Version 4 Release 2

*IBM i2 Enterprise Insight Analysis
Data Model White Paper*

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 17.

The i2 Analyze data model

The i2 Analyze data model is the foundation on which the abilities of the platform to analyze and visualize data are based. On the server and in the user interface, i2 Analyze models data in terms of entities, links, and properties (ELP).

- An *entity* represents a real-world object, such as a person or a car.
- A *link* associates two entities with each other. For example, a Person entity might be associated with a Car entity through an Owns link.
- A *property* stores a value that characterizes an entity or a link. For example, a Person entity might have properties that store their given name, surname, date of birth, and hair color.

Every deployment of i2 Analyze includes an implementation of the data model that states exactly what kinds of entities and links can appear in the data for that deployment. This implementation also states exactly what properties the entities and links can have, and defines the relationships that can exist between entities and links of different types.

In i2 Analyze, the *schema* defines this deployment-specific implementation of the data model. Therefore, the schema determines the structure of the Information Store, and the shape that your data must have so that the Information Store can process it. More generally, the schema is the starting point for your work when you import or ingest data into an i2 Analyze deployment, or when you write a connector for i2 Connect.

To create a schema for an i2 Analyze deployment, you must examine the data that is likely to be available for analysis, and understand how that data is used during an investigation. Understanding the aims of your investigations is key to helping you to organize data effectively.

ELP relationships in i2 Analyze

Depending on the nature of the data that you want to process in your deployment of i2 Analyze, you might need to shape it to fit the ELP format. Putting data into ELP format enables many of the analytical functions that i2 Analyze provides.

The simplest ELP relationship involves two entities that are connected with a single link. These kinds of relationships are the building blocks for networks that contain groups and chains of entities with any number of links between them.

In i2 Analyze, a simple relationship that involves two entities, a link, and their properties can be visualized like this example:

First Name: Anna
Last Name: Harvey
DOB: 5/5/74

Owns

Manufacturer: Ford
Model: Mondeo
Year: 2007
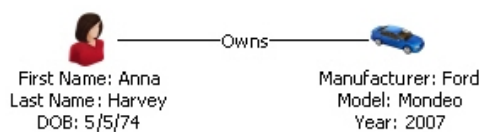
Note: Because of the way that these relationships appear in visualizations, the structure is sometimes called a *dumbbell*.

Some of the information that users see in a relationship like this one comes from the data itself:

- For the entity on the left, the data includes the property values "Anna", "Harvey", and "5/5/74".
- Similarly, for the entity on the right, the property values include "Ford", "Mondeo", and "2007".
- The data for the link includes a way of identifying the two entities that it connects.

The remainder of the information in the example comes from definitions in the i2 Analyze schema:

- The default icons for the entities, and the names ("First Name", "Manufacturer") and logical types of their properties, are all defined in the i2 Analyze schema.
- The default label for the link ("Owns") is also defined in the i2 Analyze schema.

In practice, it can be best to make links lightweight and use intermediate entities to model the details of complex associations. Among other things, this approach allows improved modeling of multi-way associations, such as a conference call that has multiple participants. The following diagram shows the difference:
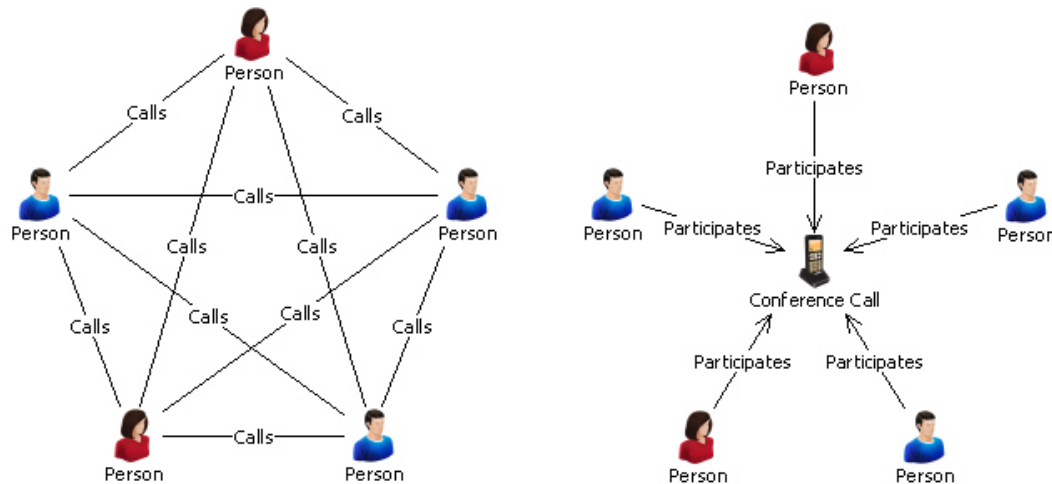
To align your data with an i2 Analyze schema, you must resolve it into the component parts of ELP relationships. If your source is a relational database, it is possible that your tables each correspond to particular kinds of entities, while foreign key relationships form the basis for particular kinds of links. If your source is a delimited text file, it is possible that rows in the file contain the data for one or more entities and links.

