

Version 4 Release 1

*IBM i2 Analyze
Information Store Data Deletion Guide*



Note

Before you use this information and the product that it supports, read the information in [“Notices”](#) on [page 21](#).

This edition applies to version 4, release 1, modification 7 of IBM® i2® Analyze and to all subsequent releases and modifications until otherwise indicated in new editions. Ensure that you are reading the appropriate document for the version of the product that you are using. To find a specific version of this document, access the Configuring section of the [IBM Knowledge Center](#), and ensure that you select the correct version.

© **Copyright International Business Machines Corporation 2015, 2018.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

- Information Store data deletion..... 1**
 - About this guide.....1
 - Contacting IBM Support.....1
 - Overview..... 2
 - Deleting ingested records..... 2
 - Deleting shared records..... 3
 - Purging soft-deleted records.....3
 - Deleting records by rule..... 5
 - Deleting data by rule.....6
 - Verifying deletion by rule..... 9
 - Automating deletion by rule.....10
 - Troubleshooting.....11
 - Deletion views and columns.....11
 - Stored procedures..... 13
 - DELETION_BY_RULE_LOG view.....15
 - Changing the automated job creation schedule..... 16
 - Authorization to delete by rule.....17
 - Sample use cases..... 17
- Notices.....21**
 - Trademarks.....22

Information Store data deletion

About this guide

IBM i2 Analyze supports several different mechanisms for creating records in the Information Store. It also supports different mechanisms for deleting them. This document describes how to delete records individually, selectively, or in bulk; and it describes the circumstances in which each approach is possible or desirable.

This guide is intended for system implementers and database administrators who want to delete data directly from the Information Store. It applies to data added by ingestion from one or more external data sources, and to data added by users who share information through IBM i2 Analyst's Notebook Premium.

Note: For more information about adding data by ingestion, see [Information Store data ingestion](#). For more information about adding data through IBM i2 Analyst's Notebook Premium, see [Uploading records to the Information Store](#).

The tasks that the following topics describe variously require familiarity with the IBM i2 Analyze data model, DB2 database management systems, SQL queries, and IBM Data Studio. Some of the approaches also require specialized authorization with the Information Store.

Contacting IBM Support

IBM Support provides assistance with product defects, answers FAQs, and helps users to resolve problems with the product.

About this task

After trying to find your answer or solution by using other self-help options such as technotes, you can contact IBM Support. Before contacting IBM Support, your company or organization must have an active IBM software subscription and support contract, and you must be authorized to submit problems to IBM. For information about the types of available support, see the Support portfolio topic in the *Software Support Handbook*.

Procedure

To contact IBM Support about a problem:

1. Define the problem, gather background information, and determine the severity of the problem.
For more information, see the Getting IBM Support topic in the *Software Support Handbook*.
2. Gather diagnostic information.
3. Submit the problem to IBM Support in one of the following ways:
 - Online through the IBM Support Portal at [Support Portal](#). You can open, update, and view all of your service requests from the Service Request portlet on the Service Request page.
 - By phone. For the phone number to call in your region, see the Directory of worldwide contacts web page at <https://www.ibm.com/planetwide/>

Results

If the problem that you submit is for a software defect or for missing or inaccurate documentation, IBM Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the IBM Support website daily, so that other users who experience the same problem can benefit from the same resolution.

Overview

To comply with data protection regulations, storage restrictions, or other requirements, you might need to delete data from the Information Store. IBM i2 Analyze provides several mechanisms for doing so. The mechanism that you choose depends on how the records were created in the Information Store and your reason for deleting them.

At version 4.1.7 of i2 Analyze, there are three ways to delete records from the Information Store. Your approach can vary depending on how the records were created:

- Some of the records in the Information Store are created through a process of ingestion from an external source. If the data for those records is deleted in its source, you can arrange for the ingestion pipeline to delete the records from the Information Store as well.
- Some of the records in the Information Store are created and uploaded through Analyst's Notebook Premium. If a user deletes a record that they or a colleague uploaded, it becomes unreachable for all users, but remains in the Information Store. You can arrange to delete these *soft-deleted* records permanently, either automatically or on demand.
- For all records in the Information Store, you can write identifying rules that target them for deletion individually or as a group. This approach deletes records no matter how they were originally created or what their current state is.

It is common to all record deletion that deleting an entity record forces i2 Analyze to delete the link records that are attached to that entity. A link must always have two ends. If one end is deleted, the link is automatically deleted as well.

It is also common to all these ways of deleting records from the Information Store that the procedure is permanent. If you might need to recover the deleted data in future, ensure that you have a backup plan in place before you begin.

Deleting ingested records

If records that you want to delete from the Information Store are ingested from an external source, you can use the same tools to remove them that you used to create them. If you can generate the origin identifier of an ingested i2 Analyze record, you can use that information to delete it.

