ACATS Modification List for ACATS 3.1
Version 3.1Y
December 31, 2019

What is this?
This is the ACATS modification list for ACATS 3.1. This document includes test corrections, test removals, and new tests for ACATS 3.1. ACATS 3.1 is the current test suite for the version of Ada that became current with the approval and publication of Amendment 1 by ISO as ISO/IEC 8652:1995/AMD 1:2007 (often known as "Ada 2005").

ACATS Modification List version 2.6G for ACATS 2.6 is the current list for that test suite. ACATS 2.6 should be used for testing Ada 95 implementations, and will continue to be maintained as necessary for the foreseeable future.

ACATS Modification List version 4.1P is the current list for ACATS 4.1. ACATS 4.1 should be used for testing the third edition of Ada, ISO/IEC 8652:2012 (usually known as "Ada 2012").

Changes from the last list:
Various tests changed status on October 1st. See the list for details.

Tests B32201A, B33101A, B38103A, B38103B, B38103C, B38103D, B38103E, B48002A, B48002G, B48003B, B48003E, B56001A, B57001B, B660002, B66001A, B66001B, BC51019, BC51020, BC51B02, BC51C01, and BC51C02 were modified to add error location indicators (so that the grading tools can reliably grade these tests).

Test B33001A was modified to add error location indicators (so that the grading tools can reliably grade these tests), and to modify the third test case to be more different than the second test case. (Note that the lack of an OK tag on T4A suggests that it was intended to be part of the cycle; without the change, there is no mixed array and record test case.)

Note: Some Allowed Modification tests have a required date of “ACATS 3.2”. These are tests that don't have substantive changes, but rather change comments (including error locations for B-Tests). For ACATS users that are using traditional results comparison rather than the ACATS grading tools, changing these tests may cause work without benefit (since the tests don't have significant changes). As such, we're allowing the use of the old versions of the tests until the time when the next ACATS version (ACATS 3.2) of this series is issued (there are no current plans for such a version).

New tests posted between October 1, 2019 and December 31, 2019 will become required for conformity assessments on July 1, 2020. New tests posted between July 1, 2019 and September 30, 2019 will become required for conformity assessments on April 1, 2020. New tests posted between April 1, 2019 and June 30, 2019 will become required for conformity assessments on January 1, 2020. New tests posted before April 1, 2019 are required for conformity assessments. The effective dates of tests are marked in the lists.

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

Ada 2012 compatibility:
This test suite is designed for Ada 2005. ACATS 4.1 should be used to test Ada 2012 implementations. There are a significant number of tests in ACATS 3.1 that are incompatible with the requirements of Ada 2012.

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 consolidated Ada standard as available at http://www.adaic.org/resources/add-contentstandards/05rm/html/RM-TOC.html.

Main list:

====== Core Language ======

Name     Category   Version label     (effective date; old version label)
Short description.

B32201A [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

B33001A [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required after 2020-04-01
Added error location indicators, added additional possible error
locations, restored likely intended test for Set2.

B33101A [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

C352001 [Modified] VCS Label: A3_1Q
Corrected Failed message @267 to reflect intended result.
Test basic requirements for the Mod attribute.

Test basic operation of the Mod attribute.

Added additional error locations to directly support common error reporting strategies.

Added additional error locations to directly support common error reporting strategies.

Added additional error locations to directly support common error reporting strategies.

Corrected error tags @70,76,82,88 so “optional” and “error” are on the same line, as required by the test standards.

Added error location indicators.

Added error location indicators.

Added error location indicators.

Added error location indicators.

Added error location indicators.

Replaced by tests B3A1007 and B3A1A05 (and existing tests B3A1A01 and B3A1A02).

Check the operation of Ada.Tags.Generic_Dispatching_Constructor.

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.

Eliminated an overlong line.

Revised to use Possible Error tags, and added error location
indicators.

B392003 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B392004 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B392005 [Modified] VCS Label: A3_1R
Revised to use Possible Error tags, and added error location indicators.

B393001 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B393005 [Modified] VCS Label: A3_1B
Declaration @114 is illegal because it inherits an abstract function that is not overridden, as described in AI05-0097-1.

B393007 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies. Fixed overlong lines.

B393010 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies. Fixed overlong lines.

B393011 [New] VCS Label: A3_1E
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.

