

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.

Don't crash when the loose parser is called without options object.

New features

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

Make it possible to use `visit.ancestor` with a walk state.

3.1.0 (2016-04-18)

Bug fixes

Fix issue where the loose parser created invalid TemplateElement nodes for unclosed template literals.

Properly tokenize the division operator directly after a function expression.

Allow trailing comma in destructuring arrays.

New features

The walker now allows defining handlers for `CatchClause` nodes.

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.

Make sure the loose parser always attaches a `local` property to `ImportNamespaceSpecifier` nodes.

Disallow `y` regexp flag in ES5.

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

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

2.6.4 (2015-11-12)

Fixes

Fix crash in loose parser when parsing invalid object pattern.

New features

Support plugins in the loose parser.

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.

2.5.2 (2015-10-27)

Fixes

Fix bug where the walker walked an exported `let` statement as an expression.

2.5.0 (2015-10-27)

Fixes

Fix tokenizer support in the command-line tool.

In the loose parser, don't allow non-string-literals as import sources.

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 parameter names in arrow functions.