

# Создание режима быстрого прототипирования в CASE-системе QReal

Такун Евгения Игоревна, 545 гр.

Научный руководитель:  
ст. преп. Литвинов Ю.В.

Рецензент:  
к. ф.-м. н. Иванов А.Н.

# CASE-системы

- Охватывают обширные области поддержки проектирования программного обеспечения
  - Анализ программного обеспечения
  - Визуализация архитектуры
  - Передача знаний о системе
  - Документация
- Представители
  - Rational Rose
  - Altova UModel
  - Enterprise Architect
  - Visio

# metaCASE-системы

- предназначены для создания предметно-ориентированных визуальных языков и инструментов работы с ними с помощью метамоделирования
- Мета модель – описание синтаксиса языка
  - Может создаваться как вручную, так и с помощью специальных метаредакторов
  - Создается один раз и редко редактируется
- Существующие metaCASE-средства
  - MS DSL Tools
  - Eclipse GMF
  - MetaEdit+

# Задачи CASE-систем

Характер решаемой задачи	CASE-средства	metaCASE-средства
Задача хорошо решается с помощью стандартных языков общего назначения	Идеально подходят	Требует предварительного создания метамодели. После чего задача решается хорошо
Задача лучше решается с помощью предметно-ориентированных языков	Позволяют удобно анализировать только самые абстрактные вещи	Требует создания и поддержки изменений метамодели

# Постановка задачи

- Метамоделирование «на лету»
  - Динамическое изменение метамоделей
  - Скрыть от пользователя уровень метамоделей
  - Все изменения должны быть корректно обработаны системой
- Режим быстрого прототипирования
  - Возможность изменять тип добавленного на диаграмму элемента
  - Валидация и поддержка целостности и корректности системы после изменений

# Метамоделирование «на лету»

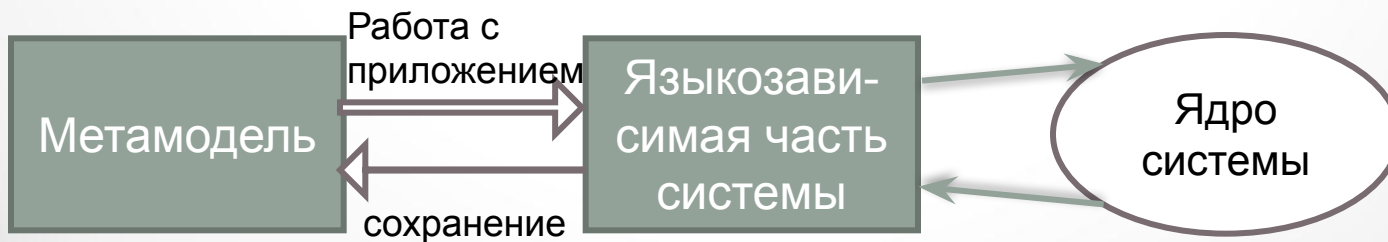
- MS DSL Tools, Eclipse GMF, QReal (раньше)

