



实例化(constructor)

此时还没有数据获取，但已经存在默认配置项，在此周期可以做一些构造工作，如：生成组件的id、设置组件与数据无关的初始态等

awake周期

由于 `awake` 触发在平台注册事件 `eventsInit` 前，如果要补充 `play` 或 `stop` 函数可以在这个周期补充

平台注册事件(eventsInit)

这里组件无需感知具体注册了哪些事件，只是由于在平台侧存在默认隐藏的配置项，所以在这个周期中会触发一次 `play` 或 `stop` 函数，此后改变组件的显示隐藏也会触发对应函数。

start周期

此周期触发时认为组件渲染前准备工作都已完成，接下来将会获取数据，以及执行 `setOptions`

组件可以在此周期进行事件注册，比如 `_visibleChange` 事件等。

setOptions周期

组件后续的改变都经由 `setOptions` 实现，可能包括配置项 `_options`、数据 `_result`、布局 `_layout` 等，其中如果只有布局变化，则会经过 `resize` 函数实现

此周期已经获取了数据，与数据强相关的组件可能在此时才需要执行渲染。

对于无数据配置的组件也会触发一次此周期，传递参数为一个空对象。

组件抛出事件(emit)

组件渲染后需抛出 `rendered` 事件，以通知平台渲染成功，若失败则需抛出 `error` 事件。

e.g. `this.emit('rendered');` 或 `this.emit('error');`

异步组件的 `mounted` 事件也以相同方式抛出 `this.emit('mounted')`