Программная инженерия
Модели разработки ПО

Модели разработки ПО

Можно выделить 3 основных модели разработки ПО:

 Водопадная модель

 Поэтапная (эволюционная) разработка

 Компонентно-ориентированная разработка

Часто эти модели объединяются в рамках одного проекта (подсистемы могут создаваться посредством различных подходов)

Водопадная модель

Фазы водопадной модели

- Результатом каждой фазы в водопадной модели является один или несколько утвержденных документов.
- Последующая фаза не может начаться пока предыдущая не завершена.
- Но только в идеальном мире процесс разработки линеен и не обладает обратными связями
- => в реальном мире в водопадной модели подразумевается несколько итераций, после которых результат «замораживается» и происходит переход ко следующей фазе.

Достоинства и недостатки

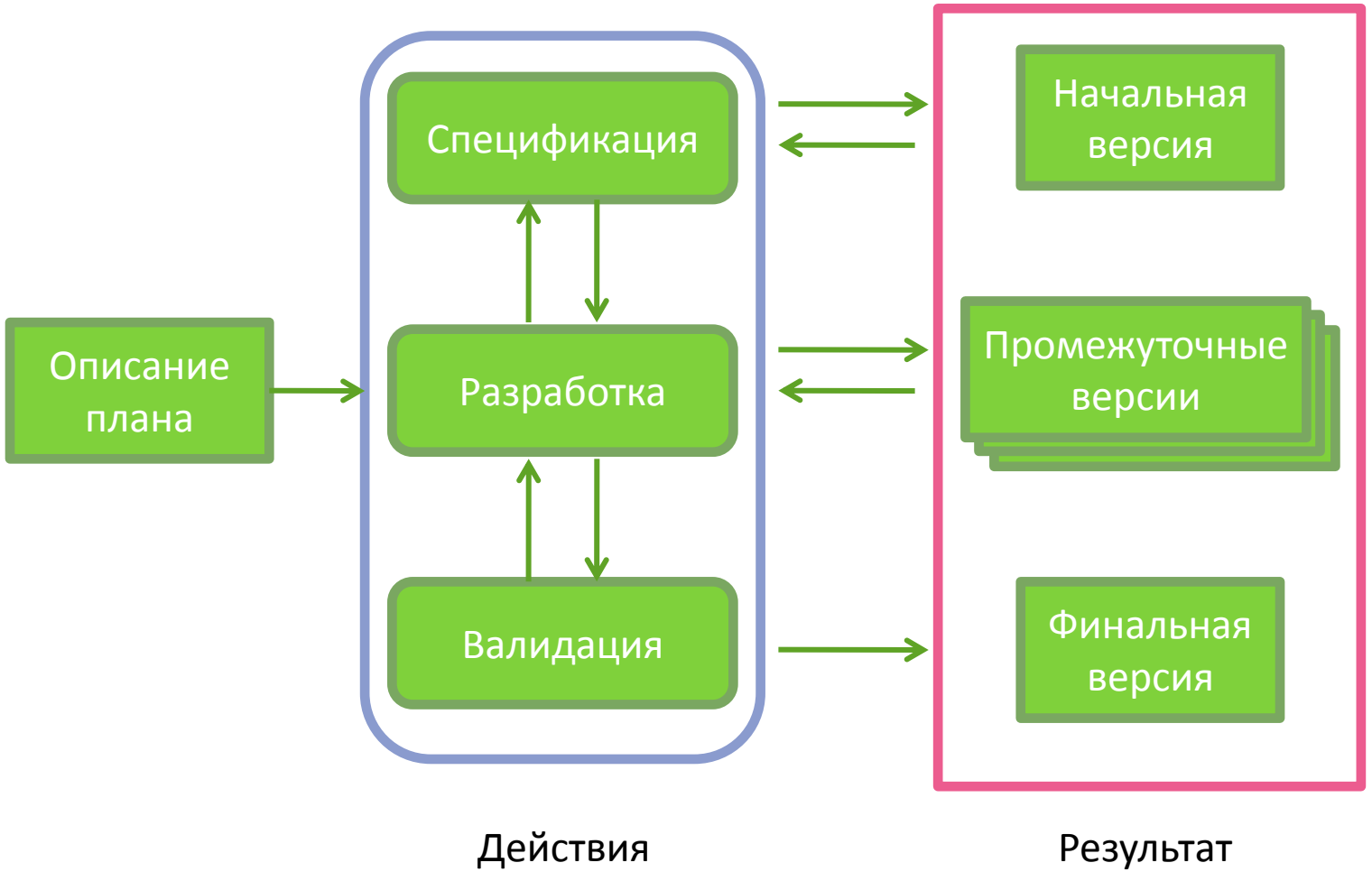
Достоинства:

