面向嵌入式系统的UML模型的仿真工具用户使用手册

（V1.0 版）

**2015** 年 **10** 月 **7** 日

北航软件工程研究所

目录

[1. 工具概述 3](#_Toc432195614)

[1.1 工具运行环境 3](#_Toc432195615)

[1.1.1 硬件环境 3](#_Toc432195616)

[1.1.2 软件环境 3](#_Toc432195617)

[1.2 工具安装 3](#_Toc432195618)

[1.3 工具简介 3](#_Toc432195619)

[2. 使用说明 4](#_Toc432195620)

[2.1建立设计模型 4](#_Toc432195621)

[2.1.1 顺序模型 4](#_Toc432195622)

[2.1.2 部署模型 7](#_Toc432195623)

[2.2 导入设计模型 8](#_Toc432195624)

[2.3 分析仿真结果 9](#_Toc432195625)

# 1. 工具概述

## 1.1 工具运行环境

### 1.1.1 硬件环境

仿真工具应满足如下硬件环境要求：

|  |  |  |
| --- | --- | --- |
| **CPU** | 硬盘 | 内存 |
| PIII以上 | 10G以上 | 1G以上 |

### 1.1.2 软件环境

仿真工具运行需满足如下软件运行条件：

|  |
| --- |
| 操作系统平台及应用软件 |
| (1) windows、Linux、OSX均可  (2) 需要安装JAVA运行环境JRE Version7及以上 |

## 1.2 工具安装

该仿真工具无需安装，解压后直接双击SimulationTool.bat即可运行，该批处理脚本中包含执行Jar包的命令（java -jar SimulationTool.jar pause）。

## 1.3 工具简介

该工具是用于分析面向嵌入式系统UML设计模型的性能。主要功能主要包括以下三个方面：支持不同的建模工具所生成的.UML模型文件；支持实时和嵌入式系统建模与分析语言（Modeling and Analysis of Real-Time and Embedded Systems，MARTE）所建立的设计模型；支持设计模型的自动导入以及详细地分析过程和分析结果的可视化。用户使用建模工具建立面向嵌入式系统的设计模型，并配置相应的非功能性参数值。将所生成的设计模型文件为仿真引擎的输入，即可自动生成从设计模型到仿真模型的转换，并仿真系统运行，收集系统运行时数据。

# 2. 使用说明

## 2.1建立设计模型

首先使用可视化建模工具建立面向嵌入式系统的设计模型。以本课题数据传输中间件作为背景使用开源建模工具Papyrus建立其设计模型作为仿真引擎的输入，下面对该设计模型进行简单描述。

### 2.1.1 顺序模型

* 发布端请求发布流程

该顺序图用来描述DDS中间件发布过程，为消息的发送做准备工作。如图1所示，首先发布者依据所提供的主题ID、域ID寻找域的管理对象，查看该主题是否是第一次发布，如果不是则需要建立相应的DataWriter和TransportInterface，准备数据发送。

