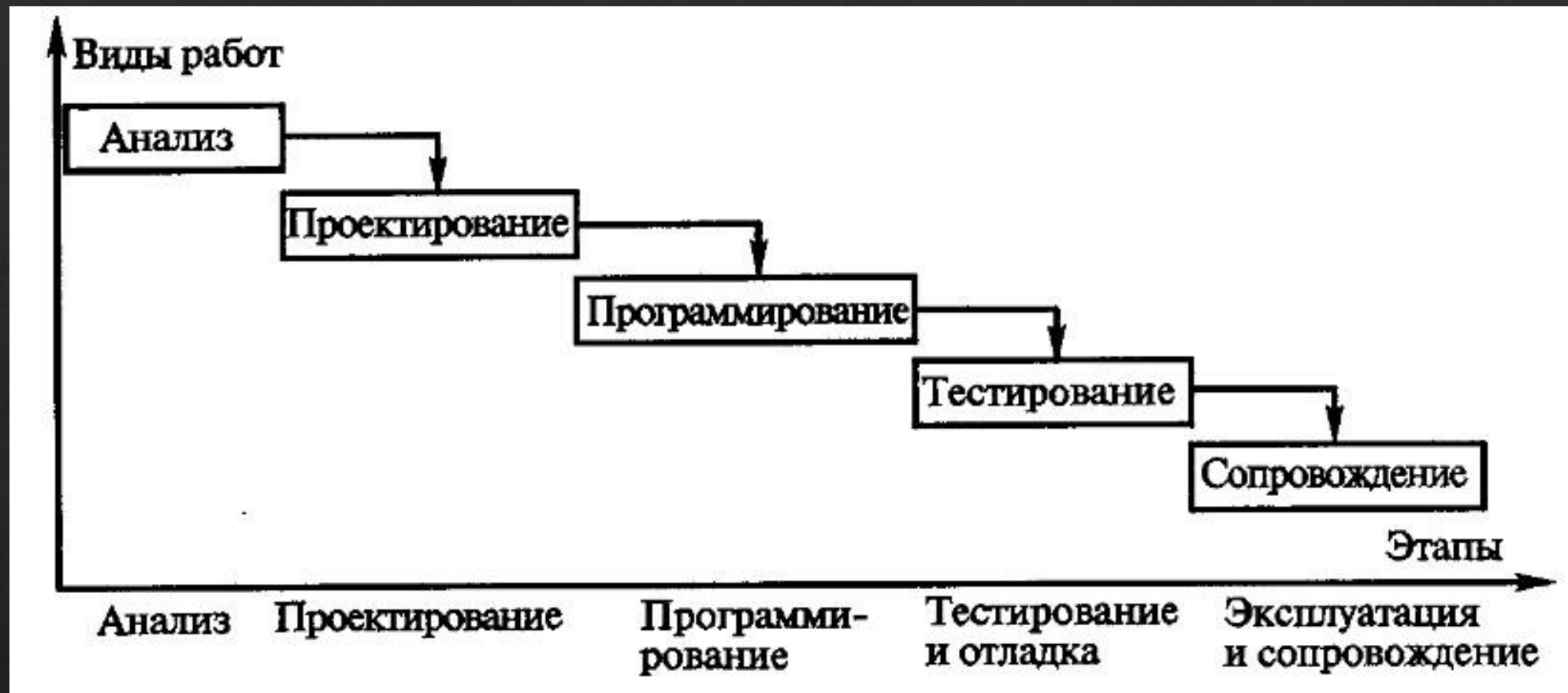


Современная Методология Разработка программного обеспечения

Методы разработки программного обеспечения

1. Каскадный/ «водопад»
2. Каскадно-возвратный
3. Каскадно-итерационный
4. Каскадный с перекрывающимися видами работ (англ. waterfall with overlapping)
5. Каскадный подход с подвидами работ (англ. waterfall with subprocesses)
6. Спиральная модель (spiral model)

Каскадный / «водопад»