- Очень формализованная, в результате каждой фазы формируется утвержденный документ
- Соответствует стандартным моделям инженерной разработки.

Недостатки:

- Отсутствие гибкости
- Все договоренности – на ранней стадии, соответственно нет возможности подстроиться под изменяющиеся требования пользователя.

Поэтапная разработка



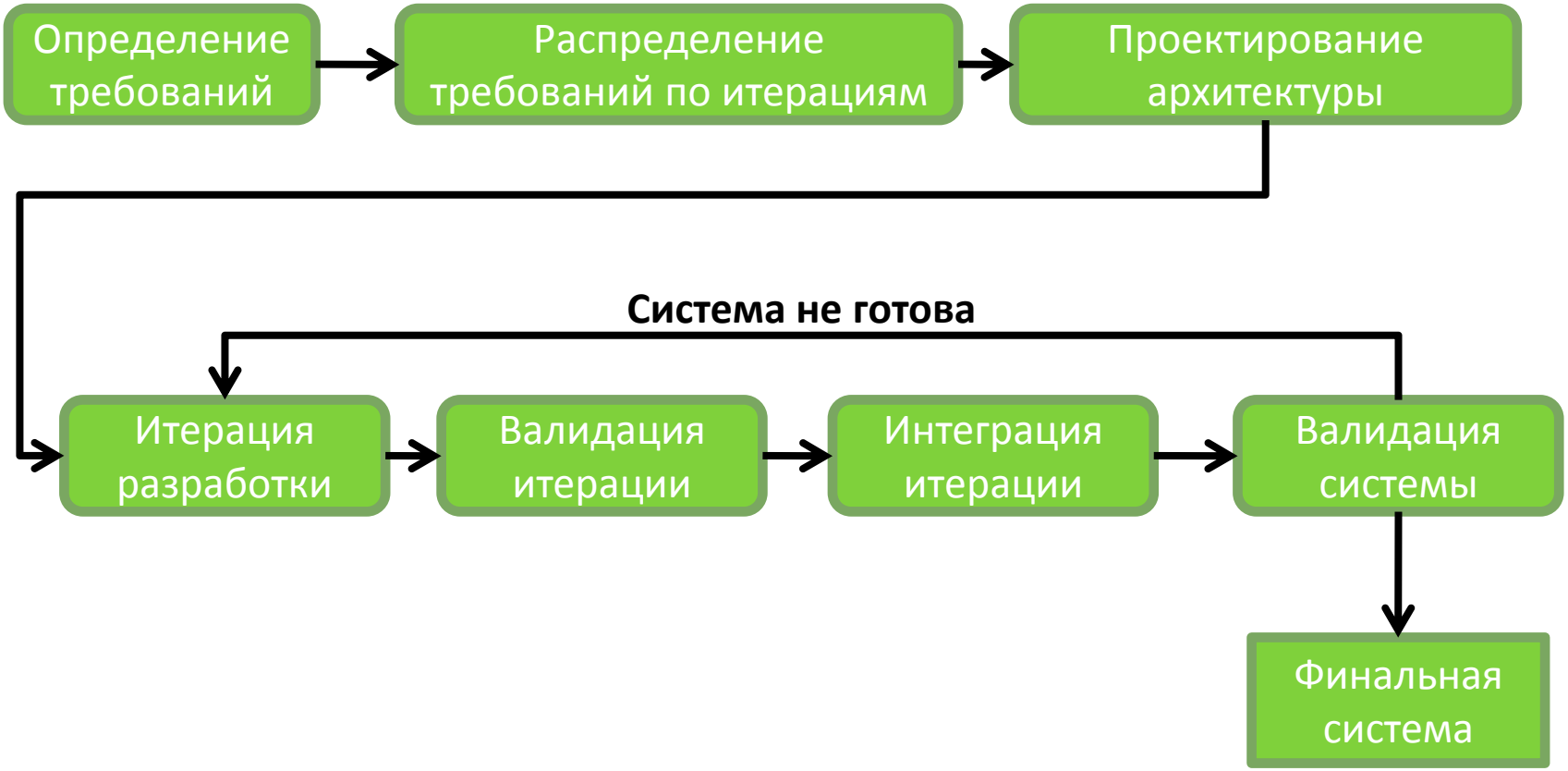
Поэтапная разработка

- ***Постепенная (эволюционная) разработка***: тесная работа с клиентом, для выяснения требований и постепенное создание финальной версии.
  - В начале – те части, которые понятно как делать. Развитие системы: добавление новых возможностей, предлагаемых клиентом.

Поэтапная разработка

- ***Разработка на основе прототипов***: постепенно создаются прототипы разрабатываемой системы, для того чтобы понять конкретные требования заказчика.
  - Как только прототип выполнил свою задачу (требования заказчика стали понятны), он выбрасывается. Его код не используется в конечном продукте.

