

ACATS Modification List for ACATS 4.0

Version 4.0J

December 10, 2015

What is this?

This is the ACATS modification list for ACATS 4.0. This document includes test corrections, test removals, and new tests for ACATS 4.0. ACATS 4.0 is the current test suite for Ada, covering the third edition of Ada, ISO/IEC 8652:2012 (usually known as "Ada 2012").

ACATS Modification List version 3.1K for ACATS 3.1 is the current list for that test suite. ACATS 3.1 should be used for testing implementations that support Amendment 1 (ISO/IEC 8652:1995/AMD 1:2007, often known as "Ada 2005"), and will continue to be maintained as necessary for the foreseeable future.

ACATS Modification List version 2.6D for ACATS 2.6 is the current list for that test suite. ACATS 2.6 should be used for testing implementations of the second edition of Ada, ISO/IEC 8652:1995 (usually known as "Ada 95"); it also will continue to be maintained as necessary for the foreseeable future.

Changes from the last list:

Note that there is no modification list 4.0I, as I can be confused with 1.

Many tests changed status on October 1st, this list reflects those changes.

Test C416A02 required an incorrect trace.

New tests B457006, B457007, C457006, C457007, and B860001 check various syntax and resolution rules for conditional expressions.

New tests C457005 and C540003 check that case statements and expressions operate properly when given values that don't meet an ignored static predicate.

New tests BD11002 and CD11001 check visibility rules for aspect specifications.

Tests CD33001 and CD33002 were effectively renamed to CD30008 and CD30009 as the original names violate the ACATS naming conventions by referring to a non-existent subclause.

New tests CXB3023 and CXB3024 check that a value of an unchecked union type can be passed to C code and used as expected.

New tests posted between October 1, 2015 and December 31, 2015 will become required for conformity assessments on July 1, 2016. New tests posted between July 1, 2015 and September 30, 2015 will become required for conformity assessments on April 1, 2016. New tests posted between April 1, 2015 and June 30, 2015 will become required for conformity assessments on January 1, 2016. New tests posted before March 31, 2015 are now required for conformity assessments. All tests posted to modification lists for other versions of the ACATS are included in ACATS 4.0 if appropriate and thus are required for conformity assessment using ACATS 4.0. The effective dates of tests are marked in the lists.

Questions about this document? Contact the ACAA Technical Agent at agent@ada-auth.org.

2015 Corrigendum:

Beginning with ACATS Modification List 4.0B, we have included tests for corrections (including new features) in the upcoming 2015 Corrigendum. All of these corrections have already been approved by WG 9. All of these changes are technically corrections to Ada 2012; as such we will assume all such corrections are implemented for Ada 2012 compilers.

Notice:

The ACAA has created and maintains the Ada Conformity Assessment Test Suite for the purpose of conformity assessments conducted in accordance with the International Standard ISO/IEC 18009 - Ada: Conformity assessment of a language processor. This test suite should not be used to make claims of conformance unless used in accordance with ISO/IEC 18009 and any applicable ACAA procedures.

List explanation:

Test numbers given are for the basic test. Categories are “Withdrawn” (the test should not be processed for Ada conformity assessments); “Allowed Modification” (either the original test or the modified one should be processed for Ada conformity assessments); “Modified” (the modified test must be processed for Ada conformity assessments); “Pending New” (the test does not need to be processed for Ada conformity assessments, but will be added in the future); “New” (the test is processed for Ada conformity assessments).

ACATS version labels are provided with all new and modified tests. Use this label to retrieve the test from the ACATS VCS. See “How to get new and modified tests”.

In citations, line numbers are given relative to the start of a test file in the format of '@<line#,line#,...>'; if a test program comprises more than one file, then the particular file will be indicated by "fn", where "n" is the test-program name's file-designator numeral. E.g., "B38103E : @f1-31, ... " denotes test file b38103e1.ada. In cases where there are two or more rationales for different errors in a test program, these different rationales with their respective line citations are listed on separate lines. References to the Ada standards are given in the format:

<section|chapter>.<clause>[.<subclause>]:<paragraph>

These references are to the Ada standard as available at <http://www.adaic.org/resources/add-contentstandards/12rm/html/RM-TOC.html>.

Main list:

==== Core Language =====

Name	Category	Version label	(effective date; old version label)
C324001	[Modified]	VCS Label: A4_0E	Eliminated overlong lines.
C324002	[Modified]	VCS Label: A4_0F	[VCS Label A4_0E] Eliminated overlong lines. [VCS Label A4_0F] Replaced last test case (qualified expressions), as that check is required by the Corrigendum.
C324003	[Pending New]	New VCS Label: A4_0H	Required after 2016-04-01 [VCS Label A4_0E] Check that predicates are checked in the correct

order and that the specified exception is raised.
[VCS Label A4_0H] Missing function Name is defined.

