

Среда визуального программирования роботов на .NET

Шигаров Никита (444 гр.)

Научный руководитель: к.т.н., доц. Литвинов Ю. В.

Рецензент: д.ф.-м.н., проф. Терехов А. Н.

Цель и задачи

Разработка среды визуального программирования роботов на .NET с помощью Modeling SDK for Visual Studio

- Выполнить обзор возможностей среды программирования роботов TRIK Studio
- Исследовать технологию Modeling SDK
- Создать архитектуру решения
- Реализовать решение
- Выполнить апробацию решения

Domain Specific Language

- *DSM-платформа* — это инструментальная технология разработки DSM-решений
- *Модель* — это некое упрощение предметной области, нужное для выполнения там некоторых полезных действий
- *Метамодель* — это описание визуального языка, то есть множества всех синтаксически корректных диаграмм на этом языке

TRIK Studio

The screenshot displays the TRIK Studio 3.1.4-β3 interface. The main window title is "TRIK Studio 3.1.4-β3 C:/TRIKStudio/examples/trik/stepic/labyrinth-with-sensors.qrs [изменён]". The menu bar includes "Файл", "Правка", "Вид", "Инструменты", "Настройки", and "Справка". The toolbar contains icons for file operations, a 2D view, and the TRIK logo.

The interface is divided into three main panels:

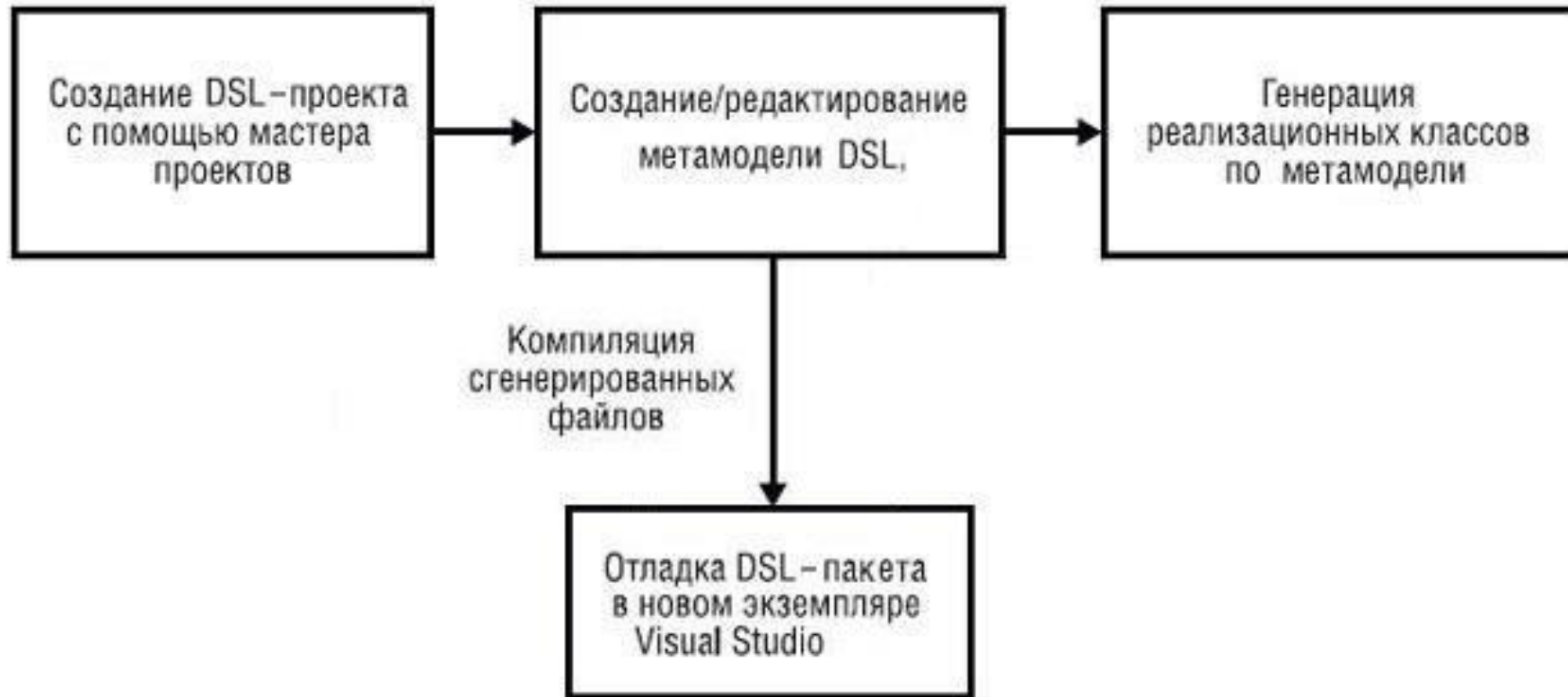
- Behavior Diagram (left):** Titled "Диаграмма поведения робота". It shows a flowchart starting with "Порты: M3, M4" and "Задержка 1000 мс" leading to a central yellow node. A decision box asks "Есть ли стена справа и спереди?". If "Да" (Yes), it triggers a "Регулятор" (red box) and "Задержка 0 мс", which then leads to a "лозь" (yellow box) and "Налево" (yellow box) action. If "Нет" (No), it leads to a "лозь" (yellow box) and "Налево" (yellow box) action.
- 2D Model (middle-right):** Titled "Двумерная модель". It shows a 2D maze with a robot icon and a sensor beam. The grid is set to "9,860 сек.". The axes range from -157.5 to 105 on the x-axis and -157.5 to 0 on the y-axis.
- Graphs (bottom-right):** Titled "Графики". It shows a green line graph on a black grid. The y-axis ranges from 17.0 to 100.0. The x-axis is labeled "A1: Инфракрасный сенсор". The graph shows a signal that starts at 100.0, drops to a minimum of 17.0, and then rises back to 100.0.

