atitit.软件与sql设计模式原理与本质 大总结attialx总结v6 qc26.docx

[1.1. 版本历史 2](#_Toc7194)

[2. 设计模式是什么 2](#_Toc9008)

[2.1.](#_Toc14089) **[模式就是在一种场合下对某个问题的一个解决方案。"](#_Toc14089)** [2](#_Toc14089)

[2.2.](#_Toc2593) **[模式只是行话、规则、编程技巧、数据结构……"](#_Toc2593)** [2](#_Toc2593)

[3. 设计模式的优点 3](#_Toc1824)

[3.1. 知识固化 知识沉淀 精华沉淀，最佳实践固化 文档固化 3](#_Toc1725)

[3.2. 复用解决方案——通过复用已经公认的设计，我能够在解决问题时取得先发优势，而且避免重蹈前人覆辙 3](#_Toc22474)

[3.3. 大部分情况下可以提升可读性 3](#_Toc6631)

[3.4. 规范性 确立通用术语——开发中的交流和协作都需要共同的词汇基础和对问题的共识 3](#_Toc1272)

[3.5. 大多数设计模式还能使软件更容易修改和维护。 3](#_Toc31740)

[3.6. 提高观察高度－－模式还为我们提供了观察问题、设计过程和面向对象的更高层次的视角，这将使我们从“过早处理细节”的桎梏中解放出来 4](#_Toc18683)

[3.7. 提升某个方面的比如扩展性，兼容性 4](#_Toc31693)

[3.8. 综合起来提升开发效率 4](#_Toc17947)

[3.9. 防止发生 编码与文档不同步 4](#_Toc28241)

[4. 软件设计模式起源于建筑领域，而建筑工程中设计模式的数量（253个） 4](#_Toc4604)

[5. 设计模式的结构 5](#_Toc32002)

[5.1. 软件模式并非仅限于设计模式，还包括架构模式、分析模式和过程模式 6](#_Toc30991)

[6. 设计模式与框架的区别 比框架更抽象 7](#_Toc20478)

[7. 具体的设计模式（大概100个） 7](#_Toc8861)

[7.1. Gof 23个经典设计模式(截止1994年） 7](#_Toc14953)

[7.2. 软件新发展的设计模式(1995-2016年）attilax总结总和大约80个.. 7](#_Toc16953)

[8. 反模式 7](#_Toc673)

[8.1. Reset web 7](#_Toc10639)

[9. 参考资料 7](#_Toc27967)

软件的发展日新月异。。设计模式也是一直在发展。Gof 23个经典设计模式发表与1994年，也只记录了最常用的设计模式23个，但是，这个最常用的标准数量与内容因人而异和具体项目而不同，在当时1994年，设计模式的数量加上不常用的，按照20 80法则，应该总数也在100个左右。。

那么1995年以后，当然一直也有新的设计模式出现，只是没有经典书籍来总结了

软件新发展的设计模式(1995-2016年）attilax大概统计了下自身用到的设计模式大约80个

具体80个新总结的设计模式如下表所示。。

Mvc是不是设计模式，当然是设计模式来。。Struts，jsf，h5这类才是框架，才是实现mvc模式的代码实现。。..

## 版本历史

Frm atitit.软件设计模式大总结attialx总结v3 pa5

V5 完善了设计模式的原理 分离原理与具体的设计模式列表。。

# 设计模式是什么

软件模式是在软件开发中某些可重现问题的一些有效解决方法

有若干通用性的solu

## ****模式就是在一种场合下对某个问题的一个解决方案。"****

       这是Christopher Alexander的定义

## ****模式只是行话、规则、编程技巧、数据结构……"****

Design Pattern（设计模式）的目标是，把共通问题中的不变部分和变化部分分离出来。不变的部分，就构成了Design Pattern（设计模式）。这一点和Framework（框架）有些象。

#---如何抽取问题中的不变部分

下面举个排序的例子，说明如何抽取问题中的不变部分。

再次，按照field3的大小从小到大进行升序排序。

...

这种要求太多了，我们写了太多的重复代码。我们可以看到，问题的变化部分，只有判断条件部分（黑体的if条件判断语句）。

我们可以引入一个Comparator接口，把这个变化的部分抽取出来。

# 设计模式的优点

## 知识固化 知识沉淀 精华沉淀，最佳实践固化 文档固化

设计模式把最佳实践固化，防止各种乱七八糟的千疮百孔的不良实践

## 复用解决方案——通过复用已经公认的设计，我能够在解决问题时取得先发优势，而且避免重蹈前人覆辙

## 大部分情况下可以提升可读性

如果不用设计模式，一万个人可能一万个方法，设计模式熟悉起来，就那么有限的几个模式，容易看懂，提升可读性。

## 规范性 确立通用术语——开发中的交流和协作都需要共同的词汇基础和对问题的共识

## 大多数设计模式还能使软件更容易修改和维护。

其原因在于，它们都是久经考验的解决方案。所以，它们的结构都是经过长期发展形成的，比新构思的解决方案更善于应对变化。而且，这些模式所用代码往往更易于理解——从而使代码更易维护。