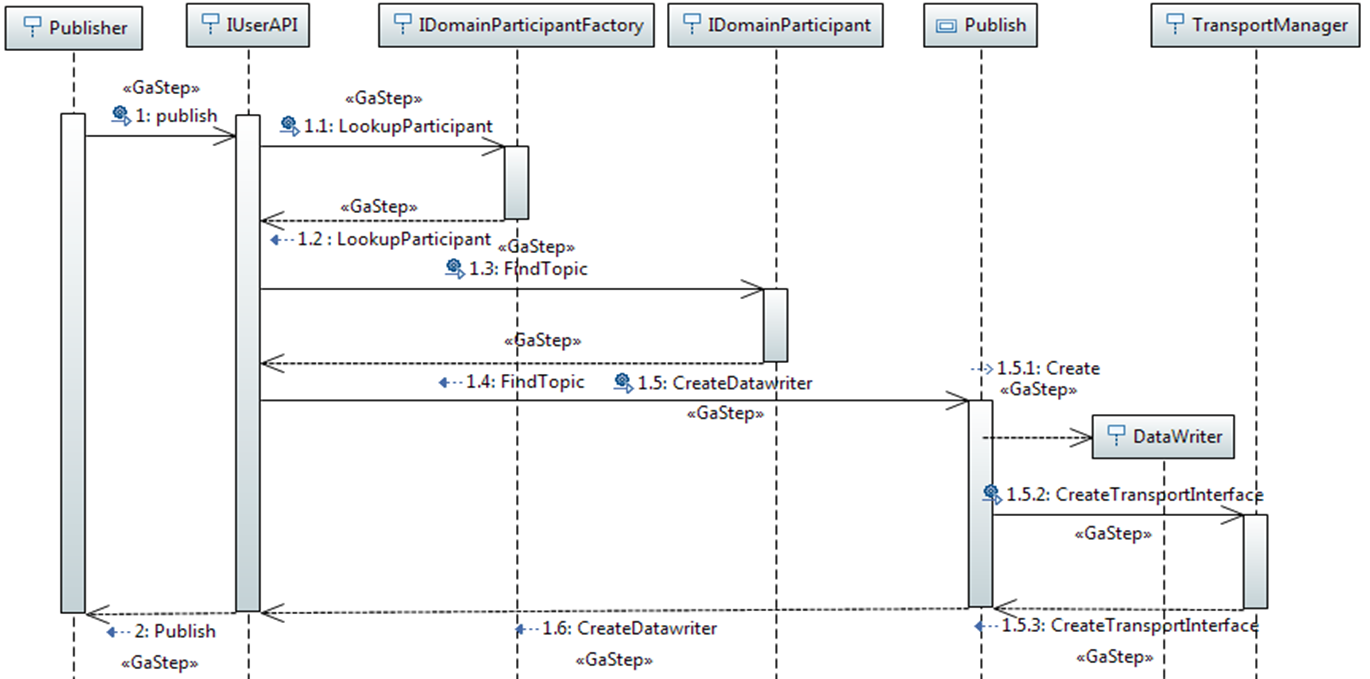


图1：发布端请求发布流程

* 发布端发送数据流程

发布者发布了主题信息后，之后向订阅者发送所订阅主题的数据。当调用发送操作后，实际的发送是由传输协议进行发送。发送流程如图2所示，首先应用程序调用发送信息接口，根据参数域ID获得该域的管理类，然后调用该类的查找发布者操作，获取发布者实体，发布者通过查找该主题的数据写入实体，对信息进行写入操作，调用在请求发布模块创建好的传输接口对信息进行封装，通过网络套接字将信息发送出去。

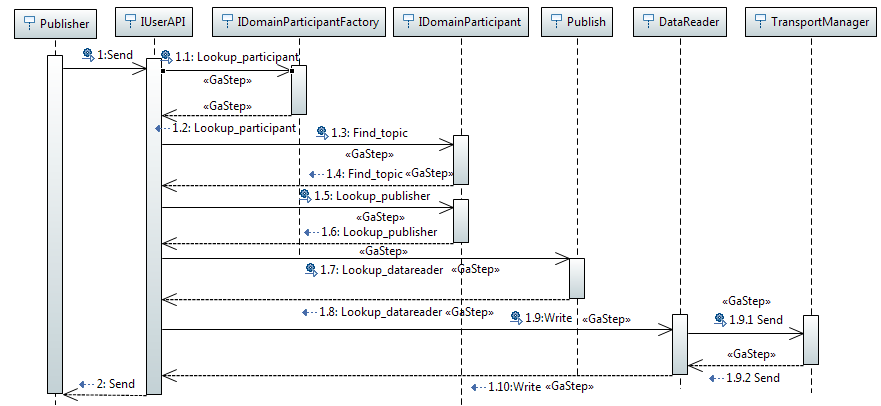


图2：发布端发送数据流程

* 收端接收数据流程

订阅者在订阅了相关主题信息后，便可以采用回调机制实时接收信息，其接收流程如图3所示。首先负责网络通讯的模块接收到主题信息后，全局的IAppBaseHandle类，有该类通知对应域的Subscriber，同时通知SubscriberListener处理同时订阅多个主题的联合订阅，而每次订阅一种类型的订阅都用（DataReader）来处理，与DataReader对应的DataReaderListener接收到有信息到达的事件后，通过主题信息查找对应的Handler类，自动回调应用程序的接收函数，便可完成信息的接收。

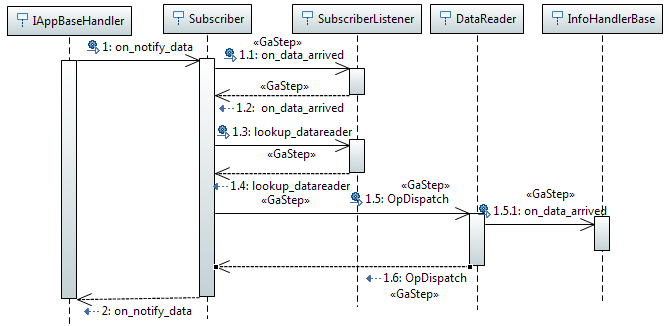


图 3：订阅端接收数据流程

* 链路中传输数据流程

分布式系统节点之间的数据传输过程如图4所示，主要用来分析节点之间数据传输过程中丢包重传过程。使用MARTE性能分析建模包PAM的构造型GaCommStep描述每次传输数据的大小，以及基于注释的形式标注传输不同的数据量所需要的时间和重传定时器的设置规则。

