



salesforce

Understanding Guide Template Language

Semantic Template Development
in Salesforce Marketing Cloud

Eliot Harper, AttributeValue
eliot@eliot.com.au, @eliotharper



Agenda



Template Languages

AMPscript, SSJS & GTL

About JSON

GTL Advantages

GTL Syntax



Template Languages



Template Engine is software designed to combine templates with a data model to produce result documents.

Language that the templates are written in is known as a *template language*

Many different template engines available today for different languages and runtimes

Vast majority have an emphasis on for processing structured data or text, including:

- variables and functions
- text replacement
- file inclusion (or transclusion)
- conditional evaluation and loops



Mustache and Handlebars



Guide Template Language is based on **Mustache** and **Handlebars**

Mustache

- provides a ‘logic-less’ approach to templating
- no `if` statements, `else` clauses or `for` loops, only tags

Handlebars

- superset of Mustache
- adds extensibility and minimal logic to Mustache through helpers, literals and partials





AMPscript, SSJS & GTL

AMPscript



Server-side scripting language

Simple syntax and semantics

Focuses primarily on content personalization

Interaction with Data Extensions, Sales Cloud, API functions, string transformation

Does not support JSON



Server Side JavaScript



Server Side JavaScript

Similar to AMPscript, but based on JavaScript

Extensive set of functions compared to AMPscript

Unlike JavaScript, does not work with the DOM and will not function with exterior libraries



Guide Template Language



Semantic templating language (not scripting language)

Based on Mustache and Handlebars

Natively parses JSON

Typically used in conjunction with AMPscript or SSJS

Interpreted by Marketing Cloud Outbound Mail Management (OMM) process



About JSON



Lightweight data-interchange format

Easy for humans to read and write

Easy for machines to parse and generate

Based on a subset of the JavaScript Programming Language

Language independent

Ideal for describing structured data

Schema-less



GTL Advantages



Mustache is a popular, easy-to-use templating language

JSON is schema-less

Ideal for structured data sources that use JSON (API feeds, ecommerce data)

Suited for non-complex data structures (cannot register your own Helpers)

Facilitates rapid development

Use cases include:

- Transactional emails; ecommerce order emails, data-based emails
- Content based feeds in emails and landing pages, social media, weather, blogs and more





Syntax

Tags



Indicate where a data substitution occurs and what information should be used for the substitution

Two types

- **Simple tags** outline a single line
- **Block tags** contain line(s) between the tags

```
{ {tag} }
```

```
{ {#if total > 100} }free shipping{ {/if} }
```



Data Sources



Different user-defined data sources:

- **Dataobjects** use a global scope, making data available for reuse multiple times via section tags.
- **Datasources** reside locally within the tags, making data accessible only inside the scope of the datasource tags

```
importData = { "data" : "Data from Dataobject using Section Tag" }
```

```
{{.dataobject sectionName}}
{{.data}}
  { "target" : "@importData" }
{{/data}}
{{/dataobject}}
```

```
{{#sectionName}}
{{importData.data}}
{{/sectionName}}
```

```
{{#sectionName}}
{{importData.data}}
{{/sectionName}}
```

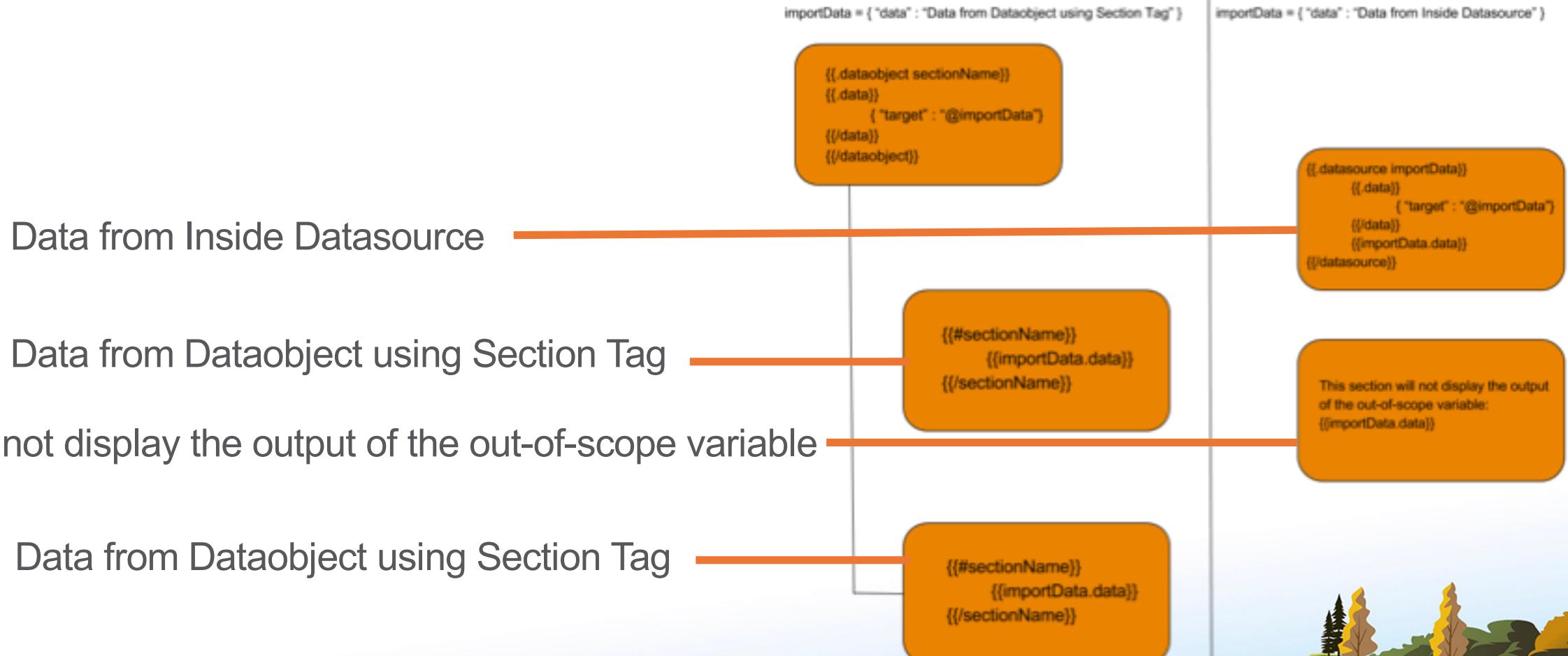
```
importData = { "data" : "Data from Inside Datasource" }
```

```
 {{.datasource importData}}
  {{.data}}
    { "target" : "@importData" }
  {{/data}}
  {{importData.data}}
{{/datasource}}
```

This section will not display the output
of the cut-of-scope variable:
{{importData.data}}



Data Sources



THANK YOU

