[Implementing security authentication using LDAP in IBM Integration Bus →](https://eaideveloper.wordpress.com/2014/01/29/implementing-security-authentication-using-ldap-in-ibm-integration-bus/)

[**ESQL Variable Scope and lifetime**](https://eaideveloper.wordpress.com/2014/01/18/esql-variable-scope-and-lifetime/)

Posted on [January 18, 2014](https://eaideveloper.wordpress.com/2014/01/18/esql-variable-scope-and-lifetime/)by [vishnu](https://eaideveloper.wordpress.com/author/vishnurajnr/" \o "View all posts by vishnu)

***Introduction***

The variables declared and used in ESQL will have a scope (Flow/Module/Nodes to which the variable is accessible) and lifetime (time till the variable holds the value). When we declare a variable in ESQL, it will have the above two characteristics defined already based on the declaration syntax and place where it is declared.

***Objective***

This article objective is to understand the scope and lifetime of the ESQL variables used in WMB.

There are mainly 3 types of variables in ESQL as below:

1. Module Variable ( Normal variable)
2. External Variable (UDP variable)
3. Shared Variable (long Lived variable)

**Normal or Module variable**

When we declare a variable within an ESQL module or without any special keyword, those are called Normal variable. This variable’s lifetime start with a node and end within that node. The visibilities of those varaibles are within the module only. For every message which pass through the **node**,  a new instance of these normal variables will be created.

But if we define a normal variable outside the node, ie on schema level, the variable will be accessible to all the modules in the schema. But a separate instance will be given for each module.

In general, if **DECLARE**ed a variable in schema level without any keyword, its accessible across the nodes within the schema, but the value is **not** **SHARE**able.

Scope: for every single message passing through the node.

Lifetime: For a single message passing through the node.

Usage: for normal function/procedure calls, data processing.

**External variable or User-Defined property Variable**

When we need a variable whose values are set at deployment time, we can create User defined property or External variables. External variables must be declared within the schema or module scope before actually using it. UDPs can be accessed by any built-in node that uses ESQL or JAVA. External variables are accessible anywhere in the flow and are not restricted to any particular node. External variable can be initialised by either the message flow editor (UDP) or BAR file. For every message which passes through the **Flow**, a new instance of these variables will be created. EXTERNAL variable value cannot be changed by ESQL.

Scope: for all messages passing through the flow

Lifetime:  Till the message flow stops.

Usage: For configuration data like database name, schema name, WS-Use rid, Password etc.

**Shared Variable (Long Lived variable)**

Shared variables are those as the name indicates share the value across multiple message flow instances. Shared variable lifetime is till the execution group restart and scope is throughout the message flow. It can be accessed in any nodes which uses ESQL. It can be modified using any node/ESQL in the flow.

SHARED variables can be accessed by multiple instance of same message flow in the same execution group. SHARED variable is accessible to all the flows in the execution group, but, a different instance will be getting for each different flow. Means, you cannot actually SHARE the data using SHARED variable between different flows in same execution group. Only the Same flow instances will get the SHARED data using SHARED variables.

Scope: for all messages passing through the flow.

Lifetime:  Till the execution group stops.

**Note:** If you declare a variable in Schema level without any keyword (normal), it will be accessible to each ESQL node. But each node get a new instance of this variable, ie, data cannot be shared by normal schema level variables. If the data need to be shared, the ESQL variable in schema should be declared as SHARED.

You should use the ATOMIC block if want to update the SHARED Variable in thread safe manner.

Table summary:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variable Type | Scope | Lifetime | Modification | Initialise | Usage |
| Normal | Within Module or node | Within Node | will reflect in Node | each instance, via ESQL | Normal data |
| EXTERNAL | Message flow | Message Flow | Can modify at runtime using CMP API or WMB explorer. | Only once via Message flow editor or BAR file | Configurable data |
| SHARED | Message Flow | Execution Group | Will reflect in all ESQL under same schema | Initialise and update whenever required in the flow | Caching data |

**References**