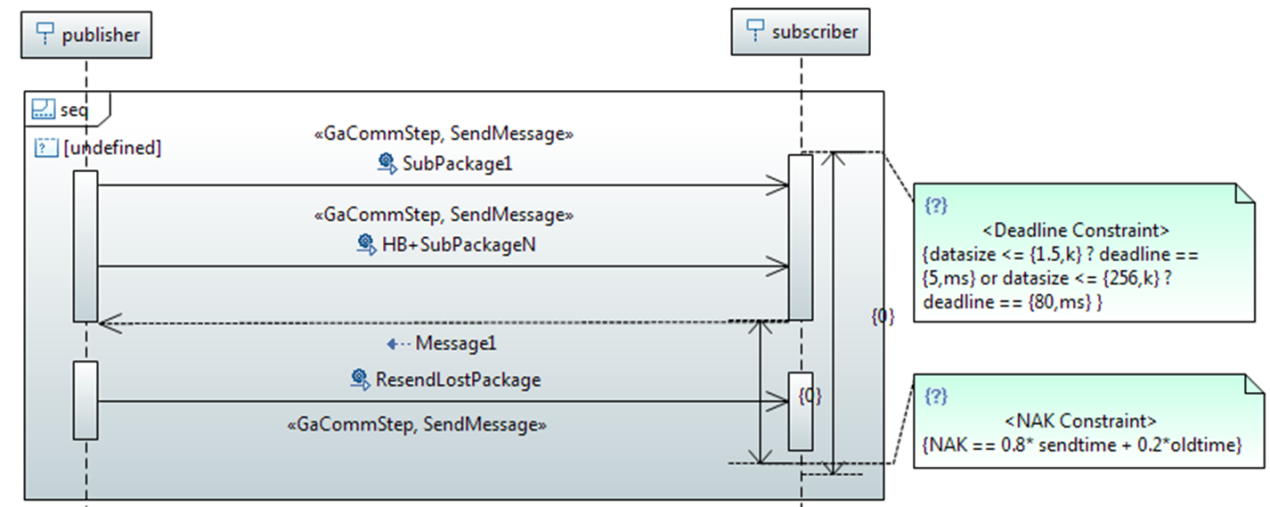


图4：链路数据传输流程

* 系统调用进程行为

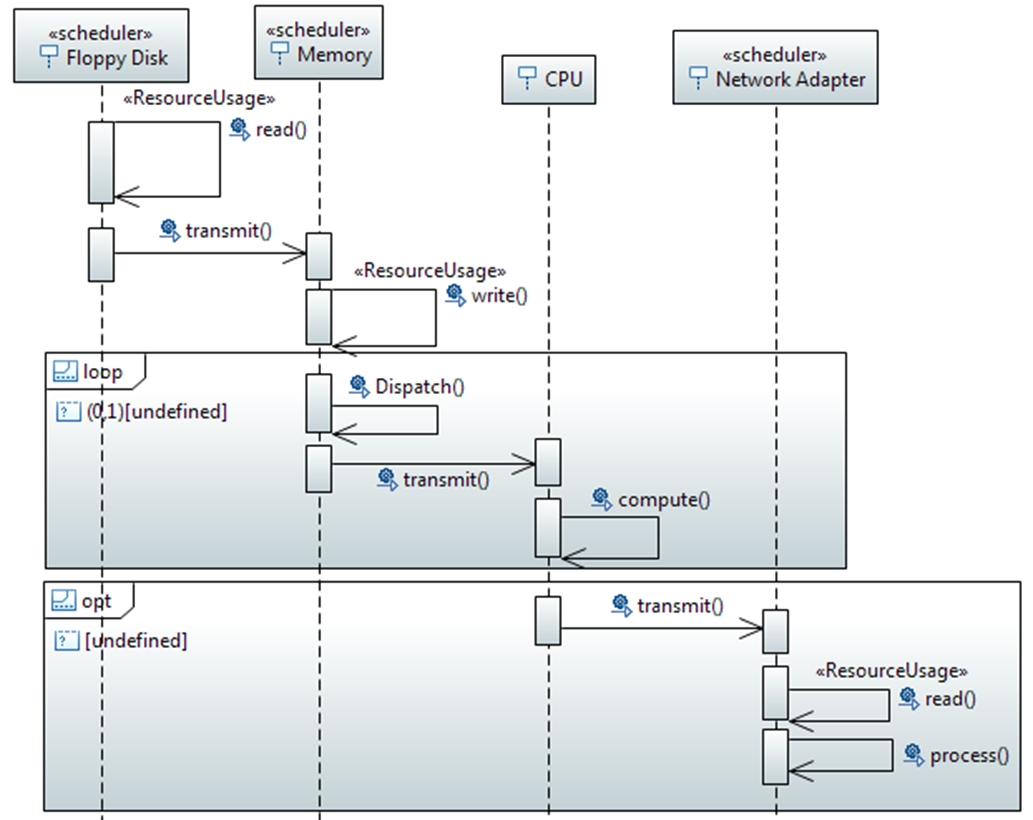
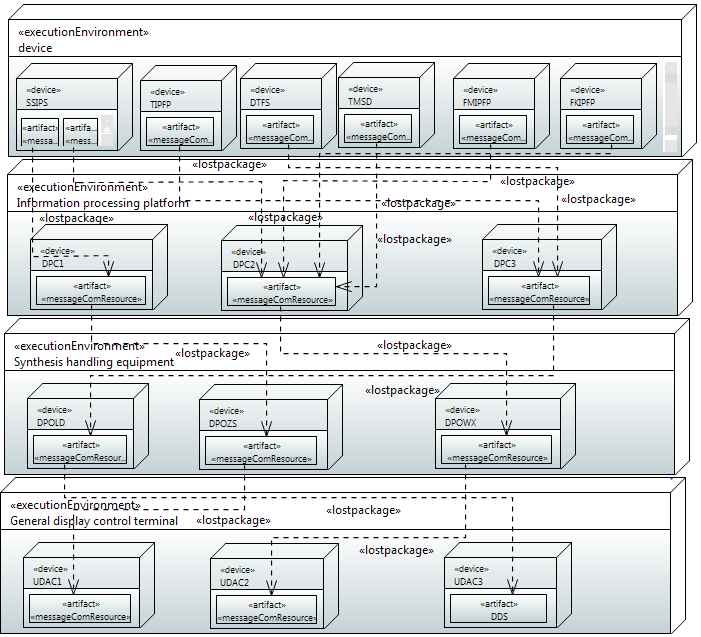


