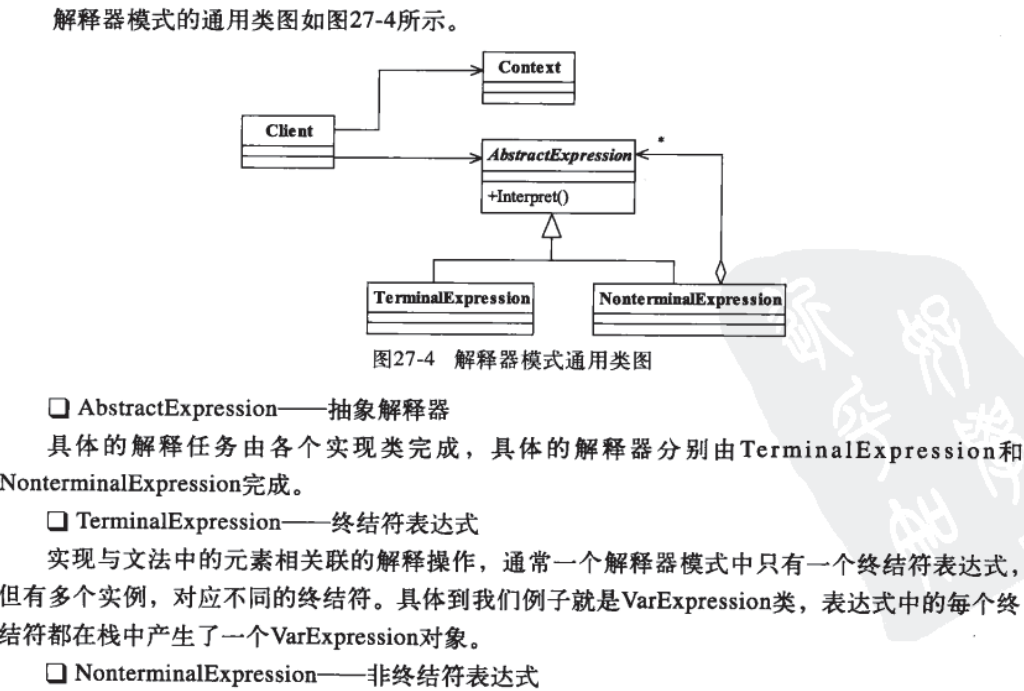
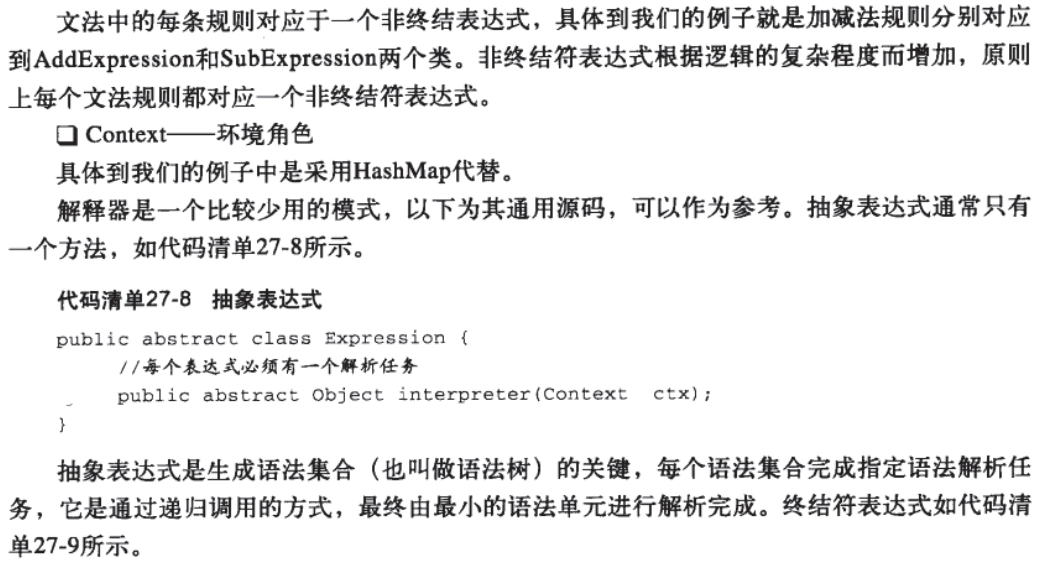
# 解释器模式

1. 定义

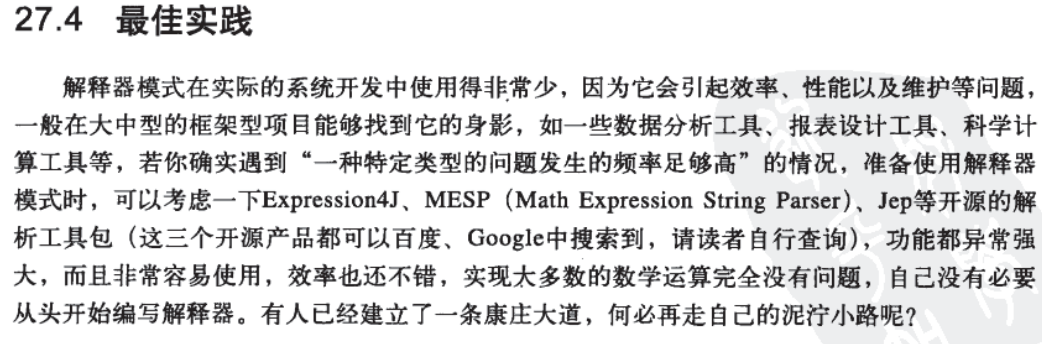
Given a language,define a representation for its grammar along with a interpreter that uses the representation to interpret sentences in the language(给定一种语言，定义它的文法的一种表示，并定义一个解释器，该解释器使用该表示来解释语言中的句子)

1. 类图





1. 模式的应用
2. 模式的优点
3. 扩展性好，修改语法规则，只需修改相应的非终结符表达式就可以。
4. 模式的缺点
5. 引起类的膨胀：每个语法都要产生一个非终结符表达式，语法比较复杂时，可能会产生大量的类文件；
6. 解释器采用递归调用方法：每一个非终结符表达式只关心自己有关的表达式，每一个表达式需要知道最终的结果必须一层一层地削茧，递归会调至调试非常困难；
7. 效率问题：大量循环和递归
8. 实际应用场景
9. 重复发生的问题；
10. 一个简单语法需要解释的场景；
11. 最佳实践



1. 注意事项