C324004 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check predicate checks are not performed for static and dynamic subtype predicates when the assertion policy is Ignore, but memberships, the Valid attribute, and for loops continue to use the predicates -- case A: global assertion policy pragmas

C324005 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check predicate checks are not performed for static and dynamic subtype predicates when the assertion policy is Ignore, but memberships, the Valid attribute, and for loops continue to use the predicates -- case B: specific assertion policy pragmas

C390012 [Pending New] New VCS Label: A4_0H

Required after 2016-04-01

Check the operation of Ada.Tags.Generic_Dispatching_Constructor.

C391003 [New] VCS Label: A4_0E

[VCS Label A4_0D] Check that a type extension can be declared in a more nested scope than its parent type, and that overridden subprograms of that extension can be dispatched to.

[VCS Label A4_0E] Eliminate an overlong line.

B393011 [New] VCS Label: A4_0D

Check that an abstract routine of an abstract partial view overridden by a non-abstract routine in the private part requires overriding when it is inherited if the private part is not visible where it is inherited.

C3A1003 [New] VCS Label: A4_0E

[VCS Label A4_0D] Check that a dereference of an access-to-tagged-incomplete type can be passed as a parameter.

[VCS Label A4_0E] Eliminated overlong lines.

C3A1004 [New] VCS Label: A4_0E

[VCS Label A4_0D] Check that a parameter of a tagged incomplete type can be passed as a parameter.

[VCS Label A4_0E] Eliminated overlong lines.

B3A1008 [New] VCS Label: A4_0E

[VCS Label A4_0D] Check that a dereference of an access-to-incomplete type cannot be used as a prefix.

[VCS Label A4_0E] Eliminated overlong lines.

B3A1009 [New] VCS Label: A4_0E

[VCS Label A4_0D] Check that a dereference of an access-to-incomplete view cannot be used as a prefix. Check that a parameter of an incomplete view cannot be used as a prefix.

[VCS Label A4_0E] Eliminated overlong lines.

B3A1010 [New] VCS Label: A4_0D

Check that a dereference of an access-to-formal incomplete type cannot be used as a prefix. Check that a parameter of a tagged formal incomplete type cannot be used as a prefix.

C3A2004 [New] VCS Label: A4_0C

Check that the type of the prefix of 'Access is used to resolve

the expected type of the use of the attribute.

- B3A2017 [New] VCS Label: A4_0E
[VCS Label A4_0C] Check P'Access is not allowed in a generic body if P is declared in the generic (including in the parent of a child or in a formal package) and the access type is a formal type or outside of the generic.
[VCS Label A4_0E] Eliminated overlong lines.
- B415001 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check various rules associated with the specification of aspect Implicit_Dereference.
- C415001 [Modified] VCS Label: A4_0F
Expression @155-6 violates 6.4.1(6.17/3).
- B415002 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that the object denoted by a generalized reference cannot be used as a variable if the discriminant has an access-to-constant type.
- B416002 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that a generalized indexing is illegal in a variable context if no Variable_Indexing is specified and Constant_Indexing specifies a function returning an ordinary object.
- B416A01 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
When both Constant_Indexing and Variable_Indexing are specified, check that a generalized indexing is illegal if it is called in variable contexts when the prefix is a constant and Constant_Indexing specifies a function returning an ordinary object.
- C416A01 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that the name specified for a Constant_Indexing or Variable_Indexing aspect can refer to a set of overloaded functions, with two or more parameters, and that all of those functions can be used in a generalized indexing.
- C416A02 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
[VCS Label A4_0F] Check that the overloaded routines can be added to an inherited indexing aspect and the new routines can be used in a generalized indexing.
[VCS Label A4_0H] Correct package C416A02_A to be a child of the foundation.
[VCS Label A4_0J] Remove .K from line @211 so test reflects expected trace.
- C431002 [New] VCS Label: A4_0E
[VCS Label A4_0B] Check that a component association of others => <> in a record or extension aggregate may have any number of associated components, including none. (Case 2: Null records and extensions.)
[VCS Label A4_0E] Eliminated overlong line.

B431005 [New] VCS Label: A4_0B
Check that a component association with an expression in a record or extension aggregate that have more than one associated component is illegal if the expression is illegal for any one of the associated components.