图5：系统调用进程行为

进程在节点中执行所需要的时间，受到内核调度进程策略和磁盘读写速度的影响，其中进程是由性能仿真指令所组成。在本课题中，为了评估进程在节点中执行的时间，主要考虑了磁盘读写进程所需要的时间、网卡读取数据所需要的时间、内核调度进程策略以及网络协议对数据处理所需要的时间。系统调度进程的行为如图5所示，首先仿真指令先保存在磁盘上，接收到来自CPU的指令，先将逻辑块号翻译为扇区地址（由于该时间消耗值比较小，本课题没有考虑），读取扇区地址的数据，将数据通过直接存取访问技术将数据传送至内存。然后由内核依据不同的调度策略调度进程至CPU，由CPU对进程执行的时间进行计算，如果是需要发送的数据，将数据发送至网卡适配器，然后网络协议将数据进一步封装，通过网卡发送至链路中。

### 2.1.2 部署模型

图6：基于DDS中间件的船舶某控制系统的部署模型

用来描述基于DDS中间件分布式系统节点之间的发布订阅关系，以及数据传输的情形，如图6所示。在本课题的研究中考虑了一个节点中多个进程给订阅者发送消息，以及一个订阅端订阅了多个发布端的主题的情形。

## 2.2 导入设计模型

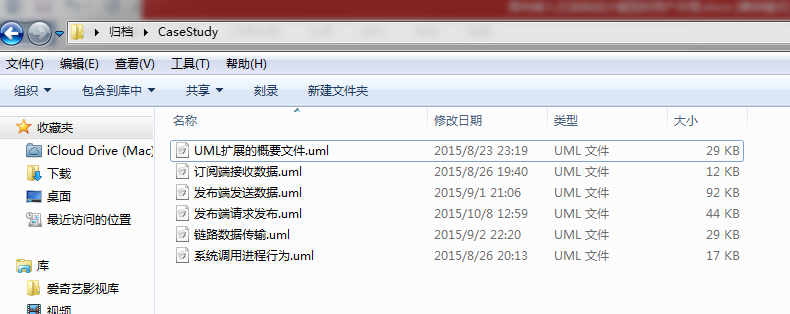


图 7：模型文件生成目录

所生成的设计模型文件如上图所示，其中本工具支持三种内核调度进程方式，分别为时间片轮转方式（TimeTableDriven）、进程优先级以及先进先出运行方式，表示方式在模型文件中如下图7、图8、图9所示。

