Data Pools

With Event Dataflow programming all data that a *Node* needs to process must either be present in a) incoming event(s), b) attributes of the *Node* or c) shared data pools.

With VDAB pools are used to persist and share data that is not in the form of an Event.

This document details how pools of data can be used to share data between nodes in a flow, different flows in a container and different containers.

Before This	• You should have reviewed the VDAB introductory tutorial and documentation to understand how to create basic flows from nodes.

Contents

Related Documentation

The following document and tutorials either are a) available or b) being developed to further support this subject. Those available are highlighted in green while those under development are currently grey.

Related Guides	Details

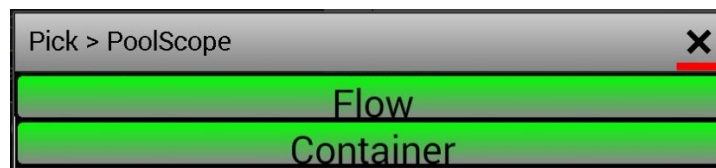| Related Tutorial	Details

Data Pool Overview

Data pools are used to temporarily or permanently hold data that needs to be shared between nodes.

Unlike event data which always has a time associated with the data, pool data is not associated with a particular point in time and an update to a particular pool data item will overwrite the previous data.

Data pools are supported by two different nodes, one which puts data in the pool and the other that retrieves data from the pool.

Like data used with other programming languages, the data in the pools has a scope which defines where this data can be accessed.

The following different pool scopes have been defined. (Current versions of VDAB only support the *Flow* and *Container* scope.)

Pool Scope	Visibility
Flow Instance*	Data is available within the currently executing session of this flow.
Flow	Data is available within the flow and is shared between all currently running instances of the flow.
Application*	Data is shared between all flows considered in the same application.
Container	Data is shared globally between all flows and applications running on the container.

* This scope types are not currently supported but will be in future releases.
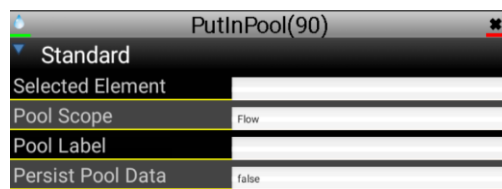
Putting Data in a Pool

Data is put into a pool using the *PutInPool* function node.
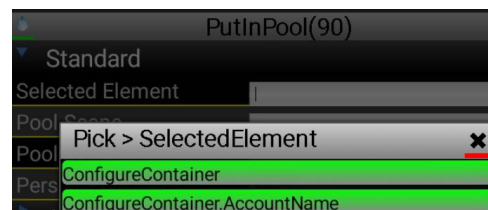
Using the PutInPool Node

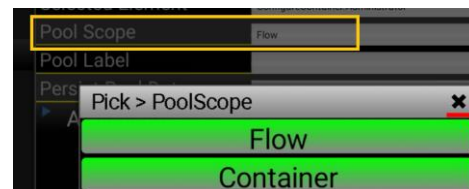1. Add the *PutInPool* function node to an existing or new flow.

2. After selecting the PutInPool node click on it then click on the Edit action.
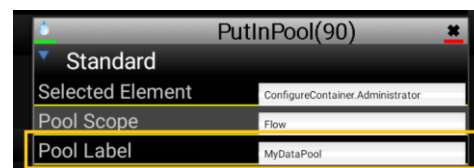
3. Select the element that should be put in the pool. If the flow has been run previously, you can select the available data elements from a pop up list.
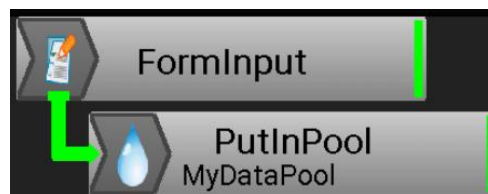
4. Chose the scope of the Pool. If the scope is set to *Flow* the pool data can be read only in the same flow. If the scope is set to *Container* the pool data can be read by any flow running on the container.

5. Give the pool a label or name that will be used when retrieving the data. Click in the upper left to save these node edits.

6. Start the flow. The selected data will be put in the pool and can be retrieved by other nodes or flows.

Retrieve Data from a Pool

Data can be retrieved from a pools using either the *GetFromPool* or the *PoolInput* nodes.

These should be used in the following circumstances:

Node	Node Type	When Used
GetFromPool	Function	When the pool data should be retrieved whenever another event or trigger occurs. When an event goes into the node, the data from the pool and the incoming event are merged.
PoolInput	Input	When the pool data should be published to this flow whenever the pool data is changed.

Using the GetFromPool Node

Using the PoolInput Node

Reference Pool Data in a Template

Appendix: Documentation for Nodes

This same information is available for all nodes by going to your local vdab web server or to
http://vdabsoft.net/vdab/docs_node*.*

PutInPool		Beta
Puts selected event data in a named data pool that can be later accessed by later nodes.		
SelectedElement	Selects the data item that will be put into the pool.	
PoolScope	Select the scope for the pool data. Current choice are Flow and Container.	
PoolLabel	Sets the label for the pool data.	
PersistPoolData	NOT CURRENTLY SUPPORTED: In future releases if set to true, the pool data will be saved permanently and be available when the container is restarted.	

GetFromPool		Beta
Retrieves data from a named data pool that has been populated by a PutInPool node.		
PoolScope	Select the scope for the pool data. Current choice are Flow and Container.	
PoolLabel	Selects the label for the pool data that will be retrieved.	
SelectedElement	Selects the data item that will be retrieved from the pool.	
ClearPoolAfter	null	

PoolInput		Beta
Listens for changed data from the selected pool.		
PoolScope	Select the scope for the pool data. Current choice are Flow and Container.	
PoolLabel	Selects the label for the pool data that will be monitored for changes.	
SelectedElement	Selects the data item that will be retrieved from the pool.	