For many external sources, the purpose of setting up an ingestion pipeline is not only to create records in the Information Store from that source, but also to maintain synchronization.

In i2 Analyze, ingestion pipelines can automatically update records when the data changes from one invocation of an ingestion command to the next. If the data for a record is deleted from its source, and the source can inform you of that fact, then you can use the ingestion pipeline to delete the record from the Information Store too.

For more information about using the ingestion pipeline to delete records, see [Deleting data from the Information Store](#) in the *Information Store data ingestion guide*.

Deleting shared records

Analyst's Notebook Premium enables authorized users to create and delete records in the Information Store. However, even authorized users can delete only records that they or their colleagues create, and the operation is not immediately permanent. To delete records completely, you must interact with the Information Store database.

When a user deletes a record in the Information Store through Analyst's Notebook Premium, the effect for all users is instant. The record is no longer returned in the results of search or expand operations, and copies of the record on other Analyst's Notebook charts become orphaned.

However, by default, the data for the record remains in the database until you decide to delete it permanently. The Information Store has a mechanism for removing these "soft-deleted" records that you can invoke on an automatic or a manual basis.

The mechanism for purging (that is, fully deleting soft-deleted) records uses three stored procedures from the IS_PUBLIC schema of the Information Store database.

Procedure	Parameters	Description
PURGE_SOFT_DELETED_RECORDS		Permanently deletes all soft-deleted records of all types from the Information Store.
SET_PURGE_SOFT_DELETE_SCHEDULE	SCHEDULE	Sets and activates the schedule on which soft-deleted records are automatically deleted from the Information Store. The parameter must be a string in UNIX cron format. For more information, see the DB2 documentation .
REMOVE_PURGE_SOFT_DELETE_SCHEDULE		Clears and deactivates the automatic deletion schedule.

To do its work, the purging mechanism uses the same infrastructure as the mechanism for [deleting records by rule](#). Each manual or automated request to purge soft-deleted records from the Information Store causes a set of deletion jobs to be created. There is one job for each entity type and link type in the i2 Analyze schema, and the DB2 scheduler runs them at the earliest opportunity.

i2 Analyze keeps logs of manual and automated purge operations alongside its logs of deletion-by-rule operations, in the IS_PUBLIC.DELETION_BY_RULE_LOG view. In that view, the rule name for all jobs that are associated with purging soft-deleted records is PURGE_SOFT_DELETE.

Purging soft-deleted records

Users who delete records from the Information Store through Analyst's Notebook Premium have no way to recover them afterward, but by default they are not completely removed. As the database

administrator, you can decide to leave these "soft-deleted" records in the database, or to purge them manually or automatically.

Before you begin

Purging soft-deleted records from the Information Store requires the same authorization as deleting records by rule. You must connect to the database as a user with the [DELETION_BY_RULE](#) role. The following instructions assume a DB2 command line that you opened as a user with that role.

In addition, if you want to automate purging, the DB2 administrative task scheduler must be enabled on the DB2 server that hosts the Information Store. For more information about enabling the administrative task scheduler, see the IBM i2 Analyze [software prerequisites](#).

About this task

When users delete records from the Information Store through Analyst's Notebook, the data remains in the Information Store (but is inaccessible to users) unless you do something to change that. The following procedures describe how to purge soft-deleted records manually, how to automate that process, and how to understand the effect of a purge operation.

Procedure

To perform a one-off, manual purge of all soft-deleted records from the Information Store:

- Call the `PURGE_SOFT_DELETED_RECORDS` stored procedure from the DB2 command line.

```
call IS_PUBLIC.PURGE_SOFT_DELETED_RECORDS;
```

You can use this procedure regardless of whether you also configure automatic purging.

It immediately creates jobs for purging soft-deleted records of every type.

To set up the Information Store so that soft-deleted records are purged automatically, on a schedule:

- Call the `SET_PURGE_SOFT_DELETE_SCHEDULE` stored procedure from the DB2 command line, passing the schedule that you want to use.

```
call IS_PUBLIC.SET_PURGE_SOFT_DELETE_SCHEDULE('* * 0 0 0');
```

In this example, the Information Store is configured to create the jobs that purge any soft-deleted records every day, at midnight.

To turn off automatic purging, call the `REMOVE_PURGE_SOFT_DELETE_SCHEDULE` stored procedure.

Regardless of whether a purge was started manually or automatically, the effect is always the same: a set of jobs is created to remove soft-deleted records from the Information Store. To inspect the status and outcome of those jobs, you can use a view that the Information Store provides:

- In IBM Data Studio (or similar software), open the `IS_PUBLIC.DELETION_BY_RULE_LOG` view and look for blocks of jobs where the `RULE_NAME` is `PURGE_SOFT_DELETE`.