C3A0030 [New] VCS Label: A3_1L
Check that an access discriminant is only null excluding if it includes a null exclusion.

B3A1001 [Modified] VCS Label: A3_1M
[VCS Label A3_1B] Type UT1_@78 is supposed to be tagged incomplete, else it's the same as type U1.
[VCS Label A3_1M] Split Pack6 so possible error sets don't intersect; fixed overlong lines.

B3A1002 [Modified] VCS Label: A3_1M
[VCS Label A3_1B] Type UT1_@77 is supposed to be tagged incomplete, else it's the same as type U1.
[VCS Label A3_1M] Split Pack6 so possible error sets don't intersect; fixed overlong lines.

C3A1003 [New] VCS Label: A3_1E
Check that a dereference of an access-to-tagged-incomplete type can be passed as a parameter.

C3A1004 [New] VCS Label: A3_1E
Check that a parameter of a tagged incomplete type can be passed as a parameter.

B3A1007 [New] VCS Label: A3_1B
Check that constraints and exclusions aren't allowed on an incomplete type in access type and subtype declarations.

B3A1008 [New] VCS Label: A3_1F
[VCS Label A3_1E] Check that a dereference of an access-to-incomplete type cannot be used as a prefix.
[VCS Label A3_1F] Eliminated overlong lines.

B3A1009 [New] VCS Label: A3_1F
[VCS Label A3_1E] 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 A3_1F] Eliminated overlong lines.

B3A1A05 [New] VCS Label: A3_1B
Check that generic formal objects cannot be declared with an incomplete view.

C3A2004 [New] VCS Label: A3_1D
Check that the type of the prefix of 'Access is used to resolve the expected type of the use of the attribute.

B3A2007 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B3A2008 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B3A2009 [Modified] VCS Label: A3_1R
Removed tabs to meet test format standards; added error location indicators.

B3A2011 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B3A2012 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B3A2016 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error reporting strategies.

B3A2017 [New] VCS Label: A3_1F
[VCS Label A3_1D] 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 A3_1F] Eliminated overlong lines.
PPA and PFA trigger the issue of AI12-0289-1; explicit null exclusions were added to the parameters to avoid it.

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.

Replaced by B431006.

Check that Resolution and Legality Rules for the universal access equality operator are enforced.

Check that "=" for language-defined private types behaves as if it is predefined. Part 1: core types.

Check that "=" for language-defined private types behaves as if it is predefined. Part 2: types of the Ada.Strings subsystem.

Check that "=" for language-defined private types behaves as if it is predefined. Part 3: Annex C types.

Check that modular adding operators wrap around rather than raising Constraint_Error.

Corrected casing of error tags and fixed overlong lines.

Check that Constraint_Error is raised appropriately when converting access-to-classwide to another access type.

Check that Constraint_Error is raised appropriately when converting access-to-limited-classwide to another access type.

Check that an initialized allocator for a class-wide type does an accessibility check on the tag of the initializing object. Changed mode to "in" (instead of "in out") @103 & 144 as those parameters are used in functions, making that an Ada 2012 feature, and this is an Ada 2005 test.

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. Eliminate overlong lines and tab characters.
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.

Eliminate overlong lines and tab characters.

Corrected test objective and test description; the test code is unmodified.

Check that an allocator is illegal if the access type has a Storage_Size that is statically zero.

Added error location indicators.

Added additional error locations to directly support common error reporting strategies.

Corrected comment @49 not to have the form of an error tag.

Check that case statement coverage rules are enforced for modular types.

Added additional error locations to directly support common error reporting strategies.

Check that the selecting_expression of a case statement can be resolved using the requirement that it be any discrete type.

Eliminated overlong lines.

Added additional error locations to directly support common error reporting strategies.

Added additional error locations to directly support common error reporting strategies.

Added additional error locations to directly support common error reporting strategies.
reporting strategies.

B54B02C [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B54B02D [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B54B04A [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B54B04B [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B54B05A [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B55B12B [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B55B12C [Modified] VCS Label: A3_1R
  Added additional error locations to directly support common error reporting strategies.

B56001A [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
  Required in ACATS 3.2
  Added error location indicators.

B57001B [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
  Required in ACATS 3.2
  Added error location indicators.

C620001 [New] VCS Label: A3_1B
  Check that elementary types are passed by-copy.

