



TRAILHEAD

OmniStudio Student Take Home Guide

Job Aids

Version 1.0.0

TABLE OF CONTENTS

FlexCards Elements Reference	3
FlexCards Element Properties Reference	4
LWC OmniScript Element References	22
Integration Procedure Element References	29
Data Decision Flowcharts	33
FlexCard Data Best Practices	34
OmniScript: Data Outputs	36
OmniScript: Data Outputs - DocuSign	37
DataRaptor: Functional Venn Diagram	37
Integration Procedures: Data Input	38
OmniStudio Naming Conventions	40
Troubleshooting Guide	43

FlexCards Elements Reference

Element	Function
Action	Renders text or a button that executes an action when clicked.
Block	Enables grouping elements inside a collapsible container.
Chart	Displays data as a chart.
Custom LWC	Embeds a Custom LWC inside a FlexCard state.
Datatable	Creates a tabular structure from the data provided.
Field	Displays the output from a data field.
FlexCard	Embeds a FlexCard inside a state.
Icon	Displays a custom or Salesforce SVG icon. The icon can be linked to an action.
Image	Displays a custom image from a given URL. Upload an image, use an image from your org's library, or use an external source. The image can be linked to an action.
Menu	Displays a list of actions as a dropdown menu.
State	Adds a state to a FlexCard.
Text	Renders text and parses merge fields with a rich text editor.
Toggle	Trigger an action when a user selects a Toggle element. The value is sent as a parameter to the action as true or false, or a comma-separated list of user-selected options.

FlexCards Element Properties Reference

Action Properties

Property	Description
Common Action Type Properties	Properties shared by all Action Types
Action Type	Select the type of action to execute.
Conditions	Add a condition that must be met for the element to display.
Display As Button	Check to display the action as a button.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Enable Tracking	Enable click-based event tracking through OmniAnalytics. Track the UI Action event.
Hide Icon	Check to hide the icon and only show the label.
Icon	Enter the name of the icon, as such as <code>utility:down</code> , or click into the field to search for an icon.
Label	Enter a visible label for the element.
Show Only Icon	Check to show an icon without a label.
Variant	Select from predefined SLDS styles to change the appearance of the element.
Card Action	Properties unique to the Card Action
Set Values	Enter key/value pairs to update field values.
Type	Select the type of action to launch. <ul style="list-style-type: none">● Reload – Reloads the FlexCard.

- Remove – Removes all or specific records from the FlexCard, such as alerts and notifications.
- Set Values – Updates a field value.
- Update Data Source – Changes the data source, or updates the parameters of an existing data source.

Event Action	Properties unique to the Event Action
Bubbles	Check to enable the event to bubble up through the DOM.
Channel Name	Enter a channel name for the pubsub event.
Composed	Check to enable the event to pass through the shadow boundary.
Event Name	Enter the name of the custom event.
Event Type	Select an event type.
Input Parameters	Enter parameters to pass contextual data to the event as key/value pairs.
Flyout Action	Properties unique to the Flyout Action
Attributes	Set attributes available from the flyout's component.
Data Node	To pass parent data to the Child FlexCard, select a data node.
Flyout	Select the component to display in the flyout.
Flyout Type	Select the type of component to display.
Layout Type	Select a theme for the flyout.
Open Flyout In	Select the flyout's container type. When enabling a flyout as a clickable event on an element such as a Block element, this feature defaults to modal and is disabled because popover positioning cannot be controlled.

Reset Component On Close	Resets the component to its on load state when the flyout closes.
--------------------------	-------------------------------------------------------------------

Navigate Action	Properties unique to the Navigate Action	Target
Action Name	Enter the name of the action.	App
Api Name	Enter the API name of the app.	App
App Target	The app that you are navigating to. Pass either the appId or appDeveloperName to the appTarget.	App
App Type	The standard or custom app available from the App Launcher.	App
Article Url	The value of the urlName field on the target KnowledgeArticleVersion record. The urlName is the article's URL.	Knowledge Article
Community Page Name	The unique name of the Lightning community page. The value for name is the API Name value for a supported page.	Community Named Page
Component Name	The Lightning component name in the format <code>namespace__componentName</code> .	Component
Custom Tab Name	The unique name of the custom tab.	Navigation Item
Input Parameters	Enter parameters to pass contextual data to the action as key/value pairs.	Component, Login, Navigation Item
Object API Name	The API name of the standard or custom object. For custom objects that are part of a managed package, prefix the custom object	Object



	with <code>ns__</code> , where <code>ns</code> is your org's namespace.	
Open Target In	Select whether to open the target page in the current window or a new tab/window.	All
Page Name	The unique name of the page.	Named Page
Record Object API Name	The API name of the record's object.	Record
RecordId	Enter the record Id of the record to navigate to.	App
Relationship Object API Name	The API name of the object that defines the relationship.	Record Relationship
Target	Select the PageReference type.	All
Target Action	Enter a valid action name to call.	Login, Object, Record, Record Relationship
Target Article Type	Enter the API name of the article type.	Knowledge Article
Target Id	Select a ContextId from the data source.	Record, Record Relationship
Target Relationship	The API name of the object's relationship field.	Record Relationship
URL	The URL of the page you are navigating to.	Web Page
Update OmniScript Action	Properties unique to the Update OmniScript Action	
Input Parameters	Enter the OmniScript data fields to update.	

Parent Node	Point to a specific data node in OmniScript JSON to update. Select from the merge fields available from the data source or enter a data node manually.
OmniStudio Action	Properties unique to the OmniStudio Action
Action Name	Enter a visible name for the action.
Context ID	Select a data field to use as the ContextId, such as <code>{'AccountId'}</code> .
Fetch Action Name from Datasource	Select the action name from a data source field.
Input Parameters	Enter parameters to pass contextual data to the action as key/value pairs.
Object	Select the sObject.
Open Target In	(Disabled by default) Opens target in the current window or a new tab/window.

FlexCards Block Properties

Property	Description
Add Action	Make the element a clickable action.
Collapsed By Default	If selected, the block element is collapsed by default.
Collapsible	Makes the block collapsible, hiding its content.
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Label	Displays a visible label for the collapsible block element.

FlexCards Chart Properties

Property	Description
Color Palette	Select a predefined color scheme.
Conditions	Add a condition that must be met for the element to display.
Cutout Percentage	Enter the size of the donut chart hold by percentage.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Hide Header	Show or hide the chart's title.
Label Node	Select the field whose values are used to categorize the data, such as <i>Name</i> of Accounts.
Title	Displays a title above the chart when Hide Header is disabled.
Type	Select the type of chart to display.
Value Node	Select the values to populate the chart, such as <i>AnnualRevenue</i> for Accounts.

