

Cascader

Si les options ont une structure hiérarchique claire, Le composant Cascader peut être utilisé pour les afficher et les sélectionner.

Basic usage

Il y a deux manières d'étendre les options enfants.

:::demo Assigner l'attribut `options` à un tableau d'options fournit un composant Cascader. L'attribut `props.expandTrigger` définit comment les options enfants sont étendues.

```
<div class="block">
  <span class="demonstration">Les options enfants se développent lorsque vous
cliquez dessus (par défaut)</span>
  <el-cascader
    v-model="value"
    :options="options"
    @change="handleChange"></el-cascader>
</div>
<div class="block">
  <span class="demonstration">Les options enfants se développent lorsqu'elles sont
survolées</span>
  <el-cascader
    v-model="value"
    :options="options"
    :props="{ expandTrigger: 'hover' }"
    @change="handleChange"></el-cascader>
</div>

<script>
export default {
  data() {
    return {
      value: [],
      options: [{
        value: 'guide',
        label: 'Guide',
        children: [{
          value: 'disciplines',
          label: 'Disciplines',
          children: [{
            value: 'consistency',
            label: 'Consistency'
          }, {
            value: 'feedback',
            label: 'Feedback'
          }, {
            value: 'efficiency',
            label: 'Efficiency'
          }, {
            value: 'controllability',
```

```
      label: 'Controllability'
    }]
  }, {
    value: 'navigation',
    label: 'Navigation',
    children: [{
      value: 'side nav',
      label: 'Side Navigation'
    }, {
      value: 'top nav',
      label: 'Top Navigation'
    }]
  }]
}, {
  value: 'component',
  label: 'Component',
  children: [{
    value: 'basic',
    label: 'Basic',
    children: [{
      value: 'layout',
      label: 'Layout'
    }, {
      value: 'color',
      label: 'Color'
    }, {
      value: 'typography',
      label: 'Typography'
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }]
  }, {
    value: 'form',
    label: 'Form',
    children: [{
      value: 'radio',
      label: 'Radio'
    }, {
      value: 'checkbox',
      label: 'Checkbox'
    }, {
      value: 'input',
      label: 'Input'
    }, {
      value: 'input-number',
      label: 'InputNumber'
    }, {
      value: 'select',
```

```
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }
]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
}, {
```

```
value: 'notice',
label: 'Notice',
children: [{
  value: 'alert',
  label: 'Alert'
}, {
  value: 'loading',
  label: 'Loading'
}, {
  value: 'message',
  label: 'Message'
}, {
  value: 'message-box',
  label: 'MessageBox'
}, {
  value: 'notification',
  label: 'Notification'
}]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
    label: 'Steps'
  }]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }
}
```

```

      }, {
        value: 'carousel',
        label: 'Carousel'
      }, {
        value: 'collapse',
        label: 'Collapse'
      }
    ]
  ]
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}]
};
},
methods: {
  handleChange(value) {
    console.log(value);
  }
}
};
</script>

```

...

Disabled option

Désactivez une option en définissant un champ `disabled` dans l'objet d'option.

:::demo Dans cet exemple, le premier élément du tableau `options` a un champ `disabled: true`, il est donc désactivé. Par défaut, Cascader vérifie le champ `disabled` dans chaque objet option. Si vous utilisez un autre nom de champ pour indiquer si une option est désactivée, vous pouvez l'affecter dans l'attribut `props.disabled` (voir le tableau des API ci-dessous pour plus de détails). Et bien sûr, les noms de champs `value`, `label` et `children` peuvent également être personnalisés de la même manière.

```

<el-cascader :options="options"></el-cascader>

<script>
  export default {
    data() {
      return {
        options: [{

