What is this?

This is the ACATS modification list for ACATS 4.1. This document includes test corrections, test removals, and new tests for ACATS 4.1. ACATS 4.1 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 3.1CC 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 only as necessary for the foreseeable future.

ACATS Modification List version 2.6J 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 only as necessary for the foreseeable future.

Changes from the last list:

A handful of tests changed status on July 1st; see the lists for the details.

New test B433004 checks that an **others** choice is not allowed in an aggregate that appears as the dependent expression of a conditional expression in an unconstrained context.

New test C433007 checks that an aggregate with an **others** choice can be assigned into an array variable even if the nominal subtype of the variable is unconstrained.

Test B611004 was modified to replace most of the OK lines with ERROR lines. Class-wide aspects are only allowed on tagged types and primitive subprograms, and most of the subprogram bodies in this test are not primitive. This somewhat obscures the objective of the test, but that cannot be helped.

Note: Some Allowed Modification tests have a required date of “ACATS 4.2”. 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 4.2) is issued (most likely in the next quarter).

New tests posted between July 1, 2022 and September 30, 2022 will become required for conformity assessments on April 1, 2023. New tests posted between April 1, 2022 and June 30, 2022 will become required for conformity assessments on January 1, 2023. New tests posted between January 1, 2022 and March 31, 2022 will become required for conformity assessments on October 1, 2022. New tests posted before January 1, 2022 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.

**ACATS 4.1 includes optional Test Grading tools**

ACATS 4.1 introduces new test grading tools.
When the ACATS was designed (as the ACVC in the early 1980s), the intention always was that running it would give a simple and clear Pass or Fail result. However, grading of tests (particularly of B and L Tests) is somewhat subjective and very time-consuming. (Test grading typically involved poring over compiler listings of the entire ACATS with a large highlighter.) The subjectivity was acceptable in the days of common formal testing, as the formal testers would enforce a common grading standard (even if no one could describe what that was), but in an age of self-testing, it reduces the value of the ACATS.

As such, we're introducing a long-overdue set of automated grading tools. These tools are intended to:

- Greatly reduce the effort to grade an ACATS test run (and, hopefully to reduce the cost of formal testing to make it a more practical option);
- Ensure that all users of the ACATS grade tests in the same way, thus making it more valid to compare ACATS results between different implementations and different testers;
- Give implementers certainty as to whether a particular test result is passing (whether the location of an error is “close enough” to the canonical position);
- Be able to detect process errors (especially errors of omission) – that is, tests improperly run (for instance, files compiled in the wrong order where the ACATS requires an order), or not run at all (if one is checking off lines in a listing, it's hard to notice that some test or portion of a test is missing).

The tools also will optionally give feedback to implementers about quality of error message positioning and of error cascades (does one error generate multiple messages?). (Since “goodness of an implementation” is not something that the ACATS tests for, this is not an official goal for the tools.)

This is an ambitious set of goals. It's unlikely that they'll be completely met by the initial versions of the tools. As such, their use will be completely optional (but encouraged!) for ACATS 4.1. The tools will be updated periodically by the normal process for updating the ACATS – an ACATS Modification List like this one.

Details on the operation of the tools can be found in the ACATS 4.1 documentation (that documentation can be found at http://www.ada-auth.org/acats-files/4.1/docs/UG-6.HTM). As always, if you have questions, contact the ACAA Technical Agent at agent@ada-auth.org.

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:
These references are to the consolidated Ada standard as available at http://www.ada-auth.org/standards/rm12_w_tc1/html/RM-TOC.html.

---

**Main list:**

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

<table>
<thead>
<tr>
<th>Name</th>
<th>Category</th>
<th>Version label</th>
<th>(effective date; old version label)</th>
<th>Short description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B324004</td>
<td>[New] VCS Label: A4_1W</td>
<td></td>
<td></td>
<td>Check that the Predicate_Failure aspect is only allowed in conjunction with a predicate aspect, and that the expression is of type String.</td>
</tr>
<tr>
<td>C324006</td>
<td>[New] VCS Label: A4_1W</td>
<td></td>
<td></td>
<td>Check that for a Predicate_Failure aspect whose expression does not raise an expression, the value of the aspect is used as the Exception_Message in the exception occurrence when the appropriate predicate check fails.</td>
</tr>
<tr>
<td>B350001</td>
<td>[New] VCS Label: A4_1Q</td>
<td></td>
<td></td>
<td>Check that the requirements on the specification of the Default_Value aspect are enforced.</td>
</tr>
<tr>
<td>C350001</td>
<td>[New] VCS Label: A4_1D</td>
<td></td>
<td></td>
<td>Check that the object form of the Image attribute is implemented.</td>
</tr>
<tr>
<td>C350A01</td>
<td>[New] VCS Label: A4_1R</td>
<td></td>
<td></td>
<td>Check that objects of types with specified Default_Values are properly initialized. Part 1: Normal discrete types.</td>
</tr>
<tr>
<td>C350A02</td>
<td>[New] VCS Label: A4_1R</td>
<td></td>
<td></td>
<td>Check that objects of types with specified Default_Values are</td>
</tr>
</tbody>
</table>
properly initialized. Part 2: Generic discrete types.

C350B01 [New] VCS Label: A4_1R
Check that objects of types with specified Default_Values are properly initialized. Part 3: Normal real types.

C350B02 [New] VCS Label: A4_1R
Check that objects of types with specified Default_Values are properly initialized. Part 4: Generic real types.

C352001 [Modified] VCS Label: A4_1D
Corrected Failed message @267 to reflect intended result.

B354002 [New] VCS Label: A4_1D
Test basic requirements for the Mod attribute.

C354004 [New] VCS Label: A4_1D
Test basic operation of the Mod attribute.

B35901C [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators; moved error tags to correct locations.

B35A001 [New] VCS Label: A4_1D
Check Legality Rules are enforced when Small is specified by an aspect specification.

