6.4.1 (2020-03-09)

Bug fixes

More carefully check for valid UTF16 surrogate pairs in regexp validator.

7.1.1 (2020-03-01)

Bug fixes

Treat `\8` and `\9` as invalid escapes in template strings.

Allow unicode escapes in property names that are keywords.

Don't error on an exponential operator expression as argument to `await`.

More carefully check for valid UTF16 surrogate pairs in regexp validator.

7.1.0 (2019-09-24)

Bug fixes

Disallow trailing object literal commas when ecmaVersion is less than 5.

New features

Add a static `acorn` property to the `Parser` class that contains the entire module interface, to allow plugins to access the instance of the library that they are acting on.

7.0.0 (2019-08-13)

Breaking changes

Changes the node format for dynamic imports to use the `ImportExpression` node type, as defined in ESTree.

Makes 10 (ES2019) the default value for the `ecmaVersion` option.

6.3.0 (2019-08-12)

New features

`sourceType: "module"` can now be used even when `ecmaVersion` is less than 6, to parse module-style code that otherwise conforms to an older standard.

6.2.1 (2019-07-21)

Bug fixes

Fix bug causing Acorn to treat some characters as identifier characters that shouldn't be treated as such.

Fix issue where setting the `allowReserved` option to `"never"` allowed reserved words in some circumstances.

6.2.0 (2019-07-04)

Bug fixes

Improve valid assignment checking in `for/in` and `for/of` loops.

Disallow binding `let` in patterns.

New features

Support bigint syntax with `ecmaVersion` >= 11.

Support dynamic `import` syntax with `ecmaVersion` >= 11.

Upgrade to Unicode version 12.

6.1.1 (2019-02-27)

Bug fixes

Fix bug that caused parsing default exports of with names to fail.

6.1.0 (2019-02-08)

Bug fixes

Fix scope checking when redefining a `var` as a lexical binding.

New features

Split up `parseSubscripts` to use an internal `parseSubscript` method to make it easier to extend with plugins.

6.0.7 (2019-02-04)

Bug fixes

Check that exported bindings are defined.

Don't treat `\u180e` as a whitespace character.

Check for duplicate parameter names in methods.

Don't allow shorthand properties when they are generators or async methods.

Forbid binding `await` in async arrow function's parameter list.

6.0.6 (2019-01-30)

Bug fixes

The content of class declarations and expressions is now always parsed in strict mode.

Don't allow `let` or `const` to bind the variable name `let`.

Treat class declarations as lexical.

Don't allow a generator function declaration as the sole body of an `if` or `else`.

Ignore `"use strict"` when after an empty statement.

Allow string line continuations with special line terminator characters.

Treat `for` bodies as part of the `for` scope when checking for conflicting bindings.

Fix bug with parsing `yield` in a `for` loop initializer.

Implement special cases around scope checking for functions.

6.0.5 (2019-01-02)

Bug fixes

Fix TypeScript type for `Parser.extend` and add `allowAwaitOutsideFunction` to options type.

Don't treat `let` as a keyword when the next token is `{` on the next line.

Fix bug that broke checking for parentheses around an object pattern in a destructuring assignment when `preserveParens` was on.

6.0.4 (2018-11-05)

Bug fixes

Further improvements to tokenizing regular expressions in corner cases.

6.0.3 (2018-11-04)

Bug fixes

Fix bug in tokenizing an expression-less return followed by a function followed by a regular expression.

Remove stray symlink in the package tarball.

6.0.2 (2018-09-26)

Bug fixes

Fix bug where default expressions could fail to parse inside an object destructuring assignment expression.

6.0.1 (2018-09-14)

Bug fixes

Fix wrong value in `version` export.

6.0.0 (2018-09-14)

Bug fixes

Better handle variable-redefinition checks for catch bindings and functions directly under if statements.

Forbid `new.target` in top-level arrow functions.

Fix issue with parsing a regexp after `yield` in some contexts.

New features

The package now comes with TypeScript definitions.

Breaking changes

The default value of the `ecmaVersion` option is now 9 (2018).

Plugins work differently, and will have to be rewritten to work with this version.

The loose parser and walker have been moved into separate packages (`acorn-loose` and `acorn-walk`).

5.7.3 (2018-09-10)

Bug fixes

Fix failure to tokenize regexps after expressions like `x.of`.

Better error message for unterminated template literals.

