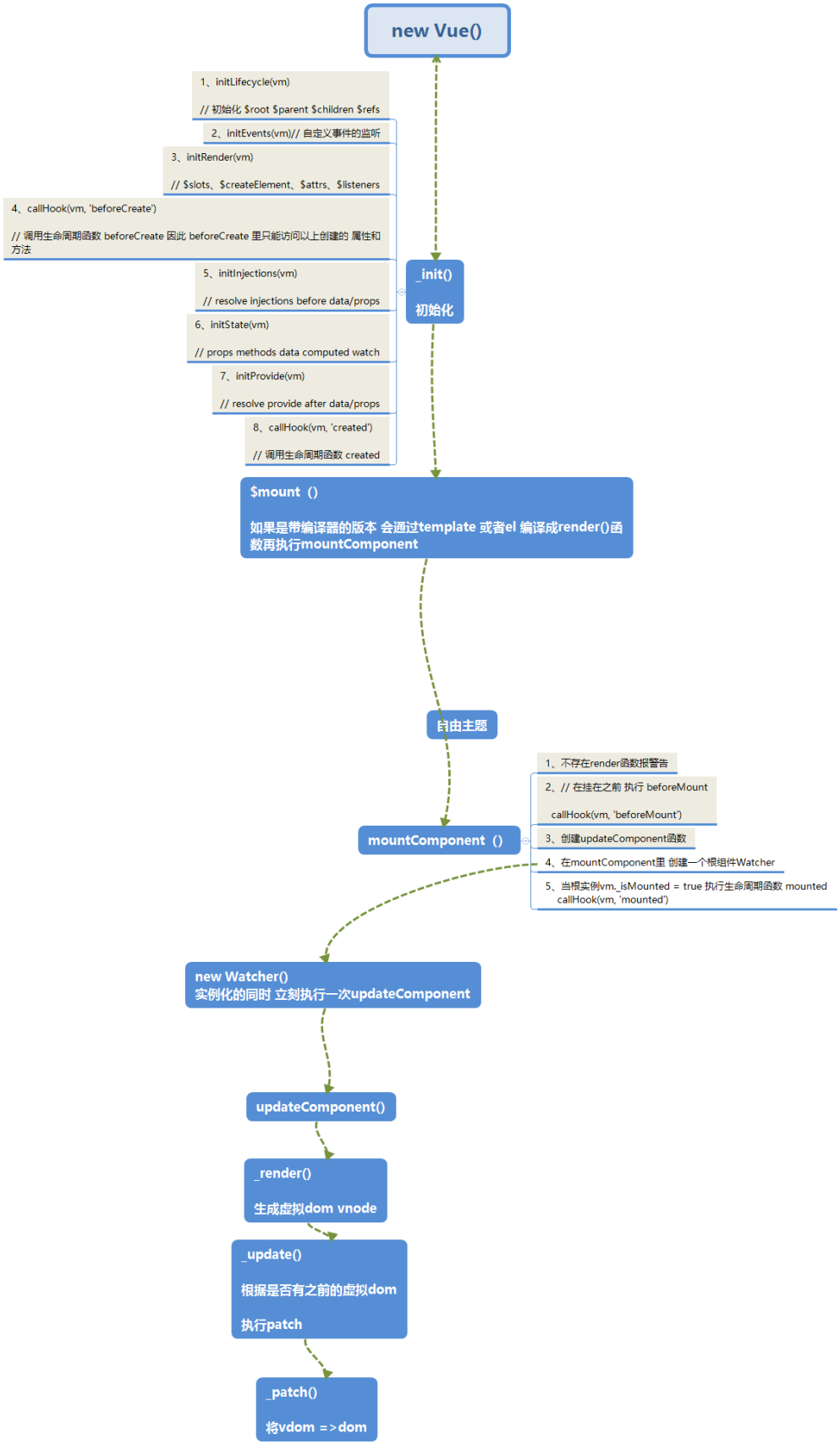
# new Vue()


new Vue()
1、initLifecycle(vm)
// 初始化 $root $parent $children $refs
2、initEvents(vm)// 自定义事件的监听
3、initRender(vm)
// $slots、$createElement、$attrs、$listeners
4、callHook(vm, 'beforeCreate')
// 调用生命周期函数 beforeCreate 因此 beforeCreate 里只能访问以上创建的 属性和方法
5、initInjections(vm)
// resolve injections before data/props
6、initState(vm)
// props methods data computed watch
7、initProvide(vm)
// resolve provide after data/props
8、callHook(vm, 'created')
// 调用生命周期函数 created
_init()
初始化
$mount ()
如果是带编译器的版本 会通过template 或者el 编译成render()函数再执行mountComponent
自由主题
mountComponent ()
1、不存在render函数报警告
2、// 在挂在之前 执行 beforeMount
callHook(vm, 'beforeMount')
3、创建updateComponent函数
4、在mountComponent里 创建一个根组件Watcher
5、当根实例vm._isMounted = true 执行生命周期函数 mounted
callHook(vm, 'mounted')
new Watcher()
实例化的同时 立刻执行一次updateComponent
updateComponent()
_render()
生成虚拟dom vnode
_update()
根据是否有之前的虚拟dom
执行patch
_patch()
将vdom =>dom

参见: _init()

初始化

## _init()

## 初始化

参见: new Vue(), $mount（）

如果是带编译器的版本 会通过template 或者el

编译成render()函数再执行mountComponent

**1、initLifecycle(vm)**

**// 初始化 $root $parent $children $refs**

**2、initEvents(vm)// 自定义事件的监听**

**3、initRender(vm)**

**// $slots、$createElement、$attrs、$listeners**

**4、callHook(vm, 'beforeCreate')**

**// 调用生命周期函数 beforeCreate 因此 beforeCreate 里只能访问以上创建的属性和方法**

**5、initInjections(vm)**

**// resolve injections before data/props**

**6、initState(vm)**

**// props methods data computed watch**

**7、initProvide(vm)**

**// resolve provide after data/props**

**8、callHook(vm, 'created')**

**// 调用生命周期函数 created**

**$mount（）**

**如果是带编译器的版本 会通过template 或者el**

**编译成render()函数再执行mountComponent**

参见: _init()

初始化, mountComponent（）

**mountComponent（）**

参见: $mount（）

如果是带编译器的版本 会通过template 或者el

编译成render()函数再执行mountComponent

**1、不存在render函数报警告**

**2、// 在挂在之前 执行 beforeMount**

**callHook(vm, 'beforeMount')**

**3、创建updateComponent函数**

**4、在mountComponent里 创建一个根组件Watcher**

参见: new Watcher()

实例化的同时 立刻执行一次updateComponent

**5、当根实例vm._isMounted = true 执行生命周期函数 mounted**

**callHook(vm, 'mounted')**

## new Watcher()

## 实例化的同时 立刻执行一次updateComponent

参见: 4、在mountComponent里 创建一个根组件Watcher, updateComponent()

## 自由主题

## updateComponent()

参见: new Watcher()

实例化的同时 立刻执行一次updateComponent, _render()

生成虚拟dom vnode

## _render()

## 生成虚拟dom vnode

参见: updateComponent(), _update()

根据是否有之前的虚拟dom

执行patch

## _update()

### 根据是否有之前的虚拟dom

### 执行patch

参见: _render()

生成虚拟dom vnode, _patch()

将vdom =>dom

## _patch()

### 将vdom =>dom

参见: _update()

根据是否有之前的虚拟dom

执行patch