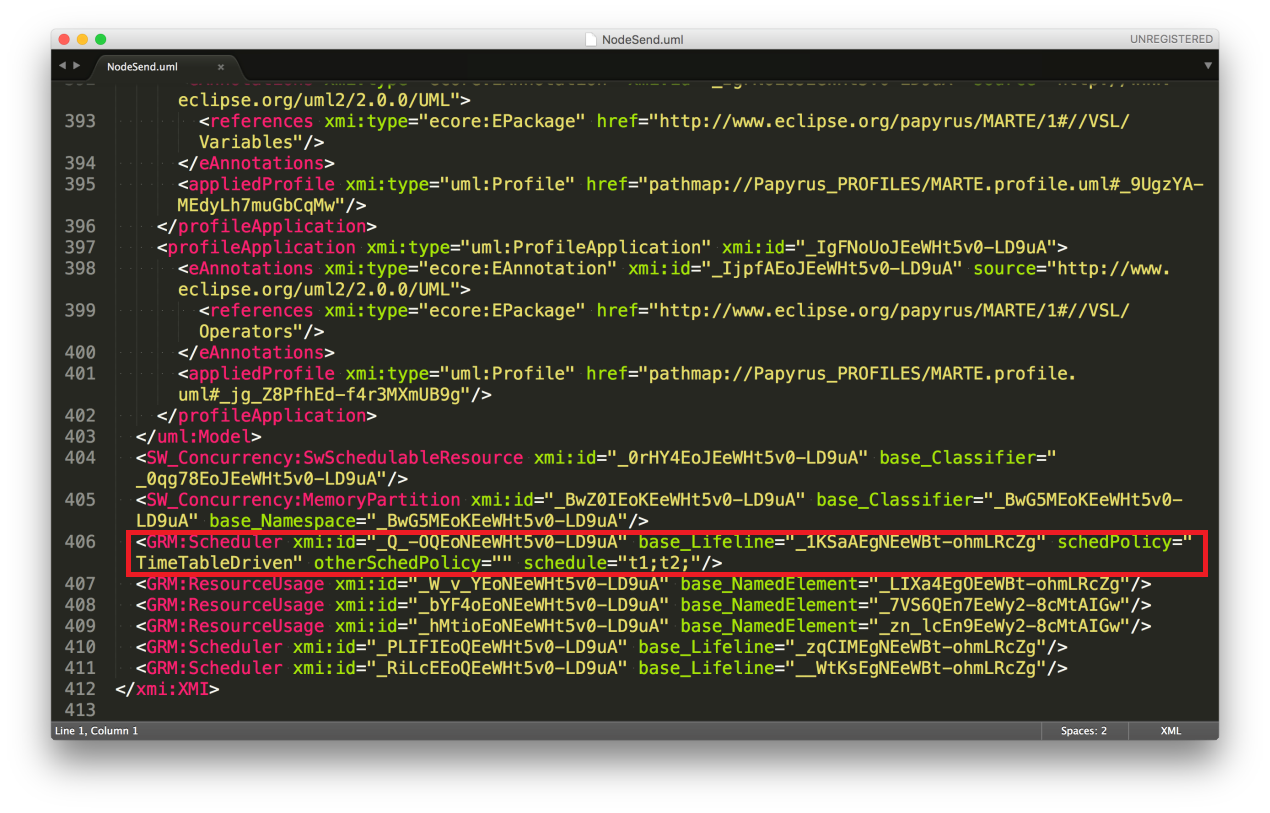


图 8：时间片轮转调度方式

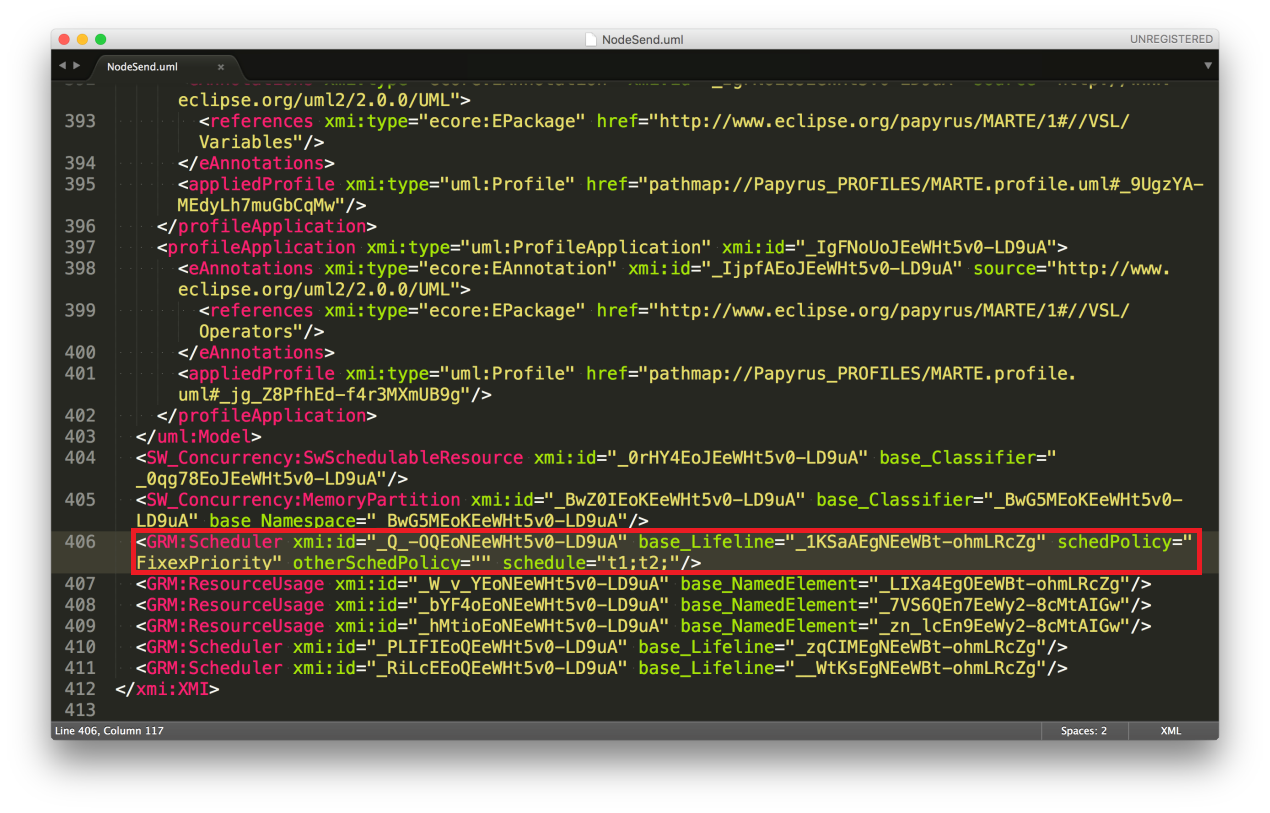


图9：进程优先级调度方式

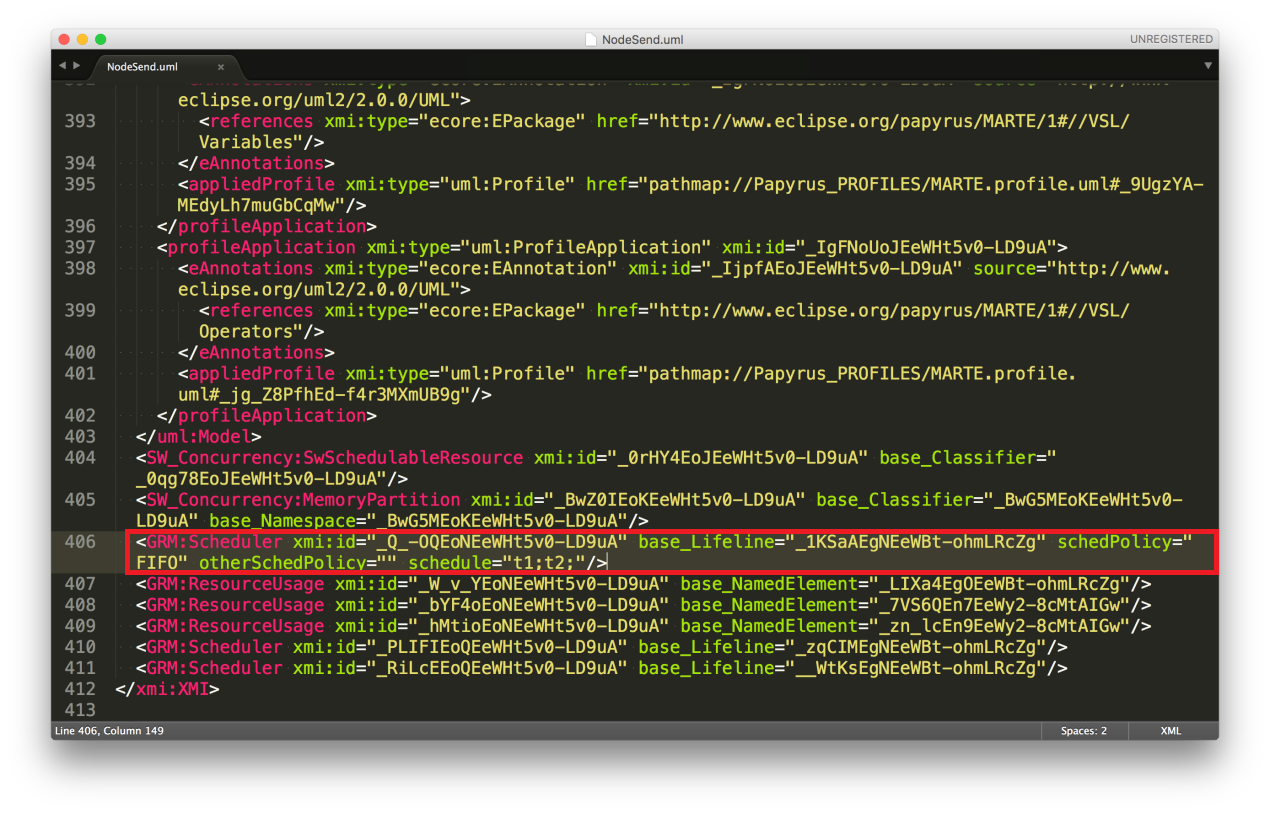


图 10：先来先出调度方式

## 2.3 分析仿真结果

双击SimulationTool.bat运行仿真引擎，系统会自动生成性能仿真指令，并编译执行，将性能度量目标以及分析下过程以文本格式保存在当前项目目录下，生成的性能度量目标文件和分析过程如下图所示。其中SchedPolicy为调度方式，from列表示部署图中所配置的构件名称，to列表示部署图中所配置的构件名称，sendDataSize表示构件发送的数据大小以及数目，time表示发送这些数据所需要的时间，reliability表示构件之间数据传输的可靠性。

