Atitit 编程范式之道 attilax著 艾龙 著

[1. 编程范式与编程语言的关系是什么？ 1](#_Toc29096)

[2. LOP 面向语言编程（LOP, Language Oriented Programming） 2](#_Toc32242)

[3. OOP 2](#_Toc17494)

[4. AOP 2](#_Toc14865)

[5. Fp 函数式编程 2](#_Toc7707)

[6. 命令式 Imperative  2](#_Toc637)

[7. 其他 2](#_Toc17221)

[7.1. 泛型式、 2](#_Toc6357)

[7.2. 产生式编程（Generative Programming） 自动生成源代码的编程也属于另一种编程范式 2](#_Toc27479)

[7.3. 过程化（命令式）编程  3](#_Toc26988)

[7.4. 事件驱动编程  ]响应式编程范式-- 3](#_Toc207)

[7.5. 4个常用的编程范式：泛型式、元编程、切面式和事件驱动式。 3](#_Toc28272)

[8. 元编程 原来元编程就是编写能自动生成源代码的程序。" 3](#_Toc4130)

[8.1. 通过向导、拖放控件等方式自动生成源码 3](#_Toc6522)

[8.2. 产生式编程（Generative Programming 区别 3](#_Toc26518)

[8.3. 许多脚本语言都提供eval函数，可以在运行时将字符串作为表达式来运算[4]。 4](#_Toc24197)

[9. 参考资料 4](#_Toc14046)

[9.1. 编程范式，程序员的编程世界观 -- 简明现代魔法.html 4](#_Toc5012)

[9.2. 3.2 超级范式--提升语言的级别（2） - 51CTO.COM.mhtml 4](#_Toc13475)

[9.3. 编程范式思考问题 - huangshanchun的专栏 - CSDN博客.mhtml 4](#_Toc12945)

# 编程范式与编程语言的关系是什么？

如果把一门编程语言比作兵器，它的语法、工具和技巧等是招法，它采用的编程范式则是心法。

抽象的编程范式须要通过具体的编程语言来体现。范式的世界观体现在语言的核心概念中，范式的方法论体现在语言的表达机制中。一种语言的语法和风格与其所支持的编程范式密切相关。

# LOP 面向语言编程（LOP, Language Oriented Programming）

有人认为LOP是继OOP之后的下一个重要的编程范式，我们不妨拭目以待。＂ 句号整理了一下头绪：＂能不能这么说：如果处理一些复杂、非标准格式的文档，可以考虑用元...

语言导向式编程（LOP）通过创建一套专用语言DSL来编写程序。相比通用语言，DSL更简单、更抽象、更专业、更接近自然语言和声明式语言、开发效率更高，同时有助于专业程序员与业务分析员之间的合作。

语言导向式编程一般通过元编程将专用语言转化为通用语言。

# OOP

# AOP

# Fp 函数式编程

# 命令式 ****Imperative****

# 其他

## 泛型式、

## 产生式编程（Generative Programming） 自动生成源代码的编程也属于另一种编程范式

--产生式编程（Generative Programming）[3]的范畴

## 过程化（命令式）编程

## 事件驱动编程  ]响应式编程范式--

## 4个常用的编程范式：泛型式、元编程、切面式和事件驱动式。

# 元编程 原来元编程就是编写能自动生成源代码的程序。"

## 通过向导、拖放控件等方式自动生成源码

元编程的例子比比皆是：许多IDE如Visual Studio、Delphi、Eclipse等均能通过向导、拖放控件等方式自动生成源码；UML建模工具将类图转换为代码；Servlet引擎将JSP转换为Java代码；包括Spring、Hibernate、XDoclet在内的许多框架和工具都能从配置文件、annotation/attribute等中产生代码。"

## 产生式编程（Generative Programming 区别

也不尽然。"冒号马上修正道，"自动生成源代码的编程也属于另一种编程范式--产生式编程（Generative Programming）[3]的范畴。区别在于后者更看重代码的生成，而元编程看重的是生成代码的可执行性。另外，除了在编译期间生成源代码的静态元编程，还有能在运行期间修改程序的动态元编程。从低级的汇编语言到一些高级的动态语言如Perl、Python、Ruby、JavaScript、Lisp、Prolog等均支持此类功能。比如，

产生式编程与静态元编程都能自动生成源代码。产生式编程强调代码的生成，元编程强调生成代码的可执行性。此外，动态元编程并不生成源代码，但能在运行期间修改程序。

元程序将程序作为数据来对待，有着其他程序所不具备的自觉性、自适应性和智能性，可以说是一种最高级的程序。

## 许多脚本语言都提供eval函数，可以在运行时将字符串作为表达式来运算[4]。

# 参考资料

## 编程范式，程序员的编程世界观 -- 简明现代魔法.html

## 3.2 超级范式--提升语言的级别（2） - 51CTO.COM.mhtml

[编译]响应式编程范式--(1) - ttylinux - 博客园.mhtml

## 编程范式思考问题 - huangshanchun的专栏 - CSDN博客.mhtml