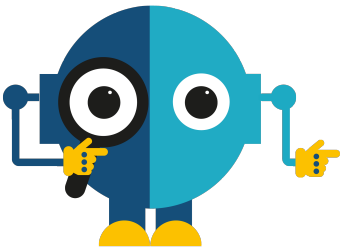


D.O.T

User Guide

DOT Extract

Version 23.0

Publication Date: January, 2023

Prepared by the DOT Software Documentation Team

North America & LATAM
1 N. State St, 15th Floor
Chicago, IL
USA
1-603-371-9074
1-603-371-3256 (support calls only)
sales-us@arcadsoftware.com

EMEA (HQ)
55 Rue Adrastée – Parc Altaïs
74650 Chavanod/Annecy
France
+33 450 578 396
sales-eu@arcadsoftware.com

Asia Pacific
5 Shenton Way #22-04
UIC Building
Singapore 068808
sales-asia@arcadsoftware.com

Contact DOT Software

Headquartered in France at the foot of the Alps, DOT Software offers global services and has offices and partners all over the world. ARCAD partners with leading-edge companies throughout the world to offer full services, close to home.

Visit our website to Contact Us and find out more about our company and partners, or to request a demo.

The ARCAD Customer Portal is intended for current and potential customers that have full or trial versions of ARCAD software. If you already use or are interested in using an ARCAD product, the portal lets you view all of your current licenses and generate your own temporary license keys for most ARCAD products. It grants you access to the ARCAD product knowledge base (new releases, release notes and current documentation).

Do you have a request for change or have you encountered a bug? Log into the ARCAD Helpdesk and create a ticket.

DOT Software guarantees consultant support 24 hours a day, 5 days a week (24/5) to registered members. Calls received are redirected, according to the hour, to put you in contact with a support team in or near your timezone.

Country	Address	Account Contact	Support Contact
France	ARCAD Software (HQ) 55 Rue Adrastée 74650 Chavanod ARCAD Software 17 chemin de la plaine 07200, Saint-Didier-sous-Aubenas	+33 4 50 57 83 96 sales-eu@arcadsoftware.com	Worldwide 24/7: +1 603 371 3256 France only: +33 450 57 28 00 support@arcadsoftware.com ARCAD Helpdesk
Germany	ARCAD Software Deutschland GmbH c/o Pramex International GmbH Savignystr. 43, 60325 Frankfurt am Main		
China	ARCAD Software #2035, Yuehai Plaza, 180 Wanbo 2nd Road, Nancun, Panyu District, Canton	+86 (020)22324643 +86 (020)22324649 sales-asia@arcadsoftware.com	
India	ARCAD Software D-280/281/282, Vibhuti Khand Gomti Nagar, Lucknow		
Singapore	ARCAD Software 5 Shenton Way #22-04 UIC Building Singapore 068808		
USA	ARCAD Software 1 N. State St, 15th Floor Chicago, IL	+1 (603) 371-9074 +1 (603)-371-3256 (support calls only) sales-us@arcadsoftware.com	

Table 1: Contact DOT Software

Preface

Document purpose

This document is intended to guide you through installing and using DOT Extract.

Intended Audience

This document is intended for all DOT Extract users.

Related Documentation

Related Documentation
Release Notes

Table 2: Related Documentation

Publication Record

Unless stated otherwise, all content is valid for the most current version of DOT Extract listed as well as every subsequent version.

Product Version	Document Version	Publication Date	Update Record
≥ 23.0	1.0	January, 2023	Initial Publication.

Table 3: DOT Extract User Guide Publication Record

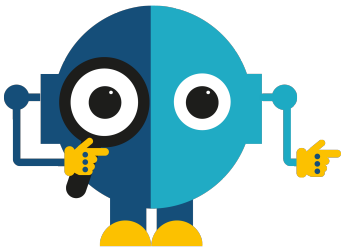
Contents

Injection

Appendix

Tables

Figures



D.O.T

Introduction

1 About DOT Extract

Sample groups of data in the blink of an eye

With DOT Extract, it is possible to create realistic test data sets that are small enough to support fast test cycles, yet large enough to accurately reflect the variety of production data.

Modern software development requires representative test data that must be easily and quickly made available. Providing a copy of the entire production database to data users would be too costly in terms of time and disk space.

DOT Extract can extract realistic and consistent data for testing, analysis, training, or sharing outside your organization.

Data sets created with DOT Extract have the following advantages:

- having access to data sets that contain all the table relationships needed to run programs very quickly and inexpensively,
- limiting the costs related to storage and administration,
- accelerating the development by providing each team with its own test data in parallel.

DOT Extract can extract data from relational databases, that guarantee the fact that the stored data points that are related to one another.

The list of relational databases covered by DOT Extract is the following:

- Microsoft SQL-Server,
- PostGreSQL,
- Oracle,
- MySQL,
- DB2 for i.

2 Functional Description

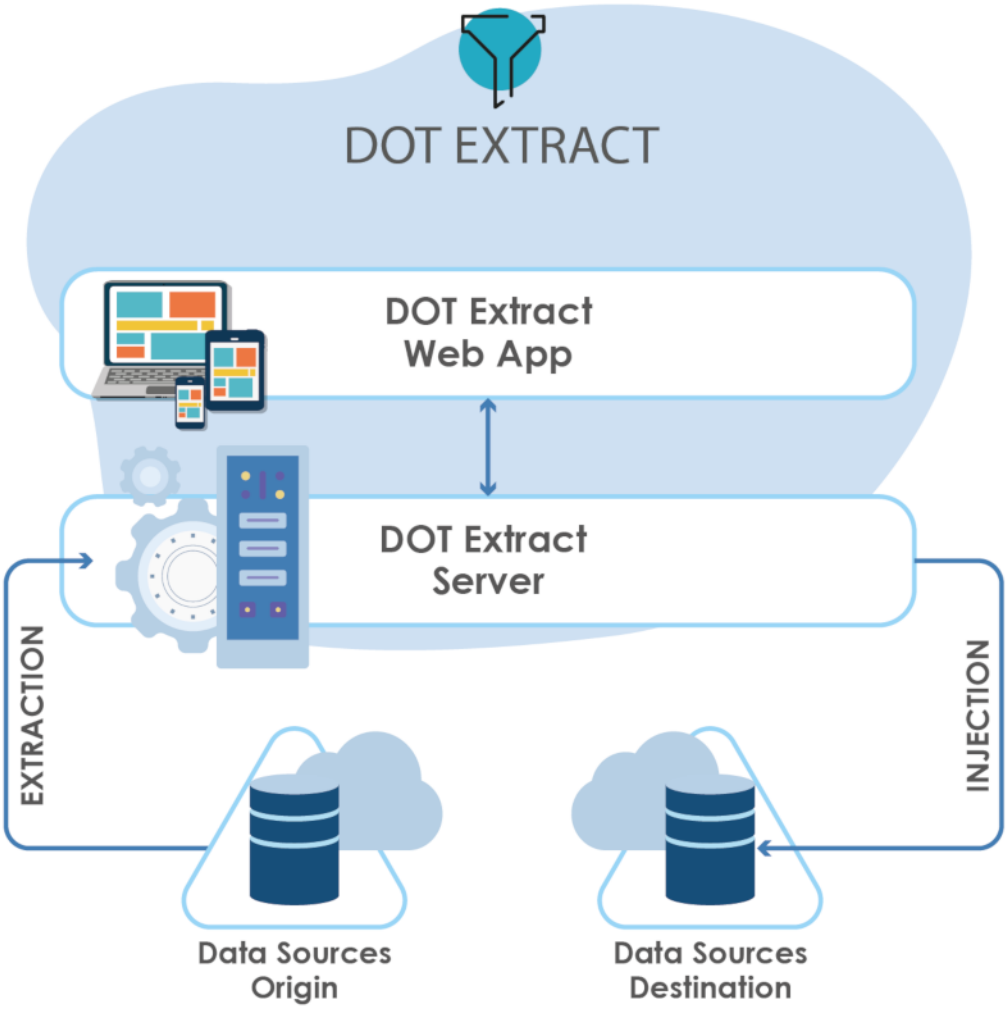


Figure 1: The DOT Extract architecture diagram

DOT Extract is composed by one main element, which is the Extract Server, that needs to be installed. Once it is installed, you can have access to the web interface of DOT Extract.

