**实验八 软件需求规格说明SRS（3）**

实验目的：

1. 学习Petri网基本知识及如何应用Petri网建模
2. 练习用各种动态建模工具 (状态图、Petri网、数据流图、OCL逻辑等）对所负责的项目进行建模
3. 完善自己项目的SRS

实验内容：

1.阅读“SYSTEM MODELLING WITH PETRI NETS”，进一步学习Petri网知识，了解如何应用Petri网对系统进行建模  
2.针对各自负责项目的不同场景，练习用各种动态建模工具 (状态图、Petri网、数据流图、OCL逻辑等）建模需求，与用户沟通。

参考第4章，4.12 皮卡迪里电视广告售卖系统，分析、归纳、总结出符合实际的需求规格。

3.分工协作，用上面的工作补充完善SRS和所负责的项目。

项目跟踪，建立能反映项目及小组每个人工作的进度、里程碑、工作量的跟踪图或表，将其保存到每个小组选定的协作开发平台上，每周更新。

**1.进一步学习Petri网知识，了解如何应用Petri网对系统进行建模**

Petri网是一种适合于系统描述和分析的数学模型，主要描述异步和并发关系。Petri网模型自然，直观，简单易懂的描述了在分析并行系统的状态行为的技术。Petri网主要用于：计算机协议模型、柔性系统模型、计算机集成制造、人工智能、系统分析等领域。

Petri网由位置（P），变迁（T），有向弧线，令牌（token）所构成，其中P表示状态元素，T为变化元素，有向弧线既可以由P到T,又可以由T到P，token表示一种属性。其中P,T平等。T由P来描述，P由T而变化，T引起P中资源流动，F联系P,T。Petri网可描述为：{S,T,F,M0} S：表示状态，T：表示变化条件，F：表示有向弧，M0表示令牌的初始位置。

因为petri网中的弧是有方向的，所以petri网图是有向图。又因为petri网中的节点可以分为两个集合：place和transition，并且每条弧都是从一个集合中的元素连到另一个集合中的元素，所以petri网图是一个有向二分图。

Petri网的元素：

(1)库所（place）：圆形节点

(2)变迁（transition0）：方形节点

(3)有向弧（connection）：库所与变迁之间的有向弧线

(4)托肯（token）：库所中的动态对象，可以从一个库所移动到另一个库所

如果一个变迁的每个输入都至少有一个token，则这个transition被enabled。变迁发生时，会从它的每个输入移去一个token，在它的每个输出放置一个token，以此类推。有两个或多个变迁都被允许的可能，但是一次只能发生一个变迁。随机数确定发生的变迁。

一个petri网的状态是它的所有标识的集合（向量μ）。当一个变迁发生时引起的状态变化由一个局部函数δ来定义，叫做下一状态函数。

从一个标识μ可以同时发生一组变迁。如果从μ同时发生一些变迁可以得到μ’，称μ’是立即可达( immediately reachable )的。可达集合( reachability set )R(M)被定义为M = (P, T, I, O, μ)从μ出发可以得到的所有状态的集合。

给定一个标识的petri网，标识记作u。给定一个标识u’。是否可以从u得到u’是petri网的可达性问题。可以看作集合可达性问题的一个特例，很多问题都可以归约到可达性问题。

如果没有一个变迁激活序列可以触发一个变迁，我们称这个变迁是死的( dead )；反之，变迁是活的( live )。为了研究操作系统的死锁问题，在Perti网中定义了变迁的deadness和liveness。

如果出现一个变迁，其输入库所的个数与输出库所的个数不相等，令牌的个数将发生变化，即就是，消耗托肯数目=前驱库所数目-后继库所数目

两个变迁争夺一个托肯的情形被称之为冲突。当发生冲突的时候，由于Petri网的时序是不确定的，因此具体哪个变迁得以发生也是不确定的。实际应用中，往往需要避免这种情形。用于描述现象的Petri网也可能自然出现冲突，这表明我们对于变迁发生的条件没有完全了解。

Petri是静态的，也就是说，不存在发生了一个变迁之后忽然冒出另一个变迁或者库所，从而改变Petri网结构的可能。Petri网的状态由托肯在库所的分布决定。也就是说，变迁发生完毕、下一个变迁等待发生的时候才有确定的状态，正在发生变迁的时候是没有一个确定的状态的。

图灵机在已知输入输出情况下研究处理和数据的实现问题，即Turing’s machine = （I，O；P，D）–注意其中用的是分号（；），处理就是算法、程序Programming，数据是DataStructure。图灵机的工程化技术已经成熟，包括从汇编语言到UML语言在内的全系列软件工程和程序设计语言，核心是结构化程序设计语言。

Petri网在已知变化状态条件下研究输入和输出的网络结构问题，即Petri nets =（P，D；I，O）–注意其中用的是分号（；）。

图灵机是“从蛋（I，O）开始研究蛋（I，O）孵鸡（P，D）”问题，而Petri网是“从鸡（P，D）开始研究鸡（P，D）生蛋（I，O）”问题。两者精确对偶，统一起来就可以完全解决了“蛋（I，O）孵鸡（P，D）、鸡（P，D）生蛋（I，O）”的自然循环，因此被我统称为自然（或实）计算模型，目的是区别于我预计10年后才有可能研究成熟并公开的“虚计算模型”。

对偶定律告诉我们，对偶模型可以相互解决它俩各自所不能解决的问题。Petri网的实践（语用网）可以解决Turing机所不能解决的“软件模块复用（也就是计算协作）”问题；而 Turing机的实践（算法分析-也就是软件工程）解决了Petri网所不能解决的“节点爆炸”问题。这就是我把它俩统一起来研究的原因。