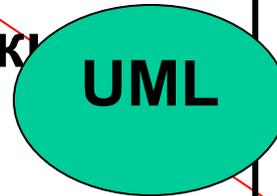


# UML = Unified Modeling Language

- **Язык**
  - Нотация
  - Семантика
  - Прагматика
- **Моделирования**
  - Modeling vs. Simulation
- **Унифицированный**
  - Универсальный? Единый?

# UML – ЭТО ЯЗЫК

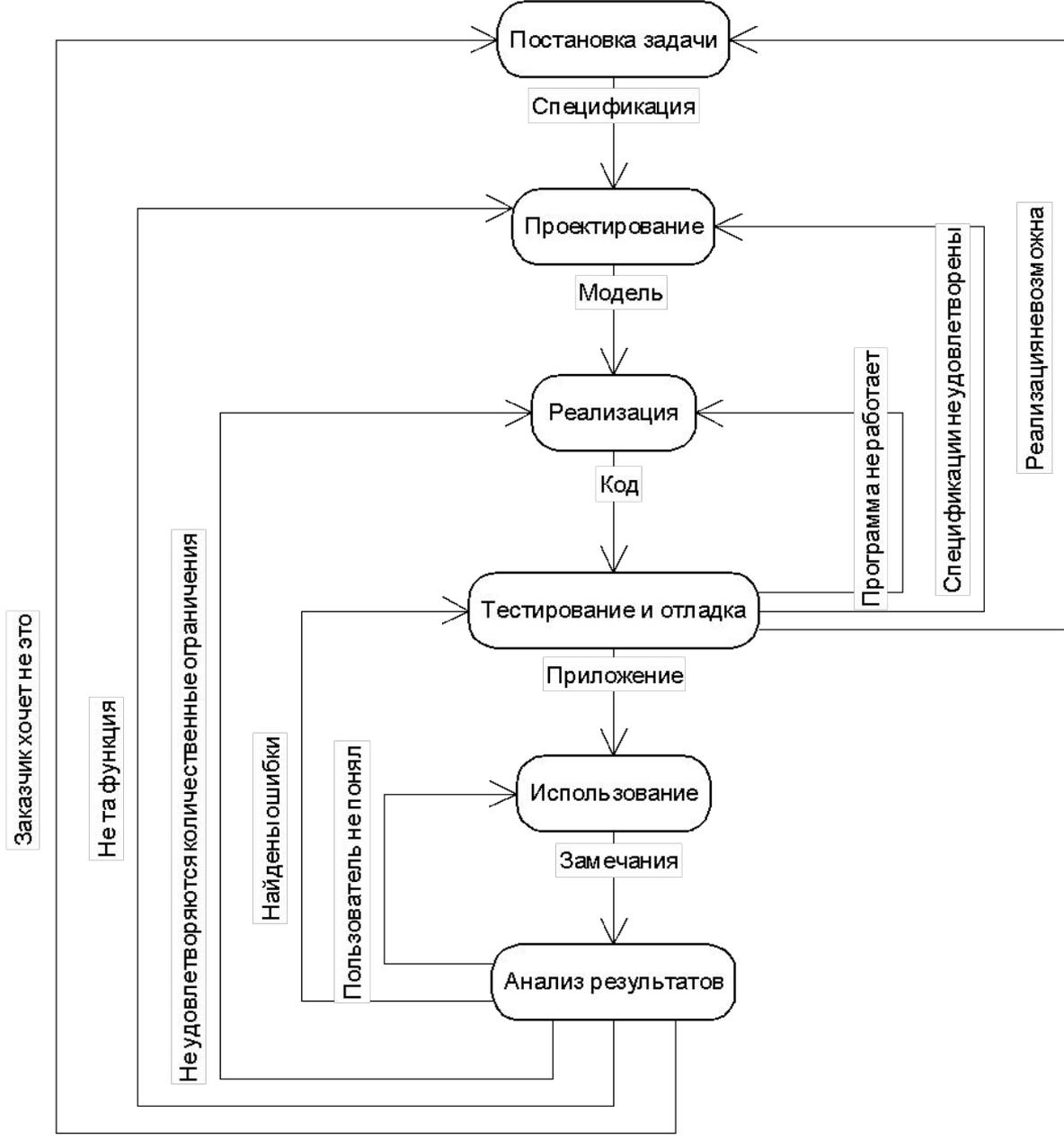
	Формальный	Неформальный
Искусственный	Паскаль	Эсперанто
Естественный	Математика	Русский