FlexCards Custom LWC Properties

Property	Description
Attributes	Set the values for attributes defined in the selected custom LWC.
Attribute	Enter the name of the attribute defined in the selected custom LWC.
Conditions	Add a condition that must be met for the element to display.
Custom LWC Name	Select the custom LWC to embed.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Value	Enter the value of the Attribute .

FlexCards Datatable Properties

Property	Notes
Column	Update the data fields to populate the table and set options for columns, such as type, selectability, and so on.
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Records	Specify the record data that populates the table. Defaults to <code>{records}</code> .

FlexCards Field Properties

Property	Description
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Field Type	Select the type of Field element to display.
Label	Enter a visible label for the element.
Output	Select the data field to display.
Placeholder	Enter placeholder text.
Field Type Properties	
Currency	Enter a currency code to update the supported currency.
Format	Select a format for the date, or date and time if the Field Type is Date Time .
Locale	Enter a locale code to update the supported language.
Mask	Enter a string expression to limit the user's input.

(Child) FlexCard Properties

Property	Description
Attribute	Define a key that represents an attribute.
Attributes	Pass data from the parent to the Child FlexCard by creating an attribute and assigning it a value, such as a data merge field.
Conditions	Add a condition that must be met for the element to display.
Data Node	To use the data source from the parent FlexCard, select a data node.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Enable Tracking	Enable event tracking through OmniAnalytics. Track Card Load, Card Unload and State Load events.
FlexCard Name	Select the Child FlexCard to display.
Select State	View a state from the FlexCard Name selected.
Value	Select the data to pass for the defined Attribute .

Icon Properties

Property	Description
Add Action	Make the element a clickable action.
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Extra Class	Add an additional class to the element.
Icon	Enter the name of the Salesforce Icon.
Image Source	Enter the URL of a custom icon.
Size	Select the size of the icon.
Type	Select the type of icon to display.
Variant	Select from predefined SLDS styles to change the appearance of the element.

FlexCards Image Properties

Property	Description
Add Action	Make the element a clickable action.
Alternative Text	For accessibility, enter text describing the image, which displays when the image is not available.
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Extra Class	Enter a class appended to the container of the element. Select from a class defined in the Add Custom CSS feature of the FlexCard Designer or any stylesheet available to the FlexCard.
Height	Enter a value available to the <i>height</i> CSS property, such as <i>100px</i> , <i>4rem</i> , or a <i>calc()</i> function.
Image Source	Enter the image source for the icon. Upload the icon here, select an image from your org's libraries, or enter a secure absolute URL from an external source.
Size	Update the size of the image relative to the Canvas size. By default, Fit Content is selected, enabling the image to fill the container of the image element.
Title	Enter an image caption.
Title Class	Enter a class for the image caption.
Width	Enter a value available to the width CSS property, such as <i>100px</i> , <i>4rem</i> , or a <i>calc()</i> function.

FlexCards Menu Properties

Property	Notes
Add New Action	Make the element a clickable action.
Conditions	Add a condition that must be met for the element to display.
Dropdown Position	Select the position of the dropdown relative to the menu button.
Dropdown Size	Select the size of the dropdown.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Enable Overflow	To disable scrolling in the dropdown when there are more than 5 menu items, disable this feature. Enabled by default.
Icon	Select the icon to display.
Icon Position	Select position of the icon relative to the label.
Icon Size	Select the size of the icon.
Label	Enter a visible label for the element.
Status	Select an SLDS style for your menu item.
Variant	Select from predefined SLDS styles to change the appearance of the element.

FlexCards State Properties


Property	Notes
Blank Card State	Display this state when there is no data returned from a data source.
Name	Enter a descriptive name for the state. The state Name is for reference only and does not appear on the generated LWC or in Preview.
Conditions	Set when and how a state displayed based on specified conditions.

FlexCards Text Properties

Property	Notes
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.

FlexCards Toggle Properties

Property	Description
Common Toggle Element	Properties shared by all Toggle element types
Add Action	Add an action that gets triggered when a user clicks the toggle element.
Conditions	Add a condition that must be met for the element to display.
Element Name	The name of the element as seen on the Canvas. Used to distinguish one element from another as you build your FlexCard in the designer. Is used only within the designer.
Extra Class	Enter a class appended to the container of the element. Select from a class defined in the Add Custom CSS feature of the FlexCard Designer or any stylesheet available to the FlexCard.
Toggle Type	Set the type of toggle element to display.
Toggle Element Properties	
Alignment	Select whether to align each checkbox horizontally or vertically.
Alternative Text	Enter text for accessibility, visible when image is not present.
Checked	Set the toggle element to be checked by default.
Checked Icon Name	Select the icon visible when the element is checked.
Checked Label	Select the label visible when the element is checked.
Disabled	Disable the toggle element. Or disable the toggle element based on the value of a data field whose value returns true or false. The field must have a Boolean value. If the value returns a string, Disabled is false.

Help Text	Add text visible when user hovers over the help  icon.
Icon Name	Select the icon.
Icon On Hover	Select the icon visible on hover of the element.
Label	Enter a visible label for the element.
Label On Hover	Select the label visible on hover of the element.
Options	Enter a key/value pair for each checkbox option.
Required	Make the toggle element a required field. Adds an asterisk next to the element.
Required Label	Add text next to the asterisk when Require is enabled.
Unchecked Icon Name	Select the icon visible when the element is unchecked.
Unchecked Label	Select the label visible when the element is unchecked.
Update Data Source	Updates the data source with the value of the selected state of the toggle element.
Value	Add default value. For example, set a value as true or enter a comma-separated list of selected options.
Variant	Select from predefined SLDS styles to change the appearance of the element.