```

```
value: 'guide',
label: 'Guide',
disabled: true,
children: [{
  value: 'disciplines',
  label: 'Disciplines',
  children: [{
    value: 'consistency',
    label: 'Consistency'
  }, {
    value: 'feedback',
    label: 'Feedback'
  }, {
    value: 'efficiency',
    label: 'Efficiency'
  }, {
    value: 'controllability',
    label: 'Controllability'
  }]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'side nav',
    label: 'Side Navigation'
  }, {
    value: 'top nav',
    label: 'Top Navigation'
  }]
}]
}, {
  value: 'component',
  label: 'Component',
  children: [{
    value: 'basic',
    label: 'Basic',
    children: [{
      value: 'layout',
      label: 'Layout'
    }, {
      value: 'color',
      label: 'Color'
    }, {
      value: 'typography',
      label: 'Typography'
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }]
  }]
}]
```

```
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }]
```

```
    }, {
      value: 'tag',
      label: 'Tag'
    }, {
      value: 'progress',
      label: 'Progress'
    }, {
      value: 'tree',
      label: 'Tree'
    }, {
      value: 'pagination',
      label: 'Pagination'
    }, {
      value: 'badge',
      label: 'Badge'
    }
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
```



```

        label: 'Steps'
      })
    }, {
      value: 'others',
      label: 'Others',
      children: [{
        value: 'dialog',
        label: 'Dialog'
      }, {
        value: 'tooltip',
        label: 'Tooltip'
      }, {
        value: 'popover',
        label: 'Popover'
      }, {
        value: 'card',
        label: 'Card'
      }, {
        value: 'carousel',
        label: 'Carousel'
      }, {
        value: 'collapse',
        label: 'Collapse'
      }]
    })
  ], {
    value: 'resource',
    label: 'Resource',
    children: [{
      value: 'axure',
      label: 'Axure Components'
    }, {
      value: 'sketch',
      label: 'Sketch Templates'
    }, {
      value: 'docs',
      label: 'Design Documentation'
    }]
  })
}
};
</script>

```

...

Clearable

Définissez l'attribut `clearable` pour `el-cascader` et une icône claire apparaîtra une fois sélectionnée et survolée

...demo

```
<el-cascader :options="options" clearable></el-cascader>
```

```
<script>
```

```
  export default {
```

```
    data() {
```

```
      return {
```

```
        options: [{
```

```
          value: 'guide',
```

```
          label: 'Guide',
```

```
          children: [{
```

```
            value: 'disciplines',
```

```
            label: 'Disciplines',
```

```
            children: [{
```

```
              value: 'consistency',
```

```
              label: 'Consistency'
```

```
            }, {
```

```
              value: 'feedback',
```

```
              label: 'Feedback'
```

```
            }, {
```

```
              value: 'efficiency',
```

```
              label: 'Efficiency'
```

```
            }, {
```

```
              value: 'controllability',
```

```
              label: 'Controllability'
```

```
          ]]
```

```
        }, {
```

```
          value: 'navigation',
```

```
          label: 'Navigation',
```

```
          children: [{
```

```
            value: 'side nav',
```

```
            label: 'Side Navigation'
```

```
          }, {
```

```
            value: 'top nav',
```

```
            label: 'Top Navigation'
```

```
          ]]
```

```
        ]]
```

```
      }, {
```

```
        value: 'component',
```

```
        label: 'Component',
```

```
        children: [{
```

```
          value: 'basic',
```

```
          label: 'Basic',
```

```
          children: [{
```

```
            value: 'layout',
```

```
            label: 'Layout'
```

```
          }, {
```

```
            value: 'color',
```

```
            label: 'Color'
```

```
          }, {
```

```
            value: 'typography',
```

```
            label: 'Typography'
```

```
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }
  ]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }
}
```

```
    }}
  }, {
    value: 'data',
    label: 'Data',
    children: [{
      value: 'table',
      label: 'Table'
    }, {
      value: 'tag',
      label: 'Tag'
    }, {
      value: 'progress',
      label: 'Progress'
    }, {
      value: 'tree',
      label: 'Tree'
    }, {
      value: 'pagination',
      label: 'Pagination'
    }, {
      value: 'badge',
      label: 'Badge'
    }
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {

```

```

        value: 'breadcrumb',
        label: 'Breadcrumb'
    }, {
        value: 'dropdown',
        label: 'Dropdown'
    }, {
        value: 'steps',
        label: 'Steps'
    }
  ]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }, {
    value: 'carousel',
    label: 'Carousel'
  }, {
    value: 'collapse',
    label: 'Collapse'
  }
  ]
}
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
  ]
}
}
}
</script>

```

Display only the last level

L'entrée peut afficher uniquement le dernier niveau au lieu de tous les niveaux.