Entity types, link types, and property types

The *entity types* and *link types* in the schema for an i2 Analyze deployment determine what entities and links can appear in the data. The *property types* determine what properties the entities and links can have.

Entity types and link types

In an i2 Analyze schema, entity types and link types have similar definitions. Among several common features, entity types and link types both have:

- Identifiers, so that types can refer to each other in rules about their relationships
- Names that users see when they interact with entities and links of particular types
- Definitions of the properties that entities and links of particular types can contain
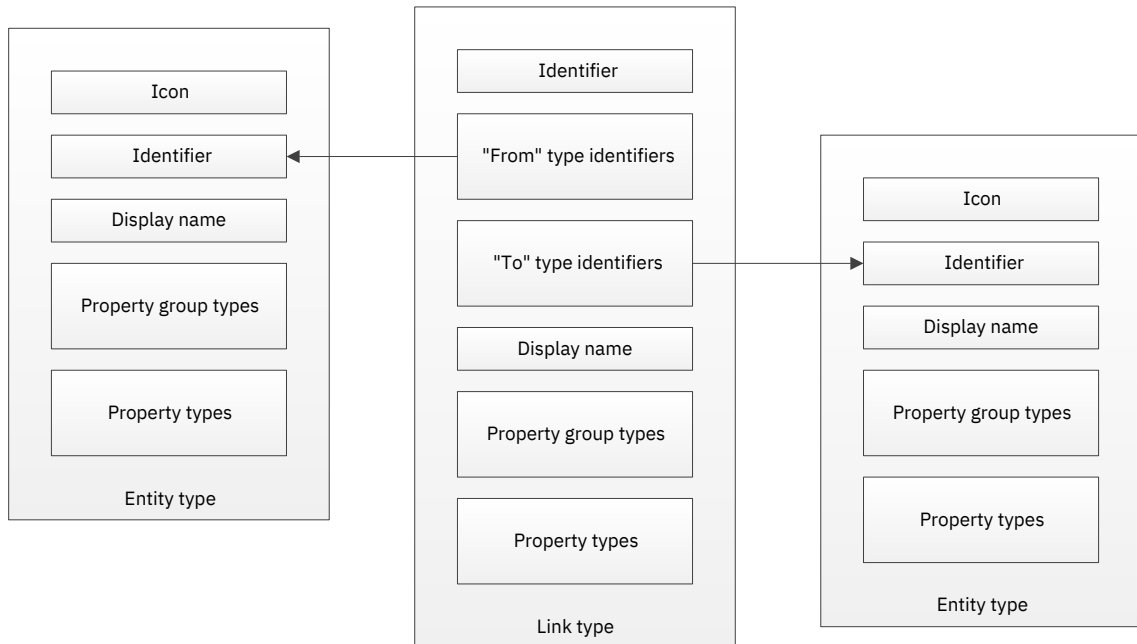
As well as the common features, each *entity* type contains the icon that represents entities with that type in visualizations. *Link* types do not contain icons, but they do contain lists of "from" and "to" entity type identifiers. For a link that has a particular link type, these lists determine what entity types the entities at each of the link can have.

In a valid i2 Analyze schema, the identifiers that appear in the "from" and "to" lists of link types must also appear as the identifiers of entity types.

Property types and property group types

In an i2 Analyze schema, entity types and link types both contain property types. For an entity or a link that has a particular type, the property types specify the names and the logical types of the properties that the entity or link can have.
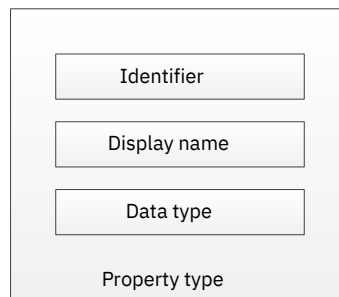
Note: This representation is simplified. A property type can also specify a list of possible values for properties of that type. Furthermore, it can declare whether a property of that type is mandatory for an entity or a link that has the containing type.