B62001C [Modified] VCS Label: A3_1M
  Changed error @35 to be optional, as the syntax error @33 is required (and the comment that only one error need be detected is bogus).

B62001D [Modified] VCS Label: A3_1M
  Changed error @41 to be optional, as the syntax error @33 is required (and the comment that only one error need be detected is bogus).

C62003A [Withdrawn] Org VCS Label: A3_1B
  Replaced by test C620001.

B631001 [Pending New] New VCS Label: A3_1X
  Required after 2020-04-01
  Check that a prefixed view is intrinsic.
Check that abstract nondispatching operations do not participate in resolution.

Eliminate overlong lines and tab characters.

Check that an abstract nondispatching operator provides a way to eliminate an unwanted predefined operator.

Eliminate overlong lines.

Check that a return statement cannot apply to a nonreturning procedure, and that a renames-as-body of a nonreturning procedure has to be a nonreturning procedure (pragma No_Return version).

Check that a nonreturning procedure can raise an exception and raises Program_Error otherwise (pragma No_Return version).

Check that a matching "/=" is defined for an explicitly declared "=" operator with a result of Boolean.

Added error location indicators.

Required in ACATS 3.2

Added error location indicators.

Added error location indicators.

Corrected format of error tags; added location indicators.

Check that a tagged partial view and its completion must have the same set of interfaces.

Completion of Priv12 @131 violates 7.3(10.1/3).
Check 7.3(10.1/3).

B730011 [New] VCS Label: A3_1P
Test various rules about completions of private types and extensions; replaces test B74001A.

B730012 [New] VCS Label: A3_1P
Test various rules about completions of private types and extensions in generic units; replaces test B74001B.

B730013 [New] VCS Label: A3_1Q
Test operations available for a private type; replaces test B74203E.

B73001A [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B73001B [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B73001C [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B73001D [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B73001E [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B73001F [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B740003 [Modified] VCS Label: A3_1S
[VCS Label A3_1M] Modified criteria @75 so that it doesn't look like an error tag to the grading tools; also fixed an overlong line.
[VCS Label A3_1S] Modified grading to document that processing the main is optional. Added error location indicators.

B74001A [Withdrawn] VCS Label: A3_1P
Replaced by B730011.

B74001B [Withdrawn] VCS Label: A3_1P
Replaced by B730012.

B74203E [Withdrawn] VCS Label: A3_1Q
Replaced by B730013.

B750A01 [Modified] VCS Label: A3_1J
The qualification @129 should be for Lim_Ext, not Lim_Rec.

B750A02 [Modified] VCS Label: A3_1J
The qualification @131 should be for Lim_Ext, not Lim_Rec.

B750A04 [New] VCS Label: A3_1J
Check that only legal expressions are allowed in the qualified expression of an allocator for a limited object.

B750A06 [New] VCS Label: A3_1J
Check that only legal expressions are allowed in the expression of a simple return statement for a function that returns a limited type.

C760015 [New] VCS Label: A3_1L
[VCS Label A3_1E] Check that a controlled type can be declared within a subprogram.
[VCS Label A3_1F] Eliminated overlong lines.
[VCS Label A3_1L] Corrected one of the potential results for Subtest2(4) to be “Iafcf”, not “IafcF”.

C760A02 [New] VCS Label: A3_1R
Checks that functions that return immutably limited types are built-in-place.

B83027B [Modified] VCS Label: A3_1R
The discriminant @90 needs a default expression as all or none of the discriminants of a type need defaults. Also added error location indicators.

B83027D [Modified] VCS Label: A3_1R
Corrected bad error tag, and added error location indicators.

B83E01B [Modified] VCS Label: A3_1R
Removed possible error cascades, reformatted some error tags, and added error location indicators.

B831004 [New] VCS Label: A3_1G
Check that overriding indicators are properly enforced for types derived from a generic formal type. (Nested generic specification cases.)

B83F02C [Modified] VCS Label: A3_1M
Updated many optional error markers to match ACATS standards.

B840002 [New] VCS Label: A3_1G
Check a package use clause cannot denote a limited view or a non-package.

B851004 [Modified] VCS Label: A3_1T
Revised test to eliminate cases whose result would be
changed by pending AI12-0287-1, and added cases to preserve complete testing.