Инкрементальная разработка



Инкрементальная разработка

- В начале описываются и ранжируются по значимости базовые сервисы, которые должна предоставлять система.
- Определяется количество итераций, за которое должны создать готовую систему.
- В начале каждой итерации формируются детальные требования к набору сервисов, которые будут созданы в ее рамках (дальнейшее их изменение невозможно).

Достоинства

- Не надо ожидать, пока вся система будет полностью готова. Результат первой итерации может быть использован сразу.
- Ранние итерации могут служить прототипами и давать основу для построения требований ко следующим итерациям.
- Наиболее важные сервисы создаются в первую очередь, соответственно обеспечивается их всестороннее тестирование на более поздних этапах.

Недостатки

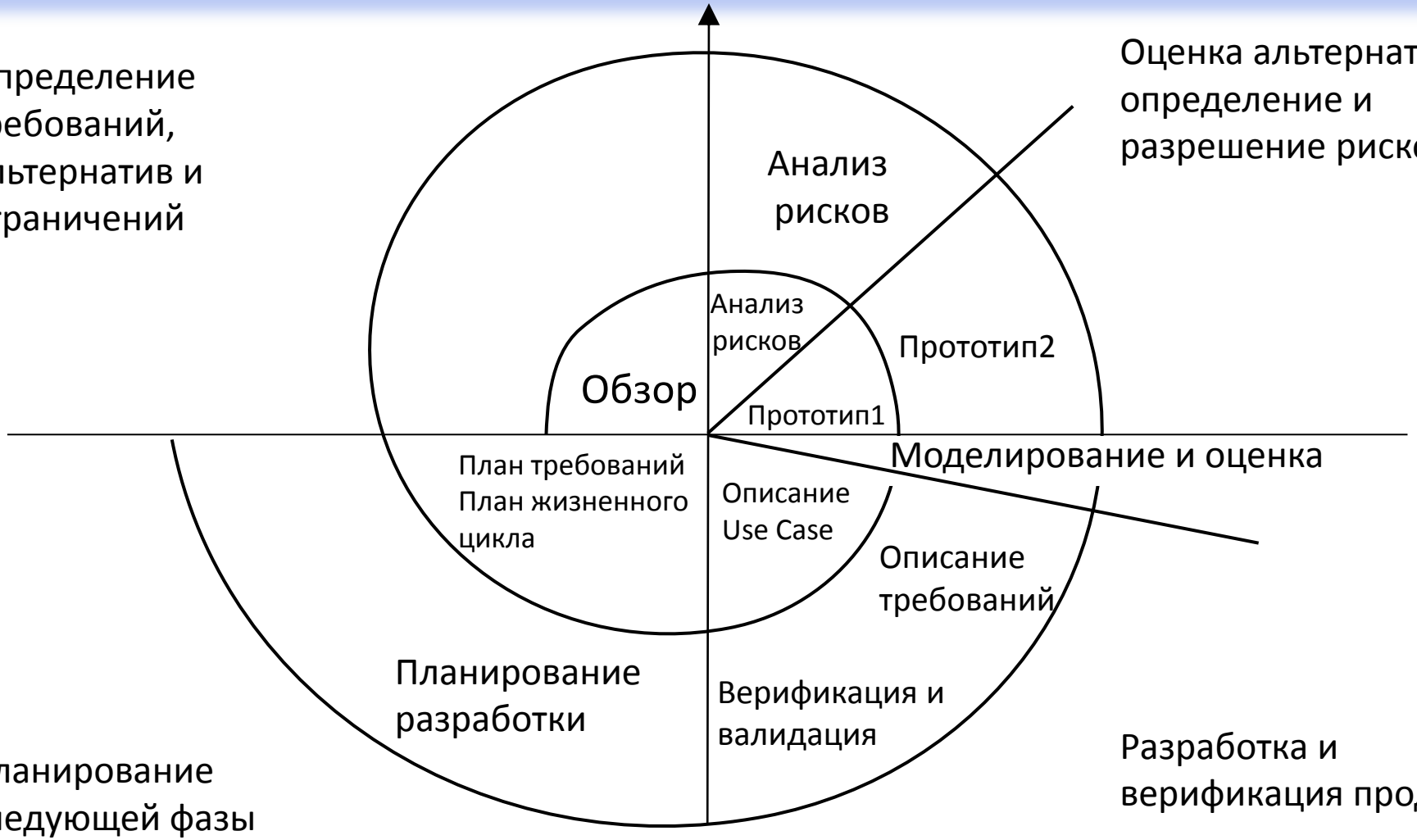
- Необходимо обеспечить малый объем работ в рамках итерации (не более 20 000 строк кода). При этом, должна обеспечиваться определенная функциональность.