For more information about the contents of the `DELETION_BY_RULE_LOG` view, see the documentation for [deleting records by rule](#).

Deleting records by rule

If you need to delete records from the Information Store for policy reasons, rather than to reflect delete operations that happened elsewhere, you can use the deletion-by-rule mechanism. You can directly delete any records that you select by specifying their property values and metadata.

Through deletion by rule, you can target records that match specific conditions and automate deletion to occur regularly. Deletion can be applied consistently across all of the records in the Information Store, whether ingested from an external source or created in i2 Analyst's Notebook. You can delete records by their source, their update time, or by any of their property values.

A deletion-by-rule operation has the following components:

- A condition specifies the particular records in the Information Store that you want to delete.
- A rule contains the condition in a form that you can store, together with a name and information about how you want a deletion job to be created.
- A job contains the rule and is placed in a queue to run deletion on the specified records.

For deleting records by rule, IBM i2 Analyze provides the following procedures and views:

- Deletion views are based on the entities and links that are defined in the i2 analyze schema. These views are a simplified representation of the underlying data structures for composing conditions.
- Procedures are supplied for defining and managing deletion rules. If you want a job to be scheduled to run automatically, set the rule to be automated.
- Procedures are supplied for creating deletion jobs based on existing rules.
- Views provide a list of deletion rules, mappings between views and schema types, and a log of deletion jobs, status, and outcomes.

The procedures and views that are associated with deletion by rule are generated automatically when you deploy an Information Store or when you upgrade an Information Store to i2 Analyze version 4.1.7 from an earlier version.

You can construct an SQL query that identifies the data that is to be deleted from the Information Store. When you verify that the correct data is returned in response to the SQL query, you can use that condition to create a rule as a basis for a deletion job. You do not need to have a detailed understanding of the database structure. You do need a knowledge of the i2 Analyze schema. For more information, see the i2 Analyze [data model](#) documentation.

When you want to run a deletion job only once or irregularly, you can create it on a manual basis. You might also have deletion jobs that you want run regularly on automated basis. All automated jobs are created according to the same schedule and placed in a queue of jobs to run in order of their creation time.

Running deletion-by-rule jobs on an automated basis requires the DB2 administrative task scheduler. Ensure that DB2 administrative task scheduler is enabled on the DB2 server that hosts the Information Store. For more information about enabling Administrative Task Scheduler, see the i2 Analyze [software prerequisites](#). You can configure the schedule for deletion-by-rule job creation according to your requirements.

Deleting data by rule

Use an SQL query to identify the data to be deleted, then create and store the deletion rule to identify the corresponding data records for deletion. You can define the rule to specify deletion on an automated schedule or a manual basis.

Before you begin

If you want to apply deletion by rule to a remote Information Store, you must catalog the remote database by using the local IBM DB2 client. For more information, see the first note in [Configuring remote IBM DB2 database storage](#).

About this task

By default, deletion-by-rule privileges are granted to the Information Store database administration user. If necessary, the privileges can be granted to other users by giving them the DELETION_BY_RULE role. For more information, see [Authorization to delete by rule](#). The privileges give you access to all the deletion-by-rule views and procedures, as summarized in the following outline of the main tasks.



CAUTION: Grant the DELETION_BY_RULE role only to users with sufficient knowledge and authority. Exercise caution when you complete deletion-by-rule tasks as they constitute a powerful mechanism for deletion of data that might be difficult to recover. Ensure that there is a reliable backup in place before you delete by rule.

Procedure

1. [Identify the data](#) to be deleted by composing and running an SQL query to select the data from the deletion view.
You can use a different view to see details of the deletion views for your database.
2. Check the SQL results to confirm that the selected data is as you expected.
You can use the condition in subsequent steps to identify the particular records in the Information Store that you want to be deleted.
3. [Create a deletion rule](#) based on the SQL query you verified at step 2 by using the supplied stored procedure.
By default when a rule is created, automated job creation is switched off. You can use another view to see a list of rules and their automation status.
4. [Create a deletion job](#) manually by using the supplied stored procedure.
The procedure creates a deletion job that contains the rule, which is queued to run.
5. Check the status of the job, and that deletion occurred as you expected, by consulting the DELETION_BY_RULE_LOG view.
For more information, see [Verifying deletion by rule](#).
6. Optional: [Automate deletion by rule](#) by using the supplied stored procedure to include the rule in automated job creation.

Identifying the data

How you compose a rule to delete data from the Information Store varies according to the structure and content of your database. The first step is to understand what you need to delete, and then create an SQL statement that selects that data from the deletion views.