B431006 [New] VCS Label: A4_0C
Check that if a variant part is not nested in an unselected variant, the value of the governing discriminant of a variant in a record aggregate cannot be non-static.

B43102A [Withdrawn] VCS Label: A4_0C
Replaced by B431006.

C433002 [New] VCS Label: A4_0E
[VCS Label A4_0B] Check that no array components can be outside of the index subtype for an array aggregate, even when the components are specified by <>.
[VCS Label A4_0E] Eliminated overlong lines.

B433003 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that an others choice is not allowed in an array aggregate that is used as the return expression of an expression function or the expression of a return statement if the result subtype of the function is unconstrained.

C433003 [New] VCS Label: A4_0E
[VCS Label A4_0B] Check that for each association with a box in an array aggregate, the component is initialized by default (scalar types with Default_Values).
[VCS Label A4_0E] Eliminated overlong lines.

C433004 [New] VCS Label: A4_0E
[VCS Label A4_0B] Check that for each association with a box in an array aggregate, the component is initialized by default (array types with Default_Component_Values).
[VCS Label A4_0E] Object 06 @136 expects the wrong result.
Eliminated overlong lines.

C433005 [New] VCS Label: A4_0E
[VCS Label A4_0B] Check that the constraint of the constrained array subtype of a function return is used to determine the bounds of an array aggregate with an others choice in the expression of a return statement.
[VCS Label A4_0E] Eliminated overlong lines.

C433006 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that the constraint of the constrained array subtype of a function return is used to determine the bounds of an array aggregate with an others choice in the return expression of an expression function.

B457002 [New] VCS Label: A4_0D
Check that 'others' must be last and alone in a case expression.

B457003 [New] VCS Label: A4_0D
Check that 'else' cannot be omitted for an if expression if the type of the expression is not a boolean type.

B457004 [New] VCS Label: A4_OE
[VCS Label A4_OD] Check that case choice coverage and staticness are enforced for case expressions.
[VCS Label A4_OE] Eliminated overlong lines.

B457005 [New] VCS Label: A4_OD
Check that all or none of the dependent expressions of a conditional expression can be dynamically tagged if the type of the expression is a specific tagged type.

C457005 [Pending New] New VCS Label: A4_OJ
Required after 2016-07-01
When the selecting expression of a case expression is a name with a static nominal subtype and has a static predicate, the static predicate is disabled, and the value of the selecting expression does not satisfy the predicate, check that the others choice is returned, or, if there is no others choice, that Constraint_Error is raised.

B457006 [Pending New] New VCS Label: A4_OJ
Required after 2016-07-01
Check that an if expression has to be surrounded by parentheses if it is not already surrounded by them.

C457006 [Pending New] New VCS Label: A4_OJ
Required after 2016-07-01
Check that overloaded functions can be resolved when they appear as dependent expressions in a conditional expression. Part 1: Enumeration literals.

B457007 [Pending New] New VCS Label: A4_OJ
Required after 2016-07-01
Check that if the type of a conditional expression is not determined by the resolution rules, it is illegal.

C457007 [Pending New] New VCS Label: A4_OJ
Required after 2016-07-01
Check that literals can be resolved when they appear as dependent expressions in a conditional expression.

B480002 [New] VCS Label: A4_OE
[VCS Label A4_OD] Check that an initialized allocator for an access type with a class-wide designated type does not create an object whose type is more local than the access type.
[VCS Label A4_OE] Eliminated overlong lines and tab characters.

B480003 [New] VCS Label: A4_OE
[VCS Label A4_OD] Check that an uninitialized allocator for an access type with a class-wide designated type does not create an object whose type is more local than the access type.
[VCS Label A4_OE] Eliminated overlong lines and tab characters.

C540002 [Modified] VCS Label: A4_OE
Eliminated overlong lines.

C540003 [Pending New] New VCS Label: A4_OJ
Required after 2016-07-01
When the selecting expression of a case statement is a name with a static nominal subtype and has a static predicate, the static predicate is disabled, and the value of the selecting expression does not satisfy the predicate, check that the others choice is

returned, or, if there is no others choice, that Constraint_Error is raised.

