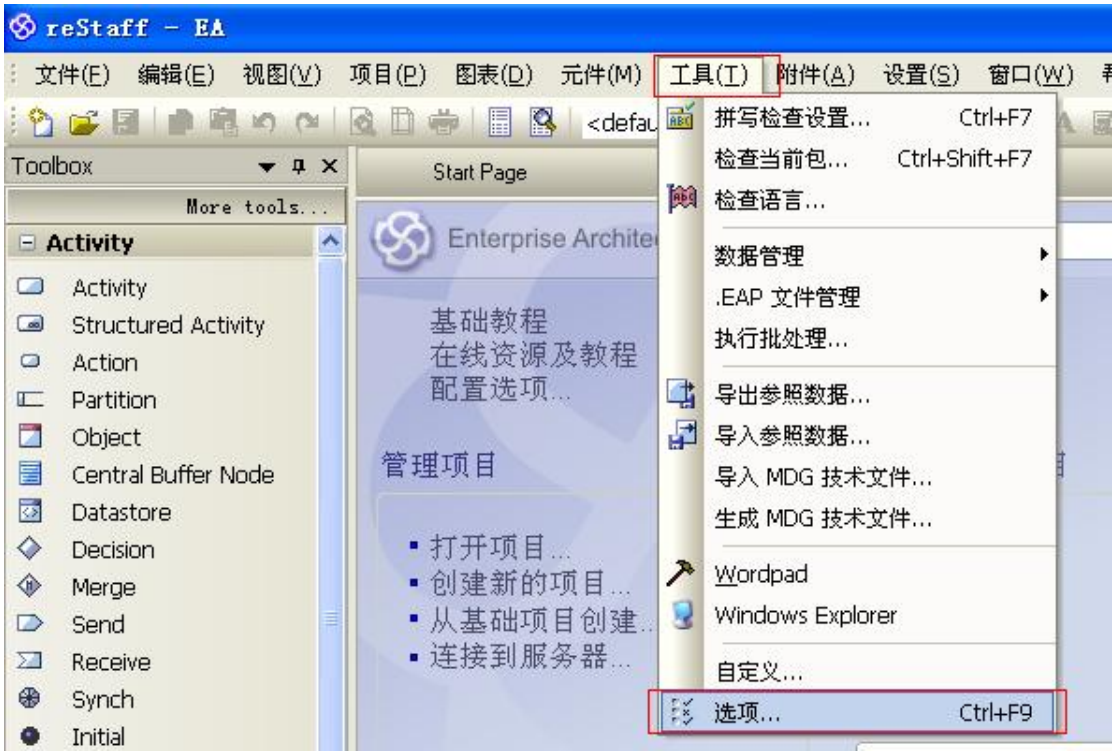
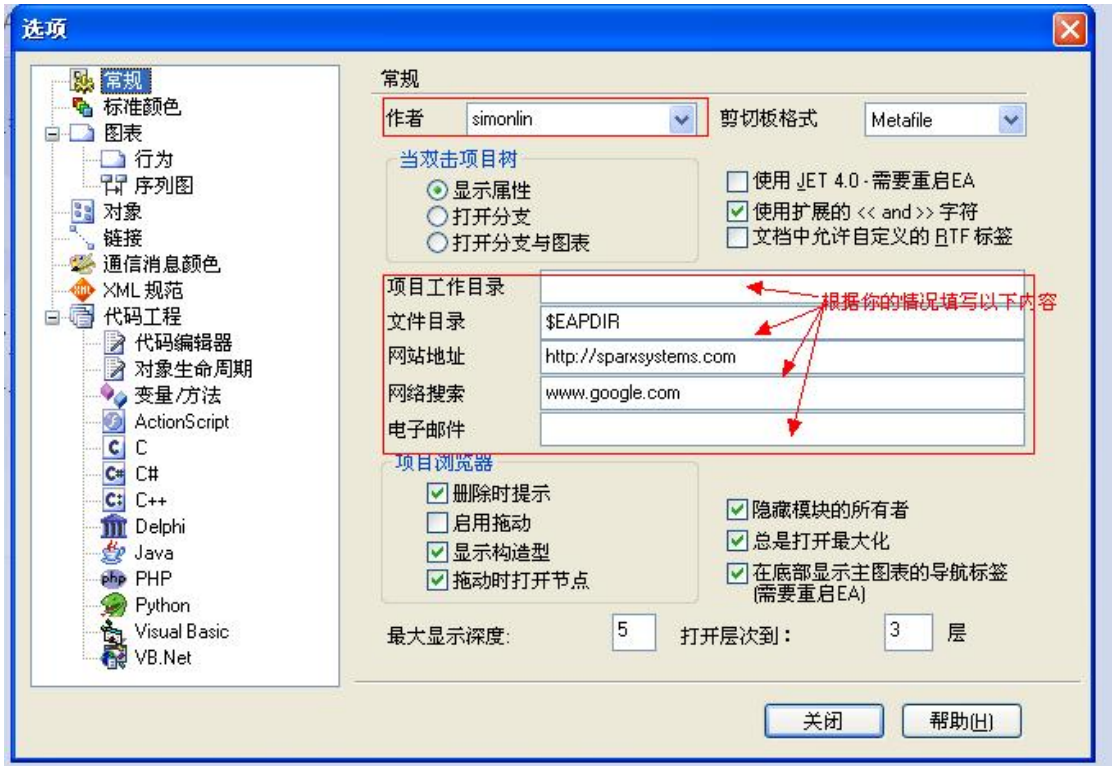
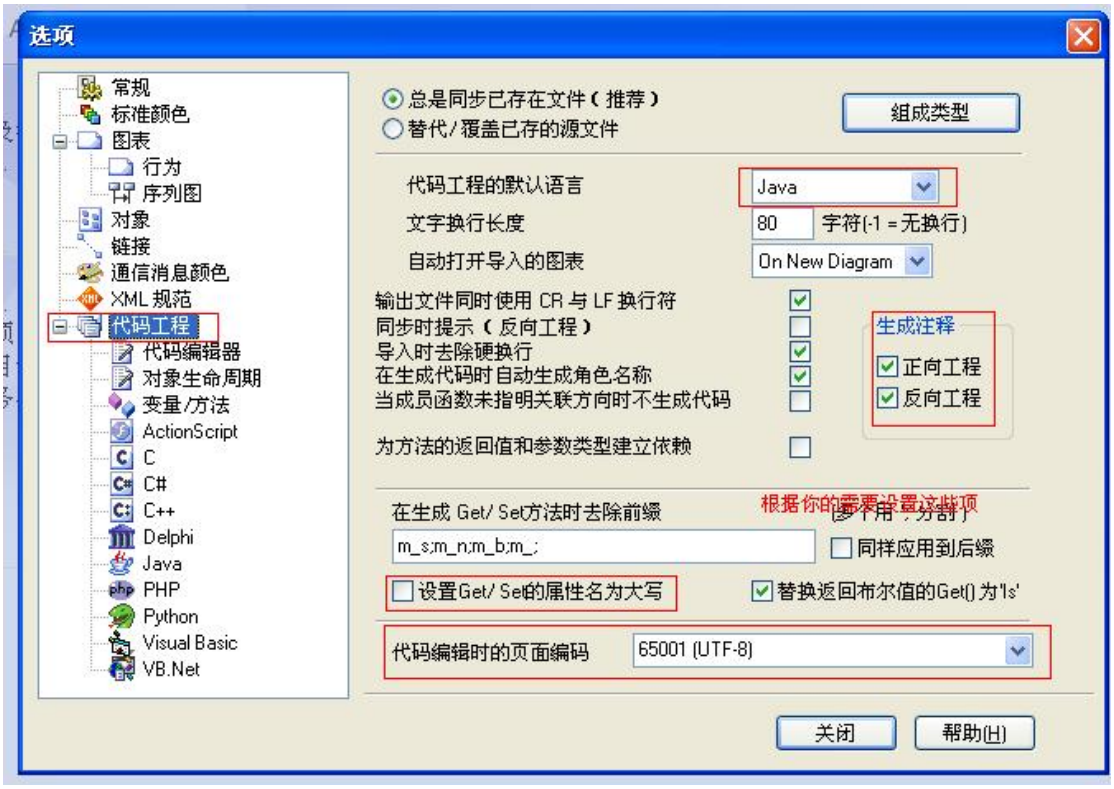
用 EA 轻松进行分析设计