In an i2 Analyze schema, property types can appear in an entity type or a link type individually, or as members of a property group type. The purpose of a property group type is to indicate that any properties with the property types that it contains are to be considered as a single unit.

```
┌─────────────────────────────┐
│  ┌───────────────────────┐  │
│  │      Identifier       │  │
│  └───────────────────────┘  │
│  ┌───────────────────────┐  │
│  │     Display name      │  │
│  └───────────────────────┘  │
│  ┌───────────────────────┐  │
│  │                       │  │
│  │    Property types     │  │
│  │                       │  │
│  └───────────────────────┘  │
│      Property group type    │
└─────────────────────────────┘
```
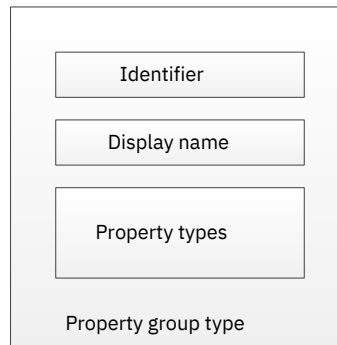
For example, an entity type with the name Person might contain a property group type with the name Body Mark. That property group type might then contain property types with the names Category, Appearance, and Location.
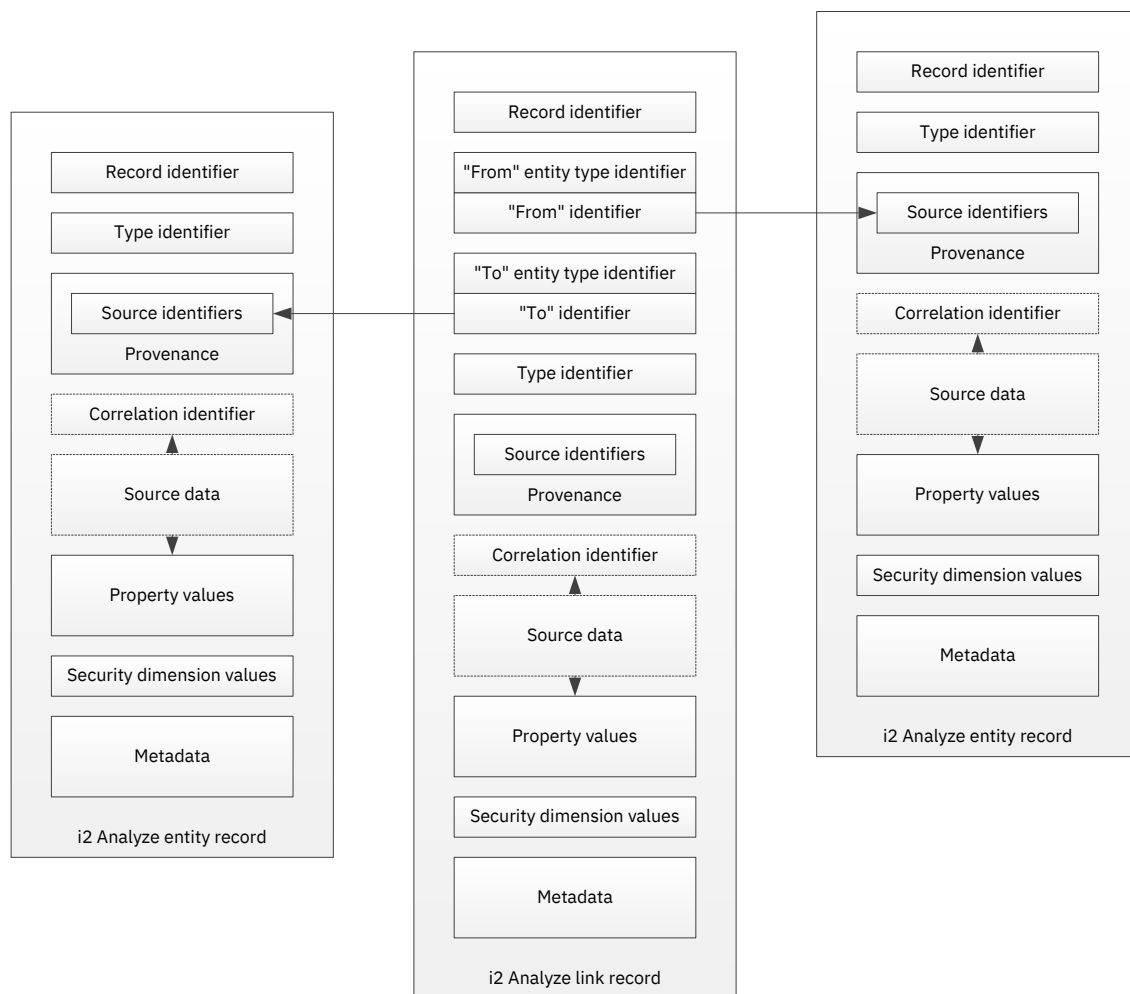
Note: The data stores in an i2 Analyze deployment do not have to provide specific storage for property group types, and the Information Store does not do so. The Information Store implements the data model by treating the property types in a property group type as if the entity type or link type contained them directly.