LWC OmniScript Element References

Element	Function
Actions	
Calculation Action	Call a Calculation Procedure and return the results to the Integration Procedure. This enables both DataRaptors to run in a single Apex call.
DataRaptor Extract Action	Call a DataRaptor Extract.
DataRaptor Post Action	Call a DataRaptor Load.
DataRaptor Transform Action	Call a DataRaptor Transform.
DataRaptor Turbo Action	Invoke a DataRaptor Turbo to retrieve data from a single Salesforce object. Doesn't support formulas or complex field mappings.
Delete Action	Enables users to delete Salesforce records. Use an Object's Record Id to determine which record to delete. Recommended to use a merge field in the Path to Id field that refers to an Id or a list of Ids in the data JSON.
DocuSign Envelope Action	Send DocuSign emails for signature. <ul style="list-style-type: none">• Uses DocuSign Template.• Requires DataRaptor Transform.
DocuSign Signature Action	Sign documents with DocuSign. <ul style="list-style-type: none">• Uses DocuSign Template.• Requires DataRaptor Transform.
Email Action	Populate and send emails. <ul style="list-style-type: none">• Salesforce email template.
HTTP Action	Get, save or delete data through a web API. <ul style="list-style-type: none">• Methods available: GET, POST, DELETE, PUT

Matrix Action	Enables retrieval of a value from a calculation matrix by specifying the required input parameters.
Navigate Action	Enables the OmniScript to open different Salesforce experiences and external endpoints. (Redirects the user at the end of the OmniScript.)
PDF Action	Fill a PDF form. <ul style="list-style-type: none"> • Must have a DataRaptor Transform.
Integration Procedure Action	Call an Integration Procedure.
Remote Action	Do anything that can be done with an Apex class. <ul style="list-style-type: none"> • Calls an Apex class and method. • May need a DataRaptor Transform.
Set Errors	Apply error or validation messaging to elements in a previous step based on inputs of a future step.
Set Values	Set default values. Simple transforms, functions and mapping within the OmniScript.
Display	
Line Break	Line break for formatting purposes.
Text Block	Display text, images, links, and data.
Functions	
Aggregate	Perform aggregation functions.
Formula	Perform functions and algebraic calculations.
Messaging	Display a banner message based on a condition.
Groups	
Block	Group elements together within the page/view.
Action Block	Groups Action elements together to enable backend calls to fire asynchronously using the same configuration

Edit Block	Add, edit or delete records in real time. <ul style="list-style-type: none"> • Can use standard Apex classes in Remote Actions.
Radio Group	A group of radio buttons with common choices, such as a questionnaire.
Step	Create a page/view for the interaction.
Type Ahead Block	Auto-search and auto-complete dropdown. <ul style="list-style-type: none"> • Must have a data source (DR Extract, Remote or HTTP action). • Google Maps API is built in.
Inputs	
Checkbox	Display a checkbox. Boolean true/false output.
Currency	Input field for currency input. <ul style="list-style-type: none"> • Configurable currency symbol
Custom LWC	Enable custom components that do not extend an OmniScript element to exist in OmniScript.
Date	Date picker for one or more dates. <ul style="list-style-type: none"> • Calendar UI
Date/Time	Date/time picker for one or more dates/times. <ul style="list-style-type: none"> • Calendar UI and scrolling time UI
Disclosure	Display Disclosure text and has a checkbox for the user to select whether or not they agree with the disclosure.
Email	Input field for email address. Format: a@a.aa
File	Upload a file as an sObject attachment.
Image	Upload an image with preview in the OmniScript.
Lookup	Get data from Salesforce one field at a time.
Multi-select	Pick one or more present values from a checklist. <ul style="list-style-type: none"> • Displays as checklist.



Number	Input field for numbers.
Password	Input field for hidden text. <ul style="list-style-type: none"> • Format enforced with JS RegEx. • Not encrypted!
Range	Numeric sidebar. User chooses one value.
Select	Pick one preset value from a dropdown list.
Radio	Pick one preset value from list of options. <ul style="list-style-type: none"> • Can be displayed horizontally, vertically, as an image with or without text, or as buttons.
Telephone	Phone number format input. <ul style="list-style-type: none"> • Use for Salesforce's phone field.
Text	Input field for one line of text. <ul style="list-style-type: none"> • Use for Salesforce's text field.
Text Area	Input field for multiple lines of text. <ul style="list-style-type: none"> • Use for Salesforce's text area (long) field
Time	Input field to select a time. <ul style="list-style-type: none"> • Scrolling time UI
URL	Input field for a URL. Requires http:// or https://
OmniScripts	
(Reusable LWC OmniScript Name)	LWC OmniScript that has been marked as reusable.

OmniScript Properties

Feature	Function
Script Configuration	
Hard Coded language versions	Create a different version for each language.
Multi Language	Automatically translate OmniScript.
Right to left language	Display a language right to left.
Element Type To LWC Component Mapping	Change look and feel with a custom Lightning web component for all instances of the indicated element type.
Enable LWC	Enable as a Lightning web component.
Reusable	Allow this OmniScript to be embedded in another OmniScript.
Fetch Picklist Values At Script Load	Get Salesforce Picklist Values at runtime instead of design time.
Enable Unload Warning	Warn the user if navigating away from the script.
Enable Tracking	Track page view data.
Enable SEO	Make the OmniScript's URL available to search engines.
Currency Code	Hard coded currency code, specific to the OmniScript.
Console Tab Title	Customize the title of the console tab the OmniScript displays on.
Console Tab Icon	Select the Salesforce icon to display on the console tab the OmniScript displays on.
Seed Data JSON	Use to seed basic data in the OmniScript when it loads.
KNOWLEDGE OPTIONS	Display Salesforce Knowledge article. <ul style="list-style-type: none"> • API Driven
CANCEL OPTIONS	Options for when a user cancels an interaction.

SAVE OPTIONS	Save an incomplete OmniScript to resume later.
STEP CHART OPTIONS	Options to display the Step Chart.
ERROR MESSAGES	Options for error messages within the OmniScript.
MESSAGING FRAMEWORK	Options for Window, Pub/Sub, and temporary data storage.
STYLING OPTIONS	Options for the styling used on this OmniScript.
Common Properties	
Control Width	Set the display width of an element with dynamic spacing.
Label	Display label for the field.
Default Value	Display a default value instead of a blank field.
Help Text	Show help text.
Mask	Set format for user input. For example, a U.S. phone number would display as (999)999-9999.
Maximum Length	Check maximum length of entry.
Minimum Length	Check minimum length of entry.
Options	Set input values for dropdown, multi-select, and radio button controls. <ul style="list-style-type: none"> • Can use Salesforce picklist values.
Pattern	Enter the RegEx pattern required for the field.
Pattern Error Text	Enter an example of the correct pattern.
Required	Set the completion of a field as required.
Read-only	Make field display only, not editable.
Repeat	Enable a user to add one or more fields. <ul style="list-style-type: none"> • Adds an array to the data JSON.
ERROR MESSAGE	Create a custom error message.



USER MESSAGES	Configure what appears in failure modals.
Internal Notes	Leave notes for developers.
SEND/RESPONSE TRANSFORMATIONS	Limit, map, or rename data via JSON path.
Extra Payload	Limit, map, or rename data as key/value pairs.
CONDITIONAL VIEW	Show or hide element based on logical expression(s).

