GENESYS™

Genesys Interaction Recording Solution Guide

Elasticsearch

5/9/2025

Elasticsearch

Elasticsearch 1.x (deprecated)

Interaction Recording Web Services uses Elasticsearch — an open-source, full-text search engine with a RESTful web interface — to index recording metadata.

For new deployments, Genesys recommends that you set up a cluster of Elasticsearch nodes that is separate from your Interaction Recording Web Services nodes. See Configuring Interaction Recording Web Services to Use a Standalone Elasticsearch Cluster for details. It's possible to set up a co-located Elasticsearch cluster, which means that Elasticsearch is included in your Interaction Recording Web Services nodes. See Configuring Interaction Recording Web Services to Use a Co-located Elasticsearch Cluster for details.

Important

- If you are using GIR with Workspace Web Edition, a shared deployment of Elasticsearch should be used. Ensure you also review the Web Services and Applications documentation for Elasticsearch. For details see: Elasticsearch.

- The Elasticsearch deployment used by SpeechMiner cannot be used with Interaction Recording Web Services.

Configuring Interaction Recording Web Services to Use a Standalone Elasticsearch Cluster

You can configure Interaction Recording Web Services to work with a standalone Elasticsearch cluster by completing the steps below.

Important

Contact your Genesys representative for information about how to migrate from co-located to standalone. For additional information, refer to the latest stable 1.x version of Elasticsearch.

Prerequisites

Verify that a cluster of Elasticsearch nodes have been deployed and configured using the latest stable 1.x version of Elasticsearch. Refer to the Elasticsearch documentation for details. Note that the Elasticsearch deployment used by SpeechMiner cannot be used with Interaction Recording Web Services.

Complete the following steps for each Elasticsearch node

1. Copy the **elasticsearch.yml.sample** file from the **installation_CD/config-templates/** folder, to the Elasticsearch configuration folder on a standalone machine, and rename it to **elasticsearch.yml**. If you use **.rpm** for Elasticsearch, use **/etc/elasticsearch/** as the configuration folder. If you use the **gzipped tarball**, use **$installDir/config**.

2. Open the **elasticsearch.yml** configuration file in a text editor and verify that the following lines are included in the file:

   ```
   index.analysis.analyzer.whitespace_lowercase.tokenizer: whitespace
   index.analysis.analyzer.whitespace_lowercase.filter: lowercase
   index.analysis.analyzer.mediaPartitionAnalyzer.tokenizer: path_hierarchy
   threadpool.index.queue_size: -1
   threadpool.bulk.queue_size: -1
   bootstrap.mlockall: true
   indices.fielddata.cache.size: 75%
   indices.breaker.fielddata.limit: 80%
   path.conf: <Elasticsearch configuration path>
   path.data: <Elasticsearch installation path>/esdata
   node.name: ToBeChanged: <name of the Elasticsearch node.  Set uniquely for each node>
   cluster.name: ToBeChanged: <name of the Elasticsearch cluster>
   transport.tcp.port: 9300
   http.port: 9200
   discovery.zen.ping.multicast.enabled: false
   discovery.zen.ping.unicast.hosts: ToBeChanged: <comma separated list of Elasticsearch
   nodes>
   discovery.zen.minimum_master_nodes: ToBeChanged: <set to the minimum number of master
   nodes>
   gateway.recover_after_nodes: ToBeChanged: <calculate based on the number of
   Elasticsearch nodes with rule: '<NUMBER_ES_NODES> / 2 + 1'>
   gateway.recover_after_time: 1m
   gateway.expected_nodes: ToBeChanged: <set to the number of Elasticsearch nodes>
   ```

3. Copy the **installation_CD/elasticsearch/templates** folder, along with its **.json** file contents, to a new templates folder under the configuration folder of Elasticsearch (for example, **/etc/elasticsearch/templates** if you use **.rpm** for Elasticsearch, or **$installDir/config/templates** if you use the **gzipped tarball**) on each node.

4. Increase the permitted number of open file descriptors for the operating system on the node by referring to the documentation at https://www.elastic.co/guide/en/elasticsearch/reference/current/file-descriptors.html.

> ## Important
> The Elasticsearch engine requires a large Metaspace space. To increase the Metaspace space, pass the following argument to the JVM used to run Elasticsearch: `"-XX:MaxMetaspaceSize=512m"`

Complete the following steps for each Interaction Recording Web Services node

Configure the **application.yaml** file as follows:

1. Set the crClusterName option to the name of the cluster as specified by **cluster.name** in the **elasticsearch.yml** configuration file.
2. Set the elasticSearchSettings option to appropriate values for your environment.