A status bar at the bottom indicates "Режим отладки - нажмите Ctrl+1 или кликните здесь для переключения в режим редактирования".

Modeling SDK for Visual Studio

- Шаблонный язык описания генераторов T4
- Доменные классы, отношения и фигуры
- Графический редактор
- Можно расширять функциональность с помощью partial-классов на языке C#
- Механизмы валидации
- Автопрокладывание связей

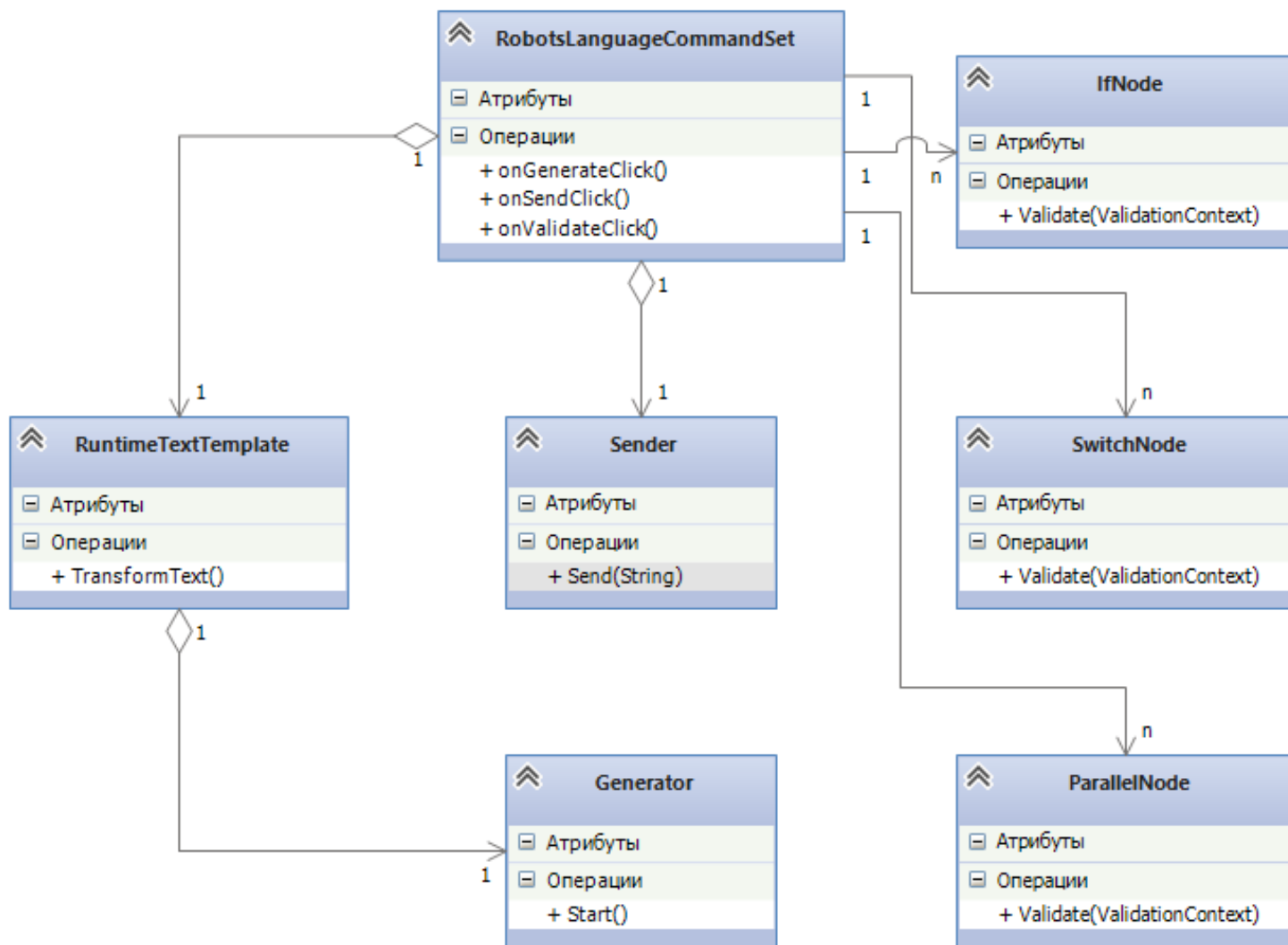
Схема разработки DSL в Modeling SDK



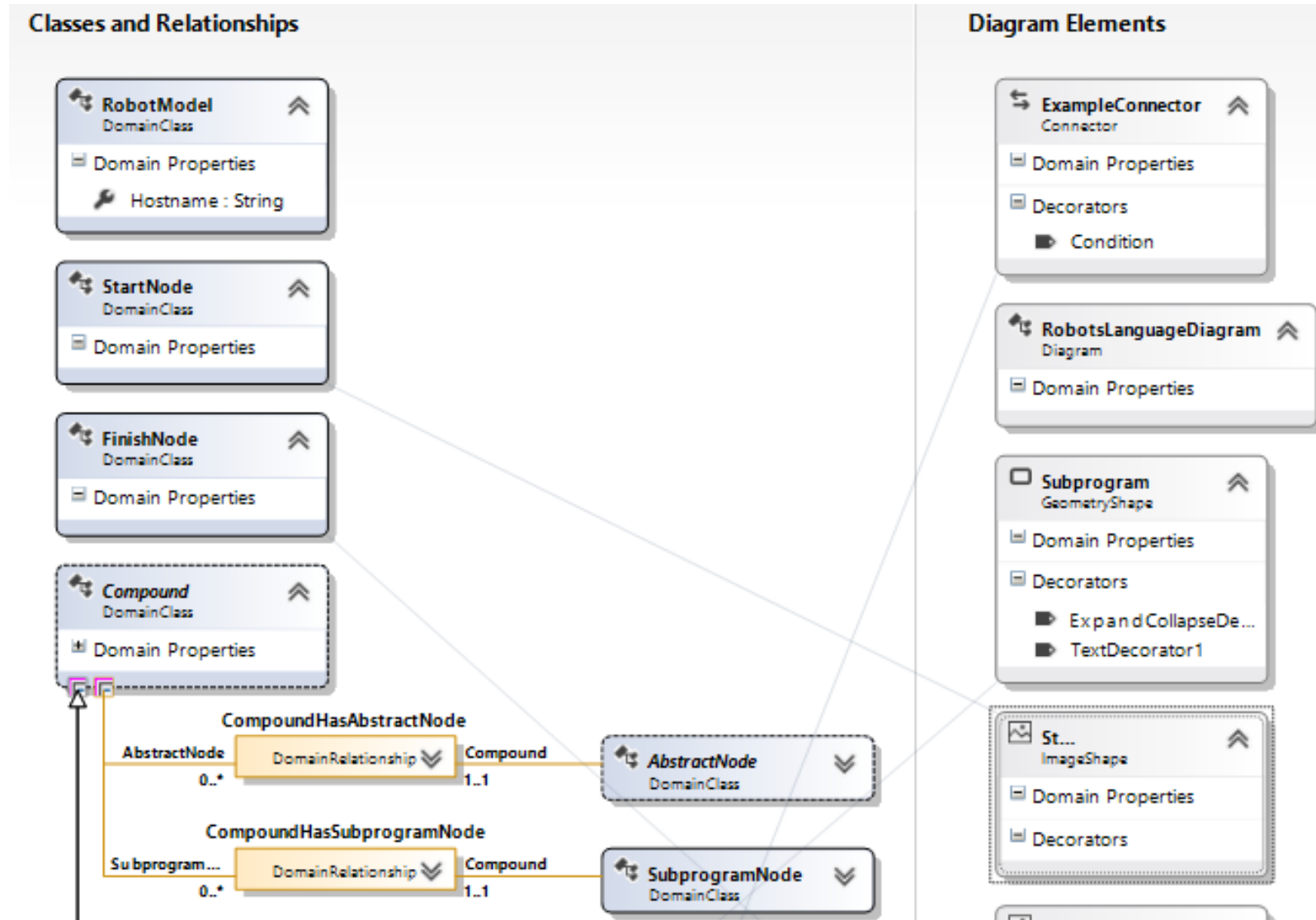
Требования к решению

- Редактор, позволяющий составлять корректные и некорректные диаграммы
- Генератор кода на JavaScript
- Валидатор
- Отправка на работа по TCP-соединению
- Выбор действий из меню

Диаграмма классов