Integration Procedure Element References

Element	Function
Groups	
Cache Block	Cache data for specific actions with cache keys.
Conditional Block	Group elements with common conditions.
Loop Block	Iterate actions over items in a data array.
Try Catch Block	Return specified output or call an Apex class if an element fails.
Actions	
Batch Actions	Run a scheduled job.
Calculation Action	Call a Calculation Action <ul style="list-style-type: none">• May require a DataRaptor Transform.
DataRaptor Extract Action	Call a DataRaptor Extract.
DataRaptor Post Action	Call a DataRaptor Load.
DataRaptor Transform Action	Call a DataRaptor Transform.
DataRaptor Turbo Action	Call a DataRaptor Turbo.
Delete Action	Delete Salesforce records.
DocuSign Envelope Action	Send DocuSign emails for signature.
Email Action	Populate and send emails.
Integration Procedure Action	Call an Integration Procedure.
List Action	Merge lists.
Matrix Action	Call a Calculation Matrix.

Remote Action	Most things you can do with an Apex class and method. <ul style="list-style-type: none">• May require a DataRaptor Transform.
Response Action	Pass data back to launching object.
HTTP Action	Get, save or delete data through a web API. Methods: GET, POST, DELETE, PUT
Set Values	Set default values. Perform simple transforms, functions and mapping.

Integration Procedure Properties

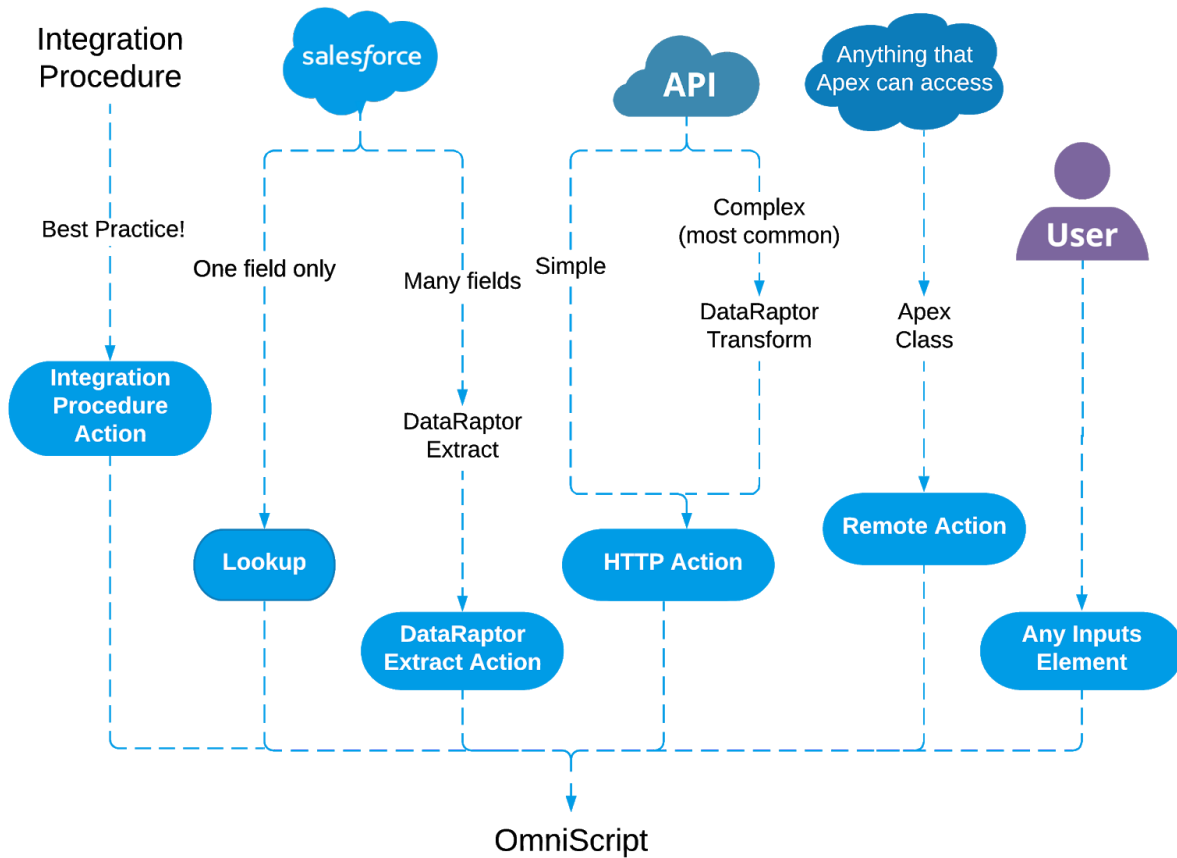
Element	Function
Procedure Configuration	
Tracking Custom Data	Define data passed to tracking
Include All Actions in Response	Write all action responses to the root JSON
Rollback On Error	Undo anything that changed if there is an error
Require Permission (Optional)	Use Salesforce Permissions set
CHAINABLE CONFIGURATION	Control how long-running procedures are executed
CACHE CONFIGURATION	Control when and where cache is stored
QUEUEABLE CHAINABLE LIMITS	Control how long-running elements are executed
TEST CONFIGURATION	Control if the procedure is a test.
Common Element Properties	
SEND/RESPONSE TRANSFORMATIONS	Transform data
Additional Input	Transform data
Additional Output	Transform data
Failure Response	Custom error messages
Execution Conditional Formula	Create a branch in a workflow
Failure Conditional Formula	Data validation
Fail On Step Error	Data validation