Configuring Interaction Recording Web Services to Use a Co-located Elasticsearch Cluster

You can configure Interaction Recording Web Services to work with a co-located Elasticsearch cluster by completing the steps below.

> ### Important
> Contact your Genesys representative for information about how to migrate from co-located to standalone. For additional information, refer to the latest stable 1.x version of Elasticsearch.

Complete the following steps for each Interaction Recording Web Services node that you want to host Elasticsearch

1. Install Elasticsearch using the latest stable 1.x version of Elasticsearch.
2. Copy the **elasticsearch.yml.sample** file from the **installation_CD/config-templates/** folder, to the Elasticsearch configuration folder on the Interaction Recording Web Services node, and rename it to **elasticsearch.yml**. If you use **.rpm** for Elasticsearch, use **/etc/elasticsearch/** as the configuration folder. If you use the **gzipped tarball**, use **$installDir/config**.
3. Open the **elasticsearch.yml** configuration file in a text editor and verify that the following lines are included in the file: **Note: <Elasticsearch installation path>** refers to the location on which Elasticsearch has been installed.

```
index.analysis.analyzer.whitespace_lowercase.tokenizer: whitespace
index.analysis.analyzer.whitespace_lowercase.filter: lowercase
index.analysis.analyzer.mediaPartitionAnalyzer.tokenizer: path_hierarchy
threadpool.index.queue_size: -1
threadpool.bulk.queue_size: -1
bootstrap.mlockall: true
indices.fielddata.cache.size: 75%
indices.breaker.fielddata.limit: 80%
path.conf: <Elasticsearch configuration path>
path.data: <Elasticsearch installation path>/esdata
node.name: ToBeChanged: <name of the Elasticsearch node. Set uniquely for each node>
cluster.name: ToBeChanged: <name of the Elasticsearch cluster>
transport.tcp.port: 9300
http.port: 9200
discovery.zen.ping.multicast.enabled: false
discovery.zen.ping.unicast.hosts: ToBeChanged: <comma separated list of Elasticsearch
```

```
nodes>
discovery.zen.minimum_master_nodes: ToBeChanged: <set to the minimum number of master
nodes>
gateway.recover_after_nodes: ToBeChanged: <calculate based on the number of
Elasticsearch nodes with rule: '<NUMBER_ES_NODES> / 2 + 1'>
gateway.recover_after_time: 1m
gateway.expected_nodes: ToBeChanged: <set to the number of Elasticsearch nodes>
```

> ## Important
>
> Do not forget to update **<Elasticsearch installation path>** to the appropriate value.

4. Copy the **installation_CD/elasticsearch/templates** folder, along with its **.json** file contents, to a new templates folder under the configuration folder of Elasticsearch (for example, **/etc/elasticsearch/templates** if you use **.rpm** for Elasticsearch, or **$installDir/config/templates** if you use the **gzipped tarball**) on each node.

5. Increase the permitted number of open file descriptors for the operating system on the node by referring to the documentation at https://www.elastic.co/guide/en/elasticsearch/reference/current/file-descriptors.html.

6. Set the crClusterName option in the **application.yaml** file to the name of the cluster, as specified by **cluster.name** in the **elasticsearch.yml** configuration file.

7. Set the elasticSearchSettings option in the **application.yaml** file to the appropriate values for your environment.

> ## Important
>
> The Elasticsearch engine requires a large Metaspace space. To increase the Metaspace space, pass the following argument to the JVM used to run Elasticsearch: `"-XX:MaxMetaspaceSize=512m"`

Migrating an Existing Elasticsearch Deployment to Schema V2

> ## Important
>
> The following procedure should only be performed once per contact center (that is, for each contact center tenant in a multi-tenant deployment). This procedure should not be performed for a new GIR installation.

Perform the following steps while your system is running, without service interruption.

1. Copy the **call_recordingv2_template.json** and **screen_recordingv2_template.json** files from the **installation_CD/elasticsearch/templates/** folder to the **templates** folder in each node in your Elasticsearch cluster.

2. Perform a rolling restart of each node in your Elasticsearch cluster. Stop and restart each node and wait until it is restarted and is operational before stopping and restarting the next node.