> ***Reference***
> For more information about installing the Extract Server, refer to Installing the Extract Server on page 16.
> For more information about the connection to the Extract Web App, refer to Connecting to the Extract Server on page 24.

The data sources are the databases external to DOT Extract from where data is extracted and to where it is injected.

The Extract Server contains the data and executes extractions. It stores all the elements needed and required to set up and execute the projects, and gives access to the entities that are delivered with DOT Extract.

The Extract Web App is a Web-based application that connects to the Extract Server. It allows you to create and manage all the entities required to set up and execute extraction and injection projects,

launch those projects and view the results. It also allows you to view and edit the data model of each data source.

3 Simplified concepts

DOT Extract is intended to proceed to the extraction of consistent and smaller sized databases. The data extracted can come from a production database and can be used for test and training environments. The data extractions are done by respecting relational constraints, key uniqueness and the non-generation of duplicate data. DOT Extract also offers the possibility to filter table rows by selecting values or data ranges.

The goal of this page is to be able to create data sources, filters, data extractions and data injections, with concrete examples to follow through, in order to give you some context during the tour of the features available in the Extract Web App.

With DOT Extract, it is possible to get a final extraction project by following four major steps:

1. **Define the connection to data sources and display data models.**

   Data sources represent the databases where the data is stored, from where it is extracted and to where it can be injected.

   The data sources are defined via a wizard on the Extract Web App and once a data source is created and its status is marked as **Available**, you can immediately view its data model. The data model page gathers all the details and information needed about the tables contained in a specific data source, as well as their primary and foreign keys.

   > ***Example***
   > Assuming that an Oracle data source created and defined on DOT Extract contains information about customers and their orders.
   > The data model can then display the description tables contained in the data source, with information such as the **Customer ID**, which is the primary key in a Customers table and foreign key in an Orders table.

   For more details about data source creation and data models, we recommend that you read the following documentation: Data Sources on page 27 and Data Models on page 30.

2. **Create one or several filters on existing data sources.**

   Use filters on large data sources to reduce the scope of data to take into account before launching an extraction. Filters are indeed very precise extraction requests.

   The filters are created via a wizard on the Extract Web App and once they are created, you can view them and you have to edit them to add your customized filter selection.

   > ***Example***
   > Assuming we are still using that same Oracle data source we created, which contains information about customers and their orders.
   > With DOT Extract, you can create a specific filter that can, for instance, retrieve only French customers.
   > To do so, you can create a Selection with a **Binary** filter on the **Address** table, put the **Country** column as first operand, use the **=** operator and set the second operand to **France**.

   For more details about how to create and manage filters and the available operators, we recommend that you read the following documentation: Filters on page 34.

3. **Configure and execute the extraction project.**

   Once the data source and filters on data source are set, you can now create your extraction projects.

   The extraction projects are created via a wizard on the Extract Web App and once they are created, you can view them, edit them and eventually launch them.

   On DOT Extract, an extraction project is created in three steps:

   - Main properties definition,
   - Entry tables selection,
   - Extraction branches definition.

   Once all the parameters have been completed and the extraction project is created, it appears in the list of extractions and it is available for modifications. You can track the status of the extraction process and view all the details of the logs in the Extraction Results view.

   For more details about the creation of an extraction project, its execution and results, we recommend that you read the following documentation: Extractions on page 40 and Extraction results on page 44.

4. **Configure and execute the injection.**

   The final part of the main processes available on DOT Extract is the creation and execution of injection projects, based on the selected extracted data and a chosen injection destination.

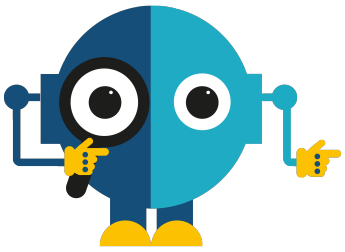
   The injection projects are created via a wizard on the Extract Web App and once they are created, you can view them, edit them and eventually launch them.

   On DOT Extract, an injection project is created in two steps:

   - Main properties definition,
   - Target data source selection.

   Once all the parameters have been completed and the injection project is created, it appears in the list of injections and it is available for modifications.

   For more details about the creation of an injection project, its execution and results, we recommend that you read the following documentation: Injections on page 48 and Injection results on page 51.



D.O.T

Installation

4 Installing the Extract Server

The Extract Server centralizes all the necessary elements to carry out the extraction and injection processes. The server is the core of DOT Extract, as it is required to use the Extract Web App.

> **Important!**
> Before installing or updating DOT Extract, first read carefully the Release Notes.
>
> In these release notes, you will find all the information specific to each release of DOT Extract, especially specific warnings you should be aware of before installing or upgrading the product (compatibility breaks, exceptional migration procedures, etc.).

4.1 Windows

The Extract Server runs as a Windows service.

The installation process takes approximately 5 minutes. It is a simple procedure similar to most other Windows applications. For trouble-free installation, it is recommended that you close all active Windows applications before beginning the installation.

> **Note**
> If the tool is already installed on your computer, reinstalling it will update it to the new version automatically.

4.1.1 Installing

Follow the subsequent steps to install the Extract Server on Windows.

Step 1 Copy the *.exe* file to your machine and execute it to launch the wizard. Your profile must have administrator privileges to run the execution file.

Step 2 From the **Select Setup Language** window, select the language for the installation.

Step 3 Review and accept the license agreement.

The **License Agreement** page presents the DOT Software license agreement for you to review. Please read it carefully. When you have reviewed the agreement, select **I accept the agreement**.

Step 4 Confirm the installation location.

The **Select Destination Location** screen displays the default location where DOT Software elements will be installed. If you prefer to install the software elements in a different location, either type in the location, or click **Browse...** to navigate to and select the alternate location.

The default root location is *C:\Program files\DOT Solutions*.

A sub-folder for each DOT Software application you install will be created at this location.

> ***Warning!***
> The access rights to this folder can then be restricted to the user used to run the Server's Windows service.

Step 5 Install.

The **Ready to Install** screen enables you to review and change or confirm the setup parameters provided and to launch the installation.

To change a setup parameter, click **< Back** and return to the necessary screen.

If you agree with the installation parameters displayed, click **Install** to start the copy phase of the installation process. The process may take a moment while the windows services are launched.

As the installation proceeds, a status bar displays its progress. Each element copied appears above the status bar. Click **Cancel** to interrupt the installation.

Step 6 Define the TCP ports to use.

> **Important!**
> The ports defined here will be used to configure the connection.

Press **Enter**/**Next** to use the default ports.

The Extract Server's default HTTP port is 5257, and the default HTTPS port is 52570.

If an application is already using it, the default port number is incremented until a free port number is found. It is possible to manually change the port after installation.

Step 7 Select the Windows user account.

The Extract Server runs as a Windows service. Select the user account that will be used to manage the service.

By default, the local system account is selected. If you do not want to use that account, you can select any user account with enough privileges to run a Windows service, and have a read and write access to the folder where DOT Extract is installed.

Step 8 Complete the setup process. The final page of the wizard displays confirmation that the setup was a success. Click **Finish** to close the setup wizard.

Step 9 Verify the Windows Service is running.

The final screen of the installation process enables you to automatically open the Local Services Management Console. If the **Open Windows service** checkbox is selected, the **Services** window opens after clicking **Finish**. This window enables you to verify that its status is set to "Started".

Result The Extract Server is installed and available for use.

4.1.2 Uninstalling

Follow the subsequent steps to uninstall the Extract Server from Windows.

Step 1 Log into the system with an administrator account.

Step 2 Either launch the *uninstall.exe* located in the installation directory

- or -

Open **Add or Remove Programs** (*Start > Control Panel*), find the module in the list of installed software and select **Uninstall**.

Step 3 Remove any remaining files in the installation path and remove any remaining configuration files in the user directory.

Result The Extract Server is now completely uninstalled.

4.2 Linux

4.2.1 Install

Follow the subsequent steps to install Extract Server on Linux.

Step 1 Transfer the *.tar.gz* to the */tmp* directory on the target system.

Step 2 Open a terminal session and go to the installation directory.

> ***Example***
> `cd /opt`

> **Note**
> By default, the server is intended to be installed in the */opt* directory. If you install it in a different directory, you will have to update the server's scripts (see Step 7 Edit the scripts.).

