



# Custom Persistence with DataScript

Avi Technical Reference (v17.2)

Copyright © 2018

## Custom Persistence with DataScript

[view online](#)

Avi Vantage includes a number of common methods of persisting clients to the same server for a duration of time. For some applications, it may be valuable to have more customization to the behavior of the persistence, or to support a wider array of application types. See [Overview of Server Persistence](#) for other methods.

Persistence is based on the ability to uniquely identify a user, and to stick or persist them to a destination server for a period of time.

The primary DataScript function for creating custom persistence are the table commands, which enable storing data locally on an Avi Service Engine for a configurable length of time. The stored data is also replicated to other relevant Service Engines for high availability and scale.

<a href="#">avi.vs.table_insert([table_name,] key, value [, lifetime])</a>	Store custom data in a time-based table
<a href="#">avi.vs.table_lookup([table_name,] key [, lifetime_exten])</a>	Lookup data in a table
<a href="#">avi.vs.table_remove([table_name,] key)</a>	Remove data from a table
<a href="#">avi.vs.table_refresh([table_name,] key [, lifetime_exten])</a>	Update the expire time for a table entry

### JSessionID Example

The following example persists based upon an HTTP cookie sent from a server with a unique JSessionID. Subsequent client requests include that cookie, which is used to persist requests to the same server.

HTTP Response event: Add persist for 20 minutes, or update timer if entry already exists

```
if avi.http.get_cookie("JSESSIONID") then
  avi.vs.table_insert(avi.http.get_cookie("JSESSIONID"), avi.pool.server_ip(), 1200)
end
```

HTTP Request event: Rename the pool before applying to a virtual service

```
if avi.http.get_cookie("JSESSIONID") then
  avi.pool.select("poolname", avi.vs.table_lookup(avi.http.get_cookie("JSESSIONID")))
end
```