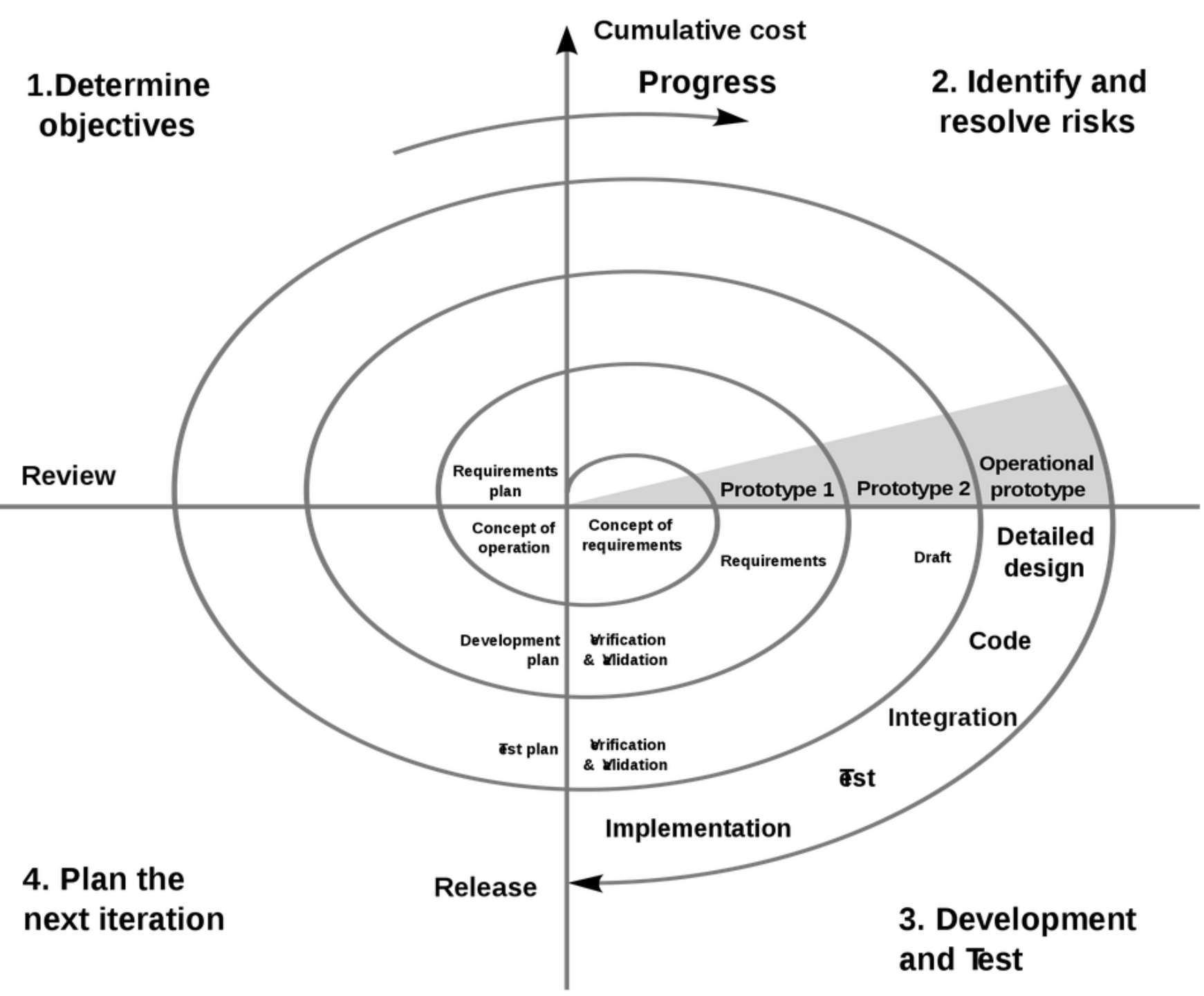
- Но множество систем обладают большим набором базовых сервисов, необходимых всем дальнейшим надстройкам и разделить их на итерации не всегда легко.

