



Schweizerische Eidgenossenschaft  
Confédération suisse  
Confederazione Svizzera  
Confederaziun svizra

Eidgenössisches Departement des Innern EDI

**Schweizerisches Bundesarchiv BAR**

Abteilung Informationsüberlieferung

Dienst Sicherung und Archivierungslösungen

# SIARD Suite Handbuch

Version 0.1.0-SNAPSHOT

# Inhaltsverzeichnis

1. Einleitung .....	1
2. Kontext .....	1
3. Über dieses Dokument .....	1
4. Systemanforderungen und Installation .....	1
4.1. Installation und Ausführen der Applikation .....	2
5. SIARD Suite verwenden .....	2
5.1. SIARD Suite GUI .....	2
5.2. SIARD Suite CLI .....	5
5.3. SiardFromDb .....	5
5.4. SiardToDb .....	7
6. Tipps und Tricks .....	9
6.1. Vorbereitung einer Datenbank für den Download .....	9
6.2. Vorbereitung einer Datenbank für den Upload .....	9
6.3. Zeit, Zeitstempel und Zeitzonen .....	10
7. Einschränkungen .....	10

# 1. Einleitung

Der Standard SIARD (Software-Independent Archival of Relational Databases) definiert ein Format, um Inhalte relationaler Datenbanken langfristig zu archivieren. Die Anwendung "SIARD Suite" bietet eine grafische Benutzeroberfläche zum Archivieren, Wiederherstellen, Durchsuchen und Exportieren von SIARD-Archiven.

## 2. Kontext

In Zusammenarbeit mit dem EU-Projekt E-ARK und der KOST wurde zwischen 2015 und 2018 die Version 2.1 des SIARD-Formats spezifiziert. Wie schon zuvor die Version 1.0 wurde auch SIARD-Format 2.1 als Standard eCH-0165 vom Verein "eCH E-Government Standards" homologiert. Im Jahr 2021 hat der DILCIS-Vorstand das erweiterte SIARD-Format 2.2 entwickelt und genehmigt.

SIARD Suite ist die Referenzimplementation für das Archivieren relationaler Datenbanken im standardisierten SIARD-Format 2.2. Weiter kann SIARD Suite auch SIARD-Archive in den Formaten SIARD 2.1 und SIARD 1.0 einlesen und zu SIARD 2.2 konvertieren.

## 3. Über dieses Dokument

Dieses Dokument gibt einen kurzen Überblick über die Anwendung und ihre Benutzung. Ein ausführliches Handbuch in englischer Sprache steht in dieser Applikation sowie auf der Projektwebsite des Schweizerischen Bundesarchivs zur Verfügung: <https://www.bar.admin.ch/bar/de/home/archivierung/tools---hilfsmittel/siard-suite.html>

## 4. Systemanforderungen und Installation

Die Applikation SIARD Suite ist eine Java FX Applikation. Als solche kann sie grundsätzlich auf jedem System ausgeführt werden, auf welchem ein Java Runtime Environment 8 (JRE) mit der JavaFX Extension installiert ist.

Zusätzlich stehen Distributionen mit inkludierter JRE sowie native Installer für Windows, Mac und Linux zur Verfügung, so dass keine JRE auf dem System installiert sein muss.

In der Folge sind die einzelnen Distributionen aufgeführt. Alle Distributionen können hier heruntergeladen werden: <https://github.com/sfa-siard/siard-suite/releases>

Distributionsname	Beschreibung
SIARD-Suite-v2.2.133	Distribution ohne Runtime, erfordert installierte JRE mit JavaFX auf dem System
SIARD-Suite-v2.2.133-Linux	Distribution mit inkludierter Runtime für Linux
SIARD-Suite-v2.2.133-Windows	Distribution mit inkludierter Runtime für Windows

Distributionsname	Beschreibung
SIARD-Suite-v2.2.133-macOS	Distribution mit inkludierter Runtime für MacOS
SIARD-Suite-v2.2.133-Linux-deb	Nativer Installer für Linux
SIARD-Suite-v2.2.133-win-exe	Nativer Installer für Windows (exe)
SIARD-Suite-v2.2.133-win-msi	Nativer Installer für Windows (msi)
SIARD-Suite-v2.2.133-macOS-dmg	

**Hinweis:** Wird der native Installer verwendet, steht das Siard CLI (Command Line Interface) nicht zur Verfügung. Um das CLI verwenden zu können, sollte ein Distribution mit oder ohne inkludierter JRE verwendet werden.