## 提高观察高度－－模式还为我们提供了观察问题、设计过程和面向对象的更高层次的视角，这将使我们从“过早处理细节”的桎梏中解放出来

## 提升某个方面的比如扩展性，兼容性

设计模式已经预留了合适的接口模式等，未了进来的扩展性兼容性做了准备。。

## 综合起来提升开发效率

以前要实现摸个东西，要自己实现，现在如果涉及模式，就会以文档，类库的模式固化

可以实现效率提升。。

## 防止发生 编码与文档不同步

# 软件设计模式起源于建筑领域，而建筑工程中设计模式的数量（253个）

与很多软件工程技术一样，模式起源于建筑领域，毕竟与只有几十年历史的软件工程相比，已经拥有几千年沉淀的建筑工程有太多值得学习和借鉴的地方。

      那么模式是如何诞生的？让我们先来认识一个人——Christopher Alexander（克里斯托弗.亚历山大），哈佛大学建筑学博士、美国加州大学伯克利分校建筑学教授、加州大学伯克利分校环境结构研究所所长、美国艺术和科学院院士……头衔真多，IMG_256，不过他还有一个“昵称”——模式之父(The father of patterns)。Christopher Alexander博士及其研究团队用了约20年的时间，对住宅和周边环境进行了大量的调查研究和资料收集工作，**发现人们对舒适住宅和城市环境存在一些共同的认同规律**，Christopher Alexander在著作A Pattern Language: Towns, Buildings, Construction中把这些认同规律归纳为253个**模式**，对每一个模式(Pattern)都从Context（前提条件）、Theme或Problem（目标问题）、 Solution（解决方案）三个方面进行了描述，并给出了从用户需求分析到建筑环境结构设计直至经典实例的过程模型。

   1990年，软件工程界开始关注ChristopherAlexander等在这一住宅、公共建筑与城市规划领域的重大突破。最早将模式的思想引入软件工程方法学的是1991-1992年以“四人组(Gang of Four，简称GoF，分别是**Erich Gamma, Richard Helm, Ralph Johnson和John Vlissides**)”自称的四位著名软件工程学者，他们在1994年归纳发表了23种在软件开发中使用频率较高的设计模式，旨在用模式来统一沟通面向对象方法在分析、设计和实现间的鸿沟

# 设计模式的结构

应用场景and条件Context

Problem

解决

不适合领域

优缺点

  软件模式是在软件开发中某些可重现问题的一些有效解决方法，软件模式的基础结构主要由四部分构成，包括问题描述【待解决的问题是什么】、前提条件【在何种环境或约束条件下使用】、解法【如何解决】和效果【有哪些优缺点】，如图1-1所示：

**图1-1 软件模式基本结构**

    软件模式与具体的应用领域无关，也就是说无论你从事的是移动应用

## 软件模式并非仅限于设计模式，还包括架构模式、分析模式和过程模式

GoF将模式的概念引入软件工程领域，这标志着软件模式的诞生。软件模式(Software Patterns)是将模式的一般概念应用于软件开发领域，即软件开发的总体指导思路或参照样板。软件模式并非仅限于设计模式，还包括架构模式、分析模式和过程模式等，实际上，**在软件开发生命周期的每一个阶段都存在着一些被认同的模式**。

# 设计模式与框架的区别 比框架更抽象

设计模式是对在某种环境中反复出现的问题以及解决该问题的方案的描述，它比框架更抽象

# 具体的设计模式（大概100个）

## Gof 23个经典设计模式(截止1994年）

Gof只总结了最常用的23个经典模式。当然还有很多次常用的设计模式没有总结进去，还有数据库与sql方面的设计模式也是么有考虑的，他们貌似只考虑了编程语言中的通用性的设计模式

## 软件新发展的设计模式(1995-2016年）attilax总结总和大约80个..

控件的onchange事件不能生效..

# 反模式

## Reset web

# 参考资料

Atitit. 设计模式的总结Oa7.doc

atitit.软件设计模式大总结attialx总结v3 pa5.doc

atitit.设计模式的理解.txt

atitit.设计模式总结o5o

Atitit 设计模式的本质思考】

作者:: 绰号:老哇的爪子 （ 全名：：Attilax Akbar Al Rapanui 阿提拉克斯 阿克巴 阿尔 拉帕努伊 ）

汉字名：艾提拉（艾龙），   EMAIL:1466519819@qq.com

转载请注明来源： http://blog.csdn.net/attilax

Atiend

设计模式的好处\_井底蛙\_新浪博客.html