**一、时序图简介（Brief introduction）**

时序图（Sequence Diagram）是显示对象之间交互的图，这些对象是按时间顺序排列的。顺序图中显示的是参与交互的对象及其对象之间消息交互的顺序。时序图中包括的建模元素主要有：对象（Actor）、生命线（Lifeline）、控制焦点（Focus of control）、消息（Message）等等。

**二、时序图元素（Sequence Diagram Elements）**

角色（Actor）

系统角色，可以是人、及其甚至其他的系统或者子系统。

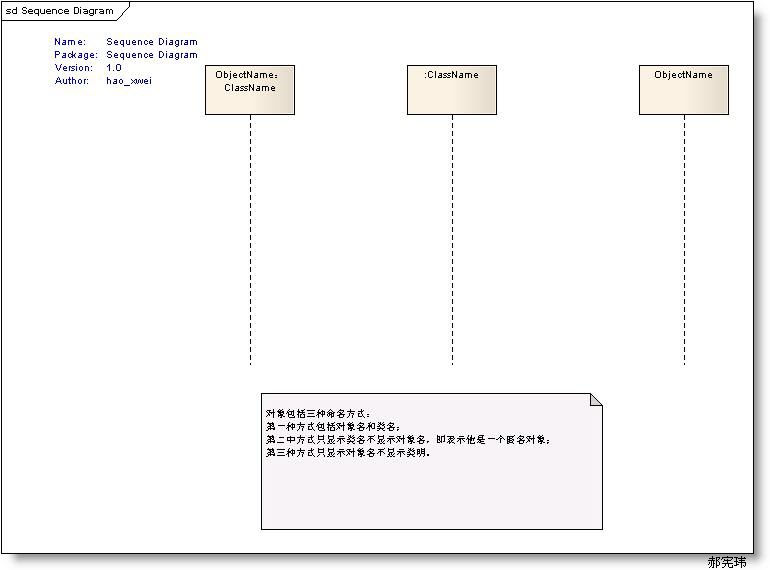
对象（Object）

对象包括三种命名方式：

第一种方式包括对象名和类名；

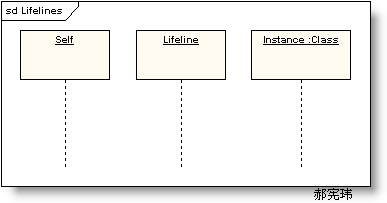
第二中方式只显示类名不显示对象名，即表示他是一个匿名对象；

第三种方式只显示对象名不显示类明。



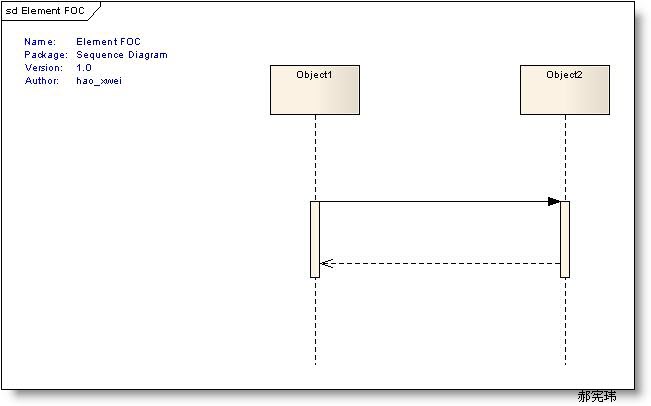
生命线（Lifeline）

生命线在顺序图中表示为从对象图标向下延伸的一条虚线，表示对象存在的时间，如下图



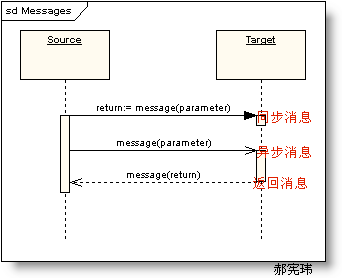
控制焦点（Focus of Control）

控制焦点是顺序图中表示时间段的符号，在这个时间段内对象将执行相应的操作。用小矩形表示，如下图。



消息（Message）

消息一般分为同步消息（Synchronous Message），异步消息（Asynchronous Message）和返回消息（Return Message）.如下图所示：