3. Prepare a new dedicated **Interaction Recording Web Services** node as follows:

    a. Install Interaction Recording Web Services in the same way a regular **Interaction Recording Web Services** node is installed. Do not add this node to the Interaction Recording Web Services Load Balancer.

    b. Edit the Interaction Recording Web Services **application.yaml** file, by adding the following configuration. Verify that you add lines under nodes for all the existing Interaction Recording Web Services nodes in your deployment:

```
elasticSearchSettings:
  useTransportClient: true
  transportClient:
      nodes:
          - {host: <elastic-search-node1>, port: 9300}
          - {host: <elastic-search-node2>, port: 9300}
          - {host: <elastic-search-node3>, port: 9300}
      useSniff: true
      ignoreClusterName: true
      pingTimeout: 10000
      nodesSamplerInterval: 10000
  enableIndexVerificationAtStartUp: false
  indexPerContactCenter: true
```

> ## Important
> The **application.yaml** file on the Interaction Recording Web Services node must be set with the correct region information in the **crRegion** parameter within the **serverSettings** Call Recording section. This configuration is used as part of the new index name.

3. Increase the **Hystrix** timeout for **RecordingOperationApiTaskV2** on the new Interaction Recording Web Services node by adding the following line to the Hystrix configuration:

```
hystrix.command.RecordingOperationApiTaskV2.execution.isolation.thread.timeoutInMilliseconds=<max
time acceptable in milliseconds>
```

> ## Important
> <max time acceptable in milliseconds> should exceed the time that the re-indexing operation is expected to take. This value varies depending on how you elect to divide the re-indexing iterations. If you expect each re-indexing operation to take approximately one hour, then set this parameter to a value such as 7200000.

4. Determine the Contact Center ID using the following command:

```
curl -u <ops-user>:<ops-pass> http://< selected-web-services-node>: <web-services-
listening-port>/api/v2/ops/contact-centers; echo
```

The following output is returned:

```
{"statusCode":0,"uris":["http://< selected-web-services-node>: <web-services-
listening-port>/api/v2/ops/contact-centers/<contact-center-id>"]}
```

Migrate Call Recording Index

1. Start the migration process for call recording by issuing the following command:

```
curl -u <ops-user>:<ops-pass> -XPOST -H "Content-Type:application/json"
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/features" -d '{
     "uris":["schema-elasticsearch-migration-to-v2-call-recording"]
}'
```

2. Immediately after performing step #1, note the current time and initialize the new index by using the following command:

```
curl -u <ops-user>:<ops-pass> -XPOST -H "Content-Type:application/json"
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/recordings" -d '{
  "operationName":"forceIndex",
  "from": <start-range-in-milliseconds>,
  "to": <stop-range-in-milliseconds>,
  "purgeOld":<value>
}'
```

Genesys recommends to perform re-indexing in multiple iterations depending on how many records exist in Cassandra. The key aspect when determining **"from"** and **"to"** values is to use these parameters to specify the number of records to be re-indexed at a time. A reasonable estimate for the time taken to re-index can be 5,000,000 records in one hour, although this is dependent on your Cassandra and Elasticsearch deployment. Therefore, depending on the number of records in your deployment, this could be accomplished with a single iteration, where the **"from"** and **"to"** values specified cover the entire time range of content within Cassandra.

Important

purgeOld is set to **true** initially, and to **false** for all subsequent invocations.

3. Repeat the command from step #2 varying the "from" and "to" values to completely cover all recordings that exist (up to and including the time noted at the beginning of step 2), so that they are included in the new index.
For each iteration, ensure that **purgeOld** is set to **false** so that the newly created index is not removed.

4. Once the **forceIndex** commands are completed (so that the entire set of recordings have been re-indexed), configure Interaction Recording Web Services to use the new index by using the following command:

```
curl -u <ops-user>:<ops-pass> -XPOST -H "Content-Type:application/json"
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/features" -d '{
     "uris":["schema-elasticsearch-v2-call-recording"]
}'
```

5. Verify that the Search functionality is working properly using the **GetRecordings API**. For additional information refer to Genesys Interaction Recording API.

> ## Important
>
> SpeechMiner cannot be used to perform this validation since it uses a different mechanism to search for call recordings.

Once this procedure is completed both the old index and the new index are maintained and the new index is used for all searches.

