CAPE-OPEN

Errata and clarifications for

Collection Common interface specification

www.colan.org

ARCHIVAL INFORMATION

Filename	Collection_Errata_1.0_1.008.docx
Authors	CO-LaN consortium
Status	Approved for public release
Date	February 2014
Version	Version 1.00
Number of pages	7
Versioning	1.000 created on Aug 25, 2012 by Michel PONS (CO-LaN)
	1.001 edited on Sep 12, 2012 by Michel PONS (CO-LaN)
	1.002 edited on July 10, 2013 by Michel PONS and Bill BARRETT: partial restructuring.
	1.003 edited following M&T conf call on July 10, 2013
	1.005 edited following M&T conf call on Aug 7, 2013
	1.006 edited following M&T conf call on Sep 4, 2013
	1.007 edited following comments received from UPC
	1.008 public release approval on Feb 17, 2014
Additional material	
Web location	
Implementation specifications version	Version 1.0
Comments	

IMPORTANT NOTICES

<u>**Disclaimer of Warranty**</u>

CO-LaN documents and publications include software in the form of *sample code.* Any such software described or provided by CO-LaN --- in whatever form --- is provided "as-is" without warranty of any kind. CO-LaN and its partners and suppliers disclaim any warranties including without limitation an implied warrant or fitness for a particular purpose. The entire risk arising out of the use or performance of any sample code --- or any other software described by the CAPE-OPEN Laboratories Network --- remains with you.

Copyright © 2014 CO-LaN and/or suppliers. All rights are reserved unless specifically stated otherwise.

CO-LaN is a non for profit organization established under French law of 1901.

<u>**Trademark Usage**</u>

Many of the designations used by manufacturers and seller to distinguish their products are claimed as trademarks. Where those designations appear in CO-LaN publications, and the authors are aware of a trademark claim, the designations have been printed in caps or initial caps.

Microsoft, Microsoft Word, Visual Basic, Visual Basic for Applications, Internet Explorer, Windows and Windows NT are registered trademarks and ActiveX is a trademark of Microsoft Corporation.

Netscape Navigator is a registered trademark of Netscape Corporation.

Adobe Acrobat is a registered trademark of Adobe Corporation.

SUMMARY

The Errata and Clarifications document for the Collection Common Interface specification 1.0 explains several aspects of the Collection item property including the lower bound of the Collection index, case insensitiveness for item name matching and the behaviour to be implemented when the item requested cannot be obtained.

FOREWORD

The errata and clarifications presented in this document have been discussed and developed by the CO-LaN Methods & Tools Special Interest Group. Note that the approved interface specification document version 1.0 is to be followed when implementing CAPE-OPEN interfaces.

CLARIFICATIONS

1.   Collection Item Property

1.1   Lower bound for index

Context: The *ICapeCollection.Item* method accepts a *CapeVariant* argument to identify the Collection member to be returned. The specification indicates that the value can be either a *CapeLong* (32-bit) integer or a *CapeString*. In the case of an integer-valued index, the Collection member is returned by position in the Collection.

Issue: Historically, integer-valued indices used in CAPE-OPEN Collections have had a lower bound of one (1). This usage was reinforced by the CAPE-OPEN Tester Suite that checked the implementation of the *ICapeCollection* of Parameters and Ports for Unit Operations. Presently, PMCs supporting *ICapeCollection* use one (1) as the lower bound of the Collection's indices. On the other hand a number of PMEs check systematically if an item index of zero (0) is supported for Parameters or ports of Unit Operations.

Recommendation (to appear in description of argument id within Arguments table): for the purposes of accessing a CAPE-OPEN Collection using the *ICapeCollection.Item* method, the index of the first item in the Collection obtained using an integer-valued *CapeVariant* must be one (1). The valid range for integer-valued items in the Collection is 1 to *ICapeCollection.Count*, inclusive.

1.2   Item matching by name is case insentive

Context: The *ICapeCollection.Item* method accepts a *CapeVariant* argument to identify the Collection member to be returned. The specification indicates that the value can be either a CapeLong (32-bit) integer or a CapeString. Use of a *CapeString* index value returns the Collection member with an *ICapeIdentification.ComponentName* property that matches the requested index.

Issue: The *ICapeCollection.Item* method does not specify whether the string value is case sensitive or not. Some PMEs do not distinguish case differences in the *ICapeIdentification.ComponentName* of a PMC.

Recommendation (to appear in description of argument id within Arguments table): When a Collection member is requested by name using the *ICapeCollection.Item* method with a string argument, the string matching is case insensitive.

1.3   Error condition when index requested is not corresponding to any item

Context: when an item is requested from a Collection using the *ICapeCollection.Item* method with an invalid argument, the method will raise an *ECapeInvalidArgument* error condition. This condition occurs when an integer-valued index outside the range of 1 to *ICapeCollection.Count* is requested, or the string argument does not match the *ICapeIdentification.ComponentName* of any members of the Collection. Historically, the error condition raised may have been *ECapeUnknown* or a generic platform error.

Recommendation (to appear in notes of Item method): in the case of an out of bounds index, *ECapeOutOfBounds* should preferably be returned or else *ECapeUnknown*. In the case of a wrong parameter name, *ECapeBadArgument* should preferably be returned or else *ECapeUnknown*. As with all routines that return an *ICapeInterface*, if an error is raised, the caller should not assume that the returned value is valid and

the caller should not attempt to release the returned interface. If the method does not return an error, the caller may assume that the method returns a valid (non NULL) interface.

2. Naming of Collection members

Context: the name of a Collection member used by the *ICapeCollection.Item* method to identify an item is defined by the *ICapeIdentificationComponentName* property. Each CAPE-OPEN object is required to implement *ICapeIdentification* interface, and provide a value for the *ICapeIdentification.ComponentName* property for the CAPE-OPEN object.

Issue: there is no mechanism to ensure that each CAPE-OPEN object within a CAPE-OPEN Collection has a unique *ICapeIdentification.ComponentName* property.

Recommendation: the Collection needs to ensure that the *ICapeIdentification.ComponentName* property of each object within the Collection is unique. At a minimum, the Collection needs to check that the names are unique each time an object is added to the Collection, or when the *ICapeIdentification.ComponentName* property of a Collection member is changed.