:::demo L'attribut `show-all-levels` définit si tous les niveaux sont affichés. S'il est `false`, seul le dernier niveau est affiché.

```
<el-cascader :options="options" :show-all-levels="false"></el-cascader>
<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {
              value: 'controllability',
              label: 'Controllability'
            }
          ]
        }, {
          value: 'navigation',
          label: 'Navigation',
          children: [{
            value: 'side nav',
            label: 'Side Navigation'
          }, {
            value: 'top nav',
            label: 'Top Navigation'
          }
        ]
      }
    ], {
      value: 'component',
      label: 'Component',
      children: [{
        value: 'basic',
        label: 'Basic',
        children: [{
          value: 'layout',
          label: 'Layout'
        }
      ], {
```

```
    value: 'color',
    label: 'Color'
  }, {
    value: 'typography',
    label: 'Typography'
  }, {
    value: 'icon',
    label: 'Icon'
  }, {
    value: 'button',
    label: 'Button'
  }
]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
```

```
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  ]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  ]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
```



```

        label: 'NavMenu'
    }, {
        value: 'tabs',
        label: 'Tabs'
    }, {
        value: 'breadcrumb',
        label: 'Breadcrumb'
    }, {
        value: 'dropdown',
        label: 'Dropdown'
    }, {
        value: 'steps',
        label: 'Steps'
    }
  ]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }, {
    value: 'carousel',
    label: 'Carousel'
  }, {
    value: 'collapse',
    label: 'Collapse'
  }
  ]
}
], {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
  ]
}
];

```

```
    }  
  };  
</script>
```

...

Multiple Selection

Définissez `props.multiple = true` pour utiliser la sélection multiple.

:::demo Lors de l'utilisation de la sélection multiple, toutes les balises sélectionnées seront affichées par défaut. Vous pouvez définir `collapse-tags = true` pour replier les balises sélectionnées.

```
<div class="block">  
  <span class="demonstration">Afficher toutes les balises (par défaut)</span>  
  <el-cascader  
    :options="options"  
    :props="props"  
    clearable></el-cascader>  
</div>  
<div class="block">  
  <span class="demonstration">Réduire les balises</span>  
  <el-cascader  
    :options="options"  
    :props="props"  
    collapse-tags  
    clearable></el-cascader>  
</div>  
  
<script>  
  export default {  
    data() {  
      return {  
        props: { multiple: true },  
        options: [{  
          value: 1,  
          label: 'Asia',  
          children: [{  
            value: 2,  
            label: 'China',  
            children: [  
              { value: 3, label: 'Beijing' },  
              { value: 4, label: 'Shanghai' },  
              { value: 5, label: 'Hangzhou' }  
            ]  
          }, {  
            value: 6,  
            label: 'Japan',  
            children: [  
              { value: 7, label: 'Tokyo' },  
              { value: 8, label: 'Osaka' },  
              { value: 9, label: 'Kyoto' }  
            ]  
          }  
        ]  
      }  
    }  
  }  
</script>
```

```

    ]
  }, {
    value: 10,
    label: 'Korea',
    children: [
      { value: 11, label: 'Seoul' },
      { value: 12, label: 'Busan' },
      { value: 13, label: 'Taegu' }
    ]
  }]
}, {
  value: 14,
  label: 'Europe',
  children: [{
    value: 15,
    label: 'France',
    children: [
      { value: 16, label: 'Paris' },
      { value: 17, label: 'Marseille' },
      { value: 18, label: 'Lyon' }
    ]
  }], {
    value: 19,
    label: 'UK',
    children: [
      { value: 20, label: 'London' },
      { value: 21, label: 'Birmingham' },
      { value: 22, label: 'Manchester' }
    ]
  }]
}, {
  value: 23,
  label: 'North America',
  children: [{
    value: 24,
    label: 'US',
    children: [
      { value: 25, label: 'New York' },
      { value: 26, label: 'Los Angeles' },
      { value: 27, label: 'Washington' }
    ]
  }], {
    value: 28,
    label: 'Canada',
    children: [
      { value: 29, label: 'Toronto' },
      { value: 30, label: 'Montreal' },
      { value: 31, label: 'Ottawa' }
    ]
  }]
}]
};

```

```

    }
  };
</script>

```

...

