

ARIS PROCESS MINING DATA INGESTION API

VERSION 10.0 - SERVICE RELEASE 27 AND HIGHER
OKTOBER 2024

This document applies to ARIS Process Mining Version 10.0 and to all subsequent releases. Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2020 - 2024 Software GmbH, Darmstadt, Germany and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software GmbH product names are either trademarks or registered trademarks of Software GmbH and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software GmbH and/or its subsidiaries is located at <https://softwareag.com/licenses>.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software GmbH Products / Copyright and Trademark Notices of Software GmbH Products". These documents are part of the product documentation, located at <https://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software GmbH.

Inhalt

| | | |
|---------|--|----|
| 1 | Daten mit einer öffentlichen API aufnehmen | 1 |
| 1.1 | Anmerkungen | 2 |
| 1.2 | Vorbereitungen in ARIS Process Mining | 2 |
| 1.2.1 | Systemintegration für die Data Ingestion API erstellen | 2 |
| 1.2.2 | Verbindung im Datenset erstellen | 3 |
| 1.3 | Data Ingestion API verwenden | 5 |
| 1.3.1 | API-Mandant authentifizieren | 5 |
| 1.3.1.1 | Authentifizierung über URL-Parameter ist veraltet | 8 |
| 1.3.1.2 | Technischen Schlüssel des Datensets notieren | 8 |
| 1.3.2 | Quellentabellen-Definitionen abrufen | 9 |
| 1.3.3 | Quellentabellen erstellen oder ersetzen | 9 |
| 1.3.4 | Prüfen, ob das Datenset zum Hochladen bereit ist | 11 |
| 1.3.5 | Daten-Upload-Zyklus erstellen | 12 |
| 1.3.6 | Daten hochladen | 13 |
| 1.3.7 | Daten-Upload-Zyklus committen | 13 |
| 1.3.8 | Zyklus-Status abrufen | 14 |
| 1.3.9 | Prüfen, ob das Datenset zum Laden bereit ist | 14 |
| 1.3.10 | Laden der Daten starten | 15 |
| 1.3.11 | Zyklus-Status abrufen | 15 |
| 1.3.12 | Quellentabelle einfügen | 16 |
| 1.3.13 | Ingestionszyklen abrufen | 16 |
| 1.3.14 | Ingestionszyklus abbrechen | 16 |
| 1.4 | API-Methoden | 16 |
| 1.4.1 | Pfadabschnitt: Datenset | 16 |
| 1.4.2 | Pfadabschnitt: Ingestionszyklus | 17 |
| 1.4.3 | API-Version abrufen | 17 |
| 1.4.4 | Quellentabellen-Definitionen abrufen | 17 |
| 1.4.5 | Quellentabellen erstellen oder ersetzen | 18 |
| 1.4.6 | Quellentabellendefinition aktualisieren | 19 |
| 1.4.7 | Prüfen, ob Datenset für die Ingestion bereit ist | 20 |
| 1.4.8 | Neuen Ingestionszyklus erstellen | 20 |
| 1.4.9 | Ingestionszyklus abbrechen | 21 |
| 1.4.10 | Daten hochladen | 21 |
| 1.4.11 | Quellentabelle einfügen | 22 |
| 1.4.12 | Daten-Upload-Zyklus committen | 22 |
| 1.4.13 | Ingestionszyklen abrufen | 22 |
| 1.4.14 | Status des Aufnahmezyklus zurückgeben | 22 |
| 1.5 | Data Transfer Objects (DTOs) | 23 |
| 1.5.1 | SourceTableDefinition | 23 |
| 1.5.2 | DataIngestionReadyState | 25 |
| 1.5.3 | DataIngestionCycle | 27 |
| 1.5.4 | DataIngestionCycleState | 29 |
| 1.5.5 | Authentifizierungsantwort | 30 |
| 1.6 | Wertvolle Informationen | 31 |
| 1.6.1 | Persistenzmodus | 31 |
| 1.6.2 | Grenzwerte | 32 |

| | | |
|-----|--|----|
| 1.7 | Konnektor webMethods.io für ARIS Process Mining..... | 34 |
| 2 | Support und rechtliche Hinweise..... | 36 |
| 2.1 | Dokumentationsumfang | 36 |
| 2.2 | Datenschutz..... | 37 |
| 2.3 | Support | 37 |

1 Daten mit einer öffentlichen API aufnehmen

ARIS Process Mining unterstützt eine öffentliche Data Ingestion API. Sie verwenden die API, indem Sie HTTP-Anfragen erstellen und senden. Mithilfe der API können Sie Daten aus einer beliebigen Datenquelle nach ARIS Process Mining übertragen. Die nach ARIS Process Mining übertragenen Daten liegen in einer logischen Tabellenstruktur vor und müssen dem JSON-Format entsprechen.

Sie können einen entsprechenden API-Mandanten verwenden, um die HTTP-Anfragen zu erstellen und Sie benötigen entsprechende API-Programmierkenntnisse.

Im Kapitel Data Transfer Objects (DTOs) (Seite 23) finden Sie eine Liste der DTOs, die Sie zur Übertragung der Daten verwenden können.

Das Kapitel API-Methoden (Seite 16) enthält eine Liste der Endpunkte, die Sie für Ihre HTTP-Anfragen verwenden können.

VORBEREITUNGEN IN ARIS PROCESS MINING

Damit Daten mithilfe der API nach ARIS Process Mining übertragen werden können, sind die folgenden Schritte erforderlich:

- Systemintegration für die Data Ingestion API anlegen (Seite 2).
- Erstellen Sie ein Datenset zum Speichern der übertragenen Daten.
- Verbindung für die Data Ingestion API erstellen. (Seite 3)

DATA INGESTION API ZUR ÜBERTRAGUNG VON DATEN VERWENDEN

Die folgenden Schritte haben sich bei der Übertragung von Daten mithilfe einer API bewährt.

- API-Mandant authentifizieren (Seite 5)
- Quelltabellen-Definitionen abrufen (Seite 9)
- Quelltabellen erstellen oder ersetzen (Seite 9)
- Prüfen, ob das Datenset zum Hochladen bereit ist (Seite 11)
- Daten-Upload-Zyklus erstellen (Seite 11)
- Daten hochladen (Seite 12)
- Daten-Upload-Zyklus committen (Seite 13)
- Zyklus-Status abrufen (Seite 13)
- Prüfen, ob das Datenset zum Laden bereit ist (Seite 14)
- Laden der Daten starten (Seite 14)
- Zyklus-Status abrufen (Seite 15)

Bei Bedarf können Sie auch die folgenden Schritte ausführen.

- Quelltabelle einfügen (Seite 15)
- Ingestionszyklen abrufen (Seite 16)
- Ingestionszyklus abbrechen (Seite 16)

1.1 Anmerkungen

Beachten Sie, dass neue JSON-Felder und Aufzählungswerte im Rahmen kleinerer API-Revisionen von der Software GmbH zu Ingestion API-Eingaben oder -Ausgaben hinzugefügt werden können.

1.2 Vorbereitungen in ARIS Process Mining

1.2.1 Systemintegration für die Data Ingestion API erstellen

Damit Sie die Data Ingestion API verwenden können (Seite 1), müssen Sie eine entsprechende Systemintegration anlegen.

ARIS Process Mining unterstützt die OAuth2-Flüsse mit den Gewährungstypen **Mandant-Anmeldedaten** und **Autorisierungscode** als Authentifizierungsmethoden.

Die Authentifizierungsmethoden mit Mandant-Anmeldedaten liegen außerhalb des Kontexts eines Benutzers und werden für die Maschine-zu-Maschine-Kommunikation empfohlen.

Voraussetzung

Sie haben die Lizenz ARIS Process Mining Enterprise installiert.

Vorgehen

1. Klicken Sie in der Programmkopfzeile auf das Symbol ☰ **Navigationsmenü** > **Administration**.
2. Klicken Sie im Bereich **Administration** auf **Systemintegration**.
3. Klicken Sie auf **Systemintegration hinzufügen** > **Dateningestion (API)**. Der zugehörige Dialog wird geöffnet.
4. Geben Sie einen Namen ein, z. B. "Datenaufnahme" sowie optional eine Beschreibung.
5. Wählen Sie im Dropdownmenü **Gewährungstyp (OAuth)** eine Authentifizierungsmethode aus.

Die empfohlene Authentifizierungsmethode ist **Mandant-Anmeldedaten**. Sie ist für die Maschine-zu-Maschine-Kommunikation vorgesehen und liegt außerhalb des Kontexts eines tatsächlich angemeldeten Benutzers.

Wenn Sie den Gewährungstyp **Autorisierungscode** wählen, geben Sie die für die Authentifizierung verwendete **Autorisierungs-Callback-URL** an.

`https://<region.ariscloud>/umc/rest/oauth/callback?tenant=<project_room>&provider=umc`

Ersetzen Sie <region.ariscloud> durch den Hostnamen der ARIS Process Mining-Installation und <project_room> durch den ARIS Process Mining-Projektraum, bei dem Sie sich anmelden möchten.

Den Hostnamen der URL (z. B. processmining.ariscloud.com) können Sie in der Adressleiste des Browsers ablesen, wenn Sie angemeldet sind.

Beispiele

Autorisierungs-Callback-URL für die ARIS Cloud

`https://processmining.ariscloud.com/umc/rest/oauth/callback?tenant=myprojectroom&provider=umc`

Autorisierungs-Callback-URL für die ARIS Enterprise Cloud

`https://<my_company_name>.ariscloud.com/umc/rest/oauth/callback?tenant=<project_room>&provider=umc`

6. Klicken Sie auf **Hinzufügen**. Der Dialog **Zugangsdaten für Dateningestion** wird geöffnet. Im Dialog werden die Mandanten-ID, der geheime Schlüssel und der Projektraumname angezeigt.