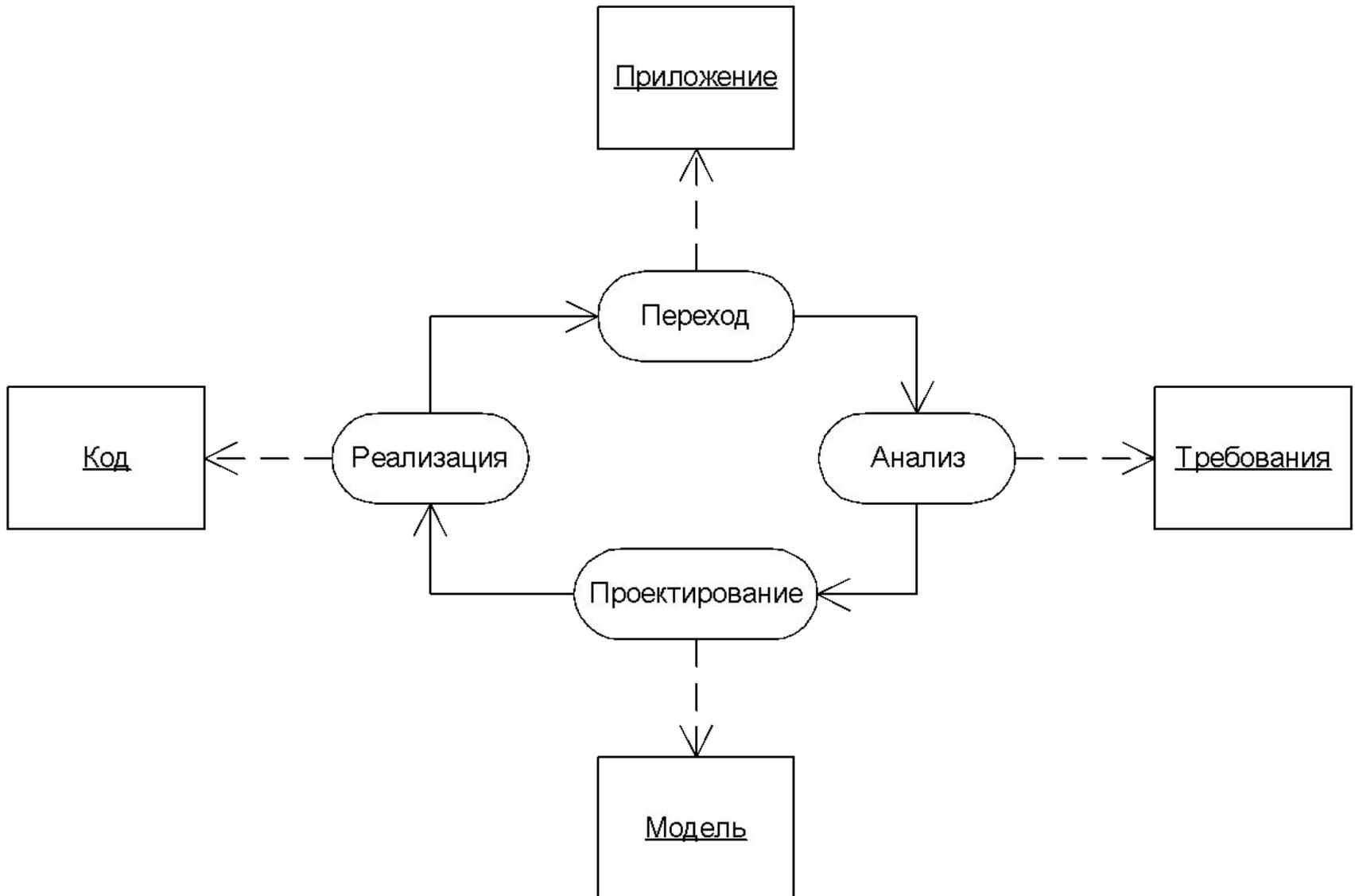
# **UML – это язык моделирования**

- **Жизненный цикл приложения**
- **Модель процесса разработки**
- **Модель – главный артефакт фазы проектирования итеративного процесса разработки**
- **Моделирование – составление и использование моделей**

