

Распределенные информационные системы

Практическое занятие: разработка
FOM

Вопросы

- Повтор: FOM, SOM, формат, средства разработки
- Практический пример: РИС ППУ
 - Архитектура
 - Интерфейсы
- Разработка FOM:
 - Построение общих схем: Data Flow Diagram, Activity Diagram, Sequence Diagram
 - Использование нестандартизованных процедур сбора информации
 - Переход к описанию FOM

FOM, SOM

Формат

Средства разработки

ПОВТОР: ОБЪЕКТНАЯ МОДЕЛЬ ИСА

FOM, SOM

- Из правил для федерации – Правило 1:
 - Каждая федерация должна иметь свою объектную модель FOM, задающую описание и взаимосвязи классов всех потенциально возможных в данной федерации объектов. FOM должна быть документирована в соответствии с шаблоном OMT.
- Из правил для федератов – Правило 6:
 - Каждый федерат должен иметь свою объектную модель SOM, документированную в соответствии с OMT HLA.

Практический подход: FOM

- FOM – описывает ВСЕ возможные типы данных, которыми могут обмениваться федераты в процессе выполнения федерации
 - Примечание 1: Другие типы данных не могут использоваться
 - Примечание 2: RTI не контролирует, будут ли использоваться ВСЕ типы данных (одновременно или на протяжении выполнения федерации)
 - Примечание 3: В FOM ВСЕ типы данных PS (некоторые NA) – иначе нет смысла

Практический подход: SOM

- SOM – описывает типы данных, которые «понимает» федерат. Для типов данных указывается P, S или PS – соответственно федерат публикует, подписывается на типы данных или делает и то, и то, одновременно
 - Примечание 1: Часть типов данных, образующих верхние уровни иерархии может быть помечена NA
 - Примечание 2: На каждом уровне иерархии ВСЕ типы данных используются федератом, иначе не включаются в SOM
 - RTI не контролирует SOM и вообще не использует его. SOM предназначен только для разработчиков федератов и федераций

OMT

- Форматом файлов FOM, SOM является Object Model Template, описанный стандартом IEEE-1516.2-2010
- Использование OMT обеспечивает:
 - Автоматическое задание состава данных, циркулирующих в федерации (RTI проверяет соответствие FOM в процессе выполнения федерации)
 - Стандартизированный механизм описания [потенциальных] членов федерации, их возможностей и поведения (SOM не используется RTI непосредственно)
 - $\Sigma \rightarrow$ Облегчает разработку федерации и федератов, позволяет использовать специализированные редакторы

Средства разработки FOM, SOM

- Текстовый редактор
- XML-редактор
- Специализированные OMT редакторы:
 - OMDT
 - SimGe
 - Pitch OMT Editor
 - Редактор объектных моделей (РусБИТех)

