ASM

Versions

17 October 2026: ASM 9.11.1 (tag ASM_9_11_1)

- !480: fix regression introduced in 9.11 version (!460).
- !481: performance optimization in computeAllFrames(), fix LimitExceededException on some complex classes.

4 October 2026: ASM 9.11 (tag ASM_9_11)

- new Opcodes.V28 constant for Java 28 (no support of preview features yet)
- !460: store the forward references to a label inline in the bytecode, instead of in a separate array
- !464: improve AdviceAdapter for classes with stack map frames
- security fixes
    - !456: validate the BootstrapMethods attribute length
    - !457: more code_length validation in ClassReader.readCode()
    - !458: check that each subroutine has at most one ret instruction
    - !459: avoid recursion in replaceAsmInstructions()
    - !461: do not create Edge instances for the exception handlers
    - !462: large methods optimization
    - !463: add an API to provide control over the resources used by ClassWriter
    - !464: add a setComputeLimits() method in AdviceAdapter
    - !465: add a recursion limit in SignatureReader
    - !466: add an API to provide control over the resources used by JSRInlinerAdapter
    - !468: add an API to provide control over the resources used by Analyzer
    - !469: add an API to provide control over the resources used i asm.util
    - !474: add a recursion limit for ConstantDynamic constants in ClassReader
    - !475: make SimpleVerifier.isAssignableFrom() non-recursive
    - !476: add a recursion limit for annotations in ClassReader
    - !477: make Printer.printList() non-recursive and inline it

23 May 2026: ASM 9.10.1 (tag ASM_9_10_1)

- bug fixes
    - 318045: The retrofitter should also remove the (forRemoval) from the @Deprecated annotation associated with fields

14 May 2026: ASM 9.10 (tag ASM_9_10)

- new Opcodes.V27 constant for Java 27
- bug fixes
    - 318041: Compilation failure with JDK 5
    - 318043: TraceSignatureVisitor bug

6 December 2025: ASM 9.9.1 (tag ASM_9_9_1)

- bug fixes
  - 318036: OutOfMemoryError when reading invalid class
  - 318037: Version ranges too wide on Import-Package

4 October 2025: ASM 9.9 (tag ASM_9_9)

- new Opcodes.V26 constant for Java 26
- new mapInvokeDynamicMethodName method in Remapper. Old method deprecated. New Remapper constructor, with an api parameter.
- bug fixes
  - 318028: Textifier misinterprets ACC_SUPER of inner classes as ACC_SYNCHRONIZED
  - 318032: FIPS 140-3 and SerialVersionUIDAdder's SHA-1 Use
  - 318034: Many ASM contents lack API detection.

29 March 2025: ASM 9.8 (tag ASM_9_8)

- new Opcodes.V25 constant for Java 25
- bug fixes
  - Fix one more copy operation on DUP2
  - 318015: Valid bytecode for jvm, but failed to pass the CheckClassAdapter.
  - `ASMifier` should print calls to `valueOf` instead of deprecated constructors of primitive wrappers

6 October 2024: ASM 9.7.1 (tag ASM_9_7_1)

- new Opcodes.V24 constant for Java 24
- Javadoc improvements
- new features
  - 318013: new ClassWriter setFlags() method.
  - Add ConstantDynamic serialization and Number suffixes to Textifier.
- bug fixes
  - 318014: Analyzer with a SimpleVerifier may throw an AnalyzerException on valid java code due to incompatible frame locals.
  - 318015: Valid bytecode for jvm, but failed to pass the CheckClassAdapter.
  - 318016: ClassNotFoundException with an array of the type of current class.
  - 318018: changing invokedynamic Handle itf bool flag doesn't create new methodref in symbol table.
  - 318019: Attribute::write is invoked twice.
  - Fix DUP_X1, DUP_X2, DUP2_X1, and DUP2_X2 not copying values correctly in Analyzer.
  - Fix SimpleVerifier multiple dimensions merge.
  - Fix SourceInterpreter wrongly saying that Condy long / doubles are one word elements.
  - GeneratorAdapter: fix push(Type.VOID_TYPE).
  - Fix null method parameter name in Asmifier causing NPE.

23 March 2024: ASM 9.7 (tag ASM_9_7)

- new Opcodes.V23 constant for Java 23
- bug fixes
  - 318009: Unit test regression in dex2jar.
  - 318007: 'ClassNode#outerClass' has incorrect JavaDocs.
  - 318006: asm-bom packaging should be 'pom'.
  - 318003: The Textifier prints a supplementary space at the end of each method that throws at least one exception.

