



Московский институт электроники и  
математики им.А.Н.Тихонова (МИЭМ НИУ  
ВШЭ)

Проект 2142

Разработка практико-ориентированных заданий в обучении UML-  
программирования

Прокофьева Екатерина Николаевна



Диаграммы UML применяют в проектировании, презентациях, описании или создании документации. Важным аспектом становится использование инструментов создания UML-диаграмм на основе свободного ПО и разработки собственных программных приложений/решений. Linux-подобные ОС – прекрасная платформа для реализации автоматизированного инструментария.

В условиях реализации стратегии импортозамещения становятся важны возможности и преимущества отечественных операционных систем на примере Alt Linux в решении актуальных прикладных инженерных, статистических и исследовательских задач: запуск процессов в рамках базового функционала, разграничения прав, работы с репозиториями, установки библиотек и пакетов, функционирования образовательных, программных, статистических, графических, сетевых, ГИС и специализированных управленческих приложений.



## Актуальные тематические направления:

- Основы работа с большими данными в различных областях.
- Основы программирования и работы с кодом на базе пакетов и приложений.
- Работа с пространственными данными.
- Работа с социально-экономическими задачами.
- Работа с инженерно-техническими и научными задачами.
- Работа с графикой и видео. Работа с технической документацией.
- Работа с сетью и интернет-приложениями.
- Работа с нефинансовой и экологической отчетностью.
- Обеспечение безопасности данных.



## Приложения для ручной отрисовки UML-диаграмм:

- Microsoft Visio
- Lucidchart
- Draw.io (или Diagrams.net)
- StarUML



### Возможности:

- Редактирование вручную: Пользователь сам выбирает тип диаграммы (например, диаграммы классов, прецедентов, последовательностей) и размещает элементы на холсте.
- Богатый набор инструментов: Большинство таких приложений предоставляют набор предустановленных фигур (классы, интерфейсы, соединения и т.д.).
- Настройка стилей: Поддержка настройки шрифтов, цветов, толщины линий и других визуальных параметров.
- Совместимость: Возможность экспорта в популярные форматы (PDF, PNG, SVG) и интеграции с другими инструментами (например, Confluence, Jira).
- Поддержка совместной работы: Некоторые приложения (например, Lucidchart или Creately) позволяют редактировать диаграммы нескольким пользователям в реальном времени.



### Преимущества:

- Высокая гибкость в дизайне: диаграммы можно настраивать до мельчайших деталей.
- Удобство для небольших проектов или для пользователей без навыков программирования.
- Быстрое прототипирование: легко набросать схему для обсуждения с командой.

### Недостатки:

- Затратно по времени для сложных или больших диаграмм.
- Риск ошибок из-за отсутствия проверки синтаксиса или логики.
- Низкая автоматизация: изменения в модели требуют ручного обновления диаграммы.



## Программное обеспечение для автоматического построения UML-диаграмм

Возможности: • Генерация по коду: Программы могут анализировать исходный код (например, Java, C#) и автоматически строить диаграммы классов, последовательностей и т.д.

- Интеграция с IDE: Генерация диаграмм из среды разработки (например, IntelliJ IDEA, Eclipse).

- Анализ баз данных: Построение диаграмм на основе схемы базы данных (ER-диаграмм).

- Обратное проектирование (reverse engineering): Генерация диаграмм из существующего ПО.

- Синхронизация: Автоматическое обновление диаграмм при изменении кода.



## Примеры ПО:

- PlantUML
- Enterprise Architect
- Visual Paradigm
- Mermaid.js
- Umbrello





### Преимущества:

- **Быстрота:** автоматическое создание диаграмм экономит время.
- **Снижение риска ошибок:** диаграммы строятся на основе кода, что минимизирует расхождения между проектом и диаграммами.
- **Подходит для крупных проектов:** особенно полезно, если нужно визуализировать обширные системы.