B853001 [New] VCS Label: A3_1S
Checks that a renamed limited view cannot be used where there is not a with for the original package.

B854002 [New] VCS Label: A3_1T
[VCS Label A3_1S] When parameter or result has a null exclusion for a subprogram rename, check that the subtype of the corresponding parameter or result of the renamed subprogram must exclude null.
[VCS Label A3_1T] Corrected wording of the second test objective.

B854003 [New] VCS Label: A3_1T
Check various rules about renames as body: conformance of both the renaming and renamed entity, and renaming of abstract and intrinsic subprograms.

B854004 [New] VCS Label: A3_1T
Check that renaming a subprogram that requires overriding is illegal.

B91003A [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B91003B [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B91003C [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B91003D [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

C920001 [New] VCS Label: A3_1B
Check activation (and non-activation) of tasks declared as (part of) return objects in extended return statements.

B940003 [Modified] VCS Label: A3_1P
Added additional error locations to directly support common error reporting strategies.

B940004 [Modified] VCS Label: A3_1P
Split into 8 files; added location indicators.

B940008 [Pending New] New VCS Label: A3_1W
Required after 2020-01-01
Check the overriding indicators work properly on protected subprograms.

B940009 [Pending New] New VCS Label: A3_1W
Required after 2020-01-01
Check that the mode and type of the first parameter of protected operations that implement inherited subprograms is correct.

B940010 [Pending New] New VCS Label: A3_1W
Required after 2020-01-01
Check that a protected operation and a primitive subprogram if a tagged protected type cannot have the same profile.

B940011 [Pending New] New VCS Label: A3_1W
Required after 2020-01-01
Check that a protected operation and the associated inherited operation must subtype conform. Check that inherited operations must be implemented, overridden, or be a null procedure.

C940017 [Pending New] New VCS Label: A3_1W
Required after 2020-01-01
Check that dispatching calls can be made through a protected interface and that they can be implemented with protected subprograms and entries.

C94001A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94001B [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94001C [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94002A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94004A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94004B [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94004C [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94005A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94005B [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94007A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

C94010A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

B95007B [Modified] VCS Label: A3_1M
Fixed error markers @62,66,70 to match ACATS standards.

C95012A [Modified] VCS Label: A3_1W
Revised test to replace excessive delays with Impdef constants.

B952005 [Pending New] New VCS Label: A3_1W
  Required after 2020-01-01
  Check the overriding indicators work properly on task and
  protected entries.

C97307A [Modified] VCS Label: A3_1W
  Revised test to replace excessive delays with Impdef constants.

BA1001D [Modified] VCS Label: A3_1R
  Revised to mark the main subprogram as optional, as it is not
  necessary to check the test objective. Added error location
  indicators.

BA1020F [Modified] VCS Label: A3_1R
  Revised to mark the main subprogram as optional, as it is not
  necessary to check the test objective. Added error location
  indicators.

BA11007 [Modified] VCS Label: A3_1R
  Split into six test files to avoid requiring error recovery.

BA11013 [Modified] VCS Label: A3_1R
  Revised to make processing the last compilation unit optional,
  as it is not necessary to check the test objective. Added error
  location indicators.

BA1101E [Withdrawn] Org VCS Label: A3_1R
  Replaced by test BA14001.

BA12010 [Modified] VCS Label: A3_1R
  Revised to use Possible Error tags, and added error location
  indicators.

BA12017 [New] VCS Label: A3_1F
  [VCS Label A3_1D] Check that a limited with clause is not allowed
  on a descendant of the named package.
  [VCS Label A3_1F] Eliminated overlong line.

BA12018 [New] VCS Label: A3_1S
  Check that names mentioned but not named in a private with are
  also treated as privately withed.

BA14001 [New] VCS Label: A3_1R
  Replaces test BA1101E with modern format and additional test cases
  for Ada 95 compilation units.

BA14002 [New] VCS Label: A3_1R
  Similar to BA14001, testing private with.

BA14003 [New] VCS Label: A3_1R
  Checks that a unit is illegal if it depends on two versions of the
  same unit.

BA15003 [New] VCS Label: A3_1D
  Check that a library unit pragma must apply to a library unit.
CB30001 [New] VCS Label: A3_1C
Check that the optional message parameter is supported in a raise statement, and that the message can be retrieved with Exception_Message.