30 September 2023: ASM 9.6 (tag ASM_9_6)

- new Opcodes.V22 constant for Java 22
- bug fixes
  - 317991: Analyzer produces frames that have different locals than those detected by JRE bytecode verifier
  - 317995: Invalid stackmap generated when the instruction stream has new instruction after invokespecial to <init>
  - 317998: Analyzer can fail to catch thrown exceptions
  - 318002: asm-analysis Frame allocates an array unnecessarily inside executeInvokeInsn
  - bug in CheckFrameAnalyzer with static methods

24 March 2023: ASM 9.5 (tag ASM_9_5)

- new Opcodes.V21 constant for Java 21
- new readBytecodeInstructionOffset hook in ClassReader
- more detailed exception messages
- Javadoc improvements and fixes
- bug fixes
  - 317989: Silent removal of zero-valued entries from the line-number table

2 October 2022: ASM 9.4 (tag ASM_9_4)

- new Opcodes.V20 constant for Java 20
- more checks in CheckClassAdapter
- Javadoc improvements and fixes
- module-info classes can be built without Gradle and Bnd
- parent POM updated to org.ow2:ow2:1.5.1
- bug fixes
  - 317977: CheckClassAdapter is no longer transparent for MAXLOCALS
  - 317981: Add public getDelegate method to all visitor classes
  - Analyzer does not compute optimal maxLocals for static methods
  - Fix SignatureWriter when a generic type has a depth over 30
  - Skip remap inner class name if not changed in Remapper

4 April 2022: ASM 9.3 (tag ASM_9_3)

- new Opcodes.V19 constant for Java 19
- new size() method in ByteVector

- checkDataFlow option in CheckClassAdapter can now be used without valid maxStack and maxLocals values
- new Maven BOM
- bug fixes
    - 317949: fix javadoc errors
    - remap invokedynamic field handles properly
    - add missing left curly brace in ASMifier output of visitModule

26 June 2021: ASM 9.2 (tag ASM_9_2)

- JDK 18 support
- ClassReader.readStream() performance improvements
- Javadoc fixes
- bug fixes
    - 317942: ClassCheckAdaptor output represents reference arrays as only their "root" component type

6 February 2021: ASM 9.1 (tag ASM_9_1)

- JDK 17 support
- add some input validations in ClassReader
- replace -debug flag in Printer with -nodebug (-debug continues to work)
- bug fixes
    - 317921: Make Analyzer not require correct maxs or calculate maxs fast
    - 317922: InstructionAdapter shall throw consistent exceptions
    - 317923: Incorrect documentation on ClassReader's use of AnnotationVisitor
    - 317929: ClassRemapper doesn't remap annotation values
    - 317930: Can CheckClassAdapter.checkTypeRef throw a more specific exception?
    - 317931: Shall CheckMethodAdapter.Method.visitLabel throw IllegalStateException?
    - Tests that requires preview features should only run if the preview features of the right JDK are enabled

22 September 2020: ASM 9.0 (tag ASM_9_0)

- bug fixes
    - 317897: AdviceAdapter incorrectly emits onMethodEnter from a try-catch block in a constructor

14 July 2020: ASM 9.0 beta (tag ASM_9_0_BETA)

- JDK 16 support (sealed classes), new ASM9 api version
- bug fixes
    - 317900: Javadoc of ClassVisitor does not specify order of visitRecordComponent
    - 317904: Bug with constructors in AdviceAdapter
    - 317910: Iterating over a InsnList with an iterator traverses the list twice

3 April 2020: ASM 8.0.1 (tag ASM_8_0_1)

- bug fixes
    - 317898: ASM 8 does not support writing of empty records

28 March 2020: ASM 8.0 (tag ASM_8_0)

- Java 14 support (RecordComponent)
- bug fixes
    - 317896: Performance degradation when using dynamic constants as a static paramet to another InDy/ConDy

11 January 2020: ASM 7.3.1 (tag ASM_7_3_1)

- new Opcodes.V15 constant for Java 15
- experimental support for PermittedSubtypes and RecordComponent
- bug fixes
    - 317885: SKIP_DEBUG now skips MethodParameters attributes

29 September 2019: ASM 7.2 (tag ASM_7_2)

- InsnList now implements Iterable interface
- bug fixes
    - 317875: Infinite loop when parsing invalid method descriptor

6 July 2019: ASM 7.2 beta (tag ASM_7_2_BETA)

- add support for Java 14
- fix missing 'final' on some ClassWriter methods
- ASMifiable and Textifiable moved to asm-deprecated.jar
- small refactors to reduce code duplication
- delete unused code and constants
- bug fixes
    - 317873: Add support for RET instruction in AdviceAdapter
    - 317872: Throw an exception if visitFrame used incorrectly

3 March 2019: ASM 7.1 (tag ASM_7_1)

- new Opcodes.V13 constant for Java 13
- small optimizations in asm.Type
- check signatures in CheckMethodAdapter.visitLocalVariable
- refactor unit tests to use Arrange-Act-Assert pattern
- deprecations
    - deprecate ClassReader.b
    - deprecate ASMifiable and Textifiable (replaced with ASMifierSupport and TextifierSupport)
    - move deprecated Remapping*Adapter classes to new asm-deprecated.jar
    - delete asm.util.Printer buf and appendString (were previously deprecated)
- bug fixes
    - 317863: Remapper logic for InnerClasses attribute is wrong for method-local Java classes
    - 317866: Expanded frames result in incorrect types for primitive arrays of more than seven dimensions
    - 317868: IllegalArgumentException at Frame.getAbstractTypeFromDescriptor on class name with parentheses