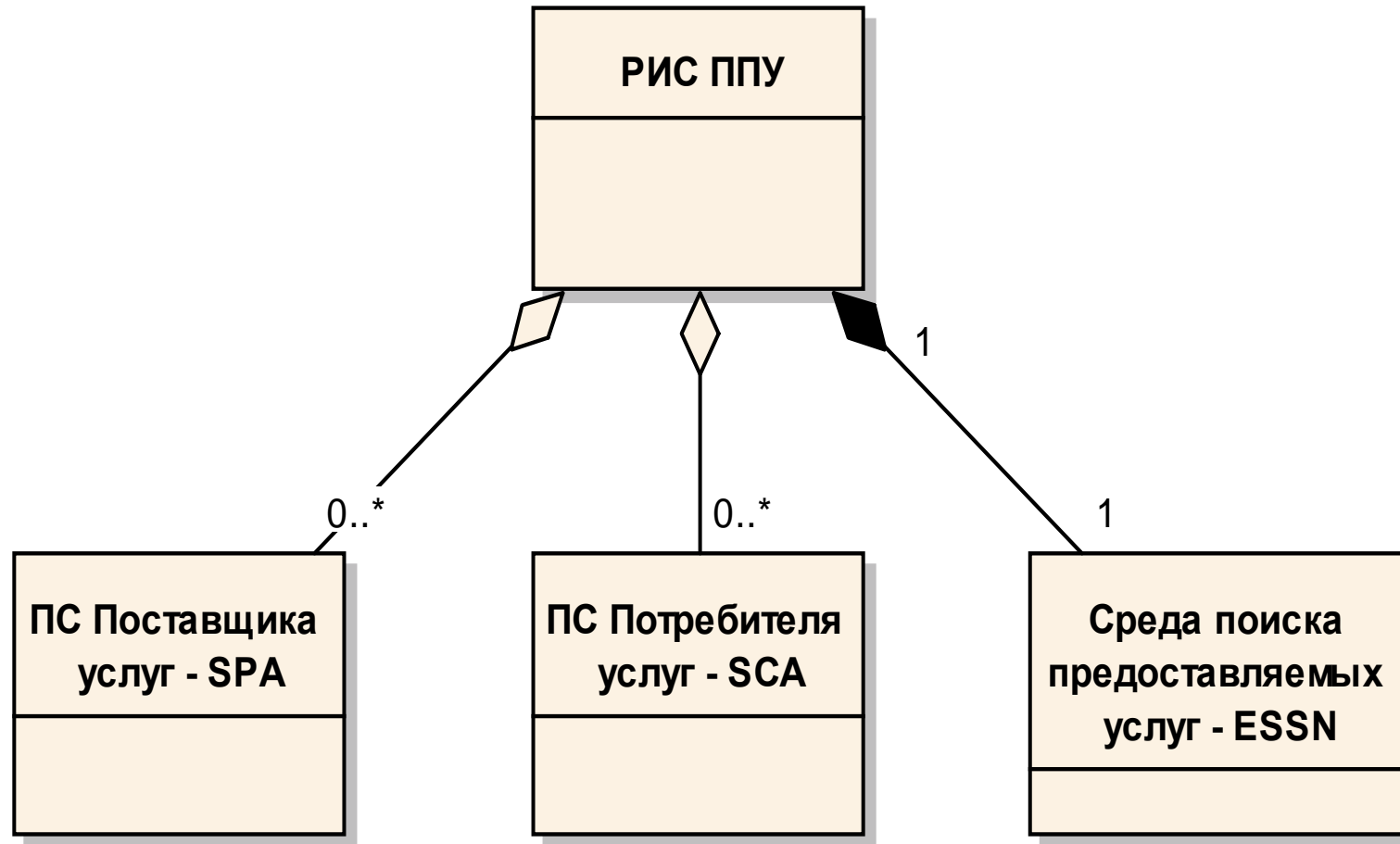
Общая архитектура

Интерфейсы

ПРАКТИЧЕСКИЙ ПРИМЕР: РИС ППУ

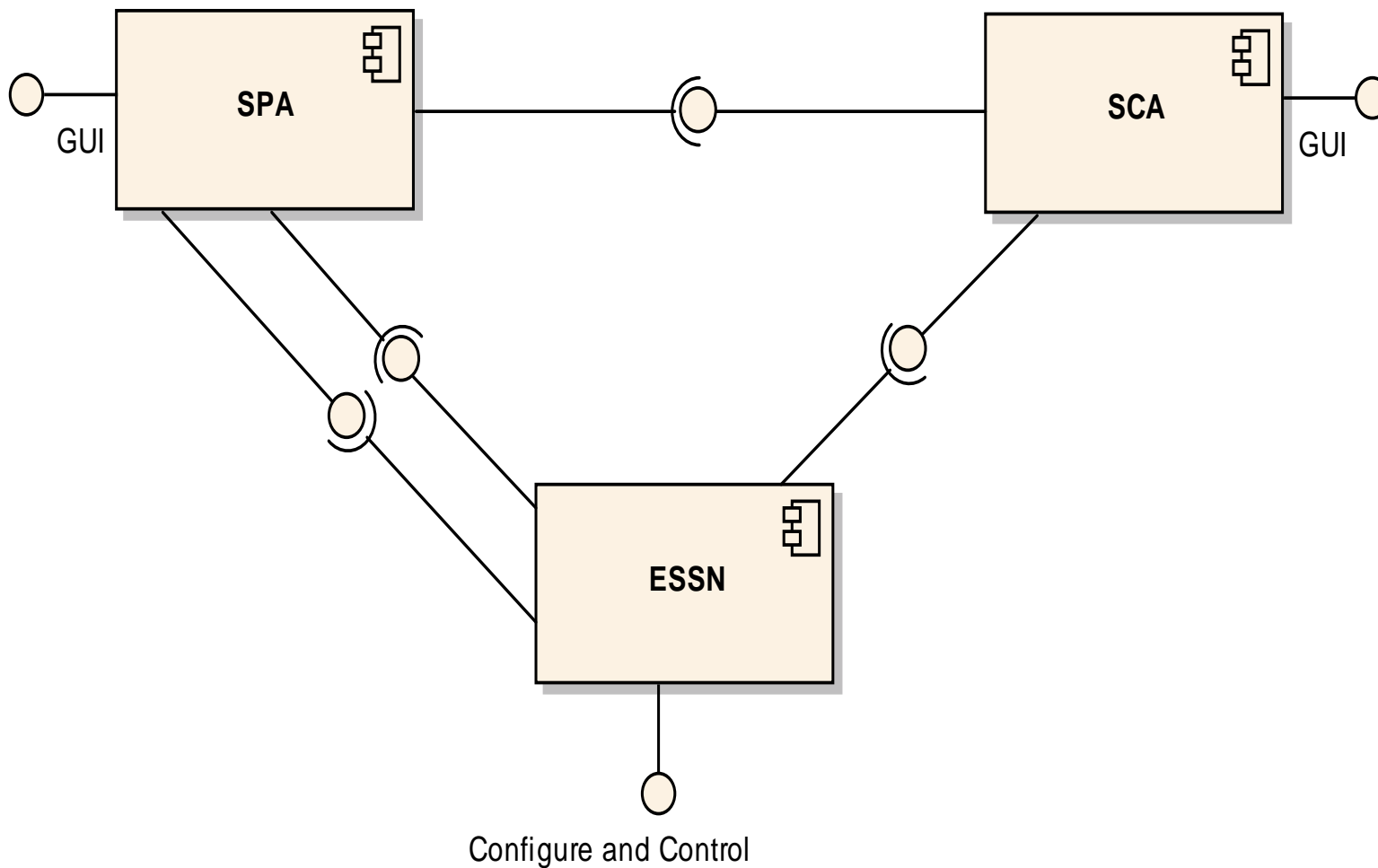
Общая архитектура РИС ППУ

class РИС ППУ

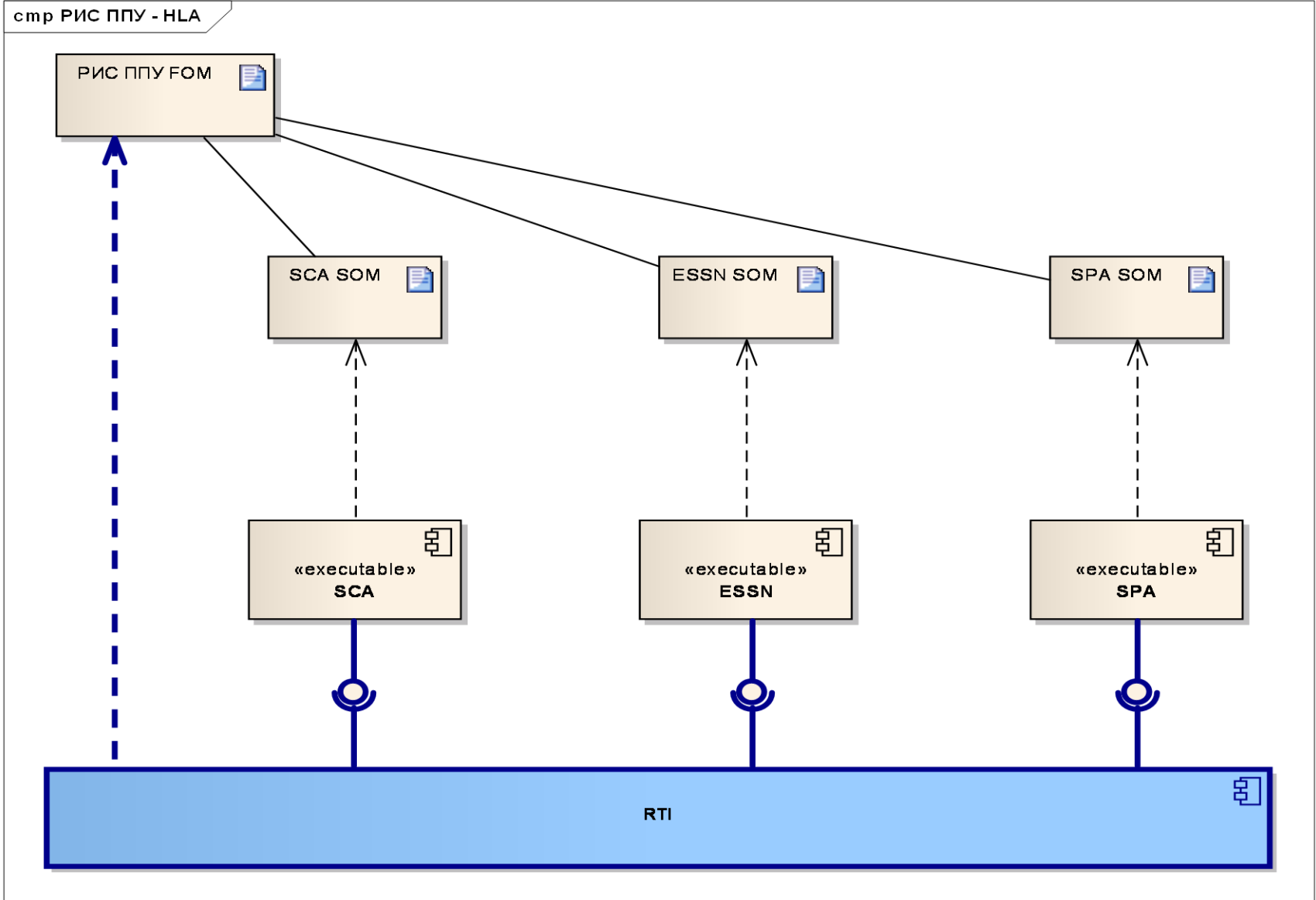


Интерфейсы РИС ППУ

стр Интерфейсы РИС ППУ



ПЗ-2: РИС ППУ на базе HLA RTI



Построение общих схем

Сбор информации

Практический переход к построению

FOM

РАЗРАБОТКА FOM

ПЗ-1 – алгоритм выполнения

✓ Выбрать индивидуальную тему

- (используя гениальность) Придумать архитектуру
 - Очень обобщенно!
 - Sic! Подумать об интерфейсах обмена данными между компонентами
 - NB! Если сделать письменно, то может стать частью пояснительной записки. Но можно и оставить в голове
- Разработать FOM → файл правильного формата
- Описать созданное гениальное творение →
Описание в произвольном формате на 3-* страниц
содержательного текста

