-s=<session_name> --session=<session_name>	Yes	Session name for session to be removed ("ConfiguredName" field in JSON session configuration file)
-f=<file_path> --file=<file_path>	Yes	File in JSON format, containing only parameters to be changed

Example1: to modify existing session run command:

```
fectl edit-session --cluster=Test-Cluster-Name --session=FIXAcceptor5 --file=edited-session.json
```

Parameters defining session ID (*SenderCompID, TargetCompID, Qualifier* for FixEdge C++ and *senderCompID, targetCompID, qualifier* for FixEdge Java) cannot be changed by `fectl edit-session` command. `fectl remove-session` and `fectl add-session` must be used in this case.

New parameters are read from the file `edited-session.json` and applied to the selected session configuration.

The content of the `edited-session.json` are similar to the `session.json` from section AddingNewSession.

Get Session Parameters

`Subcommand: get-session`

Description: command is used to get list of parameters/values for session defined in Cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
-s=<session_name> --session=<session_name>	No	Session name ("ConfiguredName" (or "sessionId" for FixEdge Java) field in JSON session configuration file)

Example1: to get session parameters run command:

The output will be in JSON format.

- For FIX_EDGE type

```
fectl get-session --cluster=FIXEdge1 --session=FIXAcceptor5
```

```
user@lbc:~/work/test$ fectl get-session --cluster=FIXEdge1 --session=FIXAcceptor5
{
    "Role":                         "ACCEPTOR",
    "ConfiguredName":               "FIXAcceptor5",
    "SenderCompID":                 "FIXEDGE",
    "TargetCompID":                 "FIXCLIENT1",
    "Qualifier":                        "",
    "DefaultApplicationProtocol":   "FIX44",
    "Version":                      "FIX44",
    "StorageType":                  "oracle",
    "InSeqNum":                     1,
    "OutSeqNum":                    1,
    ...
}
```

- For FIX_EDGE_JAVA type

```
fectl get-session --cluster=FIXEdge-Java-1 --session=SessionJava1
```

```
user@lbc:~/work/test$ fectl get-session --cluster=FIXEdge-Java-1 --session=SessionJava1
Response: OK 200
{
    "heartbeatInterval": 30,
    "password": "pass",
    "outSeqNumsForNextConnect": 0,
    "groups": "D, E, F",
    "sessionType": "acceptor",
    "targetCompID": "target1",
    "sessionID": "SessionJava1",
    "fixVersion": "FIX.4.4",
    "senderCompID": "sender1",
    "inSeqNumsForNextConnect": 0,
    "lastSeqNumResetTimestamp": 0,
```

Get Session Parameters for all sessions

`Subcommand: get-all-sessions`

Description: command is used to get list of parameters/values for all sessions defined in Cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.

Example1: to get session parameters of all sessions run command:

```
fectl get-all-sessions --cluster=Test-Cluster-Name
```

Get list of all established sessions on FIXEdge nodes

`Subcommand: get-all-established-sessions`

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.

Description: command is used to get list of all established sessions on FIXEdge nodes in Cluster.

Example1: to get all established sessions run command:

- For FIX_EDGE type

  ```
  fectl get-all-established-sessions --cluster=FIXEdge1
  ```

  Example of command output:

command output

```
FIXEdgeNode           | ConfiguredName   | SenderCompID | TargetCompID | Qualifier
https://10.3.0.2:8903 | TestFIXAcceptor  | FIXEDGE      | FIXCLIENT    |
https://10.3.0.2:8903 | TestFIXAcceptor3 | FIXEDGE3     | FIXCLIENT3   |
```

*ConfiguredName is unique session name in Configuration Service. SenderCompID, TargetCompID and Qualifier parameters combination must be unique for all established sessions in Cluster.

- For FIX_EDGE_JAVA type

  ```
  fectl get-all-established-sessions --cluster=FIXEdge-Java-1
  ```

  Example of command output:

command output

```
FIXEdgeNode            | sessionID         | senderCompID | targetCompID        | sessionQualifier
https://10.3.0.13:9010 | SessionJava1    | sender1        | target1        | null
https://10.3.0.14:9010 | SessionJava2    | sender2        | target2        | null
```

*sessionID is unique session name in Configuration Service. senderCompID , targetCompID and qualifier parameters combination must be unique for all established sessions in Cluster.

FIXEdgeNode is a FIXEdge node in format <node ip address>:<port>.

Schedule management commands

`ScheduleName` is used as a unique schedule identity for all schedule management commands.

Add new schedule

Subcommand: add-schedule

Description: command is used to add new schedule to Cluster. User have to prepare file in JSON format with all required schedule parameters.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
-f=<file_path> --file=<file_path>	Yes	File in JSON format, containing schedule parameters

Schedule definition file must contain a JSON object, e.g.:

Schedule parameters

```
{
  "ScheduleName": "ScheduleName1",
  "ConnectTime": "",
  "DisconnectTime": "",
  "StartTime": "12:00:00",
  "TerminateTime": "",
  "TimeZone": "UTC",
  "DaysOff": ""
}
```

Example1: to add new schedule to FIXEdge1 Cluster run command:

```
fectl add-schedule --cluster=FIXEdge1 --file=schedule.json
```

All common options to locate the Configuration Service can be used as shown in examples in section AddingNewSession.