Wenn Sie den Gewährungstyp **Autorisierungscode** gewählt haben, wird zusätzlich die bekannte URL angezeigt.

7. Sie können alle angezeigten Authentifizierungsdaten speichern, z. B. mithilfe eines Texteditors.

Klicken Sie auf **In die Zwischenablage kopieren**, und speichern Sie die Daten.

8. Klicken Sie auf **Fertig**.

Die Systemintegration wird erstellt und mit dem von Ihnen angegebenen Namen aufgeführt.

Beachten Sie, dass die Systemintegration der Data Ingestion API standardmäßig den Status **Ausstehend** beibehält. Die Systemintegration kann trotzdem ganz normal verwendet werden.

Tipp

Die Zugangsdaten (außer für die Endpunkte) werden in der von Ihnen erstellten Systemintegration gespeichert. Sie können die Quellsystemzugangsdaten für den Zugriff auf den Mandant-Anmeldeschlüssel aufrufen.

1.2.2 Verbindung im Datenset erstellen

Bevor Sie Daten nach ARIS Process Mining mithilfe der Ingestion-API übertragen können, müssen Sie eine entsprechende Verbindung für das Datenset erstellen, in dem die übertragenen Daten gespeichert werden. Eine Verbindung zum API-Mandant stellen Sie mithilfe der angelegten Systemintegration (Seite 2) her.

Vorgehen

1. Klicken Sie in der Programmkopfzeile auf das Symbol **☰ Navigationsmenü**.
2. Wählen Sie **☰ Datensammlung**. Die **Datensammlung** wird geöffnet und zeigt die **Datenset**-Seite an.

- a. Wenn Sie bereits ein Datenset geöffnet haben, wird das zuletzt geöffnete Datenset geöffnet. Klicken Sie im Bereich **Datenset** auf **Zurück**, um die Seite **Datenset** zu öffnen.
 - b. Klicken Sie auf der Seite **Datensets** auf das Datenset. Das ausgewählte Datenset wird geöffnet.
3. Öffnen Sie die Komponente **Verbindungen**.
 4. Klicken Sie auf **Verbindung hinzufügen**. Wenn Sie zum ersten Mal eine Verbindung zu einem Quellsystem herstellen und dem Datenset noch keine 'Living Process'-Lizenz zugewiesen haben, wird der Dialog "**Living Process**"-Lizenz zuweisen geöffnet.
 5. Wählen Sie im Dropdownmenü eine Lizenz aus. Die Lizenz 'Living Process' ist für das Extrahieren und Analysieren der Prozesse erforderlich. Die Anzahl der Prozesse, die extrahiert werden können, hängt von der ausgewählten Lizenz ab.

Assign 'Living Process' license



Enhance your data set capabilities

To connect external systems and to continuously update your data, you need to assign a 'Living Process' license to the data set.

License

Living Process 'L' - (25m cases) 

[Learn more](#)

Assign

Cancel

6. Klicken Sie auf **Zuweisen**. Der Dialog **Verbindung hinzufügen** wird geöffnet.
7. Konfigurieren Sie die Verbindung.
 - a. Geben Sie einen eindeutigen Namen für die Verbindung zum Quellsystem ein, z. B. "Datenaufnahme".

- b. Wählen Sie die für die Data Ingestion API angelegte Systemintegration aus.
- c. Klicken Sie auf **Hinzufügen**.

Sie haben für die API eine Verbindung erstellt. Die erstellte Verbindung wird auf der Seite **Verbindungen** mit den von Ihnen festgelegten Einstellungen angezeigt.

1.3 Data Ingestion API verwenden

Die folgenden Schritte haben sich bei der Übertragung von Daten mithilfe der Data Ingestion API bewährt.

Die unten beschriebenen Operationen sind bei Verwendung des webMethods.io-Konnektors für ARIS Process Mining (Seite 34) als vordefinierte Operationen verfügbar.

1.3.1 API-Mandant authentifizieren

Zum Authentifizieren Ihres Mandanten bei ARIS Process Mining müssen Sie eine HTTP-Authentifizierungsanfrage ausführen. Je nach festgelegter Authentifizierungsmethode (Seite 2) können Sie die Mandant-Anmeldedaten oder einen Autorisierungscode verwenden.

Sie finden die erforderlichen Daten in der für die Data Ingestion API erstellten Systemintegration (Seite 2).

AUTHENTIFIZIERUNG ANHAND DER ARIS CLOUD MITHILFE VON MANDANT-ANMELDEDATEN

Wir empfehlen dringend, die Authentifizierung mittels Mandant-Anmeldedaten durchzuführen.

Wenn Sie sich über die URL **mc.ariscloud.com** in Ihrem Projektraum anmelden, heißt das, dass Sie die ARIS Cloud verwenden.

Senden Sie eine HTTP-Post-Anfrage an den ARIS-Cloud-Endpunkt und -Pfad **/api/applications/login** (z. B. <https://mc.ariscloud.com/api/applications/login>) mit den folgenden Eigenschaften:

- Inhaltstyp: application/x-www-form-urlencoded
- Anfragetext mit Werten aus der entsprechenden Systemintegration (Seite 2):

clientId: Mandanten-ID

clientSecret: Mandantengeheimnis

tenant: Projektraumname

Die Antwort ist ein JSON-Objekt, das aus dem Mandanten, einer URL und einem Zugriffstoken besteht.

```
{
```

```
"tenant": "<project_room>",  
"token": "...",  
"url": "https://some_url"  
}
```

Hinweis

Verwenden Sie den Wert der URL als Hostnamen für alle nachfolgenden Aufrufe an die REST-Endpunkte.

Stellen Sie sicher, dass das generierte Bearer-Token mit den entsprechenden Headern für alle weiteren Anfragen gesendet wird. Fügen Sie hierzu jeder Anfrage diesen HTTP-Request-Header wie folgt hinzu:

```
Autorisierung: Bearer <token from response>
```

Beachten Sie das Leerzeichen nach dem Begriff "Bearer".

AUTHENTIFIZIERUNG ANHAND DER ARIS ENTERPRISE CLOUD MIT ARIS BENUTZERVERWALTUNG MITHILFE VON MANDANT-ANMELDEDATEN

Wir empfehlen dringend, die Authentifizierung mittels Mandant-Anmeldedaten durchzuführen.

Senden Sie über den Pfad **/umc/api/oauth/apptoken** (Beispiel: https://my_company_name.arisccloud.com/umc/api/oauth/apptoken) eine HTTP-Post-Anfrage an die ARIS Benutzerverwaltung mit den folgenden Eigenschaften:

- Inhaltstyp: application/x-www-form-urlencoded
- Anfragetext mit Werten aus der entsprechenden Systemintegration (Seite 2):

client_id: Mandanten-ID

client_secret: Mandantengeheimnis

tenant: Projektraumname

grant_type: client_credentials

Die Antwort ist ein JSON-Objekt, das aus einem Anwendungstoken besteht:

```
{  
"applicationToken": "..."  
}
```

Stellen Sie sicher, dass das generierte Bearer-Token mit den entsprechenden Headern für alle weiteren Anfragen gesendet wird. Fügen Sie hierzu jeder Anfrage diesen HTTP-Request-Header wie folgt hinzu:

```
Autorisierung: Bearer <token from response>
```

Beachten Sie das Leerzeichen nach dem Begriff "Bearer".

AUTHENTIFIZIERUNG DURCH AUTORISIERUNGSCODE

Beachten Sie, dass Ihre Clientanwendung OAuth 2.0 mit dem Gewährungstyp **Autorisierungscode** unterstützen muss.

Konfigurieren Sie die zu verwendende Clientanwendung:

- **Callback-URL**

Die Callback-URL, zu der Sie umgeleitet werden, um sich bei Ihrem Projektraum in ARIS Process Mining zu authentifizieren:

```
https://<region.ariscloud>/umc/rest/oauth/callback?tenant=<project_room>&provider=umc
```

Den Hostnamen der URL (z. B. processmining.ariscloud.com) können Sie in der Adressleiste des Browsers ablesen, wenn Sie angemeldet sind.

Beispiele

Wenn sich Ihr Projektraum in der ARIS Cloud befindet, könnte die Callback-URL wie folgt lauten.

```
https://processmining.ariscloud.com/umc/rest/oauth/callback?tenant=<project_room>&provider=umc
```

Wenn sich Ihr Projektraum in der ARIS Enterprise Cloud befindet, könnte die URL wie folgt lauten.

```
https://<my_company_name>.ariscloud.com/umc/rest/oauth/callback?tenant=<project_room>&provider=umc
```

- **Mandanten-ID und Mandantengeheimnis**

Sie haben diese notiert, als Sie die Systemintegration in ARIS Process Mining erstellt haben, oder Sie können sie in der ARIS Process Mining-Administration aus der Liste im Modul **Systemintegration** abrufen, wenn Sie die Systemzugriffsdaten für diese Systemintegration aufrufen.

Mandanten-ID und -geheimnis müssen im Textkörper gesendet werden, nicht als Basic OAuth-Header.

- **Authentifizierungs-, Token- und Aktualisierungs-Endpunkte** können Sie ermitteln, indem Sie die entsprechende bekannte URL in Ihrem Browser aufrufen. Sie können die entsprechende bekannte URL aus der Liste des Moduls **Systemintegration** in der ARIS Process Mining-Administration entnehmen, wenn Sie die Systemzugriffsdaten für diese Systemintegration aufrufen. Die URL gibt ein JSON-Objekt mit `authorization_endpoint`, `token_endpoint`, `refresh_endpoint` und `userinfo_endpoint` zurück.

