

## Infrastructure - Task #2380

Story # 2045 (Closed): Get the DataONE CLI ready for public release

### CLI sliently ignores invalid boolean values

2012-02-23 18:28 - Andrew Pippin

<b>Status:</b>	Closed	<b>Start date:</b>	2012-02-23
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Andrew Pippin	<b>% Done:</b>	100%
<b>Category:</b>	d1_client_cli	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	Sprint-2012.11-Block.2.2	<b>Story Points:</b>	
<b>Milestone:</b>	CCI-1.0.0		
<b>Product Version:</b>	*		

#### Description

It is possible to set the variable "verbose" to a value of "@fred@". No error is thrown and the value remains unchanged.

The CLI should report invalid values, and allow for case insensitive boolean values ("@true@", "@True@", "@tRuE@" should all succeed).

#### History

##### #1 - 2012-02-23 18:38 - Andrew Pippin

- Category set to d1\_client\_cli

- Assignee set to Andrew Pippin

Updated Assinged\_To and Category.

##### #2 - 2012-02-23 20:29 - Matthew Jones

- Tracker changed from Task to Bug

- Target version set to Sprint-2012.07-Block.1.4

- Parent task set to #2045

##### #3 - 2012-02-23 22:21 - Roger Dahl

This is caused by a design decision that might not be a good one. The issue is that in order to easily support session variables that are unset (None), boolean, integers or strings, I implemented those as native Python variables and parse them as such. It makes things very clear in the code, but it causes some unexpected results, such as being able to have "fred" a valid value for True, simply because the string "fred" evaluates to True in Python.

So, a value can be set to False by passing in values that evaluate to False in Python, such as "0" and False. It can be set to True with "1", and "fred". It can be set to None by passing "None" (for which the clear command is a shortcut).

##### #4 - 2012-02-28 17:28 - Andrew Pippin

Roger is correct. "fred" is a valid value. At first glance, the result is unexpected, but it shouldn't have been.

A possible solution would be to have a "validator" hook for each session variable. This would also help with [#2377](#) "Verify that the object format is valid in CLI".

##### #5 - 2012-03-01 01:00 - Andrew Pippin

- Status changed from New to In Progress

Valid boolean values are:

TRUE: ('true', 'True', 't', 'T', 1, 'yes', 'Yes' )

FALSE: ('false', 'False', 'f', 'F', 0, 'no', 'No' )

**#6 - 2012-03-01 17:16 - Andrew Pippin**

- % Done changed from 0 to 100

- Status changed from In Progress to Closed

SVN commit 7246.