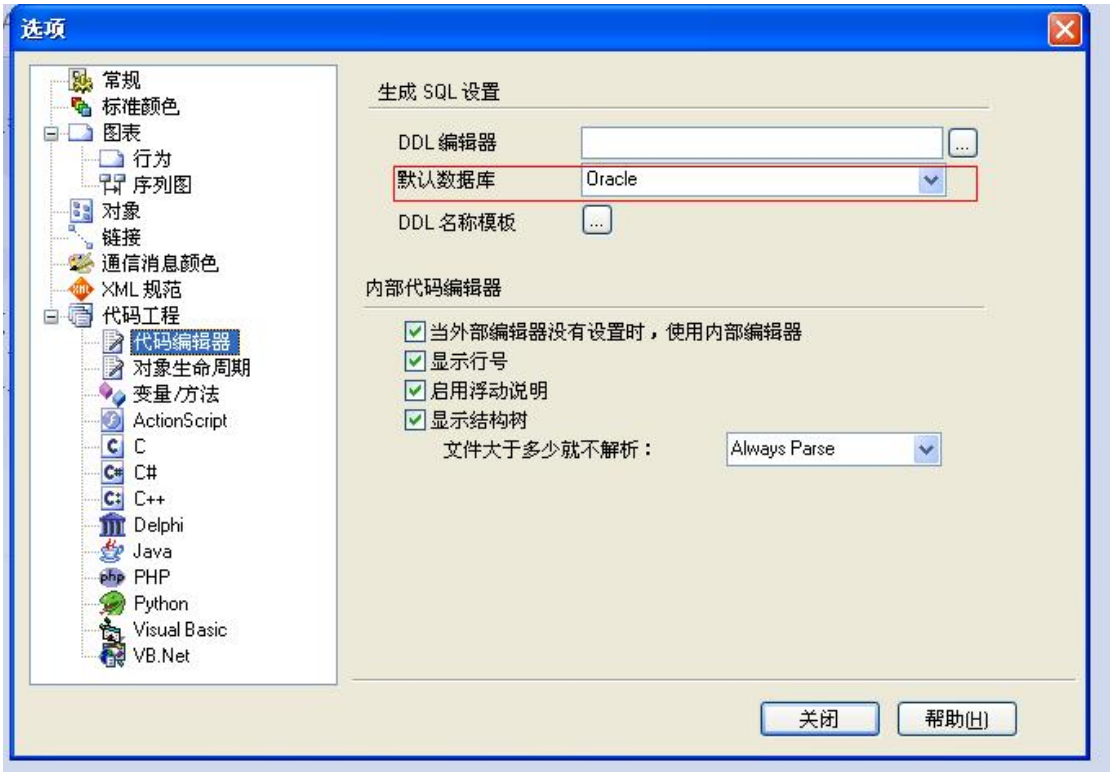
一、开始前的准备（进行前的设置）

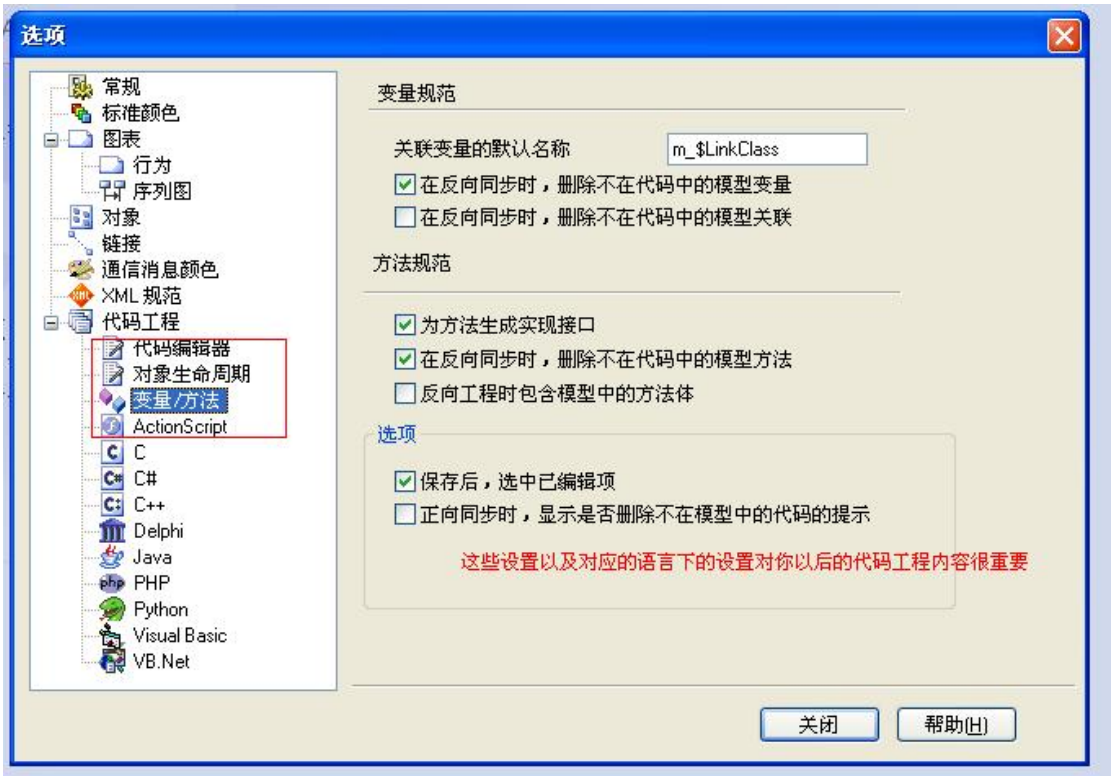






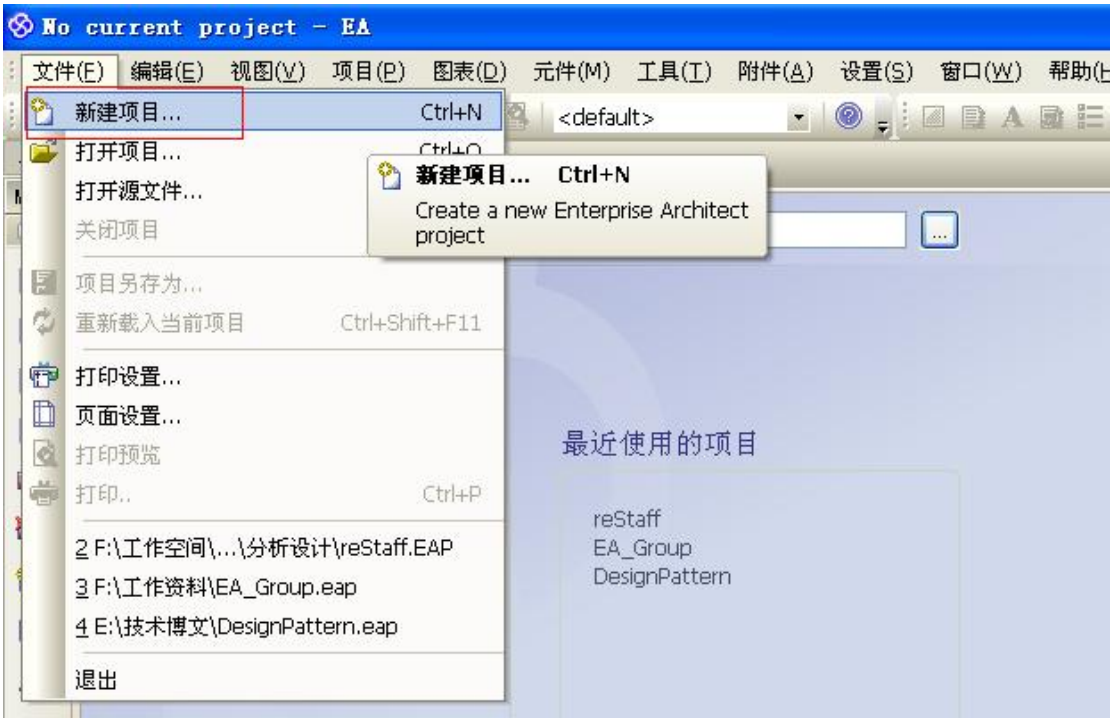
设置你默认使用的数据库



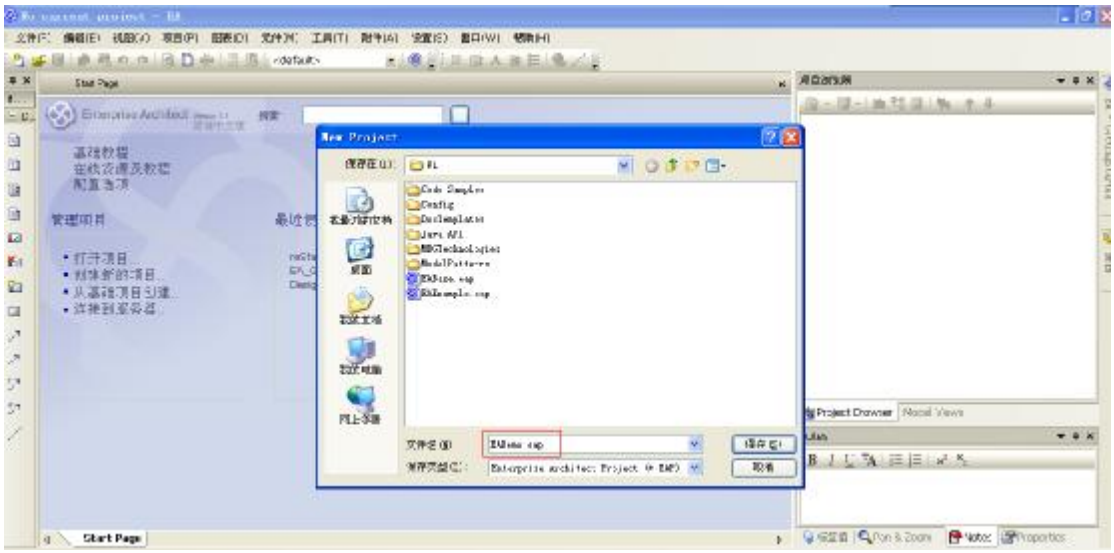


二、创建项目

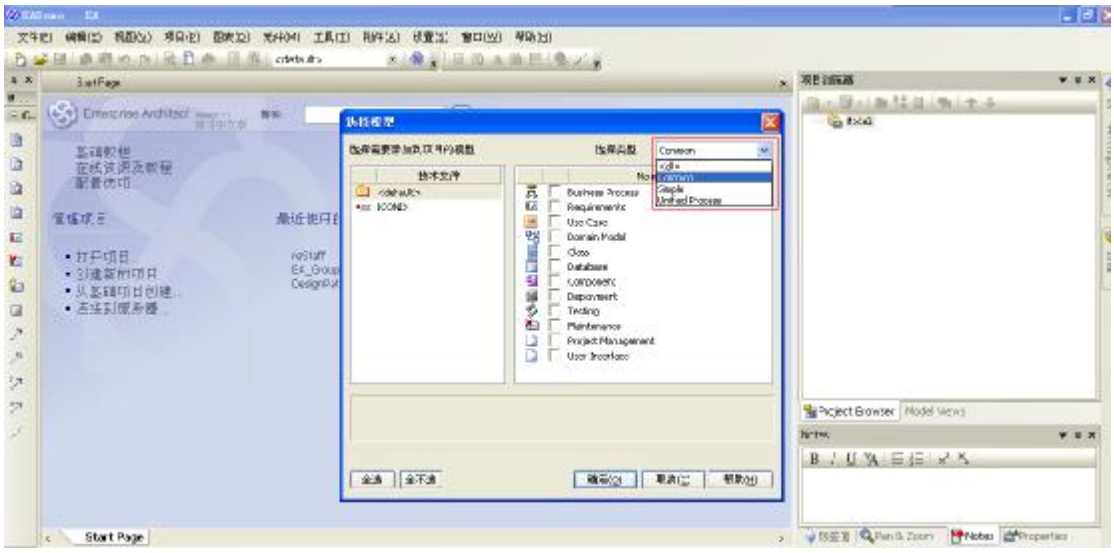
新建项目：



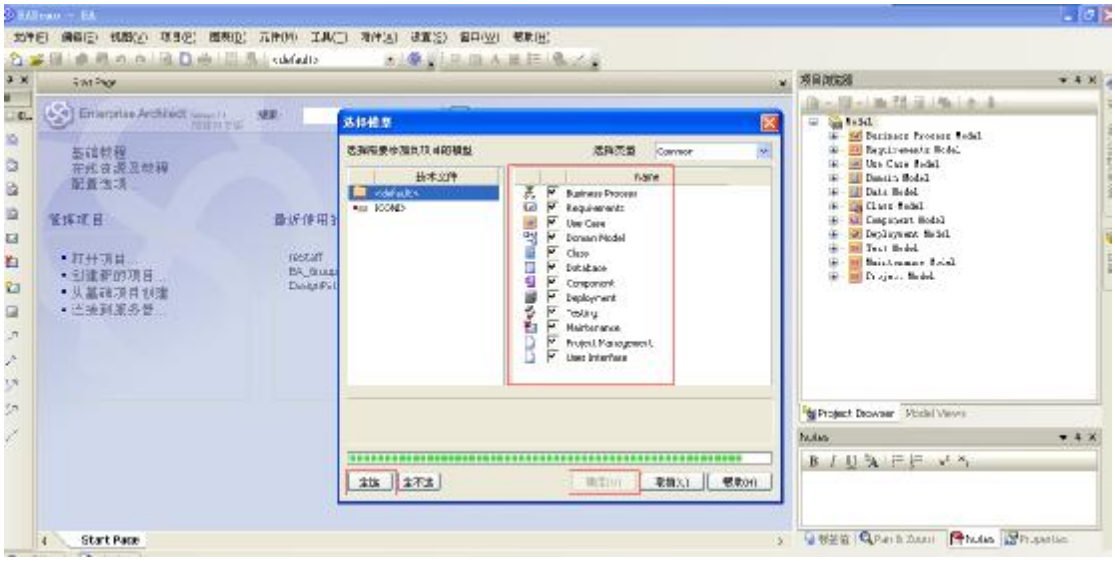
选择你要保存 EA 文件的路径，输入 EA 文件的文件名：



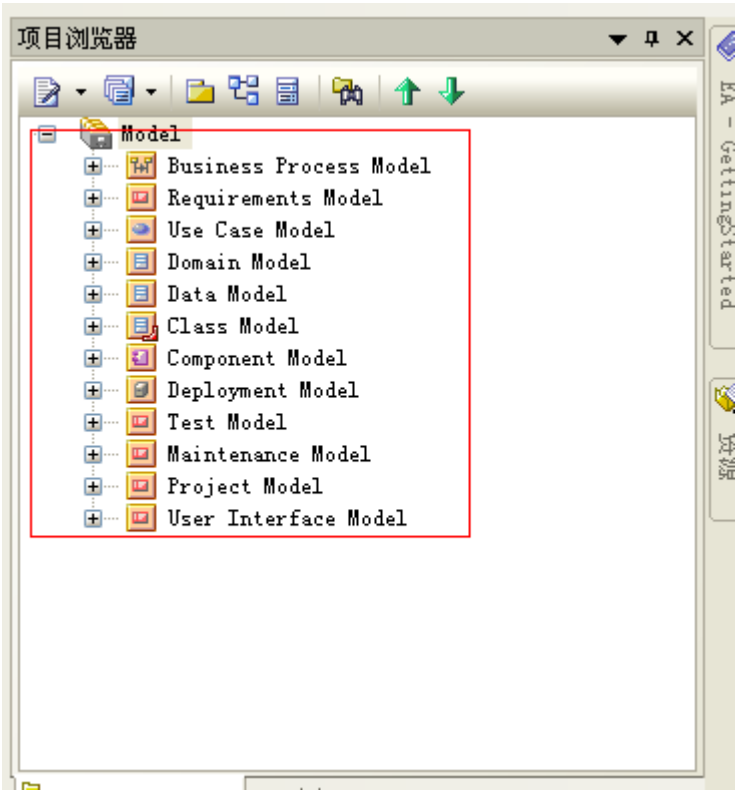
根据您项目的情况，选择需要的类型



选择模式下需要的模型（通常我是全选，在这里全选后，还可以在项目浏览器中删除，或则增添），按确定后，模型在创建。



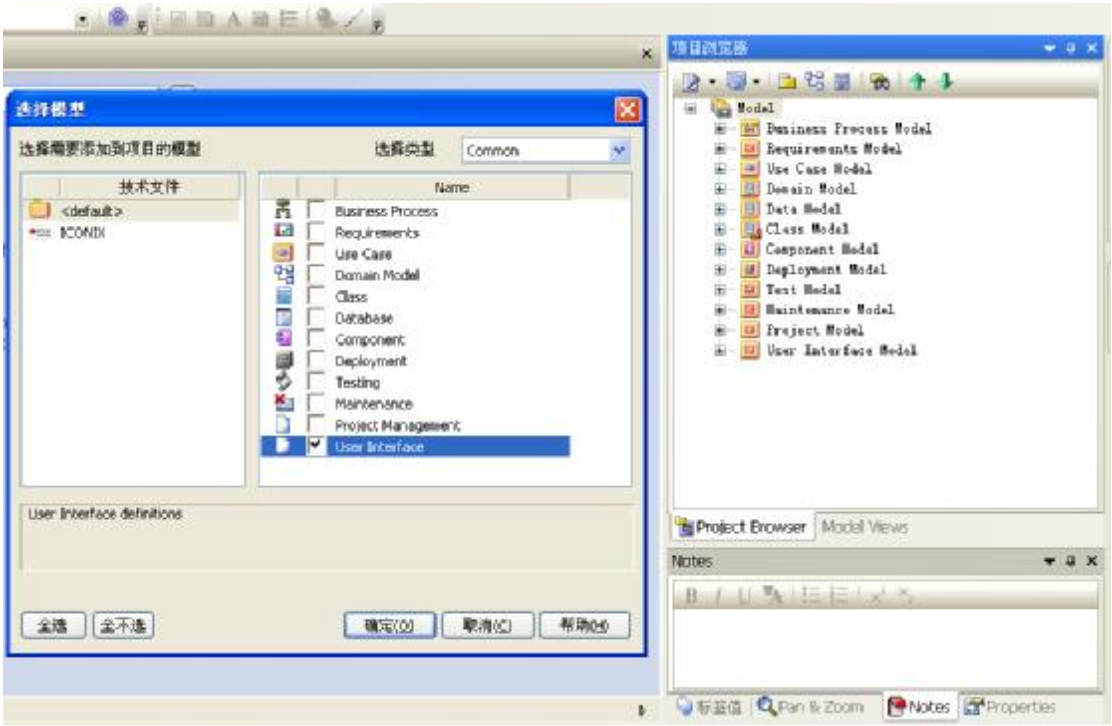
下面是创建好的项目模型



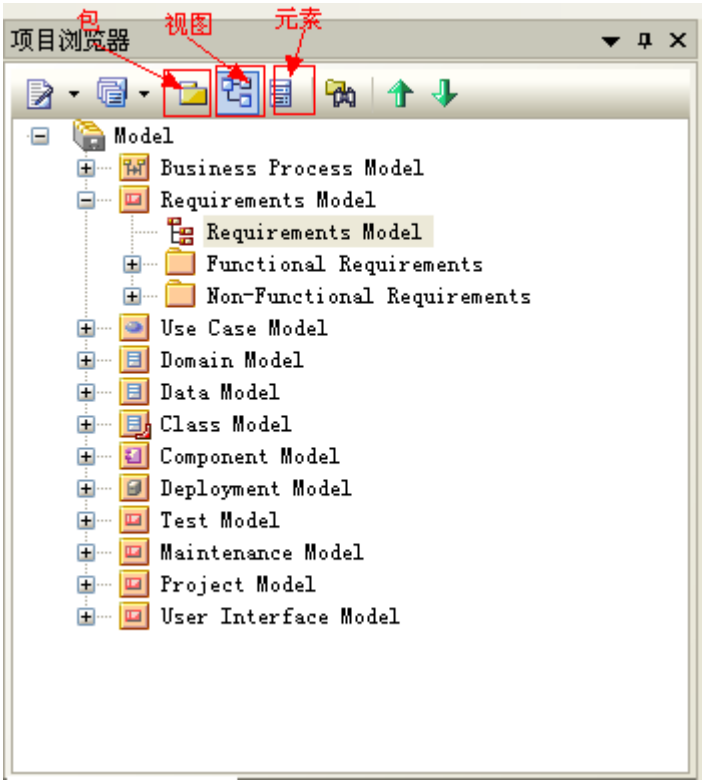
在这里可以删除某些认为不需要的模型，或者按一下步骤增添需要的模型

1、 选择 Model 点右键

2、 选择增添的模型



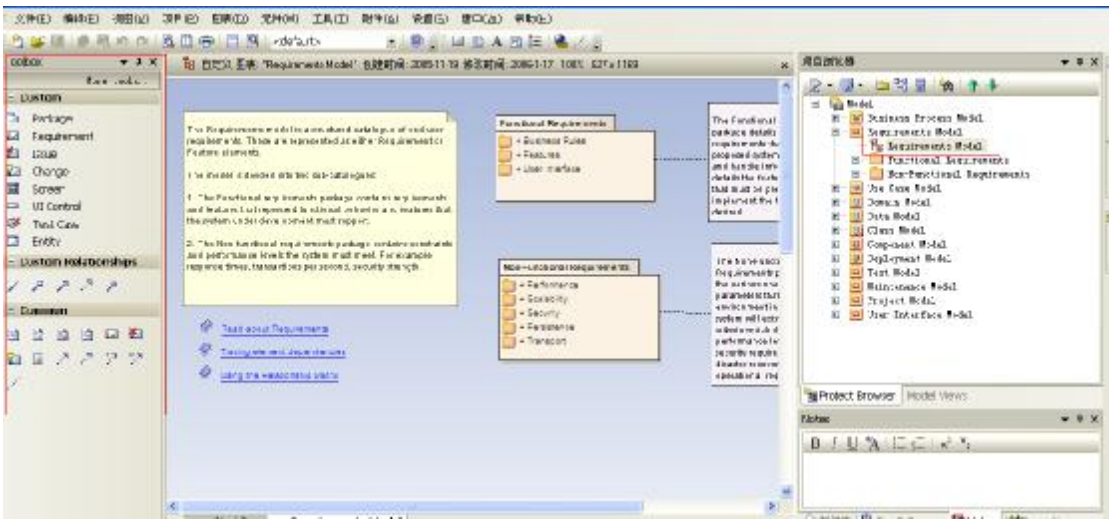
三、UML 中 3 个重要的东东



在进入实际工作之前，需要了解这 3 个 UML 中重要的东东，对在 EA 里的操作很重要（本来这是 UML 中的基础，本不该在这里讲，但是在使用 EA 时有好多人问题一些很基本的东西，所以就补上来吧）

包：是为了系统的结构划分而存在，主要是系统之间的功能分界，也可以看作是分层次，就像我们写一篇文章，需要分目录、章节一样；

视图：视图就像一块白板，用于表现存放各元素以及元素的关系，如用例图、类图，或者是存放包的包结构图。不同的视图有不同的含义，这里就不展开了，一个视图出现后，对应和此视图相关的元素会在左边出现，如：

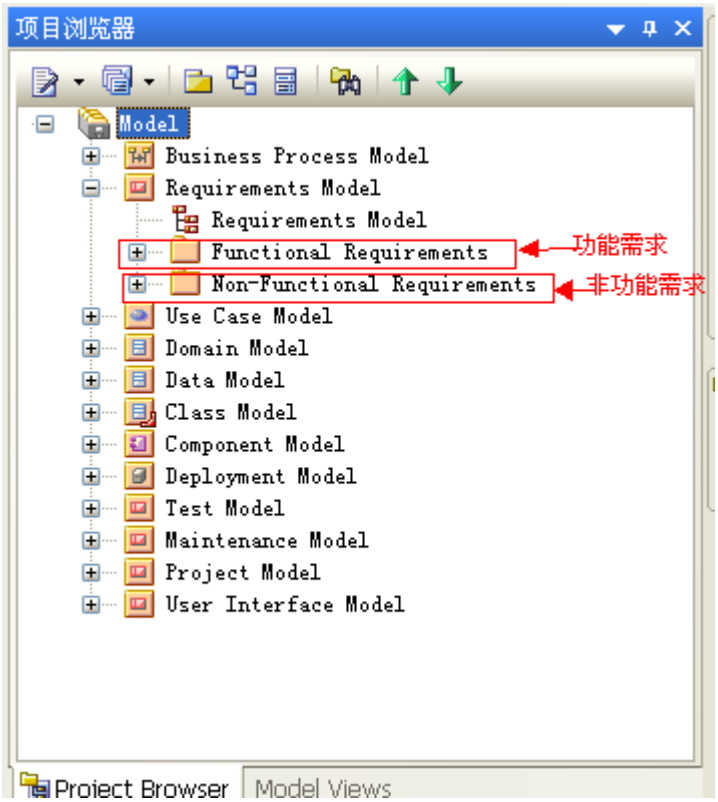


元素：既是 UML 中的元素，如：用例、类、表、包等等

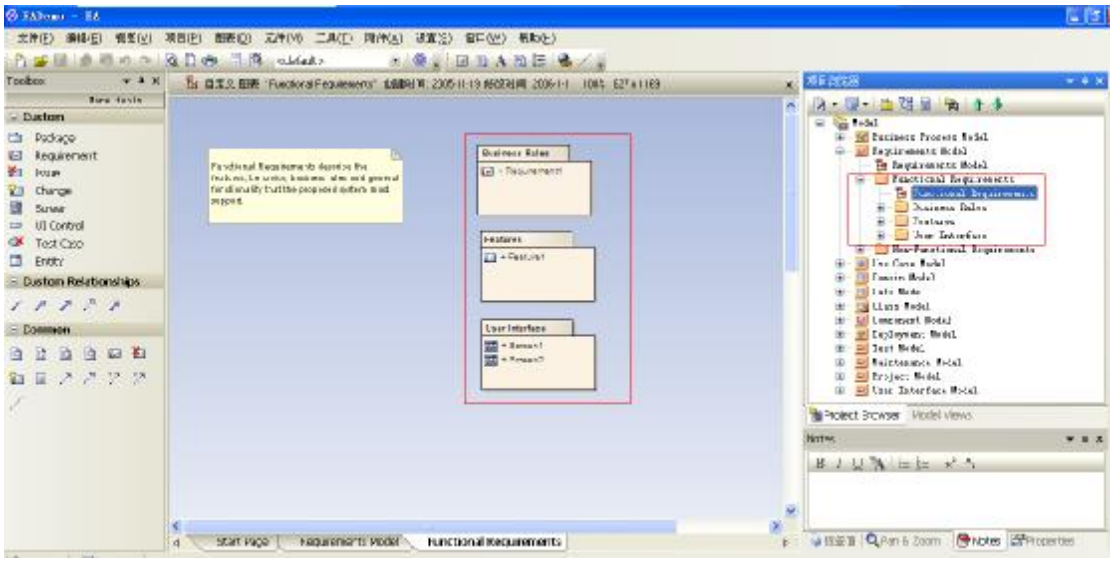
连接这些元素的不同的线，代表的是其之间的关系。

四、和客户沟通，记录需求

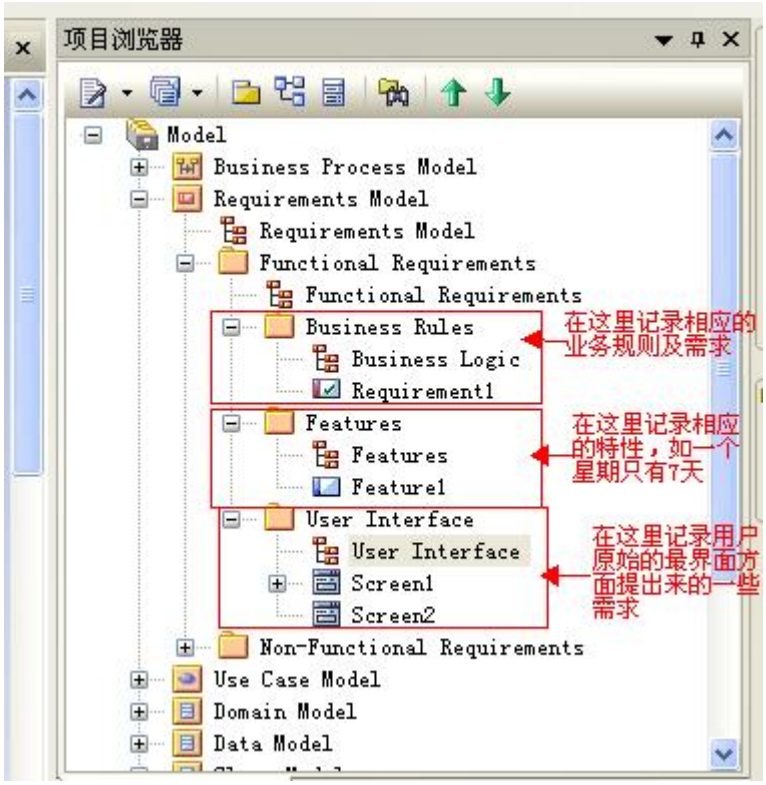
在这里记录下用户的原始需求，有分功能需求和非功能需求（如性能、兼容性、部署环境要求等）