    - 317869: CheckClassAdapter requires non-null ClassVisitor that returns non-null MethodVisitor

27 October 2018: ASM 7.0 (tag ASM_7_0)

- rollback of !177: Ensure instructions belong only to one list
- new initJumpTarget method in Frame

29 September 2018: ASM 7.0 beta (tag ASM_7_0_BETA)

- support of JDK 11 features now official (was experimental)
- xml package deleted (was deprecated since 6.1)
- small improvements
    - Type and TypePath now explicitly final
    - Eliminate double-spaced output from Textifier
    - Remap inner class names in Remapper
    - Better UnsupportedOperationException messages
    - Javadoc improvements
    - Check code quality with Checkstyle and PMD
    - Fix performance regression introduced in 6.2
- bug fixes
    - 317847: Gradle build missing Implementation-Version property
    - 317848: Type names no longer included in declaration comments produced by Textifier
    - 317851: Analyzer gets StackOverflowError when method has lots of jumps
    - 317853: SimpleVerifier merges types incorrectly when one type is an interface
    - 317856: Bug in AdviceAdapter with superclass constructors
    - MethodWriter generates incorrect bytecode for long or double ConstantDynamic
    - Remapper does not remap ConstantDynamic and Handle correctly

5 August 2018: ASM 6.2.1 (tag ASM_6_2_1)

- small improvements
    - !177: Ensure instructions belong only to one list
    - !178: Allow setting stack values in analysis frames
    - !179: Multiple methods for initializing analysis values
    - !181: Improve exception messages when exceeding classfile limits
    - !182: Minor Javadoc improvements
- bug fixes
    - 317609: Enhance ClassWriter to take a class loader for use with getCommonSuperClass
    - 317833: When ClassWriter copies constant pool of class reader, ClassWriter.COMPUTE_MAX does not consider changes in argument sizes
    - 317835: COMPUTE_MAX_STACK_AND_LOCAL_FROM_FRAMES does not handle LONG and DOUBLE types correctly
    - 317837: Apparent regression in GeneratorAdapter.java `cast` function
    - 317843: Bug in CheckMethodAdapter

24 May 2018: ASM 6.2 (tag ASM_6_2)

- experimental support for the upcoming JDK 11 features (nest mates and ConstantDynamic)
- code quality and readability of commons package improved a lot
- bug fixes
    - 317817: Allow Kotlin back-tick identifiers by CheckMethodAdapter
    - 317821: Regression in 6.1: "new Frame" instead of "newFrame" in Analyzer
    - 317822: ACC_DEPRECATED flag (or corresponding attribute) is lost if the constant pool is copied
    - 317823: Computing maximum stack size does not work correctly with JSR/RET instructions present
    - 317824: ACC_SYNTHETIC attribute is lost for < Java 1.5 if the constant pool is copied
    - 317825: idem
    - 317827: Computing maximum stack size does not work correctly with dead code present

26 March 2018: ASM 6.1.1 (tag ASM_6_1_1)

- bug fixes
    - 317815: ClassReader.readCode() throws NullPointerException.
    - 317816: ClassReader constructor goes into infinite loop on zero length input streams

11 March 2018: ASM 6.1 (tag ASM_6_1)

Diffs from 6.0:

- new V10 constant for Java 10 class files
- xml package deprecated
- project layout changed to Maven style
- build system changed from Ant to Gradle
- coding style changed to Google Java style
- code quality and readability improved a lot
- code quality tracked with SonarQube
- performance improvements
- unit tests improved: migrated to JUnit 5, faster, coverage > 95%
- non-regression tests added for past issues
- benchmarks improved: now use JMH
- new Gitlab Continuous Integration pipeline
- bug fixes:
    - 317800: Resizing jump instruction adds invalid stack frame to a class without frames
    - 317802: MethodRemapper.visitInvokeDynamicInsn mutates source bsmArgs
    - 317804: Adjust CheckModuleAdapter so Module attribute in 54.0+ class file cannot contains a requires java.base with ACC_TRANSITIVE or ACC_STATIC_PHASE
    - 317807: Asymmetry of ClassWriter for bytecode versions higher than the one that ClassReader accepts
    - 317809: CheckMethodAdapter rejects valid Java and JVM identifiers with surrogate characters
    - small bug fixes related to modules (https://gitlab.ow2.org/asm/asm/merge_requests/20/, https://gitlab.ow2.org/asm/asm/merge_requests/19/)
- backward binary compatibility is preserved, but the behavior at runtime has changed in a few areas:
    - ACC_SYNTHETIC flag vs Synthetic attribute, see https://gitlab.ow2.org/asm/asm/merge_requests/17
    - numbering of synthetic parameters: see https://gitlab.ow2.org/asm/asm/merge_requests/56
    - StringBuffer no longer used in util (Printer.buf deprecated, use stringBuilder instead)