About this task

Use the deletion views as a basis to create and validate an SQL statement that selects the data you want to delete. For more information, see [Data deletion views](#).

Procedure

1. Open IBM Data Studio.
2. Connect to your database as a user with the DELETION_BY_RULE role.
3. To see the deletion views, on the Data Source Explorer tab, select **Schemas > IS_PUBLIC > Views**.
4. Optional: To see a summary of deletion view details from your Information Store, complete the following actions.

a) Right-click DELETION_VIEW_LOOKUP.

b) Select **Data > Browse data**.

This view is useful when you know the display name (or the type ID) of the record type you want to target, but not the name of the deletion view.

5. Right-click any of the deletion views.
6. To see the deletion view columns and data, select **Data > Browse data**.

You can discover the criteria that you can use to delete records of this type.

7. Create your SQL statement.

For example, based on the deletion view IS_PUBLIC.E_EVENT_DV, compose a query that returns all event entities where the last update in the source data was more than three years ago.

```
select * from IS_PUBLIC.E_EVENT_DV where SOURCE_LAST_UPDATED < CURRENT_DATE -3  
YEARS
```

8. Verify that the data that is displayed in the SQL Results tab is the data that you want to delete.

What to do next

Create a deletion rule based on your query. For more information, see [Creating a deletion rule](#).

Creating a deletion rule

Use a valid SQL statement as a basis for a deletion rule. Store the corresponding deletion rule for use in the deletion process.

Before you begin

Verify the condition that you need for the deletion rule by creating an SQL query statement that returns the appropriate data. For more information, see [Identifying the data](#).

Procedure

1. Open IBM Data Studio.
2. Connect to your database as a user with the DELETION_BY_RULE role.
3. On the Data Source Explorer tab, select **Schemas > IS_PUBLIC > Stored Procedures**.

4. Right-click CREATE_DELETION_RULE and select **Run**.
5. In the CREATE_DELETION_RULE window, specify the parameters that you need to run the procedure.
 - a) To give the rule a name by which it can be identified, enter a unique name in RULE_NAME.
Rule names are not case-sensitive.
 - b) To identify the database view that the rule targets, enter the name in VIEW_NAME.
Enter the view name without the IS.PUBLIC schema name prefix.
 - c) Enter WHERE clause conditions from your SQL statement in CONDITIONS.
Enter the code that follows the where keyword in the SQL statement.

For example, see the sample parameter values in the following table. Mandatory values are marked with an asterisk.

Table 1: Sample parameter values for creating a rule		
Name	Type	Value
RULE_NAME*	VARCHAR(50)	OLD EVENTS
VIEW_NAME*	VARCHAR(128)	E_EVENT_DV
CONDITIONS	VARCHAR(1000)	SOURCE_LAST_UPDATED < CURRENT_DATE -3 YEARS

6. To run the procedure, thereby creating and storing the deletion rule, select **Run**.

Remember that running this procedure does not execute the rule. However, the rule is validated to inform you of any errors. If the rule has no conditions, you receive a warning message to ensure that you are aware of the consequences when you do create a deletion job based on the rule.



CAUTION: When a job is created from a rule that has no conditions, it deletes every record in the view.

What to do next

To review the rule, select **Schemas > IS_PUBLIC > Views**. Right-click DELETION_RULES and select **Data > Browse data**. For example, see the following rule details.

Table 2: Sample listing of a deletion rule			
RULE_NAME	VIEW_NAME	CONDITIONS	AUTOMATED
OLD EVENTS	E_EVENT_DV	SOURCE_LAST_UPDATED < CURRENT_DATE -3 YEARS	N

By default, the rule is set up so that you can use it to create jobs manually. For more information, see [“Creating a deletion job” on page 9](#). You can also configure the rule so that jobs are created automatically, according to a schedule. For more information, see [Automating deletion by rule](#).

You can update a stored rule by a procedure similar to rule creation, or use a procedure to delete a rule. For more information, see [“Stored procedures” on page 13](#).

Creating a deletion job

Creating a rule stores it with a name that you can use to refer to it later. To delete records that match the rule, you must create a deletion job.

Before you begin

Create a deletion rule. For more information, see [Creating a deletion rule](#).

About this task

When you create a job, the rule is placed in the job queue and initially has a Pending status. At 10-second intervals, i2 Analyze checks for deletion jobs to run.

Procedure

1. Open IBM Data Studio.
2. Connect to your database as a user with the DELETION_BY_RULE role.
3. On the Data Source Explorer tab, select **Schemas > IS_PUBLIC > Stored Procedures**.
4. Run CREATE_DELETION_JOB and enter the name of the rule.
5. To create the deletion job, select **Run**.

Results

The deletion job is created in the queue.