C35A001 [New] VCS Label: A4_1D
Check that Small can be specified by an aspect specification.

B360001 [New] VCS Label: A4_1Q
Check that the requirements on the specification of the Default_Component_Value aspect are enforced.

C360A01 [New] VCS Label: A4_1S
Check that objects of types with specified Default_Component_Values are properly initialized. Part 1: Arrays of discrete types.

C360A02 [New] VCS Label: A4_1S
Check that objects of types with specified Default_Component_Values are properly initialized. Part 2: Generic arrays of discrete types.

C360B01 [New] VCS Label: A4_1S
Check that objects of types with specified Default_Component_Values are properly initialized. Part 3: Arrays of real types.

C360B02 [New] VCS Label: A4_1S
Check that objects of types with specified Default_Component_Values are properly initialized. Part 4: Generic arrays of real types.

Required in ACATS 4.2
Added error location indicators; split constructs so there is only one error per construct.

B370002 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B37309B [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.

B37310B [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.

B37311A [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.

B38003D [Modified] VCS Label: A4_1A
Corrected error tags @70,76,82,88 so “optional” and “error” are on the same line, as required by the test standards.

Required in ACATS 4.2
Added error location indicators.

B38103B [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B38103C [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B38103D [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B38103E [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B392002 [Modified] VCS Label: A4_1G
Revised to use Possible Error tags, and added error location indicators.

B392003 [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.

B392004 [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.

B392005 [Modified] VCS Label: A4_1G
Revised to use Possible Error tags, and added error location
indicators.

B393001 [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.

B393005 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B393007 [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies. Fixed overlong lines.

B393008 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B393009 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B393010 [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies. Fixed overlong lines.

B393012 [New] VCS Label: A4_1P
Check that abstract types cannot be used in additional contexts that do not allow abstract types.

B393013 [New] VCS Label: A4_1P
Check that interface types cannot be used in additional contexts that do not allow abstract types.

B394001 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B394A01 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B394A02 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B394A03 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B394A04 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B394A05 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.
Check that an access discriminant is only null excluding if it includes a null exclusion.

Split Pack6 so possible error sets don't intersect; fixed overlong lines.

Split Pack6 so possible error sets don't intersect; fixed overlong lines.

Split Pack6 so possible error sets don't intersect; fixed 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.

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.

Check that an explicitly aliased parameter has the same accessibility level as a normal parameter unless compared to the return object. Check that master of the function call is properly determined statically.

Corrected so that it is not affected by whether AI12-0345-1, AI12-0402-1, and AI12-0390-1 are implemented.

PPA and PFA trigger the issue of AI12-0289-1; explicit null exclusions were added to the parameters to avoid it.

Required in ACATS 4.2

Added error location indicators.

 Added error location indicators.
Return @119 fails an accessibility check unless C is aliased.

The subtype of My_Colln @125 needs to be either discriminated or to statically match that of the formal types of the parameters.

Commented out Priv2 test case as it is the subject of AI12-0428-1.

Check that record aggregates that specify a discriminant with <> get the default value of the discriminant.

Check that extension aggregates that specify a discriminant with <> get the default value of the discriminant.

Check that record aggregates are not allowed for types whose components aren't visible.

Check that record aggregates that specify a discriminant with <> are illegal if the discriminant does not have a default.

Check that extension aggregates that specify a discriminant with <> are illegal if the discriminant does not have a default.

Check that all of the extensions of an extension aggregate are record extensions.

Check that the ancestor subtype_mark of an extension aggregate does not denote a view with unknown discriminants.

Added error location indicators.

Added error location indicators.

Added error location indicators.

Added error location indicators.
B433003 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
   Required in ACATS 4.2
   Added error location indicators.

B433004 [Pending New] New VCS Label: A4_1BB
   Required after 2023-04-01
   Check that an others choice is not allowed in an aggregate that
   appears as the dependent expression of a conditional expression
   in an unconstrained context.

C433007 [Pending New] New VCS Label: A4_1BB
   Required after 2023-04-01
   Check that an aggregate with an others choice can be assigned
   into an array variable even if the nominal subtype of the variable
   is unconstrained.

B440001 [New] VCS Label: A4_1Y
   Check that the operands of a membership test are
   simple_expressions, not choice_expressions.

B452001 [New] VCS Label: A4_1K
   Check that a membership that contains choice expressions is
   illegal for a limited type that does not have equality.

B452002 [New] VCS Label: A4_1L
   Check that Resolution and Legality Rules for the universal access
   equality operator are enforced.

C452002 [New] VCS Label: A4_1F
   [VCS Label A4_1D] Check that overloading resolution of Ada 2012
   membership operations works as intended.
   [VCS Label A4_1F] Fixed incorrect test number passed to Report.

C452003 [New] VCS Label: A4_1R
   [VCS Label A4_1K] Check that memberships of general access-to-
   object types work as defined.

C452004 [New] VCS Label: A4_1Q
   [VCS Label A4_1L] Check that memberships with multiple choices are
   evaluated as required, and that the correct equality operation is
   called for untagged record types.
   [VCS Label A4_1Q] LTemp and RTemp @ 136 and 164 should be declared
   Integer rather than Natural, as the numerators are of type Integer
   and can be negative.

C452005 [New] VCS Label: A4_1L
   Check that memberships with multiple choices call the correct
   equality operation for array types.

C452006 [New] VCS Label: A4_1M
   [VCS Label A4_1L] Check that memberships with multiple choices
   call the correct equality operation for access types.
   [VCS Label A4_1M] Corrected to remove cases made illegal by
   AI12-0328-1.
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.

Fixed overlong lines. Added location indicators.

There should not be an ERROR tag @282; Single_Highest @290 should be Single_High. Added location indicators.

Added error location indicators.

There should not be an ERROR tag @282; Single_Highest @290 should be Single_High. Added location indicators.

Added error location indicators.

Check that a quantified expression has to be surrounded in parentheses if it is not already surrounded by them.