23 September 2017: ASM 6.0 (tag ASM_6_0)

Diffs from 5.2:

- Codebase migrated to gitlab (feature requests 317617, 317619, 317542)
- Support for modules (JSR 376, feature request 317628)
- Support for JDK specific module attributes (ModuleTarget, ModuleHashes and ModuleResolution).
- ASM jars now contain debug info, asm-all jar removed (feature request 317622)
- Support for removable features deleted (cf SIGNATURES, ANNOTATIONS, etc constants in ClassReader)
- bug fixes
    - 317793: AnalyzerAdapter calculates wrong type for AALOAD in case of null
    - 317784: NegativeArraySizeException when resizing jump instruction in a class without frames
    - 317780: IllegalArgumentException at CheckMethodAdapter.checkInternalName for non-Java-identifier name
    - 317785: NullPointerException in MethodRemapper with 'chop' stack map frames and tree API
    - 317786: VerifyError when added code pushes the offset of a conditional backward jump below -32768
    - 317791: Invalid class files generated for large methods (regression introduced in 5.2)
    - 317792: Fix the handling of inner classes with EXPAND_ASM_INSNS
    - Bug fix related to stack map frames

16 July 2017: ASM 6.0 beta (tag ASM_6_0_BETA)

Diffs from 6.0 alpha:

- Codebase migrated to gitlab (feature requests 317617, 317619, 317542)
- Support for modules updated to the latest version of JSR 376
- Support for JDK specific module attributes (ModuleTarget, ModuleHashes and ModuleResolution).
- bug fixes
    - 317749: ClassCheckAdapter should discover invocation of default method from pre-Java 8 class file
    - 317626: AnnotationNode incorrectly process array values in 'visit(final String name, final Object value)' method
    - 317630: Frames are re-calculated for Large Methods
    - 317748: Class after instrumenting loses first BootstrapMethods entry
    - 317784: NegativeArraySizeException when resizing jump instruction in a class without frames
    - 317780: IllegalArgumentException at CheckMethodAdapter.checkInternalName for non-Java-identifier name
    - 317785: NullPointerException in MethodRemapper with 'chop' stack map frames and tree API
    - 317786: VerifyError when added code pushes the offset of a conditional backward jump below -32768

23 December 2016: ASM 5.2 (tag ASM_5_2)

Diffs from 5.1:

- Better comments in Opcodes.java
- SuppressWarnings in Opcodes.java for Integer constructor deprecation.
- More tests to check binary backward compatibility
- bug fixes
    - 317616: Misleading comment in ClassWriter

    - 317621: Type.getReturnType(desc) returns wrong for types with parenthesis in name
    - 317749: ClassCheckAdapter should discover invocation of default method from pre-Java 8 class file
    - 317626: AnnotationNode incorrectly process array values in 'visit(final String name, final Object value)' method
    - 317630: Frames are re-calculated for Large Methods
    - 317748: Class after instrumenting loses first BootstrapMethods entry

19 June 2016: ASM 6.0 alpha (tag ASM_6_0_ALPHA)

Diffs from 5.1:

- Preliminary support for modules
- bug fixes
    - 317616: Misleading comment in ClassWriter
    - 317621: Type.getReturnType(desc) returns wrong for types with parenthesis in name

5 March 2016: ASM 5.1 (tag ASM_5_1)

- Remapping...Adapter classes are now deprecated, replaced with ...Remapper classes.
- Add support for method reference on an interface.
- bug fixes
    - 317567: SerialVersionUIDAdder issues.
    - 317578: incorrect max code size check in MethodWriter getSize().
    - 317579: Signature Process in SignatureRemapper is Broken.
    - 317581: Import-Package headers should denote version by setting the version attribute.
    - 317586: VerifyError caused by LocalVariablesSorter.
    - 317602: Asmifier does not recognize null value for TypePath.
    - 317604: Bug in AdviceAdapter.visitFieldInsn().
    - 317607: Calling add or set on an InsnListIterator throws NPE when at end of list.
    - 317606: Frequent use of StringBuffer instead of StringBuilder is hurting performance.

15 May 2015: ASM 5.0.4 (tag ASM_5_0_4)

- bug fixes
    - Use the pattern .<methodname><methoddesc> to specify mapInvokeDynamicMethodName remappings in SimpleRemapper.
    - 317555: NPE in MethodWriter.resizeInstructions().
    - 317548: Mistakes in API documentation of MethodVisitor and SignatureVisitor.
    - 317545: Asm eat linenumbers if there are several of them on one label.
    - 317539: TypePath not adequately supporting types w/ >10 type arguments.
    - 317538: AnnotationNode in ASM 5.0.3 does not handle null returned from AnnotationVisitor correctly.
    - Bug fix in GeneratorAdapter.catchException.
    - 317534: Modernize ASM code: replace new WrapperType() by autoboxing.
    - 317533: MethodInsnNode.accept() doesn't call acceptAnnotations().