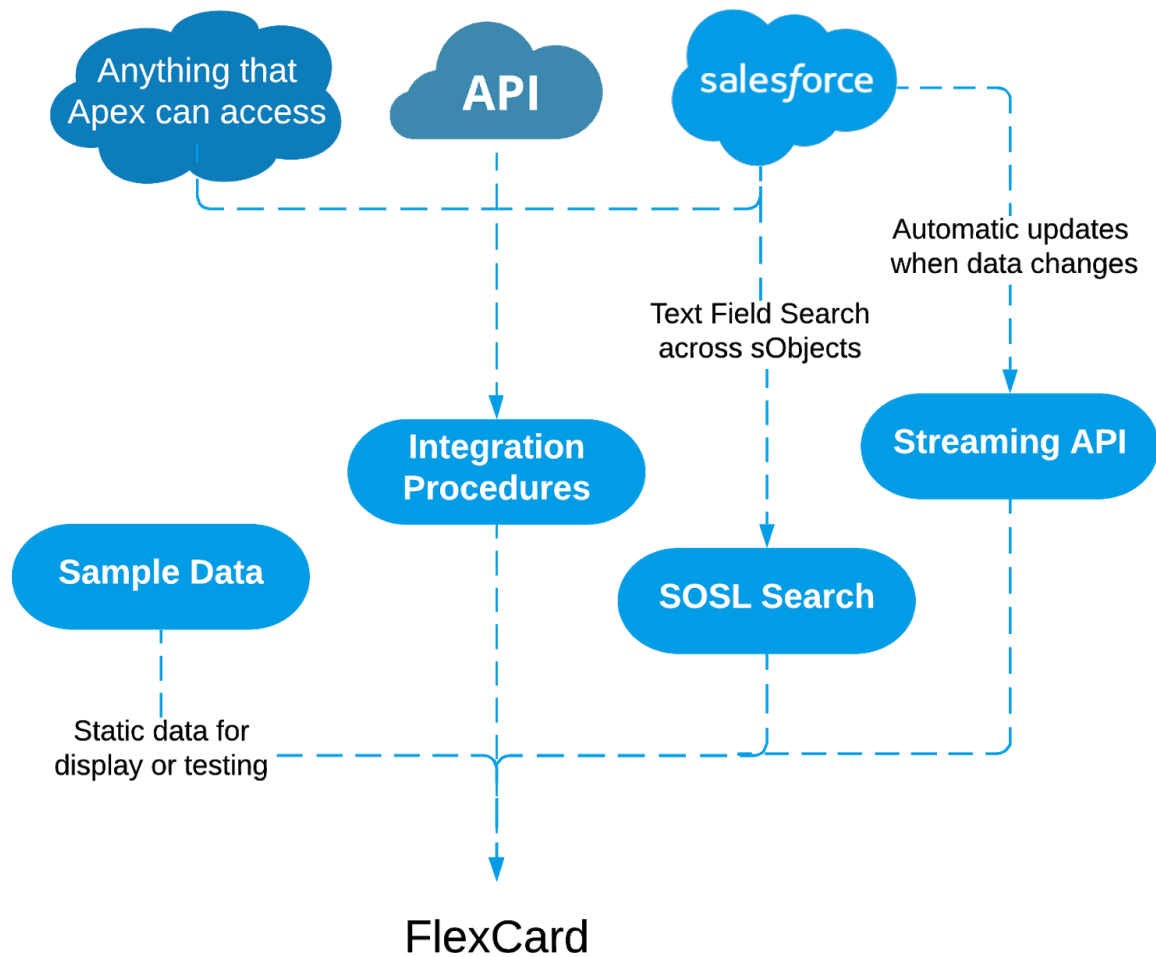
Chain On Step	Control how long-running elements are executed
Action Message	Define user message for batch processing
Internal Notes	Leave notes for developers