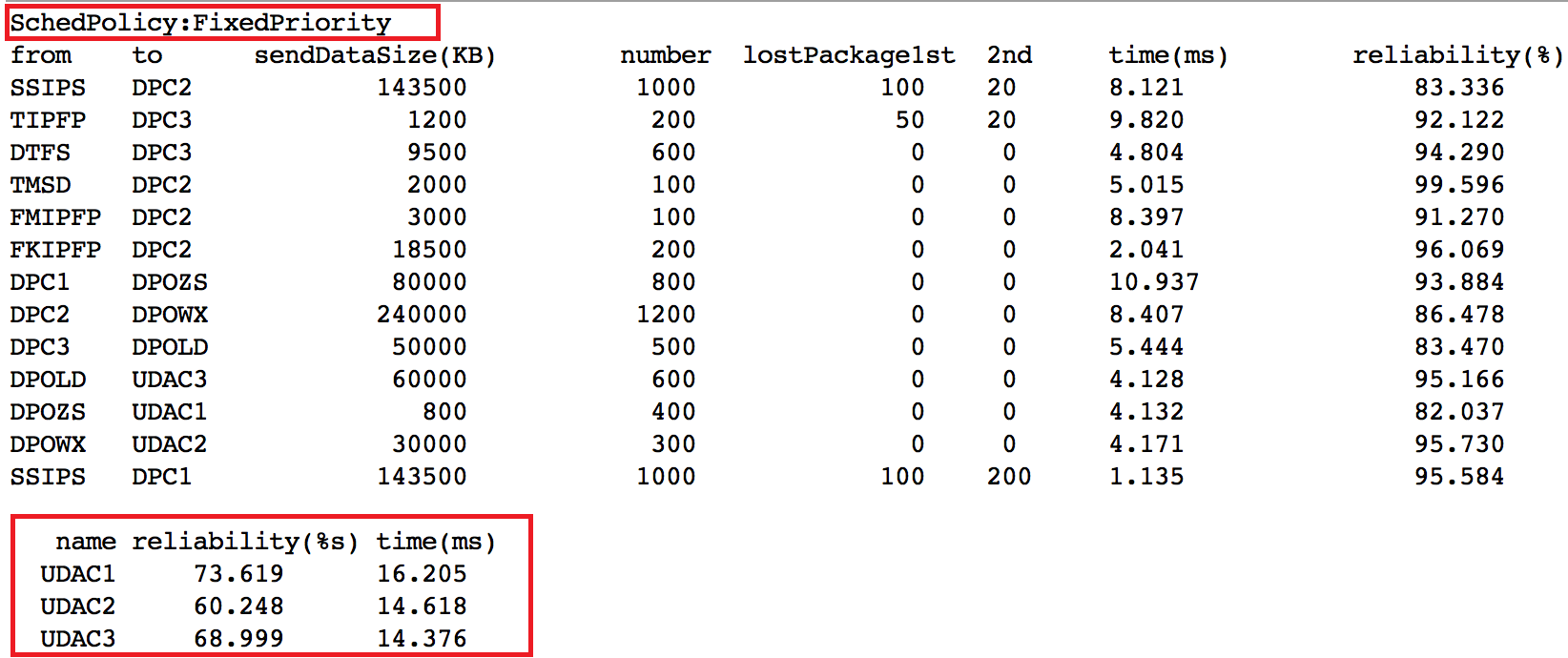


图 11：优先级优先调度方式分析结果

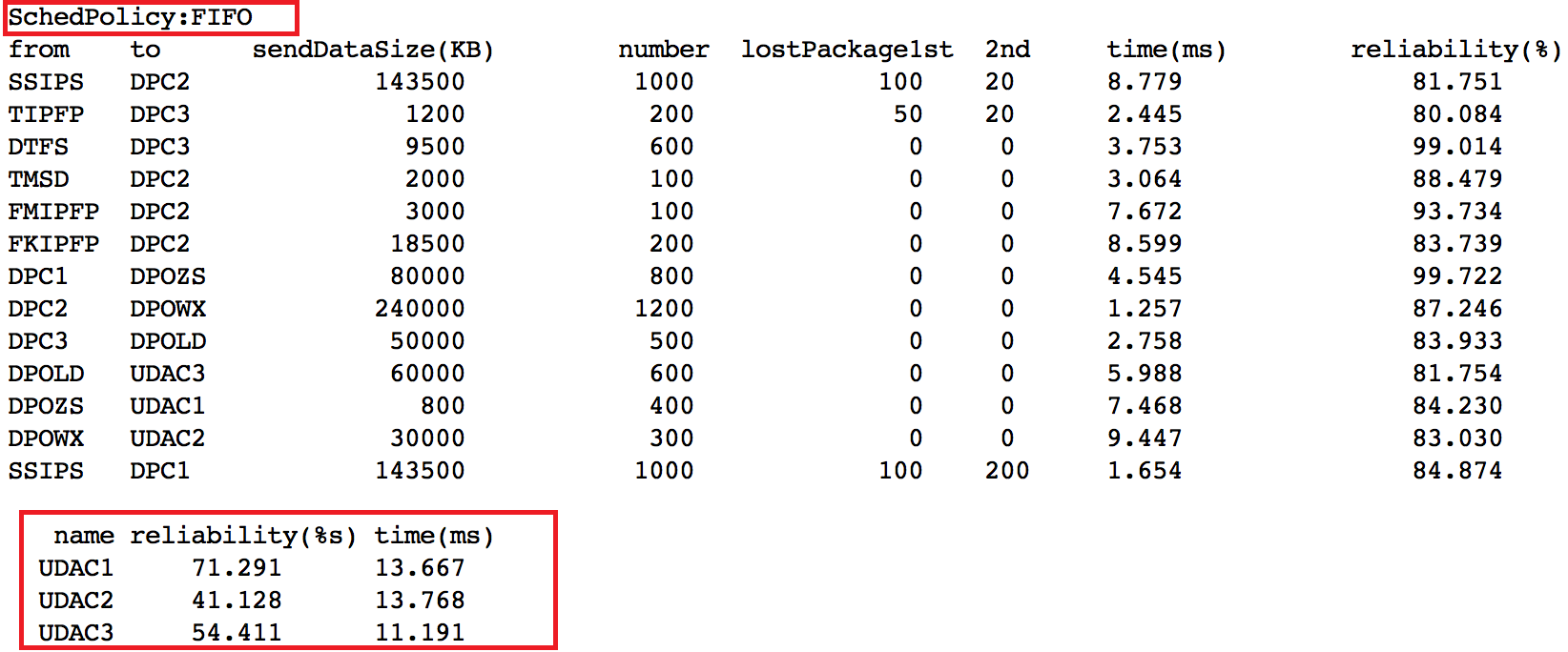