21 April 2014: ASM 5.0.2 (tag ASM_5_0_2)

- bug fixes
  - 317135 Textifier prints bridge methods to be volatile.
  - 317151 Fix for 316545 is incorrect.
  - 317136 ASMifier 5.0.1 output references package org.objectweb.asm.attrs.

23 March 2014: ASM 5.0.1 (tag ASM_5_0_1)

- bug fixes
  - 317132 ASM 5.0 do not supported JDK 1.5?
  - 317131 Infinite loop in InstructionAdapter.invokestatic(...).
  - 316691 Incorrect StackMapFrame calculations for large > 32k methods in Java 7.
  - 317123 ASM5.0: Class MethodNode broken?
  - 317127 duplicate entries in InnerClasses attribute.

16 March 2014: ASM 5.0 (tag ASM_5_0)

- visitMethodInsn(int,String,String,String) is deprecated, and replaced with visitMethodInsn(int,String,String,String,boolean), in order to support invokespecial and invokestatic on interfaces.
- Added a getMaxStackSize method in asm.tree.analysis.Frame.
- bug fixes
  - 317111 source debug extension size should not be limeted to 64k
  - 316682 maxLocals not updated in AnalyzerAdapter constructor.
  - 316630 Type verification in CheckAnnotationAdapter.
  - 316545 ASM incorrectly merges Object[] and byte[][] in StackMapTable.
  - 316555 Wrong maxLocals calculation in AnalyzerAdapter.
  - 316553 InsnListIterator can't get previous entry after exec "remove()".

12 October 2013: ASM 5.0 beta (tag ASM_5_0_BETA)

- Full support of Java 8 (annotation on Java types and MethodParameters attribute)

12 October 2013: ASM 4.2 (tag ASM_4_2)

- bug fixes
  - 316506 ByteVector doesn't validate byteLength in putUTF8().
  - Javadoc typos.
  - 316482 JarOptimizer removes directory entries in resulting jars.
  - Don't create a new ClassWriter instance in toByteArray, in order for this code to work with subclasses.
  - 316416 JSRInlinerAdapter fails with IndexOutOfBoundsException.
  - 316403 LocalVariablesSorter incorrectly assumes it can call visitFrame() with original args.
  - 316373 RemappingSignatureAdapter.visitInnerClassType is broken when handling identifiers with "$" signs.
  - 316380 Bug in org.objectweb.asm.util.CheckClassAdapter.visitInnerClass() regarding inner class name.

14 October 2012: ASM 4.1 (tag ASM_4_1)

- implicit first frame no longer visited with EXPAND_FRAMES (was necessary to fix bug 316360).

- performance improvements in ClassReader (about +10 percent speedup).
- feature requests
  - 316306: More efficient patch for Frame.merge().
  - 316318: Allow generated SVUID field to be marked synthetic.
  - 316321: asm osgi bundles based on BND.
  - Moved checkClassSignature, checkMethodSignature and checkFieldSignature to CheckClassAdapter and made them public.
  - 316345: Support "synthetic" access modifier in org.objectweb.asm.util.Textifier.appendAccess().
- bug fixes
  - 316360: MethodWriter.visitFrame(F_NEW,...) requires explicit invocation for the implicit frame at offset zero.
  - 316352: LocalVariableSorter.visitFrame() logic is incorrect.
  - 316326: analysis.Frame.merge() should use !.equals() instead of !=.
  - Fix regression: LDC with a primitive class doesn't work anymore.
  - 316347: GeneratorAdaptor.ifCmp() generates DCMPG/DCMPL incorrectly.
  - 316348: ClassVersion check in ClassReader looks at wrong offset.
  - 316359: ClassWriter toByteArray fails when there are more than 32k constant pool entries.

29 October 2011: ASM 4.0 (tag ASM_4_0)

- added automatic test of backward binary compatibility in build process.
- added automatic call to resetLabel in MethodNode when necessary.
- improved printing of method handles in Textifier.
- bug fixes
  - 316204 Analyzer produces incorrect lvt types for jsr return targets when jsr frame changes twice during analysis.
  - 316212 org.objectweb.asm.optimizer.ClassOptimizer does not work with classes in default package.
  - fixed incorrectly ASMified method handles.

11 September 2011: ASM 4.0 RC2 (tag ASM_4_0_RC2)

- ClassVisitor, MethodVisitor, etc interfaces replaced with abstract classes, in order to ensure backward binary compatibility in future ASM versions.
- ClassAdapter and MethodAdapter merged into ClassVisitor and MethodVisitor, classes using multiple interface inheritance removed (e.g. MemberNode, EmptyVisitor) or adapted (e.g. trace and asmifier classes in asm.util).
- MethodHandle and MethodType renamed to Handle and Type.
- improved getCommonSuperClass to avoid resolving loaded classes.
- improved getCommonSuperClass in ClassWriterComputeFramesTest (more efficient).
- new checks in CheckClassAdapter.
- new example, showing how to use the new invokedynamic instruction.
- bug fixes
  - 315942 asm-xml.dtd fails to mention SWAP element
  - 315984 asm-xml.dtd fails to mention FREM element
  - NOP ATHROW bug (compute frames failed on deadcode in exception handler range)
  - fixed bugs found with FindBugs
  - minor Javadoc improvements and fixes
- feature requests
  - 316005 Constant pool access