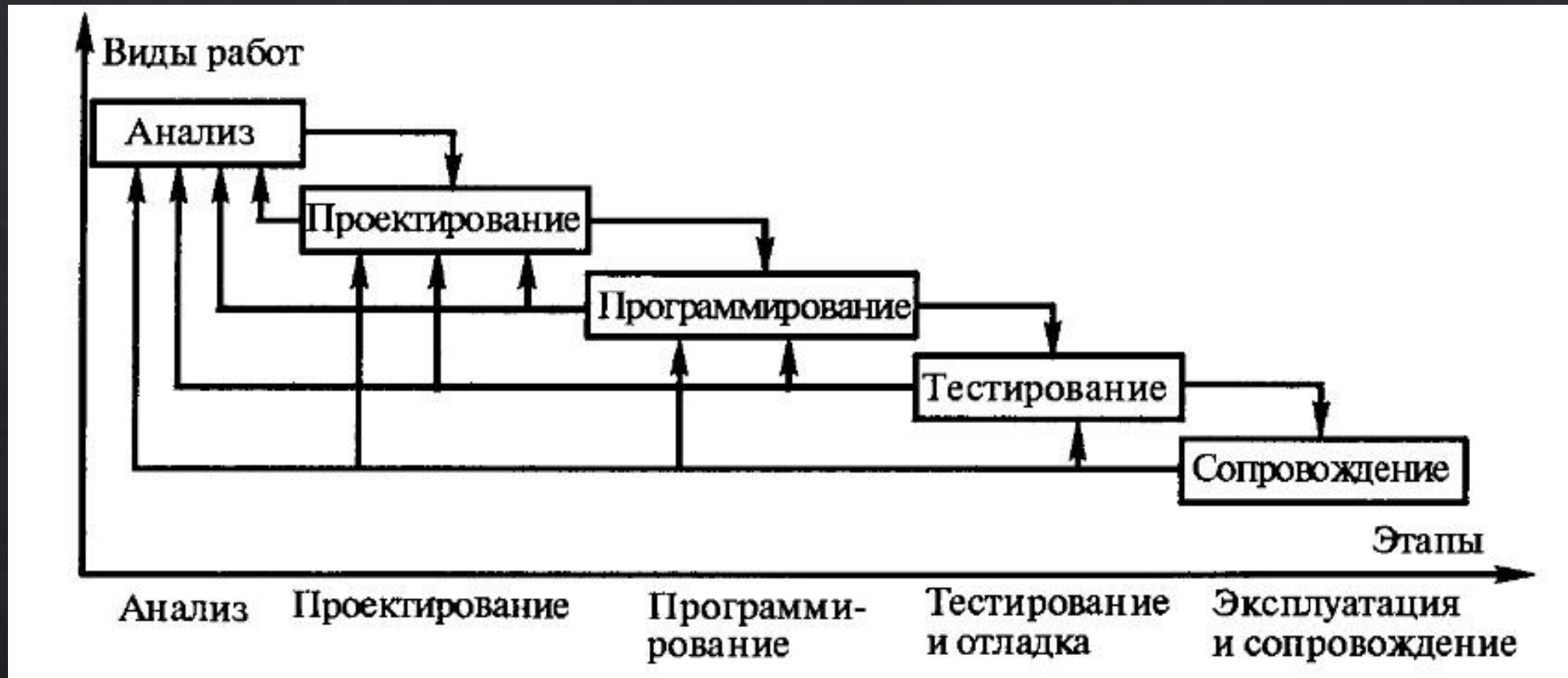


Достоинства: полная документация, простота

Недостатки: невозможность вернуться на предыдущий этап

Применение: для разработки небольших программ

Каскадно-возвратный

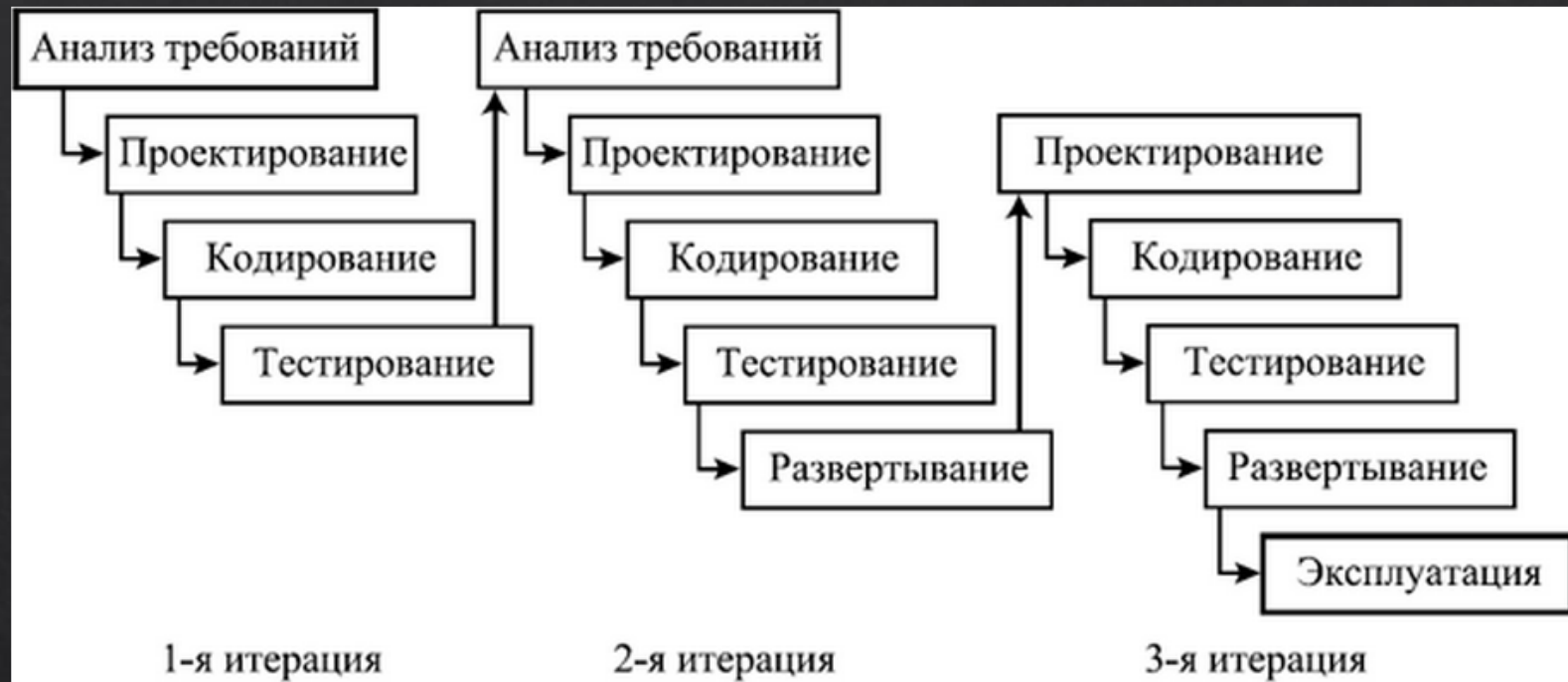