图 12：先来先出调度方式的分析结果

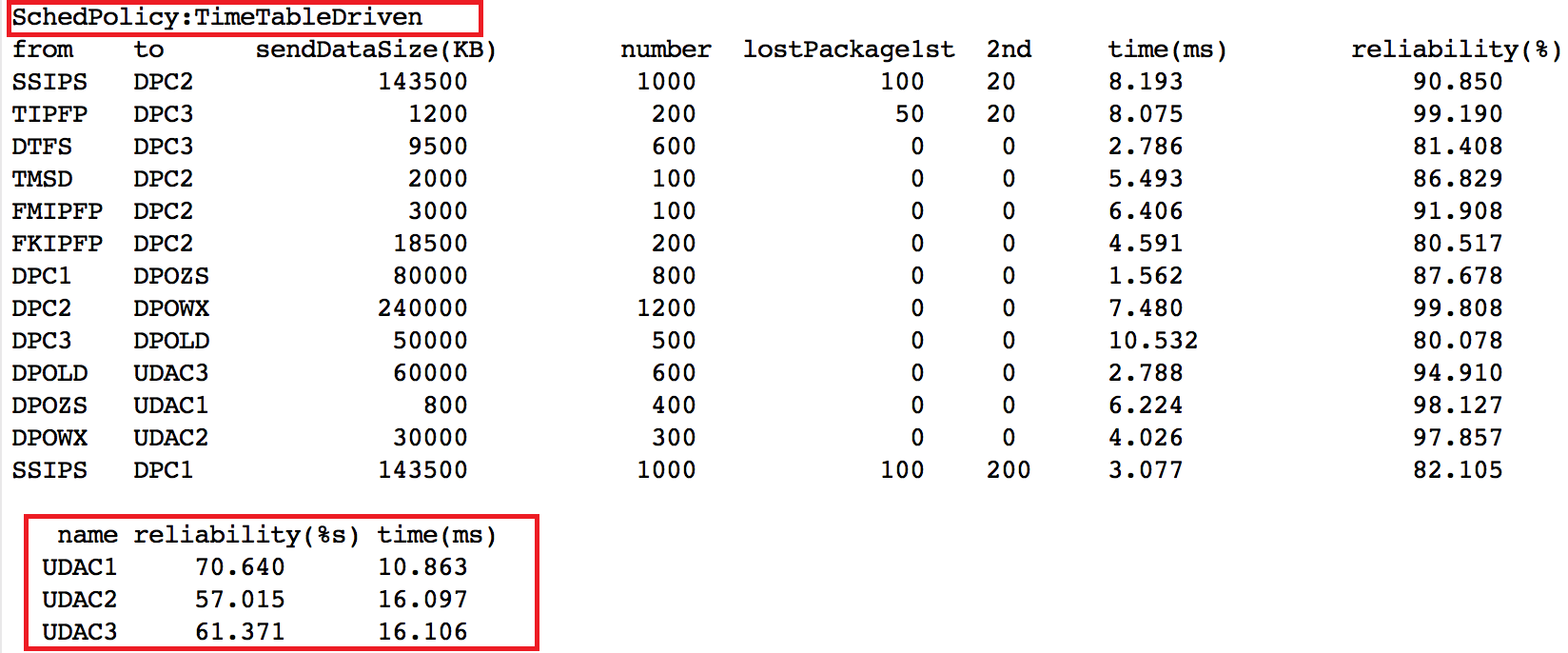


图 13：时间片轮转调度方式的分析结果