Step 3 Execute the following command to extract the *tar.gz* archive into the current directory:

```
tar xzf Extract-Server.tar.gz
```

Step 4 [*Optional*] If you need to change the listening port(s) for the server, you must do so now.

The Extract Server's default HTTP port is 5257, and the default HTTPS port is 52570.

To change the ports, open the *configuration/com.arcadsoftware.server.restful.cfg* in a text editor and change the following properties:

- **port=***<new HTTP port number>* -or- *<0>* (zero) to disable the HTTP port
- **portssl=***<new HTTPS port number>* -or- *<0>* (zero) to disable the HTTPS port

Step 5 Create a specific user and group to run the server's process with the command:

```
adduser --system --no-create-home --group name
```

Step 6 Change the ownership of the installation directory to give it to the user and group that will run Extract Server:

```
chown username:groupname -R /opt/Extract-Server
```

Step 7 Edit the scripts.

Open the following scripts in a text editor to set the variables to match your execution environment:

- *bin/Extract-Server*
  - Line 8: set to the installation directory if it was not installed in */opt/...* by default.
  - Line 12: set to the new user.

- *bin/Extract-Server.service*
  - Lines 8-14: update the paths so they match the installation directory.
  - Line 16: set the user to the new user.
  - Line 17: set the group to be the new user's or any group of your choice.
- configuration files in the *configuration* folder

Step 8 [*Optional*] Execute the following commands to install Extract Server as a systemd service:

```
cd /etc/systemd/system
systemctl link /opt/Extract-Server/bin/Extract-Server.service
systemctl enable Extract-Server
systemctl start Extract-Server
```

Result Extract Server is installed and available for use.

4.2.2 Uninstall

Follow the subsequent steps to uninstall the Extract Server on Linux.

Step 1 Stop the service using the service manager of your Linux distribution or use the script provided.

Example

Service command:

```
systemctl stop Extract-Server
```

Script command:

```
/opt/Extract-Server/bin/Extract-Server stop
```

Step 2 Uninstall the service, if any, using the tools from your Linux distribution.

Example

```
systemctl disable Extract-Server
rm /etc/systemd/system/Extract-Server.service
systemctl daemon-reload
systemctl reset-failed
```

Step 3 Remove the software from the installation directory.

Example

```
rm -rf /opt/Extract-Server
```

Step 4 Remove any remaining files based on the server's configuration.

Result The Extract Server is completely uninstalled.

4.3 IBM i

The Extract Server can be installed and updated manually on IBM i or, depending on your security policy, remotely from any machine running Java.

> **Note**
> If the tool is already installed on your computer, reinstalling it will update it to the new version automatically.

4.3.1 Installing

Follow the subsequent steps to install the Extract Server on IBM i.

Step 1 Copy the installation *.jar* file to any directory (such as the */tmp*) on the target IBM i IFS or the machine that will orchestrate the remote installation.

Step 2 Launch the installation setup.

If you are installing manually on IBM i:

1. Open a session on the target IBM i using the `QSECOFR` profile or an equivalent.
2. Open a command line interpreter using the command `QSH`. Reach the location where you have copied your installation file.
3. In the command line interpreter, launch the setup using the installation command.

> ***Example***
> `java -jar Setup_Extract-Server-x.x.x._IBM i.jar`

If you are installing remotely from Windows or Linux:

1. Launch the setup from the location on the machine orchestrating the installation using the command above. This will open a prompt asking for the **Remote IBM i name/address**.
2. Enter the name/address of the target IBM i.
3. Login using the `QSECOFR` profile or an equivalent.

Step 3 Define the **AFS Starter Installation library** and iASP which will contain the AFS Starter (a utility program used to start the Extract Server on IBM i).

By default they are **AFSSTARTER** and ***SYSBAS**.

Step 4 Define the installation location.

> ***Example***
> */HOME/...*

Step 5 Define the TCP ports to use.

> **Important!**
> The ports defined here will be used to configure the connection.

Press **Enter**/**Next** to use the default ports.

The Extract Server's default HTTP port is 5257, and the default HTTPS port is 52570.

If an application is already using it, the default port number is incremented until a free port number is found. It is possible to manually change the port after installation.

Step 6 Define the **Job user** that will run the Extract Server.

Step 7 Define the job queue library (***LIBL**) then the job queue (**JOBQ**) in which the job will be submitted.

> ***Example***
> Job queue library [ARCAD_SYS]
> Job queue [ARCAD_CTL]

Step 8 Define the name of the AFS instance to register in the AFSSTARTER.

By default it is **PRODUCT-AFS-ID**.

Result The Extract Server is installed and available for use.

4.3.2 Uninstalling

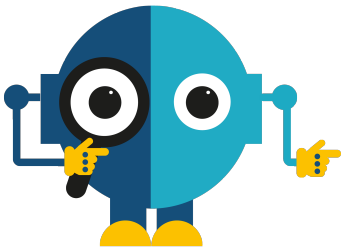
Follow the subsequent steps to uninstall the Extract Server on IBM i.

Step 1 Open a session on the IBM i where the server is installed, using the `QSECOFR` profile or an equivalent.

Step 2 Stop and delete the service using the following commands:

```
ADDLIBLE <AFSSTARTER LIBRARY NAME>
ENDAFSSVR EXTRACT
DLTAFSSVR EXTRACT DELETE(*YES)
```

Result The Extract Server is completely uninstalled.



D.O.T

Getting Started

5 Overview of the Extract Web App

Once the Extract Server is installed, the Extract Web App is accessible from any web-browser.

The navigator provides access to all of the elements required to manage and execute the extraction and injection processes, to view their results, as well as all the administration tools and the documentation of the product.

6 Connecting to the Extract Server

Once the Extract Server is installed, the Extract Web App is accessible from any web-browser.

To access the Extract Web App, enter the following address in your browser:

<web_server_IP_address>:<web_server_port>/dote

The connection dialog opens in the web-browser.

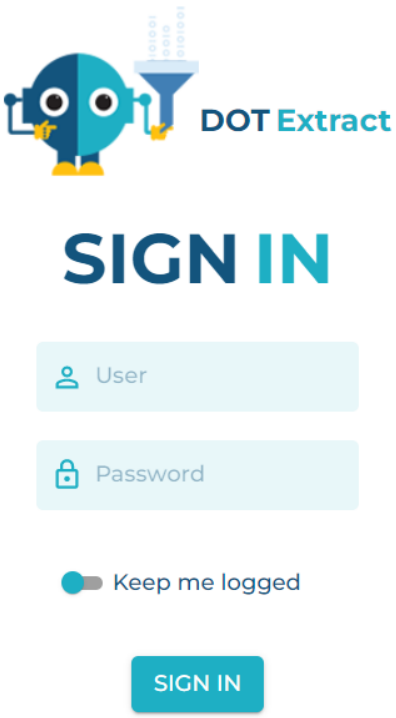
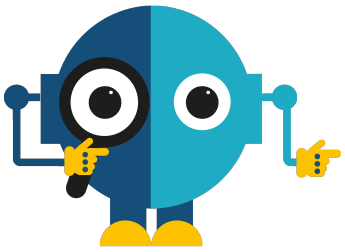


Figure 2: Extract Server connection

Enter your **User Name** and **Password** and click the **Sign In** button to connect to the Extract Server.

The admin-level user/password is: **admin**/**quadra**.



D.O.T

Data Sources

Introduction to setting up the data sources

Data sources allow the management of the databases as well as their data structure in the Extract Server. Connections to the origin and destination databases are configured in the data sources. For each data source, DOT Extract generates its data model, that can be visualized in the Data Sources view.

DOT Extract makes it easy to configure the connection to data sources. These entities must be configured before defining extraction and injection processes.

Data Sources and Data Models are defined, configured, managed and accessed in the Extract Web App.

7 Data Sources

Chapter Summary

The data sources are the databases that contain the data:

- The *origin* data source is the database where the data to subset is extracted from.
- The *destination* data source is the database where the data subset is injected to.