Плюсы: Возможность вернуться на любой уровень для внесения изменений

Минусы: «Отличное – враг хорошего»; затягивание процесса разработки

Применение: для программа среднего размера

Итеративная (эволюционная) модель



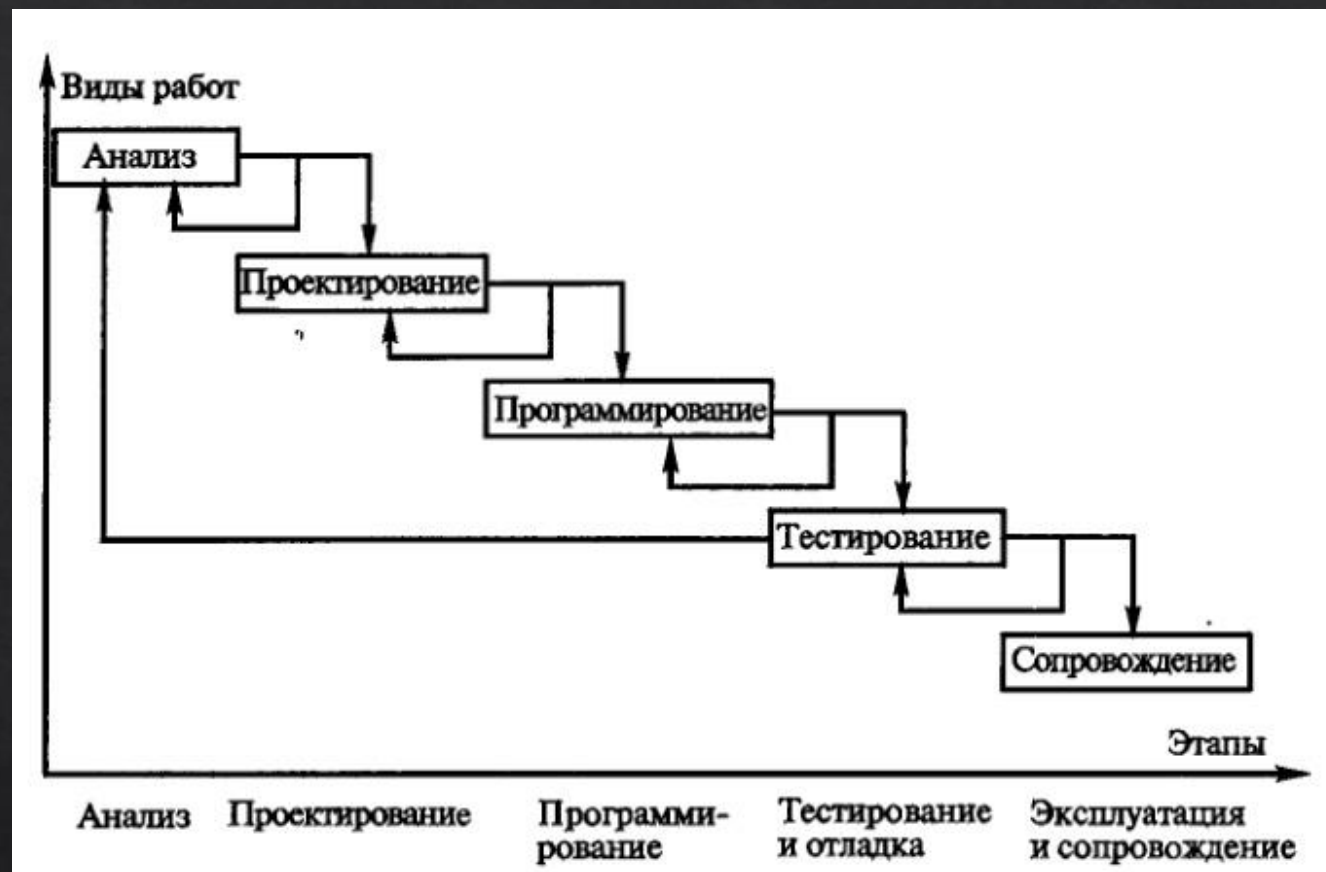
Плюсы: Возможность «отката» при обнаружении ошибки;

Минусы:

- 1.Отсутствие в течение долгого времени понимания целостного проекта.
- 2.При итерациях приходится отбрасывать часть сделанной ранее работы;
- 3.«Халтура», т.к. у разработчики знают, что огрехи можно исправить позже.
- 4.Затруднено долгосрочное планирование.