## 4.1. Installation und Ausführen der Applikation

Werden die nativen Installer verwendet, erfolgt die Installation einfach über das System. Allenfalls werden dazu Administratorrechte verwendet. Die Installer erstellen auch die entsprechenden Verknüpfungen zum Starten der Applikation.

Die Distributionen mit und ohne inkludierter JRE werden als ZIP Archiv ausgeliefert. Dieses kann an einem beliebigen Ort entpackt werden. Anschliessend kann die Applikation über eines der Startskripte gestartet werden:

Skriptname	Beschreibung
siard-suite (.bat)	Starten der grafischen Benutzeroberfläche für Unix bzw. Windows Systeme
siard-from-db (.bat)	Archivieren einer Datenbank (CLI)
siard-to-db (.bat)	Upload eines SIARD Archives in eine Datenbank (CLI)

## 5. SIARD Suite verwenden

Die SIARD Suite besteht aus folgenden Komponenten:

- Grafische Benutzeroberfläche (GUI)
- SiardFromDb (CLI)
- SiardToDB (CLI)

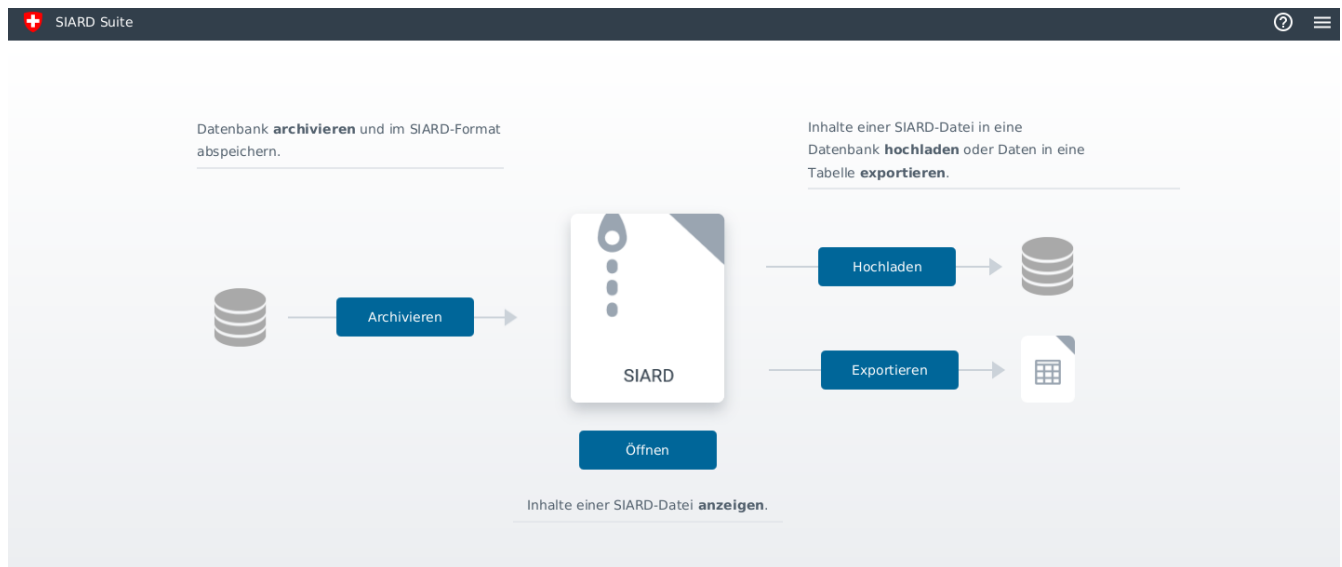
Zweck und Verwendung der Komponenten wird in der Folge erklärt.

### 5.1. SIARD Suite GUI

Die grafische Benutzeroberfläche ist die zentrale Anwendung um SIARD-Archive zu erstellen und später auch wieder zu sichten, ohne dass die Primärdaten des Archivs geändert werden können.