Data Decision Flowcharts

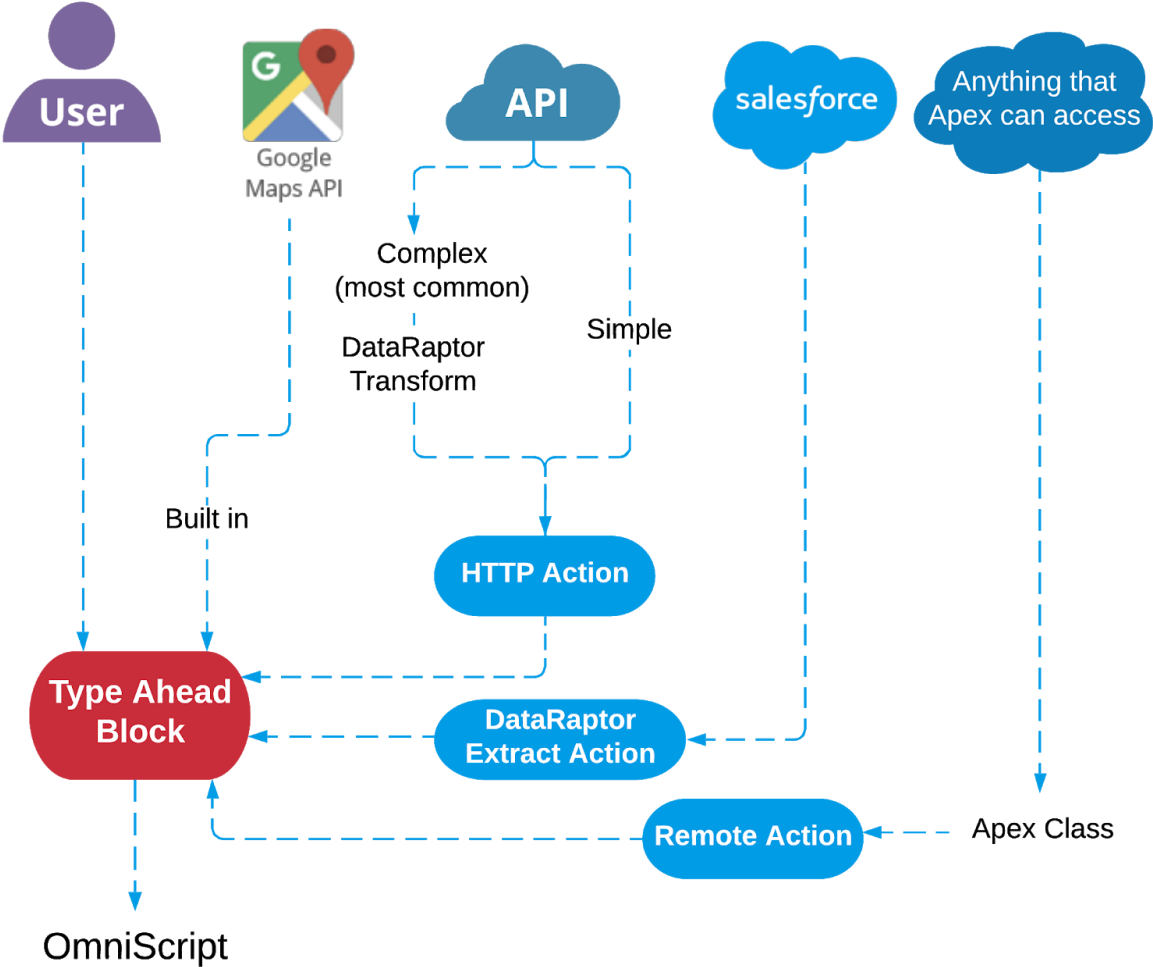
OmniScript: Data Inputs



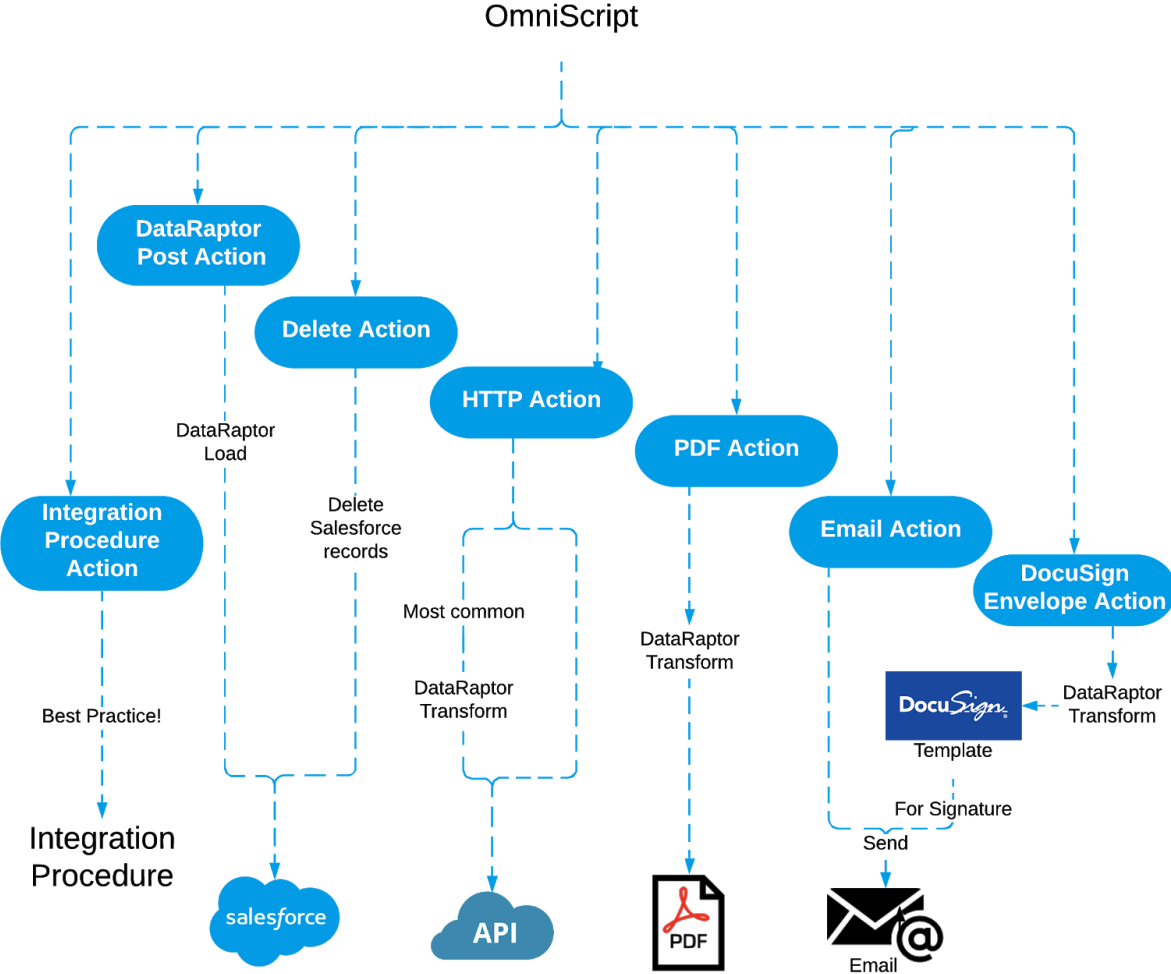
FlexCard Data Best Practices



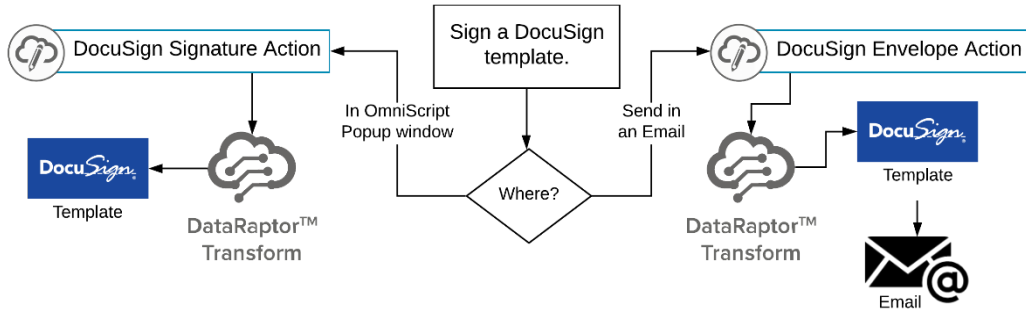
OmniScript: Data Inputs - Type Ahead



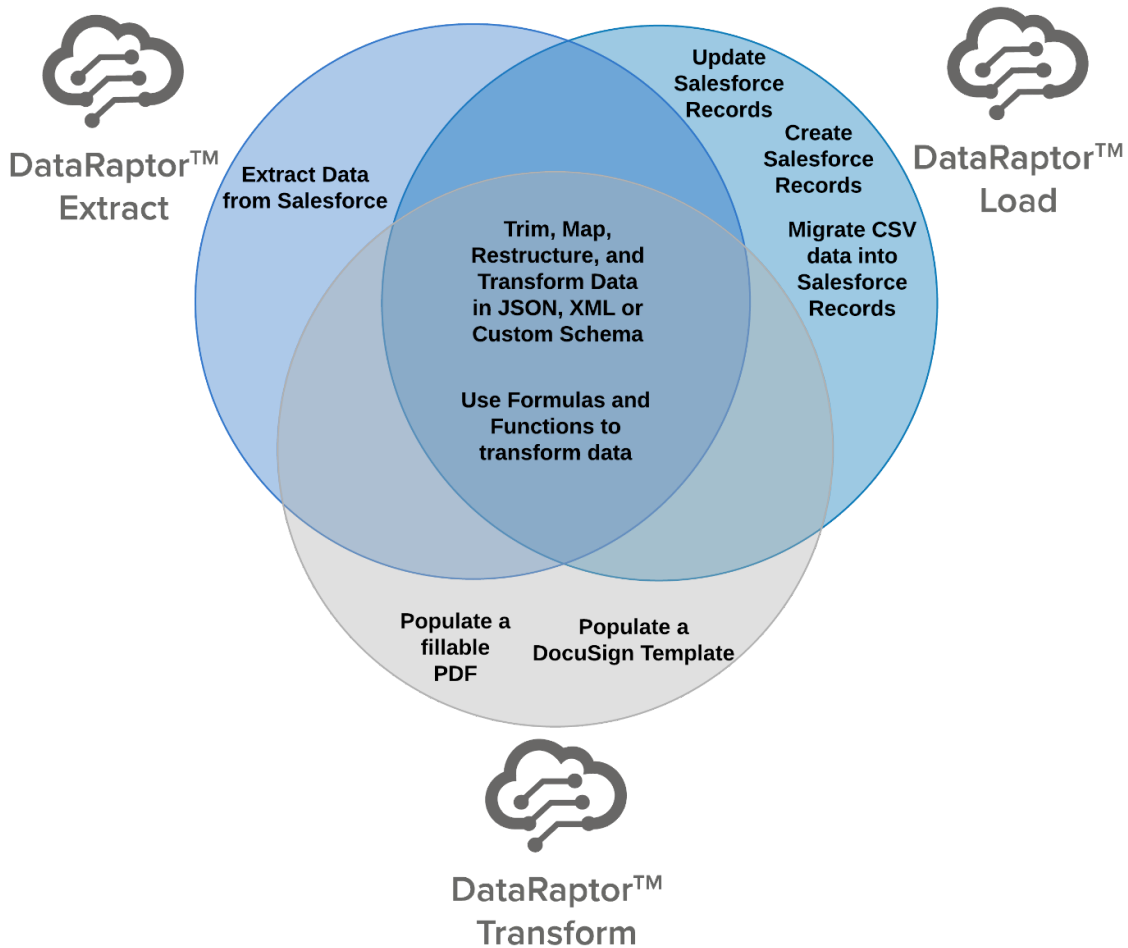
OmniScript: Data Outputs



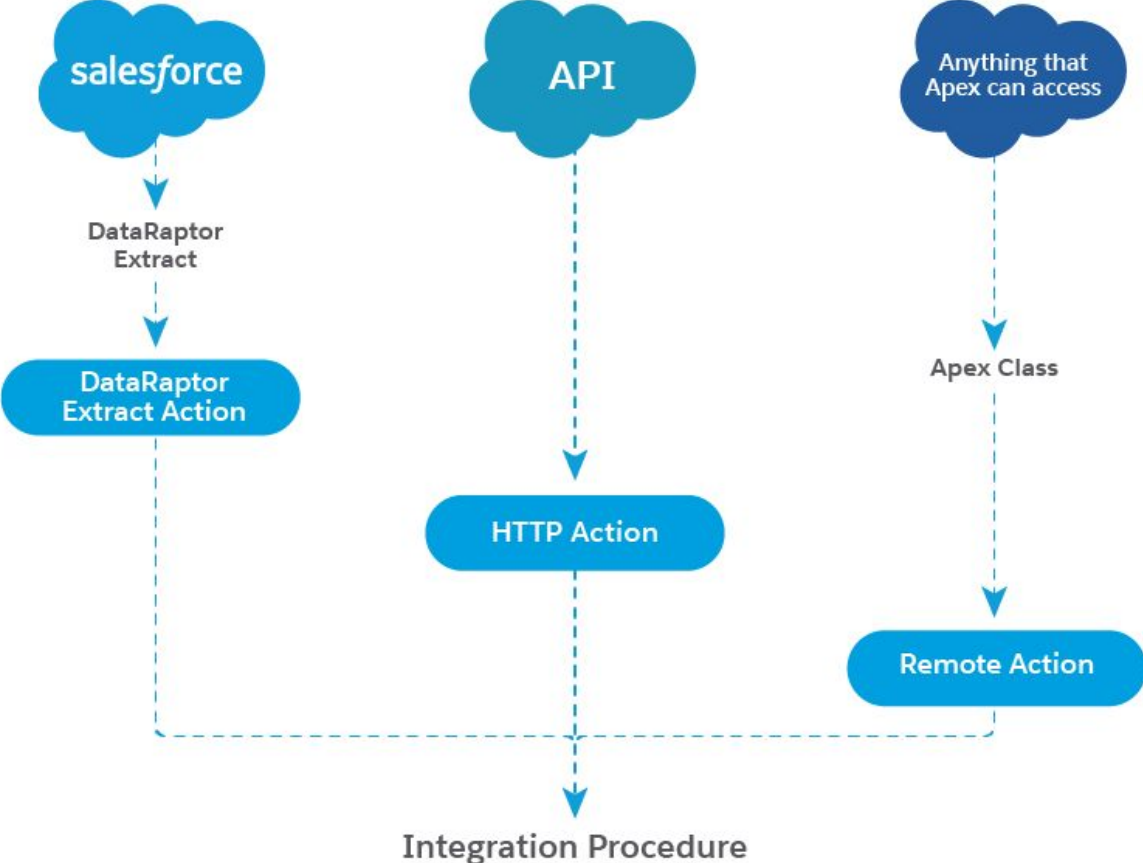
OmniScript: Data Outputs - DocuSign



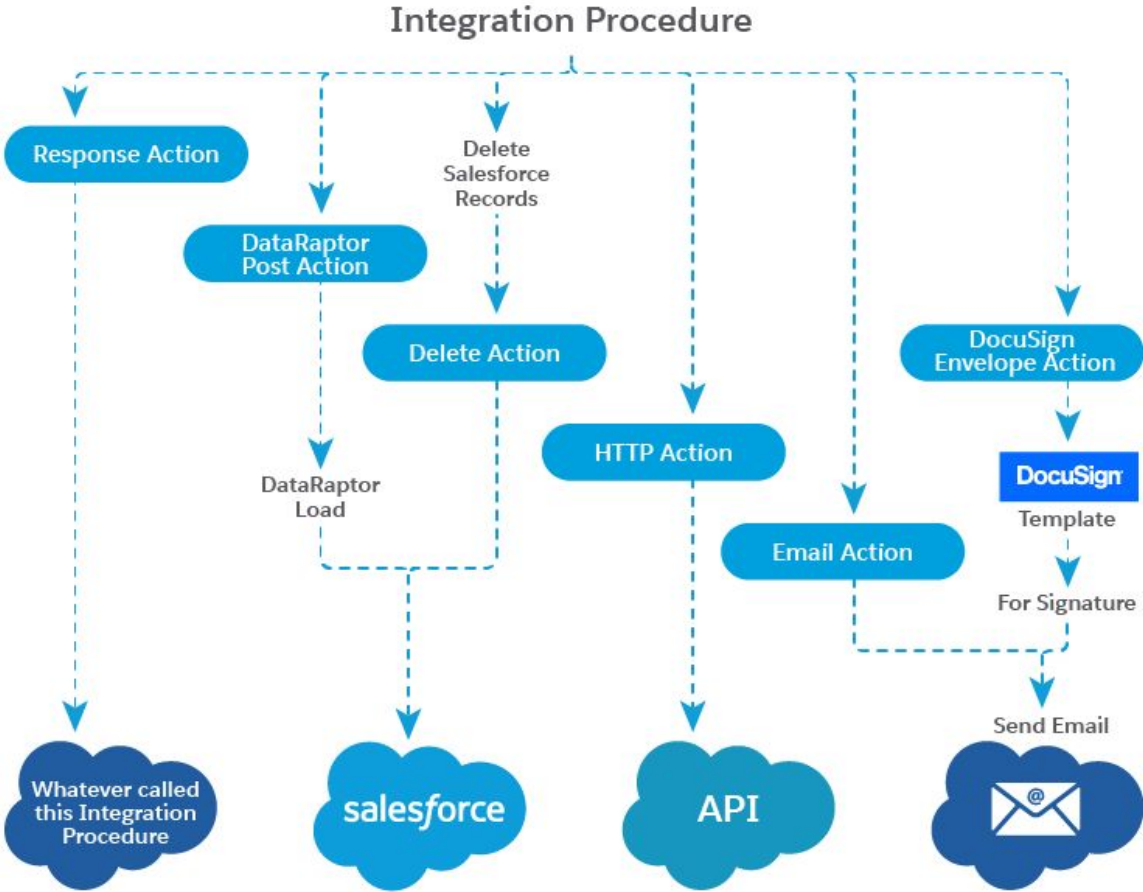
DataRaptor: Functional Venn Diagram



Integration Procedures: Data Input



Integration Procedures: Data Output



OmniStudio Naming Conventions

Application	Convention	Examples
DataRaptor Names	<p>Requirements:</p> <ul style="list-style-type: none"> • Must be unique within the org. • No spaces <p>Best Practices</p> <ul style="list-style-type: none"> • Use camelCase – prefix, Verb, Object and Detail • Use an action verb and descriptive nouns • Use abbreviations 	<p>prfxVerbObjDetail trngTypeAheadCase teamGetAcctCases</p>
DataRaptor JSON Paths	<p>Requirements</p> <ul style="list-style-type: none"> • Unique in the DataRaptor • Use ASCII characters • No spaces (some exceptions for XML) • To parse to OmniScript elements the node name must match the element name <p>Best Practices</p> <ul style="list-style-type: none"> • Use PascalCase • Short, descriptive names • Abbreviate • For Ids, Specify the object – if there is more than one extract from an object type, assign a label to each object 	<p>AccountName CallerName OwnerName AcctBillingStreet AccountId CallerId</p>