    - 313999 Maven POM should include source-code
    - patch from Ivo Anjo in Analyzer

17 April 2011: ASM 4.0 RC1 (tag ASM_4_0_RC1)

- full support of Java 7
- support of the new invokedynamic encoding and bootstrap method arguments. Done with a new visitInvokeDynamicInsn method.
- added two new classes MethodHandle and MethodType. Can be used either in visitLdcInsn or as last arguments of visitInvokeDynamicInsn.
- generified the API to use generics and varargs. However, almost all jars are still small and 1.2 compatible.

21 December 2010: ASM 3.3.1 (tag ASM_3_3_1)

- bug fixes
    - 315135 COMPUTE_FRAMES with CheckClassAdapter gives IndexOutOfBoundsException
    - 315248 Bug in CheckClassAdapter? Fails for untransformed class
    - 315399 AnalyzerAdapter.execute() method doesn't process null array correctly for AALOAD instruction
    - 315417 LocalVariablesSorter.remap() incorrectly process long/double types
    - 315697 DependencyTracker example fails in handling class Object
    - 315698 DependencyTracker example fails in handling finally blocks

30 May 2010: ASM 3.3 (tag ASM_3_3)

- added new InstructionAdapter class
- added TryCatchBlockSorter (provided by Adrian Sampson)
- improved extensibility of analysis package (patch from Markus Heiden)
- bug fixes
    - 314969 Counterintuitive interaction of arguments to ClassWriter(cr, flags)
    - 314964 AnalyzerAdapter does not treat <init> calls correctly
    - 314934 ASM Handling of method Synthetic attribute causes exception during retransformation
    - 314878 NPE in AnalyzerAdapter, putfield object comes from different control flow branches
    - fixed bug with remapping of RET instructions
    - 314563 exception from asm.commons.LocalVariablesSorterremap
    - fixed bug in GASMifier
    - fixed bug for F_CHOP frames (locals are ignored for these frames)
    - 314719 method length is sometimes calculated incorrectly
    - 314584 ByteCode Outline Eclipse Plugin -- access flags comment is showing in decimal rather than hex
    - 314481 Bug in Remapper related to annotations
    - 314486 SerialVersionUIDAdder generates an incorrect serialVersionUID under 1.6 VMs for anonymous classes
    - fixed bug with padding for xxxSwitch instructions
    - 313909 VOID_TYPE.getSize() != 0
    - 312464 ClassReader does not ensure that NEW has a label associated with it
- patches (minor code improvements)
    - 314328 minor improvement to serialveruidadder
    - 314329 unnecessary casting, using HashMap v. Map

    - 314330 make sure files get closed
- feature requests
    - 314119 Provide read access to uninitializedTypes in AnalyzerAdapter
    - 313804 Improve analysis results
- Javadoc fixes and improvements

11 June 2009: ASM 3.2 (tag ASM_3_2)

- added support for new invokedynamic opcode
- added new Type.getArgumentsAndReturnSizes method
- added new GeneratorAdapter.valueOf method
- added two new Method.getMethod methods (feature request #308554)
- added new checks in Check*Adapter
    - check of xRETURN instructions against declared method return type
    - check that labels are not reused across methods (inside same class; check for labels reuse across classes still missing)
    - checks for debug labels and for visitTryCatchBlock calling order
    - new option for data flow checks with BasicVerifier
- added Jasmin class adapter (with round trip tests) in examples
- bug fixes
    - 313314 ClassReader.readClass(InputStream) unnecessary copies read buffer two times
    - 311642 StackOverflowException in Label.visitSubroutine
    - 311100 patch to make verifying multiple classes easier
    - 309819 java/lang/Character typo in GeneratorAdapter
    - 312468 Incorrect merging of frames
    - 309037 GeneratorAdapter generates jump instructions wrongly
    - 308875 ClassWriter COMPUTE_MAXS produces invalid class (verify error)
    - 308807 AnalyzerException running CheckClassAdapter.verify() on java.lang.String
    - 308745 BasicInterpreter.newOperation lacks a throws AnalyzerException clause
    - fixed bug submitted by Simon Goldsmith for inner classes
    - fixed bug submitted by Michael Rudolf
    - fixed bug with usage of copyOperation
    - fixed bug in test case
- Javadoc fixes and improvements
- added performance tests with JBET, JClassLib, Jiapi, mozilla.classfile, BCEL and AspectJ BCEL using HelloWorld example
- added performance tests with Cojen library
- added performance tests for gnu.bytecode and CSG bytecode using HelloWorld example
- added FindBugs preferences and builder for ASM
- added debug jar
- improved Ant build scripts
    - replaced ow_util_ant_tasks (no longer maintained) with standard ant tasks
    - updated test/lib jars with newer versions
    - removed hardcoded JUNIT Eclipse path, made JSE 1.5.0 container as default VM (previously random system JRE)
    - added license info to the parent pom