- C550001 [New] VCS Label: A4_0E
[VCS Label A4_0C] Check that for loops iterate properly over subtypes with static predicates.
[VCS Label A4_0E] Eliminated overlong lines.
- B551001 [New] VCS Label: A4_0E
Check legality rules for aspect Default_Iterator.
- B551002 [New] VCS Label: A4_0E
Check legality rules for aspect Iterator_Element.
- B552001 [New] VCS Label: A4_0E
Check legality rules for array component iterators from the 2015 Corrigendum.
- C552001 [New] VCS Label: A4_0E
Check correct operation of array component iterators.
- C552002 [New] VCS Label: A4_0E
Check correct operation of array component iterators where the array has convention Fortran.
- C552A01 [Modified] VCS Label: A4_0E
Eliminated violations of ACATS lexical rules.
- C552A02 [Modified] VCS Label: A4_0E
Eliminated violations of ACATS lexical rules.
- B552A04 [New] VCS Label: A4_0E
Check legality rules for container element iterators (most of the checked rules are from the 2015 Corrigendum).
- B552A05 [New] VCS Label: A4_0E
Check legality rules for generalized iterators from the 2015 Corrigendum.
- B640001 [New] VCS Label: A4_0E
[VCS Label A4_0C] Check that abstract nondispatching operations do not participate in resolution.
[VCS Label A4_0E] Eliminated overlong lines and tab characters.
- C640002 [New] VCS Label: A4_0E
[VCS Label A4_0C] Check that an abstract nondispatching operator provides a way to eliminate an unwanted predefined operator.
[VCS Label A4_0E] Eliminate overlong lines.
- B650005 [New] VCS Label: A4_0C
Check that the accessibility level of the return subtype indication of an extended return statement cannot be statically deeper than that of the function result if that result is class-wide.
- B650006 [New] VCS Label: A4_0C
Check that an extended return statement with a constant return object cannot omit the initializing expression.
- C650A02 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01

Check that the result of a function that returns a class-wide tagged type has the tag of the return expression, if the `return_subtype_indication` (if any) is class-wide.
Case A: limited types.

C650B01 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that the result of a function that returns a specific tagged type has the tag of the tagged result type, even if the return expression has a different tag.

C650B02 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that the result of a function that returns a class-wide tagged type has the tag of the return expression, if the `return_subtype_indication` (if any) is class-wide.
Case B: non-limited types.

C650B03 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that the result of a function that returns a class-wide tagged type has the tag of the `return_subtype_indication` if it is a specific tagged type.

B730009 [Modified] VCS Label: A4_0E
Completion of Priv12 @131 violates 7.3(10.1/3).

B730010 [New] VCS Label: A4_0E
Check 7.3(10.1/3).

B732001 [New] VCS Label: A4_0E
[VCS Label A4_0D] Check various resolution and legality rules for aspect `Type_Invariant`.
[VCS Label A4_0E] Length @80 should have return type `Natural`.
Changed `Is_Valid` to `Is_Clean` @95, 97, 146 so that the inherited routine doesn't come into play.

C732001 [New] VCS Label: A4_0E
[VCS Label A4_0D] Check that assertion checks are made after assignments to and calls with actuals of appropriate view conversions.
[VCS Label A4_0E] Eliminated overlong lines.

C732002 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check some differences between specific and class-wide type invariants.

C732A01 [New] VCS Label: A4_0E
[VCS Label A4_0D] Check that the failure of a `Type_Invariant` check raises `Assertion_Error` (constrained array).
[VCS Label A4_0E] Eliminated overlong lines.

C732A02 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that the failure of a `Type_Invariant` check raises `Assertion_Error` (part of unconstrained array).

C732B01 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that the failure of a `Type_Invariant` check raises

Assertion_Error (discriminated record).

C732B02 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that the failure of a Type_Invariant check raises
Assertion_Error (access-to-record).

B732C01 [New] VCS Label: A4_0E
[VCS Label A4_0D] Check that the aspect Type_Invariant can only
be specified on private types, private extensions, and full types
that complete one of the others.
[VCS Label A4_0E] Eliminate overlong lines.

C732C01 [New] VCS Label: A4_0E
[VCS Label A4_0D] Check that the failure of a Type_Invariant check
raises Assertion_Error (tagged record).
[VCS Label A4_0E] Eliminated overlong lines.

B732C02 [New] VCS Label: A4_0E
[VCS Label A4_0D] Check that the aspect Type_Invariant'Class can
only be specified on private types, private extensions, and
interfaces.
[VCS Label A4_0E] Eliminate overlong lines.

B750A01 [Allowed Modification] New VCS Label: A4_0H Org VCS Label: A4_0
Required after 2016-01-01
The qualification @129 should be for Lim_Ext, not Lim_Rec.

B750A02 [Allowed Modification] New VCS Label: A4_0H Org VCS Label: A4_0
Required after 2016-01-01
The qualification @131 should be for Lim_Ext, not Lim_Rec.

B750A04 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that only legal expressions are allowed in the qualified
expression of an allocator for a limited object.

