

新特性

- 综合
 - 新增 `theme-chalk` 主题
 - 增强以下组件的可访问性: Alert、AutoComplete、Breadcrumb、Button、Checkbox、Collapse、Input、InputNumber、Menu、Progress、Radio、Rate、Slider、Switch 和 Upload
 - 新增布局组件 Container、Header、Aside、Main 和 Footer
 - 新增 TypeScript 类型声明
 - 重绘了全部图标，并新增了部分图标
 - 新增了一系列基于断点的工具类，用于当视口尺寸满足一定条件时隐藏元素
 - 新增全局配置组件尺寸的功能。在引入 Element 时，配置 `size` 字段可以改变所有组件的默认尺寸
- Button
 - 新增 `round` 属性，用于圆角按钮 #6643
- TimeSelect
 - 可以用 Up 、 Down 导航，用 Enter 选中时间 #6023
- TimePicker
 - 可以用方向键导航，用 Enter 选中时间 #6050
 - 新增 `start-placeholder` 和 `end-placeholder`，用于设置范围选择时两个输入框的占位符 #7169
 - 新增 `arrow-control` 属性，提供另一种交互形式，#7438
- Tree
 - 子节点在首次被展开之前不进行渲染 #6257
 - 新增 `check-descendants` 属性，设置 `lazy` 模式下勾选节点时，是否完全展开整个子树 #6235
- Tag
 - 新增 `size` 属性 #7203
- Datepicker
 - `type` 为 `datetimerange` 时可以使用 `timeFormat` 格式化时间选择器 #6052
 - 新增 `start-placeholder` 和 `end-placeholder`，用于设置范围选择时两个输入框的占位符 #7169
 - 新增 `value-format` 属性，支持对绑定值的格式进行自定义，#7367
 - 新增 `unlink-panels` 属性，用于在选择日期范围时取消两个日期面板之间的联动
- MessageBox
 - 新增 `closeOnHashChange` 属性 #6043
 - 新增 `center` 属性，提供居中布局 #7029
 - 新增 `roundButton` 属性，使得内部按钮为圆角按钮 #7029
 - 新增 `dangerouslyUseHTMLString` 属性，使得 `message` 支持传入 HTML 字符串* #6043
 - 新增 `inputType` 属性，用户指定内部输入框的类型，#7651
- Dialog
 - 新增 `width`、`fullscreen`、`append-to-body` 属性，支持嵌套使用
 - 新增 `center` 属性，提供居中布局 #7042
 - 新增 `focus-after-closed`、`focus-after-open` 属性，支持无障碍访问 #6511
- ColorPicker
 - 增加手动输入色值的支持 #6167
 - 新增 `size` 属性，用于控制组件的大小 #7026

- 新增 `disabled` 属性, 用于禁用组件 #7026
- 新增 `popper-class` 属性, #7351
- Message
 - 图标部分使用 `icon` 代替图片, 从而支持通过 CSS 修改图标背景色 #6207
 - 新增 `dangerouslyUseHTMLString` 属性, 使得 `message` 属性支持传入 HTML 字符串* #6207
 - 新增 `center` 属性, 提供居中布局 #6875
- Notification
 - 新增 `position` 属性, 用于配置 Notification 出现的位置 #6231
 - 新增 `dangerouslyUseHTMLString` 属性, 使得 `message` 属性支持传入 HTML 字符串* #6231
 - 新增 `showClose` 属性, 用于隐藏关闭按钮 #6402
- Rate
 - 新增 `show-score` 属性, 控制是否在右侧显示当前分数 #6295
- Tabs
 - 新增 `tab-position` 属性, 控制选项面板内容显示的上、下、左、右四个方向 #6096
- Radio
 - 增加 `border` 属性和 `size` 属性 #6690
- Checkbox
 - 增加 `border` 属性和 `size` 属性 #6690
- Alert
 - 新增 `center` 属性, 提供居中布局 #6876
- Menu
 - 新增 `background-color`、`text-color` 和 `active-text-color` 属性, 分别用于设置菜单的背景色、菜单的文字颜色和当前激活菜单的文字颜色 #7064
 - 新增 `open` 和 `close` 方法, 支持手动打开和关闭 SubMenu, #7412
- Form
 - 新增 `inline-message` 属性, 设置后校验信息会以行内样式显示 #7032
 - 新增 `status-icon` 属性, 用于在输入框中显示校验结果反馈图标 #7032
 - Form 和 FormItem 新增 `size` 属性, 用于控制表单内组件的尺寸, #7428
 - `validate` 方法在不传入 `callback` 的情况下返回 `promise`, #7405
 - 新增 `clearValidate` 方法, 用于清空所有表单项的验证信息, #7623
- Input
 - 新增 `suffix`、`prefix` 的 slot, 以及 `suffixIcon`、`prefixIcon` 属性, 用于给输入框内部增加前置和后置内容 #7032
- Breadcrumb
 - 新增 `separator-class` 属性, 可使用图标作为分隔符 #7203
- Steps
 - 新增 `simple` 属性, 用于开启简洁风格的步骤条 #7274
- Pagination
 - 新增 `prev-text` 和 `next-text` 属性, 用于自定义上一页和下一页的文本 #7005
- Loading
 - 配置对象新增 `spinner` 和 `background` 字段, 支持自定义加载图标和背景色, #7390
- Autocomplete
 - 新增 `debounce` 属性, #7413