Метамодель



Редактор

Панель элементов

Поиск по панели элементов

- RobotsLanguage
 - Указатель
 - Start
 - Finish
 - ConnectionTool1
 - Subprogram
 - If
 - EndIf
 - Iterations
 - SubprogramCall
 - Parallel
 - EndParallel
 - Break
 - Switch
 - EndSwitch
 - Motors
 - WaitSensor
 - Delay
 - WaitTouch
 - MotorsOff
- Общие

В этой группе нет элементов управления. Перетащите элемент в эту область, чтобы добавить его в панель элементов.

RobotsLanguage21.rob* RobotsLanguage25.rob RobotsLanguage25.rob.diagram Debugging.csproj RobotsLanguage24.rob.diagram

4
< 40

1

10000

100

t1
t2

B,C

B,C

- Вырезать Ctrl+X
- Копировать Ctrl+C
- Вставить Ctrl+V
- Generate
- Send
- Проверить
- Проверить все
- Свойства Alt+ВВОД

Сгенерированный файл

```
RobotsLanguage21.js  RobotsLanguage21.rob*
<глобальные> ParallelN

1
2 function ParallelNode1_0() {
3   while (!(brick.sensor(4).read() < 40)) {
4     script.wait(10);
5   }
6   return;
7 }
8 function ParallelNode1_1() {
9   script.wait(10000);
10  return;
11 }
12 function main() {
13   brick.motor(B).setPower(100);
14   brick.motor(C).setPower(100);
15   Threading.startThread("t1", "ParallelNode1_0");
16   Threading.startThread("t2", "ParallelNode1_1");
17   while (brick.sensor(1).read() < 0) {
18     script.wait(10);
19   }
20   brick.motor(B).powerOff();
21   brick.motor(C).powerOff();
22   return;
23 }
24
```

Реализованные блоки

- Алгоритмические

(If, Switch, While, For, ParallelTasks, SubprogramCall, Subprograms)