5.7.2 (2018-08-24)

Bug fixes

Properly handle `allowAwaitOutsideFunction` in for statements.

Treat function declarations at the top level of modules like let bindings.

Don't allow async function declarations as the only statement under a label.

5.7.0 (2018-06-15)

New features

Upgraded to Unicode 11.

5.6.0 (2018-05-31)

New features

Allow U+2028 and U+2029 in string when ECMAVersion >= 10.

Allow binding-less catch statements when ECMAVersion >= 10.

Add `allowAwaitOutsideFunction` option for parsing top-level `await`.

5.5.3 (2018-03-08)

Bug fixes

A *second* republish of the code in 5.5.1, this time with yarn, to hopefully get valid timestamps.

5.5.2 (2018-03-08)

Bug fixes

A republish of the code in 5.5.1 in an attempt to solve an issue with the file timestamps in the npm package being 0.

5.5.1 (2018-03-06)

Bug fixes

Fix misleading error message for octal escapes in template strings.

5.5.0 (2018-02-27)

New features

The identifier character categorization is now based on Unicode version 10.

Acorn will now validate the content of regular expressions, including new ES9 features.

5.4.0 (2018-02-01)

Bug fixes

Disallow duplicate or escaped flags on regular expressions.

Disallow octal escapes in strings in strict mode.

New features

Add support for async iteration.

Add support for object spread and rest.

5.3.0 (2017-12-28)

Bug fixes

Fix parsing of floating point literals with leading zeroes in loose mode.

Allow duplicate property names in object patterns.

Don't allow static class methods named `prototype`.

Disallow async functions directly under `if` or `else`.

Parse right-hand-side of `for`/`of` as an assignment expression.

Stricter parsing of `for`/`in`.

Don't allow unicode escapes in contextual keywords.

New features

Parsing class members was factored into smaller methods to allow plugins to hook into it.

5.2.1 (2017-10-30)

Bug fixes

Fix a token context corruption bug.

5.2.0 (2017-10-30)

Bug fixes

Fix token context tracking for `class` and `function` in property-name position.

Make sure `%*` isn't parsed as a valid operator.

Allow shorthand properties `get` and `set` to be followed by default values.

Disallow `super` when not in callee or object position.

New features

Support `directive` property on directive expression statements.

5.1.2 (2017-09-04)

Bug fixes

Disable parsing of legacy HTML-style comments in modules.

Fix parsing of async methods whose names are keywords.

5.1.1 (2017-07-06)

Bug fixes

Fix problem with disambiguating regexp and division after a class.

5.1.0 (2017-07-05)

Bug fixes

Fix tokenizing of regexps in an object-desctructuring `for`/`of` loop and after `yield`.

Parse zero-prefixed numbers with non-octal digits as decimal.

Allow object/array patterns in rest parameters.

Don't error when `yield` is used as a property name.

Allow `async` as a shorthand object property.

New features

Implement the template literal revision proposal for ES9.

5.0.3 (2017-04-01)

Bug fixes

Fix spurious duplicate variable definition errors for named functions.

5.0.2 (2017-03-30)

Bug fixes

A binary operator after a parenthesized arrow expression is no longer incorrectly treated as an error.

5.0.0 (2017-03-28)

Bug fixes

Raise an error for duplicated lexical bindings.

Fix spurious error when an assignement expression occurred after a spread expression.

Accept regular expressions after `of` (in `for/of`), `yield` (in a generator), and braced arrow functions.

Allow labels in front or `var` declarations, even in strict mode.

Breaking changes

Parse declarations following `export default` as declaration nodes, not expressions. This means that class and function declarations nodes can now have `null` as their `id`.

4.0.11 (2017-02-07)

Bug fixes

Allow all forms of member expressions to be parenthesized as lvalue.

4.0.10 (2017-02-07)

Bug fixes

Don't expect semicolons after default-exported functions or classes, even when they are expressions.

Check for use of `'use strict'` directives in non-simple parameter functions, even when already in strict mode.

4.0.9 (2017-02-06)

Bug fixes

Fix incorrect error raised for parenthesized simple assignment targets, so that `(x) = 1` parses again.

4.0.8 (2017-02-03)

Bug fixes

Solve spurious parenthesized pattern errors by temporarily erring on the side of accepting programs that our delayed errors don't handle correctly yet.

4.0.7 (2017-02-02)

Bug fixes

Accept invalidly rejected code like `(x).y = 2` again.