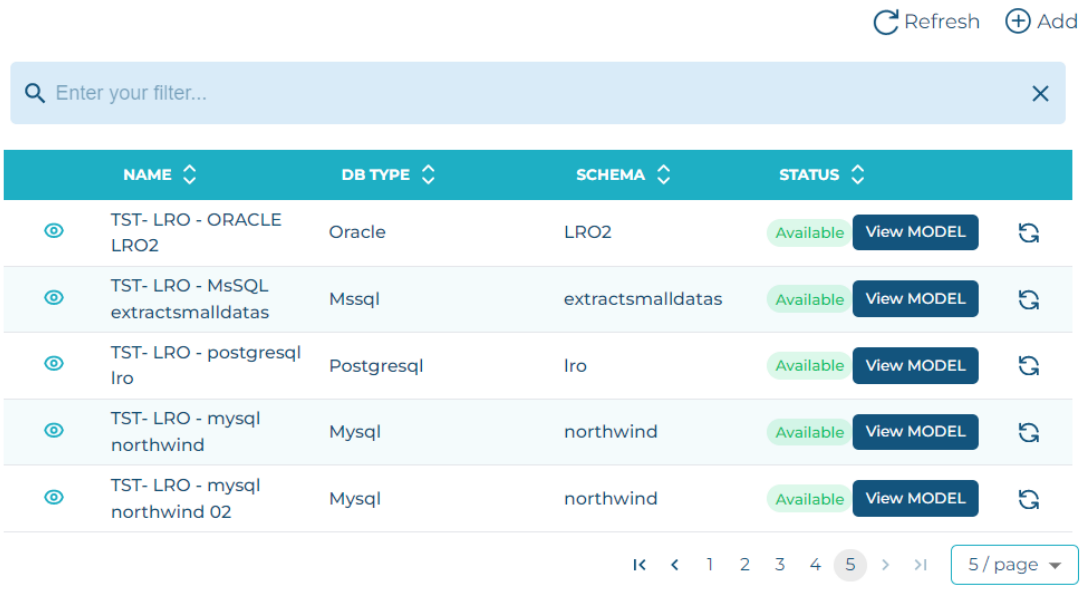


Figure 3: Data Sources

The data sources are accessed and managed in the **Data Sources** view.

7.1 Prerequisites

The users that performs the extraction must have the following user privileges on the extracted tables if they are not the owner of the objects.

Database	User privileges
MySQL	USAGE and SELECT
MS SQL Server	READ or SELECT
Oracle	SELECT object
DB2 for i	READ or SELECT
PostgreSQL	READ or SELECT

Table 4: Databases user privileges

7.2 Manage data source connections

7.2.1 Create a Data Source connection

Follow the subsequent steps to create a new data source connection.

Step 1 Open the **Data Sources** view from the **Data Sources** tab, then click the ⊕ **Add** button.

Step 2 The **Add a new Data Source** wizard opens. Fill the fields to define the required properties for the addition of a new data source.

> ⚠ **Important!**
> All fields are mandatory.

Name

Enter a **Name** for the new data source. This name must be unique.

Database Type

Select a **Database Type** in the drop-down list.

DOT Extract supports the following databases:

- MySQL,
- MS SQL Server,
- Oracle,
- DB2 for i,
- PostgreSQL.

URL / Login / Password

Enter the **URL** of the database, as well as the **Login** and **Password** to have access to it.

Schema

Define the database **Schema** to use. The database schema represents the structure of the database.

ARCAD instance library

[For DB2 for i] Enter the name of the library used by your ARCAD instance.

Application

[For DB2 for i] Enter the name of your ARCAD Application.

Version

[For DB2 for i] Enter the ARCAD version of your application.

Step 3 Click **ADD** to create the set data source, or **CANCEL** to stop the process and close the wizard.

Result The new data source connection is created.

When the data source connection is created, DOT Extract accesses the database and creates the corresponding data model. The data source creation status is **In Progress**.

Once the data model is created, the status of the data source is **Available**. Click the **Refresh** button in case your data source is not displayed yet in the list, or to check its status at all times.

If DOT Extract cannot access the database or create a data model, the status is **Failed**. When this is the case, you can click the **View log** button to display the data source description and the details of the log, providing information about why the connection failed.

7.2.2 Edit a Data Source connection

When you edit the data source connection, the data model is calculated again by DOT Extract.

To edit a data source connection, click the View button to display the details of the data source, then click the Edit button.

The **Edit** page opens and enables you to edit the **Data source Type**, the **URL**, the **Login** or the **Schema** fields.
Click the Save button to save the changes.

> ***Reference***
> For more information about data models, refer to Data Models on page 30.

7.2.3 Delete a Data Source connection

To delete a data source connection, click the View button to view the details of the data source, then click the Delete button.

A confirmation dialog opens, click **Delete** to confirm, or **Cancel** to keep the data source.

> ***Warning!***
> Deleting a data source connection also deletes its data model. Once deleted, the data source and its data model cannot be accessed or recovered.

8 Data Models

The data model is a modelization of the structure of the data in a data source. It contains all the elements necessary to be able to structure the data.

Database models in DOT Extract contain the following elements:

- Tables, as well as the different columns that populate each table, and their corresponding data,
- Foreign keys, and
- Primary keys.

The data model is generated when the data source connection is established. DOT Extract detects the data structure in the data source and creates a model of its structure. The model is stored in the Extract Server in XML format.

When the connection to the data source is modified, the data model is recalculated automatically and updated. In case the data structure in the data source is modified, you need to launch manually the update of the data model. Click the **Update** button to launch the generation of the data model update.

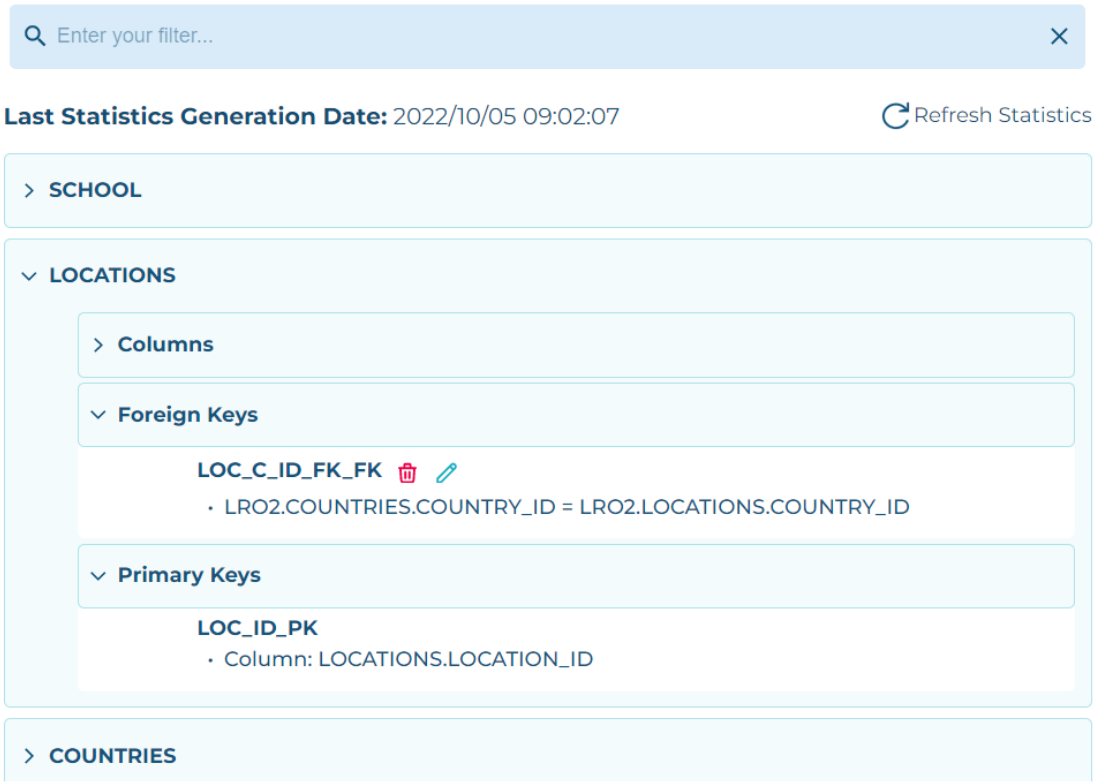


Figure 4: Data models

The Data Models are accessed and managed in the **Data Sources** view. Click the **View Model** button to access each data source model.

When the data source connection is created, DOT Extract accesses the database and creates the corresponding data model. The data source creation status is **In Progress**.