Select any level of options

Dans la sélection simple, seuls les noeuds terminaux peuvent être coché, et dans la sélection multiple, coché les noeuds parents cochera les noeuds terminaux. Lorsque cette fonctionnalité est activée, les noeuds parents et enfants peuvent être dissociés et vous pouvez sélectionner n'importe quel niveau d'options.

:::demo Définissez `props.checkStrictly = true` pour que l'état vérifié d'un noeud n'affecte pas ses noeuds parents et ses noeuds enfants. Vous pouvez ensuite sélectionner n'importe quel niveau d'options.

```

<div class="block">
  <span class="demonstration">Sélectionnez n'importe quel niveau d'options
  (Sélection unique)</span>
  <el-cascader
    :options="options"
    :props="{ checkStrictly: true }"
    clearable></el-cascader>
</div>

<div class="block">
  <span class="demonstration">Sélectionnez n'importe quel niveau d'options
  (sélection multiple)</span>
  <el-cascader
    :options="options"
    :props="{ multiple: true, checkStrictly: true }"
    clearable></el-cascader>
</div>

<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {

```

```
        value: 'controllability',
        label: 'Controllability'
    ]
}, {
    value: 'navigation',
    label: 'Navigation',
    children: [{
        value: 'side nav',
        label: 'Side Navigation'
    }, {
        value: 'top nav',
        label: 'Top Navigation'
    }]
}]
}, {
    value: 'component',
    label: 'Component',
    children: [{
        value: 'basic',
        label: 'Basic',
        children: [{
            value: 'layout',
            label: 'Layout'
        }, {
            value: 'color',
            label: 'Color'
        }, {
            value: 'typography',
            label: 'Typography'
        }, {
            value: 'icon',
            label: 'Icon'
        }, {
            value: 'button',
            label: 'Button'
        }]
    }, {
        value: 'form',
        label: 'Form',
        children: [{
            value: 'radio',
            label: 'Radio'
        }, {
            value: 'checkbox',
            label: 'Checkbox'
        }, {
            value: 'input',
            label: 'Input'
        }, {
            value: 'input-number',
            label: 'InputNumber'
        }, {
```

```
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }
]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
```

```
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
    label: 'Steps'
  }]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
```

```

        label: 'Card'
      }, {
        value: 'carousel',
        label: 'Carousel'
      }, {
        value: 'collapse',
        label: 'Collapse'
      }
    ]
  ]
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}]
};
}
};
</script>

```

...

Dynamic loading

Charge dynamiquement ses noeuds enfants lorsque un noeud est sélectionné.

...demo Définissez `lazy = true` pour utiliser le chargement dynamique. Vous devez Spécifier comment charger la source de données avec `lazyload`. Il existe deux paramètres de `lazyload`, le premier paramètre `node` est le noeud sur lequel l'utilisateur a cliqué, et `resolution` est un callback indiquant que le chargement est terminé et qu'il doit être appelé. Pour afficher le statut du noeud plus précisément, vous pouvez ajouter un champ `leaf` (modifiable par `props.leaf`) pour indiquer s'il s'agit d'un noeud feuille. Sinon, il sera déduit par ses noeuds enfants.

```

<el-cascader :props="props"></el-cascader>

<script>
  let id = 0;

  export default {
    data() {
      return {
        props: {
          lazy: true,

```



```

    lazyLoad (node, resolve) {
      const { level } = node;
      setTimeout(() => {
        const nodes = Array.from({ length: level + 1 })
          .map(item => ({
            value: ++id,
            label: `Option - ${id}`,
            leaf: level >= 2
          }));
        // Appelez le callback `resolve` pour renvoyer les données des noeuds
        // enfants et indiquer que le chargement est terminé.
        resolve(nodes);
      }, 1000);
    }
  }
};
}
};
</script>

```

...

Filterable

Rechercher et sélectionner des options avec un mot clé.