Check the operation of quantified expressions using array element iterators.

Check boundary conditions for quantified expressions using discrete iterators. (This test is a companion to the existing C458001.)

Check the operation of quantified expressions using generalized iterators.
Check the operation of quantified expressions using container iterators.

Added error location indicators.

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 qualified expressions are a constant view of the expression, even if the expression is a variable object.

Check that qualified expressions are a name, and thus can be used as a prefix and other places a name is required.

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

Added error location indicators.
Moved error tags to more usual locations, added error location indicators.

Check that static expressions that fail a static predicate check are illegal, even if the assertion policy is Ignore.

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

Corrected comment @49 not to have the form of an error tag.
Added additional error locations to directly support common error reporting strategies.

Added error location indicators; moved incorrectly located error tags.

Revised test to remove dependence on Ada.Iterator_Interfaces.

Added error location indicators.

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.

Added error location indicators.

Added error location indicators.

Check that Pre and Post are only allowed on subprograms and entries (and not on subprogram instances).

Check that Pre and Post are not allowed on subprograms acting as completions.

Corrected several error location codes.

Check that Pre'Class and Post'Class are only allowed on primitive subprograms of tagged types. (Normal tagged types and interfaces.)

Check that Pre'Class and Post'Class are not allowed on subprograms acting as completions.

All of the cases previously marked OK other than the first are illegal as the subprogram is not a primitive subprogram, and that is required by 13.1.1(16/4).
Check that Pre and Post are not allowed on abstract subprograms and null procedures.

B611006 [New] VCS Label: A4_1V
  [VCS Label A4_1A] Check that the restrictions on Pre'Class and Post'Class expressions added by AI12-0113-1 are enforced.
  [VCS Label A4_1C] Clarified error @208; changed error @233 & @247 to OK (as the inherited Proc4 routine is abstract and thus is not subject to the new rule).
  [VCS Label A4_1V] Eliminated the third objective (all of the code under the “Third Objective” comment), as the rule was replaced by AI12-0412-1 (a Binding Interpretation).

B611007 [New] VCS Label: A4_1F
  [VCS Label A4_1A] Check that Pre'Class and Post'Class are not allowed on entries and protected subprograms of untagged task and protected types.
  [VCS Label A4_1F] Corrected preconditions @77 and @110 not to violate the new rule added by Binding Interpretation AI12-0166-1.

B611008 [New] VCS Label: A4_1A
  Check various rules about the use of the Result attribute.

B611009 [New] VCS Label: A4_1A
  Check that F'Result is not allowed in the body of F.

B611010 [New] VCS Label: A4_1A
  Check that the Old attribute is not allowed in a subprogram or entry body, or in an accept statement.

B611011 [New] VCS Label: A4_1A
  Check the Old attribute cannot be used within a precondition expression nor in a generic specification outside of a postcondition expression.

B611012 [New] VCS Label: A4_1A
  Check that the prefix of an Old attribute cannot contain the Result attribute, another Old attribute, nor the index of a quantified expression.

B611013 [New] VCS Label: A4_1A
  For a discrete X, check that X'Old has the nominal subtype of X.

B611014 [New] VCS Label: A4_1C
  [VCS Label A4_1B] If an Old attribute reference occurs at a place where it is potentially unevaluated, check that the prefix must statically denote an object.
  [VCS Label A4_1C] Corrected error location codes @240 and @252.

B611015 [New] VCS Label: A4_1B
  Check that the prefix of an Old attribute cannot have a limited type.

B611016 [New] VCS Label: A4_1B
  Check that tagged task and tagged protected types do not allow Pre'Class and Post'Class, nor are they allowed on operations of those types.
[VCS Label A4_1C] Check that Pre'Class cannot be specified for an overriding of a subprogram that does not specify Pre'Class.

[VCS Label A4_1C] Declaration of New_Proc @358 is not overriding and thus not subject to the rule. The declaration of Shine @149 never overrides primitives of the actual type by 12.3(18), thus this rule does not apply to instances of Gen2. Added an example using a generic derived type.

Check 6.1.1(10-16/3).

Type T08 @230 and @241 needs to be declared abstract.

Check 6.1.1(17/3).

The declaration of Get_Count @112 never overrides primitives of the actual type by 12.3(18), thus this rule does not apply to instances of Gen2. Added examples using generic derived types.

Check that preconditions and postconditions where both specific and class-wide contracts apply are checked properly.

Check that the correct expressions are evaluated for specific preconditions and postconditions, and that they are evaluated at the correct point.

The main subprogram name was corrected to follow the rules of the ACATS User's Guide, clause 4.3.3.

Check that the correct expressions are evaluated for class-wide preconditions and postconditions, and that they are evaluated at the correct point.

The main subprogram name was corrected to follow the rules of the ACATS User's Guide, clause 4.3.3.

Check that the correct expressions are evaluated for class-wide preconditions and postconditions of an interface type, and that they are evaluated at the correct point.

Fixed overlong lines.

Check that X'Old has the tag of X and not the tag of the nominal subtype of X (for Post).

The main subprogram name was corrected to follow the rules of the ACATS User's Guide, clause 4.3.3.

Check that X'Old has the tag of X and not the tag of the nominal subtype of X (for Post'Class).

The main subprogram name was corrected to follow the rules of the ACATS User's Guide, clause 4.3.3.
C611B03 [New] VCS Label: A4_1D
[VCS Label A4_1C] Check that X'Old has the tag of X and not the
tag of the nominal subtype of X (for Post'Class of an interface
type).
[VCS Label A4_1D] Fixed overlong lines.

C620002 [New] VCS Label: A4_1R
Modernizes, adds test cases to, and eliminates illegal calls from
legacy test C95072A.

B62001C [Modified] VCS Label: A4_1A
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: A4_1A
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).

B631001 [New] VCS Label: A4_1N
Check that a prefixed view is intrinsic.