Beispiel

```
{
  "authorization_endpoint":
  "https://<hostname>/umc/oauthLogin?grant_type=authorization_code&tenant=<project_room>",
  "token_endpoint":
  "https://<hostname>/umc/api/v1/oauth/accesstoken?grant_type=authorization_code&tenant=<project_room>",
  "userinfo_endpoint":
  "https://<hostname>/umc/api/v1/oauth/userinfo?tenant=<project_room>",
  "refresh_endpoint":
  "https://<hostname>/umc/api/v1/oauth/refresh_token?tenant=<project_room>"
}
```

```
}
```

Sobald die Client-Anwendung eine Authentifizierungsanfrage mit diesen Eigenschaften sendet, antwortet der Server mit einem JSON-Objekt, das aus Mandant, URL und Zugriffstoken besteht.

```
{  
"tenant": "<project_room>",  
"token": "...",  
"url": "https://some_url"  
}
```

Hinweis

Verwenden Sie den Wert der URL als Hostnamen für alle nachfolgenden Aufrufe an die REST-Endpunkte.

Stellen Sie sicher, dass das generierte Bearer-Token mit den entsprechenden Headern für alle weiteren Anfragen gesendet wird. Fügen Sie hierzu jeder Anfrage diesen HTTP-Request-Header wie folgt hinzu:

```
Autorisierung: Bearer <token from response>
```

Beachten Sie das Leerzeichen nach dem Begriff "Bearer".

Zusätzlich muss für den Authentifizierungstyp **Autorisierungscode** mit jeder Anfrage ein CSRF-Token gesendet werden.

Sie erhalten ein CSRF-Token, indem Sie nach erfolgreicher Authentifizierung eine HTTP POST-Anfrage über den Pfad **/umc/api/v2/tokens/csrfToken** an die ARIS Benutzerverwaltung senden.

Das Ergebnis ist eine alphanumerische Zeichenfolge auf Basis Ihrer aktueller Benutzersitzung, z. B. oehltw0drUujSdWMD5TJEsXSLklwk1xKYh1LHaZ16g7. Sie müssen dieses Token mit dem Header csrfToken bei jeder nachfolgenden Anfrage senden.

1.3.1.1 Authentifizierung über URL-Parameter ist veraltet

Bitte beachten Sie, dass ab Service Release 28 (SR 28) das Übergeben der Authentifizierungs-Anmeldedaten über URL-Abfrage-Parameter nicht mehr unterstützt wird. Die Anmeldedaten müssen im Anfragetext übergeben werden. (Seite 5)

1.3.1.2 Technischen Schlüssel des Datensets notieren

Alle nachfolgenden Abschnitte zeigen unterschiedliche Anfragen, die im Zusammenhang eines bestimmten Datensets durchgeführt werden. Bei fast allen Anfragen muss der technische Schlüssel des Datensets verwendet werden. (Seite 16)

In einer Anfrage ist der technische Schlüssel wie folgt enthalten:

/dataSets/<Datenset>

Der Parameter **<Datenset>** referenziert den technischen Schlüssel des Datensets und verwendet diesen Wert zur Laufzeit. Der Wert kann beim Öffnen des entsprechenden Datensets der URL in der Adressleiste des Browsers entnommen werden.

Die URL hat folgendes Format:

`https://<hostname>/#<project_room>/dataCollection/y.dataset.<key>`

Der Parameter <key> basiert auf dem ausgewählten Anzeigenamen des Datensets und sollte lesbar sein. Verwenden Sie diesen Schlüssel in allen API-Anfragen für dieses Datenset.

Beispiel

`https://ariscloud.com/#myprojectroom/dataCollection/y.dataset.mydataset`

1.3.2 Quelltabellen-Definitionen abrufen

Führen Sie zum Abrufen von Quelltabellen-Definitionen die folgende HTTP-Anfrage aus.

GET "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/sourceTableDefinitions[?fullyQualifiedNames=default.table_a[,default.table_b]]"

Wenn keine fullyQualifiedNames angegeben sind, wird die Struktur aller verfügbaren Quelltabellen zurückgegeben.

1.3.3 Quelltabellen erstellen oder ersetzen

Führen Sie zum Erstellen oder Ersetzen von Quelltabellen die folgenden HTTP-Anfragen aus.

QUELLTABELLEN ERSTELLEN

POST "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/sourceTables"

Es ist erforderlich, einen Anfragetext an den Server in folgender Form (hier mit Beispieldaten) zu senden:

```
[
  {
    "name": "table_a",
    "namespace": "default",
    "columns": [
      {
        "dataType": "STRING",
        "name": "column_a1"
      },
      {
        "dataType": "LONG",
        "name": "column_a2"
      },
    ],
  },
]
```

```
{
  "dataType": "DOUBLE",
  "name": "column_a3"
},
{
  "dataType": "FORMATTED_TIMESTAMP",
  "name": "column_a4",
  "format": "yyyy-MM-dd HH:mm:ss.SSS"
}
],
{
  "name": "table_b",
  "namespace": "default",
  "persistenceMode": "OVERWRITE",
  "columns": [
    ...
  ]
}
]
```

QUELLTABELLEN ERSETZEN

POST "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/sourceTables?forceReplace=true"

Es ist erforderlich, einen Anfragetext an den Server in folgender Form (hier mit Beispieldaten) zu senden:

```
[
  {
    "fullyQualifiedName": "default.table_a",
    "columns": [
      {
        "dataType": "STRING",
        "name": "column_a1"
      },
      {
        "dataType": "LONG",
        "name": "column_a2"
      },
      {
        "dataType": "DOUBLE",
        "name": "column_a3"
      },
      {
        "dataType": "FORMATTED_TIMESTAMP",
        "name": "column_a4",
        "format": "yyyy-MM-dd HH:mm:ss.SSS"
      }
    ]
  },
  {
    "fullyQualifiedName": "default.table_b",
    "persistenceMode": "OVERWRITE"
  }
]
```

```
]
```

Wenn kein Persistenzmodus (Seite 31) (`persistenceMode`) festgelegt ist, wird die Tabelle erstellt oder durch "`persistenceMode = OVERWRITE`" ersetzt.

Wenn Sie den WebMethods-Konnektor für ARIS Process Mining verwenden, wird der Parameter `forceReplace` implizit festgelegt.

1.3.4 Prüfen, ob das Datenset zum Hochladen bereit ist

Das Datenset muss zum Hochladen der Daten bereit sein. Um den Status des Datensets zu prüfen, führen Sie die folgende HTTP-Anfrage aus.

```
POST "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/readyForIngestion"
```

Es ist erforderlich, einen Anfragetext an den Server in folgender Form (hier mit Beispieldaten) zu senden:

```
{
  "dataUploadTargets": [
    {
      "fullyQualifiedName": "default.table_a"
    },
    {
      "fullyQualifiedName": "default.table_b"
    }
  ]
}
```

Ist das Datenset bereit, ist die Antwort positiv. Ansonsten erhalten Sie eine negative Antwort mit dem entsprechenden Grund.

Beispiel

```
{
  "ready": false,
  "cause": {
    "code": "INR1001",
    "message": "The data set is currently being processed"
  }
}
```

1.3.5 Daten-Upload-Zyklus erstellen

Wenn das Datenset bereit ist, können Sie mithilfe der folgenden HTTP-Anfrage einen Dateningestionszyklus für den Daten-Upload erstellen.

POST

```
"https://<Hostname>/mining/api/pub/dataIngestion/v1/dataSets/<Datenset>/ingestionCycles"
```

Es ist erforderlich, einen Anfragetext an den Server in folgender Form (hier mit Beispieldaten) zu senden:

```
{
  "dataUploadTargets": [
    {
      "fullyQualifiedName": "default.table_a"
    },
    {
      "fullyQualifiedName": "default.table_b"
    }
  ]
}
```

Die Antwort für den obigen Aufruf gibt den vollständig gebildeten Dateningestionszyklus zurück, der einen technischen Schlüssel, die referenzierten Daten-Upload-Ziele (Tabellen) und einige Statusinformationen enthält. Sein Ausgangszustand ist **ACCEPTING_DATA**. Alle Tabellen, auf die vom Zyklus referenziert wird, werden für alle anderen Aktionen außer dem anstehenden Daten-Upload gesperrt.

```
{
  "key": "data_set_1_55",
  "dataUploadTargets": [...],
  "dataLoadTriggered": false,
  "state": {
    "value": "ACCEPTING_DATA"
  }
}
```

Notieren Sie sich den technischen Schlüssel des Zyklus und verwenden Sie ihn für alle nachfolgenden Anfragen, die im Kontext dieses Zyklus ausgeführt werden, z. B. beim Committen des Daten-Upload-Zyklus.

In einer Anfrage ist der technische Schlüssel wie folgt enthalten:

```
/ingestionCycles/<Ingestionszyklus>
```


1.3.6 Daten hochladen

Führen Sie die folgende HTTP-Anfrage aus, um die Daten einer Quelltable in das Datenset hochzuladen.

```
POST "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<data set>/sourceTables/<source table>/data"
```

<source table>: vollqualifizierter Name (default.table_a)

Es ist erforderlich, einen Anfragetext an den Server in folgender Form (hier mit Beispieldaten) zu senden:

```
[  
  ["Dies ist eine Beschreibung", 1255, 1385.5, "2021-07-15 18:03:25.889"],  
  ["Ein zweiter Beispieltext", 510, -23.58, "2021-07-10 10:59:05.421"],  
  ["Beispieltext", 1626347163123, 3.1415, "2021-07-01 08:00:01.002"]  
]
```

Größere Datenmengen können mithilfe von mehreren Anfragen hochgeladen werden. Mit jeder Anfrage werden die Daten temporär auf dem Server gespeichert.