- Upload
 - 新增 `limit` 和 `on-exceed` 属性，支持对上传文件的个数进行限制，#7405
- DateTimePicker
 - 新增 `time-arrow-control` 属性，用于开启时间选择器的 `arrow-control`，#7438
- Layout
 - 新增断点 `xl`，适用于宽度大于 1920px 的视口
- Table
 - 新增 `span-method` 属性，用于合并行或列
 - 新增 `clearSort` 方法，用于清空排序状态
 - 新增 `clearFilter` 方法，用于清空过滤状态
 - 对于可展开行，当该行展开时会获得一个 `.expanded` 类名，方便自定义样式
 - 新增 `size` 属性，用于控制表格尺寸
 - 新增 `toggleRowExpansion` 方法，用于手动展开或关闭行
 - 新增 `cell-class-name` 属性，用于指定单元格的类名
 - 新增 `cell-style` 属性，用于指定单元格的样式
 - 新增 `header-row-class-name` 属性，用于指定表头行的类名
 - 新增 `header-row-style` 属性，用于指定表头行的样式
 - 新增 `header-cell-class-name` 属性，用于指定表头单元格的类名
 - 新增 `header-cell-style` 属性，用于指定表头单元格的样式
 - TableColumn 的 `prop` 属性支持 `object[key]` 格式
 - TableColumn 新增 `index` 属性，用于自定义索引值
- Select
 - 新增 `reserve-keyword` 属性，用于在选择某个选项后保留当前的搜索关键词

修复

- DatePicker
 - 选择周数时，`v-model` 结果返回该周第二天的问题 #6038
 - 在 `daterange` 类型中，第一次的输入会被清空的问题 #6021
- DateTimePicker
 - 和 TimePicker 相互影响的问题 #6090
 - 选择时间小时和秒可超出限制的问题 #6076
- TimePicker
 - 失去焦点时无法正确改变 `v-model` 值的问题 #6023
- Dialog
 - 当含有下拉框时，下拉框的打开和关闭会造成文字虚晃的问题 #6088
- Select
 - 提升性能，修复组件销毁时可能导致 Vue dev-tool 卡死的问题 #6151
- Table
 - 修复 Table 在父元素从 `display: none` 变成其他状态时会隐藏的问题
 - 修复 Table 在父元素为 `display: flex` 时可能出现的宽度逐渐变大的问题
 - 修复 `append` 具名 slot 和固定列并存时，动态获取表格数据会导致固定列消失的问题
 - 修复 `expand-row-keys` 属性初始化无效的问题
 - 修复 `data` 改变时过滤条件失效的问题
 - 修复多级表头时固定列隐藏情况计算错误的问题