Data in i2 Analyze records

i2 Analyze deployments that include an Information Store or the i2 Connect gateway use *i2 Analyze records* to realize the schema. i2 Analyze records contain the property data for entities and links, plus the metadata that enhances the analysis that users can carry out.

The schema for an i2 Analyze deployment defines what the i2 Analyze records in that deployment can represent. Every i2 Analyze record has one of the types that the i2 Analyze schema defines. If a record has a link type, then the record represents a link – it is a *link record*. If a record has an entity type, then it is an *entity record*.

This diagram shows how entity and link records compare, and how they are related to each other. It also introduces some other features of the data in i2 Analyze records.

i2 Analyze entity record	i2 Analyze link record	i2 Analyze entity record
Record identifier	Record identifier	Record identifier
Type identifier	"From" entity type identifier	Type identifier
Source identifiers	"From" identifier	Source identifiers
Provenance	"To" entity type identifier	Provenance
Correlation identifier	"To" identifier	Correlation identifier
Source data	Type identifier	Source data
Property values	Source identifiers	Property values
Security dimension values	Provenance	Security dimension values
Metadata	Correlation identifier	Metadata
	Source data	
	Property values	
	Security dimension values	
	Metadata	

Note: The diagram contains some simplifications:

- *Provenance* is about the data sources that contributed to a particular i2 Analyze record. When the property values of an i2 Analyze record represent data from more than one source, that record can have more than one piece of provenance.

- When an i2 Analyze record has more than one piece of provenance, it can contain all the data from all the contributing sources. In that case, the property values that the record presents are derived from the source data.
- *Metadata* includes the following information:
  - Timestamps for each i2 Analyze record
  - *Source references*, which describe the sources that the data in the record came from
  - *Notes*, which users can write to add free-form text commentary to a record

  For link records, the metadata also includes strength and direction information.

As an example of how to represent a simple entity that contains data from one source, consider the following information about a person:

Full name	Anna Harvey
Date of birth	5/5/74
Hair color	Blonde
Eye color	Blue

The following diagram shows one way to represent this information as an i2 Analyze record:
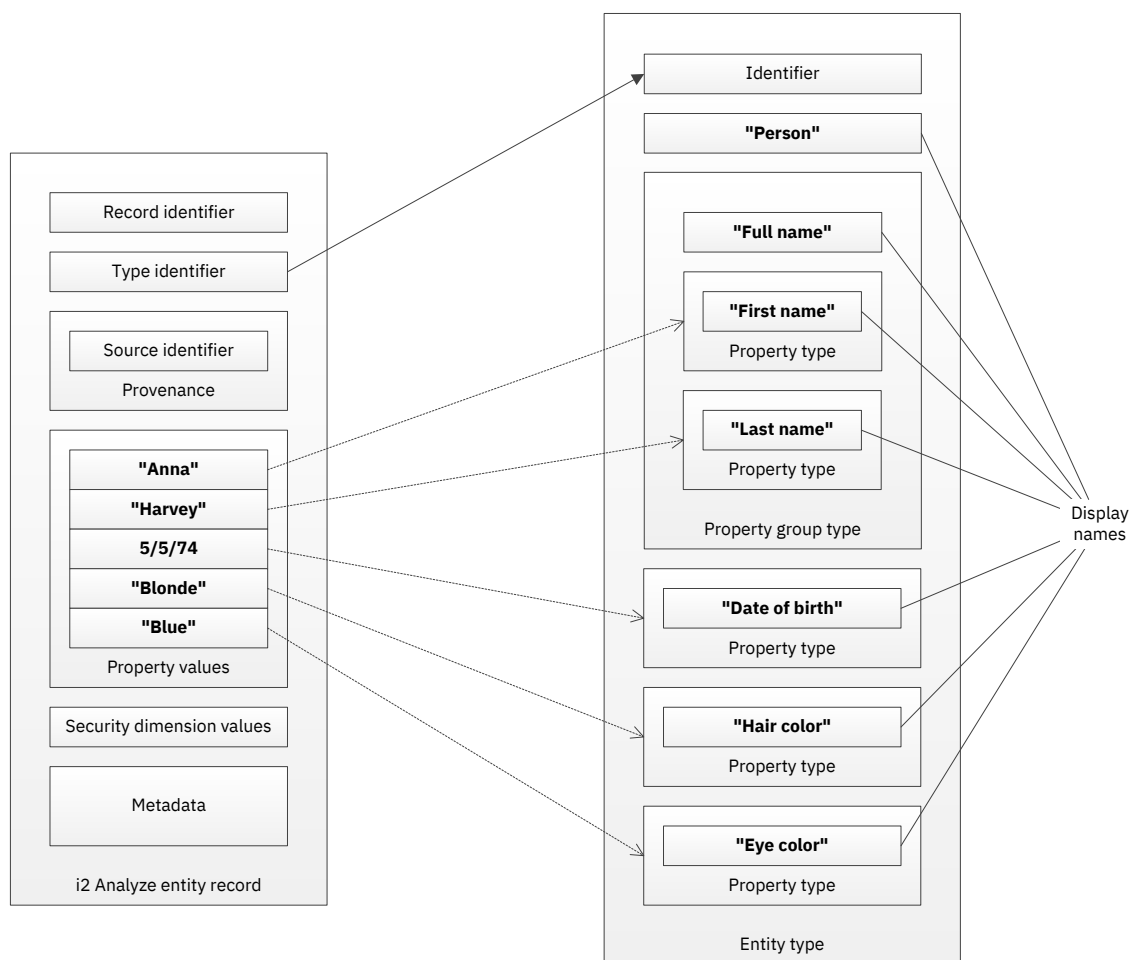
An i2 Analyze entity record can contain properties that have any of the property types that the entity type defines. However, one record can contain only one property of each defined type.