Migrate Screen Recording Index

1. Start the migration process for screen recording by issuing the following command:

```
curl -u <ops-user>:<ops-pass> -XPOST -H "Content-Type:application/json"
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/features" -d '{
     "uris":["schema-elasticsearch-migration-to-v2-screen-recording"]
}'
```

2. Immediately after performing step #1, note the current time and initialize the new index by using the following command:

```
curl -u <ops-user>:<ops-pass> -XPOST -H "Content-Type:application/json"
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/screen-recordings" -d '{
  "operationName":"forceIndex",
  "from": <start-range-in-milliseconds>,
```

```
  "to": <stop-range-in-milliseconds>,
  "purgeOld":<value>
}'
```

Genesys recommends to perform re-indexing in multiple iterations depending on how many records exist in Cassandra. The key aspect when determining **"from"** and **"to"** values is to use these parameters to specify the number of records to be re-indexed at a time. A reasonable estimate for the time taken to re-index can be 5,000,000 records in one hour, although this is dependent on your Cassandra and Elasticsearch deployment. Therefore, depending on the number of records in your deployment, this could be accomplished with a single iteration, where the **"from"** and **"to"** values specified cover the entire time range of content within Cassandra.

> ## Important
>
> **purgeOld** is set to **true** initially, and to **false** for all subsequent invocations.

3. Repeat the command from step #2 varying the "from" and "to" values to completely cover all recordings that exist (up to and including the time noted at the beginning of step 2), so that they are included in the new index.
   For each iteration, ensure that **purgeOld** is set to **false** so that the newly created index is not removed.

4. Once the **forceIndex** commands are completed (so that the entire set of recordings have been re-indexed), configure Interaction Recording Web Services to use the new index by using the following command:

```
curl -u <ops-user>:<ops-pass> -XPOST -H "Content-Type:application/json"
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/features" -d '{
     "uris":["schema-elasticsearch-v2-screen-recording"]
}'
```

5. Verify that the Search functionality is working properly against the full range of screen recordings, by using the SpeechMiner Screen Recording grid.

   Once this procedure is completed both the old index and the new index are maintained and the new index is used for all searches.

Completing the Migration

Once you have migrated both the Call Recording and Screen Recording indexes, both the old index and the new index are updated for every new recording. This process consumes additional disk space. To avoid the use of additional disk space, perform the following steps to remove the old indexes once testing has confirmed that the new indexes are fully operational:

> ## Important

> Once the following steps are performed, it will not be possible to roll back the migration.

1. Turn off the schema migration feature flag for the index being migrated, by using the following command:

   For **Call Recordings**:

   ```
   curl -u <ops-user>:<ops-pass> -XDELETE -H "Content-Type:application/json"
   "http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
   center-id>/features" -d '{
        "uris":["schema-elasticsearch-migration-to-v2-call-recording"]
   }'
   ```

   For **Screen Recordings**:

   ```
   curl -u <ops-user>:<ops-pass> -XDELETE -H "Content-Type:application/json"
   "http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
   center-id>/features" -d '{
        "uris":["schema-elasticsearch-migration-to-v2-screen-recording"]
   }'
   ```

2. Delete the old indexes by using the following command:

   ```
   curl -XDELETE http://<es-node>:9200/<index-name>
   ```

   Where:

   - **<es-node>** is one of the Elasticsearch nodes in the cluster.

   - **<index-name>** is the index name for the original schema:

     - **{contact-center-id}** for call recording when an embedded Elasticsearch cluster is used, or **call-recording-{contact-center-id}** for a standalone Elasticsearch cluster deployment. For example, **f3eec6cb-f624-4ac2-975e-6a60e0ebf878** or **call-recording-f3eec6cb-f624-4ac2-975e-6a60e0ebf878**.

     - **screen-recording-{contact-center-id}** for screen recording. For example, **screen-recording-f3eec6cb-f624-4ac2-975e-6a60e0ebf878>**.

   If you are unsure, the index names in use on Elasticsearch can be determined by using the following command (where **<es-node>** is one of the Elasticsearch nodes in the cluster):

   ```
   curl -XGET http://<es-node>:9200/_cat/indices?v
   ```

> **Important**
>
> At this point the new Interaction Recording Services node that was used for these migration steps is no longer required and can be shut down or re-purposed.

Rolling Back the Migration

In the event of a problem with the index migration, perform the following steps to implement the old (previous) index and remove the new index.