C641002 [New] VCS Label: A4_1V
Check that for an out parameter of a scalar type with a specified
Default_Value, the value of the actual parameter is passed in
without any constraint check.

B641003 [New] VCS Label: A4_1V
Check that for an out parameter of a scalar type with a specified
Default_Value, it is illegal to pass a type conversion of an
object unless the type of the object is related and has a
specified Default_Value.

B641003 [New] VCS Label: A4_1Y
[VCS Label A4_1V] Check that for an out parameter of a scalar type
with a specified Default_Value, the value of the actual parameter
is passed in without any predicate check.
[VCS Label A4_1Y] Corrected comments to talk about predicates and
Assertion_Error.

B650003 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B650005 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B650007 [New] VCS Label: A4_1Y
Check that an extended return statement containing the reserved
word aliased is only allowed if the type of the return object is
immutably limited.

B65005A [Modified] VCS Label: A4_1D
Added additional error locations to directly support common error
reporting strategies.

B65005B [Modified] VCS Label: A4_1D
Added additional error locations to directly support common error reporting strategies.

C650B04 [New] VCS Label: A4_1A
Check that the result of a function that returns a class-wide tagged type does an accessibility check on the tag of the result object when the result expression is class-wide.

B651001 [New] VCS Label: A4_1S
[VCS Label A4_1D] 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).
[VCS Label A4_1S] Corrected pragmas @176 and @179 to refer to the immediately preceding procedures.

C651001 [New] VCS Label: A4_1D
Check that a nonreturning procedure can raise an exception and raises Program_Error otherwise (pragma No_Return version).

B651002 [New] VCS Label: A4_1D
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 (aspect No_Return version).

C651002 [New] VCS Label: A4_1D
Check that a nonreturning procedure can raise an exception and raises Program_Error otherwise (aspect No_Return version).

B651003 [New] VCS Label: A4_1D
Check that an aspect No_Return can only be given on a procedure or generic procedure, that the expression has to be nonstatic, and that the aspect is not given on an instance or null procedure.

B651004 [New] VCS Label: A4_1V
Check that a dispatching nonreturning procedure can only be overridden by a nonreturning procedure.

B660001 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B660002 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B660003 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Fixed overlong lines, added error location indicators.

Required in ACATS 4.2
Added error location indicators.
B66001B [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B67001C [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Moved many error tags to the correct position; added error location indicators.

B680001 [Modified] VCS Label: A4_1F
The immediately enclosing expression function @140 does not declare a parameter A, but A is used @141. Added location indicators.

B730006 [Modified] VCS Label: A4_1C
Corrected format of error tags; added location indicators.

B730007 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Fixed overlong lines, header formatting issues, added error location indicators.

B730008 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B730009 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B730011 [New] VCS Label: A4_1C
Test various rules about completions of private types and extensions; replaces test B74001A.

B730012 [New] VCS Label: A4_1C
Test various rules about completions of private types and extensions in generic units; replaces test B74001B.

B730013 [New] VCS Label: A4_1D
Test operations available for a private type; replaces test B74203E.

B73001A [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B73001B [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B73001C [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B73001D [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.
B73001E [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B73001F [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B73001G [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B73001H [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B740002 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B740003 [Modified] VCS Label: A4_1H
[VCS Label A4_1A] Modified criteria @75 so that it doesn't look like an error tag to the grading tools; also fixed an overlong line.
[VCS Label A4_1H] Modified grading to document that processing the main is optional. Added error location indicators.

B74001A [Withdrawn] VCS Label: A4_1C
Replaced by B730011.

B74001B [Withdrawn] VCS Label: A4_1C
Replaced by B730012.

Required in ACATS 4.2
Added error location indicators.

Required in ACATS 4.2
Added error location indicators; added Optional Error tag for D component (that matches missing discriminant).

B74203E [Withdrawn] VCS Label: A4_1D
Replaced by B730013.

B750A07 [New] VCS Label: A4_1G
[VCS Label A4_1A] Check that only legal expressions are allowed in the return expression of an expression function that returns a limited type.
[VCS Label A1_1G] Adjusted error indicators to allow the error to be reported anywhere in a multi-line expression.

C760A02 [New] VCS Label: A4_1G
Checks that functions that return immutably limited types are built-in-place.
C760A03 [New] VCS Label: A4_1G
Checks that expression functions that return immutably limited types are built-in-place.

B830001 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Split test into 6 files; added error location indicators.

B830003 [New] VCS Label: A4_1W
Check that illegal homographs are detected in specifications of generic packages. Replaces B83004A.

B830004 [New] VCS Label: A4_1W
Check that illegal homographs are detected in body of generic packages. Replaces B83004B.

B830005 [New] VCS Label: A4_1W
Check that illegal homographs are detected in subunits of the bodies of generic packages. Replaces B83004D.

B830006 [New] VCS Label: A4_1W
Check that illegal homographs of expression functions are detected in generic packages.

Required in ACATS 4.2
Moved many error tags to the correct position; added error location indicators.

Required in ACATS 4.2
Moved many error tags to the correct position; added error location indicators.

B83003B [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Moved many error tags to the correct position; added error location indicators.

Replaced in part by test B830003, the rest was redundant with B83004B.

B83004B [Withdrawn] Org VCS Label: A4_1W
Replaced by test B830004.

B83004C [Withdrawn] Org VCS Label: A4_1W
Redundant with B83004D.

B83004D [Withdrawn] Org VCS Label: A4_1W
Replaced by test B830005.

B83027B [Modified] VCS Label: A4_1G
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: A4_1G  
Corrected bad error tag, and added error location indicators.

B831006 [New] VCS Label: A4_1K  
Checks that overriding indicators cannot be given on any subprogram declaration in a protected body.

B83E01B [Modified] VCS Label: A4_1G  
Removed possible error cascades, reformatted some error tags, and added error location indicators.

B83F02C [Modified] VCS Label: A4_1A  
Updated many optional error markers to match ACATS standards.

B85002A [Withdrawn] VCS Label: A4_1H  
Replaced by B851005.