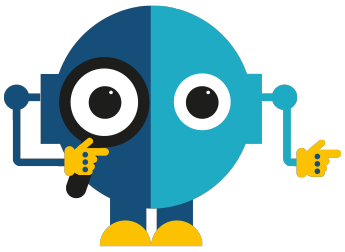
Once the data model is created, the status of the data source is **Available**. Click the **Refresh** button in case your data source is not displayed yet in the list, or to check its status at all times.

If DOT Extract cannot access the database or create a data model, the status is **Failed**. When this is the case, you can click the **View log** button to display the data source description and the details of the log, providing information about why the connection failed.

You are warned when trying to get the data model of a schema that does not exist in the database.

Additionally, it is possible to modify manually the foreign keys in a data model. To do so, click the View button to view the details of the data model, then click the Edit button appearing next to the foreign keys. Click the **Update** button to save the changes.

To remove foreign key links, click the Delete button.



D.O.T

Extraction

Introduction to the extraction process

The extraction process fetches a consistent data set from a data source and stores this data in the Extract Server, for the purpose of injecting it into a testing database later on. The extraction process is based on the data model and the filters to extract the data. The extraction results are generated when the extraction is launched and completed. The extraction results consist of the data archive and instance logs.

The extraction engine allows the definition of queries that include:

- one mode of extraction that retrieves references only, or
- one of the two modes of extraction that retrieve both references and dependencies.

> ***Reference***
> For more information about the possible ways to extract data, refer to Extractions on page 40.

The factorization of extractions with DOT Extract has the advantage of not generating any duplicates, which would go against data uniqueness and the referential integrity constraints.

DOT Extract makes it easy to configure and carry out the extraction process. All of the items required to do so are managed by the Extract Server and configured in the Extract Web App.

Once the extracted data is stored in the archive, it is available to push to other compatible systems. The injection process is in charge of taking all of the data from the archive and putting it into the target environment.

Filters, Extractions, and Extraction results are defined, configured, managed and accessed in the Extract Web App.

9 Filters

Chapter Summary

Filters are *Where* clauses that define the table from which the data is extracted and other filtering clauses on the data to extract.

DOT Extract has three types of filters:

- Binary Clause,
- Unary Clause,
- Logical Clause.

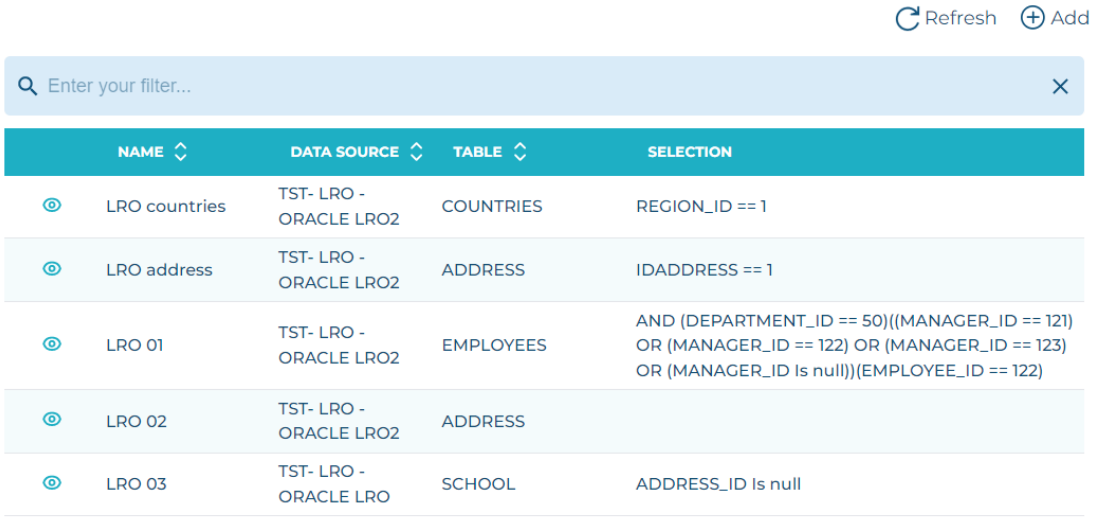


Figure 5: Filters view

Filters are accessed and managed in the **Filters** view.

9.1 Manage filters

9.1.1 Create a filter

Follow the subsequent steps to create a new filter.

Step 1 Open the **Filters** view from the **Filters** tab, then click the ⊕ **Add** button.

Step 2 The **Add a new Filter** wizard opens. Fill the fields to define the required properties for the creation of a new filter.

> **Note**
> The completion of all fields is mandatory.

Name

Enter a **Name** for the new filter. This name can be modified later.

Data Source

Select a **Data Source** in the drop-down list, from one of the data sources previously created.

Table

Select a **Table** in the drop-down list. The tables displayed in the drop-down list match the chosen data source.

Step 3 Click the **Add** button to create the set data source, or the **Cancel** button to stop the process and close the wizard.

Result The new filter is created and appears in the list of filters.

> **Note**
> For the filters to be completely set and work properly, you have to edit them.

To copy an existing filter, click the Duplicate icon. Edit the name of the filter to copy to avoid having duplicates in the list, then click **Done**. Any joins, where clauses, selection criteria or parameters created for your filter is also duplicated.

9.1.2 Edit a filter

Follow the subsequent steps to edit an existing filter.

Step 1 Click the View icon to open the filter you need to edit, then click the Edit icon.

> **Note**
> All existing fields set during the creation of the filter are editable.

> ***Warning!***
> Editing the **Data Source** and **Tables** fields to replace the existing ones with another inevitably removes the previously created Selection and Joins for a specific filter.

Step 2 Click the Selection icon, then select a filter. The available clauses are the following:

Logical

The Logical Clause is used to define multiple filters joined by **AND** or **OR** conditions.

> **Note**
> Logical Clause filters must be immediately followed by a set **Binary** or **Unary** filter clause.

Binary

The Binary Clause is used when an operator is applied between a column and value operand.

> **Note**
> Some operators work with only one operand, it is the case for the **is Null**, **Max** and **Min** operators.

Unary

The Unary Clause is used when a function operator is applied to a column or between a column and a value operand.

> **Important!**
> Make sure that you click the Save icon every time you add a new filter in the Selection, as it is not possible to save all filters at once nor create a new filter without saving the one in the process of creation first.

Result The new filter is edited and the modifications appear in the filter details page.

> ***Reference***
> For more information about the available operators for each filter, refer to the List of operators corresponding to operands on the facing page.

9.1.3 Delete a filter

To delete a filter, click the View icon to open the **Edit** page of the filter, then click the Delete icon.

A confirmation dialog opens, click **Delete** to confirm, or **Cancel** to keep the filter.

> ***Warning!***
> Deleted filters cannot be accessed or recovered.

9.2 Variabilisation of filters

The variabilisation of a filter makes it possible to define some variable values during the execution of an extraction project that runs with said filter. To do so, a dedicated tag is used on filter selection fields, to indicate that the set value is exclusively defined during the execution.

> **Note**
> The variabilisation tags are defined using the following syntax:
> `${PARAMETER_NAME}`.

During the definition of a selection criterion in a filter, it is possible to use a variabilisation tag for the value set in the second operand field.

Attribute	Description
Parameter	Sets the name of the parameter as defined by the user during the filter creation.

Attribute	Description
Description	Sets the description of the parameter. This description appears while requesting for the value.
Default value	Sets the value displayed by default while requesting the parameter value.

Table 5: Execution parameters

After saving the filter, the tags are used to create the Execution Parameters, that can also be modified later on. When a user launches the execution of an extraction project, all the Execution Parameters of all the involved filters are retrieved in the process.

> **Note**
> A graphical element is displayed to allow the user to define the value of each execution parameter. During the Extraction Execution, the variabilisation tag linked to the Execution Parameter is replaced by the input values.

> **Important!**
> It is not possible to edit the parameter name, nor add or remove a parameter.

9.3 List of operators corresponding to operands

> **Note**
> All the filter clauses operators are available in their **Negative** form, meaning that it filters in reverse.