Für komplexe Analysen der archivierten Daten sollte das Archiv wieder in eine Datenbank hochgeladen werden.

Der Startscreen bietet den Einstiegspunkt in die Applikation. Hier können die vier wesentlichen Funktionalitäten der Applikation initiiert werden:




- Datenbank archivieren
- SIARD Archiv öffnen
- SIARD Archiv in eine Datenbank hochladen
- Daten eines SIARD Archives exportieren

Die Applikation führt den Benutzer durch die jeweiligen Prozesse.

### 5.1.1. Archivieren von Datenbanken mit Binary Data

Der Standard SIARD Format 2.2 hält fest, dass grosse Objekte (LOBs) einer Datenbank im externen Filesystem gespeichert werden können.

Der Speicherort der Daten kann während der Archivierung im Schritt "Metadaten" erfasst werden.

 SIARD Suite
 ☰ ?

✓ Datenbank-Management-System
✓ Verbindung
✓ Vorschau
4 Metadaten
5 Download

### Metadaten

Metadaten für Ihr Archiv bearbeiten

#### Was wird archiviert?

Datenbankname

Datenbankbeschreibung (optional)

Zustellungsbüro: Datendatenbankbesitzer vor der Archivierung

Ursprungszeit

#### Wer hat archiviert?

Ihr Name (optional)

Ihre Kontaktdaten (z. B. E-Mail, Telefonnummer) ⓘ

Ordner zum Exportieren großer Objekte (optional)

☐ Views als Tabellen exportieren

Zurück
Abbrechen
Weiter
Nur Metadaten exportieren

Der Pfad kann absolut als File-URI (**file:/**) oder relativ angegeben werden, z.b. **../lobs/** - auf jeden Fall muss der Pfad mit einem Schrägstrich (/) abgeschlossen werden. Es wird empfohlen relative Pfade anzugeben, so dass das Archiv mitsamt der externen LOB Daten verschoben werden kann - dies ist nicht möglich wenn absolute Pfade verwendet werden. Die Struktur innerhalb des spezifizierten Ordners wird anschliessend automatisch generiert und folgt der Struktur der Datenbank: **schema/table/column**. SIARD Suite detektiert automatisch den MIME-Type und die Dateiendungen werden automatisch korrekt gesetzt. Dies ermöglicht auch, dass die Binary-Daten direkt aus der SIARD Suite in der korrekten auf dem System installierten Applikation geöffnet werden können.

### 5.1.2. Einstellungen

Unter dem Menüpunkt "Optionen" kann die Anwendung konfiguriert werden.

## Optionen

X

Login-Timeout (s)

Abfrage-Timeout (s)

Abbrechen

Speichern

Folgende Einstellungen können vorgenommen werden:

Option	Beschreibung
Login Timeout (s)	Maximale Wartezeit auf eine Datenbankverbindung (in Sekunden). 0 für unbegrenzt
Query Timeout (s)	Maximale Wartezeit für eine Datenbankanfrage (0 für unbegrenzt).

## 5.2. SIARD Suite CLI

Neben der grafischen Benutzeroberfläche können Datenbanken auch über die Kommandozeile archiviert bzw. ein Archiv wieder in eine Datenbank eingelesen werden.

Dazu stehen in der ZIP-Distribution die Scripte `siard-from-db` und `siard-to-db` zur Verfügung.

## 5.3. SiardFromDb

Mit `siard-from-db` kann eine Datenbank ohne Verwendung des GUIs archiviert werden.