Спиральная разработка

- Каждый шаг спирали – фаза процесса разработки ПО (постановка задачи, определение требований, дизайн архитектуры и т.д.)
- В каждом шаге выделяют 4 сектора:
  - Определение требований
  - Оценка и уменьшение рисков
  - Разработка и валидация
  - Планирование

Спиральная разработка




Cumulative cost
Progress
1.Determine objectives
2. Identify and resolve risks
Review
Requirements plan
Prototype 1
Prototype 2
Operational prototype
Concept of operation
Concept of requirements
Requirements
Draft
Detailed design
Development plan
Code
Integration
Test plan
Test
Implementation
Release
4. Plan the next iteration
3. Development and Test

Риски

- **Основное отличие спиральной разработки** от других методов разработки ПО – это явная оценка рисков.
- Риск – это вероятность того, что что-то может пойти не так как хотелось бы (например, при использовании нового языка программирования есть риск, что существующие компиляторы **не** будут создавать высокоэффективный код).

Применение поэтапной разработки

- Поэтапная разработка более гибкая, чем водопадная модель, позволяет легче подстраиваться под изменяющиеся требования заказчика
- Хорошо подходит для небольших и средних по размерам проектов (порядка 500 000 строк кода)

Итерации позволяют

- контролировать и корректировать ход выполнения проекта
- эффективнее работать с изменяющимися требованиями;
- эффективнее работать с рисками
- на ранних этапах оценивать потенциальные характеристики системы

Проблемы поэтапной разработки

- **Процесс разработки не виден**
  - Ради скорости разработки в жертву приносится формальность: практически отсутствует документация, производимая на каждом этапе водопадной модели
- **Системы часто слабоструктурированы**
  - Постоянные изменения приводят к повреждению структуры ПО. Поддержка и дальнейшее изменение становится дорогой и сложной процедурой.

