

Double column transfer choice box.

## When To Use

- It is a select control essentially which can be use for selecting multiple items.
- Transfer can display more information for items and take up more space.

Transfer the elements between two columns in an intuitive and efficient way.

One or more elements can be selected from either column, one click on the proper `direction` button, and the transfer is done. The left column is considered the `source` and the right column is considered the `target` . As you can see in the API description, these names are reflected in.

## API

Property	Description	Type	Default	Version
dataSource	Used for setting the source data. The elements that are part of this array will be present the left column. Except the elements whose keys are included in <code>targetKeys</code> prop	<a href="#">RecordType</a> extends <a href="#">TransferItem</a> = <a href="#">TransferItem</a> []	[]	
disabled	Whether disabled transfer	boolean	false	
filterOption	A function to determine whether an item should show in search result list	(inputValue, option): boolean	-	
footer	A function used for rendering the footer	(props, { direction }) => ReactNode	-	direction: 4.17.0
listStyle	A custom CSS style used for rendering the transfer columns	object   ({direction: left   right}) => object	-	
locale	The i18n text including filter, empty text, item unit, etc	{ itemUnit: string; itemsUnit: string; searchPlaceholder: string; notFoundContent: ReactNode   ReactNode[]; }	{ itemUnit: item, itemsUnit: items, notFoundContent: The list is empty, searchPlaceholder: Search here }	
oneWay	Display as single direction style	boolean	false	4.3.0
operations	A set of operations that are sorted from top to bottom	string[]	[>, <]	
operationStyle	A custom CSS style used for rendering the	object	-	

	operations column			
pagination	Use pagination. Not work in render props	boolean   { pageSize: number, simple: boolean, showSizeChanger?: boolean, showLessItems?: boolean }	false	4.3.0
render	The function to generate the item shown on a column. Based on an record (element of the dataSource array), this function should return a React element which is generated from that record. Also, it can return a plain object with <code>value</code> and <code>label</code> , <code>label</code> is a React element and <code>value</code> is for title	(record) => ReactNode	-	
selectAllLabels	A set of customized labels for select all checkboxes on the header	(ReactNode   (info: { selectedCount: number, totalCount: number }) => ReactNode)[]	-	
selectedKeys	A set of keys of selected items	string[]	[]	
showSearch	If included, a search box is shown on each column	boolean	false	
showSelectAll	Show select all checkbox on the header	boolean	true	
status	Set validation status	'error'   'warning'	-	4.19.0
targetKeys	A set of keys of elements that are listed on the right column	string[]	[]	
titles	A set of titles that are sorted from left to right	ReactNode[]	-	
onChange	A callback function that is executed when the transfer between columns is complete	(targetKeys, direction, moveKeys): void	-	
onScroll	A callback function which is	(direction, event):	-	

	executed when scroll options list	void		
onSearch	A callback function which is executed when search field are changed	(direction: left   right, value: string): void	-	
onSelectChange	A callback function which is executed when selected items are changed	(sourceSelectedKeys, targetSelectedKeys): void	-	

## Render Props

Transfer accept `children` to customize render list, using follow props:

Property	Description	Type	Version
direction	List render direction	left   right	
disabled	Disable list or not	boolean	
filteredItems	Filtered items	RecordType[]	
selectedKeys	Selected items	string[]	
onItemSelect	Select item	(key: string, selected: boolean)	
onItemSelectAll	Select a group of items	(keys: string[], selected: boolean)	

## example

```
<Transfer {...props}>{listProps => <YourComponent {...listProps} />}</Transfer>
```

## Warning

According the [standard](#) of React, the key should always be supplied directly to the elements in the array. In Transfer, the keys should be set on the elements included in `dataSource` array. By default, `key` property is used as an unique identifier.

If there's no `key` in your data, you should use `rowKey` to specify the key that will be used for uniquely identify each element.

```
// eg. your primary key is `uid`
return <Transfer rowKey={record => record.uid} />;
```

## FAQ

### How to support fetch and present data from a remote server in Transfer column.

In order to keep the page number synchronized, you can disable columns you checked without removing the option:

<https://codesandbox.io/s/objective-wing-6iqbx>