Note: i2 Analyze records do not take account of property group types in the i2 Analyze schema. The Information Store, the i2 Connect gateway, and their client applications interpret all property types in the schema as if they are not contained by property group types.

The diagram also shows how the property types in the schema only partially determine the contents of an i2 Analyze record. Some of the other contents are due to the security schema, while others still are about identification:

- All i2 Analyze records contain security dimension values, which i2 Analyze uses to determine the access level to the record that a particular user has.

- When they enter the system (through an ETL pipeline or Analyst's Notebook Premium), all i2 Analyze records receive a universally unique *record identifier* that is permanent for the lifetime of the record. Assuming that they have the necessary access level, all users of the system can use the record identifier to refer to a particular record.

- All i2 Analyze records contain one or more pieces of provenance. Each piece has a source identifier that references the data for the record in its original source. One record can have provenance from more than one source.

Note: For records in the Information Store that were loaded through ingestion, source identifiers have the additional feature of being unique within the store. These source identifiers are known as *origin identifiers*.

- All i2 Analyze records can contain timestamps in their metadata that specify when source data for the record was created or edited.
- i2 Analyze link records contain an indication of their direction. i2 Analyze considers links to go "from" one entity "to" another. The direction of a link can be with or against that flow, or it can run in both directions or none.

When i2 Analyze records are stored in the Information Store, they contain a few extra pieces of data:

- All i2 Analyze records retain timestamps in their metadata for when they were first created or uploaded to the Information Store, for when they were most recently uploaded, and for when they were last updated.
- All i2 Analyze records can contain a correlation identifier. If two records have the same correlation identifier, the application considers that they represent the same the real-world object and might merge them together.

Your data sources are likely to contain some, but not all, of the data that i2 Analyze records require. To enable the Information Store to ingest your data, or to develop a connector for the i2 Connect gateway, or to write an import specification, you must provide the extra information to i2 Analyze.

Identifiers in i2 Analyze records

i2 Analyze records make extensive use of identifiers. Records refer to their type in the i2 Analyze schema, to their original source data, and to other records in ELP relationships. Preparing data for compatibility with i2 Analyze often involves creating or providing the identifiers that form the basis for the reference mechanisms.

Type identifiers

Every i2 Analyze record contains a type identifier, which is a reference to one of the entity types or link types that the i2 Analyze schema defines. When you create an ingestion mapping file, an import specification, or a connector, you must arrange for each incoming record to receive the identifier of a type definition.

Every i2 Analyze *link* record contains two further type identifiers, which are references to the entity types of the records at the ends of the link. You must arrange for incoming link records to receive these identifiers as well.

This strong typing of records in i2 Analyze is key to the analytical functions that the platform provides. It allows users to consider not only the existence of relationships between records, but also the nature of those relationships. The schema defines exactly what relationships to allow between record types, and i2 Analyze enforces those rules during record creation.

Record identifiers

i2 Analyze records are created when you ingest data into the Information Store, or when a user creates an item on the chart surface that contains an i2 Analyze record by:

- Importing data through an aligned import specification
- Adding the results of an operation against an external source
- Using the Information Store palette in Analyst's Notebook Premium

At creation, every i2 Analyze record automatically receives a universally unique record identifier that is permanent for the lifetime of that record. Users and administrators of an i2 Analyze deployment

can use the record identifier as a convenient way to refer to a record in features such as text search and the Investigate Add-On.