- 修复 `max-height` 变更后无法恢复的问题
- 修复一些样式上的计算错误

非兼容性更新

- 综合
 - 移除 `theme-default`
 - 最低兼容 Vue 2.5.2 和 IE 10
 - 表单组件的 `change` 事件和 Pagination 的 `current-change` 事件现在仅响应用户交互
 - Button 和表单组件的 `size` 属性不再接受 `large` 值, 可接受 `medium`、`small` 和 `mini`
 - 为了方便使用第三方图标, Button 的 `icon` 属性、Input 的 `prefix-icon` 和 `suffix-icon` 属性、Steps 的 `icon` 属性现在需要传入完整的图标类名
- Dialog
 - 移除 `size` 属性。现在 Dialog 的尺寸由 `width` 和 `fullscreen` 控制
 - 移除通过 `v-model` 控制 Dialog 显示和隐藏的功能
- Rate
 - `text-template` 属性更名为 `score-template`
- Dropdown
 - `menu-align` 属性变更为 `placement`, 增加更多方位属性
- Transfer
 - `footer-format` 属性更名为 `format`
- Switch
 - 由于 `on-*` 属性在 JSX 中会被识别为事件, 导致 Switch 所有 `on-*` 属性在 JSX 中无法正常工作, 所以 `on-*` 属性更名为 `active-*`, 对应地, `off-*` 属性更名为 `inactive-*`。受到影响的属性有: `on-icon-class`、`off-icon-class`、`on-text`、`off-text`、`on-color`、`off-color`、`on-value`、`off-value`
 - `active-text` 和 `inactive-text` 属性不再有默认值
- Tag
 - `type` 属性现在支持 `success`、`info`、`warning` 和 `danger` 四个值
- Menu
 - 移除 `theme` 属性。现在通过 `background-color`、`text-color` 和 `active-text-color` 属性进行颜色的自定义
- Input
 - 移除 `icon` 属性。现在通过 `suffix-icon` 属性或者 `suffix` 具名 slot 来加入尾部图标
 - 移除 `on-icon-click` 属性和 `click` 事件。现在如果需要为输入框中的图标添加点击事件, 请以具名 slot 的方式添加图标
 - `change` 事件现在仅在输入框失去焦点或用户按下回车时触发, 与原生 input 元素一致。如果需要实时响应用户的输入, 可以使用 `input` 事件
- Autocomplete
 - 移除 `custom-item` 属性。现在通过 `scoped slot` 自定义输入建议列表项的内容
 - 移除 `props` 属性, 现在使用 `value-key` 属性指定输入建议对象中用于显示的键名
- Steps
 - 移除 `center` 属性
 - 现在步骤条将默认充满父容器
 - DatePicker

- `change` 事件参数现在为组件的绑定值，格式由 `value-format` 控制
- Table
 - 移除通过 `inline-template` 自定义列模板的功能
 - `sort-method` 现在和 `Array.sort` 保持一致的逻辑，要求返回一个数字
 - 将 `append slot` 移至 `tbody` 元素以外，以保证其只被渲染一次
 - `expand` 事件更名为 `expand-change`，以保证 API 的命名一致性
 - `row-class-name` 和 `row-style` 的函数参数改为对象，以保证 API 的一致性

* 在网站上动态渲染任意 HTML 是非常危险的，因为容易导致 [XSS 攻击](#)。因此请在 `dangerouslyUseHTMLString` 打开的情况下，确保 `message` 的内容是可信的，**永远不要**将用户提交的内容赋值给 `message` 属性。

New features

- General
 - A new theme: `theme-chalk`
 - Accessibility of the following components are improved: Alert, AutoComplete, Breadcrumb, Button, Checkbox, Collapse, Input, InputNumber, Menu, Progress, Radio, Rate, Slider, Switch, Upload
 - Added TypeScript typings
 - All existing icons are redesigned. Some new icons are added
 - Added a series of breakpoint-based utility classes that hide elements when the viewport size meets certain conditions
 - Added layout components: Container, Header, Aside, Main, Footer
 - Now you can configure component sizes globally. When importing Element, you can add a global config object with a `size` prop to configure default sizes for all components.
- Button
 - Added `round` attribute. It's used for round-cornered Buttons #6643
- TimeSelect
 - Now can be navigated by `Up` and `Down`, and hitting `Enter` selects the time #6023
- TimePicker
 - Now can be navigated by arrow keys, and hitting `Enter` selects the time #6050
 - Added `start-placeholder` and `end-placeholder`. They're placeholders for the two input boxes in range mode #7169
 - Added `arrow-control` attribute to spin the time with arrows #7438
- Tree
 - Now child nodes don't render before the first expand #6257
 - Added `check-descendants` attribute. It determines if child nodes are checked when checking their parent node in `lazy` mode #6235
- Tag
 - Added `size` attribute #7203
- DatePicker
 - Now `timeFormat` can format the TimePicker when type is set to `datetimerange` #6052
 - Added `start-placeholder` and `end-placeholder`. They're placeholders for the two input boxes in range mode #7169
 - Added `value-format` attribute to customize the format of the binding value, #7367
 - Added `unlink-panels` attribute to unlink the two date panels when selecting a date range

