

[Ant Motion](#) 是 Ant Design 中提炼出来的动效语言。他不仅仅是动效语言，同时也是一套 React 框架动效解决方案，可以帮助开发者，更容易在项目中使用动效。我们提供了单项，组合动画，以及整套解决方案。

界面动效能加强用户认知且增加活力。

## 动效价值

- **增加体验舒适度：** 让用户认知过程更为自然。
- **增加界面活力：** 第一时间吸引注意力，突出重点。
- **描述层级关系：** 体现元素之间的层级与空间关系。
- **提供反馈、明确意向：** 助力交互体验。

## 衡量动效意义

衡量一个动效是否有意义，我们可以通过以下几个标准来考核：

- **一个动效的存在是否合理：** 是否带有明确的目的性，助力交互体验，没有多余的动效。
- **动效与性能：** 不能出现大幅度波动丢帧或者卡顿现象，动效的体验须是流畅的，并且不影响产品的性能。

## 原则

在企业级应用的产品设计中，使用动效和前台类产品有很大的不同，助力交互行为和信息认知的有效性会显得尤为重要，在 Ant Design 设计价值观的基础之上，我们衍生出动效设计的三原则：

```
import { Col, Row } from 'antd';

const text = [
  { title: '自然', img:
'https://gw.alipayobjects.com/zos/rmsportal/LyTPSGknLULxiVdwMWyu.gif', content: '自然
运动规律，保证视觉连贯，让用户感知到动作是成立的' },
  { title: '高效', img:
'https://gw.alipayobjects.com/zos/rmsportal/SQOZVQVIossbXpzDmiHu.gif', content: '尽量
节省过渡的时间，快速完成过渡的动画效果' },
  { title: '克制', img:
'https://gw.alipayobjects.com/zos/rmsportal/OkIXkscKxywYLSrilPIf.gif', content: '做有
意义的动效，不去做太多的修饰和干扰用户' },
];

function Principle() {
  const childrenToRender = text.map(item => (
    <Col key={item.title} sm={24} md={8} >
      <div className="principle">
        <div><img src={item.img} width="80%" /></div>
        <h4>{item.title}</h4>
        <p>{item.content}</p>
      </div>
    </Col>
  ));
  return (
    <div className="motion-principle-wrapper">
      <Row gutter={{ md: 32, sm: 0 }} className="principle-wrapper">
```

```
        {childrenToRender}
      </Row>
    </div>
  );
}

ReactDOM.render(<Principle />, mountNode);
```

## 自然

自然的动效背后体现的是自然运动规律。这就要求动效在转换时保证视觉上的连贯性，让用户感知到这个动作是成立的，是能够引起共鸣的。