...demo Ajouter `filterable` à `el-cascader` permet le filtrage. Cascader recherchera les noeuds dont l'étiquette ou l'étiquette du parent (selon `show-all-levels`) inclut le mot-clé input. Bien sûr, vous pouvez personnaliser la logique de recherche en utilisant `filter-method` qui accepte une fonction, le premier paramètre est `node`, le second est `keyword` et doit renvoyer une valeur booléenne indiquant si le résultat est positif.

```

<div class="block">
  <span class="demonstration">Filterable (Sélection unique)</span>
  <el-cascader
    placeholder="Try searchingL Guide"
    :options="options"
    filterable></el-cascader>
</div>

<div class="block">
  <span class="demonstration">Filterable (Sélection multiple)</span>
  <el-cascader
    placeholder="Try searchingL Guide"
    :options="options"
    :props="{ multiple: true }"
    filterable></el-cascader>
</div>

<script>
  export default {
    data() {
      return {
        options: [{

```

```
value: 'guide',
label: 'Guide',
children: [{
  value: 'disciplines',
  label: 'Disciplines',
  children: [{
    value: 'consistency',
    label: 'Consistency'
  }, {
    value: 'feedback',
    label: 'Feedback'
  }, {
    value: 'efficiency',
    label: 'Efficiency'
  }, {
    value: 'controllability',
    label: 'Controllability'
  }]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'side nav',
    label: 'Side Navigation'
  }, {
    value: 'top nav',
    label: 'Top Navigation'
  }]
}]
}, {
  value: 'component',
  label: 'Component',
  children: [{
    value: 'basic',
    label: 'Basic',
    children: [{
      value: 'layout',
      label: 'Layout'
    }, {
      value: 'color',
      label: 'Color'
    }, {
      value: 'typography',
      label: 'Typography'
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }]
  }, {
    value: 'button',
    label: 'Button'
  }]
}, {
```

```
value: 'form',
label: 'Form',
children: [{
  value: 'radio',
  label: 'Radio'
}, {
  value: 'checkbox',
  label: 'Checkbox'
}, {
  value: 'input',
  label: 'Input'
}, {
  value: 'input-number',
  label: 'InputNumber'
}, {
  value: 'select',
  label: 'Select'
}, {
  value: 'cascader',
  label: 'Cascader'
}, {
  value: 'switch',
  label: 'Switch'
}, {
  value: 'slider',
  label: 'Slider'
}, {
  value: 'time-picker',
  label: 'TimePicker'
}, {
  value: 'date-picker',
  label: 'DatePicker'
}, {
  value: 'datetime-picker',
  label: 'DateTimePicker'
}, {
  value: 'upload',
  label: 'Upload'
}, {
  value: 'rate',
  label: 'Rate'
}, {
  value: 'form',
  label: 'Form'
}]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
```

```
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
    label: 'Steps'
  }
]
```

```

    }}
  }, {
    value: 'others',
    label: 'Others',
    children: [{
      value: 'dialog',
      label: 'Dialog'
    }, {
      value: 'tooltip',
      label: 'Tooltip'
    }, {
      value: 'popover',
      label: 'Popover'
    }, {
      value: 'card',
      label: 'Card'
    }, {
      value: 'carousel',
      label: 'Carousel'
    }, {
      value: 'collapse',
      label: 'Collapse'
    }]
  }]
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }]
}]
};
}
};
</script>

```

...

Custom option content

Vous pouvez personnaliser le contenu du noeud cascader.

:::demo Vous pouvez personnaliser le contenu du noeud cascader avec `scoped slot`. Vous aurez accès à `node` et `data` dans la portée, correspondant respectivement à l'objet Node et aux données de noeud du noeud actuel.

```

<el-cascader :options="options">
  <template slot-scope="{ node, data }">
    <span>{{ data.label }}</span>
    <span v-if="!node.isLeaf"> ({{ data.children.length }}) </span>
  </template>
</el-cascader>

<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {
              value: 'controllability',
              label: 'Controllability'
            }
          ]
        }, {
          value: 'navigation',
          label: 'Navigation',
          children: [{
            value: 'side nav',
            label: 'Side Navigation'
          }, {
            value: 'top nav',
            label: 'Top Navigation'
          }
        ]
      }
    ], {
      value: 'component',
      label: 'Component',
      children: [{
        value: 'basic',
        label: 'Basic',
        children: [{
          value: 'layout',
          label: 'Layout'
        }, {