B750A05 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that only legal expressions are allowed in the
initialization expression of an extended return statement for a
limited object.

B750A06 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that only legal expressions are allowed in the expression
of a simple return statement for a function that returns a
limited type.

B750A07 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that only legal expressions are allowed in the return
expression of an expression function that returns a limited type.

B750A08 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
Check that only legal expressions are allowed as the dependent
expressions of a conditional expression used as the initialization
expression of an object declaration of a limited type.

B750A09 [Pending New] New VCS Label: A4_0H

Required after 2016-04-01

Check that only legal expressions are allowed as the dependent expressions of a conditional expression used as the default expression of a component declaration of a limited type.

B750A10 [Pending New] New VCS Label: A4_0H

Required after 2016-04-01

Check that only legal expressions are allowed as the dependent expressions of a conditional expression used as the qualified expression of an initialized allocator for a limited object.

B750A11 [Pending New] New VCS Label: A4_0H

Required after 2016-04-01

Check that only legal expressions are allowed as the dependent expressions of a conditional expression used as the initializing expression of an extended return statement that returns a limited type.

B750A12 [Pending New] New VCS Label: A4_0H

Required after 2016-04-01

Check that only legal expressions are allowed as the dependent expressions of a conditional expression used as the expression of a simple return statement that returns a limited type.

B750A13 [Pending New] New VCS Label: A4_0H

Required after 2016-04-01

Check that only legal expressions are allowed as the dependent expressions of a conditional expression used as the return expression of an expression function that returns a limited type.

C760014 [New] VCS Label: A4_0A

Check that Ada.Finalization is declared pure.

C760015 [New] VCS Label: A4_0E

[VCS Label A4_0D] Check that a controlled type can be declared within a subprogram.

[VCS Label A4_0E] Eliminate overlong lines.

C831001 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check that overriding indicators do not change the dynamic effect of calls.

B831004 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check that overriding indicators are properly enforced for types derived from a generic formal type. (Nested generic specification cases.)

B831005 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check that overriding indicators are properly enforced for types derived from a generic formal type. (Library unit and body cases.)

B840002 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check a package use clause cannot denote a limited view or a non-package.

B840003 [Pending New] New VCS Label: A4_0F

Required after 2016-01-01

Check a use all type clause does not make non-primitives or class-wide operations that aren't directly declare where a primitive operation would be visible.

- B860001 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Check that the selecting expression of a case statement or case expression cannot be resolved if information from the case choices is needed to resolve it.
- BA12017 [New] VCS Label: A4_0E
[VCS Label A4_0C] Check that a limited with clause is not allowed on a descendant of the named package.
[VCS Label A4_0E] Eliminated overlong line.
- BA15003 [New] VCS Label: A4_0C
Check that a library unit pragma must apply to a library unit.
- CA21002 [New] VCS Label: A4_0E
Check that Pure and Preelaborate aspects can be specified with an object declared in another package.
- CB30001 [New] VCS Label: A4_0B
Check that the optional message parameter is supported in a raise statement, and that the message can be retrieved with Exception_Message.
- CB30002 [New] VCS Label: A4_0E
[VCS Label A4_0B] Check that the optional message parameter is supported in a raise expression, and that the message can be retrieved with Exception_Message.
[VCS Label A4_0E] Eliminated overlong lines and tab characters.
- CC30003 [Modified] VCS Label: A4_0E
Eliminated overlong lines.
- CC30004 [Modified] VCS Label: A4_0E
Eliminated overlong lines and tab characters.
- CC51010 [New] VCS Label: A4_0D
[VCS Label A4_0D] Check that a generic unit with a tagged incomplete formal type can be instantiated with an incomplete view of a type from a limited withed package, and that operations in the instance can be used normally when the full view of the type is visible.
[VCS Label A4_0E] Added constructor function to generic, as Elem'Unchecked_Access is illegal when Elem is incomplete. Moved body of CC51010_2 so it compiles after another package on which it depends.
- BC60001 [New] VCS Label: A4_0C
Test rules associated with formal subprogram defaults of null.
- CC60001 [Pending New] New VCS Label: A4_0G
Required after 2016-01-01
Check that calls on a formal abstract subprogram are dispatching.
- BC60002 [Pending New] New VCS Label: A4_0G
Required after 2016-01-01
Test rules associated with the controlling type of a formal abstract subprogram.

BC60003 [Pending New] New VCS Label: A4_0G
Required after 2016-01-01
Check that the actual subprogram for a formal abstract subprogram must be a dispatching operation of the controlling type.