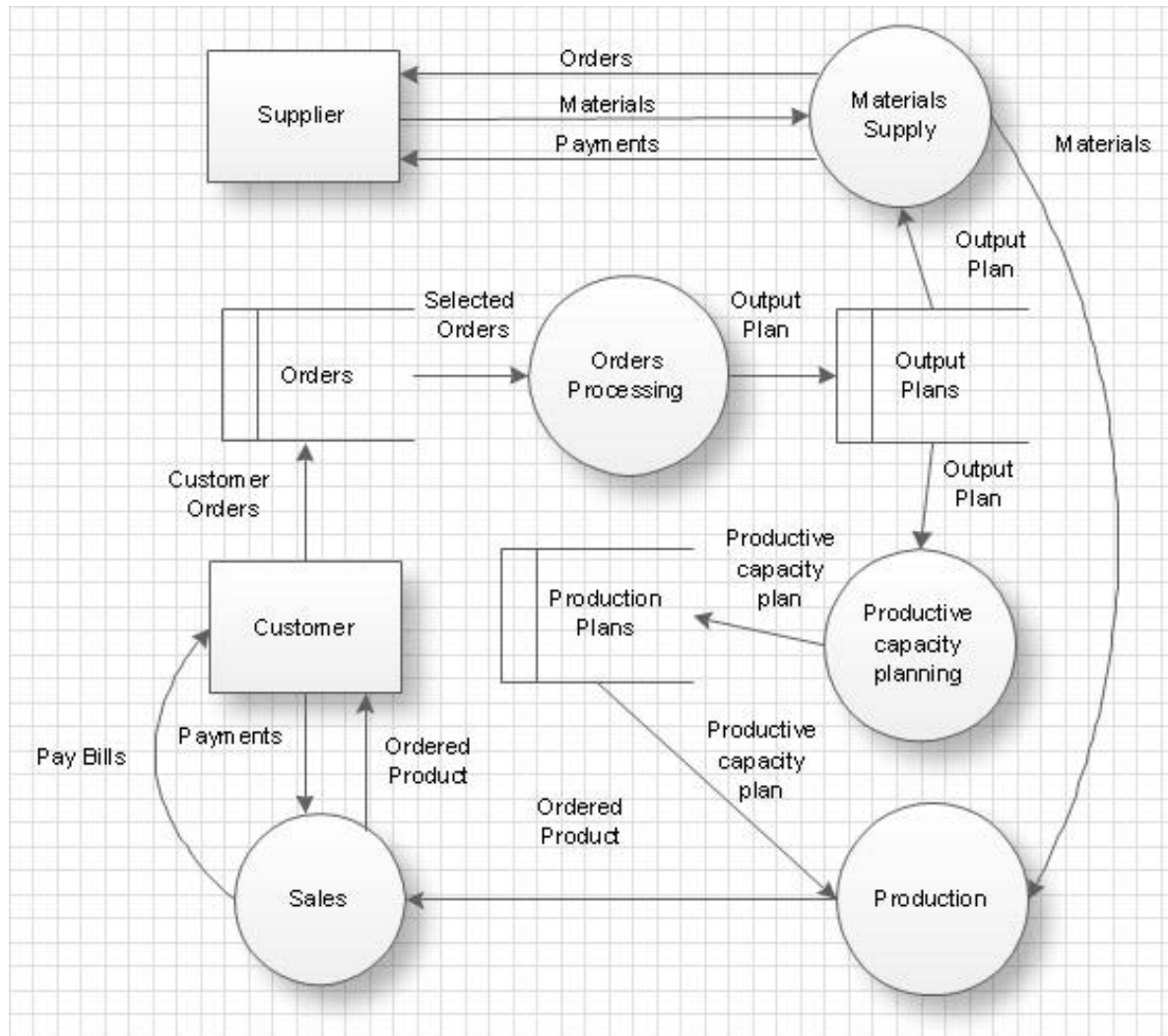
Нотация: Data Flow диаграмма – 1/3

- DFD (Data Flow Diagrams) — диаграммы потоков данных
- Преимущества:
 - Иерархическая
 - Верхний уровень иерархии описывает внешние интерфейсы системы
 - Поддерживает понятие подсистемы
- Недостатки:
 - Сложное поведение (не обработку потока данных) представить затруднительно