What to do next

You can check the status of the job in the DELETION_BY_RULE_LOG view. For more information, see [Verifying deletion by rule](#).

You can automate the rule for deletion on a schedule. For more information, see [“Automating deletion by rule” on page 10](#).

Verifying deletion by rule

You can verify the outcome of each deletion-by-rule job by consulting the DELETION_BY_RULE_LOG view. The status indicates whether an instance of deletion based on a rule is pending, succeeded, or failed.

Procedure

1. Open IBM Data Studio.
2. Connect to your database as a user with the DELETION_BY_RULE role.
3. On the Data Explorer tab, select **Schemas > IS_PUBLIC > Views**.
4. To verify the status of deletion jobs, right-click **DELETION_BY_RULE_LOG**.
5. Select **Data > Browse data**.
6. Find the rule and job that you are interested in by RULE_NAME and JOB_CREATION_TIME.
7. Verify the status or the outcome of the deletion process in the STATUS column.

What to do next

For more information about the contents of the log, see [“DELETION_BY_RULE_LOG view” on page 15](#).

When the job status is Succeeded, you can browse the data in the deletion view for indications that the effect of the job on the database is as expected. Alternatively, you can run the SQL statement and verify that there are no matching records.

When the job fails, the job status indicates an error. For more information, see [Troubleshooting](#).

Automating deletion by rule

By default, new deletion rules are configured so that you can create deletion jobs manually. You can configure each rule so that deletion jobs are created automatically according to a schedule.

Before you begin

Ensure that the DB2 administrative task scheduler is enabled on the DB2 server that hosts the Information Store. For more information about enabling administrative task scheduler, see [Software prerequisites](#).

About this task

When you automate a deletion rule, it is included in a deletion job that is created by a scheduled task. i2 Analyze processes in sequence all the jobs that are in the queue whether created manually or automatically.

To automate a rule, complete the following steps.

Procedure

1. Open IBM Data Studio.
2. Connect to your database as a user with the DELETION_BY_RULE role.
3. On the Data Source Explorer tab, select **Schemas > IS_PUBLIC > Stored Procedures**.
4. Run AUTOMATE_DELETION_RULE and enter the name of the rule.
5. To automate job creation from the rule, select **Run**.
The AUTOMATED flag for the rule is changed from N to Y.
6. To view the rule details and confirm an automated setting.
 - a) Select **Views**.
 - b) Right-click DELETION_RULES and select **Data > Browse data**.

Results

At the time when automated deletion by rule is scheduled, a job is created for each rule that has been set to automated. Each job is subsequently queued and run in sequence.

You can check the status of the job in the DELETION_BY_RULE_LOG view. For more information, see [Verifying deletion by rule](#).

What to do next

You can configure the schedule for deletion-by-rule job creation according to your requirements. For more information, see [“Changing the automated job creation schedule” on page 16](#).

To disable automated deletion for a rule, run the procedure DISABLE_DELETION_RULE and enter the rule name.

Troubleshooting

When a deletion by rule job is completed, information that indicates success or failure is sent to the DELETION_BY_RULE_LOG view. More details are recorded in the INGESTION_DELETION_REPORTS view.

You can see the status of all deletion-by-rule jobs in the DELETION_BY_RULE_LOG view. For more information, see [Verifying deletion by rule](#).

If the value of STATUS is not Succeeded, the type of error that is recorded informs your diagnosis and troubleshooting steps. The possible status values, explanations, and steps to take for each value are described as follows.

Pending: status unchanged after 10 minutes

1. Check other jobs in the DELETION_BY_RULE_LOG view to see whether there is a backlog due to the number and complexity of jobs; or jobs are not being processed as expected and a significant amount of time has elapsed since the most recent job was processed.
2. Check whether the server is running. You might need to restart the server. For more information, see [Deploying i2 Analyze](#).
3. Check whether the server is connected to the database. There might be a network issue. You can find the Opal server logs here: IBM\i2analyze\deploy\wlp\usr\servers\opal-server\logs

If you are applying deletion by rule to a remote Information Store, the remote database must be cataloged by using the local DB2 client. If the configuration files are not already updated, complete the update by using the toolkit. For more information, see [Configuring remote IBM DB2 database storage](#).

Server error

1. To assess the extent of the problem, in the DELETION_BY_RULE_LOG view, check the REJECT_COUNT value to see the number of items that are rejected.
2. Access the INGESTION_DELETION_REPORTS view for more information. A specific deletion job can be identified by the JOB_ID value, which is identical to the JOB_ID value in the DELETION_BY_RULE_LOG view.

Note: Deletion-by-rule automatically populates the LABEL column in the INGESTION_DELETION_REPORTS view with the value Deletion rule:<Deletion Rule name>.