29 October 2007: ASM 3.1 (tag ASM_3_1)

- extended semantic of 'internal name' to array classes (internal name equal to descriptor in this case); renamed 'desc' parameter of 'visitTypeInsn' to 'type'
- new readLabel method in ClassReader
- new getLabelNode method in MethodNode (see bug 306920)
- new resetLabel(s) methods in LabelNode and InsnList (feature request 306499)
- new checks in CheckMethodVisitor (for stack frames)
- new CheckSignatureAdapter class
- new Remapper* classes in commons
- support for primitive types in GeneratorAdaptor.push (feature request 307378)
- new build options (no writer, no frames support, etc)
- new tests
- bug fixes
    - 307600 StackOverflowError in org.objectweb.asm.MethodWriter.findSubroutine()
    - 307392 MethodWriter generates invalid RuntimeInvisibleParameterAnnotations for enum constructors
    - 307242 Patch to compile ASM 2.2.3 against Java 1.4
    - 307011 Typo error in ASM 3.0 User Guide
    - 306921 SignatureReader does not handle inner class extends of other inner class
    - 306920 MethodNode.labelNode(label) smashes "user" data Label.info
    - 306521 Parameter local of visitFrame() is lost when using a ClassNode
    - 306503 CheckClassAdapter doesn't flush its output
    - 306422 Another code cleanup patch
    - 306349 SerialVersionUIDAdder breaks compile if using JDK 1.4
    - bug fix related to xSTORE in tree.analysis.Frame
    - bug fix in LocalVariableSorter
    - javadoc fixes

1 November 2006: ASM 3.0 (tag ASM_3_0)

- visit*Annotation methods can now return null
- COMPUTE_MAXS now works with all subroutines, even very weird ones that cannot be produced by javac
- new getters in ClassReader, SKIP_DEBUG now skips visitSource
- new methods in Type: getObjectType, getConstructorDescriptor
- InsnList now supports ListIterator interface
- new insertBefore() and insert*() methods in InsnList class
- improved Analyzer API, support for abstract and native methods
- improved Method class behavior with default package
- minor bug fixes
    - 305716 bug in floating point constant handling in constant pool writing
    - 305917 optimizer.ClassConstantsCollector generates an invalid class constant
    - 306058 FieldNode.accept() cannot handle ClassVisitors yielding null as FieldVisitor
- new tests
- javadoc improvements

06 July 2006: ASM 3.0 RC1 (tag ASM_3_0_RC1)

- new API for instruction lists in tree package
- new JSRInliner class in commons
- new CodeSizeEvaluator class in commons

- changes and optimizations in AnalyzerAdapter
- DataflowInterpreter renamed into SourceInterpreter
- port of changes from ASM 2.x

06 July 2006: ASM 2.2.3 (tag ASM_2_2_3)

- compute maxs option now takes visitLocalVariable into account
- bug fix in TraceSignatureVisitor
- minor Javadoc improvements

20 April 2006: ASM 3.0 beta2 (tag ASM_3_0_BETA2)

- visitFrame is now called after visitLabel in ClassReader
- new SKIP_FRAMES and SKIP_CODE flags in ClassReader
- fixed issue with dead code and COMPUTE_FRAMES option in ClassWriter
- port of bug fixes from ASM_2_2_2
- refactorings in core package to get same performances as in ASM 2.x (3.0 beta was slightly less performant than 2.x)
- refactored tests and added many new tests. Tests now cover 100% of the code, except in commons package.

7 April 2006: ASM 2.2.2 (tag ASM_2_2_2)

- bug fix in ClassWriter for SourceDebugExtension attribute
- bug fix in LocalVariableSorter when same variable slot is used to save different types
- all fields in classes ClassNode, MethodNode, LookupSwitchInsnNode and TableSwitchInsnNode of the tree package are mutable (non-final)
- additional checks in CheckClassAdapter, CheckMethodAdapter and CheckAnnotationAdapter
- new constructor in SimpleVerifier with list of interfaces of the class being verified to allow isAssignableFrom check without loading this class (which may be invalid)

14 December 2005: ASM 3.0 beta (tag ASM_3_0_BETA)

- Major changes for the new Java 6 StackMapTable attribute
  - StackMapTable support integrated in core package (new visitFrame method in MethodVisitor)
  - asm.attrs package removed
  - new commons.AnalyzerAdapter class
  - incremental stack map frames update in LocalVariablesSorter
- ClassReader and ClassWriter constructors changed
- ...