1.3.7 Daten-Upload-Zyklus committen

Teilt ARIS Process Mining mit, dass alle Daten hochgeladen wurden und in die Quelltabellen geladen werden können.

```
PUT "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<data set>/ingestionCycles/<ingestion cycle>/dataComplete"
```

Der Status des Ingestionszyklus wechselt zu **INGESTING_DATA**. Die hochgeladenen temporären Daten werden nun in der Quelldatenbank fest gespeichert.

Der Status des angegebenen Zyklus wird auf "COMPLETED_SUCCESSFULLY" gesetzt, wenn der Upload in ARIS Process Mining ohne Probleme abgeschlossen wurde.

1.3.8 Zyklus-Status abrufen

Führen Sie die folgende HTTP-Anfrage aus, um den aktuellen Zyklus-Status zu lesen.

```
GET "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/ingestionCycles/<ingestion cycle>/state"
```

War die Dateningestion nicht erfolgreich, erhalten Sie eine Antwort mit dem entsprechenden Grund:

```
{
  "value": "FAILED",
  "cause": {
    "code": "IER1000",
    "message": "An unexpected error occurred"
  }
}
```

Ansonsten kann der Aufnahmestatus **INGESTING_DATA** sein, falls der Zyklus immer noch läuft, **COMPLETED_SUCCESSFULLY**, falls er ohne Probleme durchgelaufen ist, oder **CANCELED**, falls er abgebrochen wurde (z. B. über die API).

1.3.9 Prüfen, ob das Datenset zum Laden bereit ist

Das Datenset muss zum Laden der Daten in den Prozessspeicher bereit sein. Um den Status des Datensets zu prüfen, führen Sie die folgende HTTP-Anfrage aus.

```
POST "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/readyForIngestion"
```

Es ist erforderlich, einen Anfragetext an den Server in folgender Form zu senden:

```
{
  "dataLoadTriggered": true
}
```

Ist das Datenset bereit, ist die Antwort positiv. Ansonsten erhalten Sie eine negative Antwort mit dem entsprechenden Grund

1.3.10 Laden der Daten starten

Wenn das Datenset für das Laden der Daten bereit ist, erstellen Sie einen Dateningestionszyklus, um das Laden der Daten mithilfe der folgenden HTTP-Anfrage zu starten.

POST

```
"https://<Hostname>/mining/api/pub/dataIngestion/v1/dataSets/<Datenset>/ingestionCycles"
```

Es ist erforderlich, einen Anfragetext an den Server in folgender Form zu senden:

```
{  
  "dataLoadTriggered": true  
}
```

Die Antwort für die obige Anfrage gibt den vollständig formulierten Dateningestionszyklus zurück. Der Ausgangszustand ist **INGESTING_DATA**. Der entsprechende Datenladevorgang wird direkt gestartet.

Beachten Sie, dass der Parameter "dataLoadTriggered" implizit festgelegt wird, wenn Sie den WebMethods-Konnektor für ARIS Process Mining verwenden.

1.3.11 Zyklus-Status abrufen

Zum Überprüfen, ob das Laden der Daten erfolgreich war, führen Sie die folgende HTTP-Anfrage aus.

```
GET "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<dataset>/ingestionCycles/<ingestion cycle>/state"
```

Sie erhalten eine Antwort mit dem aktuellen Statuswert (**INGESTING_DATA**, **COMPLETED_SUCCESSFULLY** oder **FAILED**) und optional einen entsprechenden Grund.

Weitere Informationen finden Sie im Kapitel Zyklus-Status abrufen (Seite 13).

1.3.12 Quelltabelle einfügen

Fügt die angegebene Quelltabelle ein.

```
DELETE "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<data set>/sourceTables/<source table>"
```

1.3.13 Ingestionszyklen abrufen

Ruft alle bestehenden Ingestionszyklen ab.

```
GET "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<data set>/ingestionCycles"
```

1.3.14 Ingestionszyklus abbrechen

Bricht den angegebenen Zyklus ab.

```
PUT "https://<hostname>/mining/api/pub/dataIngestion/v1/dataSets/<data set>/ingestionCycles/<ingestionCycle>/canceled"
```

1.4 API-Methoden

Sie können die folgenden Endpunkte verwenden, die von der Data Ingestion API für Ihre HTTP-Anfragen bereitgestellt werden.

1.4.1 Pfadabschnitt: Datenset

Außer dem Endpunkt der API-Version werden alle Endpunkte im Kontext eines bestimmten Datensets verwendet. Alle Endpunkte mit einem bestimmten Datenset-Kontext enthalten den folgenden URL-Abschnitt:

```
/dataSets/{dataSet}
```

Der Parameter **{dataSet}** referenziert den technischen Schlüssel des Datensets und verwendet diesen Wert zur Laufzeit. Der Wert kann beim Öffnen des entsprechenden Datensets der URL in der Adressleiste des Browsers entnommen werden.

Die URL hat folgendes Format:

```
https://<hostname>/#<project_room>/dataCollection/y.dataset.<key>
```

Der Parameter <key> basiert auf dem ausgewählten Anzeigenamen des Datensets und sollte lesbar sein. Verwenden Sie diesen Schlüssel in allen API-Anfragen für dieses Datenset.

Beispiel

<https://ariscloud.com/#myprojectroom/dataCollection/y.dataset.mydataset>

1.4.2 Pfadabschnitt: Ingestionszyklus

Im Kontext eines bestimmten Ingestionszyklus werden mehrere Endpunkte verwendet, z. B. Daten-Uplad-Zyklus committen (Seite 22), Status des Ingestionszyklus zurückgeben (Seite 22) und Ingestionszyklus abbrechen (Seite 21). Aus diesem Grund enthalten alle von ihnen den folgenden Abschnitt:

/ingestionCycles/{ingestionCycle}

Der Parameter **{ingestionCycle}** referenziert den technischen Schlüssel des Ingestionszyklus und verwendet diesen Wert zur Laufzeit. Ein solcher Schlüssel wird immer dann erzeugt, wenn ein neuer Ingestionszyklus erstellt wird. Er wird als Wert der Schlüsseleigenschaft des Zyklus zurückgegeben, nachdem er erstellt wurde, oder wenn alle vorhandenen Ingestionszyklen mithilfe der API aus dem System ermittelt werden.

1.4.3 API-Version abrufen

Ruft die aktuelle API-Version von ARIS Process Mining ab.

GET /api/pub/dataIngestion/version

Ausgabe: Aktuelle API-Version

1.4.4 Quelltabellen-Definitionen abrufen

Ruft die Spaltenstruktur (Name, Datentyp, Format) der angegebenen Quelltabellen ab.

GET /api/pub/dataIngestion/v1/dataSets/{dataSet}/sourceTableDefinitions

Eingabe: Abfrageparameter **'fqns'** zum Filtern nach vollqualifizierten Namen

Ausgabe: Liste der **SourceTableDefinition**-Objekte

1.4.5 Quelltabellen erstellen oder ersetzen

Erstellt oder ersetzt eine Quelltable in ARIS Process Mining.

Dies ist abhängig vom Parameter **forceReplace**. Wenn der Parameter forceReplace "true" ist, werden die Tabellen, die den denselben Identifizierer haben, ersetzt und alle zuvor gespeicherten Daten gelöscht.

POST /api/pub/dataIngestion/v1/dataSets/{dataSet}/sourceTables

Eingabe:

- Liste der **SourceTableDefinition**-Objekte
 - Beim Erstellen einer Quelltable sind Name, Namespace und Spalten erforderlich. Die anderen Eigenschaften sind optional.
Beachten Sie, dass "_ARIS" als Namespace nicht zulässig ist.
 - Wenn eine Quelltable ersetzt wird,
 - muss der Identifizierer für die bestehende Tabelle als Schlüssel, vollqualifizierter Name oder Name und Namespace angegeben werden. Wenn Sie mehrere Identifizierer angeben, gilt die Prioritätenreihenfolge Schlüssel > vollqualifizierter Name > Name und Namespace. Identifizierer mit niedriger Priorität werden ignoriert, wenn ein Identifizierer mit höherer Priorität vorhanden ist.
 - Alle anderen Eigenschaften (außer Identifizierer) sind optional. Wenn eine Eigenschaft nicht festgelegt ist, greift sie auf die Werte der bestehenden Tabelle zurück. Beachten Sie, dass die Spalten einer Tabelle in derselben Eigenschaft festgelegt werden. Wenn Sie eine Spalte entfernen möchten, lassen Sie sie im Textkörper weg. Wenn Sie eine Spalte hinzufügen möchten, wiederholen Sie die Informationen der bestehenden Spalten und fügen Sie die neue Spalte ein.
Beachten Sie, dass "_ARIS" als Namespace nicht zulässig ist.
- Der Abfrage-Parameter **forceReplace** bestimmt, ob die bestehenden Quelltabellen durch denselben Identifizierer ersetzt werden sollen. Wenn die Tabellen ersetzt werden, werden alle zuvor gespeicherten Quelldaten gelöscht. Wenn der Parameter nicht auf "true" gesetzt ist, werden Anfragen, die eine Ersetzung erfordern würden (z. B. den Identifizierer einer bestehenden Tabelle einschließen) zurückgewiesen.

