2001

EIA i2 Connectors

INTEGRATION INTRODUCTION

BLUE LIGHT LLC

Contents

Introduction

A deployment of IBM’s Enterprise Insight Analysis (EIA) that includes IBM® i2® Analyze with the Opal services can use the i2 Connect gateway to query and retrieve data from external data sources. By implementing a connector to an external data source, one enables i2 Analyze to create and display records that represent the data in that source.

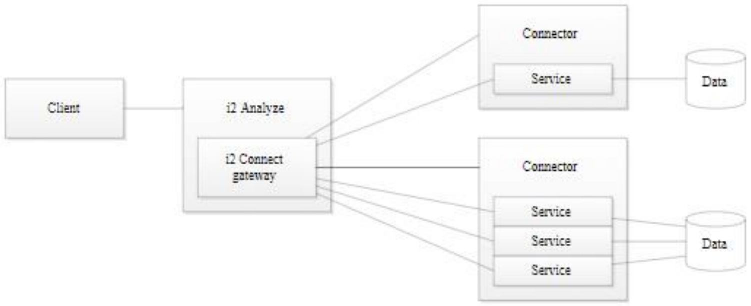
If the deployment and their permissions support it, users who create records from external sources can upload them to the Information Store for storage, sharing, and subsequent analysis.

Solutions that use the i2 Connect gateway depend on *connectors* that one creates to make the bridge between i2 Analyze and external data sources. Conventionally, one writes a connector for each *unique* source that one wants to connect to. Each connector provides one or more *services* that present a querying interface to users and perform the actual queries on the data source.

IBM i2 Analyze and the i2 Connect gateway

At version 4.3.1 of IBM® i2® Analyze, one can provide users with access to an Information Store, or enable them to query data in one or more external sources, or both. i2 Analyze deployments that are targeted at external data access include the i2 Connect gateway alongside or instead of the Information Store.

In a complete solution, the i2 Connect gateway provides i2 Analyze with information about what connectors and services are available, and how clients are to use them. i2 Analyze passes client requests for data to the gateway, which receives and responds to the requests on behalf of those services.



Each connector contains code that interacts with an external data source, the definitions of the services that it supports, and descriptions of how clients might present services to users. For example, clients need to know the following information:

- The name and description of each service
- Whether a service supports parameters that users can specify when they run it
- The input controls to display, and the validation to perform, for each parameter
- Whether a service behaves differently as a result of the current chart selection (any selected records are then seeds for the operation that the service performs)

The role of the gateway is to retrieve this information from all the connectors that i2 Analyze knows about. It can then help to ensure that requests to and responses from services are formatted correctly.

In solutions like these, a connector is an implementation of a REST interface that enables the gateway to perform its role. More specifically, a connector must support the following tasks:

- Respond to requests from the gateway for information about its services
- Validate and run queries against an external data source, including any queries that require parameters or seeds
- Convert query results to a shape that the gateway can process into i2 Analyze records

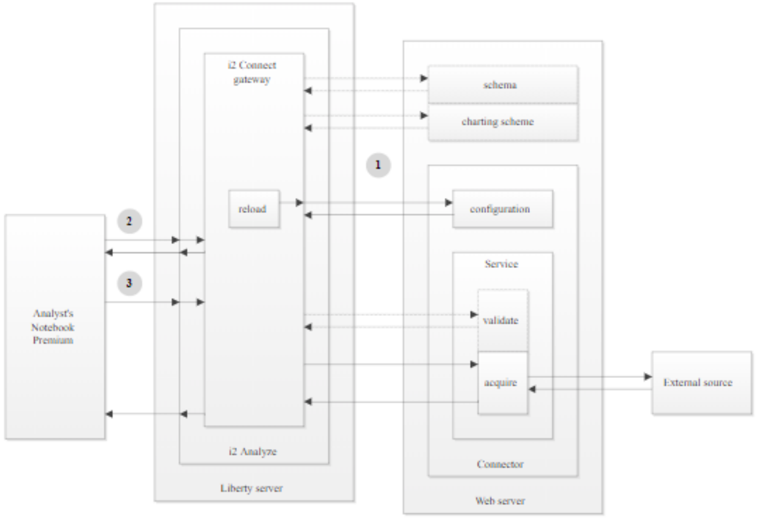
Writing a connector requires development of an implementation of the REST interface that the gateway expects, and to write queries that retrieve data from an external source. One must also understand the i2 Analyze data model, and be able to convert the retrieved data so that it matches the schema that a particular deployment of i2 Analyze supports.