Source identifiers

The role of a source identifier is to reference the data for a record reproducibly in its original source. If a record represents data from several sources, then it contains several source identifiers. The nature of a source identifier depends on the source and the creation method, and sometimes on whether the record is a link or an entity.

For a record that a user creates from the palette or imports through an aligned specification, Analyst's Notebook Premium generates a source identifier automatically.

When you write ingest mappings or develop connectors for the i2 Connect gateway, you are responsible for specifying the identifying information. For example, if the original source is a relational database, then entity data is likely to have ready-made source identifiers: table names and primary keys. Link data can also have ready-made source identifiers, but it might not, especially if the relationship that the link represents exists only as a foreign key. If the source of a record is a text file, then the file name might form part of the source identifier, along with some reference to the data within the file.

Origin identifiers

In general, source identifiers are not certain to be unique within a deployment of i2 Analyze. Several users might independently retrieve the same data from an external source, resulting in several records with the same source identifier. However, when you ingest data into the Information Store, i2 Analyze compares the incoming source identifier with existing records. If it finds a match, i2 Analyze updates a record instead of creating one.

The source identifiers that records receive during ingestion therefore *are* unique within i2 Analyze, and they have a special name in this context. They are called *origin identifiers*.

Correlation identifiers

The purpose of a correlation identifier is to indicate that the data in an i2 Analyze record pertains to a particular real-world object. As a result, correlation identifiers are usually related to property values rather than other identifiers. When two records have the same correlation identifier, they represent the same real-world object, and are candidates to be merged.

When you ingest data into the Information Store, you can provide a correlation identifier for each incoming record. For more information about correlation identifiers and how to create them, see Correlation identifiers.

Security of i2 Analyze records

i2 Analyze records are subject to the i2 Analyze security rules. The security schema defines the security model for your i2 Analyze deployment, and every i2 Analyze record must have at least one value from each security dimension in the schema.

When a user runs a query against the Information Store, i2 Analyze looks up which groups the user belongs to, and determines their security permissions. Then, it compares their security permissions to the security dimension values of the records in the query results. In this way, i2 Analyze calculates which records the user has access to.

If your deployment of i2 Analyze includes an Information Store that you populate through ingestion, then you must add security information to the data during that process. Each time the process runs, you can specify which security dimension values the incoming records receive:

- If you decide that all the data from a given external source must have the same security settings, you can specify the same dimension values for records of all types.
- Alternatively, you can dictate that all the records of a particular entity type or link type receive the same security dimension values.
- You can also configure the process so that each record receives security dimension values that you specify or determine programmatically.

Note: In this version of IBM i2 Analyze, you can change the security dimension values of an ingested record only by ingesting it again with a different set of values.

All other i2 Analyze records receive dimension values when users create them in Analyst's Notebook Premium by importing them, or searching an external source, or using the Information Store palette. These records start with default values that users can edit in the same way that they can edit property values.

Charts in the Information Store

Starting with IBM i2 Analyze 4.3.1 Fix Pack 1, the Information Store can act as a shared repository for Analyst's Notebook charts as well as i2 Analyze records. Users of Analyst's Notebook Premium can search for, download, edit, and upload charts in the Information Store.

To store charts, the Information Store uses an extension of the model for storing i2 Analyze records. Charts in the store automatically benefit from features of records, which include timestamps, notes, and source references. They also use the same security model as the rest of the system, so user access to charts is controlled in the same way as access to records.

The extensions to the model mean that in addition to the properties and notes of a chart, the Information Store indexes the whole of its contents for searching. It also keeps an image of the chart's appearance when it was uploaded. Furthermore, the Information Store retains a version history and performs basic version control operations for the charts it stores.

Note: At this version, the Information Store treats charts as stand-alone entities, and there is no modeling of any connections between charts. Also at this version, the only way to add charts to the Information Store is for users to do it through Analyst's Notebook Premium.

Data in the Analysis Repository

i2 Analyze deployments that include the Analysis Repository and the Intelligence Portal use items to realize the schema. In the Intelligence Portal and the Analysis Repository, the i2 Analyze schema defines what items can represent, and items contain cards that store the data.

In the Analysis Repository, and in any other store whose data you view through the Intelligence Portal, every item has a type that the i2 Analyze schema defines. If an item has a link type, then the item is a link, and the link type definition determines what properties the link can have.

However, items in the Analysis Repository do not contain their properties directly. Instead, properties are stored in *cards*, and the cards are stored in items. All items contain at least one card, and can contain many more.