Ausgabe:

Liste von **SourceTableDefinition**-Objekten, die auf neu erstellten oder ersetzten Quelltabellen basieren

1.4.6 Quelltabellendefinition aktualisieren

Aktualisiert eine Quelltabellendefinition, aber lässt die bestehenden Daten unverändert. Kann verwendet werden, um

- den vollqualifizierten Namen, den Namespace und/oder den Namen zu ändern. Beachten Sie, dass "_ARIS" als Namespace nicht zulässig ist.
- den Tabellentyp "Normal" durch "Inkrementell" zu ersetzen.
- einen Merge-Schlüssel für eine inkrementelle Tabelle (neu) zu definieren.
- Fügen Sie einer inkrementellen Tabelle Spalten hinzu.
- Konfigurieren Sie das Schema der nachfolgenden Datenlieferungen so, dass es keine bestimmten Spalten mehr enthält. Die bestehenden Daten werden weiterhin die Spalte enthalten.

Spalten einer normalen Tabelle können mit dieser Methode nicht geändert werden.

Verwenden Sie stattdessen den Endpunkt Quelltabellen erstellen oder ersetzen (Seite 17).

POST /api/pub/dataIngestion/v1/dataSets/{dataSet}/sourceTables/{sourceTable}/definition

Eingabe:

SourceTableDefinition-Objekt

- Die Quelltable muss bereits vorhanden sein.
 - Die bestehenden Daten bleiben erhalten.
 - Deshalb sind nicht alle Änderungen an der Definition zulässig.
 - Die Quelltable in der URL kann entweder ein Schlüssel oder ein vollqualifizierter Name sein.
 - Alle Eigenschaften der Quelltabellendefinition sind optional. Sind die Eigenschaften nicht gesetzt, werden die Werte der bestehenden Tabelle wiederverwendet.

Beachten Sie Folgendes.

Die Spalten einer Tabelle werden im Ganzen gesetzt. Zum Entfernen einer Spalte können Sie diese also einfach im Textkörper weglassen. Wenn Sie eine Spalte hinzufügen möchten, wiederholen Sie die Informationen der bestehenden Spalten und fügen Sie die neue Spalte ein.

Wenn im Textkörper sowohl der vollqualifizierte Name als auch der Name und/oder Namespace angegeben werden, hat der vollqualifizierte Name Vorrang.

Ausgabe:

Liste von **SourceTableDefinition**-Objekten, die auf neu erstellten oder ersetzten Quelltabellen basieren

Beachten Sie, dass die Aktualisierung der Daten einer inkrementellen Tabelle Änderungen am Schema der Quelltable nach sich ziehen kann.

1.4.7 Prüfen, ob Datenset für die Ingestion bereit ist

Die Prüfung bestätigt dass das Hochladen oder das Laden der Daten starten kann.

Beachten Sie, dass Sie den Bereitschaftsstatus ENTWEDER für das Starten eines Datenladevorgangs ODER für das Hochladen von Daten prüfen können, nicht jedoch für beides zugleich!

POST /api/pub/dataIngestion/v1/dataSets/{dataSet}/readyForIngestion

Eingabe: DataIngestionCycle mit

- entweder SourceTableDefinitions basierend auf voll konfigurierten Datenquellen zum Aktualisieren. Sie können einen beliebigen Identifizierer angeben. Die anderen Eigenschaften sind optional und werden ignoriert.
- oder ein boolesches Flag, das angibt, dass der Datenladevorgang starten kann. Das Festlegen einer Liste von Quelltabellen-Definitionen und das Setzen des booleschen Flags auf "true" wird derzeit nicht unterstützt. Wenn Sie nach einem Datenladevorgang einen Upload ausführen möchten, müssen Sie beides getrennt als zwei Dateningestionszyklen durchführen.

Ausgabe: IngestionReadyState. Beachten Sie, dass die Bereitschaftsprüfung für einen Datenladevorgang nicht unbedingt alle vorhandenen Überprüfungsprobleme berücksichtigt.

Auch wenn die Benutzeroberfläche einige Validierungsprobleme anzeigt, kann die Bereitschaftsprüfung unter Umständen ein erfolgreiches Ergebnis anzeigen.

1.4.8 Neuen Ingestionszyklus erstellen

Erstellt einen neuen Ingestionszyklus.

Beachten Sie, dass der Ingestionszyklus ENTWEDER für das Starten eines Datenladevorgangs ODER für das Hochladen von Daten erstellt wird, nicht jedoch für beides zugleich!

POST /api/pub/dataIngestion/v1/dataSets/{dataSet}/ingestionCycles

Eingabe: DataIngestionCycle mit

- entweder SourceTableDefinitions basierend auf voll konfigurierten Datenquellen zum Aktualisieren. Sie können einen beliebigen Identifizierer angeben. Die anderen Eigenschaften sind optional und werden ignoriert.
- oder ein boolesches Flag, das angibt, dass der Datenladevorgang starten kann. Das Festlegen einer Liste von Quelltabellen-Definitionen und das Setzen des booleschen Flags auf "true" wird derzeit nicht unterstützt. Wenn Sie nach einem Datenladevorgang einen Upload ausführen möchten, müssen Sie beides getrennt als zwei Dateningestionszyklen durchführen.

Sie müssen die Eingabe im Antworttext angeben. Im Falle eines Datenladevorgangs müssen Sie im Antworttext "dataLoadTriggered" auf "True" setzen:


```
{  
  "dataLoadTriggered": true  
}
```

Ausgabe: New DataIngestionCycle

1.4.9 Ingestionszyklus abbrechen

Löscht einen bestehenden Ingestionszyklus (identifiziert durch einen Ingestionszyklusschlüssel).

PUT

/api/pub/dataIngestion/v1/dataSets/{dataSet}/ingestionCycles/{ingestionCycle}/canceled

Ausgabe: Canceled DataIngestionCycle

1.4.10 Daten hochladen

Lädt Daten für die angegebenen Quelltabellen nach ARIS Process Mining hoch und sortiert die Spalten automatisch anhand der Spaltenreihenfolge in ARIS Process Mining.

Die Daten müssen die richtige Struktur aufweisen (Anzahl und Reihenfolge der Spalten mit korrekten Datentypen und -formaten).

POST /api/pub/dataIngestion/v1/dataSets/{dataSet}/sourceTables/{sourceTable}/data

Eingabe:

- Quelltabellenidentifizierer als Pfad-Parameter. Der Identifizierer kann ein Schlüssel oder ein vollqualifizierter Name sein.
- Eine Liste von Objekten als Text, die die neuen Quelldateneinträge repräsentieren.
 - Die Reihenfolge der Spalten entspricht der Reihenfolge, die bei der Erstellung der Quelltable festgelegt und von der GET-Operation bei den sourceTableDefinitions zurückgegeben wird.
 - Die Zeitstempeldaten können nur als Zeichenfolgen weitergegeben werden, die im Datums- und Uhrzeitformat der entsprechenden Quelltabellenspalte formatiert sind.
 - Große Datensets können mithilfe von mehreren Anfragen hochgeladen werden. Die Daten der einzelnen Anfragen werden serverseitig in temporärer Form gespeichert.

Ausgabe: Erfolgsergebnis, wenn die Daten ohne Fehler empfangen wurden.

1.4.11 Quelltabelle einfügen

Fügt eine angegebene Quelltable (Definition und Inhalt) ein.

DELETE /api/pub/dataIngestion/v1/dataSets/{dataSet}/sourceTables/{sourceTable}

Eingabe: Quelltabellenidentifizierer als Pfad-Parameter. Der Identifizierer kann ein Schlüssel oder ein vollqualifizierter Name sein.

Ausgabe: Erfolgsergebnis, wenn die Löschung ohne Fehler ausgeführt wurde.

1.4.12 Daten-Upload-Zyklus committen

Benachrichtigt ARIS Process Mining darüber, dass der Daten-Upload abgeschlossen ist und die Ingestion in ARIS Process Mining beginnt.

Wenn der Upload in ARIS Process Mining ohne Probleme erfolgt ist, wird der Status des Ingestionszyklus auf "COMPLETED_SUCCESSFULLY" aktualisiert. Dies ist eine Voraussetzung für den Start eines neuen Ingestionszyklus zum Laden der Daten.

PUT

/api/pub/dataIngestion/v1/dataSets/{dataSet}/ingestionCycles/{ingestionCycle}/dataComplete

Ausgabe: Running DataIngestionCycle

1.4.13 Ingestionszyklen abrufen

Ruft alle bestehenden Ingestionszyklen für ein Datenset ab.

GET /api/pub/dataIngestion/v1/dataSets/{dataSet}/ingestionCycles

Ausgabe: Liste der DataIngestionCycle-Objekte

Der Endpunkt ist verfügbar ab ARIS Process Mining Version 10.18.

1.4.14 Status des Aufnahmezyklus zurückgeben

Ruft den Status des angegebenen Ingestionszyklus ab.

GET /api/pub/dataIngestion/v1/dataSets/{dataSet}/ingestionCycles/{ingestionCycle}/state

Ausgabe: Der Statuswert basiert auf dem Status des entsprechenden Eintrags im Ausführungsprotokoll.

1.5 Data Transfer Objects (DTOs)

Sie können die folgenden Data Transfer Objects (DTOs) für die Data Ingestion API verwenden.