System architecture with the i2 Connect gateway

The simplest possible deployment of IBM® i2® Analyze that uses connectors to access an external data source has four parts. The client, Analyst's Notebook Premium, communicates with a deployment of i2 Analyze that includes the i2 Connect gateway. The gateway manages at least one connector, which in turn exchanges data with an external source.

System interactions

The following diagram represents the deployment that is described in the introduction, and the interactions that take place between each part.



The diagram also identifies five REST endpoints that connectors can implement and use:

Configuration endpoint

The gateway sends a request to the mandatory configuration endpoint to gather information about all the services that the connector supports. All connectors have at least one service, and it is a service that implements the validate and acquire endpoints.

Schema endpoint

The schema endpoint is optional, and so is where it is implemented. If one or more of the connectors can return results whose types are not in the i2 Analyze schema, one might be able to write **schema fragments** that define those types. The configuration specifies whether and where a schema endpoint is available.

Charting scheme endpoint

The charting scheme endpoint is optional, and closely associated with the schema endpoint. A connector configuration that specifies a schema endpoint also specifies a charting scheme endpoint.

Validate endpoint

The validate endpoint is optional. If the configuration endpoint says that a particular service supports it, then the gateway sends a request to the validation endpoint immediately before a request to the acquire endpoint. The purpose of the validate endpoint is to test whether a request is reasonable before it is run.

Acquire endpoint

The acquire endpoint is mandatory for all services. The gateway sends a request to the acquire endpoint that tells the service to query the external source. The service receives data back from the source and places it into the response that it returns to the gateway.

Interaction sequence

The interaction between the i2 Connect gateway and the other parts of an i2 Analyze deployment takes place in three distinct phases, labeled 1, 2, and 3 in the diagram.

1. When i2 Analyze starts up, the i2 Connect gateway sends a request to the configuration endpoint of every connector that is listed in the topology. It caches the information about the connectors and services that it receives in response. If the configuration for any of the connectors specifies schema and charting scheme endpoints, and if the deployment supports it, the gateway sends requests to retrieve fragments from those endpoints.

2. When a client connects to i2 Analyze, the latter gets the cached information from the gateway and returns it to the client. The client can then present the queries that the services implement to users, and help users to provide valid parameters and seeds to those queries where appropriate.
3. When a user runs a query, i2 Analyze passes it to the gateway for processing. The gateway packages the query into a request to the acquire endpoint on the

service in question, preceded by a request to the validate endpoint if the service supports it.

4. On receiving the response from the acquire endpoint, the gateway converts the data that it contains into results that i2 Analyze can ultimately return to the client.

Understanding the external source

Data is only valuable in i2 Analyze if users can examine it alongside, and with the same tools as, all the other data in the system. Just like importing data, connecting to an external source requires one to align its contents with the i2 Analyze schema.

Before writing a connector, one must understand what data is available from the external source, and how that data maps to the types in the schema. If no straightforward mapping exists between the data in the source and one or more of the entity or link types, then one might need to extend the i2 Analyze schema.

Typical Procedure

1. Consider the type of the source and how it can be queried.
    a. For spreadsheets and text files, it is likely that one has complete access to the data they contain. For databases and web services, one might be restricted to a set of predefined queries.
2. Identify the types of data that one can retrieve from the source, and compare them with the types in the i2 Analyze schema for the deployment.

    For each type in the source, one can reach one of three conclusions:
    a. The type and its data are a match for (or a subset of) one of the entity or link types in the schema
    b. The type is a match for one of the types in the schema, but its data implies new property types
    c. The type is not currently represented in the schema
3. If the i2 Analyze schema already describes all the data in the external source, then one can start the process of writing queries that retrieve data from it.
4. If the data in the external source contains property types that the i2 Analyze schema does not describe, one can either ignore that data or edit the schema to add those property types.
5. Additive changes to the schema are always permitted.
6. If the data in the external source contains item types that the i2 Analyze schema does not describe, then one must use one of the techniques for adding them to the schema.