Deletion views and columns

IBM i2 Analyze deployment automatically generates a set of deletion views in the IS_PUBLIC schema that is part of the Information Store. Using the views, you can browse the data and select by SQL query the records for deletion.

You can access a summary list of the deletion views in your Information Store. The views are created from the i2 Analyze schema when the Information Store is deployed. For more information, see [Identifying data](#).

Use the DELETION_VIEW_LOOKUP view to determine the view names from the display names, and to understand how they relate to the schema type IDs that you might want to use in your rules. The following table contains some examples.

Table 3: Extract from a sample <i>DELETION_VIEW_LOOKUP</i> view		
Schema type identifier	Display name	Deletion view name
ET2	Event	E_EVENT_DV
ET5	Person	E_PERSON_DV
ET8	Communications device	E_COMMUNICATIONS_DEVICE_DV
ET10	Account	E_ACCOUNT_DV
LCO1	Communication	L_COMMUNICATION_DV
LAS1	Associate	L_ASSOCIATE_DV
LAC1	Access To	L_ACCESS_TO_DV

There is a deletion view for each record type. Each of these views has a suffix of *_DV*. Prefixes of *E_* and *L_* are used for entities and links. In IBM Data Studio, you can expand a view to see its columns. Any columns with a *P_* prefix contain values of properties that are defined in the i2 Analyze schema. The other columns contain metadata values that it might be useful to base rules on. Each view has columns for property values, which vary by item type, and columns for metadata values that are common to all records. The following table contains some examples.

Table 4: Sample extract of deletion views and properties from an <i>IS_PUBLIC</i> schema		
Deletion view name	Column name	Sample value
E_PERSON_DV	P_UNIQUE_REFERENCE	P86539K
	P_DATE_OF_BIRTH	1932-10-14
	P_GENDER	Male
	.	.
L_ACCESS_TO	.	.
	.	.
	.	.
	.	.
L_ACCESS_TO	P_UNIQUE_REFERENCE	ADC153
	P_TYPE_OF_USE	Account Holder
	.	.
	.	.
	.	.

Metadata columns can be useful for creating deletion rule conditions. All deletion views have a common set of metadata columns as described in the following table.

Table 5: Metadata columns that are common across deletion views	
Column name	Description
RECORD_ID	An internal identifier that is used by i2 Analyze to identify records
LAST_UPDATE_TIME	The time stamp of the most recent i2 Analyze data ingestion or update
CREATE_TIME	The time that is assigned by i2 Analyze at the point of creation of the record
SOURCE_NAMES	The names of the sources of the data

<i>Table 5: Metadata columns that are common across deletion views (continued)</i>	
Column name	Description
SOURCE_CREATED	The time stamp from the external source for the creation of the data
SOURCE_LAST_UPDATED	The time stamp from the external source for last update of the data

Note: Records that are created through IBM i2 Analyst's Notebook have a value of ANALYST in the SOURCE_NAMES column and null values for SOURCE_CREATED and SOURCE_LAST_UPDATED.

Views that represent link types have a few extra metadata columns as described in the following table.

<i>Table 6: Metadata columns that are specific to link records</i>	
Column name	Description
FROM_RECORD_ID	The identifier of the record that constitutes the start of a link
TO_RECORD_ID	The identifier of the record that constitutes the end of a link
FROM_ITEM_TYPE_ID	The identifier of the type of the record that constitutes the start of a link
TO_ITEM_TYPE_ID	The identifier of the type of the record that constitutes the end of a link

Each deletion view has security dimension columns; one column for each security dimension in the security schema. For more information, see [Security model](#). It can be useful to delete records based on their security dimension values.

Note: The deletion of data is not subject to restriction based on the i2 Analyze security dimension values. But you can filter data for deletion based on these values.

Stored procedures

The i2 Analyze deletion-by-rule functionality depends upon a set of standard stored procedures. You can use these procedures to manage deletion rules and create deletion jobs.

The procedures that are available are described in the following table. An asterisk indicates a mandatory parameter.

<i>Table 7: Deletion by rule: stored procedures</i>		
Procedure	Parameters	Description
AUTOMATE_DELETION_RULE	RULE_NAME*	Set AUTOMATED to Y for the named rule to be included in the scheduled automatic creation of deletion jobs. Deletion-by-rule jobs are placed in the queue of deletion jobs to be run in sequence.

