
Question

[Eduard Lebedyuk](#) · Feb 19, 2018

Approaches to bash scripting (passing variables from OS to Caché)

Let's say I want to execute this cache script (saved as test.txt) from OS terminal:

```
zn "USER"  
write 1  
zn "%SYS"  
write 2  
halt
```

Executing the following command in a terminal:

```
csession cache < test.txt
```

Would yield this output:

```
$ csession cache < script.txt
```

```
Node: gitlab-test, Instance: CACHE
```

```
USER>
```

```
USER>
```

```
1
```

```
USER>
```

```
%SYS>
```

```
2
```

```
%SYS>
```

```
Job succeeded
```

Is there a better way to run these scripts?

Currently I have two problems:

1. I have several variables defined in my bash script, what's the best approach to pass them into Caché?
2. I can't capture script input (for example write 1)

As for 1 there are two solutions I found:

1. Generate test.txt dynamically from bash. That way I can just hardcode variables, but I don't really like this approach because writing a script that generates another script and executes it is non-trivial, also parent script is hard to read.
2. Use `$system.Util.GetEnviron("EnvVar")` to get variables from Cache side. It works, but maybe there are

other, better approaches?

No ideas so far for 2 though.

[#Change Management](#) [#Continuous Integration](#) [#Terminal](#) [#DevOps](#) [#Caché](#)

Source

URL: <https://community.intersystems.com/post/approaches-bash-scripting-passing-variables-os-cach%C3%A9>