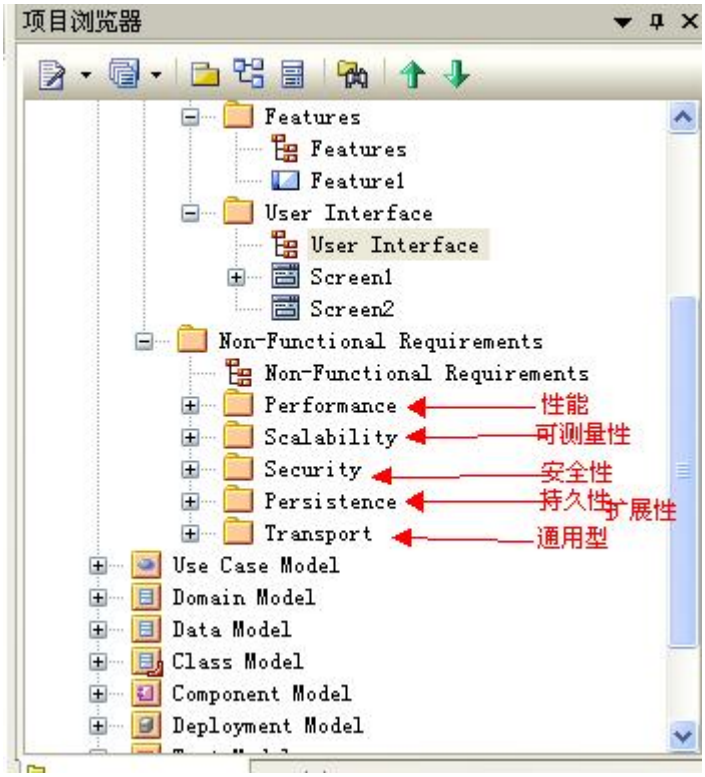
分包，同时记录用户的原始需求，这里模型中有好的一个分法，把需求从特性、规则和界面要求分开了



各个层用法（其实这里的规则和特性，概念也很模糊，通常我用的时候上面的包为 Requirement 只是记录需求，相应的规则和特性都记录在 Features 当特性表的内容）



以下是非功能性内容



然后，对每个元素和需求中填写需求的描述

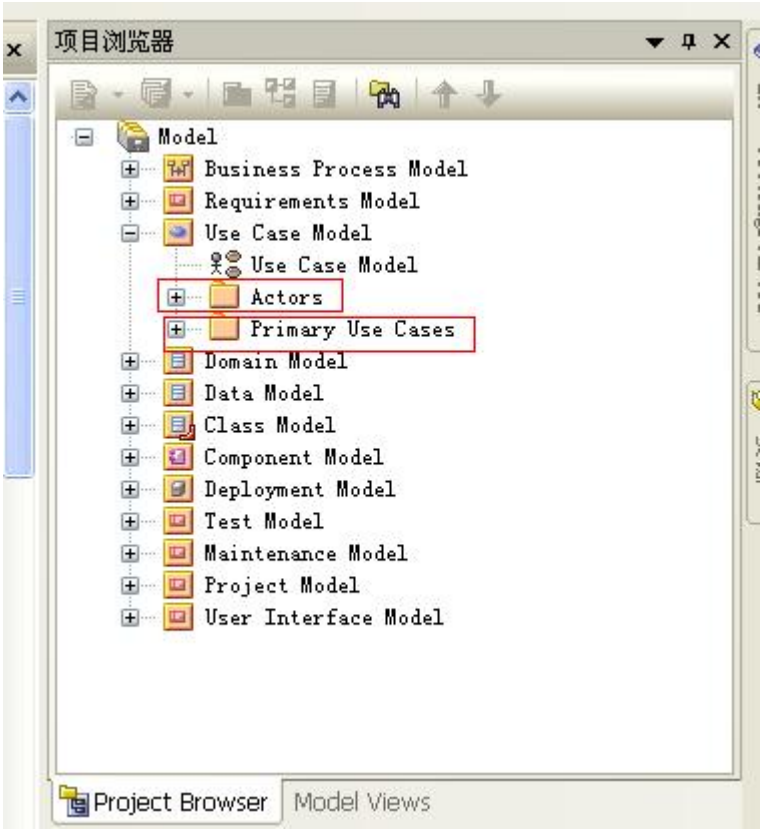
以上的工作只是收集原始需求的工作，是现场或和客户沟通、接触的最直接工作以及“证据”，同时也是为了下一步的分析的根据基础，接下来是体现系统分析师的水平的工作，用例以及用例分析；

五、建顶层用例

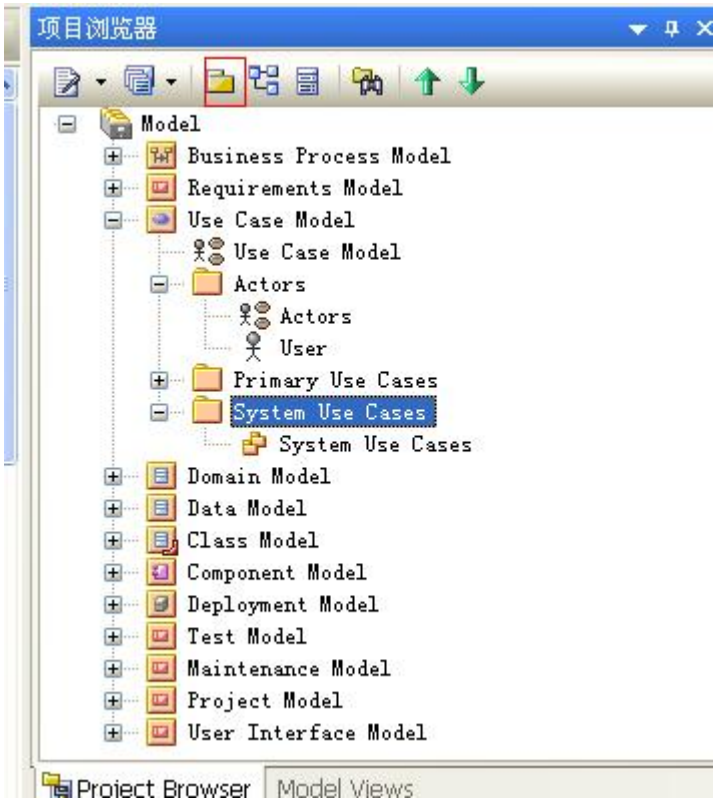
用例可分顶层用例、业务用例和系统用例（这是我自己的分法，没有教程这么讲过，只需自己理解就好，其实业务用例和系统用例可以当作一些书本中说的客户需求和系统需求），以上分法目的是：

首先，从和客户的沟通和接触中，你可能会收集到很多早期的用例，特别是一开始会是用户的最初要求，也是最大的期望，通常这些都是可以归类到顶层用例中，然后根据这些顶层用例和收集的需求，根据你的理解，以其行业（看你的系统是做什么行业）的术语和业务进行分解和细化形成业务用例，这是整理以及细化的过程，可以至顶向下，也可以由细整理再归类，最后形成业务用例；同时，根据你 IT 的经验，把业务用例进行分析（这是见你分析设计经验的时候，架构师通常的能力就表现出来了），形成可开发化的系统用例，这过程是个分析的过程，有可能一个业务用例会被你分拆成多个用例，也有可能多个业务用例合并成一个系统用例，总之，就是系统优化的那些原则，性能、可扩展性、安全性、通用型等等什么的。这些内容调研之后的首要工作，这些可以同步迭代进行。

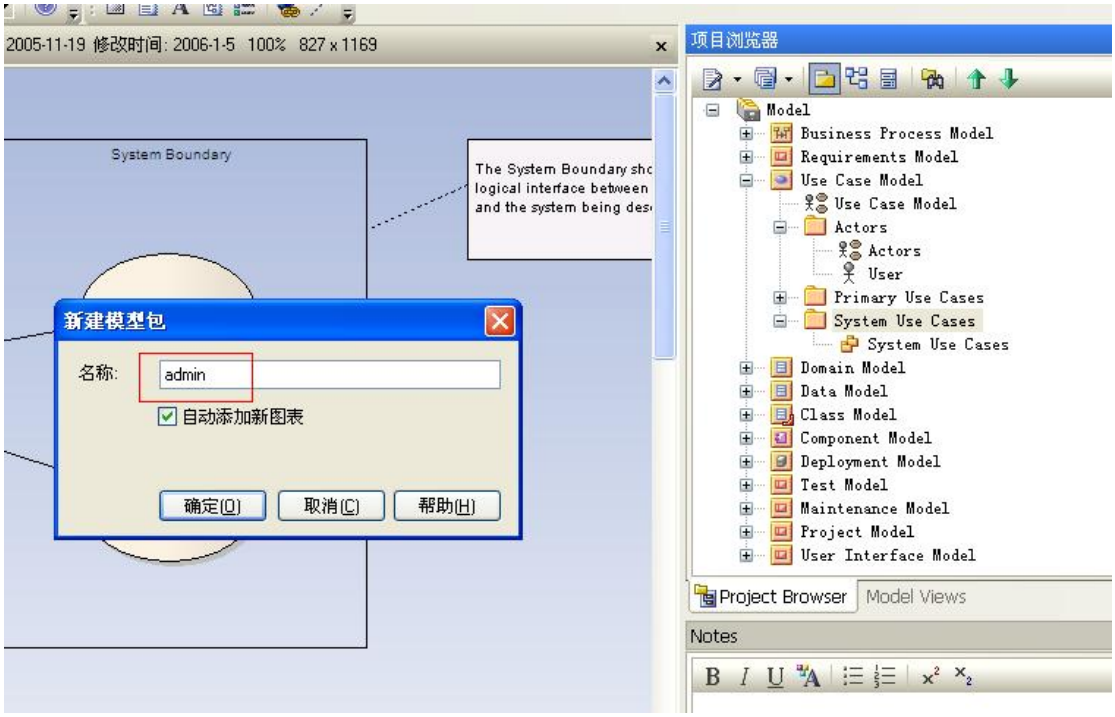
用例图是角色和用例之间的关系，所以通常做法我会给角色单独建一个包，然后用例根据边界的分法建包：



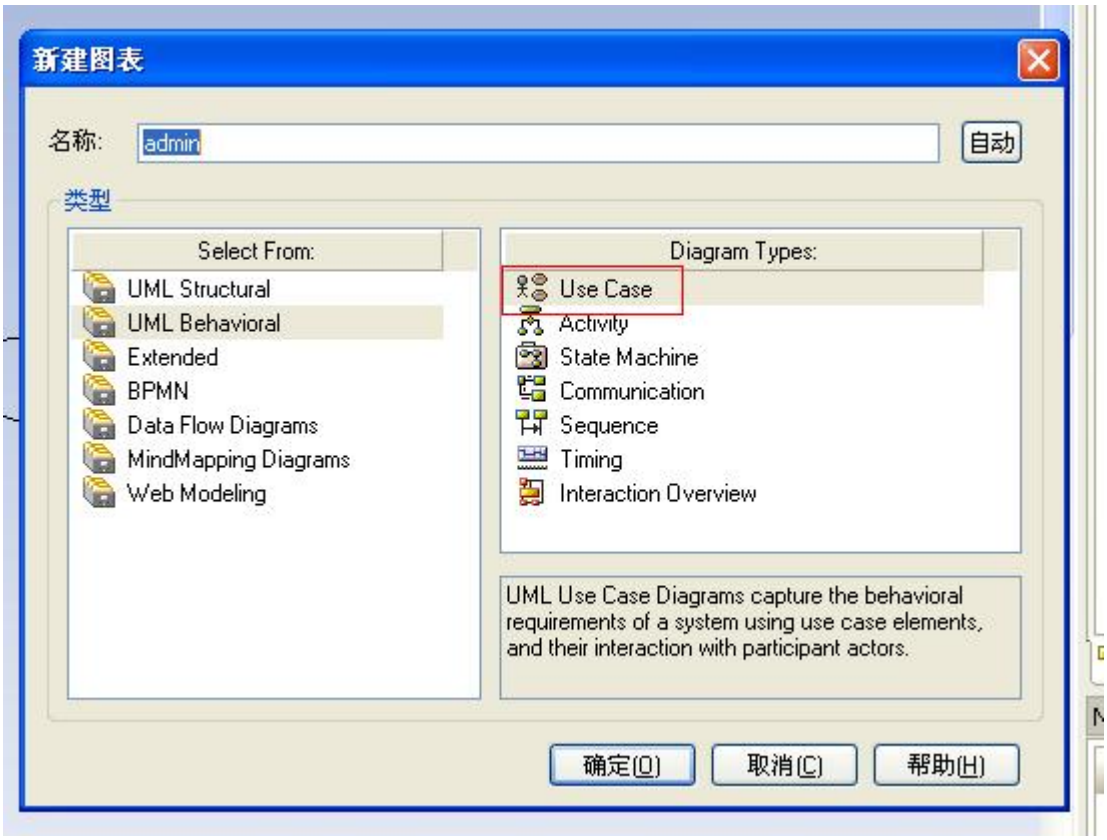
建立相关的包



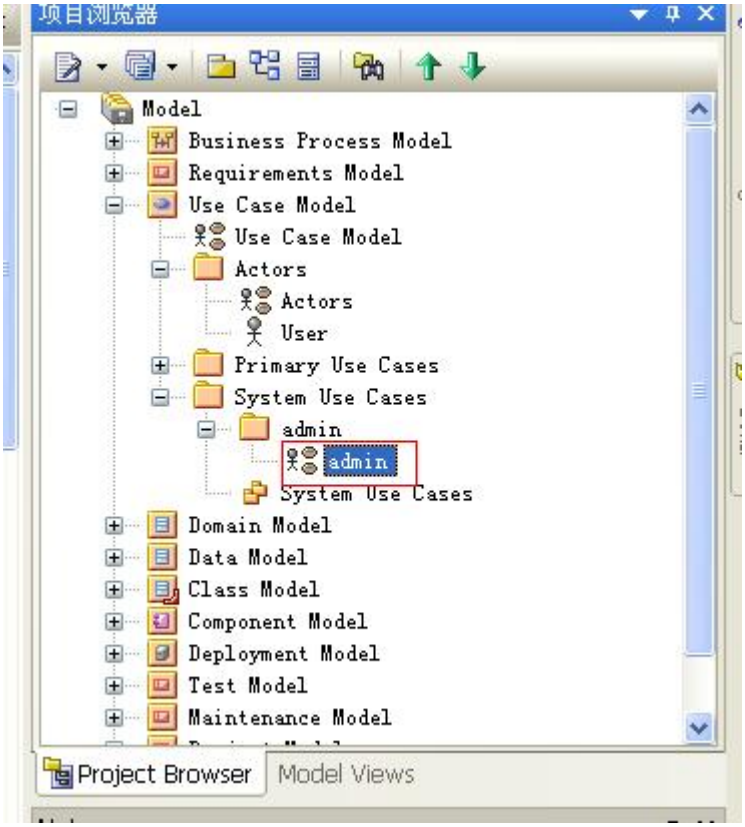
输入包名



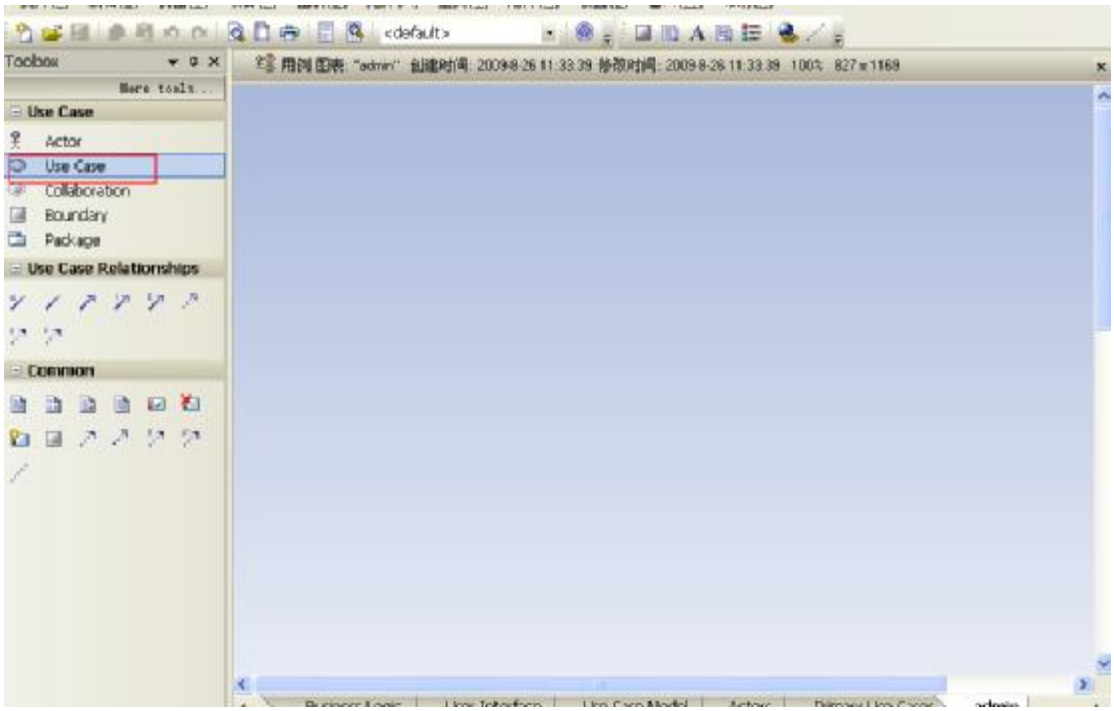
根据你这个包所扮演的作用选择相关视图，（一个包中可存在多个视图，可用前面说的视图按钮建立）