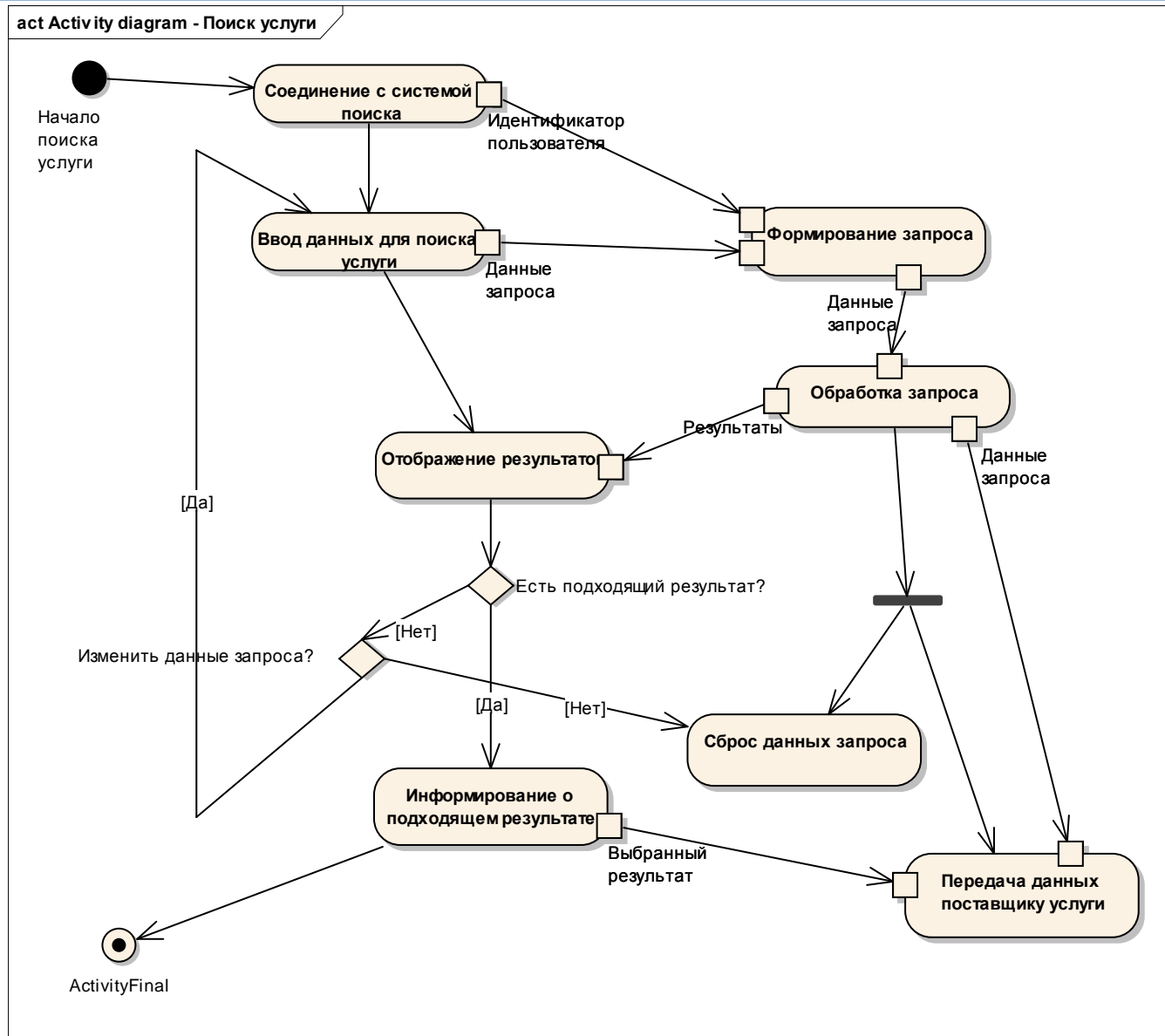
Нотация: Data Flow диаграмма – 2/3

- Элементы:
 - Внешняя сущность (External Entity)
 - Системы и подсистемы (Process)
 - Например: подразделение, программа
 - Поток данных

Нотация: Data Flow диаграмма – 3/3



Нотация: Activity Diagram – 1/2



Сравнение DFD и AD

Data Flow Diagram

- Описывает потоки данных
- Легко привязывается к объектам (экземплярам классов), которые передаются
- Подразумевают одновременность обработки данных

Activity Diagram

- Описывают одновременно потоки управления и потоки данных
- Позволяют описать одновременность, но описание тяготеет к описанию последовательной обработки

Нотация: Activity Diagram – 2/2



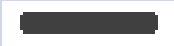


Блок	Описание
	Действие (action)
	Принятие решения (decision) Условие выбора дальнейшего пути (может делить на 2 и более путей)
	Начало / конец параллельного выполнения
	Начало обработки
	Конец обработки

Диаграмма последовательности – 1/2

- Показывает взаимодействие объектов (обмен между ними сигналами и сообщениями), упорядоченное по времени, с отражением продолжительности обработки и последовательности их проявления
- Обычно уточняет описание

Диаграмма последовательности – 2/2