Table 7: Deletion by rule: stored procedures (continued)

Procedure	Parameters	Description
CREATE_DELETION_RULE	RULE_NAME* VIEW_NAME* CONDITIONS	Convert the SQL conditions and named view to a rule that is ready for deletion job creation. To use the rule for a deletion job, see AUTOMATE_DELETION_RULE or CREATE_DELETION_JOB.  CAUTION: When a job is created from a rule that has no conditions, it deletes every record in the view.
DELETE_DELETION_RULE	RULE_NAME*	Delete the named rule.
DISABLE_DELETION_RULE	RULE_NAME*	Set the AUTOMATED parameter to N for the rule to be excluded from the scheduled automatic creation of deletion jobs. The rule can be used for manual deletion only, see CREATE_DELETION_JOB.
CREATE_DELETION_JOB	RULE_NAME*	Create a deletion job that is based on the named rule. This procedure can be run for the rule when the value of AUTOMATED is Y or N. The deletion-by-rule job is placed in the queue of deletion jobs to be run.
UPDATE_DELETION_RULE	RULE_NAME* VIEW_NAME* CONDITIONS	Update the named rule by revising the original VIEW_NAME or CONDITIONS.
VALIDATE_DELETION_RULE	RULE_NAME*	Report on the SQL Results tab any SQL errors that are found associated with the SQL conditions of the named rule. The report contains results by MESSAGE, SQLCODE, and SQLSTATE values.

DELETION_BY_RULE_LOG view

The DELETION_BY_RULE_LOG view provides information on the status of the deletion jobs in the queue in relation to each rule. The view also contains details on the results of each deletion job that is run.

You can access the view to check the status of a deletion-by-rule job. For more information, see [Verifying deletion by rule](#). After the deletion job is run, the DELETION_BY_RULE LOG view contains one or more entries per job that describe the result for each item type. For more information, see [“Sample use cases”](#) on page 17.

The information that is contained in the DELETION_BY_RULE_LOG view is described in the following table.

Table 8: DELETION_BY_RULE_LOG view columns	
Column name	Description
STATUS	Status of the deletion by rule job: Succeeded Pending Server error Validation error
RULE_NAME	Name of the deletion rule.
VIEW_NAME	Name of the view that the rule is targeting.
CONDITIONS	Deletion conditions that are applied by the rule.
JOB_ID	Identifier for the job, which is also used to identify the job in the JOB_ID column of the INGESTION_DELETION_REPORTS view.
JOB_CREATION_TIME	Time stamp that indicates when the deletion by rule job was created.
JOB_RUN_TIME	Time stamp that indicates when the deletion by rule job was run and corresponds to the value in the START_TIME_STAMP column of the INGESTION_DELETION_REPORTS view. There is no value in the JOB_RUN_TIME column while the job is pending.
ITEM_TYPE	Item type of the records that were deleted. There can be more than one entry in this view for the same job. For example, deleting entity records can also cause link records to be deleted.
DELETE_COUNT	The number of records that are deleted.
REJECT_COUNT	The number of records that are rejected.

Example of job results logged

See the following tables for an example of how the DELETION_BY_RULE_LOG view might look with completed job details.

Table 9: IS_PUBLIC.DELETION_BY_RULE_LOG job details (part 1)			
STATUS	RULE_NAME	VIEW_NAME	CONDITIONS
Succeeded	OLD ACCOUNTS	E_ACCOUNT_DV	P_DATE_OPENED < CURRENT_DATE - 50 YEARS
Succeeded	OLD ACCOUNTS	E_ACCOUNT_DV	P_DATE_OPENED < CURRENT_DATE - 50 YEARS

Table 10: Example of IS_PUBLIC.DELETION_BY_RULE_LOG job details (part 2)					
JOB_ID	JOB_CREATION_TIME	JOB_RUN_TIME	ITEM_TYPE	DELETE_COUNT	REJECT_COUNT
77	2017-12-12 14:29:11.351	2017-12-12 14:29:16.467	LAC1	3	0
77	2017-12-12 14:29:11.351	2017-12-12 14:29:16.467	LTR1	3	0

In this example, deleting three Account entity records caused the deletion in the same job of three Transaction link records that were attached to the Accounts.

Changing the automated job creation schedule

i2 Analyze creates jobs from all automated deletion rules according to the same schedule. By default, it creates deletion-by-rule jobs every hour, on the hour. You can update that schedule by running a stored procedure.

The IBM DB2 scheduler runs the task that creates deletion-by-rule jobs at intervals that you can configure. For more information about the scheduler itself, see [The administrative task scheduler](#).

The setting that controls how often the DB2 scheduler creates deletion-by-rule jobs is stored in the Information Store, in UNIX cron job format. Whenever the setting matches the current time, date, and day, the scheduler creates jobs from the deletion rules that have the value of AUTOMATED set to Y.

When you deploy i2 Analyze, the setting receives its default value, '0 * * * *', which means that the scheduler looks for jobs to create every hour, on the hour. For more information, see [UNIX C cron format](#).