FlexCard Names and Authors	<p>Requirements:</p> <ul style="list-style-type: none"> • The combination of Name and Author must be unique. • Names and Authors can only contain underscores and alphanumeric characters. • Must begin with a letter, have no spaces, not end with an underscore, and can't have two consecutive underscores. • Especially avoid the apostrophe (') such as in a name (e.g. D'Angelo) • FlexCard and Author names can be changed by cloning the FlexCard 	<p>Name = AccountActive Author = CustomerSales Name = AccountClosed Author = CustomerSales</p>



Application	Convention	Examples
OmniScript Type & Sub Type	Requirements: <ul style="list-style-type: none"> ● Combination of Type/Sub Type/Language is unique in an org ● Type must start with a lower-case letter (LWC) ● No special characters Best Practices <ul style="list-style-type: none"> ● Use camelCase – prefix, Verb, Object and Detail ● Use an action verb and descriptive nouns ● Use abbreviations 	acctMgmt/verifyCaller team/editCase
	Requirements <ul style="list-style-type: none"> ● Unique in the script ● Use ASCII characters ● No spaces (no exceptions) Best Practices <ul style="list-style-type: none"> ● See DataRaptor JSON Paths above ● Examples for Actions – VerbObjectDetails with standard verbs <ul style="list-style-type: none"> ○ DataRaptor: Read, Update, Create, Xform ○ HTTP: Get, Post ○ Set Values/Errors: Set ○ DocuSign: DocuSend, DocuSign ○ Email: Send ○ PDF: PDFGen ○ Remote: Invoke ○ Calculation: 	

Merge Code Syntax

Application	Convention	Merge Code Syntax	Array Notation
FlexCard	recordId	{recordId}	{{Contact[1].ContactName}}
OmniScript & Integration Procedure	ContextId	<u>%Step:Block:Element%</u>	<u>%Parent 2:Child%</u>

Troubleshooting Guide

- Activate everything!
- Is Salesforce hanging up? Reload the Chrome page.