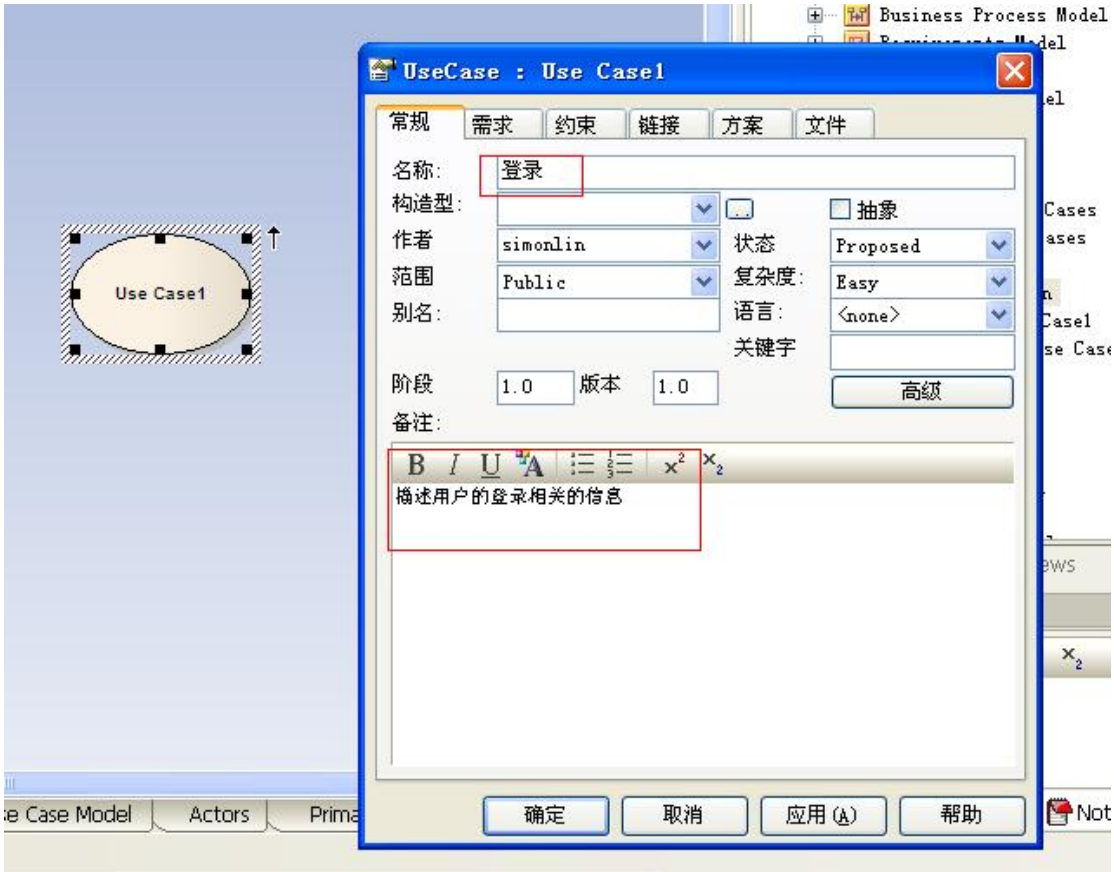
双击刚建好的用例视图



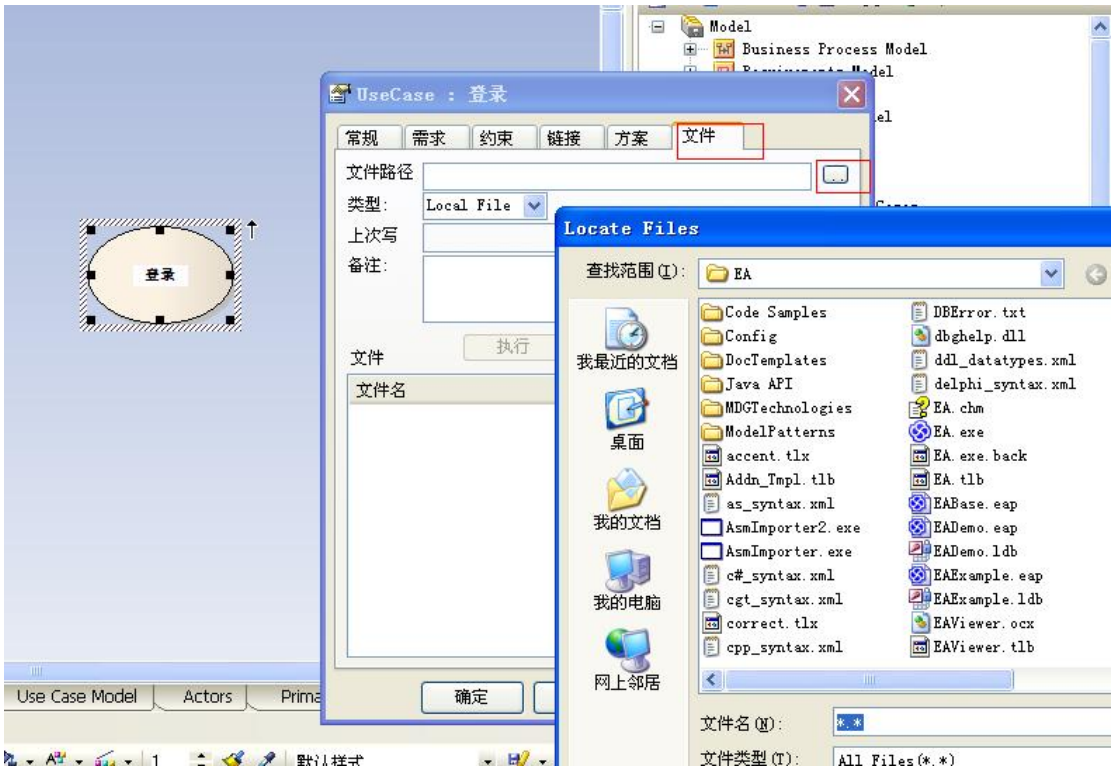
点击左边的元素集，选择用例元素



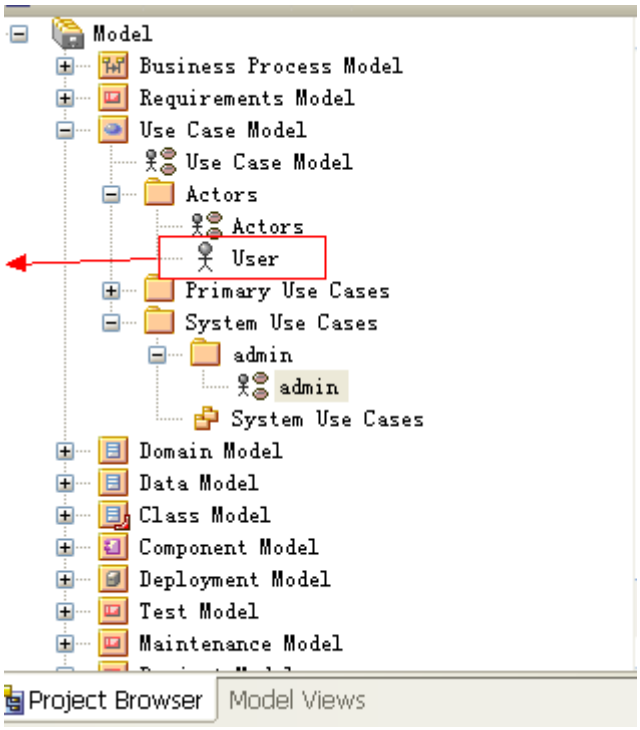
放置到用例视图中，填写用例名和用例描述以其其他相关的属性



或者是和此用例相关的文件

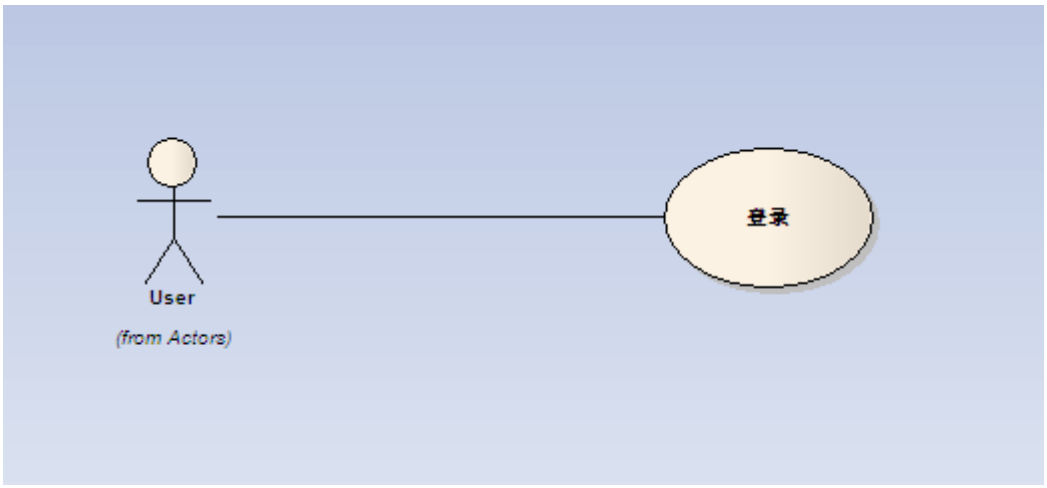


把相关的角色拖到视图中

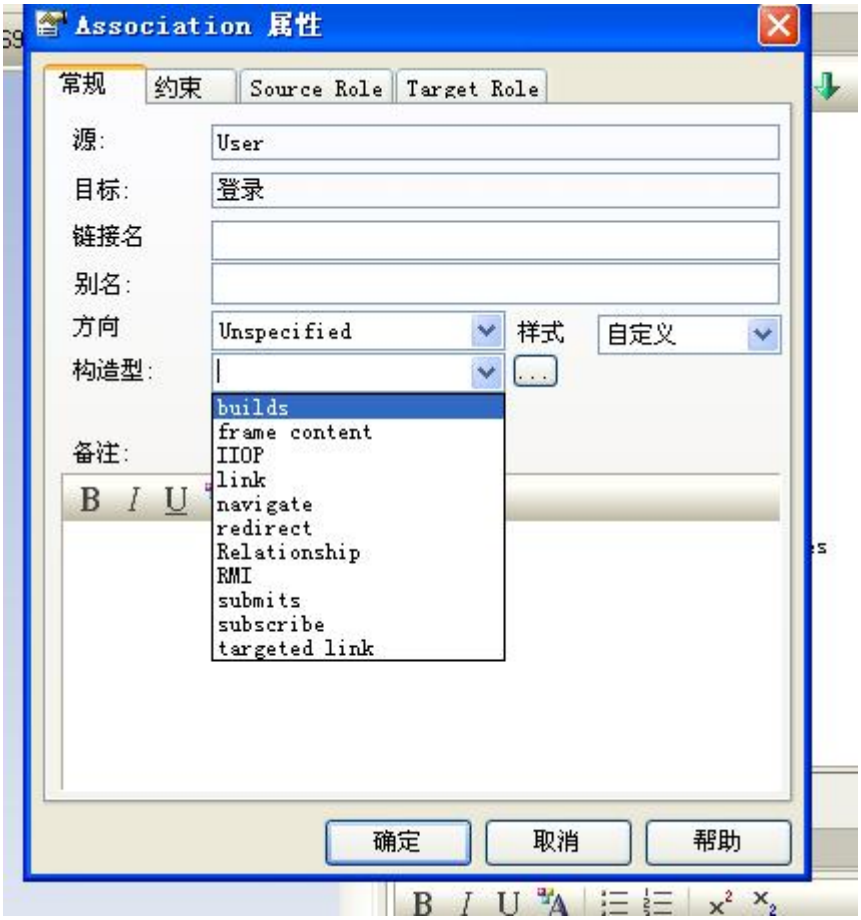


在提示框中选者简单的连接

画上角色和用例的关联

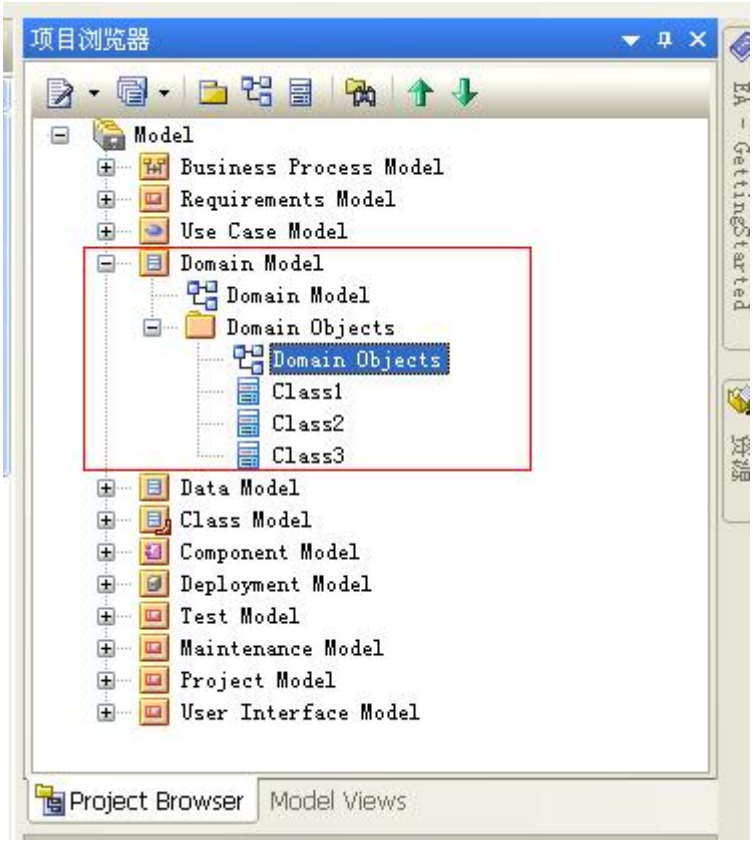


其关联还可以通过双击关联线，标注相关的构造型和链接名



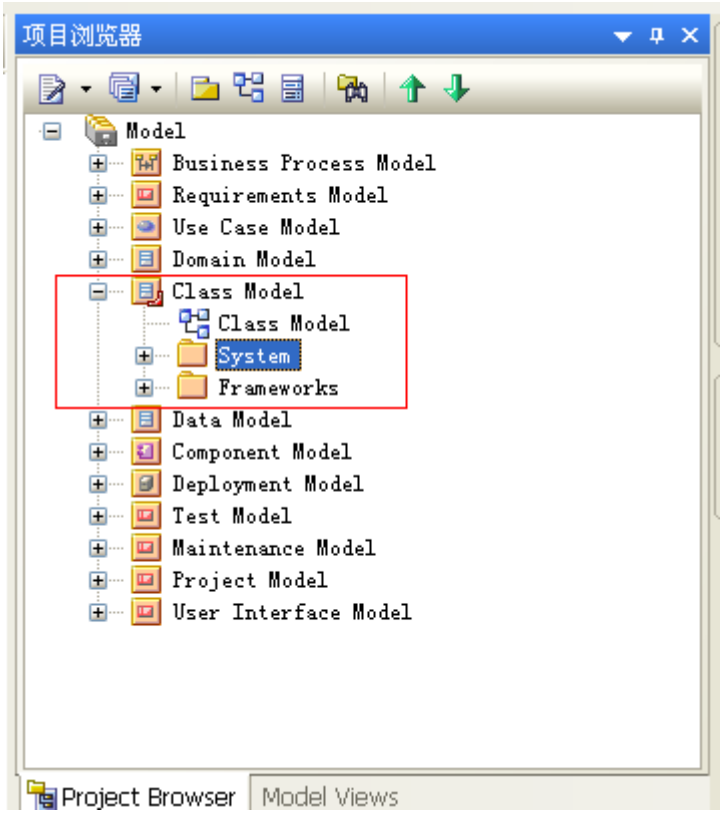
六、进行领域模型建模

看过《领域驱动设计》的人就知道领域模型的作用是什么了，主要是面向用户，面向业务的，顾名思义就是理解领域中的各层关系，其实画法和类模型相似，这里就不细说了，并不是必须的，看项目具体情况而定。