The following diagram shows how entities, links, cards, and properties are related to each other. It also introduces some other features of the data that the Analysis Repository stores:
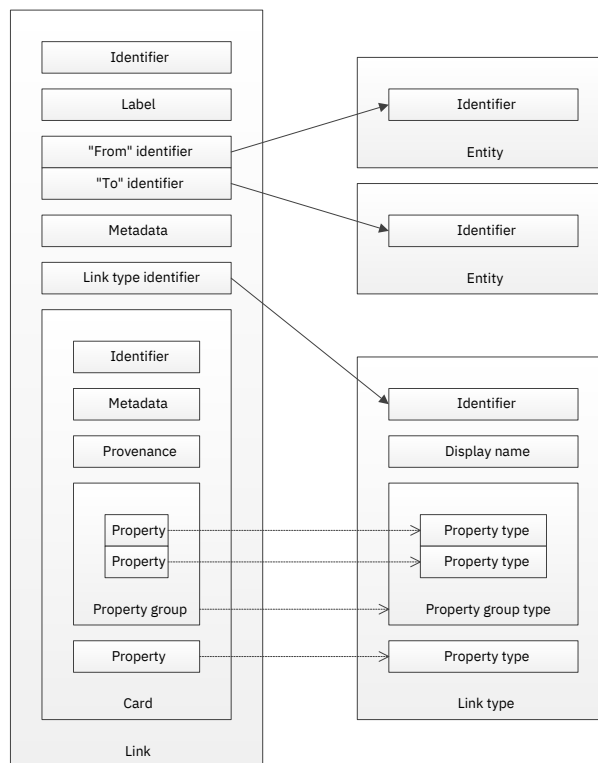

Note: The diagram contains some deliberate omissions and simplifications. For example, an entity can contain everything that a link can contain, except that entities never store "from" and "to" identifiers. In addition, items can contain multiple cards; cards can contain multiple property groups; and property groups (and cards) can contain multiple properties.

As an example of how a simple entity might be represented in the Analysis Repository, look at the following information about a person:

Full name	Anna Harvey
Date of birth	5/5/74
Hair color	Blonde

Eye color   Blue

Considering also that the appearance information might come from a different source than the identity information, the following diagram shows one way to represent the data.
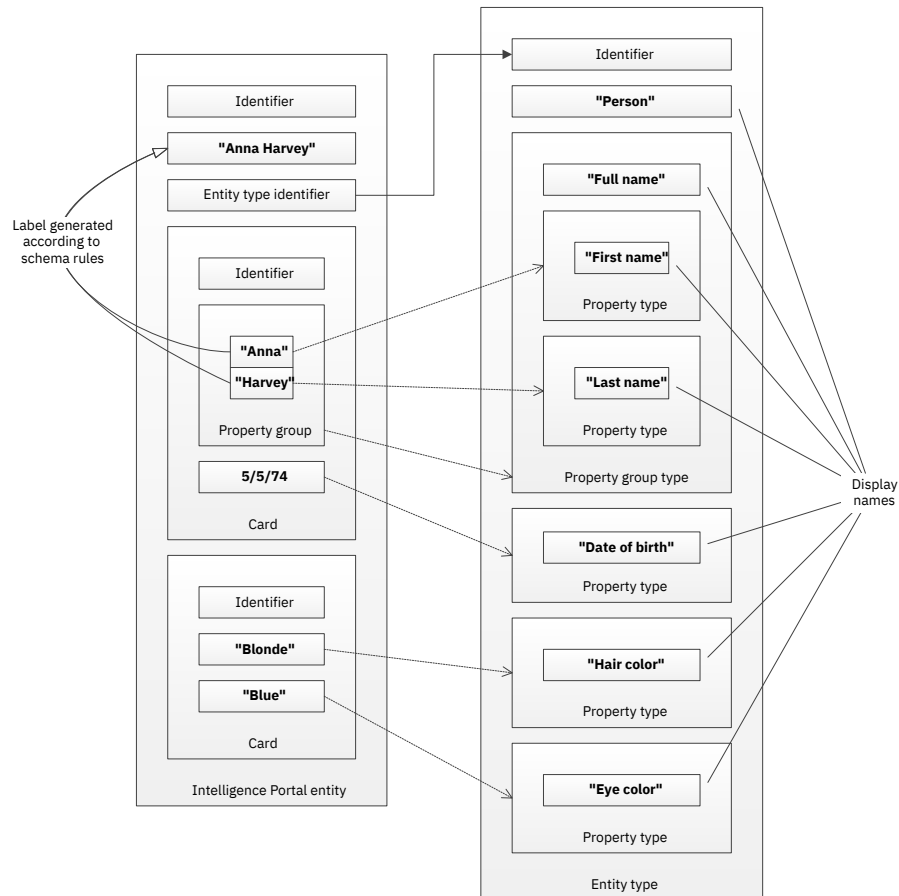
The entity contains values for five properties, two of which are the constituents of a property group. The entity type (named Person) specifies what properties and property groups an entity can have by containing the appropriate property types and property group types.