1. If the new index was enabled as the default index, run the following command to use the old index:

   For **Call Recordings**:

   ```
   curl -u <ops-user>:<ops-pass> -XDELETE -H "Content-Type:application/json"
    "http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
    center-id>/features" -d '{
          "uris":["schema-elasticsearch-v2-call-recording"]
    }'
   ```

   For **Screen Recordings**:

   ```
   curl -u <ops-user>:<ops-pass> -XDELETE -H "Content-Type:application/json"
    "http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
    center-id>/features" -d '{
          "uris":["schema-elasticsearch-v2-screen-recording"]
    }'
   ```

2. Stop updates to the new index by turning off the schema migration feature flag, using the following command:

   For **Call Recordings**:

   ```
   curl -u <ops-user>:<ops-pass> -XDELETE -H "Content-Type:application/json"
    "http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
    center-id>/features" -d '{
          "uris":["schema-elasticsearch-migration-to-v2-call-recording"]
    }'
   ```

   For **Screen Recordings**:

   ```
   curl -u <ops-user>:<ops-pass> -XDELETE -H "Content-Type:application/json"
   ```

```
"http://<selected-RWS-node>:<RWS-listening-port>/api/v2/ops/contact-centers/<contact-
center-id>/features" -d '{
        "uris":["schema-elasticsearch-migration-to-v2-screen-recording"]
}'
```

3. To delete the new index, run the following command:

```
curl -XDELETE http://<es-node>:9200/<index-name>
```

Where:

- **<es-node>** is one of the Elasticsearch nodes in the cluster.

- **<index-name>** is the index name for the new schema:

    - **call-recording-v2-{region}-{contact-center-id}** for call recordings.

    - **screen-recording-v2-{region}-{contact-center-id}** for screen recordings.

    - **region** is the value of the **crRegion** parameter specified in the **application.yaml** file in the Interaction Recording Web Services node that was used to perform the index migration process for both the call recording index and the screen recording index.

Elasticsearch 7.16.3

Interaction Recording Web Services uses Elasticsearch — an open-source, full-text search engine with a RESTful web interface — to index recording metadata.

For new deployments, Genesys recommends that you set up a cluster of Elasticsearch nodes that are separate from your Interaction Recording Web Services nodes. For more details, see Configuring Interaction Recording Web Services to Use a Standalone Elasticsearch Cluster. You can also set up a co-located Elasticsearch cluster, which means that Elasticsearch is included in your Interaction Recording Web Services nodes. For more details, see Configuring Interaction Recording Web Services to Use a Co-located Elasticsearch Cluster.

> ### Important
>
> - If you are using GIR with Workspace Web Edition, refrain from using a shared deployment of Elasticsearch. This is because Web Services and Applications support Elasticsearch 1.x only and do not support ES 7.16.3. For details see: Elasticsearch. This is applicable if you are installing Web Services and Applications version 8.5.201.09 or earlier.
>
> - The Elasticsearch deployment used by SpeechMiner cannot be used with Interaction Recording Web Services.

Configuring Interaction Recording Web Services to Use a Standalone Elasticsearch Cluster

You can configure Interaction Recording Web Services to work with a standalone Elasticsearch cluster by completing the steps below.

> ### Important
> Contact your Genesys representative for information about how to migrate from co-located to standalone. For additional information, refer to the latest stable 7.16.3 version of Elasticsearch.

Prerequisites

- Verify that a cluster of Elasticsearch nodes have been deployed and configured using the latest stable 7.16.3 version of Elasticsearch. Refer to the Elasticsearch documentation for details. Note that the Elasticsearch deployment used by SpeechMiner cannot be used with Interaction Recording Web Services.

- Interaction Recording Web Services deployment version should be 8.5.204.16 or higher.

- Interaction Recording Web Services supported Elasticsearch 7.16.3 installed on RedHat 8/9 and Java 11.

Limitations of Elasticsearch 7.16.3

- Elasticsearch 7.16.3 only supports schema V3.

- Genesys is not responsible for migration of existing data to the latest version of Elasticsearch on premise environments.

- Interaction Recording Web Services does not support scan and scroll functionality on Elasticsearch 7.16.3.