B85003A [Withdrawn] VCS Label: A4_1H  
Replaced by B851007.

B85003B [Withdrawn] VCS Label: A4_1H  
Replaced by B851007.

B851004 [Modified] VCS Label: A4_1J  
Revised test to eliminate cases whose result would be changed by pending AI12-0287-1, and added cases to preserve complete testing.

B851005 [New] VCS Label: A4_1H  
Checks that “known to be constrained” is properly enforced for renames. Part 1: “Normal” cases (no generics or access cases).

B851006 [New] VCS Label: A4_1H  
Checks that “known to be constrained” is properly enforced for renames. Part 2: Access cases.

B851007 [New] VCS Label: A4_1K  
[VCS Label A4_1H] Checks that “known to be constrained” is properly enforced for renames. Part 3: Generic cases.  
[VCS Label A4_1K] Corrected types of declaration @198 and @214, and mode @328.

B853001 [New] VCS Label: A4_1H  
Checks that a renamed limited view cannot be used where there is not a with for the original package.

B854002 [New] VCS Label: A4_1J  
[VCS Label A4_1H] 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 A4_1J] Corrected wording of the second test objective.  
[VCS Label A4_1K] Corrected error markers on RenG51, RenG61, RenGF51, and RenGF61. Corrected renaming @412 to rename the correct function.

B854003 [New] VCS Label: A4_1J
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: A4_1J
Check that renaming a subprogram that requires overriding is illegal.

B854005 [New] VCS Label: A4_1J
Check that a renaming of a prefixed view is illegal if renaming the prefix as an object is illegal.

B860002 [New] VCS Label: A4_1Q
Check that implicit conversions of access types don't allow too much.

B91003A [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B91003B [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B91003C [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B91003D [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B940003 [Modified] VCS Label: A4_1C
Added additional error locations to directly support common error reporting strategies.

B940004 [Modified] VCS Label: A4_1C
Split into 8 files; added location indicators.

B940008 [New] VCS Label: A4_1M
Check the overriding indicators work properly on protected subprograms.

B940009 [New] VCS Label: A4_1M
Check that the mode and type of the first parameter of protected operations that implement inherited subprograms is correct.

B940010 [New] VCS Label: A4_1M
Check that a protected operation and a primitive subprogram of a tagged protected type cannot have the same profile.

B940011 [New] VCS Label: A4_1M
Check that an protected operation and the associated inherited operation must subtype conform. Check that inherited operations must be implemented, overridden, or be a null procedure.

C940017 [New] VCS Label: A4_1M
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: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94001B [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94001C [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94002A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94004A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94004B [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94004C [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94005A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94005B [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94007A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

C94010A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

B950001 [Modified] VCS Label: A4_1W
Commented out test case 149; fixed overlong lines, added error location indicators.

B950002 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

B95007B [Modified] VCS Label: A4_1A
Fixed error markers @62,66,70 to match ACATS standards.

C95012A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

Replaced by C620002; it violated the anti-order dependence rules of 6.4.1.

Required in ACATS 4.2
Added error location indicators.

B952005 [New] VCS Label: A4_1M
Check the overriding indicators work properly on task and protected entries.

C97307A [Modified] VCS Label: A4_1M
Revised test to replace excessive delays with Impdef constants.

BA1001D [Modified] VCS Label: A4_1G
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: A4_1G
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: A4_1G
Split into six test files to avoid requiring error recovery.

BA11013 [Modified] VCS Label: A4_1G
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: A4_1G
Replaced by test BA14001.

BA12010 [Modified] VCS Label: A4_1G
Revised to use Possible Error tags, and added error location indicators.

BA12018 [New] VCS Label: A4_1H
Check that names mentioned but not named in a private with are also treated as privately withed.

BA14001 [New] VCS Label: A4_1G
Replaces test BA1101E with modern format and additional test cases for Ada 95 compilation units.

BA14002 [New] VCS Label: A4_1G
Similar to BA14001, testing private with.

BA14003 [New] VCS Label: A4_1G
Checks that a unit is illegal if it depends on two versions of the same unit.

BA21005 [New] VCS Label: A4_1D
Test that Pure and Preelaborate aspects cannot be specified late, with a nonstatic expression, or an expression of a type other than Boolean.

BA21A01 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.
Check that the arguments of a pragma Assert have to have the correct types.
Corrected two incorrect error location codes.

Check the usual operation of pragma Assert, including the optional message.

Check that the arguments of a pragma Assert cannot be reordered even if named notation is used.

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.

Check that a pragma Assertion_Policy is illegal if the policy is not recognized.

Check that procedure Ada.Assertions.Assert evaluates its arguments and raises Assertion_Error regardless of the assertion policy.

Moved “=” for type Rec in order that it not violate 4.5.2(9.8/4).

Check that generic formal parts cannot be prepended to task or protected specifications. Replaces BC2001D and BC2001E.

Added error location indicators.

Added error location indicators.

Added error location indicators.

Required in ACATS 4.2

Required in ACATS 4.2

Required in ACATS 4.2

Added error location indicators.

Replaced by test BC10001.

Replaced by test BC10001.

Required in ACATS 4.2

Added error location indicators.
Required in ACATS 4.2
Added error location indicators.

Required in ACATS 4.2
Added error location indicators.

Required in ACATS 4.2
Added error location indicators.

BC40003 [New] VCS Label: A4_1J
[VCS A4_1H] Checks that the subtype of the actual for a generic formal object with a null exclusion must exclude null.
[VCS A4_1J] Revised test to eliminate cases whose result would be changed by pending AI12-0287-1, and to preserve complete testing.

CC51010 [Modified] VCS Label: A4_1F
The main subprogram name was corrected to follow the rules of the ACATS User's Guide, clause 4.3.3.

BC51016 [Modified] VCS Label: A4_1G
Corrected bad error tag, and added error location indicators.