```

```
    value: 'color',
    label: 'Color'
  }, {
    value: 'typography',
    label: 'Typography'
  }, {
    value: 'icon',
    label: 'Icon'
  }, {
    value: 'button',
    label: 'Button'
  }
]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
```

```
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  ]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  ]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
```



```

        label: 'NavMenu'
      }, {
        value: 'tabs',
        label: 'Tabs'
      }, {
        value: 'breadcrumb',
        label: 'Breadcrumb'
      }, {
        value: 'dropdown',
        label: 'Dropdown'
      }, {
        value: 'steps',
        label: 'Steps'
      }
    ]
  }, {
    value: 'others',
    label: 'Others',
    children: [{
      value: 'dialog',
      label: 'Dialog'
    }, {
      value: 'tooltip',
      label: 'Tooltip'
    }, {
      value: 'popover',
      label: 'Popover'
    }, {
      value: 'card',
      label: 'Card'
    }, {
      value: 'carousel',
      label: 'Carousel'
    }, {
      value: 'collapse',
      label: 'Collapse'
    }
  ]
}
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}
}
}

```

```

    }
  }
</script>

```

...

Cascader panel

`CascaderPanel` est le composant principal de `Cascader`. Il comporte diverses fonctionnalités telles que la sélection unique, la sélection multiple, le chargement dynamique, etc.

:::demo Tout comme `el-cascader`, vous pouvez définir des options alternatives par `options`, et activer d'autres fonctionnalités par `props`, voir le formulaire d'API ci-dessous pour plus de détails.

```
<el-cascader-panel :options="options"></el-cascader-panel>
```

```

<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {
              value: 'controllability',
              label: 'Controllability'
            }
          ]
        }, {
          value: 'navigation',
          label: 'Navigation',
          children: [{
            value: 'side nav',
            label: 'Side Navigation'
          }, {
            value: 'top nav',
            label: 'Top Navigation'
          }
        ]
      }
    ], {
      value: 'component',

```

```
label: 'Component',
children: [{
  value: 'basic',
  label: 'Basic',
  children: [{
    value: 'layout',
    label: 'Layout'
  }, {
    value: 'color',
    label: 'Color'
  }, {
    value: 'typography',
    label: 'Typography'
  }, {
    value: 'icon',
    label: 'Icon'
  }, {
    value: 'button',
    label: 'Button'
  }]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
```

```
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }
]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }
], {
```

```

        value: 'notification',
        label: 'Notification'
    ]],
    {
        value: 'navigation',
        label: 'Navigation',
        children: [
            {
                value: 'menu',
                label: 'NavMenu'
            },
            {
                value: 'tabs',
                label: 'Tabs'
            },
            {
                value: 'breadcrumb',
                label: 'Breadcrumb'
            },
            {
                value: 'dropdown',
                label: 'Dropdown'
            },
            {
                value: 'steps',
                label: 'Steps'
            }
        ]
    },
    {
        value: 'others',
        label: 'Others',
        children: [
            {
                value: 'dialog',
                label: 'Dialog'
            },
            {
                value: 'tooltip',
                label: 'Tooltip'
            },
            {
                value: 'popover',
                label: 'Popover'
            },
            {
                value: 'card',
                label: 'Card'
            },
            {
                value: 'carousel',
                label: 'Carousel'
            },
            {
                value: 'collapse',
                label: 'Collapse'
            }
        ]
    }
],
{
    value: 'resource',
    label: 'Resource',
    children: [
        {
            value: 'axure',
            label: 'Axure Components'
        }
    ],
    {

```

```

    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }]
  ]]
  }
  };
}
};
</script>

```

⋮

Cascader Attributes

Attribut	Description	Type	Valeurs acceptées	Défaut
value / v-model	Valeur de liaison	-	—	—
options	Les données des options, la clé de <code>value</code> et <code>label</code> peuvent être personnalisées par <code>Props</code> .	array	—	—
props	Options de configuration, voir le tableau suivant.	object	—	—
size	Taille de l'entrée	string	medium / small / mini	—
placeholder	Placeholder de l'input	string	—	Select
disabled	Si Cascader est désactivé	boolean	—	false
clearable	Si la valeur sélectionnée peut être effacée	boolean	—	false
show-all-levels	Afficher ou non tous les niveaux de la valeur sélectionnée dans l'entrée	boolean	—	true
collapse-tags	Réduire ou non les balises en mode de sélection multiple	boolean	-	false
separator	Option label separator	string	—	' / '
filterable	Si les options peuvent être recherchées	boolean	—	—
filter-method	Personnaliser la logique de recherche, le premier paramètre est <code>node</code> , le second est <code>keyword</code> , et doit renvoyer une valeur booléenne indiquant si le résultat est positif.	function(node, keyword)	-	-
debounce	Délai de réponse lors de la saisie du mot clé de filtre, en millisecondes	number	—	300
before-	Hook fonction avant de filtrer avec la valeur à	function(value)	—	—