Complete the following steps for each Elasticsearch node

1. Refer to the Elasticsearch documentation for installing and configuring Elasticsearch 7.16.3.

2. Open the following **elasticsearch.yml** configuration file in a text editor and verify that the following lines are included in the file:

```
cluster.name: <Use a descriptive name for your cluster>
node.name: <Use a descriptive name for the node>
node.attr.rack: <Add custom attributes to the node>
path.data: <Path to directory where to store the data (separate multiple locations by
comma)>
path.logs: <Path to log files>
bootstrap.memory_lock: <Lock the memory on startup>
network.host: <By default, Elasticsearch is only accessible on localhost. Set a
different address here to expose this node on the network>
http.port: <Set a specific HTTP port here, by default is 9200>
```

```
discovery.seed_hosts: <Pass an initial list of hosts to perform discovery when this node
is started>
cluster.initial_master_nodes: <Bootstrap the cluster using an initial set of master-
eligible nodes>
action.destructive_requires_name: <Requires explicit names when deleting indices>
```

3. Increase the permitted number of open file descriptors for the operating system on the node by referring to the documentation at https://www.elastic.co/guide/en/elasticsearch/reference/7.16/file-descriptors.html.

Complete the following steps for each Interaction Recording Web Services node

Configure the **application.yaml** file as follows:

1. Set the crClusterName option to the name of the cluster as specified by **cluster.name** in the **elasticsearch.yml** configuration file.

2. Set the elasticSearchSettings option to appropriate values for your environment.

Configuring Interaction Recording Web Services to Use a Co-located Elasticsearch Cluster

You can configure Interaction Recording Web Services to work with a co-located Elasticsearch cluster by completing the steps below.

> ## Important
>
> Contact your Genesys representative for information about how to migrate from co-located to standalone as co-located is discontinued after RWS version 8.5.205.69. For additional information, refer to the latest stable 7.16.3 version of Elasticsearch.

Complete the following steps for each Interaction Recording Web Services node that you want to host Elasticsearch

1. Refer to the Elasticsearch documentation for installing and configuring Elasticsearch 7.16.3.

2. Open the **elasticsearch.yml** configuration file in a text editor and verify that the following lines are included in the file:

```
cluster.name: <Use a descriptive name for your cluster>
node.name: <Use a descriptive name for the node>
node.attr.rack: <Add custom attributes to the node>
path.data: <Path to directory where to store the data (separate multiple locations by
comma)>
path.logs: <Path to log files>
bootstrap.memory_lock: <Lock the memory on startup>
network.host: <By default, Elasticsearch is only accessible on localhost. Set a
different address here to expose this node on the network>
http.port: <Set a specific HTTP port here, by default is 9200>
discovery.seed_hosts: <Pass an initial list of hosts to perform discovery when this node
is started>
```

```
cluster.initial_master_nodes: <Bootstrap the cluster using an initial set of master-
eligible nodes>
action.destructive_requires_name: <Requires explicit names when deleting indices>
```

3. Increase the permitted number of open file descriptors for the operating system on the node by referring to the documentation at https://www.elastic.co/guide/en/elasticsearch/reference/7.16/file-descriptors.html.

4. Set the crClusterName option in the **application.yaml** file to the name of the cluster, as specified by **cluster.name** in the **elasticsearch.yml** configuration file.

5. Set the elasticSearchSettings option in the **application.yaml** file to the appropriate values for your environment.

Perform the following steps in each Elasticsearch node while your system is running, without service interruption

1. Create call recording schema V3:

   a. Copy the **call_recording_v3_template.json** file from the **installation_CD/elasticsearch/templates/** folder to the **local temp** folder.

   b. Create call-recording schema V3 using the following command:

   ```
   $ curl -XPUT http://<Elasticsearch Instance hostname>:<Elasticsearch Instance HTTP
   port>/_template/call-recording-v3-template?include_type_name=false -H "Content-
   Type:application/json" -d@call_recording_v3_template.json
   ```

3. Create screen recording schema V3:

   a. Copy the **screen_recording_v3_template.json** file from the **installation_CD/elasticsearch/templates/** folder to the **local temp** folder.

   b. Create screen-recording V3 schema using the following command:

   ```
   $ curl -XPUT http://<Elasticsearch Instance hostname>:<Elasticsearch Instance HTTP
   port>/_template/screen-recording-v3-template?include_type_name=false -H "Content-
   Type:application/json" -d@screen_recording_v3_template.json
   ```

Configure Interaction Recording Web Services using Elasticsearch 7.16.3

1. Update the **application.yaml** file on each Interaction Recording Web Services node.

2. update **useTransportClient** to be "false".