The details of each operator are the following:

Operator	Description	Value type
	Binary	
==	Retrieves values that are strictly equal. The value set in the second operand is case sensitive. ***Example*** Using the == operator on a NAME column, with the a second operand set to 'Amanda', retrieves the list of employees whose name is Amanda in an Employee table.	String, integer or date
>	Retrieves values that are greater than the one set in the second operand. ***Example*** Using the > operator on a SALARY column, with the a second operand set to 10.000, retrieves the list of employees that have a salary that is greater than 10.000 in an Employee table.	String, integer or date
>=	Retrieves values that are greater than or equal to the one set in the second operand. ***Example*** Using the >= operator on a SALARY column, with the a second operand set to 10.000, retrieves the list of employees	String, integer or date

Operator	Description	Value type
	that have a salary that is greater than or equal to 10.000 in an Employee table.	
Contains	Retrieves values that contain the characters set in the second operand. ***Example*** Using the Contains operator on a NAME column, with a second operand set to W, retrieves the list of names containing W in an Employee table.	String or integer
Starts with	Retrieves values that start with the characters set in the second operand. ***Example*** Using the Starts with operator on a NAME column, with the a second operand set to E, retrieves the list of names starting with E in an Employee table.	String or integer
Ends with	Retrieves values that end with the characters set in the second operand. ***Example*** Using the Ends with operator on a NAME column, with the a second operand set to N, retrieves the list of names ending with N in an Employee table.	String or integer
=	Retrieves values that are equal. The value set in the second operand is not case sensitive.	String
Before	Retrieves values that are anterior to a certain date set in the second operator. The date follows the following format: yyyy-mm-dd. Click the icon to select a date in the displayed calendar.	Date
Unary		
is Null	Retrieves results with empty values. This operator has only one operand. ***Example*** Using the is Null operator on an EMAIL column retrieves the employees that do not have an email address in an Employee table.	
Max	Retrieves the column that has the maximum value. This operator has only one operand. ***Example*** Using the Max operator on a SALARY column retrieves a list of the employees that have the highest salary in an Employee table.	
Min	Retrieves the column that has the minimum value. This operator has only one operand. ***Example*** Using the Min operator on a SALARY column retrieves a list of employees that have the lowest salary in an Employee table.	
First Asc	Retrieves a specified number of results in the first ascendant results. The amount of results wanted is set in the second operand. ***Example*** Using the First Asc operator on a ZIP_CODE column, with a second operand set to 10, retrieves the first 10 zip codes	Integer

Operator	Description	Value type
	contained in an Address table, in ascending order.	
Last Asc	Retrieves a specified number of results in the last ascendant results. The amount of results wanted is set in the second operand. ***Example*** Using the Last Asc operator on a COUNTRY column, with a second operand set to 10, retrieves the last 10 countries contained in an Address table, in ascending order.	Integer
Longer	Retrieves results that contain more characters than the number set in the second operand. ***Example*** Using the Longer operator on a CITY column, with a second operand set to 10, retrieves the city names that contain more than 10 characters in an Address table.	Integer
First Desc	Retrieves a specified number of results in the first descendant results. The amount of results wanted is set in the second operand. ***Example*** Using the First Desc operator on a ZIP_CODE column, with a second operand set to 10, retrieves the first 10 zip codes contained in an Address table, in descending order.	Integer
Last Desc	Retrieves a specified number of results in the last descendant results. The amount of results wanted is set in the second operand. ***Example*** Using the Last Desc operator on a ZIP_CODE column, with a second operand set to 10, retrieves the last 10 zip codes contained in an Address table, in descending order.	Integer

Table 6: Operators available according to filter clauses

10 Extractions

Chapter Summary

The extraction process consists of getting a set of consistent data from a source and create an archive stored in a H2 database file in the Extract Server, while guaranteeing the referential integrity of the database. All of the extractions defined in a request are stored in the same archive and are based on the same data model. The definition of the referenced data to retrieve is defined in the extraction package.

The extraction, based on the source data model, retrieves all of the dependent data so as to get a consistent set of data in the archive. Filters are applied during the extraction to restrict the rows selected, avoiding retrieving all rows from the tables.

The results of the extraction are kept for each extraction instance, unless the instance is deleted. The extraction results can also be injected as many times as necessary, as long as the target database is compatible, meaning that it contains the same table names and structures as the extraction tables stored in the archive.

For each extraction process, the extraction request contains the extraction branches, each of it containing extractions. The extraction request is made of one or multiple extraction definition. The initial extraction is the first request made to the database to get initial data from one table. Each referenced or dependent data linked to this first set of data fetched at level 0 constitute the next level, and so on and so forth, and eventually, each fetched data has a new level. As a level of extraction is represented by a table, filters can be applied on levels.

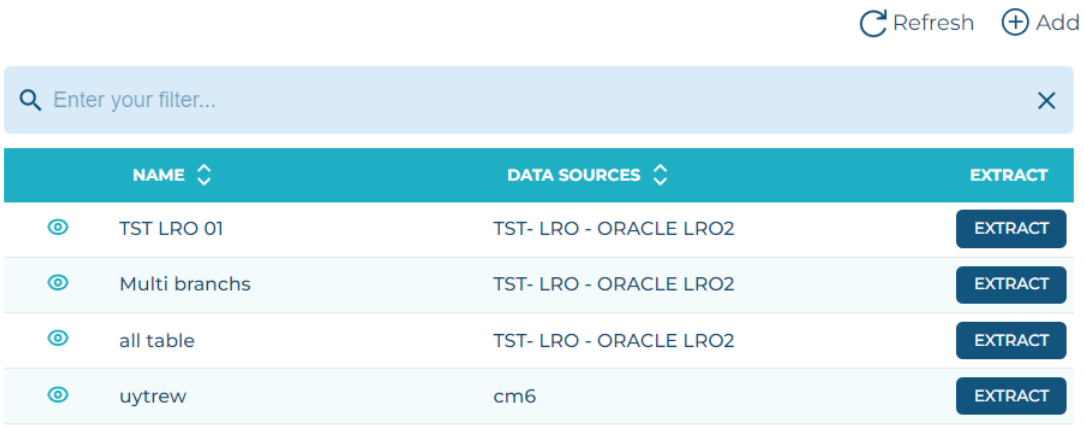


Figure 6: Extractions view

Extractions are accessed and managed in the **Extractions** view.

10.1 Extraction modes

The extraction process gets a set of consistent data from a source and stores that data in an archive. The definition of the data to retrieve is defined in the extraction package. The extraction, with the help of the provided source data model, is in charge of retrieving all of the direct and intermediate linked table data of the initial table. A filter can be defined in the extraction package so as to define the rows to extract, avoiding retrieving all rows from the initial extracted table.

Data linked to extracted data can be referenced or dependent. To comply with these two options and validate the data model with respect of referential integrity constraints, DOT Extract offers three extraction modes: **referenced**, **all** and **all limited**.

> **Note**
> Once the extraction mode is selected for an extraction request, it stays the same for all of the following extraction processes.

Referenced Data

With this mode of extraction, extract the parent data, in accordance with the declared SQL referential integrity constraints. This type of extraction extracts the data linked to the filter and the data that is referenced by it.

In this mode, only referenced tables are extracted.

All Limited

With this mode of extraction, dependent and referential tables are taken from the initial table to extract.

Note that referenced tables can only extract referenced table, whereas dependent tables can extract both referenced and dependent tables. This type of extraction allows to exclude data dependencies during the extraction.

All

With this mode of extraction, extract the parent and the child data, in accordance with the declared SQL referential integrity constraints. This type of extraction allows to extract the data linked to the filter, the data that are referenced by it as well as the data that are dependent on it.

In this mode, referenced and dependent tables are extracted.

> **Important!**
> Use this extraction mode with caution, as the extraction process is more likely to fetch the entire data included in the database.

10.2 Manage extractions

10.2.1 Create an extraction

Follow the subsequent steps to create a new extraction.

Step 1 Click on the **Extractions** tab to open the **Extraction** view, then click the ⊕ **Add** button.

Step 2 Define the main properties required for the new extraction.

> **Note**
> All fields are mandatory.

Name

Enter a **Name** for the new extraction.

Extraction Type

Select the extraction mode to be used, among the ones available in the drop-down list.

> ***Reference***
> For more information about extraction modes, refer to Extraction modes on page 40.

Archive password

Define a password to access the H2 database where the extraction is stored.

Data Source

Select a data source in the drop-down list.

> ***Reference***
> For more information about data sources, refer to Manage data source connections on page 28.

Description

Add a textual description of the extraction project.

Click **Next**.

Step 3 Define the entry tables required for the new extraction. Tick the checkbox corresponding to the table(s) that need to be included in the extraction.

