eServices Manager Plug-in for GAX

Field Code Variables

12/14/2025

Field Code Variables

Using Field Code Variables includes:

- Using UCS Data as System Variables in Standard Responses

- Custom Variables

- Using Your Own Data in Standard Responses

Using UCS Data as System Variables in Standard Responses

In the example given in Field Codes Overview, the field code `Contact.FirstName` retrieves a piece of data about the interaction. The ability to access interaction data is perhaps the most frequent use of field codes. Although field code formulas can be very complicated, many simply retrieve a single piece of data, such as a contact's name.

You access Universal Contact Server data using predefined variables, called "System Variables."

These variables access three predefined objects. Each object has a name and a set of properties. In the example, `Contact` is an object and `FirstName` is one of its properties. The system variable `Contact.FirstName` retrieves the value of the `FirstName` property of the `Contact` object.

In similar fashion, there is a system variable for each `object+property` pair. The objects and properties that you can use in field code formulas are described in the following sections.

Interaction

This object represents the particular interaction being worked on, such as an inbound email. These are its properties:

- AttachedData
- DateCreated
- FromAddress
- Id
- Subject
- TimeZone
- ToAddress

Contact

This object represents the contact associated with the interaction being worked on. These are its properties:

- FirstName

- FullName

- Id

- LastName

- PrimaryEmailAddress

- PrimaryPhoneNumber

- Title

Agent

This object represents the agent working on the interaction. These are its properties:

- FirstName

- FullName

- LastName

- Signature

> ## Important
>
> Automated responses use the default agent. Create the default agent as a Person object just like any other in Configuration Manager. Then select this Person on the Automated Reply Agent screen of the E-mail Server configuration wizard (or set this Person as the value of the autobot-agent-login-name option in the E-Mail Processing section of the E-mail Server Java application). Since this is the Person who the automated response appears to be from, you may want to name it after your company or institution.

Custom Variables

In addition to the system variables, you can use eServices Manager to create custom variables. Custom variables have the following properties:

- Their values are assigned by strategy objects.

- Therefore, standard responses that use field codes containing custom variables must have the usage type Autoresponse or Acknowledgment.

For an example of the use of a custom variable in a standard response, see Using a Custom Variable. For a complete description of the Routing objects that can use custom variables, see the Universal Routing 8.1 Reference Manual.

> **Important**
>
> The names of custom variables must begin with an alphabetic character or underscore, and the remainder of the name must consist only of alphanumeric characters or underscores. This differs from the requirements for the names of other objects, which may also contain hyphen and space. For example, 5-usercode is not an acceptable name for a custom variable, but it is acceptable as the name of a screening rule or category.

Using Your Own Data in Standard Responses

It is possible to incorporate data that you keep external to Universal Contact Server into your standard responses (including automated responses). This data could include case numbers, account information, and so on. Remember that attached data always consists of key-value pairs.

Incorporating external data into standard responses is a two-step process:

1. Retrieve the external information and add it to the interaction as attached data. One place to do this is in a routing strategy (see Interaction Routing Designer Help).

2. Now that you have attached the data to the interaction, you can use the `AttachedData` property of the Interaction object to access the data and incorporate it into your standard response. The `AttachedData` property requires one argument, which is the key name. The result of the following formula is the value associated with the `OrderStatus` attached-data key:

    ```
    <$Interaction.AttachedData("OrderStatus")$>
    ```