



Configuring Variables

- [Expanded Call Context Variables, on page 1](#)
- [Expanded Call Context Variable Configuration, on page 2](#)
- [User Variables, on page 4](#)
- [Define User Variables, on page 5](#)

Expanded Call Context Variables

Expanded Call Context (ECC) variables are variables that you define and enable in the Configuration Manager to store values for a call. You can specify the variable name and data type. The name must begin with the string "user." ECC variables are in addition to the variables the system software defines for each call (PeripheralVariable1 through PeripheralVariable10, CallerEnteredDigits, CallingLineID, and so on).

An ECC variable name can be up to 33 bytes long (1–32 usable characters). Use the following naming convention when creating an ECC variable:

user.<CompanyName>.<VariableDescription>

In this syntax:

- **<CompanyName>** is the name of your company.
- **<VariableDescription>** is a descriptive tag for the variable.

For example:

```
user.Cisco.AcctNum
```

Using this naming convention prevents naming conflicts with any third-party applications that interface with the system software.



Note For a large corporation, you can break **<VariableDescription>** down to include the Business Unit, Division, or other organizational entities.

ECC variables follow these size rules:

- An ECC variable can be either a scalar variable or an array element, each with a maximum length of 210 bytes.



Note Array types are not supported for an agent request.

- The maximum number of elements in an array is 255.
- The maximum buffer size for each scalar variable = 5 + the maximum variable length. The 5 bytes includes 4 bytes to tag the variable and 1 byte for the null terminator.
- The maximum buffer size for each array = 5 + (1 + the maximum length of an array element) * (the maximum elements in the array). There is a null terminator for each element, and a null terminator for the array as a whole.
- Since the internal buffer which stores the variables is 2000 bytes, the total sum of all the maximum buffer sizes for each variable and each array cannot exceed 2000.

For example, if you intended to use the following:

- A scalar ECC variable with a maximum length of 100 bytes
- A scalar ECC variable with a maximum length of 80 bytes
- An ECC array with a maximum of 9 elements with each element having a maximum length of 200 bytes

Totaled the buffer size is: $(5+100) + (5+80) + (5 + (1+200)*9) = 2004$. Because this size is too large, you must change the length of one of the scalar ECC variables or the length of the array ECC variables.

ECC Variables for Blended Collaboration or Voice MRDs with Collaboration

ECC variables must be configured in Configuration Manager's Expanded Call Variable List tool (for each integrated application) to route requests using the voice Media Routing Domain.

For Cisco Blended Collaboration or Voice MRDs with Collaboration, the ECC variables are:

- user.cisco.cmb
- user.cisco.cmb.callclass
- user.ece.activity.id
- user.ece.customer.name



Important While their default size is 40 characters, use the Expanded Call Variable List tool in the Configuration Manager to limit the user.cisco.cmb variable to 8 bytes and the user.cisco.cmb.callclass variable to 10 bytes to prevent ECC space limitation issues.

Expanded Call Context Variable Configuration

Expanded call context variable configuration consists of two steps:

- Enable ECC variables
- Define ECC variables

For Web Callback and Delayed Callback to work properly, an ECC variable (also known as a named variable) must be defined. The Cisco CTI driver supports the use of ECC variables in addition to the standard call variables associated with a call. Before an ECC variable can be used, it must be defined in the Unified ICM ECC variable database table.



Note For more information, refer to the *Database Schema Handbook for Cisco Unified Contact Center Enterprise*.

Enable ECC Variables

Procedure

- Step 1** Within the Configuration Manager, double-click **Tools > Miscellaneous Tools > System Information**. The System Information window appears.
- Step 2** Select the **Expanded call context enabled** check box. For additional information, refer to the online Help.
- Step 3** Click **Save** to apply your changes.
-

Define ECC Variables

Procedure

- Step 1** Within the Configuration Manager, double-click **Tools > List Tools > Expanded Call Variable List**. The **Expanded Call Variable List** window appears.
- Step 2** Click **Retrieve** to enable adding ECC variables.
- Step 3** Click **Add**. The **Attributes** property tab appears.
- Step 4** Complete the **Attributes** property tab. See the *List Tools Online Help* for details on the **Attributes** property tab.
- Step 5** Click **Save** to apply your changes.
-

What to do next

If you change the configuration of any ECC variable with the **Expanded Call Variable List** tool, restart the Unified CVP Call Server or VRU PIM to force a renegotiation of the ECC variables.

Validate ECC Variable Size for CTI Server

Before configuring ECC variables, validate the total size of the ECC variables against the following rules and limits:

- Because the total size of the buffer used to store the variables in CTI Server internally is 2500 bytes, the total sum of all the maximum buffer sizes for each scalar variable and arrays must be no greater than 2500.
- The maximum buffer size for each scalar variable = 4 + length of the ECC name + the maximum length of the variable where the 4 bytes includes a 1 byte tag, 1 byte to define the length, and 2 terminating NULL characters.
- The maximum buffer size for each array = (5 + length of the ECC name + the maximum length of array element) * (the maximum number of elements in the array) where the 5 bytes includes a 1 byte tag, 1 byte to define the length, 1 byte for the array index, and 2 terminating NULL characters.
- For example, if you intend to use one scalar ECC variable with a maximum length of 100 bytes named *user.var*, one scalar ECC variable with a maximum length of 80 bytes named *user.vartwo*, and an ECC array named *user.varthree* with a maximum of 9 elements with each element having a maximum length of 200 bytes, the buffer size would be:

$$(4+8+100) + (4+11+80) + ((5 + 13 + 200)*9) = 2169$$

where 8 is the length of *user.var*, 11 is the length of *user.vartwo* and 13 is the length of *user.varthree*.

User Variables

You can also create global user variables; for example, you can create a user variable called *usertemp* to serve as a temporary storage area for a string value used by an If node.

After you have defined a user variable, you can then use the Script Editor Formula Editor to access the variable and reference it in expressions, just as you would with a “built-in” variable.

Each user variable must:

- Have a name that begins with **user**.



Note This name cannot contain the dot/period (.) character.

- Be associated with an object type, for example, Service. (This enables the system software to maintain an instance of that variable for each object of that type in the system.)
- Be checked as persistent. A persistent variable maintains its value between script invocations. This allows you to set the variable in one script and reference later in another script.



Note Because these variables may be persisted, do not use User Variables to store sensitive information belonging to the customer or company. Using these variables to store confidential information could lead to violation of security standards, such as PCI, the Common Criteria, HIPAA, or FIPS 140-2.

A user variable can store a value up to 40 characters long.

Define User Variables

Procedure

Step 1 Within the Configuration Manager, select **Tools > List Tools > User Variable List**.

The User Variable List window appears.

Step 2 In the User Variable List window, click **Retrieve** to enable Add.

Step 3 Click **Add**.

The Attributes property tab appears.

Step 4 Complete the Attributes property tab.

Note The **Variable name**, **Object type**, and **Data type** fields are required. All other fields are optional. For additional information refer to the online Help.

Step 5 Click **Save** to apply your changes.