The entity type does not control how entities use cards, or how many properties of each type the entity can contain. In the Analysis Repository, several properties (or property groups) with the same type can appear on the same card.

In this example, by using two cards, the entity can record how different property values came from different sources. Also, since cards retain their modification dates, this entity provides information about when it was last edited on a per-card basis.

Identifiers in the Analysis Repository

In the Analysis Repository, entities, links, cards, property groups, and properties all have identifiers that make them discoverable uniquely.

The precise nature of the identifiers, and the scope within which the identifiers are unique, are different according to the requirements of i2 Analyze:

- Entities and links have identifiers that are unique at the scope of their data store.
- The identifiers on cards, property groups, and properties are unique at the scope of their containing item. Other items can contain cards, property groups, and properties with the same identifiers.

The Analysis Repository manages all these identifiers internally, and users never need to modify them directly.

Cards in the Analysis Repository

A card in the Analysis Repository contains properties whose values derive from a particular source at a particular time. For example, the name and date of birth of a person might come from one source, while information about their criminal record comes from another.
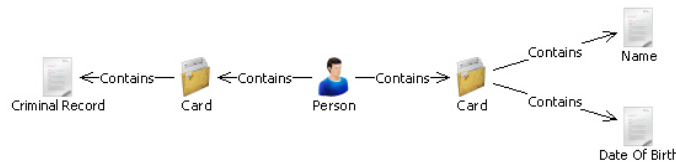
Cards do not have types. Rather, cards act as containers for properties and property groups whose types the item type specifies.

Note: Properties and property groups that have the same type can exist together on one or several cards of the same item. For example, if a person uses aliases, the entity can store multiple given names. If a person has several tattoos, you can use multiple property groups to represent them.

Like the items that contain them, every card has an identifier. A card also has metadata and provenance information that describe when and how the card was created:

- The metadata includes information about when the card was created, and the source from which its property values were obtained.
- The *provenance* is a live record of the data sources from which the data on the card originated. It comprises an identifier for the data in its original source, and a description of how to retrieve the data from its current source.

If a user realizes that two items represent the same object or relationship, they can merge the items without having to overwrite data from either. When necessary, the composite item can contain all the cards from all of its constituent items.

Note: The metadata and the provenance both contain information about the "source" of some data. The meaning of the information changes depending on how the data arrived in i2 Analyze, and how it evolved since then.

| Data origin	Current data location	Metadata "source"	Provenance "source"
Data entry, or manual copying	Analysis Repository	Document or agency that provided the data	Analysis Repository
External data source	Analysis Repository	External data source name	External data source and Analysis Repository
External data source	External data source	External data source name	External data source

If you import data from an external source into the Analysis Repository, the Intelligence Portal can retrieve the most recent version of that data from both places. The provenance retains information about retrieving the data on the card from all of the available sources.

Sets in the Analysis Repository

A set in the Analysis Repository is a deliberate logical grouping of entities and links. A set can represent the contents of a chart, or a collection of entities and links that a user brought together for analysis, or any other assembly of items.

A set contains references to other entities and links, rather than copies of them. More specifically, the set contains references to the *most recent versions* of entities and links. An update to one of the items that a set "contains" does not constitute a change to the set itself.

In the Analysis Repository, an entity can own a set, but a link cannot:
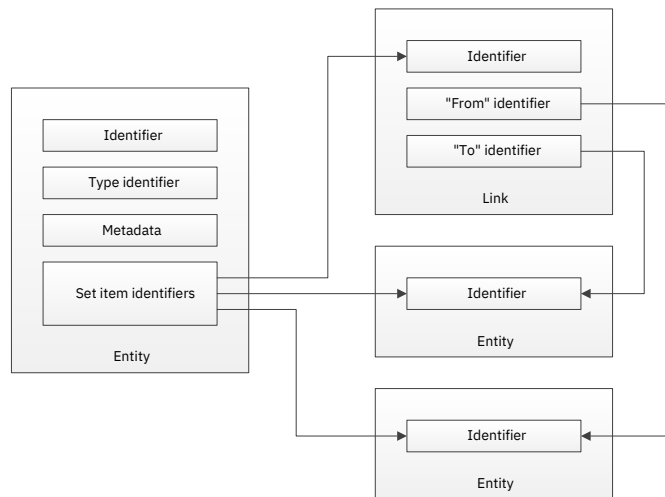
It is conventional to have a "Set" entity type in the schema that contains a limited number of property types. However, the Analysis Repository does not enforce this convention, and entities of any type can own sets.

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
UK

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" 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.

IBM®

Product Number: 5725-G22