Применение: для больших программ

Каскадно-итерационный

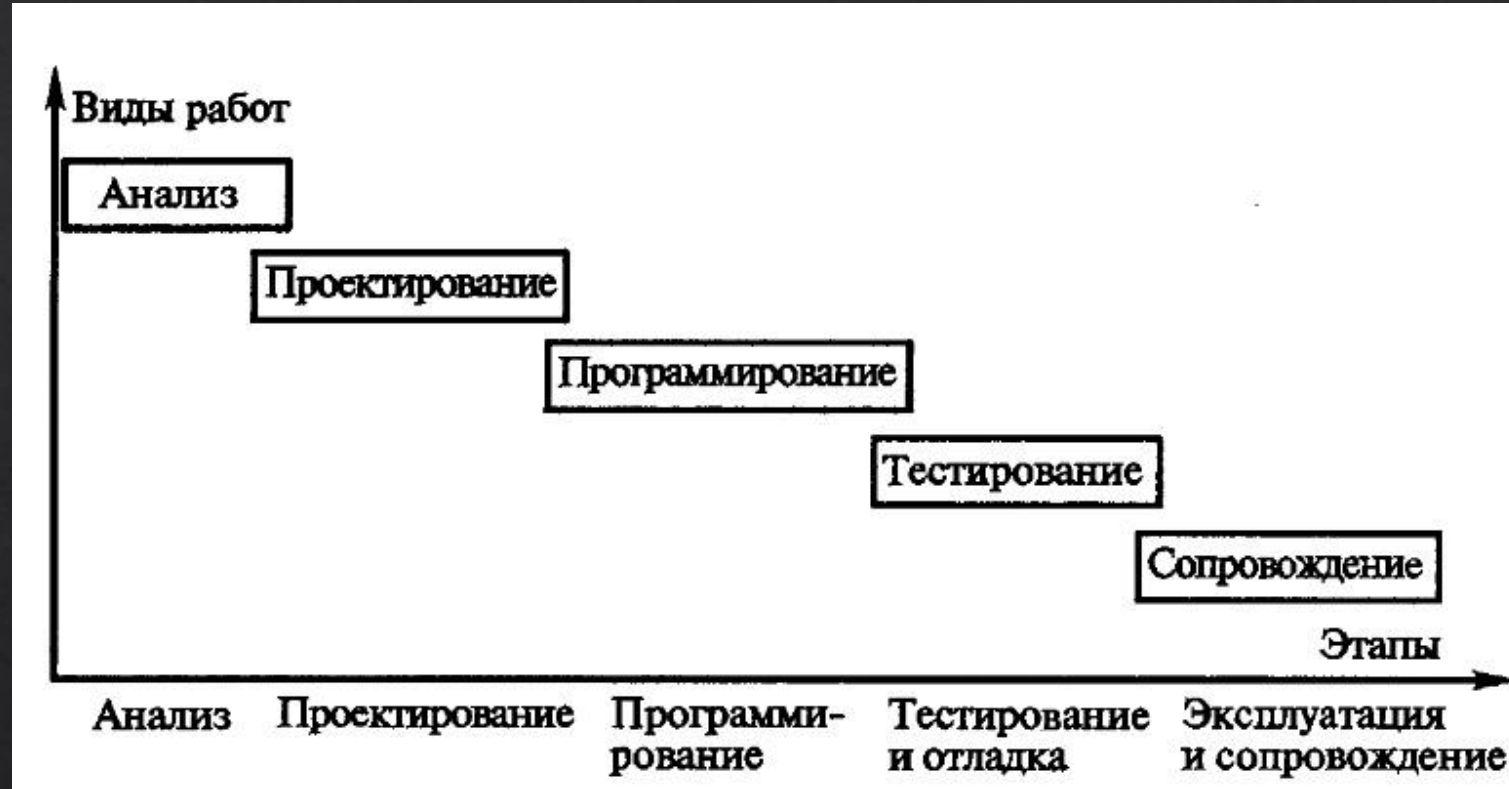


Плюсы: Возможность вернуться на предыдущий уровень для внесения изменений

Минусы: «Отличное – враг хорошего»; затягивание процесса разработки

Применение: для программа среднего размера

Каскадный с перекрывающимися видами работ

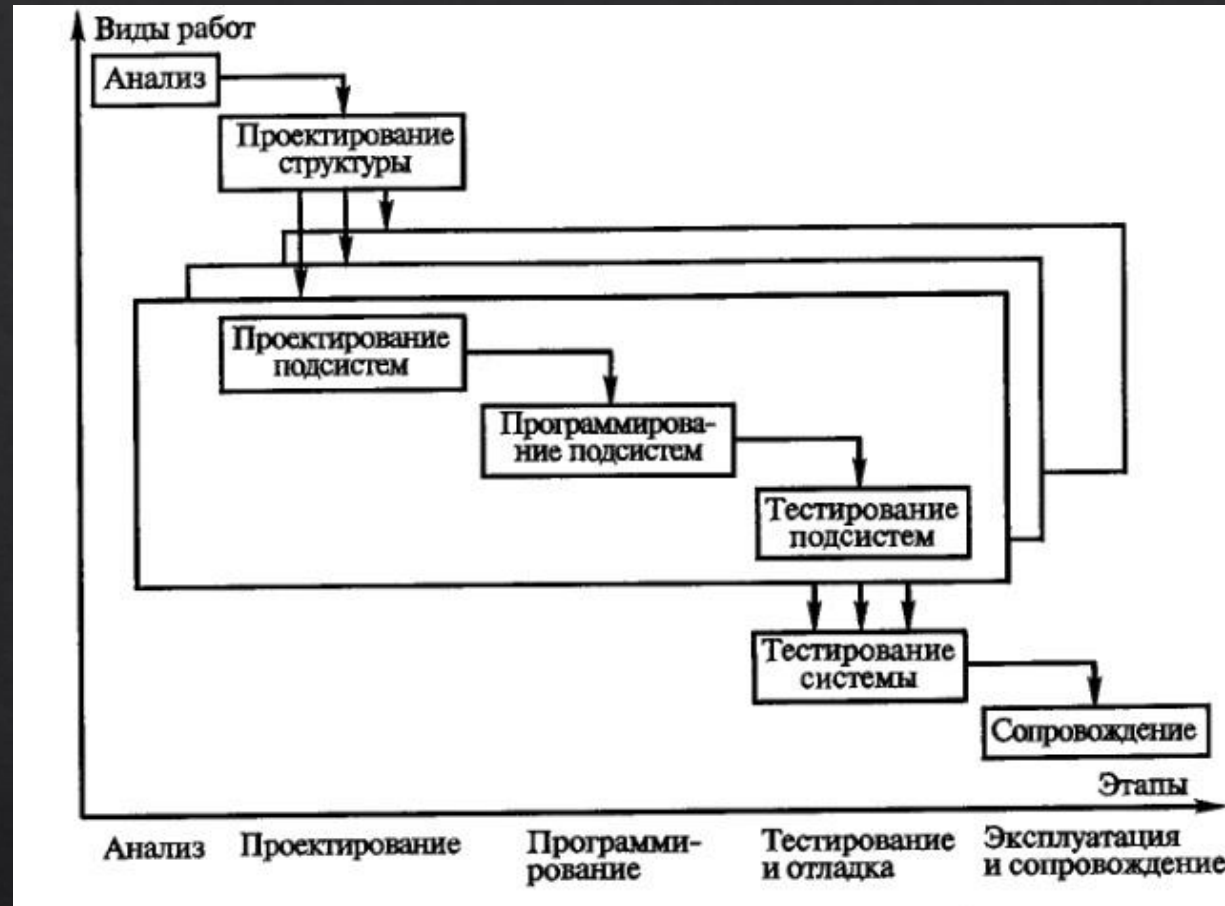


Плюсы: Ускорение разработки за счет распараллеливания работ

Минусы: Усложнение процесса синхронизации работ

