

# 原型模式

2020年2月24日 12:12

## 1. 概念

指原型实例指定创建对象的种类，并且通过拷贝这些原型创建新的对象。不需要知道任何创建的细节，不调用构造函数。属于创建型。

**\*Cloneable接口只是作为标识接口,重写的是Object父类的Clone方法**

## 2. 模式原理

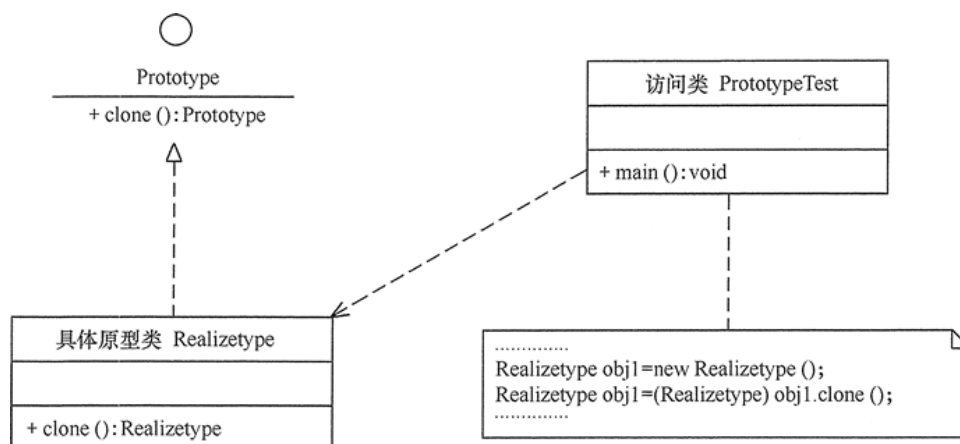


图2.1 原型模式UML结构图

1. 抽象原型类：规定了具体原型对象必须实现的接口。可把Cloneable接口当作一个抽象原型类，也可使用另外的interface对该借口进行扩展，但是接口要声明clone()方法。
2. 具体原型类：实现抽象原型类的 clone() 方法，它是可被复制的对象。
3. 访问类：使用具体原型类中的 clone() 方法来复制新的对象。

## 3. 优点

1. 性能比直接new一个对象性能高
2. 简化创建过程

## 4. 缺点

1. 必须配备克隆方法
2. 对克隆复杂对象或对克隆出的对象进行复杂改造时，容易引入风险
3. 深克隆、浅克隆得

当 - [https://blog.csdn.net/zhangjg\\_blog/article/details/18369201/](https://blog.csdn.net/zhangjg_blog/article/details/18369201/)

## 5. 应用场景

1. 类初始化消耗资源过多
2. new产生一个对象需要非常繁琐的过程（数据准备、访问权限等）
3. 构造函数比较复杂
4. 循环体中产生大量对象时

## 6. 扩展

原型模式可扩展为带原型管理器的原型模式，它在原型模式的基础上增加了一个原型管理器 `PrototypeManager` 类。该类用 `HashMap` 保存多个复制的原型。

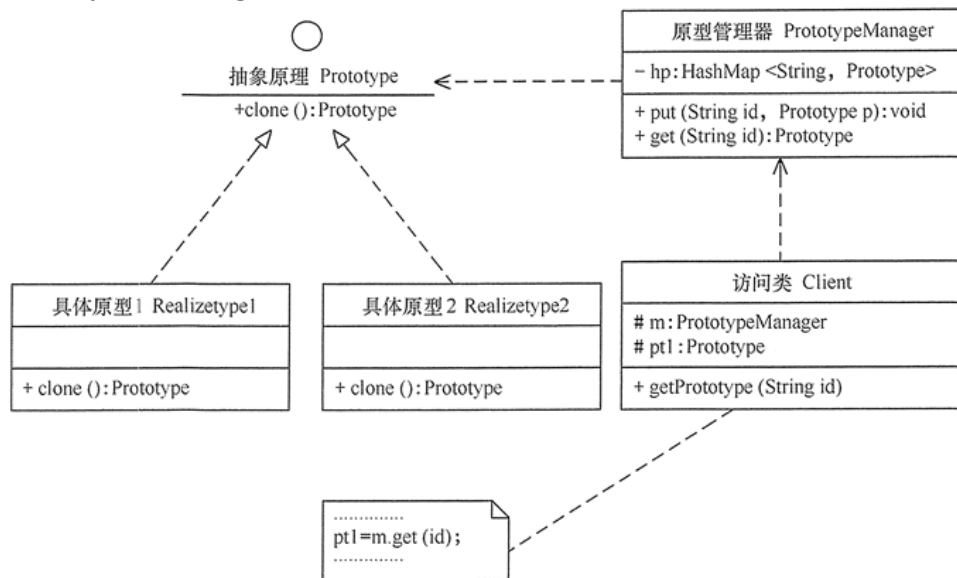


图6.1 原型模式扩展

## 7. 参考资料

[1] <http://c.biancheng.net/view/1343.html>