1.5.1 SourceTableDefinition

Als Eingabe

Nur in Listenform, entweder einzeln wie hier abgebildet oder als Teil eines Datenaufnahmezyklus (DataIngestionCycle) (siehe unten). Eigenschaften können Pflicht- oder optionale Angaben sein, je nachdem, ob eine Tabelle erstellt oder ersetzt werden soll.

```
[
  {
    "key": "prq_some_namespace_e",
    "name": "example_table_o",
    "namespace": "some_namespace",
    "fullyQualifiedName": "some_namespace.example_table_o",
    "persistenceMode": "OVERWRITE|APPEND",
    "mergeKey": ["PROCESSOR_GROUP", "PROCESSOR"],
    "columns": [
      {
        "dataType": "DOUBLE",
        "name": "CATEGORY"
      },
      {
        "dataType": "STRING",
        "name": "CATEGORY_NAME"
      },
      {
        "dataType": "FORMATTED_TIMESTAMP",
        "name": "CREATED",
        "format": "yyyy/MM/dd HH:mm:ss"
      },
      {
        "dataType": "STRING",
        "name": "PROCESSOR"
      },
      {
        "dataType": "STRING",
        "name": "PROCESSOR_GROUP"
      }
    ]
  }
]
```

Als Ausgabe

Nur in Listenform, entweder einzeln wie hier abgebildet oder als Teil eines Datenaufnahmezyklus (DataIngestionCycle) (siehe unten).

```
[
```

```
{
  "key": "prq_some_namespace_e",
  "name": "example_table_o",
  "namespace": "some_namespace",
  "fullyQualifiedName": "some_namespace.example_table_o",
  "persistenceMode": "OVERWRITE",
  "mergeKey": ["PROCESSOR_GROUP", "PROCESSOR"],
  "columns": [
    {
      "dataType": "DOUBLE",
      "name": "CATEGORY"
    },
    {
      "dataType": "STRING",
      "name": "CATEGORY_NAME"
    },
    {
      "dataType": "FORMATTED_TIMESTAMP",
      "name": "CREATED",
      "format": "yyyy/MM/dd HH:mm:ss"
    },
    {
      "dataType": "STRING",
      "name": "PROCESSOR"
    },
    {
      "dataType": "STRING",
      "name": "PROCESSOR_GROUP"
    }
  ]
}
```

- Die Schlüssel werden auf dem Server generiert.
- Der voll qualifizierte Name (fullyQualifiedName) besteht aus dem Namen und dem Namespace, durch '.' getrennt.
- Der Persistenzmodus (persistenceMode) kann entweder OVERWRITE oder APPEND sein. Im Kapitel Persistenzmodus (Seite 31) erhalten Sie weitere Informationen.
- Spalten können vom Typ DOUBLE, LONG, STRING und FORMATTED_TIMESTAMP sein.
- Der Merge-Schlüssel (mergeKey) ist optional. Der Merge-Schlüssel wird nur zum Zusammenführen neuer Daten mit bestehenden Daten benötigt.

HINWEIS

Wenn eine Quelltabellendefinition erstellt oder aktualisiert wird, wird der hochgeladenen Tabelle immer automatisch eine Spalte **_ARIS_lastChanged** vom Typ **Zeitstempel** hinzugefügt. Das Präfix **_ARIS_** ist zur internen Nutzung vorbehalten. Sie dürfen keine Spalten festlegen, deren Namen mit **_ARIS_** beginnen. Anfragen zum Erstellen von Tabellen mit einer solchen Spalte schlagen fehl.

Wenn eine Quelltabellendefinition aus der API ermittelt wird, werden Spalten, deren Namen mit **_ARIS_** beginnen, übersprungen. Wenn Sie eine Quelltabellendefinition mit der API aktualisiert haben (anstelle diese zu ersetzen), gibt ein nachfolgendes **GET** nur die Spalten zurück, die von diesem Update konfiguriert wurden, und die "entfernten" Spalten werden ausgelassen. Diese Spalten und ihre enthaltenen Daten sind aber trotzdem noch vorhanden und werden weiterhin in ARIS Process Mining verarbeitet. Dies dient lediglich der Sicherstellung, dass ein API-Client immer die Tabelle in der aktuellsten Version (unter Verwendung des aktuellsten Schemas) erhält, und nur Spalten enthalten sind, die ihm bekannt sind.

1.5.2 DataIngestionReadyState

Diese werden nur als Ausgabe nach einem Bereitschaftscheck verwendet. Im Falle von 'nicht bereit' wird die Bereitschaftseigenschaft auf 'false' gesetzt und das Objekt enthält einen Grund mit Code und Nachricht.

```
{
  "ready": false,
  "cause": {
    "code": "INR1001",
    "message": "The data set is currently being processed"
  }
}
```

Gründe bestehen aus einem Code und einer Nachricht, die den Grund angibt, warum das Datenset nicht bereit ist. Der Code besteht aus 4 Ziffern und ihm ist immer "INR" für "Ingestion - Not Ready" vorangestellt. Sie finden eine Liste mit den konkreten Codes und den Bedeutungen in der folgenden Tabelle.

| Code | Bedeutung |
|---------|--|
| INR1000 | Nicht definiert. Für unerwartete Situationen verwendet. |
| INR1001 | Datenset in Bearbeitung. Das referenzierte Datenset ist von einem anderen Prozess gesperrt (Dateningestionszyklus, manuellem Daten-Load, etc.). Der Benutzer muss warten, bis die Sperre aufgehoben wird, und die Anfrage wiederholen. |

| Code | Bedeutung |
|---------|--|
| INR1002 | <p>Living Process-Kontingent überschritten.</p> <p>Die Verwendung der Data Ingestion API ist auf Datensets mit einer 'Living Process'-Lizenz beschränkt.</p> <p>Die zugeordnete 'Living Process'-Lizenz definiert eine obere Grenze (Prozesskontingent) für die Anzahl der Prozessinstanzen innerhalb des entsprechenden Datensets.</p> <p>Der Code INR1002 gibt an, dass das durch die zugeordnete 'Living Process'-Lizenz definierte Prozesskontingent bereits überschritten ist und keine weiteren Daten aufgenommen werden können. Der Benutzer kann die Anzahl der Prozessinstanzen im Datenset reduzieren, indem er Prozessinstanzen löscht oder eine 'Living Process'-Lizenz mit einem höheren Prozesskontingent zuweist.</p> |
| INR1003 | <p>Unerwarteter Quelltabellentyp.</p> <p>Gibt an, dass mindestens eine der referenzierten Quelltabellen einen unerwarteten Typ enthält, z. B. ein Typ, der über den CSV-Upload erstellt wurde.</p> <p>Zum Verwenden einer Tabelle im Kontext der Data Ingestion API muss diese entweder von dieser API erstellt oder von dieser über die Funktionalität zum Ersetzen der Quelltable aktualisiert worden sein.</p> |
| INR1004 | <p>Keine Daten zum Laden.</p> <p>Gibt an, dass es keine neuen anstehenden Daten für die Tabellen gibt, die im Abschnitt Datenmodellierung referenziert sind. Deshalb ist kein Laden der Daten erforderlich.</p> |

1.5.3 DataIngestionCycle

Als Eingabe

Im Falle eines Daten-Uploads

```
{
  "dataUploadTargets": [
    {
      "fullyQualifiedName": "some_namespace.example_table_a"
    }
  ]
}
```

Im Falle eines Daten-Loads

```
{
  "dataLoadTriggered": true
}
```

Als Ausgabe

Entweder in Listenform

```
[
  {
    "key": "api_2",
    "dataUploadTargets": [
      {
        "key": "prq_some_namespac_38",
        "name": "example_table_a",
        "namespace": "some_namespace",
        "fullyQualifiedName": "some_namespace.example_table_a",
        "persistenceMode": "APPEND",
        "columns": [
          {
            "dataType": "DOUBLE",
            "name": "CATEGORY"
          },
          {
            "dataType": "STRING",
            "name": "CATEGORY_NAME"
          },
          {
            "dataType": "FORMATTED_TIMESTAMP",
            "name": "CREATED",
            "format": "yyyy/MM/dd HH:mm:ss"
          },
          {
            "dataType": "STRING",
            "name": "PROCESSOR"
          }
        ]
      }
    ]
  }
]
```

```
        "dataType": "STRING",
        "name": "PROCESSOR_GROUP"
      }
    ]
  },
  "dataLoadTriggered": false,
  "state": {
    "value": "INGESTING_DATA"
  }
},
{
  "key": "api_1",
  "dataLoadTriggered": true,
  "state": {
    "value": "COMPLETED_SUCCESSFULLY"
  }
}
]
```

oder eigenständig, z. B. nach Erstellung, Aktualisierung oder Abbruch.

Im Falle eines Daten-Uploads

```
{
  "key": "api_1",
  "dataUploadTargets": [
    {
      "key": "prq_some_namespac_38",
      "name": "example_table_a",
      "namespace": "some_namespace",
      "fullyQualifiedName": "some_namespace.example_table_a",
      "persistenceMode": "APPEND",
      "columns": [
        {
          "dataType": "DOUBLE",
          "name": "CATEGORY"
        },
        {
          "dataType": "STRING",
          "name": "CATEGORY_NAME"
        },
        {
          "dataType": "FORMATTED_TIMESTAMP",
          "name": "CREATED",
          "format": "yyyy/MM/dd HH:mm:ss"
        },
        {
          "dataType": "STRING",
          "name": "PROCESSOR"
        },
        {
          "dataType": "STRING",
          "name": "PROCESSOR_GROUP"
        }
      ]
    }
  ]
}
```



```
    }
  ],
  "dataLoadTriggered": false,
  "state": {
    "value": "INGESTING_DATA"
  }
}
```

Im Falle eines Daten-Loads

```
{
  "key": "api_1",
  "dataLoadTriggered": true,
  "state": {
    "value": "INGESTING_DATA"
  }
}
```

1.5.4 DataIngestionCycleState

Nur als Ausgabe, einzeln oder als Teil eines Dateneingestionszyklus (`DataIngestionCycle`) (siehe oben). Mögliche Zustände sind: `ACCEPTING_DATA`, `INGESTING_DATA`, `COMPLETED_SUCCESSFULLY`, `CANCELED` und `FAILED`.