BB42001 [New] VCS Label: A3_1L
Check that the arguments of a pragma Assert have to have the correct types.

CB42001 [New] VCS Label: A3_1L
Check the usual operation of pragma Assert, including the optional message.

BB42002 [New] VCS Label: A3_1L
Check that the arguments of a pragma Assert cannot be reordered even if named notation is used.

CB42002 [New] VCS Label: A3_1L
Check that if the assertion policy is Ignore, none of the expressions of a pragma Assert are evaluated and no exception is raised by the pragma.

BB42003 [New] VCS Label: A3_1L
Check that a pragma Assertion_Policy is illegal if the policy is not recognized.

CB42003 [New] VCS Label: A3_1L
Check that procedure Ada.Assertions.Assert evaluates its arguments and raises Assertion_Error regardless of the assertion policy.

CC30003 [Modified] VCS Label: A3_1F
Eliminated overlong lines.

CC30004 [New] VCS Label: A3_1F
[VCS Label A3_1B] Check that if a generic actual type is elementary, it is passed by copy within an instance.
[VCS Label A3_1F] Eliminated overlong lines and tab characters.

CC3017C [Withdrawn] Org VCS Label: A3_1B
Replaced by test CC30004.

BC40003 [New] VCS Label: A3_1T
[VCS A3_1S] Checks that the subtype of the actual for a generic formal object with a null exclusion must exclude null.
[VCS A3_1T] Revised test to eliminate cases whose result would be changed by pending AI12-0287-1, and to preserve complete testing.

BC51016 [Modified] VCS Label: A3_1R
Corrected bad error tag, and added error location indicators.

BC51019 [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

BC51020 [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

BC51B02 [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

BC51C01 [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

BC51C02 [Allowed Modification] New VCS Label: A3_1Y Org VCS Label: A3_1
Required in ACATS 3.2
Added error location indicators.

BC54004 [New] VCS Label: A3_1S
Checks that the null exclusion of the formal type and of the actual subtype for a generic formal access type must match.

BC60001 [New] VCS Label: A3_1D
Test rules associated with formal subprogram defaults of null.

CC60001 [New] VCS Label: A3_1H
Check that calls on a formal abstract subprogram are dispatching.

BC60002 [New] VCS Label: A3_1H
Test rules associated with the controlling type of a formal abstract subprogram.

BC60004 [New] VCS Label: A3_1H
Check that a call on a formal abstract subprogram must be dispatching.

BC60005 [New] VCS Label: A3_1T
[VCS Label A3_1S] Checks that the subtype of a parameter or result of the actual subprogram for a generic formal subprogram where the corresponding parameter or result has a null exclusion must exclude null.
[VCS Label A3_1T] Corrected wording of the second test objective.

CD1009A [Withdrawn] VCS Label: A3_1Q
Has one of the objectives of, and thus redundant with, test CD2A31C (now CD30012).

CD1009B [Withdrawn] VCS Label: A3_1Q
Has one of the objectives of, and thus redundant with, test CD2A21C (now CD30011).

CD1009L [Withdrawn] VCS Label: A3_1Q
Virtually identical to, and thus redundant with, test CD2D13A.

CD1009M [Withdrawn] VCS Label: A3_1Q
Virtually identical to, and thus redundant with, test CD3014C.

CD2A21C [Withdrawn] VCS Label: A3_1Q
Replaced by CD30011.

CD2A31C [Withdrawn] VCS Label: A3_1Q
CD30008 [New] VCS Label: A3_1K
Renaming of previous test CD30001 (there is no subclause 13.3.3).
The test is unchanged other than renaming the units involved
and fixing an overlong line.

CD30009 [New] VCS Label: A3_1K
Renaming of previous test CD30002 (there is no subclause 13.3.3).
The test is unchanged other than renaming the units involved
and fixing an overlong line.

CD30011 [New] VCS Label: A3_1Q
Merges legacy tests CD1009B and CD2A21C to eliminate duplication
(CD1009B has the same objective as one of the objectives of
CD2A21C) while adding modern formatting and applicability
criteria.

CD30012 [New] VCS Label: A3_1Q
Merges legacy tests CD1009A and CD2A31C to eliminate duplication
(CD1009A has the same objective as one of the objectives of
CD2A31C) while adding modern formatting and applicability
criteria.

