Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение науки и высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №3

по дисциплине

" Интеграция кроссплатформенных программных систем"

Тема: " Основы XSLT. Преобразования средствами JAXP"

Выполнил:

ст. гр. ИСТм-121

В.Г. Хлызова

Принял:

Спирин И.В.

Владимир, 2022 г.

ЦЕЛЬ РАБОТЫ

Данная работа предусматривает разработку описания XSLT-преобразования, результат которого был бы эквивалентен результату, получаемому в ЛР2. При этом собственно XSLT-преобразование необходимо выполнить программно, используя средства JAXP, для программы сохраняются все условия, которые указаны в задании для предыдущей работы, но добавляются те, что указаны ниже.

ХОД РАБОТЫ

Предметная область: проектирование информационных систем.

[Ключевые элементы BPMN моделей.](https://www.comindware.ru/blog/%d0%bd%d0%be%d1%82%d0%b0%d1%86%d0%b8%d1%8f-bpmn-2-0-%d1%8d%d0%bb%d0%b5%d0%bc%d0%b5%d0%bd%d1%82%d1%8b-%d0%b8-%d0%be%d0%bf%d0%b8%d1%81%d0%b0%d0%bd%d0%b8%d0%b5/)

Технические требования

1) XSLT-преобразование должно демонстрировать использование именованных параметризованных шаблонов

2) XSLT-преобразование должно демонстрировать использование инструкций ветвления и цикла

3) XSLT-преобразование должно демонстрировать использование инструкций нумерации и сортировки

4) XSLT-преобразование должно содержать указание метода вывода - HTML, сформированное с учетом требований спецификации HTML v4.01

5) программа преобразования должна использовать источник и результат преобразования различных типов (например, SAXSource и DOMResult)

6) программа преобразования должна обеспечить обработку ошибок этапа преобразования в дополнение к обработке ошибок этапа анализа по такому же принципу.

Ссылка на репозиторий: <https://github.com/arranay/Integration_of_cross_platform_software_systems>

Xsl файл:

<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

    <xsl:template match="/">

        <head>

            <META http-equiv="Content-Type" content="text/html; charset=UTF-8" />

            <link rel="stylesheet" href="mystyle.css">

                <title><xsl:value-of select="BPMN-model/model-name"/></title>

            </link>

        </head>

        <html>

            <body>

                <table>

                    <tr>

                        <td colspan="2" class="header">Common info about BPMN model</td>

                    </tr>

                    <tr>

                        <td>Name of model:</td>

                        <td><xsl:value-of select="BPMN-model/model-name"/></td>

                    </tr>

                    <tr>

                        <td>Author:</td>

                        <td><xsl:value-of select="BPMN-model/author"/></td>

                    </tr>

                    <tr>

                        <td>Date:</td>

                        <td><xsl:value-of select="BPMN-model/date"/></td>

                    </tr>

                </table>

                <table>

                    <tr>

                        <td colspan="2" class="header">Tracks</td>

                    </tr>

                    <xsl:for-each select="BPMN-model/tracks/track">

                        <tr>

                            <td class="track"><xsl:value-of select="track-name"/></td>

                            <td>

                                <xsl:if test="actions">

                                    <table>

                                        <tr>

                                            <td colspan="2" class="items">Actions</td>

                                        </tr>

                                        <tr class="item-header">

                                            <td>Operation</td>

                                            <td>Time</td>

                                        </tr>

                                        <xsl:for-each select="actions/action">

                                            <tr>

                                                <td><xsl:value-of select="operation"/></td>

                                                <td><xsl:value-of select="time" /></td>

                                            </tr>

                                        </xsl:for-each>

                                    </table>

                                </xsl:if>

                                <xsl:if test="events">

                                    <table>

                                        <tr>

                                            <td colspan="2" class="items">Events</td>

                                        </tr>

                                        <tr class="item-header">

                                            <td>Name</td>

                                            <td>Type</td>

                                        </tr>

                                        <xsl:for-each select="events/event">

                                            <tr>

                                                <td><xsl:value-of select="event-name"/></td>

                                                <td><xsl:value-of select="@type" /></td>

                                            </tr>

                                        </xsl:for-each>

                                    </table>

                                </xsl:if>

                                <xsl:if test="forks">

                                    <table>

                                        <tr>

                                            <td colspan="2" class="items">Forks</td>

                                        </tr>

                                        <tr class="item-header">

                                            <td>Condition</td>

                                        </tr>

                                        <xsl:for-each select="forks/fork">

                                            <tr>

                                                <td><xsl:value-of select="condition"/></td>

                                            </tr>

                                        </xsl:for-each>

                                    </table>

                                </xsl:if>

                            </td>

                        </tr>

                    </xsl:for-each>

                    <tr class="result">

                        <td>Result time</td>

                        <td><xsl:value-of select="sum(BPMN-model/tracks/track/actions/action/time)"/> ms</td>

                    </tr>

                </table>

                <table>

                    <tr>

                        <td colspan="3" class="header">Streams</td>

                    </tr>

                    <tr class="item-header">

                        <td>From</td>

                        <td>To</td>

                        <td>Name</td>

                    </tr>

                    <xsl:for-each select="BPMN-model/streams/stream">

                        <tr>

                            <xsl:choose>

                                <xsl:when test="from/@type = 'event'">

                                    <xsl:variable name="current-item-id" select="from/@item-id" />

                                    <xsl:for-each select="/BPMN-model/tracks/track/events/event">

                                        <xsl:if test="$current-item-id=@id">

                                            <td><xsl:value-of select="event-name"/></td>

                                        </xsl:if>

                                    </xsl:for-each>

                                </xsl:when>

                                <xsl:when test="from/@type = 'action'">

                                    <xsl:variable name="current-item-id" select="from/@item-id" />

                                    <xsl:for-each select="/BPMN-model/tracks/track/actions/action">

                                        <xsl:if test="$current-item-id=@id">

                                            <td><xsl:value-of select="operation"/></td>

                                        </xsl:if>

                                    </xsl:for-each>

                                </xsl:when>

                                <xsl:when test="from/@type = 'fork'">

                                    <xsl:variable name="current-item-id" select="from/@item-id" />

                                    <xsl:for-each select="/BPMN-model/tracks/track/forks/fork">

                                        <xsl:if test="$current-item-id=@id">

                                            <td><xsl:value-of select="condition"/></td>

                                        </xsl:if>

                                    </xsl:for-each>

                                </xsl:when>

                            </xsl:choose>

                            <xsl:choose>

                                <xsl:when test="to/@type = 'event'">

                                    <xsl:variable name="current-item-id" select="to/@item-id" />

                                    <xsl:for-each select="/BPMN-model/tracks/track/events/event">

                                        <xsl:if test="$current-item-id=@id">

                                            <td><xsl:value-of select="event-name"/></td>

                                        </xsl:if>

                                    </xsl:for-each>

                                </xsl:when>

                                <xsl:when test="to/@type = 'action'">

                                    <xsl:variable name="current-item-id" select="to/@item-id" />

                                    <xsl:for-each select="/BPMN-model/tracks/track/actions/action">

                                        <xsl:if test="$current-item-id=@id">

                                            <td><xsl:value-of select="operation"/></td>

                                        </xsl:if>

                                    </xsl:for-each>

                                </xsl:when>

                                <xsl:when test="to/@type = 'fork'">

                                    <xsl:variable name="current-item-id" select="to/@item-id" />

                                    <xsl:for-each select="/BPMN-model/tracks/track/forks/fork">

                                        <xsl:if test="$current-item-id=@id">

                                            <td><xsl:value-of select="condition"/></td>

                                        </xsl:if>

                                    </xsl:for-each>

                                </xsl:when>

                            </xsl:choose>

                            <td><xsl:value-of select="stream-name"/></td>

                        </tr>

                    </xsl:for-each>

                </table>

            </body>

        </html>

    </xsl:template>

</xsl:stylesheet>

Результат работы программы:

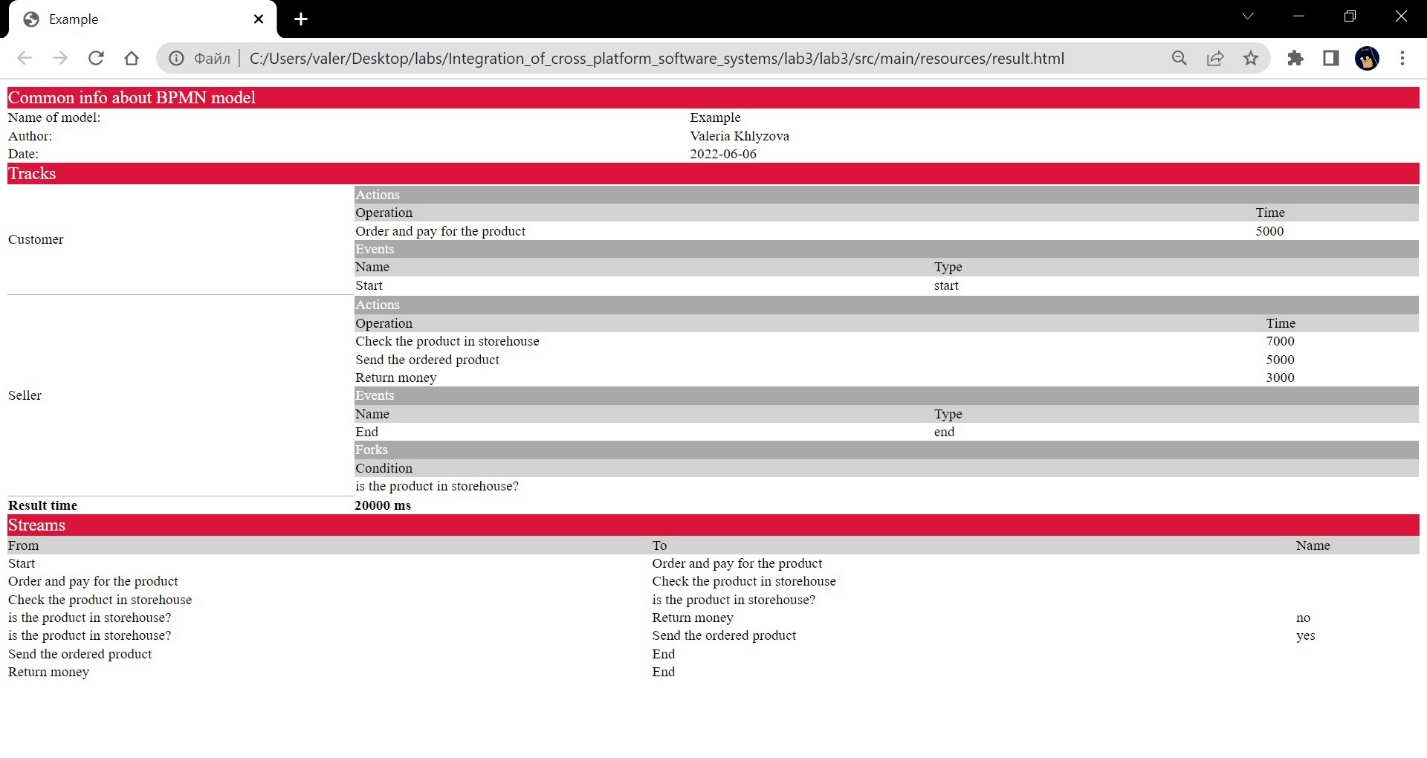


Рисунок 1 - Результат работы программы.

ВЫВОД

В ходе выполнения лабораторной работы были изучены основы XSLT. Произведена преобразования средствами JAXP.