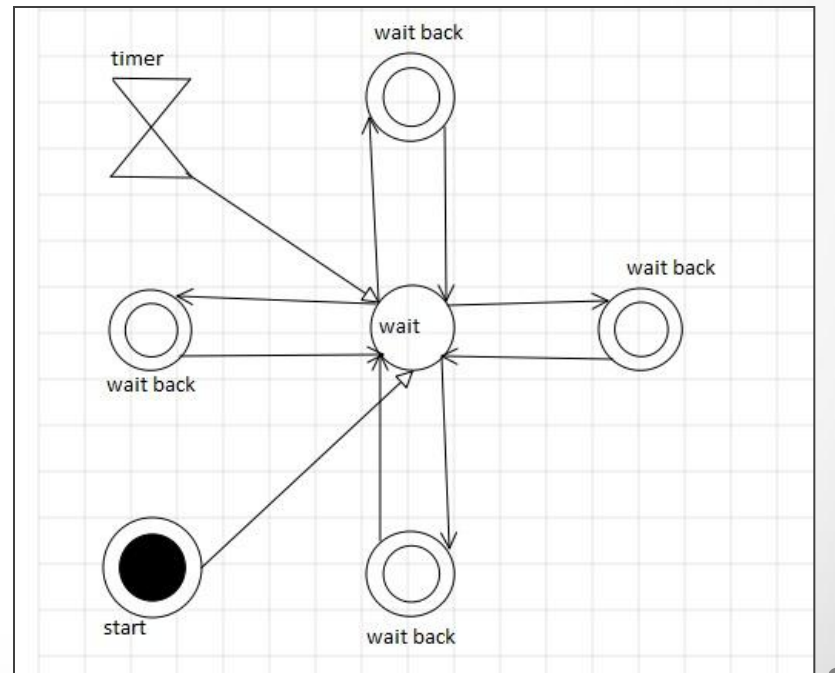
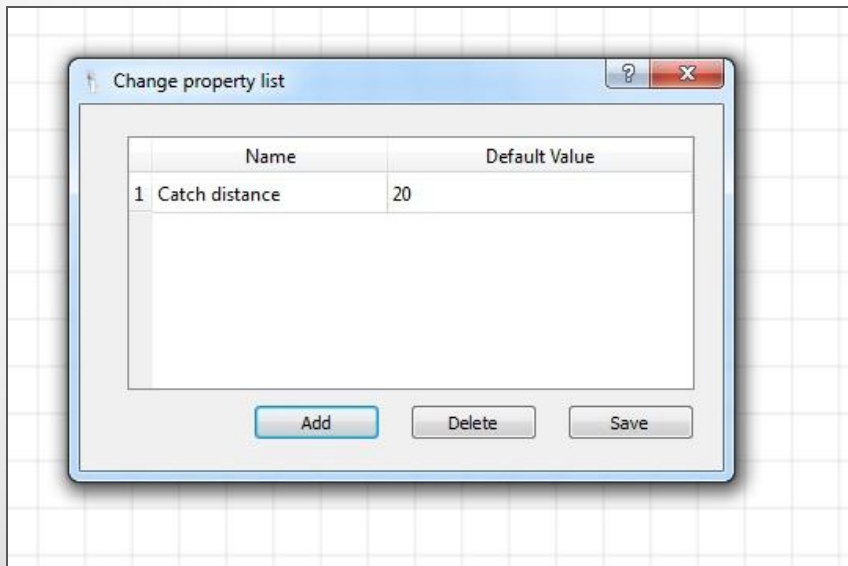
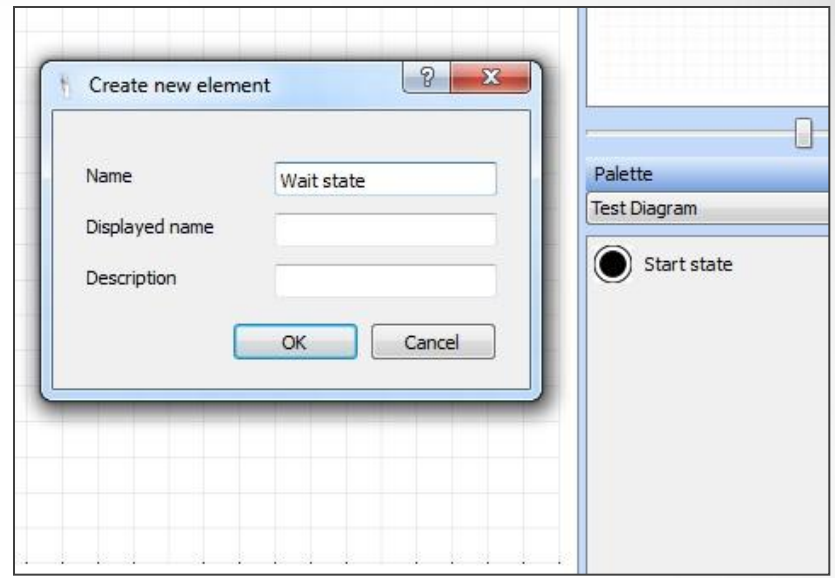
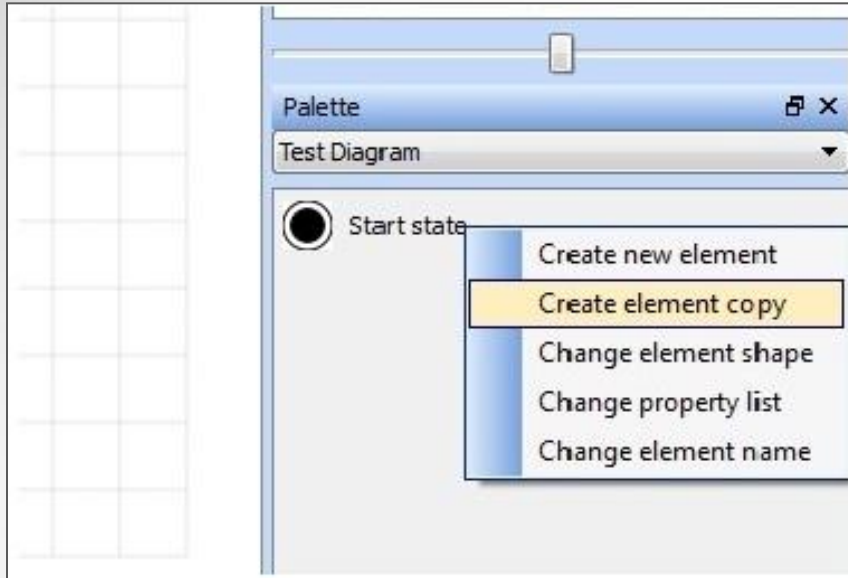


- MetaEdit+



- QReal





# Динамическая смена типа элемента

- Возможные последствия смены типа элемента
  - Потеря данных, которые были заданы для старого типа
  - Появление ограничений, заданных для нового типа
- Автоматические изменения
  - Изменение графического представления элемента
  - Изменение данных, на которые не наложены ограничения
  - Перенос значений свойств старого элемента в свойства нового элемента
- Изменения, требующие участия пользователя
  - Изменение ограничений, влияющих на поведение элемента



# Результаты

- Реализована возможность динамического изменения метамоделей
- Релизована возможность динамически изменять тип элемента
- Проведена апробация на реальной задаче