Click **Next**.

Step 4 Define the extraction branches required for the new extraction. Click the ⊕ **Add** button to add a new line.

> **Note**
> It is possible to extract a table in an extraction project without adding a filter.

Table

Select a **Table** in the drop-down list. The list of table is related to the chosen data source.

Filter

Select a **Filter** in the drop-down list. The list of filters is related to the chosen table in the data source.

Step 5 Click **Done** to confirm and close the extraction creation pages.

Result The new extraction is created and appears in the list of extractions.

To edit an extraction, click the 👁 View icon to view the details of the extraction project, then click the ✎ Edit icon. The **Edit** page opens and enables you to edit the **Name**, the **Extraction Type**, the **Data source** and the **Archive passwords** fields.

You can also add another table to the extraction project, by clicking on the ⊕ **New Branch** icon. Select the table(s) in the **Table Selection** window, then click the **Add** button.

Click the ✎ Edit icon on a table to add a filter. Click the 🔓 icon to exclude dependencies to be extracted from the master table with this filter.

For each table, you can display the list of dependent tables of the current table, by clicking the +D icon, or display the list of referenced tables by clicking on the +R icon.

It is also possible to enable the child Extraction mode with the icon, or disable the child extraction mode with the icon.

> **Note**
> Make sure you click the Save icon to save the changes.

To execute an extraction, click the **Extract** button. A dialog opens to confirm that the extraction is launched. You can track the status and the results of the extraction in the **Extraction Results** view.

If you created an extraction project with variable filter values, an **Execution Parameters** page opens when you click **Extract**, so you can define the value of these filter parameters. Once the parameters are set for the extraction, click **Done** to launch the extraction.

> ***Reference***
> For more information about Extraction Results, refer to Extraction results on page 44.

To copy an existing extraction project, click the Duplicate button. Edit the name of the extraction to copy to avoid having duplicates in the list, then click **Done**.

10.2.2 Delete an extraction

To delete an extraction, click the View icon to open the **Edit** page of the extraction, then click the Delete icon.

A confirmation dialog opens, click **Delete** to confirm, or **Cancel** to keep the extraction.

> ***Warning!***
> Deleted extractions cannot be accessed or recovered.

11 Extraction results

The Extraction Instance view is accessed and managed in the **Execution Results** view in the Extract Web App.

Figure 7: Extraction Results

Extraction Workflow

When you launch an extraction on DOT Extract, it follows the steps below:

1.	The extractor analyzes the database and defines a list of primary keys.
2.	A SELECT query extracts the data by following the constraints. **Note** Additional rights and user privileges might be needed depending on the database type, to read the data of the schema/library to extract.
3.	An archive database is generated with the structure and the extracted data.
4	The extraction process takes the data model and the extraction instructions to create an H2 archive and the extraction results.

Extraction results overview

The extraction instance is generated when the extraction is launched. To see all the extraction results updated, click the **Refresh** button.

The extraction results show for each execution of an extraction, the **Start Date** and the **Duration** of the extraction instance. The **Status** indicates if the extraction is On Going, Done or Failed.

When the extraction status is On Going, you can hover the status to have more information about the execution progress.

To view the data extracted and stored in the Extract Server H2 database, click the View button.

View extraction results logs

To display the details of the extraction process, click the View button to open the logs and details page, then click the **Details** button.

Table name	Description
Table Name	Refers to the name of the tables that are extracted.
DB Lines	Indicates the number of lines of the table that are included in the database.
DB Size in KB	Indicates the size of the database in Kilobytes, that is calculated for the extraction.

Table name	Description
Extracted Lines	Indicates the final number of lines of the table that are extracted, after being filtered for the extraction.
Extracted KB	Indicates the final size of the database in Kilobytes, that is extracted.
Lines %	Indicates the percentage of lines extracted, compared to the number of lines included in the database.
Size %	Indicates the percentage of the size of data extracted, compared to the total size of data contained in the database.

Table 7: Extraction results details

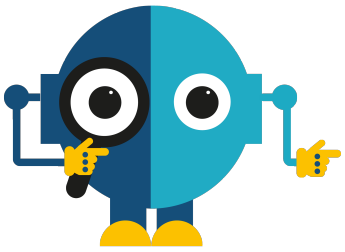
To display the logs of the extraction process, click the View button to open the logs and details page, then click the **Logs** button.

Delete an extraction result

To delete an extraction result, click the View button to open the **Edit** page of the extraction result, then click the Delete button.

A confirmation dialog opens, click **Delete** to confirm, or **Cancel** to keep the extraction result.

> ***Warning!***
> Deleted extraction results cannot be accessed or recovered. The associated extracted data is deleted as well.



D.O.T

Injection

Introduction to the injection process

Once the extracted data is stored in the archive, it is available to push to other compatible systems.

The injection process refers to the entire operation, consisting of retrieving data sets from database tables by executing a query, storing the extracted data in a temporary library, transmitting the temporary library to the test location if remote, and installing it into the appropriate tables in the testing location.

The target data source, where the data is injected, must contain the same tables name and structures as the extracted tables stored in the archive.

The target data model must also be provided, so as to quickly and precisely find the tables where to inject.

DOT Extract makes it easy to configure and carry out the database injection process.

All of the following items are defined, configured, managed and accessed in the available menus of the Extract Web App:

- Injection (see Injections on page 48),
- Injection results (see Injection results on page 51).

12 Injections

Chapter Summary

The injection process recreates the data subset in the chosen destination data source. The injection results is the log of the injection process.

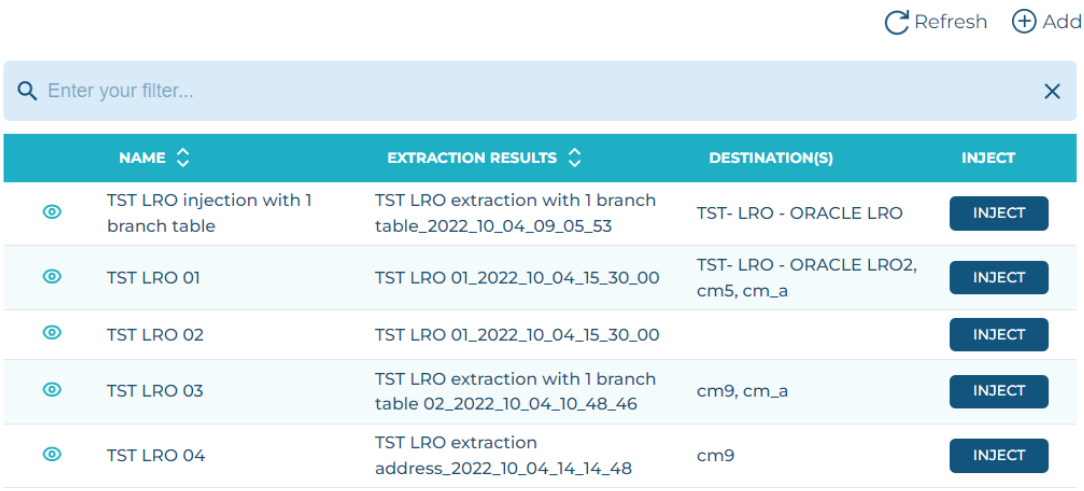


Figure 8: Injections view

Injections are accessed and managed in the **Injections** view.

12.1 Prerequisites

The users that perform the injection must have the following user privileges on the target database tables, if they are not the owner of the objects.

Database	User privileges
MySQL	USAGE, SELECT, INSERT or UPDATE
MS SQL Server	SELECT, INSERT, UPDATE or ALTER
Oracle	SELECT, INSERT, UPDATE or ALTER
DB2 for i	SELECT, INSERT or UPDATE
PostgreSQL	The user must be the owner of the table. Required by the ALTER TABLE for referential constraints.

Table 8: User privileges for injections

12.2 Manage injections

12.2.1 Create an injection

Follow the subsequent steps to create a new injection.

Step 1 Click on the **Injections** tab to open the **Injection** view, then click the ⊕ **Add** button.

Step 2 Define the main properties required for the new injection.

Name

Enter a **Name** for the new injection.

Extraction Result

Select an **Extraction Result** in the drop-down list.

Click **Next**.

Step 3 Define the Target Data Sources required for the injection.