BC60004 [Pending New] New VCS Label: A4_0G
Required after 2016-01-01
Check that a call on a formal abstract subprogram must be dispatching.

CD30006 [Modified] VCS Label: A4_0E
Eliminated overlong lines and tab characters.

CD30007 [Modified] VCS Label: A4_0E
Eliminated overlong lines.

CD30008 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Renaming of previous test CD33001 (there is no subclause 13.3.3).
The test is unchanged other than renaming the units involved and fixing an overlong line.

CD30009 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Renaming of previous test CD33002 (there is no subclause 13.3.3).
The test is unchanged other than renaming the units involved and fixing an overlong line.

CC33001 [Withdrawn] VCS Label: A4_0J
Renamed to CD30008, as there is no subclause 13.3.3.

CC33002 [Withdrawn] VCS Label: A4_0J
Renamed to CD30009, as there is no subclause 13.3.3.

CD11001 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Check that appropriate entities are visible in an aspect specification.

BD11002 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Check that entities in the private part of a package are not visible in aspect specifications given in the visible part.

BDB3A01 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test resolution and legality rules for pragma and aspect Default_Storage_Pool.
[VCS Label A4_0E] Declaration of Pool @53 is missing the TC_Id discriminant.

CDB3A01 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that aspect Default_Storage_Pool changes the default storage pool for a generic instantiation.
[VCS Label A4_0E] Eliminated overlong line.

BDD2004 [Modified] VCS Label: A4_0E
[VCS Label A4_0B] Error @229 is legal by AI12-0121-1.
[VCS Label A4_0E] Eliminated overlong lines.

BDD2005 [Modified] VCS Label: A4_0E

[VCS Label A4_0B] Replaced test cases waiting on AI12-0106-1 (now approved) and made changes required by AI12-0121-1.
[VCS Label A4_0E] Eliminated overlong lines.

- CXA4037 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check Ada.Strings.Fixed.Find-Token against rule from 2001 Corrigendum.
- CXA5016 [New] VCS Label: A4_0E
[VCS Label A4_0C] Check operation of attribute Machine_Rounding.
[VCS Label A4_0E] Eliminated overlong lines.
- CXAA021 [New] VCS Label: A4_0C
Check file operations of Text_IO.Bounded_IO.
- CXAA022 [New] VCS Label: A4_0C
Check file operations of Text_IO.Unbounded_IO.
- CXAB002 [New] VCS Label: A4_0C
Check file operations of Wide_Text_IO.Wide_Bounded_IO.
- CXAB003 [New] VCS Label: A4_0C
Check file operations of Wide_Wide_Text_IO.Wide_Wide_Bounded_IO.
- CXAB004 [New] VCS Label: A4_0C
Check file operations of Wide_Text_IO.Wide_Unbounded_IO.
- CXAB005 [New] VCS Label: A4_0C
Check file operations of Wide_Wide_Text_IO.Wide_Wide_Unbounded_IO.
- CXAC008 [Pending New] New VCS Label: A4_0H
Required after 2016-04-01
[VCS Label A4_0C] Check that Stream_IO files opened with mode Out_File are not truncated by Close or Reset.
[VCS Label A4_0D] Corrected required file size in subtest 8, and various failure messages.
[VCS Label A4_0E] Eliminated overlong lines.
[VCS Label A4_0H] Corrected a failure message.
- CXAG001 [New] VCS Label: A4_0E
Check that Ada.Directories can be used to create and delete directories.
- CXAG002 [New] VCS Label: A4_0E
Check that Ada.Directories.Hierarchical_File_Names exists and works as expected.
- CXAH001 [New] VCS Label: A4_0E
[VCS Label A4_0D] Test that package Environment_Variables can be used to read environment variables.
[VCS Label A4_0E] Added missing Applicability Criteria.
- CXAH002 [New] VCS Label: A4_0E
[VCS Label A4_0D] Test that package Environment_Variables can be used to modify environment variables.
[VCS Label A4_0E] Added missing Applicability Criteria.
- CXAH003 [New] VCS Label: A4_0E
[VCS Label A4_0D] Test the two parameter Value in package Environment_Variables.

[VCS Label A4_0E] Added missing Applicability Criteria.

BXAI001 [New] VCS Label: A4_0B
Test that package Hashed_Maps does not allow reverse iterators.

BXAI002 [New] VCS Label: A4_0B
Test that package Hashed_Sets does not allow reverse iterators.

BXAI003 [New] VCS Label: A4_0B
Test that package Bounded_Hashed_Maps does not allow reverse iterators.