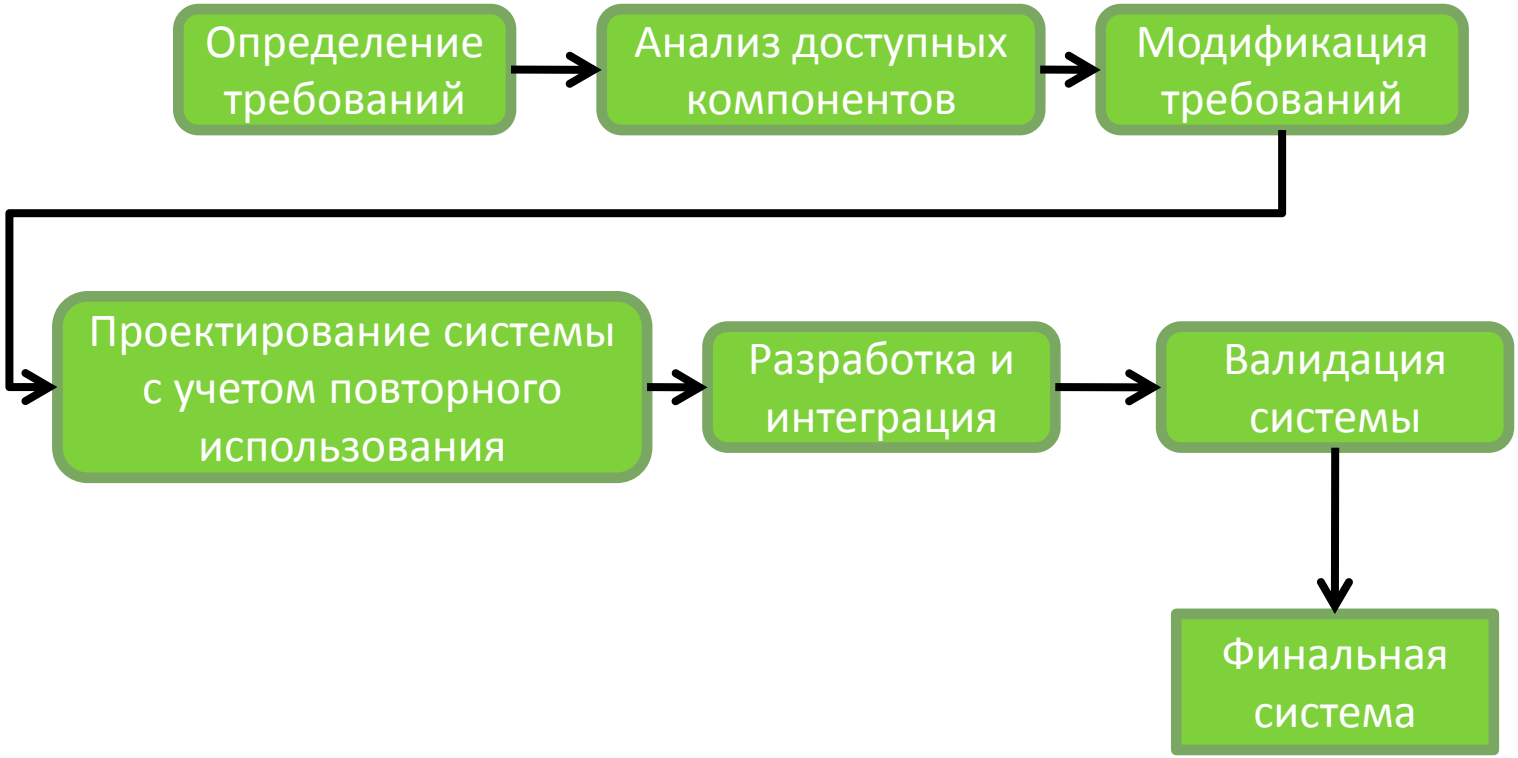
Компонентно-ориентированная разработка

- Компонентно-ориентированная разработка основывается на формальном закреплении повторного использования кода.
- **Программный компонент** – это автономный элемент программного обеспечения, предназначенный для *многократного использования*, который может распространяться для использования в других программах в виде скомпилированного кода.

Компонентно-ориентированная разработка

- Компонентно-ориентированный подход – развитие объектно-ориентированного. Создан для проектирования и реализации *крупных и распределенных программных систем (корпоративных приложений)*
- ***С точки зрения КОП*** программная система – это набор компонентов с четко определенным интерфейсом.
  - Изменения в систему вносятся путем создания новых компонентов или изменения старых.
  - **Наследование реализации запрещено. Наследуется только интерфейс.**

Компонентно-ориентированная разработка



Достоинства и недостатки компонентно-ориентированной разработки

- Достоинства
  - Уменьшается объем ПО, которое необходимо разработать => уменьшается цена и риски.
  - Уменьшается время разработки
- Недостатки
  - Компромиссы при выработке требований могут привести к тому, что реальные требования пользователей не будут учтены.
  - Контроль над системой может быть потерян при появлении новых версий используемых компонентов