BC51019 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

BC51020 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

BC51B02 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

BC51C01 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

BC51C02 [Allowed Modification] New VCS Label: A4_1P Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

BC54004 [New] VCS Label: A4_1H
Checks that the null exclusion of the formal type and of the actual subtype for a generic formal access type must match.

BC55001 [New] VCS Label: A4_1P
Check that formal interface types cannot be used in contexts that do not allow abstract types.

BC60001 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators.

BC60002 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
BC60005 [New] VCS Label: A4_1J
[BC60005] 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.
[BC60005] Corrected wording of the second test objective.
[BC60005] Corrected instance @437 to use correct actual parameter.

BD10001 [New] VCS Label: A4_1R
Check that an aspect cannot be specified in multiple ways for a single entity.

CD1009A [Withdrawn] VCS Label: A4_1D
Has one of the objectives of, and thus redundant with, test CD2A31C (now CD30012).

CD1009B [Withdrawn] VCS Label: A4_1D
Has one of the objectives of, and thus redundant with, test CD2A21C (now CD30011).

CD1009L [Withdrawn] VCS Label: A4_1D
Virtually identical to, and thus redundant with, test CD2D13A.

CD1009M [Withdrawn] VCS Label: A4_1D
Virtually identical to, and thus redundant with, test CD3014C.

CD11001 [New] VCS Label: A4_1A
Check that appropriate entities are visible in an aspect specification.

CD2A21C [Withdrawn] VCS Label: A4_1D
Replaced by CD30011.

CD2A31C [Withdrawn] VCS Label: A4_1D
Replaced by CD30012.

BD11001 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators; improved error tags for freezing subtests.

BD30001 [New] VCS Label: A4_1D
Check Legality Rules are enforced when Size is specified by an aspect specification for a type.

BD30002 [New] VCS Label: A4_1D
Check Legality Rules are enforced when Alignment is specified by an aspect specification for a type.

BD30003 [New] VCS Label: A4_1D
Check Legality Rules are enforced when Component_Size is specified...
by an aspect specification.

CD30010 [New] VCS Label: A4_1D
Check that Component_Size can be specified via an aspect specification.

CD30011 [New] VCS Label: A4_1D
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: A4_1D
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.

CD30013 [New] VCS Label: A4_1Y
Check that Program_Error is raised if two distinct declarations of tagged types have the same external tag. Part 1: Attribute definition clauses.

CD30014 [New] New VCS Label: A4_1Y
Check that Program_Error is raised if two distinct declarations of tagged types have the same external tag. Part 2: Aspect specifications.

BD4003A [Modified] VCS Label: A4_1A
Moved error marker from @66 to @67 so it is on the incorrect construct.

BDB0A01 [Modified] VCS Label: A4_1G
Revised to use Possible Error tags, and added error location indicators.

BDB2001 [New] VCS Label: A4_1G
Check that a call on an instance of Unchecked_Deallocation is illegal if the access type has a Storage_Size that is statically zero.

BDB4001 [New] VCS Label: A4_1P
Check that the object for a pool that supports subpools is declared in the same scope as the access type that uses it.

BDC1001 [New] VCS Label: A4_1X
Check that restriction No_Use_of_Attribute is implemented.

BDC1002 [New] VCS Label: A4_1X
Check that restriction No_Specification_of_Aspect is implemented.

BDC1003 [New] VCS Label: A4_1X
Check that restriction No_Use_of_Pragma is implemented.

BDD2003 [Modified] VCS Label: A4_1G
Added additional error locations to directly support common error reporting strategies.
BDD2006 [New] VCS Label: A4_1Q
Check that the requirements for specifying the Stream_Size attribute are enforced.

BDD2007 [New] VCS Label: A4_1Q
Check that the requirements for specifying the Stream_Size aspect are enforced.

CDD2B01 [New] VCS Label: A4_1Q
Check that confirming Stream_Size attributes work as expected.

CDD2B02 [New] VCS Label: A4_1Q
Check that confirming Stream_Size aspects work as expected.

CDD2B03 [New] VCS Label: A4_1Q
Check that non-confirming Stream_Size attributes work as expected.

CDD2B04 [New] VCS Label: A4_1Q
Check that non-confirming Stream_Size aspects work as expected.

CDE0002 [Modified] VCS Label: A4_1E
Changed to use Storage_Size rather than Size so that the static expression rule of 13.14(7.2/3) is not triggered.

BDE0007 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators; stripped SAIC development junk.

CXA4034 [Withdrawn] VCS Label: A4_1AA
Replaced by CXA4038.

CXA4038 [Pending New] New VCS Label: A4_1AA
Required after 2023-01-01
Check that the operations Slice, Bounded_Slice, and To_(Wide_)String have the correct results, bounds, and exceptions for Bounded_Strings and Bounded_Wide_Strings.

CXA4039 [Pending New] New VCS Label: A4_1AA
Required after 2023-01-01
Check that the operations Slice, Unbounded_Slice, and To_(Wide_)String have the correct results, bounds, and exceptions for Unbounded_Strings and Unbounded_Wide_Strings.

CXAC009 [New] VCS Label: A4_1D
Check that Stream_IO.File_Type has Preelaborable_Initialization.

CXAG003 [New] VCS Label: A4_1H
[VCS Label A4_1G] Check that function Ada.Directories.Name_Case_Equivalence exists and that it reflects the handling of file names.
[VCS Label A4_1H] Added “return;” @171; if the file can't be opened, the Read_File trial is complete.

CXAI001 [Modified] VCS Label: A4_1W
Changed the names of operations to be compatible with the ruling of AI12-0400-1. Also fixed some overlong lines.
BXAI005 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators rather than attempting to move the
tags - this test was designed expecting only that something
in each marked construct would be rejected (unlike most tests).

BXAI006 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators rather than attempting to move the
tags - this test was designed expecting only that something
in each marked construct would be rejected (unlike most tests).

BXAI007 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators rather than attempting to move the
tags - this test was designed expecting only that something
in each marked construct would be rejected (unlike most tests).