CD33001 [Withdrawn] VCS Label: A3_1K
Renamed to CD30008, as there is no subclause 13.3.3.

CD33002 [Withdrawn] VCS Label: A3_1K
Renamed to CD30009, as there is no subclause 13.3.3.

BD4003A [Modified] VCS Label: A3_1M
Moved error marker from @66 to @67 so it is on the incorrect
construct.

BDB0A01 [Modified] VCS Label: A3_1R
Revised to use Possible Error tags, and added error location
indicators.

BDD2003 [Modified] VCS Label: A3_1R
Added additional error locations to directly support common error
reporting strategies.

BDD2004 [New] VCS Label: A3_1F
[VCS Label A3_1A] Check that subprograms specified for stream
attributes meet the requirements for such subprograms. This
version corrects the location of 'Class @230, 235, 240.

CXA4037 [New] VCS Label: A3_1G
Check Ada.Strings.Fixed.Find_Token against rule from 2001
Corrigendum.

CXA5016 [New] VCS Label: A3_1F
[VCS Label A3_1D] Check operation of attribute Machine_Rounding.
[VCS Label A3_1F] Eliminated overlong lines.

CXAA021 [New] VCS Label: A3_1D
Check file operations of Text_IO.Bounded_IO.
Check file operations of Text_IO.Unbounded_IO.

Check file operations of Wide_Text_IO.Wide_Bounded_IO.

Check file operations of Wide_Wide_Text_IO.Wide_Wide_Bounded_IO.

Check file operations of Wide_Text_IO.Wide_Unbounded_IO.

Check file operations of Wide_Wide_Text_IO.Wide_Wide_Unbounded_IO.

Check that Stream_IO files opened with mode Out_File are not truncated by Close or Reset.
Corrected required file size in subtest 8, and various failure messages.
Eliminated overlong lines.
Corrected a failure message.

Check that Ada.Directories can be used to create and delete directories.

Test that package Environment_Variables can be used to read environment variables.
Added missing Applicability Criteria.

Test that package Environment_Variables can be used to modify environment variables.
Added missing Applicability Criteria.

Check the functionality defined in package Ada.Containers.Vectors.

Check the functionality defined in package Ada.Containers.Doubly_Linked_Lists.

Check the functionality defined in package Ada.Containers.Hashed_Maps.

Check the functionality defined in package Ada.Containers.Ordered_Maps.

Check the functionality defined in package Ada.Containers.Hashed_Sets.
Check the functionality defined in package Ada.Containers.Ordered_Sets.

Foundation containing types and values to be used to check the functionality of the indefinite containers.

Check the functionality defined in package Ada.Containers.Indefinite_Vectors.

Check the functionality defined in package Ada.Containers.Indefinite_Doubly_Linked_Lists.

Check the functionality defined in package Ada.Containers.Indefinite_Hashed_Maps.

Check the functionality defined in package Ada.Containers.Indefinite_Ordered_Maps.

Check the functionality defined in package Ada.Containers.Indefinite_Hashed_Sets.

Check the functionality defined in package Ada.Containers.Indefinite_Ordered_Sets.

Check that pragma Export can be used on subprograms and objects.

Check that a value of an unchecked union type (defined with pragma Unchecked_Union) can be passed to a C function. Added missing N/A => ERROR markers.

====== Annex C ======

Removed test case changed by AI05-0218-1.

====== Annex D ======

Added missing N/A => Error tag.

Added missing error tag.

Added missing error tag.
LXD7004 [Modified] VCS Label: A3_1P
   Added missing error tag.

LXD7005 [Modified] VCS Label: A3_1P
   Added missing error tag.

LXD7006 [Modified] VCS Label: A3_1P
   Added missing error tag.

LXD7007 [Modified] VCS Label: A3_1P
   Added missing error tag.

LXD7008 [Modified] VCS Label: A3_1P
   Added location indicators.

CXDB001 [Modified] VCS Label: A3_1P
   Added missing N/A => Error tag.

CXDB002 [Modified] VCS Label: A3_1P
   Added missing N/A => Error tag.

CXDB003 [Modified] VCS Label: A3_1P
   Added missing N/A => Error tag.

CXDB004 [Modified] VCS Label: A3_1P
   Added missing N/A => Error tag.