# Жизненный цикл приложения



# Итеративный процесс разработки



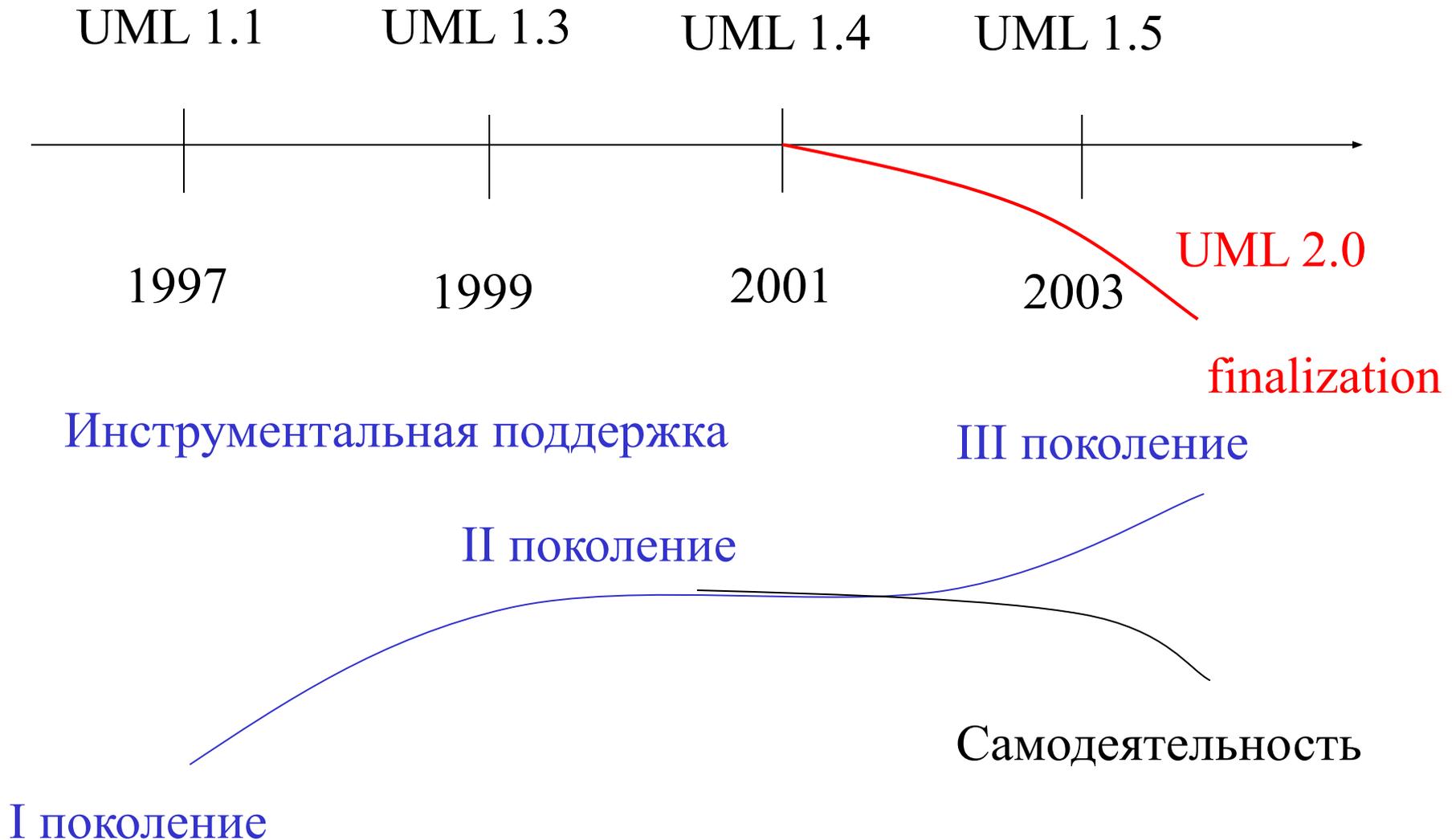
# **UML – это унифицированный язык моделирования**

- **Исторический контекст**
- **Развитие путем объединения и унификации**
- **Авторы и международное сообщество**

# Исторический контекст

- ...
- Петроглифы
- Блок-схемы
- Р-технология
- Диаграммы потоков данных (DFD)
- Диаграммы «сущность-связь» (ERD)
- Методология структурного анализа и проектирования (SADT)
- ...