BXAI008 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators rather than attempting to move the
tags - this test was designed expecting only that something
in each marked construct would be rejected (unlike most tests).

CXAI010 [Modified] VCS Label: A4_1W
Changed the names of operations to be compatible with the ruling
of AI12-0400-1. Also fixed some overlong lines.

CXAI01 [Modified] VCS Label: A4_1W
Changed the names of operations to be compatible with the ruling
of AI12-0400-1. Also fixed some overlong lines.

BXAIA03 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators rather than attempting to move the
tags - this test was designed expecting only that something
in each marked construct would be rejected (unlike most tests).

BXAIA04 [Allowed Modification] New VCS Label: A4_1W Org VCS Label: A4_1
Required in ACATS 4.2
Added error location indicators rather than attempting to move the
tags - this test was designed expecting only that something
in each marked construct would be rejected (unlike most tests).

CXAI01 [New] VCS Label: A4_1U
[VCS Label A4_1T] Check that Ada.Containers.Vectors is
preelaborated, and it and types declared in it can be used in
preelaborated packages.
[VCS Label A4_1U] Corrected instance prefix of No_Element in
some Insert calls. Qualified aggregates to avoid issues for Ada
202x.

CXAI02 [New] VCS Label: A4_1U
[VCS Label A4_1T] Check that Ada.Containers.Bounded_Vectors is
preelaborated, and it and types declared in it can be used in
preelaborated packages.
[VCS Label A4_1U] Corrected instance prefix of No_Element in some Insert calls. Qualified aggregates to avoid issues for Ada 202x.

CXAIB03 [New] VCS Label: A4_1U
[VCS Label A4_1T] Check that Ada.Containers.Doubly_Linked_Lists is preelaborated, and it and types declared in it can be used in preelaborated packages.  
[VCS Label A4_1U] Corrected instance prefix of No_Element in some Insert calls.

CXAIB04 [New] VCS Label: A4_1U
[VCS Label A4_1T] Check that Ada.Containers.Bounded_Doubly_Linked_Lists is preelaborated, and it and types declared in it can be used in preelaborated packages.  
[VCS Label A4_1U] Corrected instance prefix of No_Element in some Insert calls.

CXAIB05 [New] VCS Label: A4_1T
Check that Ada.Containers.Ordered_Maps is preelaborated, and it and types declared in it can be used in preelaborated packages.

CXAIB06 [New] VCS Label: A4_1T
Check that Ada.Containers.Bounded_Ordered_Maps is preelaborated, and it and types declared in it can be used in preelaborated packages.

CXAIB07 [New] VCS Label: A4_1T
Check that Ada.Containers.Hashed_Maps is preelaborated, and it and types declared in it can be used in preelaborated packages.

CXAIB08 [New] VCS Label: A4_1T
Check that Ada.Containers.Bounded_Hashed_Maps is preelaborated, and it and types declared in it can be used in preelaborated packages.

CXB3013 [Modified] VCS Label: A4_1W
Added "#include <stdlib.h>" @f1:57, as this is required by standard C for the call to Malloc.

CXB3019 [Modified] VCS Label: A4_1S
Added missing failures if Program_Error is not raised when expected.

CXB3020 [Modified] VCS Label: A4_1S
Added missing failures if Program_Error is not raised when expected.

CXB3021 [Modified] VCS Label: A4_1S
Added missing failures if Program_Error is not raised when expected.

CXB3022 [Modified] VCS Label: A4_1S
Added missing failures if Program_Error is not raised when expected.
CXB3023 [New] VCS Label: A4_1A
Check that a value of an unchecked union type (defined with
pragma Unchecked_Union) can be passed to a C function.

CXB3024 [New] VCS Label: A4_1A
Check that a value of an unchecked union type (defined with
aspect Unchecked_Union) can be passed to a C function.

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

BXC6A03 [Modified] VCS Label: A4_1G
Changed subtest result to require the changes of AI05-0218-1.

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

CXD2007 [Modified] VCS Label: A4_1C
Added missing N/A => Error tag.

LXD7001 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7003 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7004 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7005 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7006 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7007 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7008 [Modified] VCS Label: A4_1C
Added missing error tag.

LXD7009 [Modified] VCS Label: A4_1C
Added location indicators.

CXDB001 [Modified] VCS Label: A4_1C
Added missing N/A => Error tag.

CXDB002 [Modified] VCS Label: A4_1C
Added missing N/A => Error tag.

CXDB003 [Modified] VCS Label: A4_1C
Added missing N/A => Error tag.

CXDB004 [Modified] VCS Label: A4_1C
Added missing N/A => Error tag.

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

===== Annex F =====

[None.]

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

[None.]

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

LXH4001 [Modified] VCS Label: A4_1C
   Corrected format of error tag; added location indicators.

LXH4003 [Modified] VCS Label: A4_1C
   Corrected format of error tag; added location indicators.

LXH4006 [Modified] VCS Label: A4_1C
   Added missing error tags.

LXH4009 [Modified] VCS Label: A4_1C
   Added missing error tags.

LXH4013 [Modified] VCS Label: A4_1C
   Added missing error tags.

LXH4014 [Modified] VCS Label: A4_1C
   Added missing error tags.

===== Support =====

F350A00 [New] VCS Label: A4_1R
   A foundation for discrete specified Default_Value tests.

F350B00 [New] VCS Label: A4_1R
   A foundation for real specified Default_Value tests.

F360A00 [New] VCS Label: A4_1S
   A foundation for discrete specified Default_Component_Value tests.

F360B00 [New] VCS Label: A4_1S
   A foundation for real specified Default_Component_Value tests.

F458A00 [Pending New] New VCS Label: A4_1AA
   Required after 2023-01-01
   A foundation containing a sparse array container, used in quantified expression tests.