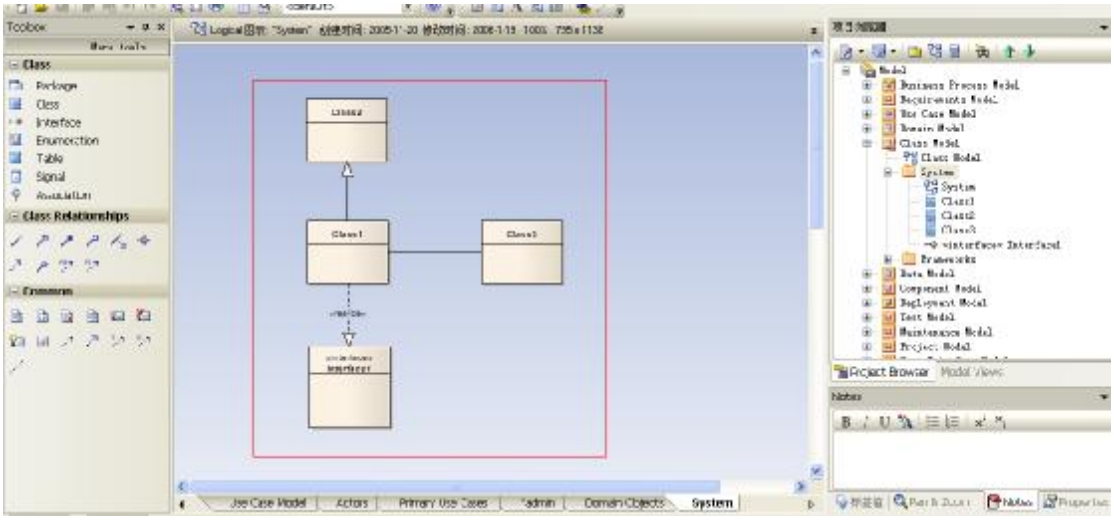


七、建类图

案例中做了个好的分法，把系统类和架构类分开了，大家可以照搬，当然也可以根据自己需要自己建包。

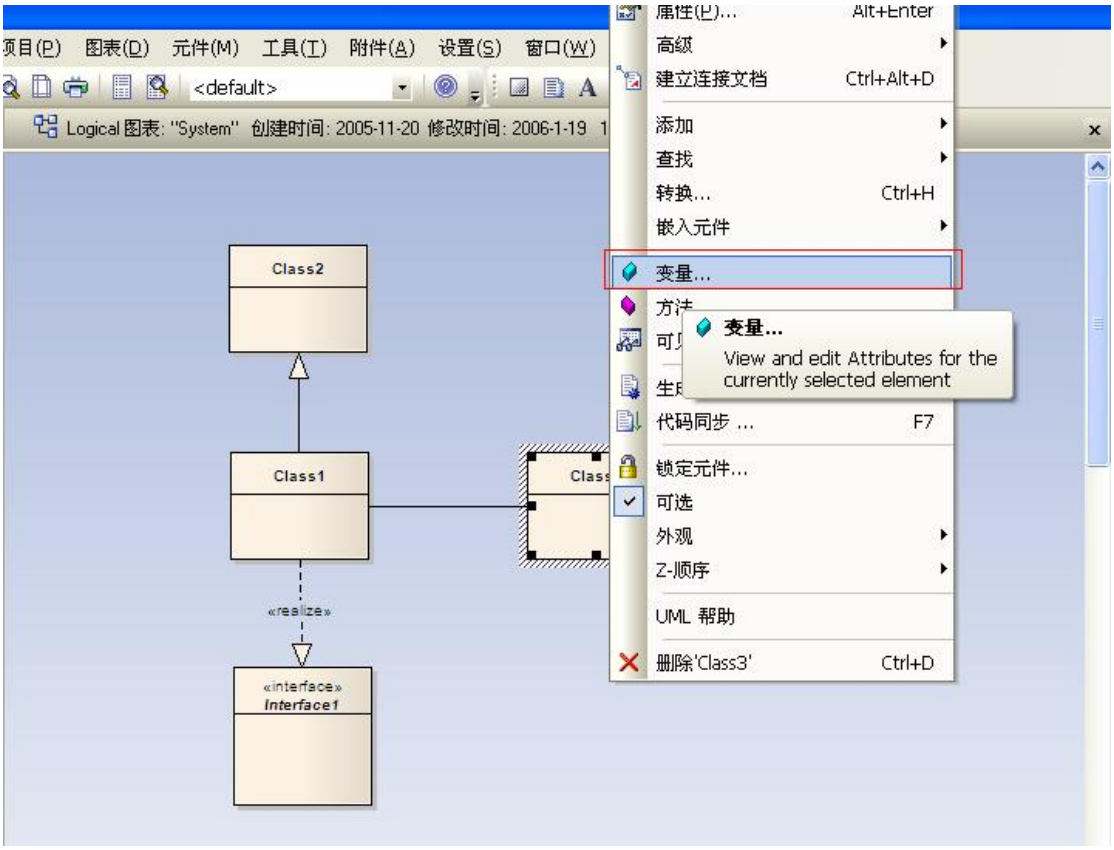


做法也是和之上画用例图的步骤一样的（重要的是分析设计的思想）：建包—建视图---画元素（用例、类、表等）--画关系（关联、继承、依赖等）

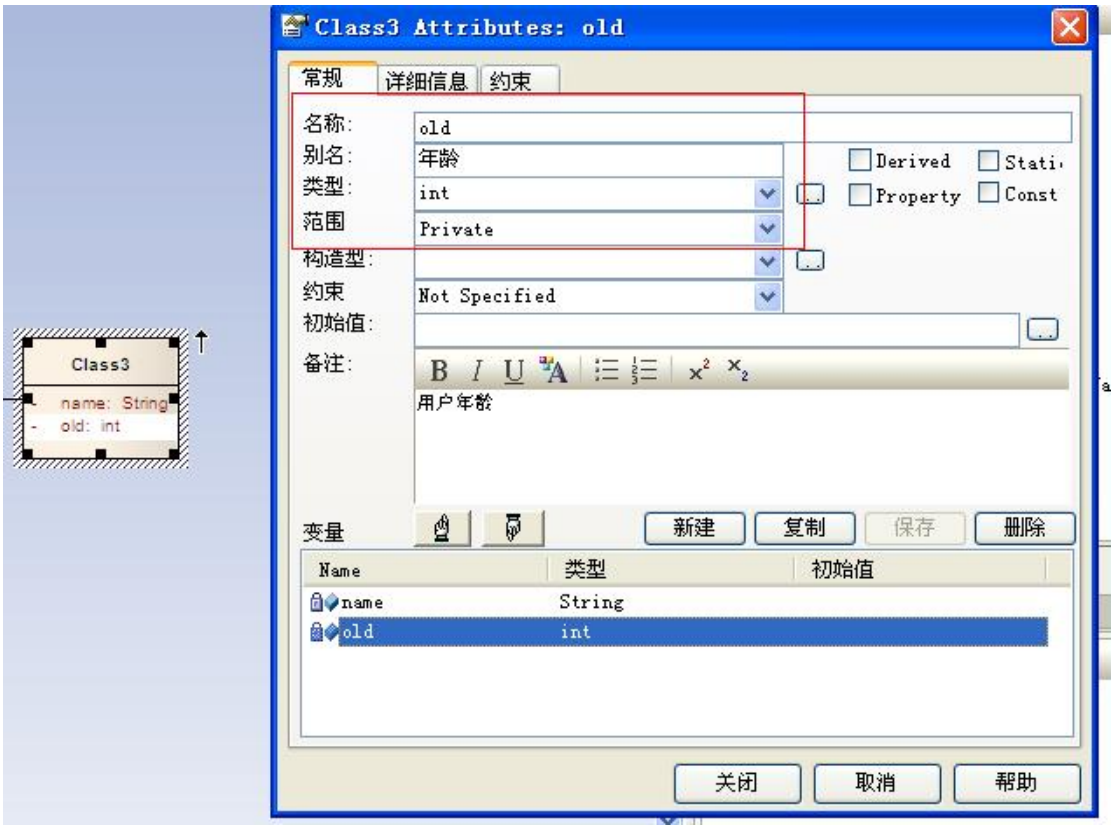


一个类它可以有属性和方法

建类的属性

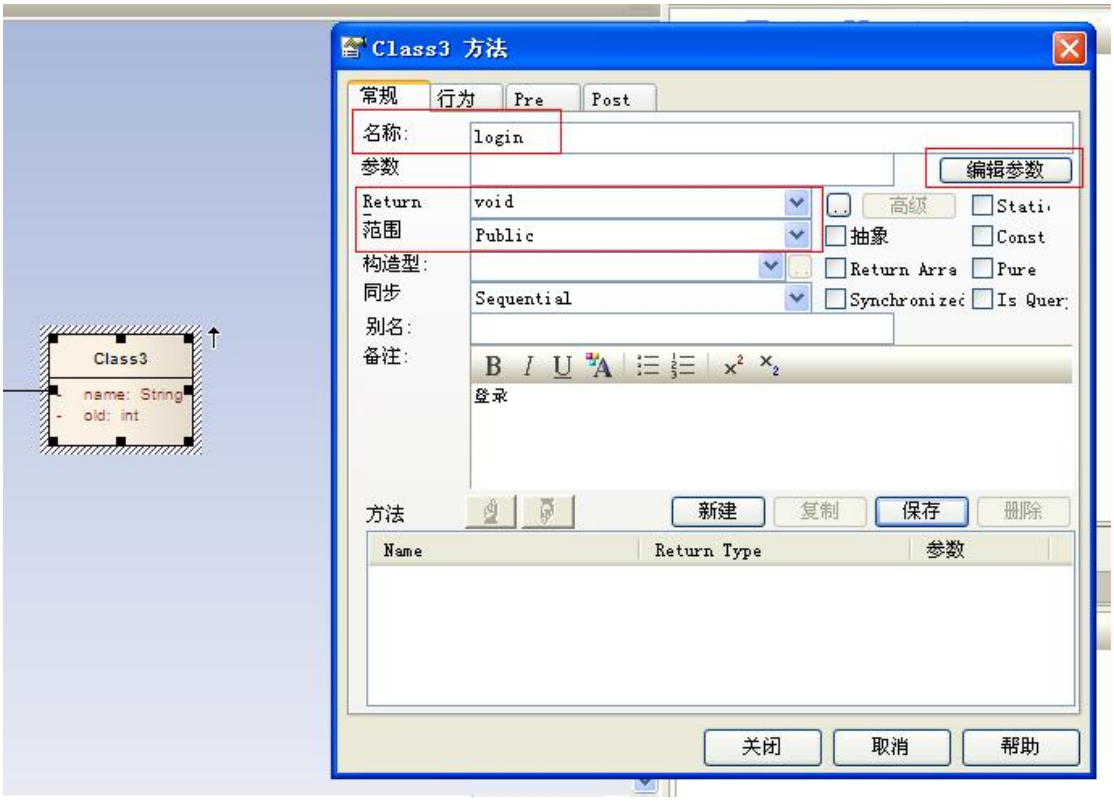


输入属性（变量）的内容



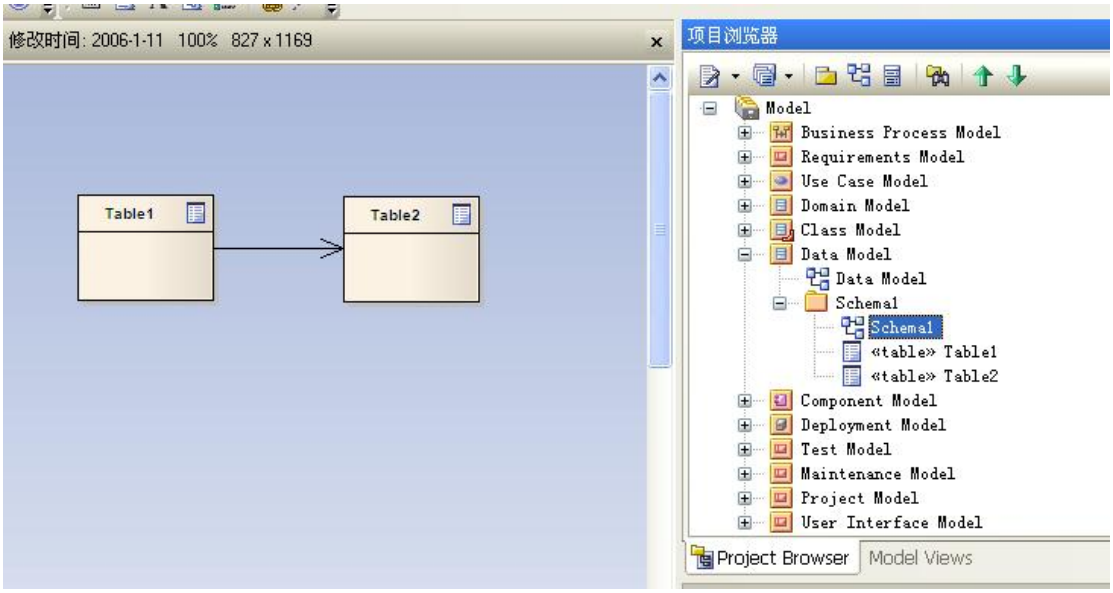
建类的方法

输入方法的内容



八、建数据模型

手动建数据模型的方法和以上内容基本相似



建字段和方法的方法和类图一样（不过通常通过类转换过来的话，这些内容都有了，只是对其进行一些修正即可）

九、通过包转换建立类模型、数据模型

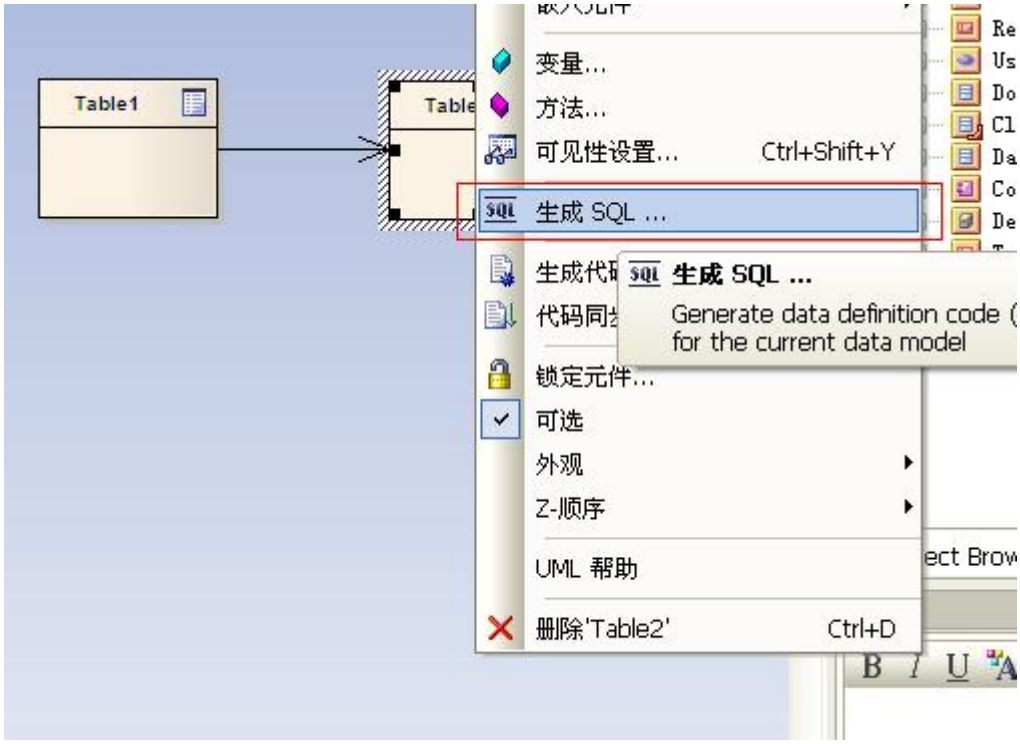
领域模型、类模型、数据模型这几者之间是可以通过包转换来进行的，也就是如果你原来建好的领域模型，可以通过领域模型转换到类模型，从类模型转换到数据模型，这是可以减少很多的工作量，而且是可以承接先前的思想。