Применение: для программа среднего размера

Каскадный подход с подвидами работ

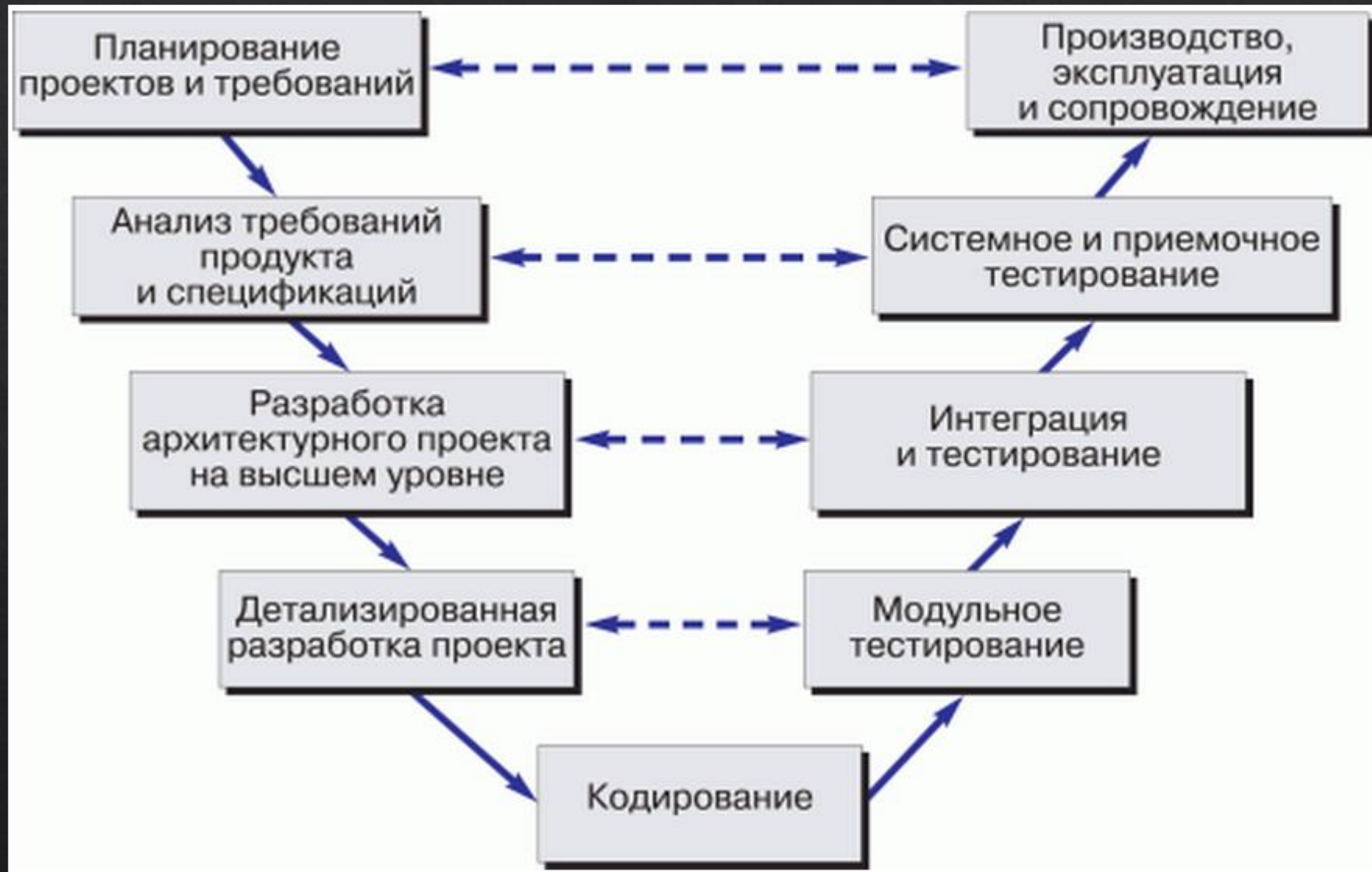


Плюсы: Ускорение разработки за счет распараллеливания работ

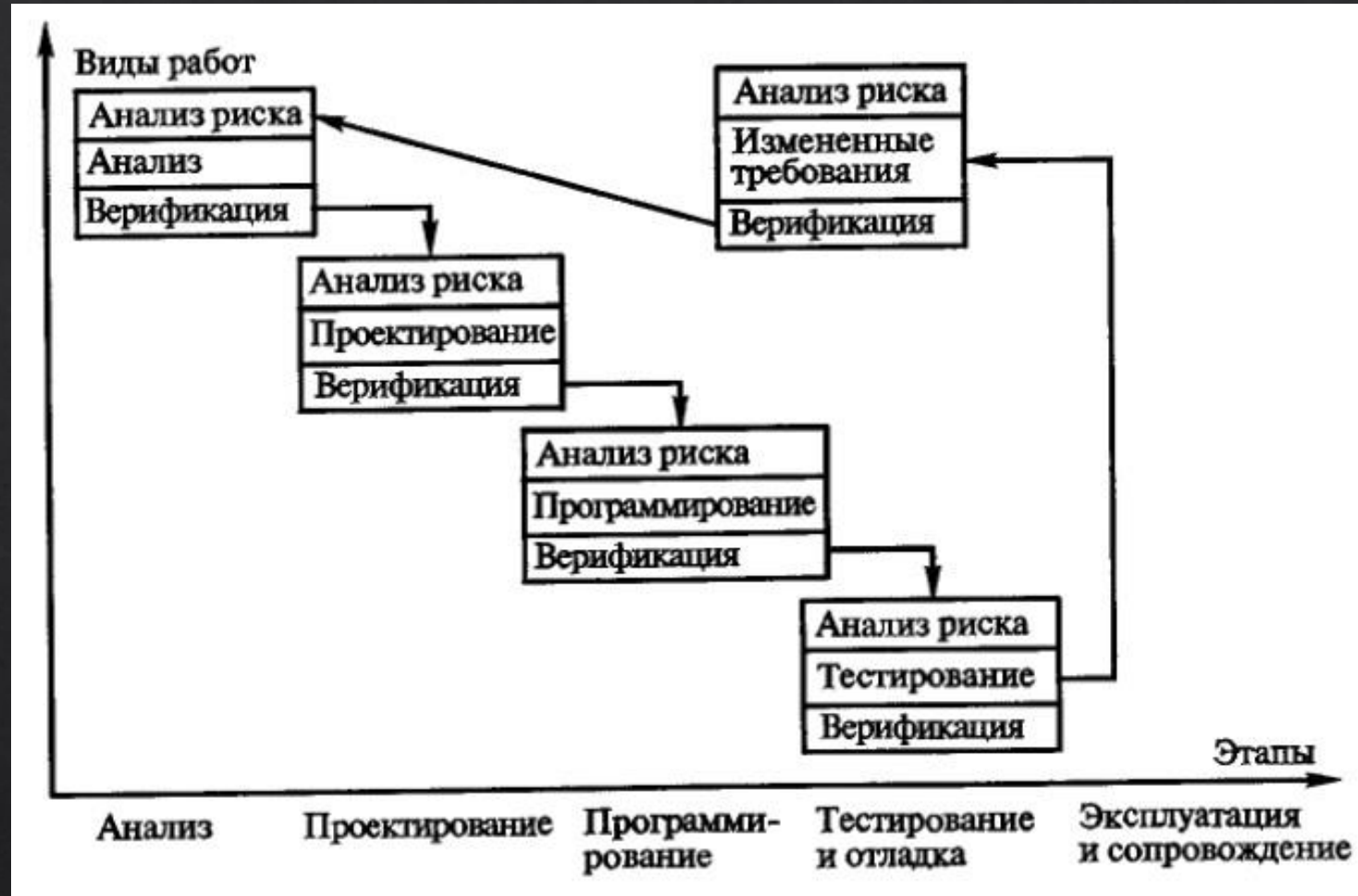
Минусы: Сложный процесс интеграции различных модулей

Применение: для программа среднего размера

Модель «Водоворот»



Спиральная модель (spiral model)



Плюсы: Наискорейшей выпуск программного продукта на рынок

Минусы: Изначально неизвестна структура и состав программы, что приводит к не оптимальности программ, множественным рефакторингам.

Применение: для больших программ

Модель «спираль»

Достоинства:

1. Разбиение проекта на небольшие части.
2. Гибкое проектирование.
3. Разделение проекта на части.
4. Взаимодействие с пользователем.
5. Получение результатов, пригодных для повторного использования.
6. Постепенное уточнение требований к ИС во время ее создания.

Недостатки:

1. Большой объем внутренней документации.
2. Большая стоимость проекта из-за затрат на планирование.
3. Дополнительная документация из-за большого количества этапов работ.
4. Сложной выделения критериев, определяющих длительность работ.
5. Необходимость мощных инструментальных средств.

Когда лучше применять:

1. Большие и сложные проекты.
2. Если необходимо тестирование базовых концепций.
3. Проекты со слишком сложными требованиями.
4. Разработка серии систем.
5. Продукты с ожиданием изменений и дополнений.
6. Долгосрочный проект.
7. Необходимость демонстрации версий ИС.