To change the schedule for automated deletion after deployment, call the following procedure from the DB2 command line. For example, to run the deletion-by-rule job creation task only at midnight, replace the asterisks as follows:

```
call IS_PUBLIC.UPDATE_DELETION_SCHEDULE('0 0 * * *');
```

Note: The DB2 scheduler checks for new or updated tasks at 5-minute intervals. If you change the setting, be aware of a potential delay of 5 minutes before it takes effect. For example, if it is 8:27 and you change the setting so that jobs are created every half hour, it might still be 9:00 before job creation occurs. You can still create a job from the rule manually if you need to.

Authorization to delete by rule

The deletion-by-rule capability is enabled by the i2 Analyze Information Store deployment. The database administration user who deploys the Information Store is by default authorized to delete by rule.

Authorization to delete by rule includes permission to access and run all of the views and procedures for doing so in the IS_PUBLIC schema. For more information, see [Deletion views](#) and [Stored procedures](#). If you do not have this authorization, you can see but not access or run these views and procedures.

The authorization to delete by rule can be granted to other users. You can create a new database user and assign the DELETION_BY_RULE role to that user, or you can assign the role to an existing user. For example, you can run the following statement.

```
grant role DELETION_BY_RULE to user deletion_user
```



CAUTION: Assign the DELETION_BY_RULE role only to users with sufficient knowledge and authority. Exercise caution when you create deletion rules as they constitute a powerful mechanism for deletion of data that cannot be recovered afterward.

Sample use cases

Stored data must be monitored for compliance with international standards, corporate regulations, or storage volume restrictions. Depending on the use case, different rules can be created to meet many different kinds of requirement.

Reasons for data deletion often fall into one or more of the following categories:

- Data is out of date
- Data is obsolete or irrelevant in the context of the purpose of the database
- Data has data privacy or security issues
- Data has reached a storage volume threshold

The SQL statements in the following examples contain the kinds of conditions that are commonly used as a basis for deletion.

Note: When you are creating SQL for time-based conditions, DB2 provides useful documentation on date and time arithmetic support. For more information, see [Datetime operations and durations](#).

Selecting by property value

Select records of all persons who are below a specific age. The rule matches records of persons who are younger than 18 years. This example is one that you might use to delete data that does not suit the specific purpose of the database.

```
select * from IS_PUBLIC.E_PERSON_DV
where
P_DATE_OF_BIRTH > CURRENT_DATE - 18 years
```

Metadata: data source

Select all records that are sourced from the "DMV" data source.

```
select * from IS_PUBLIC.E_PERSON_DV
where
SOURCE_NAMES like '%DMV%'
```

Metadata: creation date

Use SQL to select any record with a creation date that is not in the current calendar year.

```
select * FROM IS_PUBLIC.E_PERSON_DV
where
YEAR(CREATE_TIME) < YEAR(CURRENT_DATE)
```

Metadata: old data

Select communication records older than 3 months.

```
select * FROM IS_PUBLIC.L_COMMUNICATION_DV
where
CREATE_TIME < CURRENT_DATE - 3 MONTHS
```

Without the specified link

This case is an example where you require deletion of entities that are not linked to other entities. You might want this type of deletion as a general database cleanup, or to target links of a specific type, for example those granting access to a building or an account. Select records of people who do not have any access links. The rule matches records of people when the RECORD_ID is not contained in a FROM_RECORD_ID or TO_RECORD_ID link column.

```
select * from IS_PUBLIC.E_PERSON_DV
where
RECORD_ID not in
(select FROM_RECORD_ID from IS_PUBLIC.L_ACCESS_TO_DV)
```

With the specified links

Select records of any person who is associated with a person named 'Michael Wilson'.

```
select * from IS_PUBLIC.E_PERSON_DV
where
RECORD_ID in
(select FROM_RECORD_ID from IS_PUBLIC.L_ASSOCIATE_DV where TO_RECORD_ID in
(select RECORD_ID from IS_PUBLIC.E_PERSON_DV where P_FIRST_GIVEN_NAME='Michael'
and P_FAMILY_NAME='Wilson'))
```

Data source, property, and time frame

This case is an example of the kind of deletion that could be used to clean up a database or reduce storage requirements. Select records from a specific source based on when the last update occurred.

The rule matches records of people with the named data source and an update date that is older than 3 years from the current date.

```
select * from IS_PUBLIC.E_PERSON_DV
where
SOURCE_NAMES like '%LawEnforcementDB%' and SOURCE_LAST_UPDATED < CURRENT_DATE -3
YEARS
```


Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk
NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM United Kingdom Limited
Hursley House
Hursley Park
Winchester
Hants SO21 2JN
U.K.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, i2, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other names may be trademarks of their respective owners. Other company, product, and service names may be trademarks or service marks of others.