同步消息=调用消息（Synchronous Message）

消息的发送者把控制传递给消息的接收者，然后停止活动，等待消息的接收者放弃或者返回控制。用来表示同步的意义。

异步消息（Asynchronous Message）

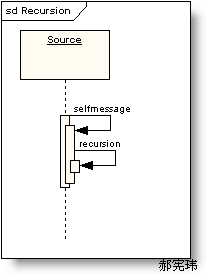
消息发送者通过消息把信号传递给消息的接收者，然后继续自己的活动，不等待接受者返回消息或者控制。异步消息的接收者和发送者是并发工作的。

返回消息（Return Message）

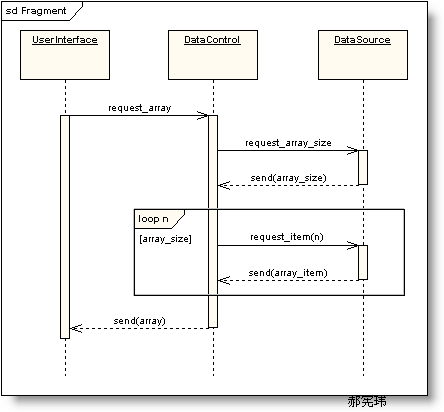
返回消息表示从过程调用返回

自关联消息（Self-Message）

表示方法的自身调用以及一个对象内的一个方法调用另外一个方法。



Combined Fragments



* Alternative fragment（denoted “alt”） 与 if…then…else对应
* Option fragment (denoted “opt”) 与 Switch对应
* Parallel fragment (denoted “par”) 表示同时发生
* Loop fragment(denoted “loop”) 与 for 或者 Foreach对应

**三、时序图实例分析（Sequece Diagram Example Analysis）**

时序图场景

完成课程创建功能，主要流程有：

1、请求添加课程页面，填写课程表单，点击【create】按钮

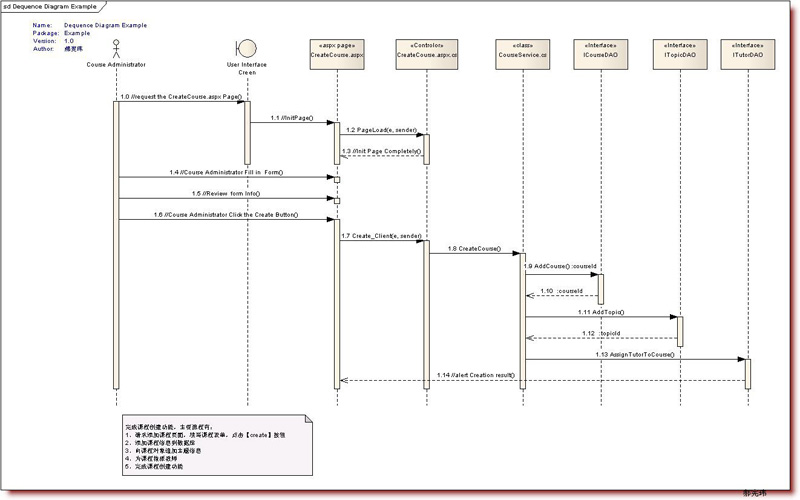
2、添加课程信息到数据库

3、向课程对象追加主题信息

4、为课程指派教师

5、完成课程创建功能

时序图实例



时序图实例分析

1、序号1.0-1.3 完成页面的初始化

2、序号1.4-1.5 课程管理员填充课程表单

3、序号1.6-1.7 课程管理员点击【Create】按钮，并响应点击事件

4、序号1.8 Service层创建课程

5、序号1.9-1.10 添加课程到数据库，并返回课程编号CourseId

6、序号1.11-1.12 添加课程主题到数据库，并返回主题编号topicId

7、序号1.13 给课程指派教师

8、序号1.14 向界面抛创建课程成功与否的消息

**四、总结（Summary）**

时序图（Sequence Diagram）是显示对象之间交互的图，这些对象是按时间顺序排列的。顺序图中显示的是参与交互的对象及其对象之间消息交互的顺序。时序图中包括的建模元素主要有：对象（Actor）、生命线（Lifeline）、控制焦点（Focus of control）、消息（Message）等等。最后，以课程创建功能演示一时序图实例。