BXAI004 [New] VCS Label: A4_0B
Test that package Bounded_Hashed_Sets does not allow reverse iterators.

BXAI005 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that package Hashed_Sets does not allow direct writing of set elements.
[VCS Label A4_0E] Eliminated overlong lines.

BXAI006 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that package Ordered_Sets does not allow direct writing of set elements.
[VCS Label A4_0E] Eliminated overlong lines.

BXAI007 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that package Bounded_Hashed_Sets does not allow direct writing of set elements.
[VCS Label A4_0E] Eliminated overlong lines.

BXAI008 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that package Bounded_Ordered_Sets does not allow direct writing of set elements.
[VCS Label A4_0E] Eliminated overlong lines.

BXAIA01 [New] VCS Label: A4_0B
Test that package Indefinite_Hashed_Maps does not allow reverse iterators.

BXAIA02 [New] VCS Label: A4_0B
Test that package Indefinite_Hashed_Set does not allow reverse iterators.

BXAIA03 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that package Indefinite_Hashed_Sets does not allow direct writing of set elements.
[VCS Label A4_0E] Eliminated overlong lines.

BXAIA04 [New] VCS Label: A4_0E
[VCS Label A4_0B] Test that package Indefinite_Ordered_Sets does not allow direct writing of set elements.
[VCS Label A4_0E] Eliminated overlong lines.

CXAI027 [Modified] VCS Label: A4_0B
Corrected the name of the package that is reported as being tested.

BXB3001 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(8/2).

BXB3002 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(9/3).

BXB3003 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(11/2).

BXB3004 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check that rules for the application of aspect Unchecked_Union are enforced.

CXB3019 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(23/2).

CXB3020 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(24/2).

CXB3021 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(25/2).

CXB3022 [Pending New] New VCS Label: A4_0F
Required after 2016-01-01
Check B.3.3(26/2).

CXB3023 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Check that a value of an unchecked union type (defined with pragma Unchecked_Union) can be passed to a C function.

CXB3024 [Pending New] New VCS Label: A4_0J
Required after 2016-07-01
Check that a value of an unchecked union type (defined with aspect Unchecked_Union) can be passed to a C function.

==== Annex C =====

[None.]

==== Annex D =====

[None.]

==== Annex E =====

[None.]

==== Annex F =====

[None.]

==== Annex G =====

[None.]

==== Annex H ====

[None.]

==== Support ====

F416A00 [Pending New] New VCS Label: A4_OF
Required after 2016-01-01
Foundation for tests B416A01, C416A01, and C416A02.

F552A00 [Allowed Modification] New VCS Label: A4_OH Org VCS Label: A4_OE
Required after 2016-01-01
The Constant_Reference routine @237 and @388 needs to return a
Bingo_Call value rather than a reference, (and then was renamed)
in order to prevent an accessibility failure.

F650B00 [Pending New] New VCS Label: A4_OH
Required after 2016-04-01
Foundation for tests C650B01, C650B02, and C650B03.

F732A00 [New] VCS Label: A4_OE
[VCS Label A4_OD] Foundation for test C732A01.
[VCS Label A4_OE] Eliminated overlong lines.

F732A00 [Allowed Modification] New VCS Label: A4_OF Org VCS Label: A4_OE
Required after 2016-01-01
[VCS Label A4_OF] Added types and operations to support test
C732A02. They are only needed for that test.

F732B00 [Pending New] New VCS Label: A4_OF
Required after 2016-01-01
Foundation for tests C732B01 and C732B02.

F732C00 [New] VCS Label: A4_OE
[VCS Label A4_OD] Foundation for tests B732C01, B732C02, and
C732C01.
[VCS Label A4_OE] Eliminated overlong lines.

FDB3A00 [New] VCS Label: A4_OE
[VCS Label A4_OB] Foundation for test BDB3A01 and CDB3A01.
[VCS Label A4_OE] Eliminated overlong line.

IMPDEF [Modified] VCS Label: A4_OE
[VCS Label A4_OD] Added constants for the names of environment
variables used by tests CXAH001, CXAH002, and CXAH003. Also
removed obsolete .AW file constants (these are not used as test
source has been provided in UTF-8 format since the release of
ACATS 3.0). The change is only needed for those tests, it has no
effect on any other tests.
[VCS Label A4_OE] Added a constant and function used by tests
CXAG001 and CXAG002. The change is only needed for those tests,
it has no effect on any other tests.

IMPDEF [Allowed Modification] New VCS Label: A4_OJ Org VCS Label: A4_OE
Required after 2016-04-01
[VCS Label A4_OJ] Added constants for the external names for the
imported routines in tests CXB3023 and CXB3024. The change is only
needed for those tests, it has no effect on any other test.