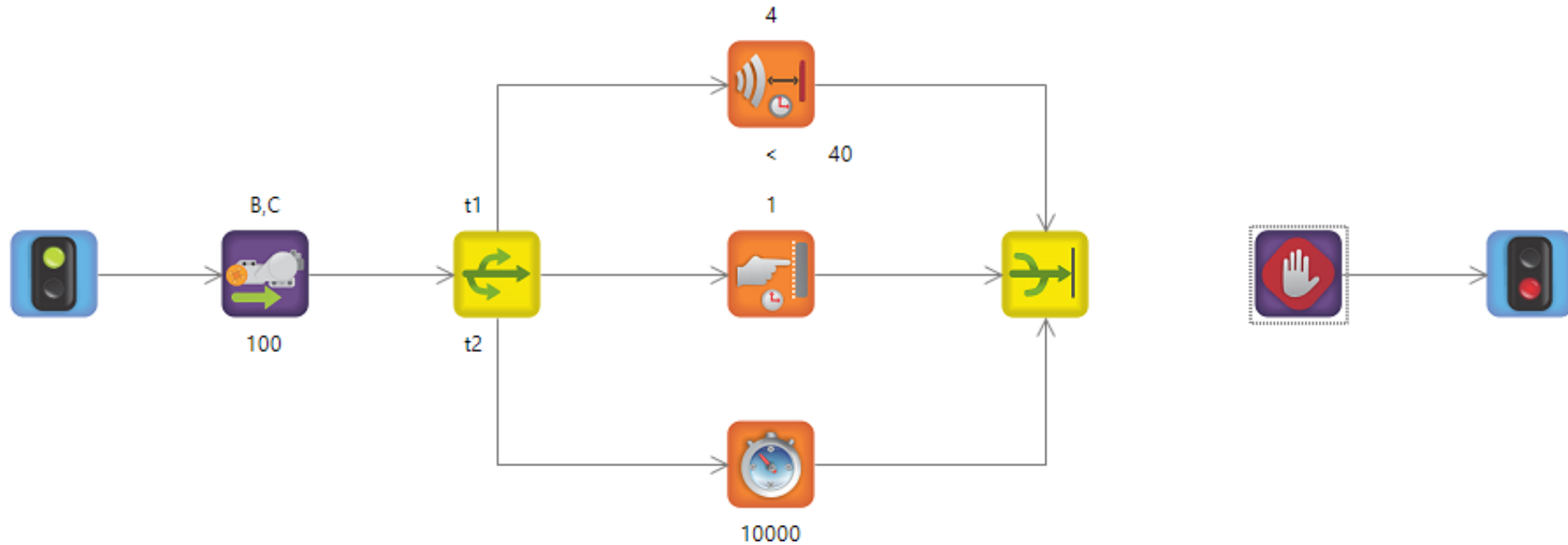
- Системные функции

(Motors, Timer, WaitTouch, WaitSensor...)

Валидатор

- Валидирующие методы, расширяющие доменные классы модели, с помощью partial-классов на языке C#
- Корректность входящих и исходящих связей
- Корректность меток
- Модифицированный алгоритм генератора данного решения для проверки структуры диаграммы

Диаграмма и окно ошибок



Список ошибок

Все решение | ✘ 3 Ошибки | ⚠ 0 Предупреждения | i 0 Сообщения | Сборка и IntelliSense

Код	Описание	Проект	Файл
✘	Incorrect source elements	Debugging	RobotsLanguage26.rob
✘	Incorrect target elements	Debugging	RobotsLanguage26.rob
✘	Fields must be non-empty	Debugging	RobotsLanguage26.rob

Апробация

- Произведено тестирование на 5 студентах
- Решение задачи
- Анкета System Usability Scale
(результаты относительно 100: 70, 75, 67.5, 70, 72.5)

Отзывы:

- + Привычный интерфейс Visual Studio
- + Автопрокладывание связей
- + Наличие отдельного валидатора
- Нужна документация
- Нужна загрузка Visual Studio для работы

Результаты

- Выполнен обзор возможностей среды программирования роботов TRIK Studio
- Выполнен обзор Modeling SDK for Visual Studio
- Разработана архитектура решения
- Реализована визуальная среда программирования роботов
- Произведена апробация решения