# Текущее состояние



# Авторы UML



Буч



Рамбо



Якобсон

# Назначение UML

**Моделирование =**

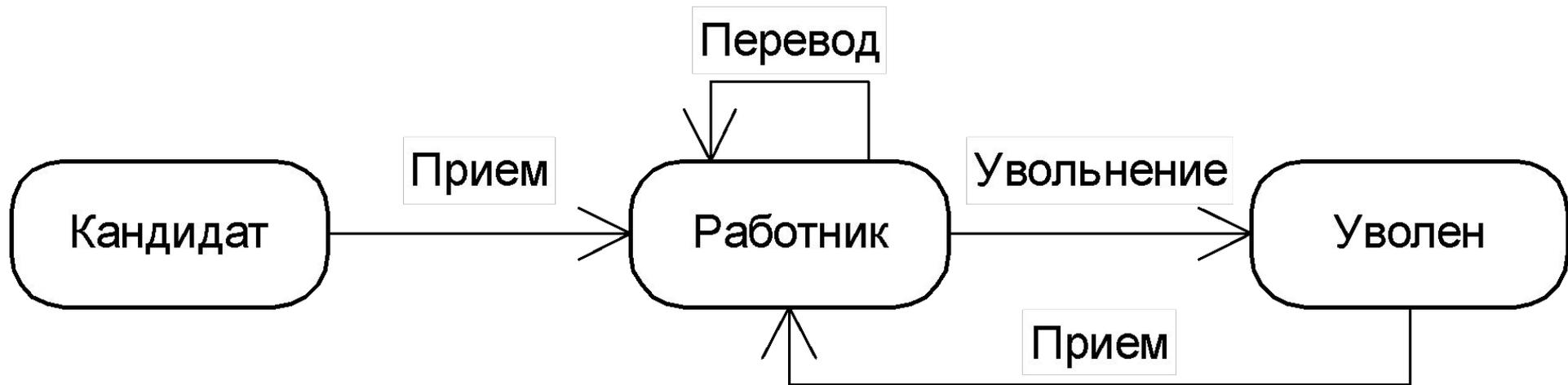
- **+ Спецификация**
  - формальная (но наглядная)
- **+ Визуализация**
  - для общения
- **+ Проектирование**
  - архитектуры (конструирование?)
- **+ Документирование**
  - всех артефактов

# Формальные спецификации

- **Спецификация = описание (программы)**
  - В понимании заказчика ≠
  - В понимании разработчика ≠
  - На самом деле
- **Полная формальная (математическая) спецификация возможна, но...**
  - Объем спецификации больше объема кода
  - Математики дороже программистов
  - Наилучшая спецификация программы – код
  - Формализация дороже автоматизации
- **Формализованная спецификация**

# Визуализация

- Сухой текст < текст с картинками < картинки с текстом = комиксы = UML



# Проектирование (construction)

- Автоматический синтез программ алгоритмически неразрешим (как массовая проблема)
  - Известны разрешимые подклассы
- Частичная генерация кода возможна
  - Опора на стереотипные компоненты
- Reverse engineering (???обратное проектирование) !анализ программ
  - Для недисциплинированных лентяев