F611A00 [New] VCS Label: A4_1C
   [VCS Label A4_1A] Foundation for tests C611A01, C611A02, and C611A03.
   [VCS Label A4_1C] Added additional data for test C611A04. The changes are only needed for that test.

F611B00 [New] VCS Label: A4_1A
Foundation for tests C611B01, C611B02, and C611B03.

FDD2B00 [New] VCS Label: A4_1Q
A foundation for Stream_Size aspect tests.

FXAIB00 [New] VCS Label: A4_1T
A foundation for container preelaboration tests.

GRADE.A [Modified] VCS Label: A4_1G
[VCS Label A4_1C] Allowed manual grading for tests that are missing execution or linking (with no earlier errors). Corrected the result for binder crashes. Added category totals to the summary.
[VCS Label A4_1D] Corrected so that Annex C RQMT failures are properly counted in their own category.
[VCS Label A4_1G] Corrected call @1578 so that the special handling flag rather than the failed flag is used with a manual grading message for a missing error set. (The actual grade was correct, only the message was wrong.)

GRD_DATA.A [Modified] VCS Label: A4_1P
[VCS Label A4_1D] Increased table sizes as the new tests have caused grading the entire ACATS to exceed the old limits.
[VCS Label A4_1M] Generalized reading of white space in order to be more resilient, especially on Unix-like systems.
[VCS Label A4_1P] MAX_SUMMARY_ITEMS @66 was increased to 40_000 as the ACATS has grown to exceed the old limit.

Required in ACATS 4.2
The ACATS Version string was moved into a separate package Version.

SPECIAL.A [Modified] VCS Label: A4_1J
[VCS Label A4_1C] Added test BA21005 to the list of optional main subprograms.
[VCS Label A4_1H] Added seven tests to the list of optional main subprograms.
[VCS Label A4_1J] Added more tests to the list of optional main subprograms.

SUMMARY.A [Modified] VCS Label: A4_1C
Corrected identification of main subprograms in modern tests. Changed the unit name for subunits to be the full expanded name including the parent name.

TCTOUCH.ADA [Modified] VCS Label: A4_1H
Added code so that Touch only reports buffer overflow once. Fixed the Id string.

Required in ACATS 4.2
The ACATS Version string is now found here.

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_1BB.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 4.1 is available in various formats on the ACAA web site.

**Summary lists:**

**Withdrawn Tests**

(Refer to the main list for details)

B74001A
B74001B
B74203E
B83004A
B83004B
B83004C
B83004D
B85002A
B85003A
B85003B
BA1101E
BC2001D
BC2001E
C95072A
CD1009A
CD1009B
CD1009L
CD1009M
CD2A21C
CD2A31C
CXA4034

**Allowed Modification Tests**

(Refer to the main list for details)

B32201A ACATS 4.2
B324001 ACATS 4.2
B33101A ACATS 4.2
B35901C  ACATS 4.2
B36102A  ACATS 4.2
B370002  ACATS 4.2
B38103A  ACATS 4.2
B38103B  ACATS 4.2
B38103C  ACATS 4.2
B38103D  ACATS 4.2
B38103E  ACATS 4.2
B393005  ACATS 4.2
B393008  ACATS 4.2
B393009  ACATS 4.2
B394001  ACATS 4.2
B394002  ACATS 4.2
B394003  ACATS 4.2
B394004  ACATS 4.2
B394005  ACATS 4.2
B41327B  ACATS 4.2
B415001  ACATS 4.2
B43201A  ACATS 4.2
B43201C  ACATS 4.2
B43202A  ACATS 4.2
B433002  ACATS 4.2
B433003  ACATS 4.2
B457002  ACATS 4.2
B457005  ACATS 4.2
B457007  ACATS 4.2
B460002  ACATS 4.2
B480002  ACATS 4.2
B48002A  ACATS 4.2
B48002G  ACATS 4.2
B48003B  ACATS 4.2
B48003E  ACATS 4.2
B490003  ACATS 4.2
B49005A  ACATS 4.2
B551001  ACATS 4.2
B56001A  ACATS 4.2
B57001B  ACATS 4.2
B611004  2023-01-01 [VCS Label A4_1BB]
B650003  ACATS 4.2
B650005  ACATS 4.2
B660001  ACATS 4.2
B660002  ACATS 4.2
B660003  ACATS 4.2
B66001A  ACATS 4.2
B66001B  ACATS 4.2
B67001C  ACATS 4.2
B730007  ACATS 4.2
B730008  ACATS 4.2
B730009  ACATS 4.2
B740002  ACATS 4.2
B74101A  ACATS 4.2
B74104A  ACATS 4.2
B830001  ACATS 3.2
B83001A  ACATS 3.2
B83003A  ACATS 3.2
B83003B  ACATS 3.2
Modified Tests

(Refer to the main list for details)

B33001A
B37309B
B37310B
B37311A
B38003D
B392002
B392003
B392004
B392005
B393001
B393007
B393010
B3A1001
B3A1002
B3A2007
B3A2008
B3A2009
B3A2011
B3A2012
B3A2016
B413004
B416001
B455002
B457002
B457001
B457004
B4A010C
New Tests

(Refer to the main list for details)

B324004
B350001
C611A04
C611B01
C611B02
C611B03
C620002
C641002
C641003
C650B04
C651001
C651002
C760A02
C760A03
C940017
CB42001
CB42002
CB42003
CD11001
CD30010
CD30011
CD30012
CD30013
CD30014
CDD2B01
CDD2B02
CDD2B03
CDD2B04
CXAC009
CXAG003
CXAIB01
CXAIB02
CXAIB03
CXAIB04
CXAIB05
CXAIB06
CXAIB07
CXAIB08
CXB3023
CXB3024
F350A00
F350B00
F360A00
F360B00
F611A00
F611B00
FDD2B00
FXAIB00

Pending New Tests

(Refer to the main list for details)

B431008  2022-10-01
B431009  2022-10-01
B433004  2023-04-01
B490004  2022-10-01
C431003  2022-10-01
C431004  2022-10-01