- Is Salesforce acting odd? Try closing out your browser window and logging in from a fresh window. This clears your browser cache.
- Use OmniScript Action Debugger to trace your data flow.
- Check the spelling of element names and JSON nodes.
- Check that abbreviations in variable and component names are always abbreviated and not spelled out in full. For example, ReadAcctData instead of ReadAccountData.
- Boolean values are case sensitive; check that true and false are all lower case. Do not use quotation marks when using a Boolean value.
- When troubleshooting in Preview mode, **Refresh** or **Clear Logs** in LWC OmniScript Designer and **Add Test Params** in FlexCards.
- If you are having problems with populating data; check that the names of variables in your OmniScript or Layout match those in the associated DataRaptor nodes.
- For Calculation Matrices and Calculation Procedures check that there is an effective version based on Start Date Time, End Date Time, and Priority.

Best Practices for VLOCITY

- Use Integration Procedures as the default data source for most cases and instead of Apex for REST and SOAP calls.
- Enable VLOCITY Metadata Platform Cache. This significantly speeds up DataRaptor and Integration Procedure performance.
- Plan your naming conventions and follow your team's guidelines.
- Plan your data flows. Use SEND/RESPONSE TRANSFORMS, Extra Payload, and Additional Input/Output to trim transferred data.
- Start small. Build your flows with a few fields first, then add the rest after you have tested the process.
- Use best practices from your team.
- Review the Success Community for documentation and training on any functionality in your project that is new to you.



Yay! All done!