### 5.3.1. Aufruf

Wechseln Sie in einem Terminal in das Verzeichnis, in dem das ZIP-Archiv entpackt wurde, öffnen Sie dort das Verzeichnis `bin` und führen Sie es aus mit

```
siard-from-db [-h] | [-o] [-v] [-l=<login timeout>] [-q=<query timeout>] [-i=<import meta data>] [-x=<external lob folder>] [-m=<mime type>] -j=<JDBC URL> -u=<database user> -p=<database password> -s=<siard file> -e=<export meta data>"
```

### 5.3.2. Argumente

Argument	Beschreibung
-o	Überschreibe (overwrite) die Ausgabedatei(en), falls vorhanden
-v	Sichten als Tabellen archivieren
<login timeout>	Timeout in Sekunden für die Anmeldung (login) (0 für unbegrenzt)
<query timeout>	Timeout in Sekunden für die Abfrage (0 für unbegrenzt)
<import meta data>	Name der Metadaten-XML-Datei, die als Vorlage verwendet werden soll
<external LOB folder>	Ordner, in dem die Daten der grössten LOB (large object) Spalte der Datenbank extern gespeichert werden sollen (gleichnamige Inhalte werden überschrieben!)
<mime type>	MIME-Typ der Daten der grössten LOB-Spalte der Datenbank (beeinflusst die Wahl der Namenserverweiterung der extern gespeicherten LOBs)
<JDBC URL>	JDBC URL der Datenbank, die heruntergeladen werden soll  Beispiele:  <b>MS Access:</b> jdbc:access:<path>/<filename>.mdb  <b>DB/2:</b> jdbc:<host>:50000/<db-name>  <b>MySQL:</b> jdbc:mysql://<host>:3306/<db-name>  <b>Oracle:</b> jdbc:oracle:thin:@<host>:1521:orcl  <b>Postgres:</b> jdbc:postgresql://<host>:5432/<db-name>.  <b>SQL-Server:</b> jdbc:sqlserver://<host>/<db-name>:1433
<database user>	Datenbank Benutzer
<database password>	Datenbank Passwort
<siard file>	Name der zu erstellenden .siard Datei
<export meta data>	Name der zu exportierenden .xml Metadaten Datei



### 5.3.3. Anmerkungen

Mindestens einer der beiden Parameter **siard-file** oder **export meta data** muss gesetzt sein (es ist auch möglich, beide Parameter zu verwenden).

*Hinweis: Während des Archivierungsprozesses sollte der Inhalt der Datenbank nicht verändert werden - verwenden Sie ggf. einen Snapshot der Datenbank.*

Die Option Views als Tabellen archivieren führt in der Regel zu einer redundanten Speicherung derselben Daten und wird daher nicht empfohlen. Sie ist dann sinnvoll, wenn der für die Archivierung verwendete Datenbankbenutzer zwar die Inhalte der Views, nicht aber die der Basistabellen anzeigen kann.

Der Archivierungsprozess ist entweder vollständig erfolgreich oder schlägt fehl.

Bei großen Datenbanken wird empfohlen, zuerst das Metadaten-XML herunterzuladen. Diese enthält alle Metadaten und Tabellengrößen und hilft so, die für den Download benötigte Ausführungszeit abzuschätzen. Außerdem sollte bei großen Tabellen die Option **-q 0** verwendet werden, da nicht abgeschätzt werden kann, wie viele Sekunden eine Größenabfrage dauern wird.

### 5.3.4. Datenbank-Benutzer

Es ist generell nicht empfehlenswert, den Datenbankadministrator (DBA, root, dbo, ...) mit allen Rechten für den Download eines SIARD-Archivs zu verwenden. Der Umfang der im SIARD-Archiv gespeicherten Daten wird durch die Objekte definiert, auf die der Archivbenutzer lesend zugreifen darf. Der globale DBA kann normalerweise alle Datenbanken auf dem System und zusätzlich viele Systemtabellen lesen, die nicht archiviert werden sollen. Aus diesem Grund ist es wichtig, für den Download einen geeigneten Archivbenutzer zu verwenden, der gegebenenfalls für diesen Zweck neu angelegt werden muss.

## 5.4. SiardToDb

SiardToDb ist ein Kommandozeilenprogramm, das ein SIARD-Archiv zu Recherchezwecken in eine Datenbankinstanz lädt.