选中需要转换的包，右键

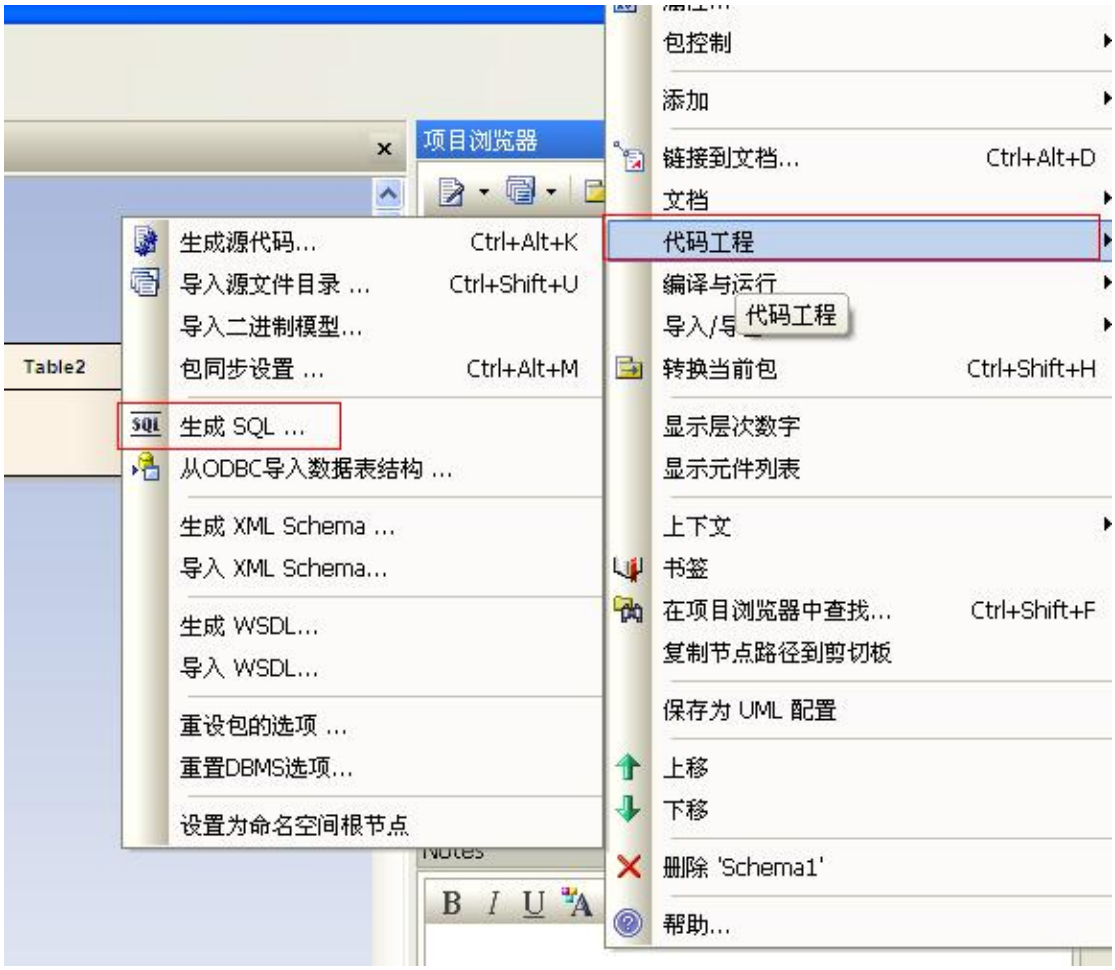
根据你要转换的目的内容选择

十、通过数据模型生成 SQL

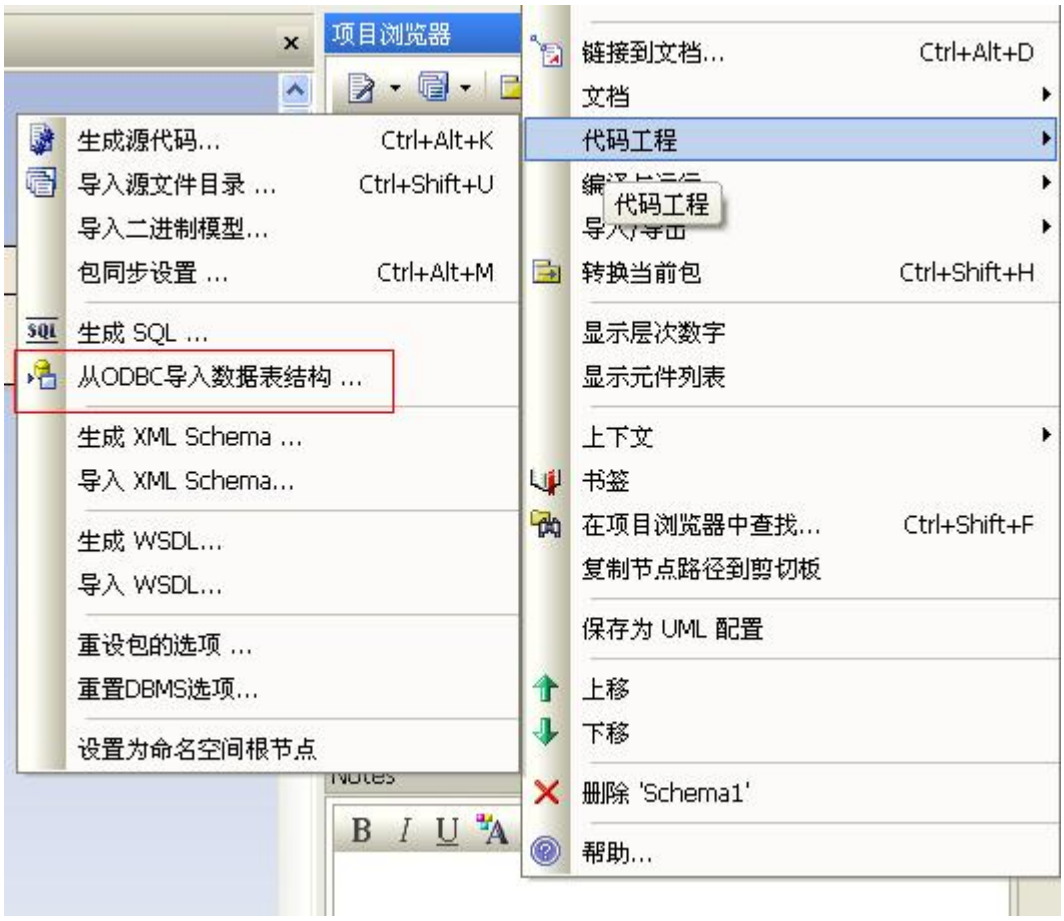
选单个表生成 SQL



选整个包生成 SQL



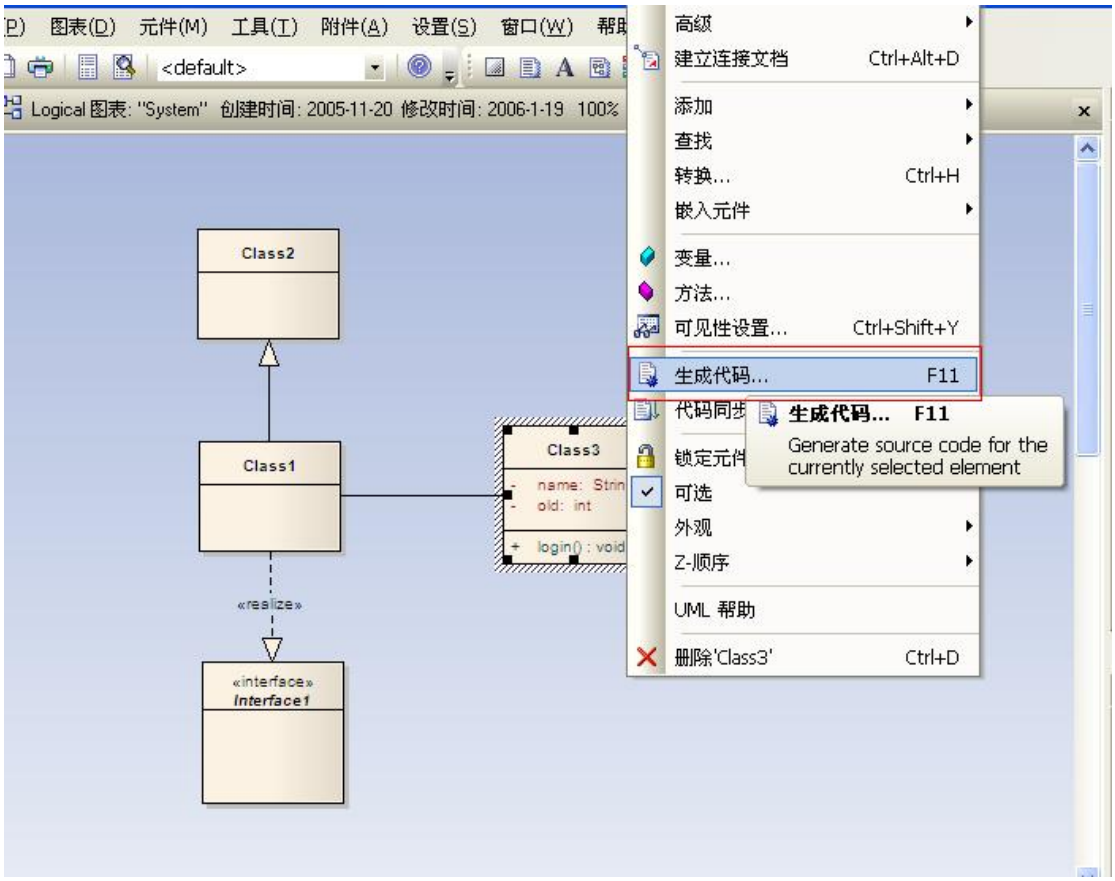
从原有数据库中导入表结构



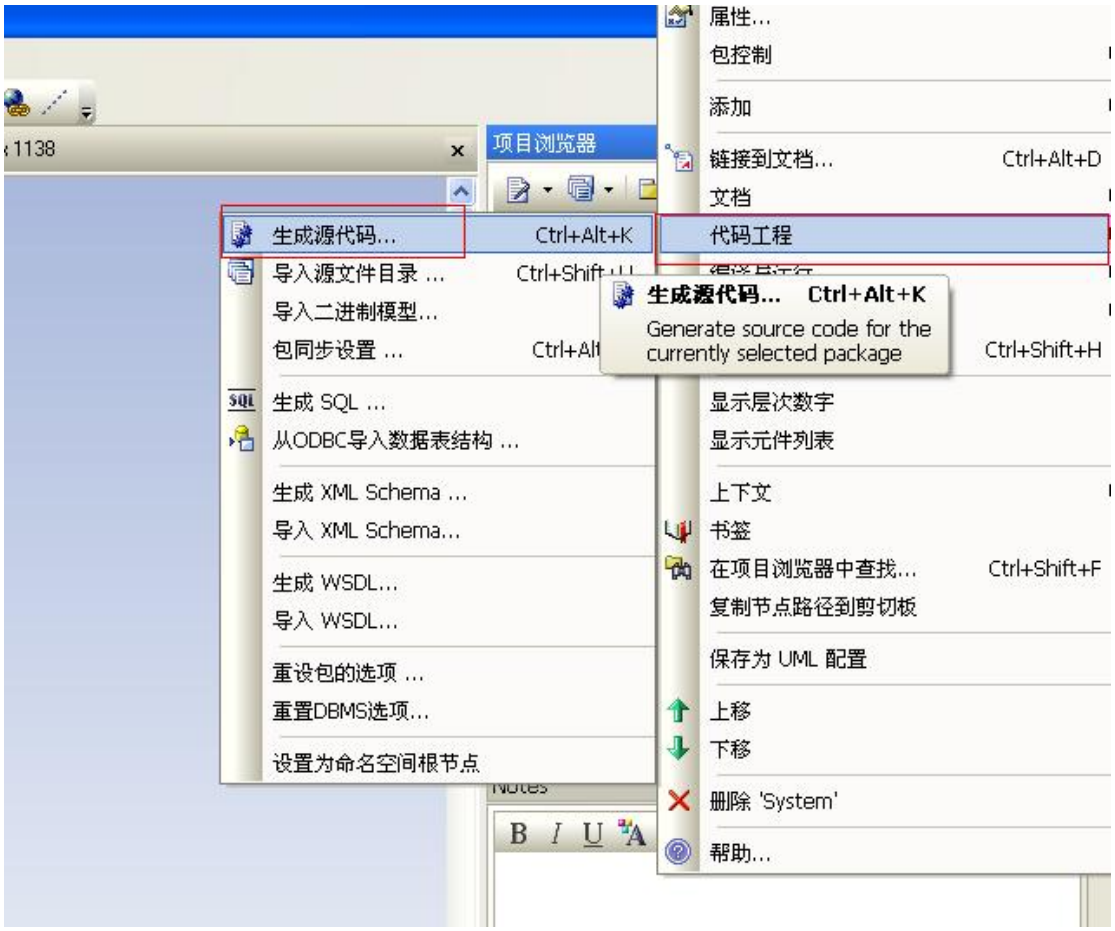
十一、 EA 进行正反向工程

用 EA 进行生成源代码暨正向工程

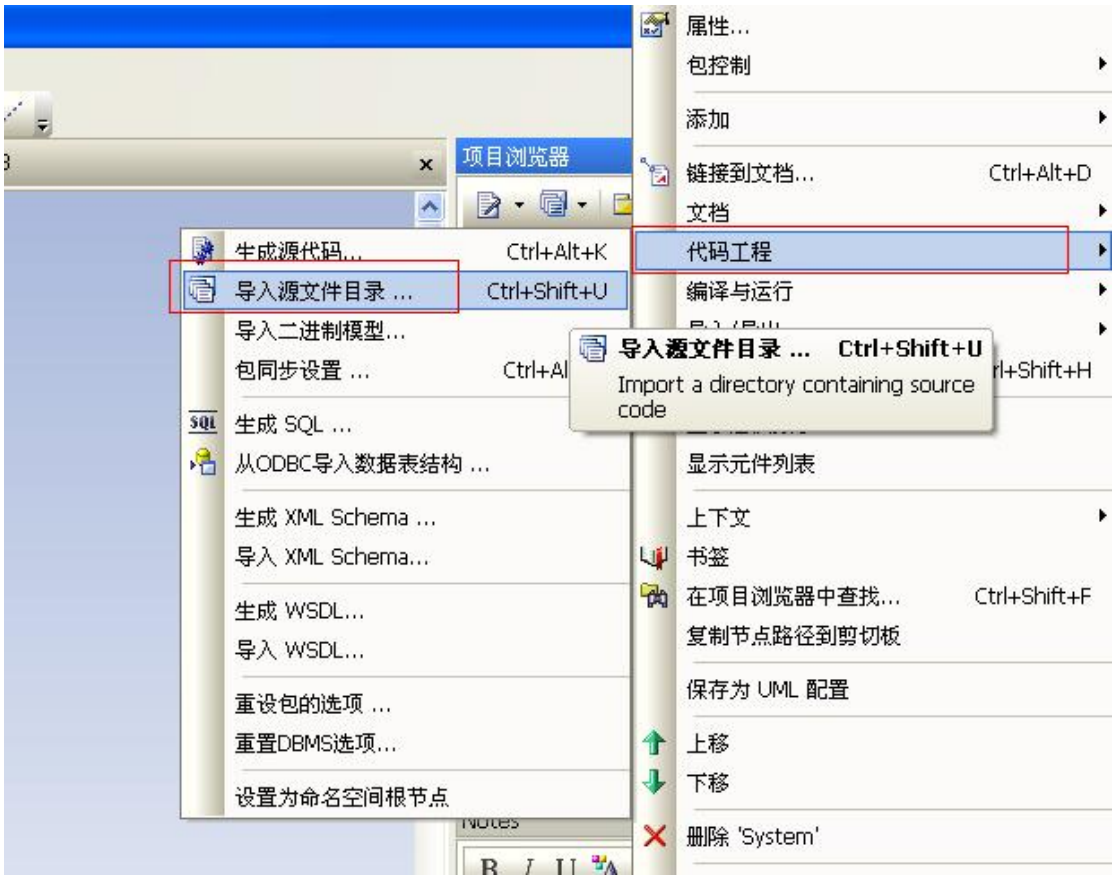
单个类生成源代码


图表(D)
元件(M)
工具(T)
附件(A)
设置(S)
窗口(W)
<default>
Logical 图表: "System" 创建时间: 2005-11-20 修改时间: 2006-1-19 100%
Class2
Class1
Class3
name: Strin
old: int
login() : void
«realize»
«interface»
Interface1
高级
建立连接文档 Ctrl+Alt+D
添加
查找
转换... Ctrl+H
嵌入元件
变量...
方法...
可见性设置... Ctrl+Shift+Y
生成代码... F11
代码同步
锁定元件
可选
外观
Z-顺序
UML 帮助
删除'Class3' Ctrl+D
生成代码... F11
Generate source code for the currently selected element

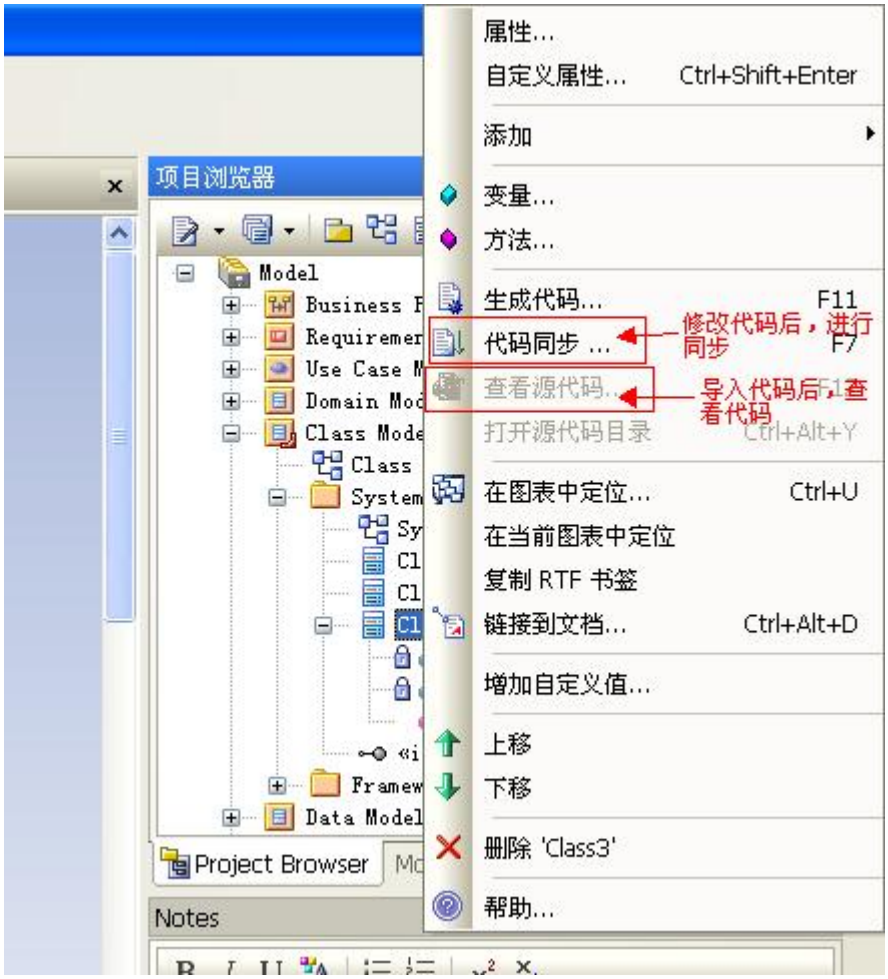
整个包生成源代码


项目浏览器
属性...
包控制
添加
链接到文档... Ctrl+Alt+D
文档
代码工程
生成源代码... Ctrl+Alt+K
导入源文件目录 ... Ctrl+Shift+U
导入二进制模型...
包同步设置 ... Ctrl+Alt
生成 SQL ...
从ODBC导入数据表结构 ...
生成 XML Schema ...
导入 XML Schema...
生成 WSDL...
导入 WSDL...
重设包的选项 ...
重置DBMS选项...
设置为命名空间根节点
显示层次数字
显示元件列表
上下文
书签
在项目浏览器中查找... Ctrl+Shift+F
复制节点路径到剪切板
保存为 UML 配置
上移
下移
删除 'System'
帮助...
生成源代码... Ctrl+Alt+K
Generate source code for the currently selected package
Ctrl+Shift+H