Data Source

Select a **Data Source** in the drop-down list. You can select multiple destinations in the list to create multiple injections. To do so, click on the ⊕ Add button to confirm the selected data source and to be able to define another, if necessary. To remove a data source, click the 🗑 Delete button.

> **Note**
> In case of a multiple injection, the data is injected successively into the selected destinations and not simultaneously.

> ***Warning!***
> Multiple injections are exclusively possible on databases with the same type and structure as the source database used for the extraction.

Step 4 Click **Done** to validate the injection and close the dialog.

Result The new injection is created and appears in the list of injections.

To edit an injection, click the 👁 View button to view the details of the injection project, then click the ✎ Edit button. The **Edit** page opens and enables you to edit the **Name**, the **Extraction Result** and the target data source(s).

> **Important!**
> Make sure you click the ⊕ Add button when you select an additional data source for injection, to save it and being able to add another one. Click the 🗑 Delete button to remove an existing data source.

Click the 💾 Save button to save the changes.
To execute an injection, click the **Inject** button. A dialog opens to confirm that the injection is launched. You can track the status and the results of the injection in the **Injection Results** view.

> ***Reference***
> For more information about the Injection Results, refer to Injection results on page 51.

To copy an existing injection project, click the Duplicate button. Edit the name of the injection to copy to avoid having duplicates in the list, then click **Done**.

12.2.2 Delete an injection

To delete an injection result, click the View button to open the **Edit** page of the injection, then click the Delete button.

A confirmation dialog opens, click **Delete** to confirm, or **Cancel** to keep the injection.

> ***Warning!***
> Deleted injections cannot be accessed or recovered.

13 Injection results

The Injection Instance view is accessed and managed in the **Injection Results** view in the Extract Web App.

Injection Workflow

When you launch an injection on DOT Extract, it follows the steps below:

1.	Disable the target foreign key constraints.
2.	Insert data in the target schema from the archive.
3.	Enable the target foreign key constraints. **Important!** If an error occurs during the activation of the target foreign key constraints, the injection is incomplete, and some of the data is inconsistent or even missing.

Injection results overview

The injection instance is generated when the injection is launched. To see all the injection results, click the **Refresh** button.

The injection results shows for each execution of an injection, the **Start Date** and the **Duration** of the injection instance. The **Status** indicates if the extraction is On Going, Done or Failed.

When the extraction status is On Going, you can hover the status to have more information about the execution progress.

To view the data injected, click the View button.

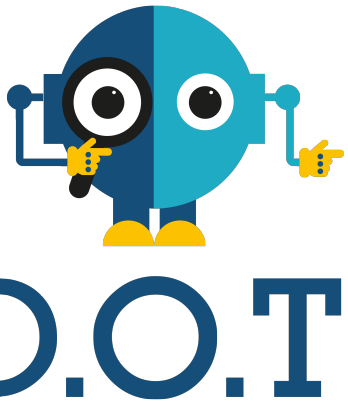
To display the logs of the injection process, click the View button to open the logs and details page, then click the **Logs** button.

Delete an injection result

To delete an injection result, click the View button to open the **Edit** page of the injection result, then click the Delete button.

A confirmation dialog opens, click **Delete** to confirm, or **Cancel** to keep the injection result.

Warning!
Deleted injection results cannot be accessed or recovered.



Appendix

Command Line Interface (CLI)

The DOT Extract Command Line Interface is used to execute predefined extraction and injection processes from the command line. Use the CLI to:

- execute extraction and injection projects,
- list the execution parameters of an extraction execution,
- create injection projects directly from the CLI.

The DOT Extract CLI is distributed as a zip file. Extract the content into the folder of your choice. Go to the */bin* folder to launch the CLI.

> **Note**
> The CLI needs to have access to a Java 8 Runtime Environment that needs to be installed and accessible from anywhere in a command line terminal (run *<code>java -version </code>* to find the current accessible Java version).

To call any action using the CLI, the following syntax is used: `extract` *<subcommand> <parameters>*.

Each subcommand requires one or more parameters. To display the list of supported parameters for any subcommand, run the subcommand without any parameters. If nothing is provided, the system prompts you to fill in the blanks required to accomplish the subcommands tasks.

Parameters between *[brackets]* are optional. Each parameter has a long and a short form (example: --project, -p) and are separated from their value by the equal sign = or a space.

- executeExtraction,
- executeInjection,
- listParameters, and
- createInjectionProject.

Common parameters

The following parameters are common to all CLI commands.

`-H, --help`
Display the help message for the command.

`-S, --server`
[Required] The URL of the Extract Server to which to connect.

`-U, --user`
[Required] The user used to connect to the Extract Server.

`-P, --password`
[Required] The password used to connect to the Extract Server.

`-C, --crypted`
Required if the password is Base64 encrypted.

`-L, --log-file`
The path to the file where the logs will be written.

`-V`, `--verbose`

Enable debug and trace log levels.

executeExtraction

This command allows the user to execute an extraction based on an Extraction project passed as a parameter. With this command, the extraction is prepared and executed.

Usage: `extract executeExtraction [-CHV] -P[=<password>] [-P[=<password>] [-L=<logFile>] -p=<extractionProjectName>S=<serverURL> -U=<user> [--param=<parameters>]`

The available parameters for the `executeExtraction` subcommand are the following:

`-p`, `--project`

The name of the Extraction project used for the extraction execution.

`--param`

An Extraction execution parameter.

executeInjection

This command allows the user to execute an injection based on an Injection project passed as a parameter. With this command, the Injection is prepared and executed.

Usage: `extract executeInjection [-CHV] -P[=<password>] [-P[=<password>] [-L=<logFile>] -p=<extractionProjectName>S=<serverURL> -U=<user>`

The available parameters for the `executeInjection` subcommand are the following:

`-p`, `--project`

The name of the Injection project used for the injection execution.

listParameters

This command allows the user to retrieve the list of the execution parameters of a specified extraction execution.

Usage: `extract listParameters [-CHV] -P[=<password>] [-P[=<password>]]...[-L=<logFile>] -S=<serverURL> -U=<user>-x=<extractionInstanceName>`

The available parameters for the `executeInjection` subcommand are the following:

`-x`, `--extraction`

The name of the Extraction execution to retrieve its corresponding list of parameters.

createInjectionProject

This command allows the user to create an Injection project directly from the CLI.

Usage: `extract createInjectionProject [-CHVx] -P[=<password>] [-P[=<password>]]... [-L=<logFile>]-n=<injectionProjectName> -S=<serverURL>-U=<user> -xn=<extractionExecutionName>-d=<datasourceNames>[-d=<datasourceNames>]..`

The available parameters for the `createInjectionProject` subcommand are the following:

`-d`, `--datasource`

The name of a target data source linked to the Injection project.

`-n`, `--name`

The name of the Injection project.

`-x`, `--execute`

Indicates whether the Injection project must be executed after its creation.

`-xn`, `--extraction`

The name of the Extraction execution that is used by the newly created Injection project.

Glossary

A

Archive

An independent container, usually an H2 file, that stores all the extracted data, the meta data of the extraction and injection tasks, as well as the administrative information.

C

Cache

There are two types of caches: the regular engine cache and the engine execution cache. The engine cache is permanent, unless you voluntarily clear the cache. The engine execution cache is temporary and is cleared at the end of the execution of the project or the chain of projects.

D

Data Model

A modelization of the structure of the data in a data source. It contains all the elements necessary to be able to structure data.

Data Source

Databases that contain the data. - The source data source is the database where the data to subset is extracted from. - The target data source is the database where the data subset is injected to.

E

Extraction

Process that consists of getting a set of consistent data from a source and create an archive stored in a H2 database file in the DOT Extract Server, while guaranteeing the referential integrity of the database.

Extraction Mode

Type of extraction that is made on a data source.

Extraction Package

The data extraction package is composed of the extraction request that defines the scope of data to extract and the archive that defines the media to store the extracted data. The extraction package defines the extraction and injection jobs.

Extraction Request

Defines the extent of data to be extracted. The extraction request retrieves the extraction mode and contains the extraction branches.

F

Filter

Where clause that defines the table from which the data is extracted and other filtering clauses on the data to extract.

I

Injection

Process that consists of installing extracted data from an extraction backup into one or several databases (datasources). DOT Extract manages repeatable and multiple injections.

R

Relational Database Management System (RDBMS)

A collection of hardware and software that organizes and provides access to a relational database.