Remove existing schedule

Subcommand: remove-schedule

Description: command is used to remove existing schedule from Cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
--schedule=<schedule_name> -s=<schedule_name>	Yes	schedule name ("ScheduleName" field in JSON schedule parameters file)

Example1: to remove schedule ScheduleName5 from FIXEdge1 Cluster run command:

```
fectl remove-schedule --cluster=FIXEdge1 --schedule=ScheduleName5
```

All common options to locate the Configuration Service can be used as shown in examples in section AddingNewSession.

Modify Existing Schedule

Subcommand: edit-schedule

Description: command is used to modify parameters of a schedule that already exists in Cluster. To modify schedule user have to prepare a file in JSON format containing schedule parameters to be changed. Command apply these parameters to specified schedule.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.
--schedule=<schedule_name>	Yes	Schedule name for schedule to be removed ("ScheduleName" field in JSON schedule configuration file)
-f=<file_path> --file=<file_path>	Yes	File in JSON format, containing only parameters to be changed

Example1: to modify schedule ScheduleName5 in FIXEdge1 cluster run command:

```
fectl edit-schedule --cluster=FIXEdge1 --schedule=ScheduleName5 --file=edited-schedule.json
```

New parameters are read from the file `edited-schedule.json` and applied to the schedule ScheduleName5 configuration.

The contents of the `edited-schedule.json` are similar to the `schedule.json` from section Addnewschedule.

Parameter "ScheduleName" cannot be changed by `fectl edit-schedule` command. `fectl remove-schedule` and `fectl add-schedule` must be used in this case.

Get Schedule Parameters

`Subcommand: get-schedule`

Description: command is used to get list of parameters/values for schedule defined in Cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.

--schedule=<schedule_name> -s=<schedule_name>	Yes	Schedule name for schedule to be removed ("ScheduleName" field in JSON schedule configuration file)

Example1: to get parameters of the schedule ScheduleName5 defined in the Cluster `FIXEdge1` run command:

```
fectl get-schedules --cluster=FIXEdge1 --schedule=ScheduleName5
```

Result is schedule parameter/value list in JSON format:

```
{
  "ScheduleName": "ScheduleName5",
  "ConnectTime": "",
  "DisconnectTime": "",
  "StartTime": "12:00:00",
  "TerminateTime": "",
  "TimeZone": "UTC",
  "DaysOff": ""
}
```

Get Schedule Parameters for all schedules

`Subcommand: get-all-schedules`

Description: command is used to get list of parameters/values for all schedules defined in Cluster.

command option	required	description
--cluster=<cluster_name> -c=<cluster_name>	Yes	specifies cluster name to use.

Example1: to get parameters of all schedules defined in the Cluster `FIXEdge1` run command:

```
fectl get-all-schedules --cluster=FIXEdge1
```

Troubleshooting

	Problem/Symptom	Description	Solution

1	user@lbc:~/work/test$ fectl add-session -c=FixEdge1 Missing required option '--file=<filePath>' Usage: fectl add-session [-hV] [--consul-agent=<hostAndPortSpec>] [--discovery=<discoveryKind>] [--service-host=<hostname>] [--service-port=<port>] -c=<clusterName> -f=<filePath> Adds session to FE cluster --consul-agent=<hostAndPortSpec> Hostname and port of the Consul Agent (local agent by default). This option only makes sense if --discovery=consul or missing otherwise it is ignored. --discovery=<discoveryKind> Service discovery kind to use (consul, nop) --service-host=<hostname> Hostname of the configuration service. This option only makes sense if --discovery=nop otherwise it is ignored --service-port=<port> Port of the configuration service. This option only makes sense if --discovery=nop otherwise it is ignored -c, --cluster=<clusterName> Name of the cluster to work with. This shall be the same as specified in FIXEdge configuration -f, --file=<filePath> The file path that contains session properties. -h, --help Show this help message and exit. -V, --version Print version information and exit.	missing required option (in example '--file=<filePath>' in fectl add-session command)	1. Read help for command (e.g. `fectl add-session -h`) 2. Add missing parameters
2	user@lbc:~/work/test$ fectl add-session -c=FixEdge1 -f=session10.json Error while running command (com.epam.fixedge.config.tool.commands.session.AddSession@64cee07): com.epam.fixedge.config.tool.commands.CommandException: Problem during reading the file: session10.json	Missing JSON file (in example session10.json)	Check file name or path
3	user@lbc:~/work/test$ fectl session -c=FixEdge -f=session3.json Unmatched arguments: session, -c=FixEdge, -f=session3.json Did you mean: add-session or edit-session or sessions-list?	Wrong command syntax (in example "session " is not recognized as a parameter or subcommand)	read command help (e.g. fectl -h)
4	user@lbc:~/work/test$ fectl add-session -c=FixEdge1 -f=session3.json Error while running command (com.epam.fixedge.config.tool.commands.session.AddSession@4ca8195f): com.epam.fixedge.config.tool.commands.CommandException: The file should contain schedule properties in JSON format: session3.json	Wrong JSON file format (in example wrong JSON syntax in session3.json file)	check JSON file contents and format errors
5	user@lbc:~/work/test$ fectl add-session -c=FixEdge1 -f=session3.json Response: Bad Request 400 Illegal value of configuredName: may not be null	Required field is absent in JSON file (In example parameter "`ConfiguredName`" is required session parameter for `add-session` subcommand but it is absent in session3.json file)	Add parameter and value to JSON file
6	user@lbc:~/work/test$ fectl remove-session -c=FixEdge1 -s=TestFIXAcceptor10 Response: Not Found 404 There is no TestFIXAcceptor10 in cluster FixEdge1	Session not found in Cluster	Change session name in command parameters. To find out what sessions exist in Cluster see