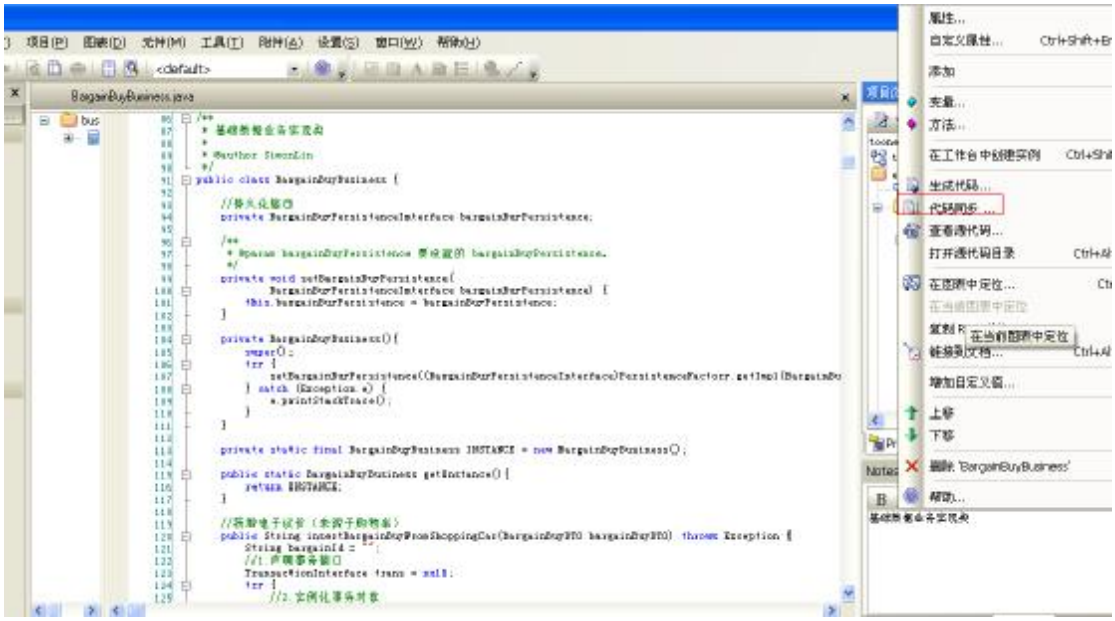
用 EA 导入源代码暨反向工程



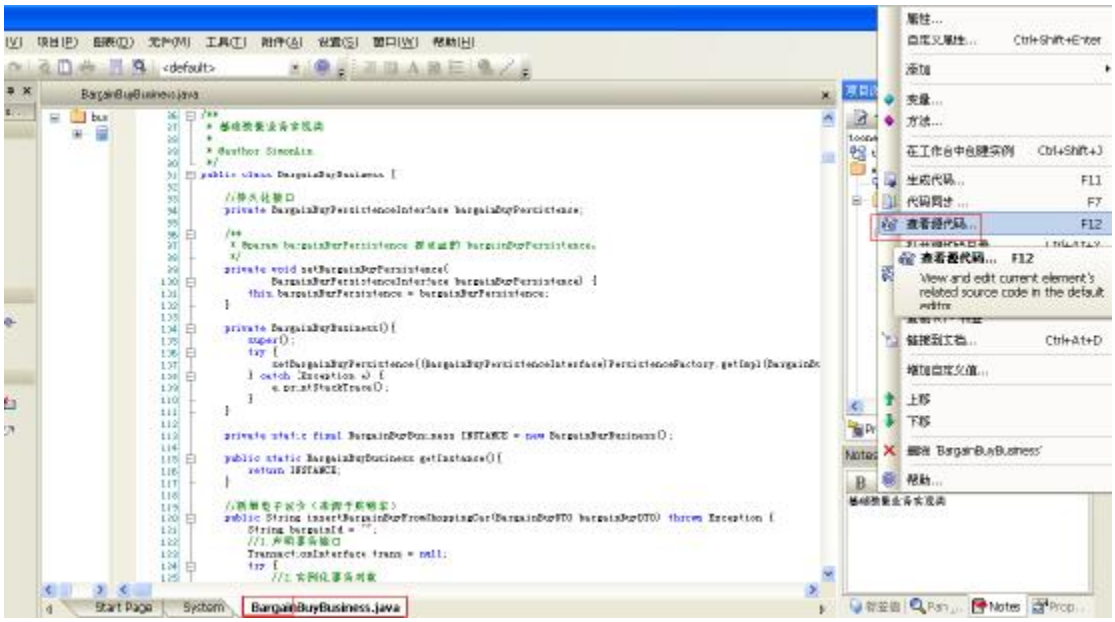
查看源代码以及修改了源码后进行代码同步



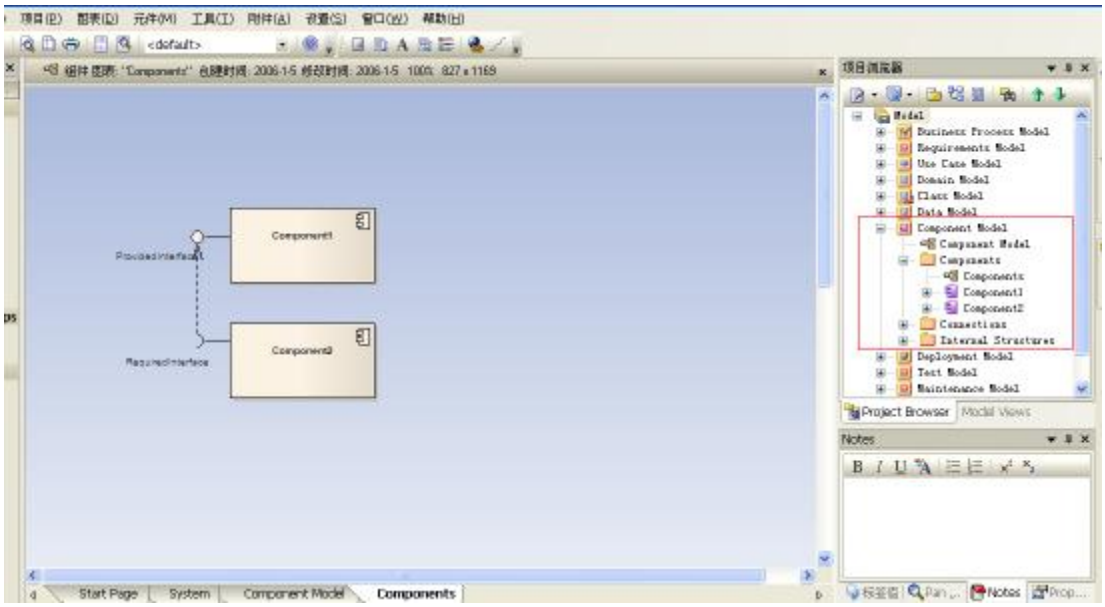
在 EA 中修改了源码以后，可以进行代码同步



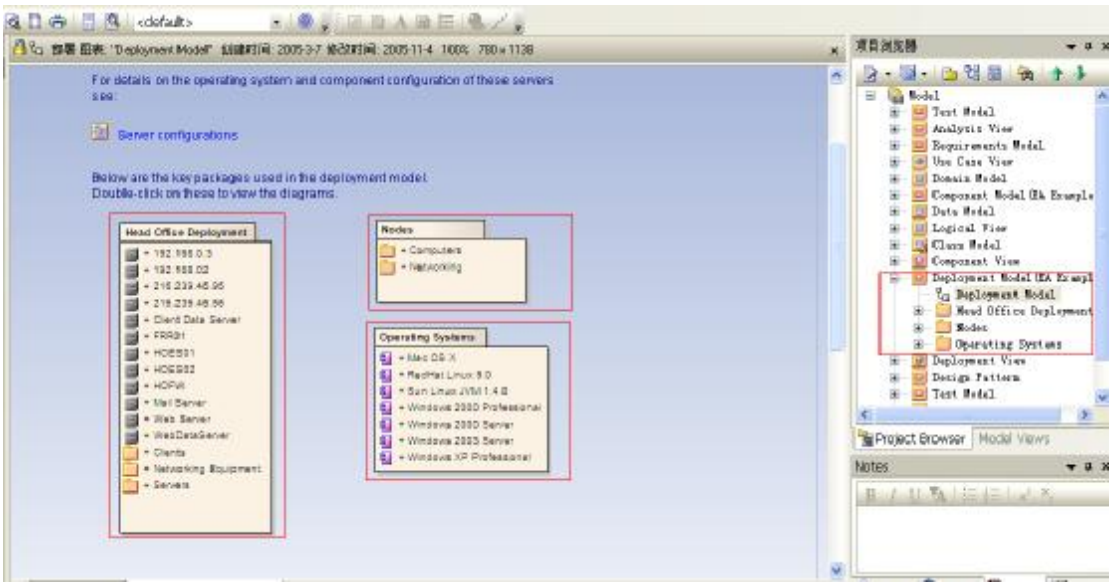
十二、 在 EA 中进行编码，当 IDE 使用（就像 eclipse）



十三、 组件模型

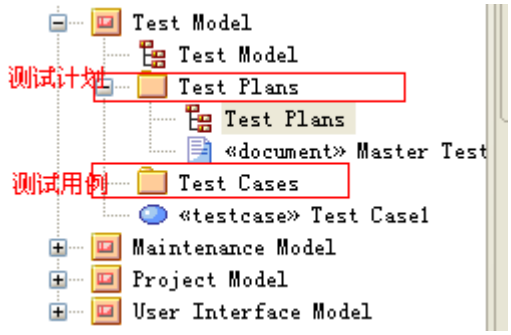


十四、 部署模型

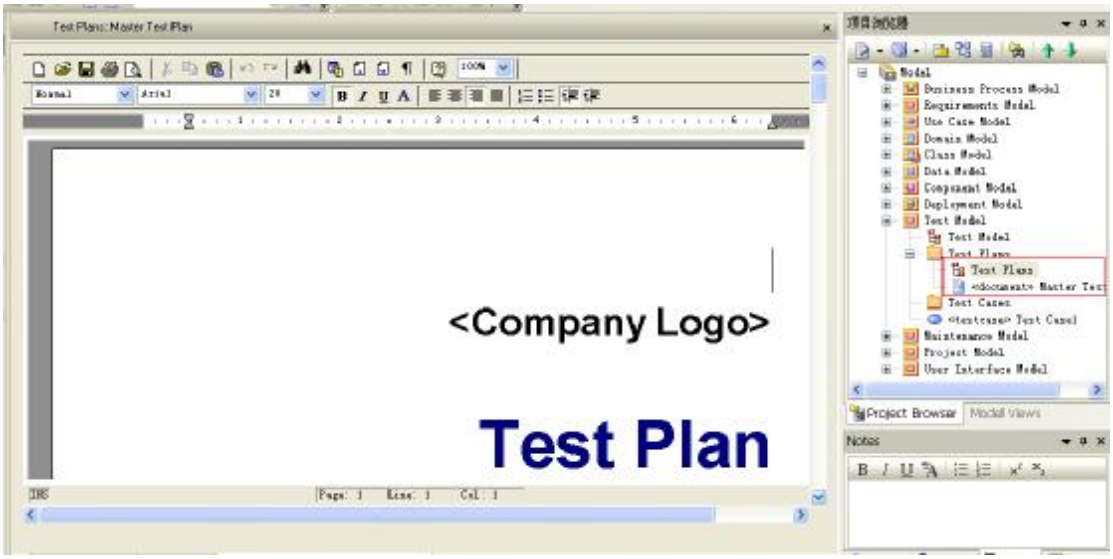


十五、 EA 中管理测试

可以在 EA 中管理测试计划和测试用例，首先可以以这分包

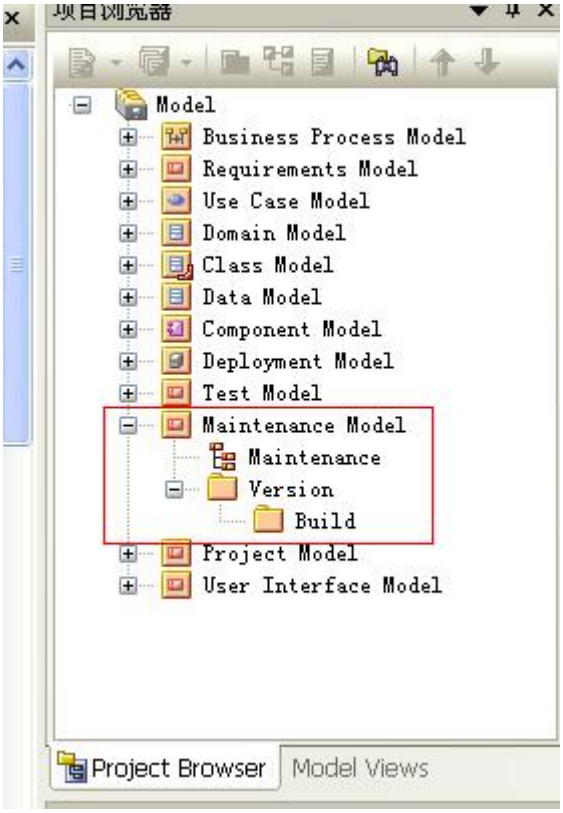


测试计划模板



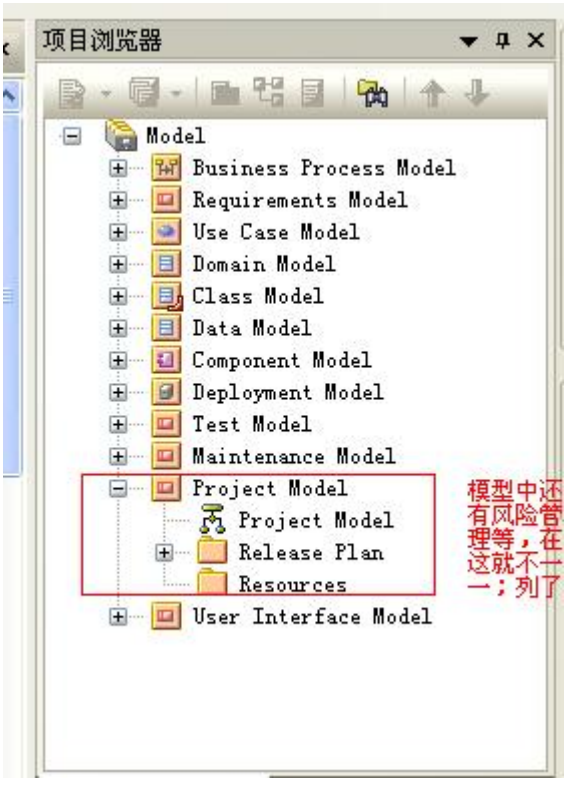
十六、 系统维护

软件管理当然少不了系统的维护，系统变更管理，EA 中同样含有维护模型，可以在这里进行版本及其版本中变更的管理

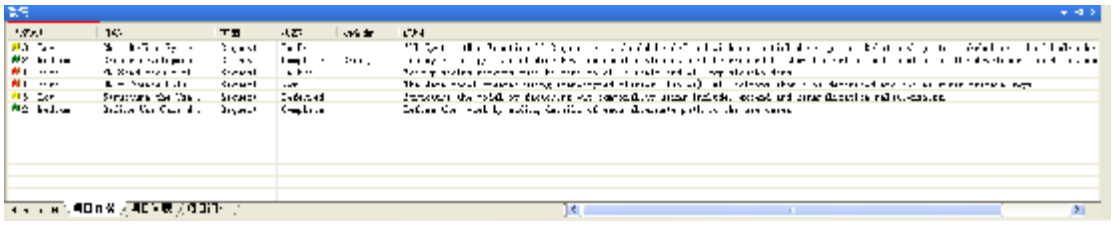


十七、　进行项目管理

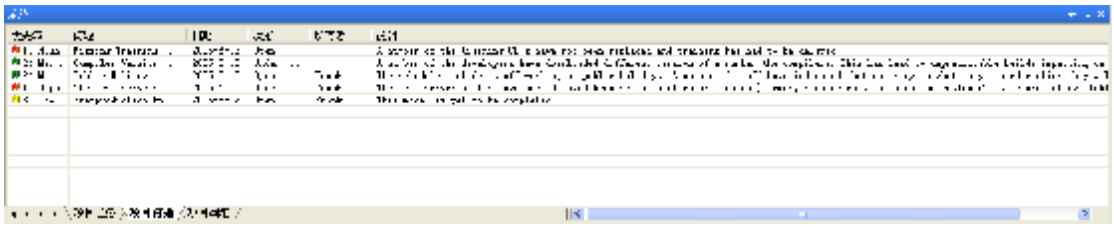
在 EA 的项目管理中还有计划管理、资源管理、风险管理等方面，和软件工程的 UML 或 ISO 的管理比较贴切