14 December 2005: ASM 2.2.1 (tag ASM_2_2_1)

- bug fix in ClassReader for empty annotation array values
- new accept method in MethodNode
- new checks in asm.util
- performance optimizations in LocalVariablesSorter

- more precise type system used in analysis package
- javadoc fixes and improvements

24 November 2005: ASM 2.2 (tag ASM_2_2)

- pass through behavior for unknown attributes
- optimizations for "mostly add" transformations (new ClassWriter constructor)
- removed constraint for visitTryCatchBlock call order with respect to visitLabel
  - visitTryCatchBlock can be called before or after its labels are visited
  - ClassReader now calls visitTryCatchBloc at the beginning of the method body
- new AdviceAdapter class in commons
- new constructor in GeneratorAdapter for better integration with the plain visitors/adapters
- fixed potential endles loop in IntMap.get()
- bug fixes in xml package
- improved javadocs (commons, tree)
- new tests
- corbertura added to the build

25 August 2005: ASM 2.1 (tag ASM_2_1)

- support for StackMapTable attribute in asm.attrs
- restructured StackMap attribute implementation (SM and SMT are using the same frames)
- addition of Traceable interface in asm.util.attrs
- new "dependencies" example
- improved commons.Method
- deprecated asm.tree.Insn.getByOpcode()
- public constructor in InsnNode
- better exception handling in analyzer
- added Opcodes.V1_6 constant
- javadoc improvements
- changed code style (Eclipse formatter + Sun conventions with minor changes)
- asm-all jar
- bug fixes
  - correct handling of synthetic attribute vs. synthetic flag
  - bug in TraceClassVisitor
  - bug in analyzer.Frame for ASTORE
  - added BasicValue.hashCode() and DataflowValue.hashCode()
  - bug about size calculation for custom attributes

17 May 2005: ASM 2.0 (tag ASM_2_0)

- Bug fixes

7 February 2005: ASM 2.0.RC1 (tag ASM_2_0_RC1)

- Bug fixes

- New visitCode() method in MethodVisitor to easily detect first instruction
- signature support (contribution from Thomas Hallgren)
- commons package (contribution from CLIB and other projects; not yet stable)
- memory usage optimization in tree package (contribution from Rene Treffer)
- performance improvements in core package
- performance improvements in analysis package (dataflowinterpreter is now two times faster)
- asm library plugin for Eclipse, used by Bytecode Outline plugin

21 January 2005: ASM 1.5.3 (tag ASM_1_5_3)

- Bug fixes
- New behavior for visitLineNumber

8 November 2004: ASM 2.0 alpha (tag ASM_2_0_ALPHA)

- Major API changes for Java 5 attributes
  - added new FieldVisitor interface
  - added new AnnotationVisitor interface
  - renamed CodeVisitor into MethodVisitor
  - renamed Constants into Opcodes
  - added and changed methods in ClassVisitor and CodeVisitor
  - removed almost all classes in attrs package
  - removed TreeClassAdapter and TreeCodeAdapter
  - added new MemberNode, AnnotationNode and LabelNode classes
  - many changes in asm.util package
- Added many new tests, reorganized tests classes and scripts

8 November 2004: ASM 1.5.2 (tag ASM_1_5_2)

- Bug fixes

23 September 2004: ASM 1.5.1 (tag ASM_1_5_1)

- Cleaned up ASMifierClassVisitor
- Removed AnnotationElementValue
- Added more tests for attributes and Java 5 annotations
- Added new annotations example
- Build package for Maven's ibiblio repository
- Bug fixes

30 August 2004: ASM 1.5 (tag ASM_1_5)

- Added support for class version in ClassVisitor
- Changed behavior for unrecognized attributes
- Added new analyzer in tree.analysis package (contribution from Bing Ran)
- Improved build files

- Bug fixes

11 May 2004: ASM 1.4.3 (tag ASM_1_4_3)

- Added tree.analysis package
- Added xml package, contributed by Eugene Kuleshov
- Bug fixes

1 March 2004: ASM 1.4.2 (tag ASM_1_4_2)

- Improved build files to facilitate Linux packaging
- Bug fixes

12 December 2003: ASM 1.4.1 (tag ASM_1_4_1)

- Added attrs package, contributed by Eugene Kuleshov

13 November 2003: ASM 1.4 (tag ASM_1_4)

- Added attribute support

3 November 2003: ASM 1.3.6 (tag ASM_1_3_6)

- Changed license from LGPL to BSD

8 October 2003: ASM 1.3.5 (tag ASM_1_3_5)

- Bug fixes
- Added thread safety support in ClassReader
- Added equals and hashcode in Type
- Added FAQ

9 July 2003: ASM 1.3.4 (tag ASM_1_3_4)

- Added kASM jar (needed a small refactoring)

3 October 2002: ASM 1.3.3 (tag ASM_1_3_3)

- Added tree package
- Added support for GOTO_W and JSR_W
- Added an ant target to create a debian package

6 September 2002: ASM 1.3.2 (tag ASM_1_3_2)

- Added util package

6 September 2002: ASM 1.3.1 (tag ASM_1_3_1)

- Bug fix

11 July 2002: ASM 1.3 (tag ASM_1_3)

- First public version