Don't raise an error when a function *inside* strict code has a non-simple parameter list.

4.0.6 (2017-02-02)

Bug fixes

Fix exponential behavior (manifesting itself as a complete hang for even relatively small source files) introduced by the new 'use strict' check.

4.0.5 (2017-02-02)

Bug fixes

Disallow parenthesized pattern expressions.

Allow keywords as export names.

Don't allow the `async` keyword to be parenthesized.

Properly raise an error when a keyword contains a character escape.

Allow `"use strict"` to appear after other string literal expressions.

Disallow labeled declarations.

4.0.4 (2016-12-19)

Bug fixes

Fix crash when `export` was followed by a keyword that can't be exported.

4.0.3 (2016-08-16)

Bug fixes

Allow regular function declarations inside single-statement `if` branches in loose mode. Forbid them entirely in strict mode.

Properly parse properties named `async` in ES2017 mode.

Fix bug where reserved words were broken in ES2017 mode.

4.0.2 (2016-08-11)

Bug fixes

Don't ignore period or 'e' characters after octal numbers.

Fix broken parsing for call expressions in default parameter values of arrow functions.

4.0.1 (2016-08-08)

Bug fixes

Fix false positives in duplicated export name errors.

4.0.0 (2016-08-07)

Breaking changes

The default `ecmaVersion` option value is now 7.

A number of internal method signatures changed, so plugins might need to be updated.

Bug fixes

The parser now raises errors on duplicated export names.

`arguments` and `eval` can now be used in shorthand properties.

Duplicate parameter names in non-simple argument lists now always produce an error.

New features

The `ecmaVersion` option now also accepts year-style version numbers (2015, etc).

Support for `async`/`await` syntax when `ecmaVersion` is >= 8.

Support for trailing commas in call expressions when `ecmaVersion` is >= 8.

3.3.0 (2016-07-25)

Bug fixes

Fix bug in tokenizing of regexp operator after a function declaration.

Fix parser crash when parsing an array pattern with a hole.

New features

Implement check against complex argument lists in functions that enable strict mode in ES7.

3.2.0 (2016-06-07)

Bug fixes

Improve handling of lack of unicode regexp support in host environment.

Properly reject shorthand properties whose name is a keyword.

New features

Visitors created with `visit.make` now have their base as *prototype*, rather than copying properties into a fresh object.

3.1.0 (2016-04-18)

Bug fixes

Properly tokenize the division operator directly after a function expression.

Allow trailing comma in destructuring arrays.

3.0.4 (2016-02-25)

Fixes

Allow update expressions as left-hand-side of the ES7 exponential operator.

3.0.2 (2016-02-10)

Fixes

Fix bug that accidentally made `undefined` a reserved word when parsing ES7.

3.0.0 (2016-02-10)

Breaking changes

The default value of the `ecmaVersion` option is now 6 (used to be 5).

Support for comprehension syntax (which was dropped from the draft spec) has been removed.

Fixes

`let` and `yield` are now "contextual keywords", meaning you can mostly use them as identifiers in ES5 non-strict code.

A parenthesized class or function expression after `export default` is now parsed correctly.

New features

When `ecmaVersion` is set to 7, Acorn will parse the exponentiation operator (`**`).

The identifier character ranges are now based on Unicode 8.0.0.

Plugins can now override the `raiseRecoverable` method to override the way non-critical errors are handled.

2.7.0 (2016-01-04)

Fixes

Stop allowing rest parameters in setters.

Disallow `y` rexexp flag in ES5.

Disallow `\00` and `\000` escapes in strict mode.

Raise an error when an import name is a reserved word.

2.6.2 (2015-11-10)

Fixes

Don't crash when no options object is passed.

2.6.0 (2015-11-09)

Fixes

Add `await` as a reserved word in module sources.

Disallow `yield` in a parameter default value for a generator.

Forbid using a comma after a rest pattern in an array destructuring.

New features

Support parsing stdin in command-line tool.

2.5.0 (2015-10-27)

Fixes

Fix tokenizer support in the command-line tool.

Stop allowing `new.target` outside of functions.

Remove legacy `guard` and `guardedHandler` properties from try nodes.

Stop allowing multiple `__proto__` properties on an object literal in strict mode.

Don't allow rest parameters to be non-identifier patterns.

Check for duplicate paramter names in arrow functions.