Configuring Data Sources for Cards Components

Configure a data source on a card, layout, or flyout to retrieve data from a Salesforce object or an external database. Display the returned data on cards in a layout or flyout.

Cards can have data sources independent of the layout or flyout data source.

## **Data Source Types**

Data sources available for card, layouts, and flyouts include:

| **Data Source Type** | **Description** | **Availability** |
| --- | --- | --- |
| [**Integration Procedures**](https://docs.vlocity.com/en/Configuring-an-Integration-Procedure-Data-Source.html) | Uses a Vlocity Integration Procedure to return data. Integration Procedures are declarative, server-side processes that execute multiple actions in a single server call. See [Integration Procedures](https://docs.vlocity.com/en/Integration-Procedures.html). **Important** Vlocity recommends using a Vlocity Integration Procedure as a data source for optimal flexibility and easier implementation. | Cards and Layouts |
| [**Apex Remote**](https://docs.vlocity.com/en/Configuring-an-Apex-Remote-Data-Source.html) | Uses an Apex Remote class and method to return data. An Apex Remote class is a standard Apex class that implements the VlocityOpenInterface. | Cards and Layouts |
| [**Apex REST**](https://docs.vlocity.com/en/Configuring-an-Apex-REST-Data-Source.html) | Uses a REST endpoint of an Apex class to return data. | Cards and Layouts |
| [**DataRaptor**](https://docs.vlocity.com/en/Configuring-a-DataRaptor-Data-Source.html) | Uses a Vlocity DataRaptor Extract interface to return data. Field-level security is fully supported. See [DataRaptors](https://docs.vlocity.com/en/DataRaptor.html). | Cards and Layouts |
| [**Dual**](https://docs.vlocity.com/en/Configuring-Dual-Data-Sources.html) | Combines Apex Remote and Apex REST technologies to support hybrid applications available for both web and mobile usage. For example, the Salesforce web environment uses Apex Remote technology while its parallel mobile App uses the Apex REST API, requiring configuration for multiple environments. | Cards and Layouts |
| **Parent** | On the card or card layout of a flyout, the Parent data source uses data from the parent card or card layout that a flyout is assigned. See [Creating a Flyout](https://docs.vlocity.com/en/Creating-a-Flyout.html). | Flyouts |
| **REST** | Uses the standard REST API call to return data. The Vlocity Rest types include:   * [**Web**](https://docs.vlocity.com/en/Configuring-a-REST-Web-Data-Source.html): Standard request from the URL of a callout endpoint. * [**Named Credentials**](https://docs.vlocity.com/en/Configuring-a-REST-Data-Source-Using-Named-Credentials.html): Specifies the URL of a callout endpoint and its required authentication parameters in one definition.  **Note** Use +Add Header Variables to add any request headers. Most third-party APIs expect various headers such as tokens, public keys, and content-type. | Cards and Layouts |
| [**SOQL Query**](https://docs.vlocity.com/en/Configuring-a-SOQL-Query-Data-Source.html) | Uses a [Salesforce Object Query Language (SOQL)](https://developer.salesforce.com/docs/atlas.en-us.soql_sosl.meta/soql_sosl/sforce_api_calls_soql.htm) to search an org's Salesforce data for specific information. For example, SELECT Name,Id FROM Account LIMIT 5. **Note** If security is a concern, use a DataRaptor instead of a SOQL query, because DataRaptors fully support field-level security. | Cards and Layouts |
| [**SOSL Query**](https://docs.vlocity.com/en/Configuring-a-SOSL-Search-Data-Source.html) | Uses a [Salesforce Object Search Language (SOSL)](https://developer.salesforce.com/docs/atlas.en-us.soql_sosl.meta/api/sforce_api_calls_sosl.htm) to construct text-based search queries against the search index. | Cards and Layouts |
| **Sample** | Uses sample JSON to set up a card with a temporary data source, which will eventually be replaced with another data source. This allows you to embed the custom JSON directly into the layout without depending on external data. | Cards and Layouts |
| **Streaming API** | Uses the [Salesforce Streaming API](https://developer.salesforce.com/docs/atlas.en-us.api_streaming.meta/api_streaming/intro_stream.htm) to push notifications from the server to the client based on criteria that you define. The Vlocity Streaming API types include:   * [**Streaming Channel**](https://docs.vlocity.com/en/Configuring-a-Streaming-Channel-Data-Source.html): Sends notifications of general events that are not tied to Salesforce data changes. * [**PushTopic**](https://docs.vlocity.com/en/Configuring-a-Push-Topic-Data-Source.html): Uses an SOQL query to return a result that notifies listeners of changes to records in a Salesforce organization. * [**Platform Event**](https://docs.vlocity.com/en/Configuring-a-Platform-Event-Data-Source.html): Uses [Salesforce's Platform Events](https://developer.salesforce.com/docs/atlas.en-us.platform_events.meta/platform_events/platform_events_intro.htm) enterprise messaging platform, which combines PushTopic and Streaming Channel to enable apps to communicate inside and outside of Salesforce. | Cards and Layouts |