- **MessageBox**
 - Added `closeOnHashChange` attribute #6043
 - Added `center` attribute so that the content can be centered #7029
 - Added `roundButton` attribute to display round Buttons #7029
 - Added `dangerouslyUseHTMLString` attribute. When set to `true`, `message` will be parsed as HTML string* #6043
 - Added `inputType` attribute to assign type for the inner input box, #7651
- **Dialog**
 - Added `width`, `fullscreen`, `append-to-body` attributes. Now Dialog can be nested
 - Added `center` attribute so that the content can be centered #7042
 - Added `focus-after-closed`, `focus-after-open` to improve accessibility #6511
- **ColorPicker**
 - Now you can type colors in the input box #6167
 - Added `size` and `disabled` attributes #7026
 - Added `popper-class` attribute #7351
- **Message**
 - Now color of the icons can be overridden by CSS #6207
 - Added `dangerouslyUseHTMLString` attribute. When set to `true`, `message` will be parsed as HTML string* #6207
 - Added `center` attribute so that the content can be centered #6875
- **Notification**
 - Added `position` attribute to configure where Notification pops up #6231
 - Added `dangerouslyUseHTMLString` attribute. When set to `true`, `message` will be parsed as HTML string* #6231
 - Added `showClose` attribute to hide the close button #6402
- **Rate**
 - Added `show-score` attribute to determine if current score is displayed #6295
- **Tabs**
 - Added `tab-position` attribute #6096
- **Radio**
 - Added `border` and `size` attributes #6690
- **Checkbox**
 - Added `border` and `size` attributes #6690
- **Alert**
 - Added `center` attribute so that the content can be centered #6876
- **Menu**
 - Added `background-color`, `text-color` and `active-text-color` attributes #7064
 - Added `open` and `close` methods to open and close SubMenu programmatically, #7412
- **Form**
 - Added `inline-message` attribute to determine if the validation message is displayed in inline style #7032
 - Added `status-icon` attribute to display a feedback icon when validated #7032
 - Form and FormItem now have a `size` attribute. Inner components will inherit this size if not specified on themselves, #7428

- `validate` method will now return a promise if the callback is omitted, #7405
- Added `clearValidate` method for clearing validating results for all form items, #7623
- Input
 - Added `suffix` and `prefix` named slots, `suffixIcon` and `prefixIcon` attributes to add contents inside the input box #7032
- Breadcrumb
 - Added `separator-class` attribute to support icons as item separators #7203
- Steps
 - Added `simple` attribute to activate simple-styled Steps #7274
- Pagination
 - Added `prev-text` and `next-text` attributes to customize texts of previous page and next page #7005
- Loading
 - Now you can customize spinner icon and background color with `spinner` and `background` prop, #7390
- Autocomplete
 - Added `debounce` attribute, #7413
- Upload
 - Added `limit` and `on-exceed` attributes to limit the amount of files, #7405
- DateTimePicker
 - Added `time-arrow-control` attribute to activate `arrow-control` of the nesting TimePicker, #7438
- Layout
 - Added a new breakpoint `xl` for viewport wider than 1920px
- Table
 - Added `span-method` attribute for merging cells
 - Added `clearSort` method to clear sorting programmatically
 - Added `clearFilter` method to clear filter programmatically
 - For expandable rows, when a row is expanded, a `.expanded` class will be added to its class list, so that you can customize its style
 - Added `size` attribute
 - Added `toggleRowExpansion` method to expand or collapse expandable rows programmatically
 - Added `cell-class-name` attribute to assign class name for cells
 - Added `cell-style` attribute to style cells
 - Added `header-row-class-name` attribute to assign class name for header rows
 - Added `header-row-style` attribute to style header rows
 - Added `header-cell-class-name` attribute to assign class name for header cells
 - Added `header-cell-style` attribute to style header cells
 - TableColumn's `prop` attribute now accepts `object[key]` notations
 - Added `index` attribute for TableColumn to customize row indices
- Select
 - Added `reserve-keyword` attribute for reserving current search keyword after selecting an option