How to get new and modified tests:

The official home of the ACATS test suite is the ACAA web site. The site can be accessed at www.ada-auth.org.

The complete set of official new and modified tests are available in the Mod_4_0J.Zip and Tar files on the ACAA web site.

Individual new and modified tests can be retrieved from the ACATS VCS. Simply navigate to the appropriate directory (folder), and select the test you want. Select the revision you want (usually the most recent) and it will be retrieved for you. Tests are stored in directories that correspond to the test's prefix. For instance, the test CD92001.A will be found in the CD directory. You can also browse the differences between versions of a test. [In case of a conflict, the files in the ACATS VCS should be considered the 'official' versions.]

If you cannot get the files from the official site or you cannot get the files in a format appropriate for your processor, please contact your ACAL.

The baseline ACATS 4.0 is available in various formats on the ACAA web site.

Summary lists:

Withdrawn Tests

(Refer to the main list for details)

B43102A
CD33001
CD33002

Allowed Modification Tests

(Refer to the main list for details)

B750A01	2016-01-01	
B750A02	2016-01-01	
IMPDEF	2016-04-01	[Only needed for new tests CXB3023 and CXB3024]

Modified Tests

(Refer to the main list for details)

B730009
BDD2004
BDD2005
C324001
C324002
C415001
CXAI027
CC30003
CC30004
CD30006
CD30007

IMPDEF [VCS_Label_A4_0E]

New Tests

(Refer to the main list for details)

B393011
B3A1008
B3A1009
B3A1010
B3A2017
B431005
B431006
B457002
B457003
B457004
B457005
B480002
B480003
B551001
B551002
B552001
B552A04
B552A05
B640001
B650005
B650006
B730010
B732001
B732C01
B732C02
BA12017
BA15003
BC60001
BDB3A01
BXAI001
BXAI002
BXAI003
BXAI004
BXAI005
BXAI006
BXAI007
BXAI008
BXAIA01
BXAIA02
BXAIA03
BXAIA04
C391003
C3A1003
C3A1004
C3A2004
C431002
C433002
C433003
C433004
C433005
C550001
C552001

C552002
C640002
C732001
C732A01
C732C01
C760014
C760015
CA21002
CB30001
CB30002
CC51010
CDB3A01
CXA5016
CXAA021
CXAA022
CXAB002
CXAB003
CXAB004
CXAB005
CXAG001
CXAG002
CXAH001
CXAH002
CXAH003
F732A00
F732C00
FDB3A00

Pending New Tests

(Refer to the main list for details)

B415001	2016-01-01
B415002	2016-01-01
B416002	2016-01-01
B416A01	2016-01-01
B433003	2016-01-01
B457006	2016-07-01
B457007	2016-07-01
B750A04	2016-04-01
B750A05	2016-04-01
B750A06	2016-04-01
B750A07	2016-04-01
B750A08	2016-04-01
B750A09	2016-04-01
B750A10	2016-04-01
B750A11	2016-04-01
B750A12	2016-04-01
B750A13	2016-04-01
B831004	2016-01-01
B831005	2016-01-01
B840002	2016-01-01
B840003	2016-01-01
B860001	2016-07-01
BC60002	2016-01-01
BC60003	2016-01-01
BC60004	2016-01-01
BD11002	2016-07-01

BXB3001	2016-01-01
BXB3002	2016-01-01
BXB3003	2016-01-01
BXB3004	2016-01-01
C324003	2016-04-01
C324004	2016-01-01
C324005	2016-01-01
C390012	2016-04-01
C416A01	2016-01-01
C416A02	2016-07-01
C433006	2016-01-01
C457005	2016-07-01
C457006	2016-07-01
C457007	2016-07-01
C540003	2016-07-01
C650A02	2016-04-01
C650B01	2016-04-01
C650B02	2016-04-01
C650B03	2016-04-01
C732002	2016-01-01
C732A02	2016-01-01
C732B01	2016-01-01
C732B02	2016-01-01
C831001	2016-01-01
CC60001	2016-01-01
CD11001	2016-07-01
CD30008	2016-07-01
CD30009	2016-07-01
CXA4037	2016-01-01
CXAC008	2016-04-01
CXB3019	2016-01-01
CXB3020	2016-01-01
CXB3021	2016-01-01
CXB3022	2016-01-01
CXB3023	2016-07-01
CXB3024	2016-07-01
F416A00	2016-01-01
F650B00	2016-04-01
F732B00	2016-01-01