Im Falle von 'FAILED' wird ein Grund zurückgegeben. Gründe bestehen aus einem Code und einer Nachricht, die beschreibt, was genau passiert ist. Der Code besteht aus 4 Ziffern und ihm ist immer "IER" für "Ingestion - Error" vorangestellt.

```
{
  "value": "FAILED",
  "cause": {
    "code": "IER1000",
    "message": "An unexpected error occurred"
  }
}
```

TABLEDATA

Wird nur als Eingabe für den Daten-Upload verwendet. Die Werte müssen dem Schema der Zieltabelle entsprechen. Null ist ein gültiger Wert.

```
[
  [1, "A", "2021/05/10 12:13:14", 1.1, "Distribution Center Team", "Distribution"],
  [2, "B", "2021/06/11 15:16:17", 2.2, "Distribution Center Team", "Distribution"],
  [3, "C", "2021/07/12 18:19:20", 3.3, null, "Sales"],
  [4, "D", "2021/08/13 21:22:23", 4.4, "Dealer Sales", "Sales"]
]
```

STRINGCOLUMN

Wird nur als Teil einer Quelltabellendefinition verwendet (SourceTableDefinition) (siehe oben).

LONGCOLUMN

Wird nur als Teil einer Quelltabellendefinition verwendet (SourceTableDefinition) (siehe oben).

DOUBLECOLUMN

Wird nur als Teil einer Quelltabellendefinition verwendet (SourceTableDefinition) (siehe oben).

FORMATTEDTIMESTAMPCOLUMN

Wird nur als Teil einer Quelltabellendefinition verwendet (SourceTableDefinition) (siehe oben).

DEFAULTRESULT

Wird nur als Einzelausgabe verwendet, entweder wenn die ausgeführte Operation selbst kein zugeordnetes Ergebnisobjekt hat (Löschen einer Quelltable, Hochladen von Quelldaten) oder wenn ein Fehler auf dem Server auftritt (beliebige Operation). Die 'Erfolgreich'-Eigenschaft dieses Objekts ist entweder auf True oder auf False gesetzt.

Wenn False, enthält das Objekt auch einen Grund mit einer Nachricht.

```
{
  "successful": false,
  "cause": {
    "message": "An unexpected error occurred"
  }
}
```

APIVERSION

Diese werden nur als Ausgabe nach einem API-Versionscheck verwendet.

```
{
  "apiVersion": "3,2"
}
```

1.5.5 Authentifizierungsantwort

Die Antwort Ihrer Authentifizierungsanfrage an die ARIS Cloud ist ein JSON-Objekt, das den Mandanten, eine URL und ein Zugriffstoken umfasst.

```
{
  "tenant": "<project_room>",
  "token": "<access_token>",
  "url": "<any_URL>"
}
```

Beispiel

```
{
  "tenant": "myProjectRoom",
  "token": "...eyJpYXQiOjE2NjE5MzY3NzgsImp0aSI6IjBqLWg2TkZqc3RLb0pTZ1U1dXJUYmRXcUs3NGplRV9EZzRyeXhOeDN5dkxkakJsRFI2Z2NzUEJueGpRTmNHTXU0cFo2R2loazMwQ0NMOUR4d0lQdiIsInN1YiI6ImR...",
  "url": "https://processmining.ariscloud.com"
}
```

Die Antwort Ihrer Authentifizierungsanfrage an die ARIS Enterprise Cloud ist ein JSON-Objekt, das ein Anwendungstoken einschließt:

```
{
  'applicationToken': '...'
}
```

1.6 Wertvolle Informationen

1.6.1 Persistenzmodus

Alle Quelltabellen haben einen Persistenzmodus, der bestimmt, wie die neuen Daten auf dem Server verarbeitet werden.

Es gibt drei unterschiedliche Persistenzmodi:

OVERWRITE

Dies ist die Standardeinstellung, die dem Standardverhalten aus vorherigen Versionen folgt. Ist dieser Modus eingestellt, werden bestehende Daten durch die neuen hochgeladenen Daten überschrieben. Die überschriebenen Daten gehen verloren und können nicht wiederhergestellt werden. Bei Bedarf müssen Sie die überschriebenen Daten erneut hochladen.

APPEND (OHNE MERGE-SCHLÜSSEL)

Wenn dieser Modus eingestellt ist, werden an die bestehenden Daten neue hochgeladene Daten angehängt, anstatt die bereits fest gespeicherten Tabellendaten auf dem Server zu überschreiben. Die neuen Datenzeilen werden in der eingehenden Reihenfolge am Ende hinzugefügt. Dies bewirkt ein Anwachsen der bestehenden Quelltabellengröße. Werden bestehende alte Daten erneut mit dieser Einstellung gespeichert (als wären es neue Zeilen), werden doppelte Einträge angelegt. Dies kann sich auf die Richtigkeit der Analyseergebnisse auswirken.

Beachten Sie, dass die derzeit einzige Möglichkeit, diesen Modus für eine Quelltablelle auszuwählen, die Verwendung der Data Ingestion API ist. Mit der API können Sie entweder eine neue Tabelle erstellen oder die bestehende ersetzen.

APPEND MIT MERGE-SCHLÜSSEL

Wenn dieser Modus eingestellt ist, werden die hochgeladenen Daten mit bereits fest gespeicherten Tabellendaten auf dem Server zusammengeführt. Im Merge-Modus werden neue Daten an die Tabelle in zusätzlichen Zeilen angehängt, und bestehende Tabellenzeilen werden nur dann einzeln überschrieben, wenn die entsprechende Zeile der hochgeladenen Tabelle neuer ist. Am Ende des Daten-Uploads enthält die Quelltablelle sämtliche Daten.

Die Dateningestions-API verwendet einen Merge-Schlüssel zum Zusammenführen der bestehenden Daten und der neuen Daten in der Quelltablelle. Sie können den Merge-Modus verwenden, indem Sie den Merge-Schlüssel in der Quelltablelledefinition festlegen (Seite 23). Darüber hinaus müssen Sie den Persistenzmodus (`persistenceMode`) in der Quelltablelledefinition auf **APPEND** setzen.

Um einen Merge-Schlüssel für eine bestehende Tabelle zu konfigurieren, müssen Sie eine Quelltablelledefinition mit dem neuen Merge-Schlüssel an den Server senden. Hierzu können Sie den Endpunkt Quelltablellen erstellen oder ersetzen (Seite 17) verwenden. Der Merge-Schlüssel wird einer Quelltablelle automatisch hinzugefügt, wenn die Quelltablelle erstellt oder ersetzt wird (Seite 17).

HINWEIS

Eine Quelltablelle mit dem Persistenzmodus `OVERWRITE` entspricht einer Standardtablelle in ARIS Process Mining und einer Quelltablelle mit dem Konsistenzmodus `APPEND`, wobei ein festgelegter Merge-Schlüssel einer inkrementellen Tablelle entspricht.

1.6.2 Grenzwerte

GRÖÖE DER ANFRAGE

Die maximal zulässige Größe einer Anfrage zur Erstellung oder Aktualisierung von Daten beträgt 100 MB. Bei Überschreitung dieser Obergrenze wird die Anfrage abgelehnt. Wenn Sie mehr Daten erstellen oder aktualisieren möchten, teilen Sie die Daten auf mehrere Anfragen auf.

QUELLTABELLEN

GESAMTANZAHL DER QUELLTABELLEN

Mit der Data Ingestion API können maximal 100 Quelltablellen erstellt werden. Immer wenn mit der API neue Tablellen erstellt werden, wird überprüft, ob die maximale Anzahl überschritten wurde. Bei Überschreitung dieser Zahl wird die entsprechende Anfrage abgelehnt. Alle vorhandenen Tablellen werden auf die zulässige Höchstzahl angerechnet, unabhängig von ihrem Ursprung (API, Extraktion, manueller Datei-Upload). Diese Obergrenze wirkt sich nicht auf die Ersetzung der Quelltablelle aus.

ANZAHL DER QUELLTABELLEN PRO ANFRAGE

Mit einer Anfrage können maximal 50 Quelltabellen erstellt werden. Bei Überschreitung dieser Zahl wird die entsprechende Anfrage abgelehnt.

ANZAHL DER SPALTEN

Mit der Data Ingestion API können maximal 500 Spalten für eine Quelltable erstellt werden. Bei Überschreitung dieser Zahl wird die entsprechende Anfrage abgelehnt. Diese Obergrenze wirkt sich sowohl auf die Erstellung als auch auf die Ersetzung der Quelltable aus.

GESAMTANZAHL DER AUFGABEN

Die maximale Anzahl der Aufgaben (einschließlich Ingestionszyklen), die gleichzeitig gepflegt werden können, beträgt 350. Immer wenn mit der API ein neuer Zyklus erstellt wird, wird überprüft, ob die maximale Anzahl überschritten wurde. Bei Überschreitung dieser Zahl wird die entsprechende Anfrage abgelehnt. Alle vorhandenen Aufgaben, die noch gepflegt werden, werden auf die zulässige Höchstzahl angerechnet, unabhängig von ihrem Typ (Ingestionszyklus, Extraktion, manueller Datei-Upload, Datenladevorgang, Neuberechnung, Prozessdatenlöschung) oder ihrem Ursprung (API, Automatisierung, manuelle Ausführung). Gepflegte Aufgaben werden in regelmäßigen Intervallen von 30 Minuten automatisch bereinigt. Die Bereinigungsroutine löscht alle abgeschlossenen Aufgaben bis auf die 250 neuesten Einträge.