Bug fixes

- DatePicker
 - Fixed `v-model` returning the second day of the selected week in week mode #6038
 - Fixed the first input being cleared in `daterange` type #6021
- DateTimePicker
 - Fixed DateTimePicker and TimePicker affecting each other when picked #6090
 - Fixed hour and second can be beyond limit when selecting time #6076
- TimePicker
 - Fixed `v-model` not update correctly when blurred #6023
- Dialog
 - Fixed texts having blurry edges when opening and closing nesting dropdowns #6088
- Select
 - Improved performance. Now Vue dev-tool won't crash when a large number of Selects are destroyed #6151
- Table
 - Fixed a bug that Table remains hiding when its parent element appears from `display: none`
 - Fixed Table expanding its width when its parent element has `display: flex`
 - Fixed a bug that fixed columns of a Table with `append` slot would disappear when data is dynamically fetched
 - Fixed `expand-row-keys` attribute not working with initial value
 - Fixed filter failing when `data` updates
 - Fixed a calculation error of fixed columns layout with grouped headers
 - Fixed a dynamic `max-height` bug
 - Fixed some style calculation errors

Breaking changes

- General
 - Removed `theme-default`
 - Compatible with Vue 2.5.2+ and IE 10+
 - `change` event of form components and `current-change` event of Pagination now only trigger on user interaction
 - `size` attribute of Button and form components now accept `medium`, `small` and `mini`
 - To facilitate the use of third-party icons, `icon` attribute of Button and Steps, `prefix-icon` and `suffix-icon` attributes of Input now require a full class name
- Dialog
 - Removed `size` attribute. Now the size of Dialog can be configured by `width` and `fullscreen`
 - Now the visibility of Dialog cannot be controlled by `v-model`
- Rate
 - `text-template` is renamed to `score-template`
- Dropdown
 - `menu-align` is renamed to `placement`. Now it supports more positions
- Transfer
 - `footer-format` is renamed to `format`

- Switch
 - Attributes starting with `on-*` will be parsed to events in JSX, making all `on-*` attributes of Switch not able to work in JSX. So `on-*` attributes are renamed to `active-*`, and accordingly `off-*` attributes are renamed to `inactive-*`. This change affects the following attributes: `on-icon-class`, `off-icon-class`, `on-text`, `off-text`, `on-color`, `off-color`, `on-value`, `off-value`
 - `active-text` and `inactive-text` attributes now don't have default values
- Tag
 - `type` attribute now accepts `success`, `info`, `warning` and `danger`
- Menu
 - Removed `theme` attribute. The color of Menu can be configured using `background-color`, `text-color` and `active-text-color`
- Input
 - Removed `icon` attribute. Now the suffix icon can be configured using `suffix-icon` attribute or `suffix` named slot
 - Removed `on-icon-click` attribute and `click` event. Now to add click handler on icons, please use named slots
 - `change` event now behaves like the native input element, which triggers only on blur or pressing enter. If you need to respond to user input in real time, you can use `input` event.
- Autocomplete
 - Removed `custom-item` attribute. Now the template of input suggestions can be customized using `scoped slot`
 - Removed `props` attribute. Now you can use `value-key` attribute to designate key name of the input suggestion object for display
- Steps
 - Removed `center` attribute
 - Now the Steps will fill its parent container by default
- DatePicker
 - The params of DatePicker's `change` event is now the binding value itself. Its format is controlled by `value-format`
- Table
 - Removed support for customizing column template using `inline-template`
 - `sort-method` now aligns with `Array.sort`. It should return a number instead of a boolean
 - `append` slot is moved outside the `tbody` element to avoid multiple rendering
 - `expand` event is renamed to `expand-change`
 - The params of `row-class-name` and `row-style` method is now an object

* Dynamically rendering arbitrary HTML on your website can be very dangerous because it can easily lead to [XSS attacks](#). So when `dangerouslyUseHTMLString` is on, please make sure the content of `message` is trusted, and **never** assign `message` to user-provided content.