### 5.4.1. Aufruf

```
siard-to-db [-h] | [-o] [-l=<login timeout>] [-q=<query timeout>]
-s=<siard-file> -j=<JDBC-URL> -u=<Datenbank-Benutzer> -p=<Datenbank-Passwort>
[<schema> <mappedschema>]
```

### 5.4.2. Argumente

Argument	Beschreibung
-o	Überschreiben (overwrite) von gleichnamigen Typen und/oder Tabellen in der Datenbank, falls vorhanden

Argument	Beschreibung
<login timeout>	Timeout in Sekunden für die Anmeldung (login) (0 für unbegrenzt)
<query timeout>	Timeout in Sekunden für eine Anfrage (query) (0 für unbegrenzt)
<siard file>	Name der hochzuladenden .siard Datei
<JDBC URL>	JDBC URL der Zieldatenbank Beispiele:  <b>MS Access:</b> jdbc:access:<path>/<file-name>.mdb  <b>DB/2:</b> jdbc:<host>:50000/<db-name>  <b>MySQL:</b> jdbc:mysql://dbserver.enterag.ch:3306/<db-name>  <b>Oracle:</b> jdbc:oracle:thin:@<host>:1521:orcl  <b>Postgres:</b> jdbc:postgres://<host>:5432/<db-name>  <b>SQL-Server:</b> jdbc:sqlserver://<host>\<db-name>:1433
<database user>	Datenbank Benutzer
<database password>	Datenbank Passwort
<schema>	Name des Schemas im SIARD-Archiv
<mappedschema>	in der Datenbank zu verwendender Schemaname

### 5.4.3. Anmerkungen

Da ältere Datenbanken nicht SQL:2008-konform sind, ist es unvermeidlich, dass einige manuelle Vorbereitungen für den Upload erforderlich sind. In MS Access gibt es keine Schemaobjekte. In Oracle sind Benutzer und Schemas untrennbar miteinander verbunden. In MySQL sind Schema-Objekte und Datenbanken dasselbe. Daher müssen alle Schemas vor dem Upload manuell erstellt werden. Außerdem muss der Datenbankbenutzer das Recht haben, Tabellen und Typen in diesen Schemas anzulegen. Da dies nicht immer einfach ist, werden die Datenbankschemas den Schemanamen entsprechend der Mapping-Liste auf der Kommandozeile zugeordnet.

Der Upload erzeugt nur Tabellen und Typen und versucht, Primär- und Fremdschlüssel zu aktivieren. Es werden keine weiteren Objekte in der Zieldatenbank erzeugt. Wenn die Schlüssel nicht aktiviert werden konnten, wird eine Warnung ausgegeben, aber der Upload gilt trotzdem als erfolgreich abgeschlossen. SQL-SELECT-Abfragen auf der Datenbank können auch ohne diese Schlüssel ausgeführt werden.

Darüber hinaus kann es beim Upload zu Einschränkungen kommen. Bei MS Access landen alle Tabellen in derselben MDB/ACCDB. Bei Oracle werden alle Namen, die länger als 30 Zeichen sind, gekürzt. Um Kollisionen zu vermeiden, werden Tabellen- und Spaltennamen mit einem Zähler versehen. (Zum Beispiel wird aus „Ein Tabellenname, der für Oracle viel zu lang ist“ „Ein Tabellenname, der Table01“).

Wenn die maximale Genauigkeit und die maximale Anzahl Dezimalstellen (z.B. unter MS Access) kleiner sind als benötigt, werden die Werte mit geringerer Genauigkeit hochgeladen. SIARD behilft sich so gut es im Zieldatenbanksystem möglich ist. Wenn die Metadaten der Datenbank mit SiardGui konsultiert werden, können die korrekten Bezeichnungen und Werte zugeordnet werden.

## **6. Tipps und Tricks**

### **6.1. Vorbereitung einer Datenbank für den Download**

SIARD Suite lädt alle Daten herunter, die für den für die Verbindung verwendeten Datenbankbenutzer sichtbar sind. Die Wahl eines geeigneten Datenbank-Benutzers bestimmt daher den Umfang der archivierten Datenbank. Häufig steht ein geeigneter "technischer Datenbank-Benutzer" einer Datenbankanwendung zur Verfügung, mit der die Datenbank verwaltet wird. Dieser hat in der Regel genau die Zugriffsrechte, die für die Archivierung der Datenbank benötigt werden.

