# **看懂UML类图和时序图**

* + 这里不会将UML的各种元素都提到，我只想讲讲类图中各个类之间的关系； 能看懂类图中各个类之间的线条、箭头代表什么意思后，也就足够应对 日常的工作和交流； 同时，我们应该能将类图所表达的含义和最终的代码对应起来； 有了这些知识，看后面章节的设计模式结构图就没有什么问题了；
  + 本章所有图形使用Enterprise Architect 9.2来画,所有示例详见根目录下的design\_patterns.EAP

## **从一个示例开始**

* + 请看以下这个类图，类之间的关系是我们需要关注的：
    - 车的类图结构为<<abstract>>，表示车是一个抽象类；
    - 它有两个继承类：小汽车和自行车；它们之间的关系为实现关系，使用带空心箭头的虚线表示；
    - 小汽车为与SUV之间也是继承关系，它们之间的关系为泛化关系，使用带空心箭头的实线表示；
    - 小汽车与发动机之间是组合关系，使用带实心箭头的实线表示；
    - 学生与班级之间是聚合关系，使用带空心箭头的实线表示；
    - 学生与身份证之间为关联关系，使用一根实线表示；
    - 学生上学需要用到自行车，与自行车是一种依赖关系，使用带箭头的虚线表示；
  + 下面我们将介绍这六种关系；

## **类之间的关系**

### **泛化关系(generalization)**

* + 类的继承结构表现在UML中为：泛化(generalize)与实现(realize)：
  + 继承关系为 is-a的关系；两个对象之间如果可以用 is-a 来表示，就是继承关系：（..是..)
  + eg：自行车是车、猫是动物
  + 泛化关系用一条带空心箭头的直接表示；如下图表示（A继承自B）；
  + 首先这不是一个重复的问题，因为大多数人在从缺少“--- BEGIN RSA CERTIFICATE ---”行的证书创建公钥时报告此异常。
  + 我想要做的是gist
  + 1.使用SHA1withRSA算法（RSA密钥是1024位）在JCOP智能卡上签名一个50字节的消息。
  + 2.将签名从智能卡导出到服务器。
  + 3.验证服务器上的签名。
  + 智能卡上的代码段用于创建签名。关键点是我在Java卡中使用算法Signature.ALG\_RSA\_SHA\_PKCS1来创建签名。
  + private void setcustccid ）{
  + byte [] buffer = apdu.getBuffer（）;
  + if（buffer [ISO7816.OFFSET\_LC]！=（byte）24）{
  + ISOException.throwIt（ISO7816.SW\_WRONG\_LENGTH）;
  + }
  + else {
  + short bytesLeft =（short）（buffer [ISO7816.OFFSET\_LC]& 0x00FF）;
  + short readCount = apdu.setIncomingAndReceive（）;
  + if（readCount< bytesLeft）{
  + ISOException.throwIt（ISO7816.SW\_BYTES\_REMAINING\_00）;
  + }
  + try {
  + 签名签名= Signature.getInstance（Signature.ALG\_RSA\_SHA\_PKCS1，false）;
  + signature.init（privKey，Signature.MODE\_SIGN）;
  + Util.arrayCopy（buffer，（short）buffer [ISO7816.OFFSET\_CDATA]，tempStorage，（short）0，（byte）24）;
  + Util.arrayCopy（transactionHistory，（short）0，tempStorage，（short）24，（byte）30）;
  + }
  + catch（Exception ex）{
  + ISOException.throwIt（ISO7816.SW\_BYTES\_REMAINING\_00）;
  + }
  + signature.sign（tempStorage，（short）0，（short）50，finalEncryptedMsg，（short）0）;
  + }
  + }
  + 服务器端尝试验证签名的代码段从抛出异常的Java智能卡导出。这里的关键是我在服务器端使用Signature.getInstance（“SHA1withRSA”）。我正在对签名的消息进行密码解密，以确认生成的公钥是否正在工作。
  + modulusString = new BigInteger（1，rsaModulus）;
  + exponentString = new BigInteger（1，rsaExponent）;
  + RSAPublicKeySpec keySpec = new RSPublicKeySpec（modulusString，exponentString）;
  + KeyFactory factor = KeyFactory.getInstance（“RSA”）;
  + PublicKey publicKey =（RSAPublicKey）factor.generatePublic（keySpec）;
  + rsaCipher = Cipher.getInstance（“RSA”）;
  + rsaCipher.init（Cipher.DECRYPT\_MODE，publicKey）;
  + signature = Signature.getInstance（“SHA1withRSA”）;
  + signature.initVerify（publicKey）;
  + signature.update（resultBytes）;
  + signature.verify（finalEncryptedMsg）;
  + tempStorage = rsaCipher.doFinal（finalEncryptedMsg）;
  + System.out.println（“Decrypted Length =”+ tempStorage.length）;
  + 在signature.verify（）上发生异常。另一个线程引用这个相同的异常，但解决方案是添加Bouncy Castle作为Provider在Signature.getInstance（）。不知道为什么Bouncy城堡将需要验证签名。
  + 任何帮助将非常感谢。如果您需要更多的代码来识别问题，请告诉我。
  + java.security.SignatureException：签名编码错误
  + at sun.security.rsa.RSASignature.engineVerify（未知源）
  + at java.security.Signature $ Delegate.engineVerify（未知源）
  + at java.security.Signature.verify（未知源） ）
  + at com.mse.reader.SmartCardReader.main（SmartCardReader.java:234）
  + 引发者：java.io.IOException：序列标记错误
  + at sun.security.util.DerInputStream .getSequence（未知源）
  + at sun.security.rsa.RSASignature.decodeSignature（unknown Source）
  + ... 4 more