

SharedModelMap

Sometimes non-persistent data needs to be shared across virtual services. The 'SharedModelMap' object supports this requirement. 'SharedModelMap' enables exchange of data by keys. One VSM can do something like this (taken from scriptable DPH sample below):

```
com.itko.lisa.vse.SharedModelMap.put("transactionName",  
"currentOperation", fileToSave);
```

And another VSM can retrieve the key's value as follows:

```
String fileName =  
com.itko.lisa.vse.SharedModelMap.get("transactionName",  
"currentOperation");
```

The first parameter in both calls is the optional namespace. It is optional but recommended so that keys are distinguished by different namespaces and unique within a namespace.

SharedModelMap supports following methods. For each method there is a variant available with and without specification of a namespace:

<code>int size()</code> <code>int size(String nameSpace)</code>	Returns the number of entries in the default or specified namespace, respectively.
<code>boolean isEmpty()</code> <code>boolean isEmpty(String nameSpace)</code>	Returns whether or not the (default or specified) namespace is empty.

boolean containsKey(String key) boolean containsKey(String nameSpace, String key)	Returns whether or not the (default or specified) namespace contains an entry with the given key.
boolean containsValue(Object key) boolean containsValue(String nameSpace, Object key)	Returns whether or not the (default or specified) namespace contains an entry with the given value.
Object getObject(String key) Object getObject(String nameSpace, String key)	Gets a value from the (default or specified) namespace.
String get(String key) String get(String nameSpace, String key)	Gets a value from the (default or specified) namespace cast as a string.
Object putObject(String key, Object value) Object putObject(String nameSpace, String key, Object value)	Puts a value into the (default or specified) namespace.
String put(String key, String value) String put(String nameSpace, String key, String value)	Puts a value into the (default or specified) namespace cast as a string, and returns the currently known value of the key cast as a String.

String remove(String key) String remove(String namespace, String key)	Removes a value from the (default or specified) namespace, and returns the currently known value of the key cast as a String
void clear() void clear(String namespace)	Clears the (default or specified) namespace
Set<String> keySet() Set<String> keySet(String namespace)	Returns the set of keys currently known in the (default or specified) namespace. The returned set will never be <i>null</i> .
void setCapacity(String namespace, int newCapacity)	Resizes the capacity of the specified namespace

Each unique namespace is backed by a map that is restricted to a default capacity of 256 key/value pairs. Each namespace has a LRU (Last Recently Used) map created on demand the first time the namespace is used. It's backed by an 'org.apache.commons.collections.map.LRUMap'. The LRU map determines which map entry to remove if there is no capacity left.

SharedModelMaps are not persisted across VirtualEnvironmentService restarts.