filter	filtrer en tant que paramètre. Si <code>false</code> est renvoyé ou si une <code>Promise</code> est renvoyée puis rejetée, le filtrage sera annulé			
popper-class	Nom de classe personnalisé pour la liste déroulante de Cascader	string	—	—

Cascader Events

Event Name	Description	Paramètres
change	Se déclenche lorsque la valeur de liaison change	value
expand-change	Se déclenche lorsque l'option d'agrandissement change	an array of the expanding node's parent nodes
blur	Se déclenche lorsque Cascader blurs	(event: Event)
focus	Se déclenche lorsque Cascader à le focus	(event: Event)
visible-change	Se déclenche lorsque le menu déroulant apparaît / disparaît	vrai quand il apparaît, et faux sinon
remove-tag	Se déclenche lors de la suppression d'une balise en mode de sélection multiple	la valeur de la balise qui est supprimée

Cascader Methods

Method Name	Description	Parameters
getCheckedNodes	get an array of currently selected node	(leafOnly) whether only return the leaf checked nodes, default is <code>false</code>

Cascader Slots

Slot Name	Description
-	Le contenu personnalisé du noeud cascader, le paramètre est <code>{ node, data }</code> , qui sont respectivement les objets de noeud et les données de noeud actuels.
empty	Contenu quand il n'y a pas d'options correspondantes.

CascaderPanel Attributes

Attribut	Description	Type	Valeurs acceptées	Défaut
value / v-model	Baleur de liaison	-	—	—
options	Les données des options, la clé de <code>value</code> et <code>label</code> peuvent être personnalisées par <code>Props</code> .	array	—	—

props	Options de configuration, voir le tableau suivant.	object	—	—
-------	--	--------	---	---

CascaderPanel Events

Event Name	Description	Paramètres
change	Se déclenche lorsque la valeur de liaison change	value
expand-change	Se déclenche lorsque l'option d'agrandissement change	an array of the expanding node's parent nodes

CascaderPanel Methods

Method Name	Description	Parameters
getCheckedNodes	get an array of currently selected node	(leafOnly) whether only return the leaf checked nodes, default is <code>false</code>
clearCheckedNodes	clear checked nodes	-

CascaderPanel Slots

Slot Name	Description
-	Le contenu personnalisé du noeud cascader, le paramètre est { node, data }, qui sont respectivement les objets de noeud et les données de noeud actuels.

Props

Attribute	Description	Type	Valeurs acceptées	Défaut
expandTrigger	Mode de déclenchement des options de développement	string	click / hover	'click'
multiple	Si la sélection multiple est activée	boolean	-	false
checkStrictly	Si l'état coché d'un noeud n'affecte pas ses noeuds parents et enfants	boolean	-	false
emitPath	Lorsque les noeuds cochés changent, émet ou non un tableau du chemin du noeud. Si la valeur est false, n'émet que la valeur du noeud.	boolean	-	true
lazy	S'il faut charger dynamiquement les noeuds enfants, utiliser avec l'attribut <code>lazyload</code>	boolean	-	false
lazyLoad	Méthode de chargement de données de noeuds enfants, ne fonctionne que lorsque <code>lazy</code> est vrai	function(node, resolve)	-	-
value	Spécifie quelle clé de l'objet noeud est utilisée comme valeur du noeud	string	—	'value'

label	Spécifie quelle clé de l'objet noeud est utilisée comme label du noeud	string	—	'label'
children	Spécifie quelle clé de l'objet noeud est utilisée en tant qu'enfant du noeud	string	—	'children'
disabled	Spécifie quelle clé de l'objet noeud est utilisée comme noeud désactivé	string	—	'disabled'
leaf	Spécifie quelle clé de l'objet est noeud utilisée comme champ feuille du noeud	string	—	'leaf'