

Infrastructure - Story #6494

Remove local caching of inputstreams from v1 API method implementations

2014-10-02 19:46 - Rob Nahf

Status: Closed	Start date: 2014-11-10
Priority: Normal	Due date: 2014-12-15
Assignee: Rob Nahf	% Done: 100%
Category: d1_libclient_java	Estimated time: 0.00 hour
Target version: CCI-1.5.0	
Story Points:	
Description buffering input streams into byte arrays to close remote input streams creates an unmanaged situation where a large data object can cause out-of-memory / heap errors. This is bad because application developers who use libclient have to pass on this weakness to their customers. The solution is to avoid buffering, but also wrap the inputstream with AutoClosingInputStream to prevent accidental input streams being left open and tying up connections. The buffering LocalCache imposes is potentially dangerous as well and needs a workaround, both for v1.x and v2.x.	
Subtasks: Task # 6559: in v1, turn off localCache by default, and add a size limit to avoid out o... Closed	
Related issues:	
Related to Infrastructure - Bug #2211: Use of d1_libclient_java may lead to o...	Closed 2012-01-19
Related to Java Client - Story #6508: libclient_java 1.4.0 Release	Rejected 2014-10-06
Related to Java Client - Story #6560: in v2, remove LocalCache from get and g...	Closed 2014-11-10

History

#1 - 2014-10-02 20:36 - Rob Nahf

- translation missing: en.field_remaining_hours set to 4.0
- Estimated time set to 4.00

probably want to test AutoCloseInputStream behavior

#2 - 2014-10-02 21:59 - Rob Nahf

- Subject changed from confirm that libclient methods don't localize input streams except in LocalCache. to Remove local caching of inputstreams from v1 API method implementations
- Parent task deleted (#3369)
- Category set to d1_libclient_java

(also implement in v2)

#3 - 2014-10-02 22:00 - Rob Nahf

- Due date set to 2014-10-17
- translation missing: en.field_remaining_hours changed from 4.0 to 0.0
- Tracker changed from Task to Story

this constitutes a behavior change, so should be part of the next minor release.

#4 - 2014-10-02 22:00 - Rob Nahf

- Target version changed from CCI-1.4.1 to CCI-1.5.0
- Due date changed from 2014-10-17 to 2014-10-02

#5 - 2014-10-20 20:55 - Rob Nahf

- *Product Version changed from * to 1.4.0*
- *Due date changed from 2014-10-02 to 2014-10-20*

#6 - 2014-10-20 21:04 - Rob Nahf

- *Target version changed from CCI-1.5.0 to CCI-1.5.1*

#7 - 2014-11-04 19:03 - Rob Nahf

- *Due date changed from 2014-10-20 to 2014-11-04*
- *Target version changed from CCI-1.5.1 to CCI-1.5.0*

#8 - 2014-11-10 18:16 - Rob Nahf

- *Status changed from New to In Progress*
- *Due date changed from 2014-11-04 to 2014-11-10*
- *Description updated*

removed input stream buffering into bytearrays for `d1node.get()` and `d1node.query()` methods - the two that return a bare input stream. `d1Node.get()` has a `LocalCache` implementation that also buffers the content into `byte[]`, so I placed a size limit on caching so that the item isn't cached if it is over a certain size. This isn't the optimal solution, but is a workaround for the unmanaged out-of-memory exception.

#9 - 2014-11-10 18:44 - Rob Nahf

- *Description updated*
- *% Done changed from 0 to 50*

#10 - 2014-11-12 22:24 - Rob Nahf

- *Status changed from In Progress to Testing*
- *Due date changed from 2014-11-11 to 2014-11-12*

#11 - 2014-12-15 18:27 - Rob Nahf

- *Status changed from Testing to Closed*
- *Product Version changed from 1.4.0 to 1.3.1*
- *Due date changed from 2014-11-25 to 2014-12-15*

no issues encountered in beta and stage testing of CN-1.5.0.