UPLOADS

ANZAHL DER DATEN-UPLOAD-ZIELE

Die maximale Anzahl der Quelltabellen (Daten-Upload-Ziele), die von einem Upload-Zyklus der Data Ingestion API referenziert werden können, ist 100. Bei Überschreitung dieses Maximums wird die entsprechende Anfrage abgelehnt. Wenn mehrere Quelltabellen hochzuladen sind, müssen Sie auf mehrere Upload-Zyklen aufgeteilt werden.

ANZAHL DER AUSSTEHENDEN DATENPAKETE

Beim Hochladen von Daten mit der Data Ingestion API sind pro Tabelle maximal 50 ausstehende Upload-Datenpakete möglich. Bei Überschreitung dieser Zahl wird die entsprechende Anfrage abgelehnt. Wenn mehr Daten an die Zieltable hochgeladen werden sollen, setzen Sie den Ingestionszyklus, der die Daten enthält, auf "abgeschlossen". Dadurch wird die serverseitige Persistenz gestartet. Sobald die Persistenz (und der Ingestionszyklus) abgeschlossen ist, kann ein neuer Zyklus zum Hochladen der übrigen Daten erstellt werden. Beachten Sie, dass der zweite Upload-Zyklus nur dann sofort gestartet werden sollte, wenn der Persistenzmodus der Zieltable auf APPEND gesetzt ist. Wenn der Modus auf OVERWRITE gesetzt ist, müssen Sie zuerst die Daten laden (Ladezyklus). Erst nach dem Laden der Daten können die übrigen Daten sicher hochgeladen werden.

1.7 Konnektor webMethods.io für ARIS Process Mining

Der Konnektor webMethods.io für ARIS Process Mining verwendet die Data Ingestion API zum Übertragen von Daten aus einer beliebigen Datenquelle nach ARIS Process Mining. Mit dem Konnektor webMethods.io für ARIS Process Mining können Sie z. B. eine Tabelle erstellen, Daten an die erstellte Tabelle hochladen und eine Datenladeoperation in ARIS Process Mining auslösen.

Mit vordefinierten Operationen können Sie die gängigsten REST-Ressourcen und -Operationen direkt nutzen oder die Komplexität der individuellen Anpassung von REST-Operationen reduzieren.

Weitere Informationen zur Verwendung des Konnektors webMethods.io für ARIS Process Mining entnehmen Sie bitte der webMethods.io-Dokumentation.

Die folgende Liste enthält alle vordefinierten Operationen, die vom Konnektor webMethods.io für ARIS Process Mining bereitgestellt werden, und gibt an, auf welche Endpunkte (Seite 16) der Dateningestion sie sich beziehen.

| Operation | Referenz zu Endpunkten |
|---------------------------------------|---|
| API-Version abrufen | API-Version abrufen (Seite 17) |
| Quelltabellen-Definitionen abrufen | Quelltabellen-Definitionen abrufen (Seite 17) |
| Quelltabellen erstellen | Quelltabellen erstellen oder ersetzen (Seite 17) |
| Quelltabellen ersetzen | Quelltabellen erstellen oder ersetzen (Seite 17) |
| Quelltabellendefinition aktualisieren | Quelltabellendefinition aktualisieren (Seite 19) |
| Bereit zum Daten-Upload | Prüfen, ob Datenset für die Ingestion bereit ist (Seite 20) |
| Bereit zum Datenladevorgang | Prüfen, ob Datenset für die Ingestion bereit ist (Seite 20) |
| Daten-Upload-Zyklus erstellen | Neuen Ingestionszyklus erstellen (Seite 20) |
| Laden der Daten starten | Neuen Ingestionszyklus erstellen (Seite 20) |
| Zyklus-Status abrufen | Ingestionszyklen abrufen (Seite 22) |
| Zyklus abbrechen | Ingestionszyklus abbrechen (Seite 21) |
| Daten-Upload-Zyklus committen | Daten-Upload-Zyklus committen (Seite 22) |
| Quelltabelle einfügen | Quelltabelle einfügen (Seite 21) |
| Zyklen abrufen | Ingestionszyklen abrufen (Seite 22) |

| Operation | Referenz zu Endpunkten |
|-----------------|---|
| Daten hochladen | Quelltabellen-Definitionen abrufen (Seite 17) Daten hochladen (Seite 21) |

2 Support und rechtliche Hinweise

Dieser Abschnitt bietet allgemeine Informationen im Hinblick auf den Produkt-Support und die rechtlichen Aspekte.

2.1 Dokumentationsumfang

Die zur Verfügung gestellten Informationen beschreiben die Einstellungen und Funktionalitäten, die zum Zeitpunkt der Veröffentlichung gültig waren. Da Software und Dokumentation verschiedenen Fertigungszyklen unterliegen, kann die Beschreibung von Einstellungen und Funktionalitäten von den tatsächlichen Gegebenheiten abweichen. Informationen über solche Abweichungen finden Sie in den mitgelieferten Release Notes. Bitte lesen und berücksichtigen Sie diese Datei bei Installation, Einrichtung und Verwendung des Produkts.

Wenn Sie das System technisch und/oder fachlich ohne die von Software GmbH angebotenen Service-Leistungen installieren möchten, benötigen Sie umfangreiche Kenntnisse hinsichtlich des zu installierenden Systems, der Zielthematik sowie der Zielsysteme und ihren Abhängigkeiten untereinander. Aufgrund der Vielzahl von Plattformen und sich gegenseitig beeinflussender Hardware- und Softwarekonfigurationen können nur spezifische Installationen beschrieben werden. Es ist nicht möglich, sämtliche Einstellungen und Abhängigkeiten zu dokumentieren.

Beachten Sie bitte gerade bei der Kombination verschiedener Technologien die Hinweise der jeweiligen Hersteller, insbesondere auch aktuelle Verlautbarungen auf deren Internet-Seiten bezüglich Freigaben. Für die Installation und einwandfreie Funktion freigegebener Fremdsysteme können wir keine Gewähr übernehmen und leisten daher keinen Support. Richten Sie sich grundsätzlich nach den Angaben der Installationsanleitungen und Handbücher der jeweiligen Hersteller. Bei Problemen wenden Sie sich bitte an die jeweilige Herstellerfirma.

Falls Sie bei der Installation von Fremdsystemen Hilfe benötigen, wenden Sie sich an Ihre lokale Software GmbH-Vertriebsorganisation. Beachten Sie bitte, dass solche Hersteller- oder kundenspezifischen Anpassungen nicht dem Standard-Softwarepflege- und Wartungsvertrag der Software GmbH unterliegen und nur nach gesonderter Anfrage und Abstimmung erfolgen.

2.2 Datenschutz

Die Funktionen der Produkte von Software GmbH werden im Hinblick auf die Verarbeitung personenbezogener Daten entsprechend der allgemeinen EU-Datenschutzverordnung (DSGVO) bereitgestellt.

In der jeweiligen administrativen Dokumentation sind gegebenenfalls entsprechende Schritte dokumentiert.

2.3 Support

Wenn Sie Fragen zu spezifischen Installationen haben, die sie nicht selbst durchführen können, wenden Sie sich bitte an Ihre örtliche Software GmbH-Vertriebsorganisation (<https://www.softwareag.com/corporate/company/global/offices/default.html>). Detaillierte Informationen und Support erhalten Sie über unsere Internetseiten.

Wenn Sie einen gültigen Supportvertrag haben, können Sie **Global Support ARIS** kontaktieren unter: **+800 ARISHELP**. Sollte diese Nummer von Ihrem Telefonanbieter nicht unterstützt werden, sehen Sie bitte in unserem Global Support Contact Directory nach.

Bei Fragen zur Produktdokumentation können Sie auch eine E-Mail an documentation@softwareag.com (<mailto:documentation@softwareag.com>) senden.

ARIS COMMUNITY

- Produkte, Updates und Fixes herunterladen
- Hier finden Sie Informationen, Fachartikel, Problemlösungen, Videos und Kommunikation mit anderen ARIS-Benutzern

Wenn Sie noch kein Konto haben, registrieren Sie sich bei der ARIS Community.

PRODUKTSCHULUNG

Hilfreiches Schulungsmaterial zu Ihrem Produkt finden Sie in unserem Learning Portal.

TECH COMMUNITY

Auf unserer Tech Community-Website können Sie mit Experten der Software GmbH zusammenarbeiten. Von hier aus können Sie z. B. Folgendes tun:

- Unsere riesige Wissensdatenbank durchsuchen.
- In unseren Diskussionsforen Fragen stellen und Antworten finden.
- Aktuelle News und Ankündigungen der Software GmbH erhalten.
- Unsere Communitys kennenlernen.

- In unseren öffentlichen Repositorys GitHub und Docker können Sie zusätzliche Software GmbH-Ressourcen entdecken.

PRODUKT-SUPPORT

Lizenzierte Kunden erhalten Support für Software GmbH-Produkte über unser Empower Portal (<https://empower.softwareag.com/>). Viele Services in diesem Portal setzen ein Konto voraus. Wenn Sie noch kein Konto haben, können Sie eines anfordern. Ein Konto bietet Ihnen z. B. folgende Möglichkeiten:

- Anfragen zu Produktmerkmalen hinzufügen
- Das Knowledge Center nach technischen Informationen und Tipps durchsuchen
- Frühwarnungen und kritische Alarmmeldungen erhalten
- Support-Vorfälle eröffnen und aktualisieren