3. Add a new property **useRestClient** as follows:

```
elasticSearchSettings:
  useTransportClient: false
  transportClient:
      nodes:
          - {host: <elastic-search-node1>, port: 9300}
          - {host: <elastic-search-node2>, port: 9300}
          - {host: <elastic-search-node3>, port: 9300}
      useSniff: false
      ignoreClusterName: false
      pingTimeout: 5000
      nodesSamplerInterval: 5000
  useRestClient: true
```

```
restClient:
   nodes:
      - {host: <elastic-search-node1>, port: 9200}
      - {host: <elastic-search-node2>, port: 9200}
      - {host: <elastic-search-node3>, port: 9200}
waitToIndexTimeout: 5000
scanReadTimeoutSeconds: 60
scrollTimeoutSeconds: 240
countReadTimeoutSeconds: 60
```

Important

The **application.yaml** file on the Interaction Recording Web Services node must be set with the correct region information in the **crRegion** parameter within the **serverSettings** Call Recording section. This configuration is used as part of the new index name.

Elasticsearch 7.17.15

Interaction Recording Web Services uses Elasticsearch — an open-source, full-text search engine with a RESTful web interface — to index recording metadata.

For new deployments, Genesys recommends that you set up a cluster of Elasticsearch nodes that are separate from your Interaction Recording Web Services nodes. For more details, see Configuring Interaction Recording Web Services to Use a Standalone Elasticsearch Cluster. You can also set up a co-located Elasticsearch cluster, which means that Elasticsearch is included in your Interaction Recording Web Services nodes. For more details, see Configuring Interaction Recording Web Services to Use a Co-located Elasticsearch Cluster.

Important

- If you are using GIR with Workspace Web Edition, refrain from using a shared deployment of Elasticsearch. This is because Web Services and Applications support Elasticsearch 1.x only and do not support ES 7.17.15. For details see: Elasticsearch. This is applicable if you are installing Web Services and Applications version 8.5.201.09 or earlier.

- The Elasticsearch deployment used by SpeechMiner cannot be used with Interaction Recording Web Services.

- Interaction Recording Web Services 8.5.500.14 (or higher) will support only ES 7.17.15.

Configuring Interaction Recording Web Services to Use a Standalone Elasticsearch Cluster

You can configure Interaction Recording Web Services to work with a standalone Elasticsearch cluster by completing the steps below.

> ### Important
> Contact your Genesys representative for information about how to migrate from co-located to standalone. For additional information, refer to the latest stable 7.17.15 version of Elasticsearch.

Prerequisites

- Verify that a cluster of Elasticsearch nodes have been deployed and configured using the latest stable 7.17.15 version of Elasticsearch. Refer to the Elasticsearch documentation for details. Note that the Elasticsearch deployment used by SpeechMiner cannot be used with Interaction Recording Web Services.

- Interaction Recording Web Services deployment version should be 8.5.205.32 or higher.

- Interaction Recording Web Services supported Elasticsearch 7.17.15 installed on RedHat 8/9 and Java 11.

Limitations of Elasticsearch 7.17.15

- Elasticsearch 7.17.15 only supports schema V3.

- Genesys is not responsible for migration of existing data to the latest version of Elasticsearch on premise environments.

Complete the following steps for each Elasticsearch node

1. Refer to the Elasticsearch documentation for installing and configuring Elasticsearch 7.17.15.

2. Open the following **elasticsearch.yml** configuration file in a text editor and verify that the following lines are included in the file:

```
cluster.name: <Use a descriptive name for your cluster>
node.name: <Use a descriptive name for the node>
node.attr.rack: <Add custom attributes to the node>
path.data: <Path to directory where to store the data (separate multiple locations by comma)>
path.logs: <Path to log files>
bootstrap.memory_lock: <Lock the memory on startup>
network.host: <By default, Elasticsearch is only accessible on localhost. Set a different address here to expose this node on the network>
http.port: <Set a specific HTTP port here, by default is 9200>
discovery.seed_hosts: <Pass an initial list of hosts to perform discovery when this node is started>
cluster.initial_master_nodes: <Bootstrap the cluster using an initial set of master-eligible nodes>
```

```
action.destructive_requires_name: <Requires explicit names when deleting indices>
```

3. Increase the permitted number of open file descriptors for the operating system on the node by referring to the documentation at https://www.elastic.co/guide/en/elasticsearch/reference/7.17/file-descriptors.html.

Complete the following steps for each Interaction Recording Web Services node

Configure the **application.yaml** file as follows:

1. Set the crClusterName option to the name of the cluster as specified by **cluster.name** in the **elasticsearch.yml** configuration file.