Steht jedoch kein solcher Datenbankbenutzer zur Verfügung, sollte für den Download nicht der Datenbankadministrator (database administrator, DBA, dbo, root, SYSTEM, sa, ...) verwendet werden. Denn dieser hat Lesezugriff auf viele Systemtabellen und Tabellen aus anderen Datenbanken, die nicht archiviert werden sollen. Stattdessen wird empfohlen, einen Datenbankbenutzer für die Archivierung anzulegen. Diesem Archivierungsbenutzer sind Leserechte auf alle Schemas, Tabellen, Views und Typen einzuräumen, die zu der zu archivierenden Datenbank gehören. Wie ein Benutzer angelegt und mit den notwendigen Privilegien ausgestattet wird, ist der Dokumentation des DBMS zu entnehmen.

Wenn ein geeigneter Benutzer bestimmt oder angelegt wurde, kann der Datenbank-Download mit dessen Anmeldedaten durchgeführt werden.

### **6.2. Vorbereitung einer Datenbank für den Upload**

Für den Upload einer Datenbank in eine Instanz eines DBMS mit SIARD Suite müssen geeignete Datenbankschemas in der Zieldatenbank vorhanden sein. Diese Schemas können dann im Schema-Mapping des Upload-Dialogs (oder im Schema-Mapping der Kommandozeile von SiardToDb) den Schemas im SIARD-Archiv zugeordnet werden.

Im Idealfall sind die Zielschemas leer. Je nach Sicherheitseinstellungen des DBMS ist es jedoch nicht immer möglich, Schemas nach Belieben zu erstellen oder auszuwählen. Sind die Zielschemas nicht leer, werden die archivierten Daten nur dann hochgeladen, wenn entweder keine Namenskollisionen bei den Typen und Tabellen auftreten oder explizit die Option Überschreiben gewählt wurde.

Der Datenbankbenutzer, dessen Anmeldedaten für den Upload verwendet werden, muss das Recht haben, Typen und Tabellen in den Zielschemas anzulegen und Daten in diese Tabellen einzufügen. Die Dokumentation des DBMS enthält Informationen darüber, wie Schemas angelegt werden und wie dem Datenbankbenutzer die erforderlichen Rechte erteilt werden. Am einfachsten ist es, wenn möglich, die Zugangsdaten des Datenbankadministrators zu verwenden. Denn dieser hat in der Regel alle notwendigen Privilegien. In diesem Fall sollte jedoch nicht die Option Überschreiben gewählt werden. Sonst ist die Gefahr zu groß, dass wichtige Tabellen oder Typen im System überschrieben werden.

## 6.3. Zeit, Zeitstempel und Zeitzonen

Die Konvertierung von TIMES und TIMESTAMPS in der Datenbank hängt von der lokalen Zeitzone ab. Die UTC-Zeit 14:30 im SIARD-Archiv wird beim Upload in Zürich als lokale Zeit 15:30 in die Datenbank geladen - im Winter. Um diese Konvertierung zu unterdrücken, muss SiardToDb mit der Option `-Duser.timezone=GMT` gestartet werden, was SIARD dazu veranlasst, alle Datenbankzeiten als UTC-Zeiten zu interpretieren.

## 7. Einschränkungen

Das SIARD-Format unterliegt folgenden Einschränkungen

- Die Grösse einer SIARD-Datei darf 18'446'744'073'709'551'615 Bytes (ca. 18 ExaBytes) nicht überschreiten (ZIP64-Beschränkung).
- Die Anzahl der Dateien (Tabellen und LOB) darf 4'294'967'295 (ca. 4 Milliarden) nicht überschreiten (ZIP64-Beschränkung).

Die SIARD Suite ist ausserdem beschränkt durch

- Die gesamten Metadaten der Datenbank müssen im JAVA-Speicher (Heap) Platz finden. Diese Beschränkung kann bei einer ungünstigen Konstellation von verfügbarem Hauptspeicher und Komplexität der Datenbankmetadaten auftreten. Sie kann umgangen werden, indem die SIARD-Anwendungen auf einer Maschine mit genügend Hauptspeicher gestartet werden und allenfalls der JAVA-Heap mit dem JAVA-Kommandozeilenparameter `-Xmx2000m` oder einem ähnlichen Parameter auf der JAVA-Kommandozeile vergrössert wird.