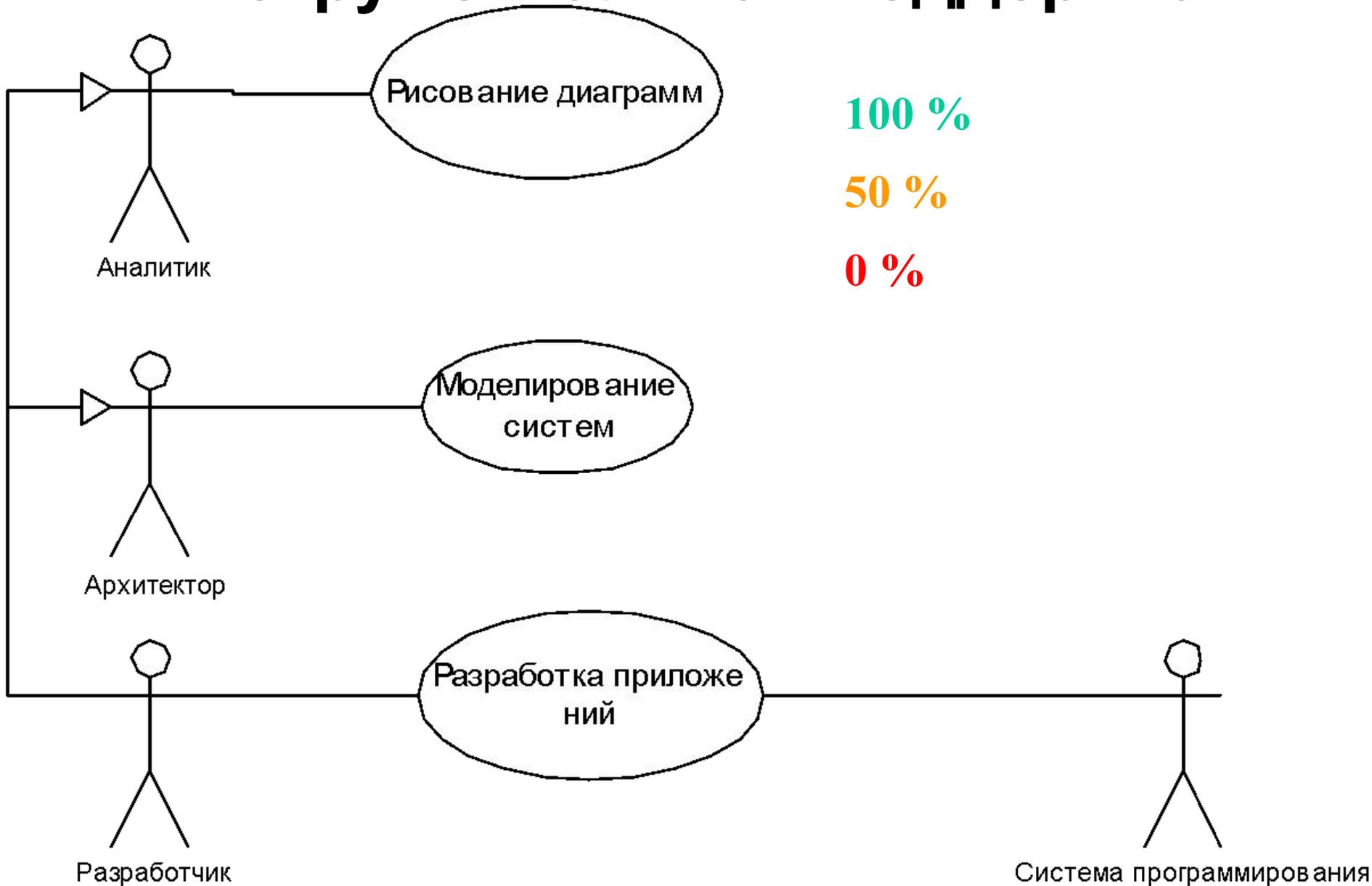
# Документирование

- Все элементы моделей UML могут содержать текстовые описания
- Почти все инструменты умеют собирать из них осмысленные документы
- Почти никто из программистов не пишет текстов

# Способы использования языка

- Рисование картинок !!!
- Обмен информацией !!
- Спецификация систем !
- Повторное использование архитектурных решений
- Генерация кода ?
- Simulation ??
- Верификация моделей ???

# Инструментальная поддержка



# Чем НЕ является UML

- **Языком программирования**
  - хотя генерация кода не возбраняется
- **Спецификацией инструмента (CASE)**
  - хотя инструменты подразумеваются и имеются (Together, Rose, Modeler, Visio, Argo)
- **Моделью процесса**
  - хотя модель необходима и имеется (Rational Unified Process → Unified Software Development Process)

# Терминология и литература

- **Перевод терминов UML на русский**
  - А.М. Вендров (UML Distilled)
  - А.А. Слинкин (UML User Guide)
  - А.В. Леоненков (Самоучитель)
  - К. и А. Максимовы (Справочник)
- **Доступная литература**
  - Специальный справочник (Питер, 2002)
  - Руководство пользователя (ДМК, 2000, Питер 2003)
  - Основы (Символ, 2002)
  - В кратком изложении (Мир, 1999)
  - Самоучитель (БХВ, 2001)

# Нотация

- **Нотация**
  - **Фигуры** – двумерные с внутренностью
  - **Линии** – одномерные с наружностью
  - **Значки** – двумерные с наружностью
  - **Тексты**
- **Инструменты рисования**
  - **Visio 2000** (отлично РИСУЕТ – не мешает)
  - **Rational Rose** (хорошо РИСУЕТ – старается)
  - **Visual Modeler** (удовлетворительно РИСУЕТ)
  - **Together** (красиво РИСУЕТ – иногда слишком)
  - **Argo UML** (неплохо РИСУЕТ, но устает и падает)
- **Стандартный стиль и вариации**
  - **Черно-белый без излишеств для печати**