2. Set the elasticSearchSettings option to appropriate values for your environment.

Configuring Interaction Recording Web Services to Use a Co-located Elasticsearch Cluster

You can configure Interaction Recording Web Services to work with a co-located Elasticsearch cluster by completing the steps below.

> ### Important
>
> Contact your Genesys representative for information about how to migrate from co-located to standalone as co-located is discontinued after RWS version 8.5.205.69. For additional information, refer to the latest stable 7.17.15 version of Elasticsearch.

Complete the following steps for each Interaction Recording Web Services node that you want to host Elasticsearch

1. Refer to the Elasticsearch documentation for installing and configuring Elasticsearch 7.17.15.

2. Open the **elasticsearch.yml** configuration file in a text editor and verify that the following lines are included in the file:

```
cluster.name: <Use a descriptive name for your cluster>
node.name: <Use a descriptive name for the node>
node.attr.rack: <Add custom attributes to the node>
path.data: <Path to directory where to store the data (separate multiple locations by comma)>
path.logs: <Path to log files>
bootstrap.memory_lock: <Lock the memory on startup>
network.host: <By default, Elasticsearch is only accessible on localhost. Set a different address here to expose this node on the network>
http.port: <Set a specific HTTP port here, by default is 9200>
discovery.seed_hosts: <Pass an initial list of hosts to perform discovery when this node is started>
cluster.initial_master_nodes: <Bootstrap the cluster using an initial set of master-eligible nodes>
action.destructive_requires_name: <Requires explicit names when deleting indices>
```

3. Increase the permitted number of open file descriptors for the operating system on the node by referring to the documentation at https://www.elastic.co/guide/en/elasticsearch/reference/7.17/file-descriptors.html.

4. Set the crClusterName option in the **application.yaml** file to the name of the cluster, as specified by **cluster.name** in the **elasticsearch.yml** configuration file.

5. Set the elasticSearchSettings option in the **application.yaml** file to the appropriate values for your environment.

Perform the following steps in each Elasticsearch node while your system is running, without service interruption

1. Create call recording schema V3:

   a. Copy the **call_recording_v3_template.json** file from the **installation_CD/elasticsearch/templates/** folder to the **local temp** folder.

   b. Create call-recording schema V3 using the following command:

   ```
   $ curl -XPUT http://<Elasticsearch Instance hostname>:<Elasticsearch Instance HTTP
   port>/_template/call-recording-v3-template?include_type_name=false -H "Content-
   Type:application/json" -d@call_recording_v3_template.json
   ```

3. Create screen recording schema V3:

   a. Copy the **screen_recording_v3_template.json** file from the **installation_CD/elasticsearch/templates/** folder to the **local temp** folder.

   b. Create screen-recording V3 schema using the following command:

   ```
   $ curl -XPUT http://<Elasticsearch Instance hostname>:<Elasticsearch Instance HTTP
   port>/_template/screen-recording-v3-template?include_type_name=false -H "Content-
   Type:application/json" -d@screen_recording_v3_template.json
   ```

Configure Interaction Recording Web Services using Elasticsearch 7.17.15

1. Update the **application.yaml** file on each Interaction Recording Web Services node.

2. update **useTransportClient** to be "false".

3. Add a new property **useRestClient** as follows:

```
elasticSearchSettings:
   useTransportClient: false
   transportClient:
      nodes:
          - {host: <elastic-search-node1>, port: 9300}
          - {host: <elastic-search-node2>, port: 9300}
          - {host: <elastic-search-node3>, port: 9300}
      useSniff: false
      ignoreClusterName: false
      pingTimeout: 5000
      nodesSamplerInterval: 5000
   useRestClient: true
   restClient:
      nodes:
          - {host: <elastic-search-node1>, port: 9200}
          - {host: <elastic-search-node2>, port: 9200}
```

```
        - {host: <elastic-search-node3>, port: 9200}
waitToIndexTimeout: 5000
scanReadTimeoutSeconds: 60
scrollTimeoutSeconds: 240
countReadTimeoutSeconds: 60
```

> ### Important
>
> The **application.yaml** file on the Interaction Recording Web Services node must be set with the correct region information in the **crRegion** parameter within the **serverSettings** Call Recording section. This configuration is used as part of the new index name.

Next Step

- Install Interaction Recording Web Services.