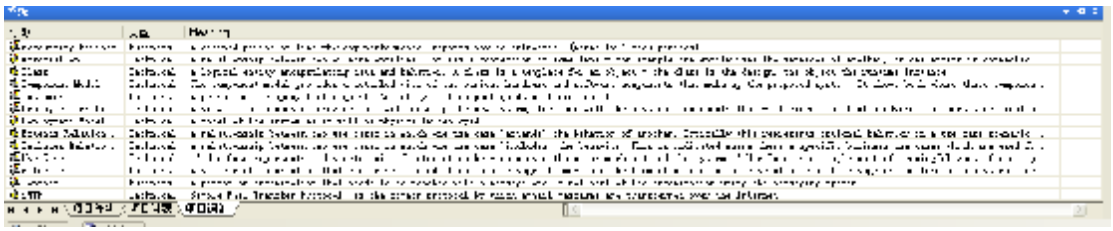
项目任务管理



项目问题管理



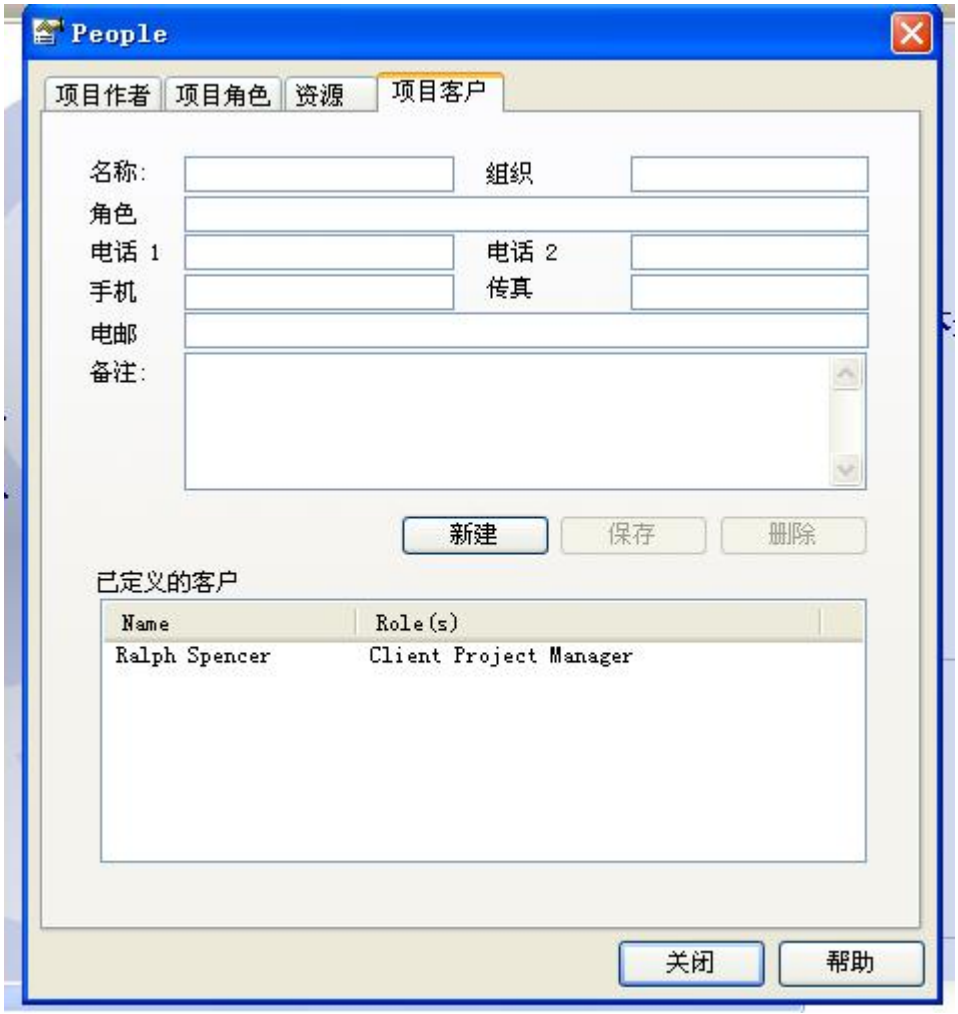
相关词汇表



管理干系人

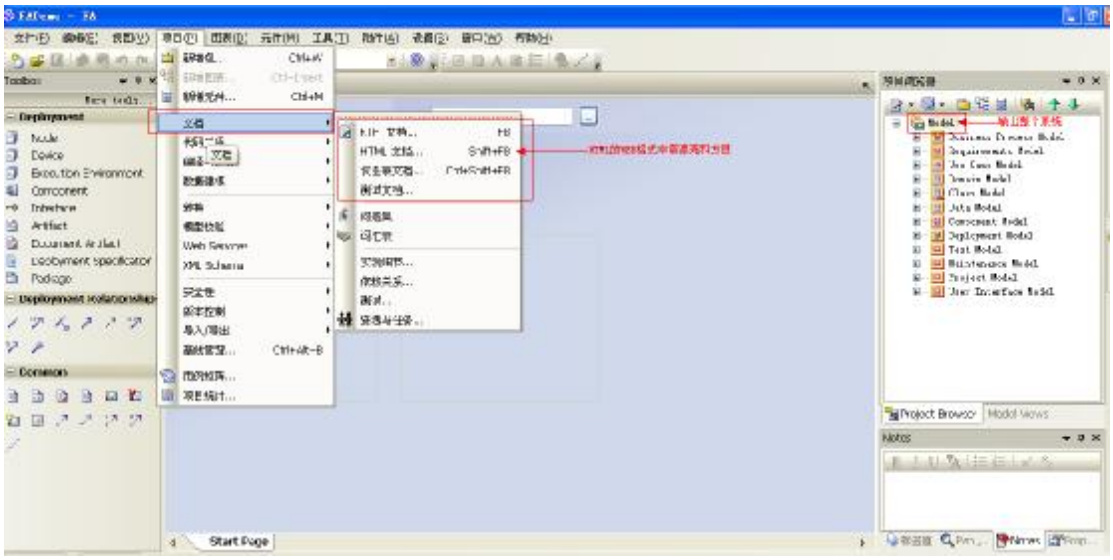

设置(S)
窗口(W)
帮助(H)
人物...
类型...
维护...
项目选项配
因素与评估配置...
UML...
MDG 配置...
命名空间...
模板目录...
路径定位...
元件自动命名配置...
语言定义...
数据库定义...
宏定义...
代码工程模板...
Ctrl+Shift+P
源代码模板...
Ctrl+Alt+H
图像...
颜色
人物...
Define the people who impact
current project

如：

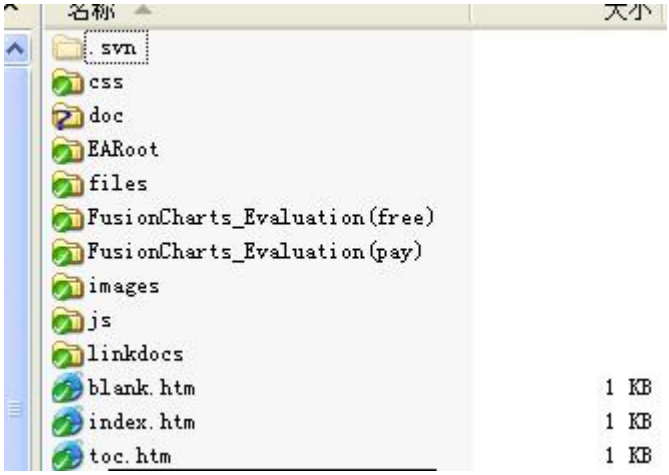


十八、　生成需要文档

用户可以通过生成文档生成自己所需要的问题，文档的格式可以使用 EA 默认的，也可以通过修改模版按自己的要求输出文档。



例子：

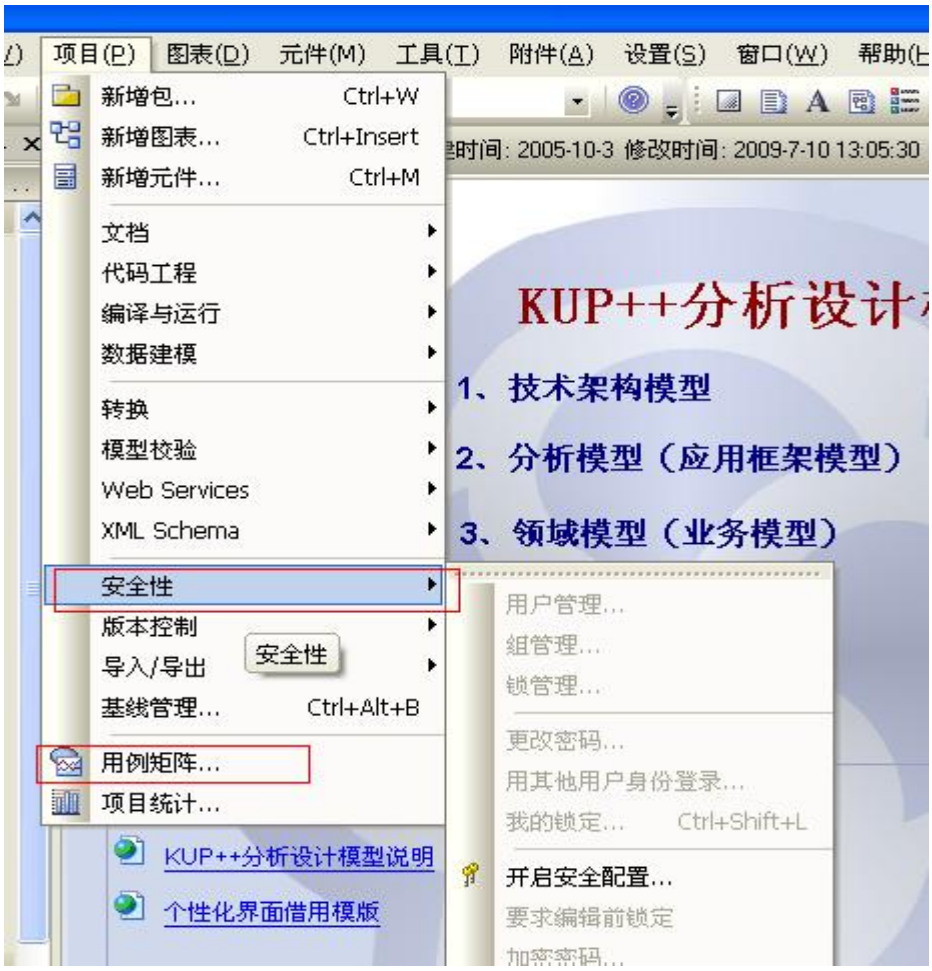


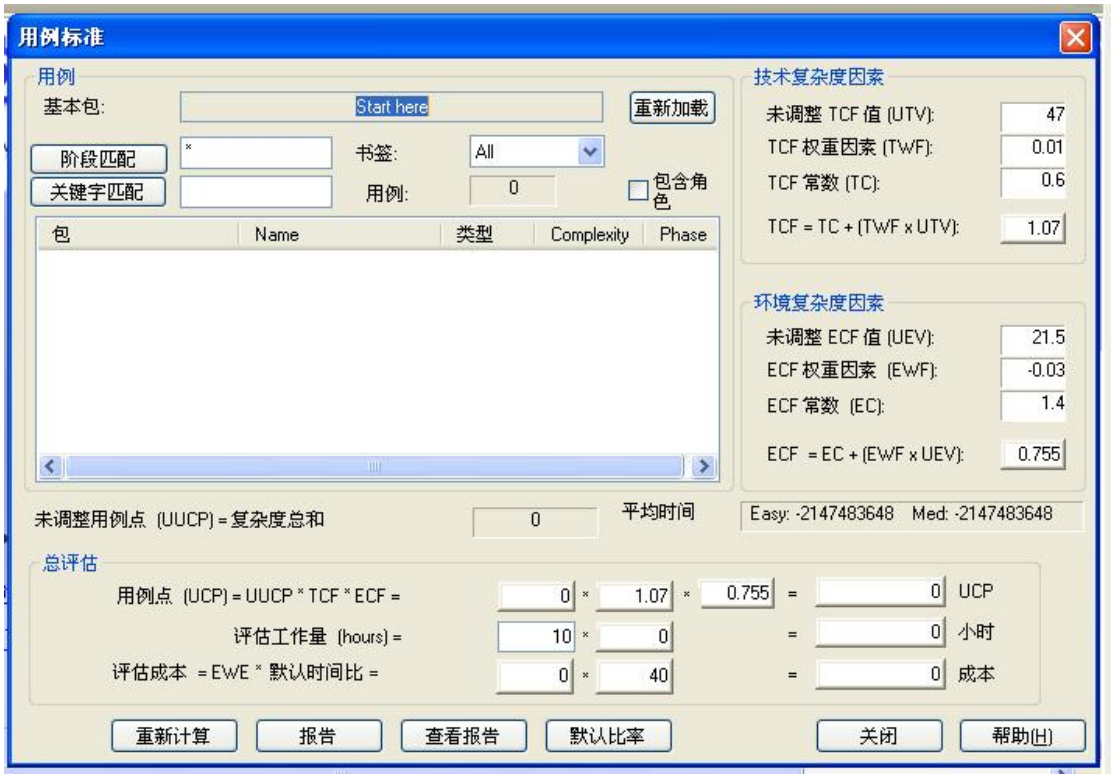
十九、 可对项目元素进行统计

二十、 还有一些不知怎么用的功能**J**

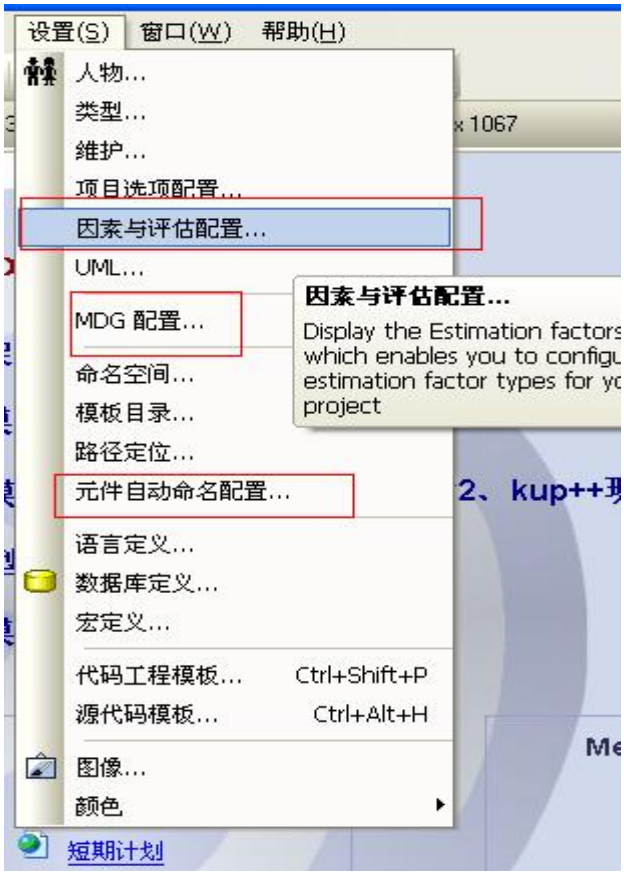

编辑(E)
视图(V)
项目(P)
图表(D)
元件(M)
工具(T
项目浏览器 Alt+0
属性 Alt+1
系统 Alt+2
测试 Alt+3
维护 Alt+4
工具箱 Alt+5
资源 Alt+6
源码编辑器 Alt+7
调试工作台 Alt+8
任务面板 Ctrl+Shift+9
备注 Ctrl+Shift+1
层次结构 Ctrl+Shift+4
标签值 Ctrl+Shift+6
项目管理 Ctrl+Shift+7
输出 Ctrl+Shift+8
模型视图 Ctrl+Shift+5
更多窗口
工具栏
元件列表 Ctrl+Alt+R
搜索模型 Ctrl+Alt+A
关系矩阵
联机讨论 Ctrl+Alt+U
审计视图
显示网格线
调整网格线
风格
布局
Welcome


项目(P)
图表(D)
元件(M)
工具(T)
附件(A)
设置(S)
窗口(W)
帮助(H)
新增包... Ctrl+W
新增图表... Ctrl+Insert
新增元件... Ctrl+M
文档
代码工程
编译与运行
数据建模
转换
模型校验
Web Services
XML Schema
安全性
版本控制
导入/导出
基线管理... Ctrl+Alt+B
用例矩阵...
项目统计...
校验当前... Ctrl+Alt+V
停止校验
配置...
时间: 2005-10-3 修改时间: 2009-7-10 13:05:30 100%
KUP++分析设计模型
1、技术架构模型
6、
模型)
附录
附录
4、类模型
5、数据模型
Links
HTML Links


项目(P)
图表(D)
元件(M)
工具(T)
附件(A)
设置(S)
窗口(W)
帮助(H)
新增包... Ctrl+W
新增图表... Ctrl+Insert
新增元件... Ctrl+M
文档
代码工程
编译与运行
数据建模
转换
模型校验
Web Services
XML Schema
安全性
版本控制
导入/导出
基线管理... Ctrl+Alt+B
用例矩阵...
项目统计...
KUP++分析设计模型说明
个性化界面借用模版
安全性
用户管理...
组管理...
锁管理...
更改密码...
用其他用户身份登录...
我的锁定... Ctrl+Shift+L
开启安全配置...
要求编辑前锁定
加密密码...
时间: 2005-10-3 修改时间: 2009-7-10 13:05:30
KUP++分析设计
1、技术架构模型
2、分析模型（应用框架模型）
3、领域模型（业务模型）


用例标准
用例
基本包:
Start here
重新加载
阶段匹配
关键字匹配
书签:
All
用例:
0
包含角色
包
Name
类型
Complexity
Phase
技术复杂度因素
未调整 TCF 值 (UTV):
47
TCF 权重因素 (TWF):
0.01
TCF 常数 (TC):
0.6
TCF = TC + (TWF x UTV):
1.07
环境复杂度因素
未调整 ECF 值 (UEV):
21.5
ECF 权重因素 (EWF):
-0.03
ECF 常数 (EC):
1.4
ECF = EC + (EWF x UEV):
0.755
未调整用例点 (UUCP) = 复杂度总和
0
平均时间
Easy: -2147483648 Med: -2147483648
总评估
用例点 (UCP) = UUCP * TCF * ECF =
0 * 1.07 * 0.755 = 0 UCP
评估工作量 (hours) =
10 * 0 = 0 小时
评估成本 = EWE * 默认时间比 =
0 * 40 = 0 成本
重新计算
报告
查看报告
默认比率
关闭
帮助(H)


元件(M)
工具(T)
附件(A)
设置(S)
窗口(W)
帮助(H)
属性... Alt+Enter
增加标记值... Ctrl+Shift+T
链接到文档... Ctrl+Alt+D
变量... F9
方法... F10
更多...
可见性设置... Ctrl+Shift+Y
高级
RTF 文档...
代码工程
在外部编辑器中打开源码... F12
在项目浏览器中定位 Alt+G
在图表中查找... Ctrl+U
自定义值... Ctrl+J
外观
对齐
匹配大小
排列
大小
移动
间距
方法重载... Ctrl+Shift+O
父类与接口... Ctrl+I
嵌入元件... Ctrl+Shift+B
更改类型...
09-7-10 13:05:30 100% 799 x 1067
设计模型



雲子嘟嘟

2009-8-27