===== Annex E =====
[None.]

===== Annex F =====
[None.]

===== Annex G =====

CXG2001 [Modified] VCS Label: A3_1B
   Formula for Model Mantissa was replaced in Ada 2005 (and really in
   the Corrigendum), but was never updated in this test. This only
   affects machines with decimal floating point.

===== Annex H =====

LXH4001 [Modified] VCS Label: A3_1P
   Corrected format of error tag; added location indicators.

LXH4003 [Modified] VCS Label: A3_1P
   Corrected format of error tag; added location indicators.

LXH4006 [Modified] VCS Label: A3_1P
   Added missing error tags.
LXH4009 [Modified] VCS Label: A3_1P
Added missing error tags.

LXH4013 [Modified] VCS Label: A3_1P
Added missing error tags.

LXH4014 [Modified] VCS Label: A3_1P
Added missing error tags.

==== Support =====

IMPDEF [Modified] VCS Label: A3_1K
[VCS Label A3_1B] Added constants for the external names for the imported and exported routines in test CXB3017. The change is only needed for that test, it has no effect on any other test.
[VCS Label A3_1E] Added constants for the names of environment variables used by tests CXAH001 and CXAH002. 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 A3_1F] Added a constant and function used by test CXAG001. The change is only needed for that test, it has no effect on any other tests.
[VCS Label A3_1K] Added constants for the external name for the imported routine in test CXB3023. The change is only needed for that test, it has no effect on any other test.

TCTOUCH.ADA [Modified] VCS Label: A3_1S
Added code so that Touch only reports buffer overflow once.

---

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_3_1Y.Zip and Tar files on the ACAA web site. As always, when a multi-file test is modified, we provide all of the files, including the unmodified files.

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 3.1 is available in various formats on the ACAA web site.

Summary lists:
Withdrawn Tests

(Refer to the main list for details)

B38105A
B43102A
B74001A
B74001B
B74203E
BA1101E
C62003A
CC3017C
CD1009A
CD1009B
CD1009L
CD1009M
CD2A21C
CD2A31C
CD33001
CD33002

Allowed Modification Tests

(Refer to the main list for details)

B32201A ACATS 3.2
B33001A 2020-04-01
B33101A ACATS 3.2
B38103A ACATS 3.2
B38103B ACATS 3.2
B38103C ACATS 3.2
B38103D ACATS 3.2
B38103E ACATS 3.2
B48002A ACATS 3.2
B48002G ACATS 3.2
B48003B ACATS 3.2
B48003E ACATS 3.2
B56001A ACATS 3.2
B57001B ACATS 3.2
B660002 ACATS 3.2
B66001A ACATS 3.2
B66001B ACATS 3.2
BC51019 ACATS 3.2
BC51020 ACATS 3.2
BC51B02 ACATS 3.2
BC51C01 ACATS 3.2
BC51C02 ACATS 3.2

Modified Tests

(Refer to the main list for details)

B37309B
New Tests
(Refer to the main list for details)

B354002
B393011
B3A1007
B3A1008
B3A1009
B3A1A05
B3A2017
B431006
B452002
B480002
B480003
B480004
B540002
B640001
B651001
B730008
B730009
B730010
B730011
B730012
B730013
B750A04
B750A06
B831004
B840002
B853001
B854002
B854003
B854004
BA12017
BA12018
BA14001
BA14002
BA14003
BA15003
BB42001
BB42002
BB42003
BC40003
BC54004
BC60001
BC60002
BC60004
BC60005
BDD2004
C354004
C390012
C391003
C3A0030
C3A1003
C3A1004
C3A2004
C452A01
C452A02
Pending New Tests

(Refer to the main list for details)
<table>
<thead>
<tr>
<th>Code</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B631001</td>
<td>2020-04-01</td>
</tr>
<tr>
<td>B940008</td>
<td>2020-01-01</td>
</tr>
<tr>
<td>B940009</td>
<td>2020-01-01</td>
</tr>
<tr>
<td>B940010</td>
<td>2020-01-01</td>
</tr>
<tr>
<td>B940011</td>
<td>2020-01-01</td>
</tr>
<tr>
<td>B952005</td>
<td>2020-01-01</td>
</tr>
<tr>
<td>C940017</td>
<td>2020-01-01</td>
</tr>
</tbody>
</table>