### Недостатки:

- **Ограниченная кастомизация:** автоматические диаграммы могут быть сложны для настройки.
- **Требует знаний о синтаксисе языка моделирования или использования ПО.**
- **Может быть сложным для начинающих из-за необходимости настройки.**



Построение UML-диаграмм программно в неспециализированных приложениях  
Возможности:

- Написание скриптов: Создание диаграмм через код с использованием специальных библиотек (например, Graphviz).
- Гибкость интеграции: Поддержка интеграции с CI/CD, генерации диаграмм в документацию.
- Масштабируемость: Можно автоматизировать построение диаграмм для сложных систем или многократно используемых шаблонов.
- Взаимодействие с другими инструментами: Встраивание диаграмм в документы, отчёты или веб-приложения.

Пример -Python-библиотеки (например, diagrams)



## Преимущества:

- **Высокая автоматизация:** диаграммы генерируются из описаний, что минимизирует ручной труд.
- **Динамическое обновление:** изменения в коде автоматически отражаются на диаграммах.
- **Удобство работы в команде:** можно интегрировать диаграммы в процессы разработки.

## Недостатки:

- **Необходимость программных навыков.**
- **Более низкая визуальная кастомизация, чем в специализированных инструментах.**
- **Могут быть сложности в отладке больших диаграмм.**



Umbrello — это программа для моделирования на универсальном языке моделирования (UML) и генерации кода. Она позволяет создавать диаграммы программного обеспечения и других систем в формате UML, а также генерировать из диаграмм UML код на различных языках программирования.

Возможности:

- Поддерживаемые форматы: XMI
- Поддерживаются несколько типов диаграмм: вариант использования, класс, последовательность, коммуникация, состояние, активность, компонент, развёртывание, отношения сущностей



Преимущества:

Комплексность

Адаптивность

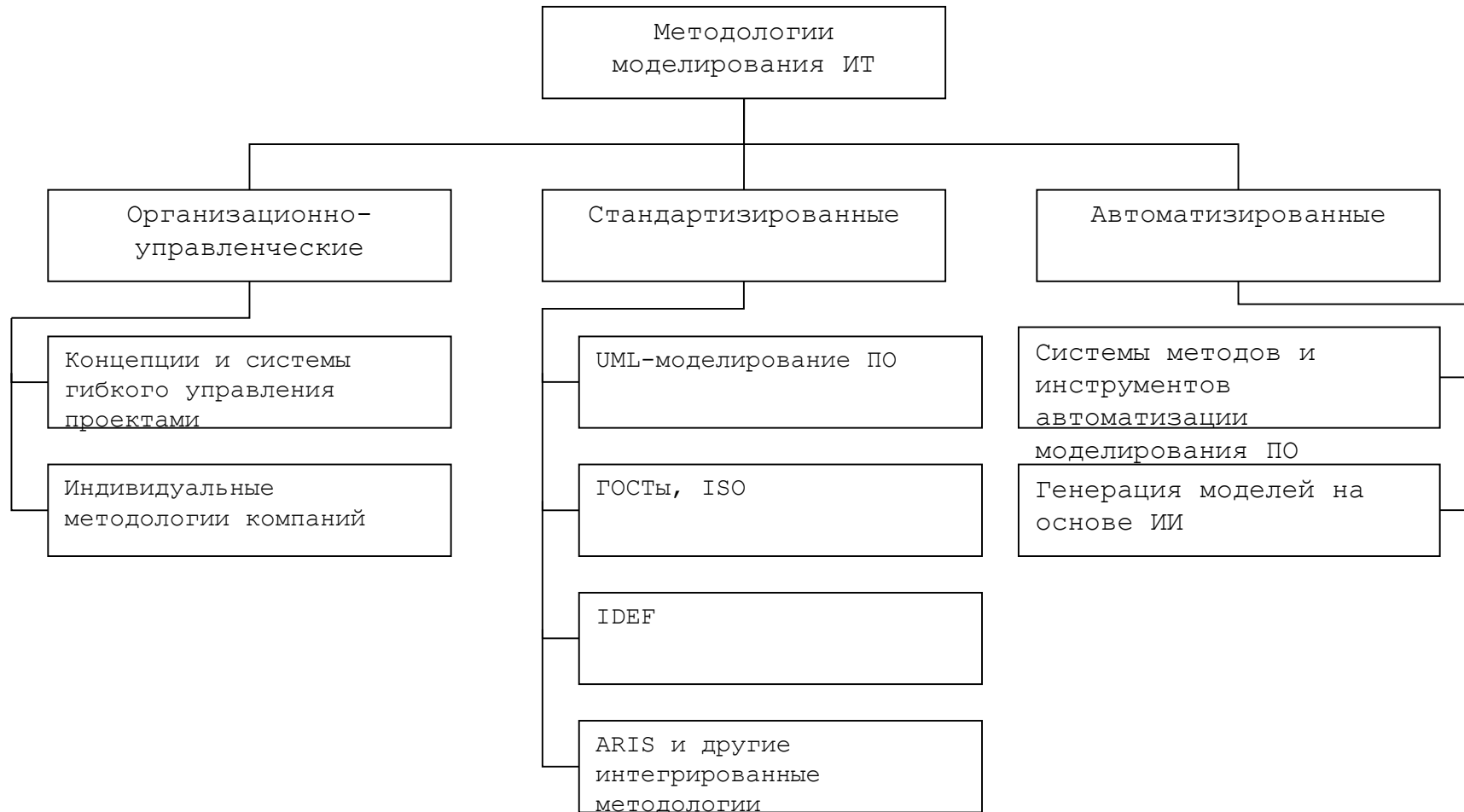
Широкие возможности

Недостатки:

Сложность

Длительность работы

# Методологии моделирования ИТ развития





На практике UML трансформируется в универсальный язык междисциплинарной коммуникации, становится ключевым вектором в методологии моделирования ИТ развития организации.

Особенно востребованны USE CASE DIAGRAMS – диаграммы прецедентов, в управлении бизнес-процессами и согласовании взаимодействия сложных подсистем.

UML выступает важным структурным инструментом построения методологии TOGAF10, с возрастанием роли ИТ-процессов (помимо BPMN)



На практике помимо Umbrello на базе Linux-подобных систем целесообразно использовать варианты ряда инструментов  
Argo, StarUML, Papyrus etc

alt linux ALT LINUX TEAM

RU

Пакеты Поиск пакета по и... →

Пакет perl-UML-Class-Simple:  
Информация

< Информация Изменения >

Исходный пакет: perl-UML-Class-Simple  
Версия: 0.22-alt3  
Собран: 20 мая 2019 г. 20:01  
Категория: **Разработка/Perl**

Сообщить об ошибке в пакете

Домашняя  
страница: <http://search.cpan.org/dist/UML-Class-Simple/>

Лицензия: Artistic and GPL  
О пакете: Render simple UML class diagrams, by loading the code  
Описание:  
UML::Class::Simple is a Perl CPAN module that generates UML class diagrams (PNG format, GIF format, XMI format, or dot source) automatically from Perl

Репозиторий: p10\_e2k »

## Пример задания

Papyrus - это инструмент для моделирования UML, доступный на Linux. С его помощью вы можете создавать диаграммы классов, определять атрибуты и методы классов, а также связи между ними. Давайте рассмотрим пример создания диаграммы классов в Papyrus:

1. Запустите Papyrus на своей системе Linux.
2. Создайте новый проект и выберите UML-профиль.
3. Добавьте новую диаграмму классов в проект.
4. Добавьте классы на диаграмму и определите их атрибуты и методы.
5. Установите связи между классами, используя различные типы связей (наследование, ассоциация и т. д.).
6. Сохраните и экспортируйте диаграмму в нужном формате.





## Рекомендации по использованию UML на Linux

На основе особенности выполняемой задачи и анала преимуществ/ недостатков следует выбрать подходящий инструмент для моделирования UML на Linux.

Проверить, есть ли поддержка нужных вам типов диаграмм и возможность экспорта/импорта диаграмм.

Полезно аккумулировать инструментальные практики в создании различных типов диаграмм.

Использовать комментарии к моделям и элементам диаграмм, чтобы документировать свою работу и делиться информацией с коллегами.

Следить за обновлениями инструментов UML на Linux. Новые версии могут добавлять новые возможности и исправлять ошибки.

Регулярно мониторить актуальные примеры использования UML на Linux. Это помогает лучше понять его применимость на самых различных проектах с учетом множества особенностей.