

Системное моделирование для разработки требований план

1. Методы моделирования для разработки требований
2. Диаграммы «сущность-связь»
3. Диаграммы состояний

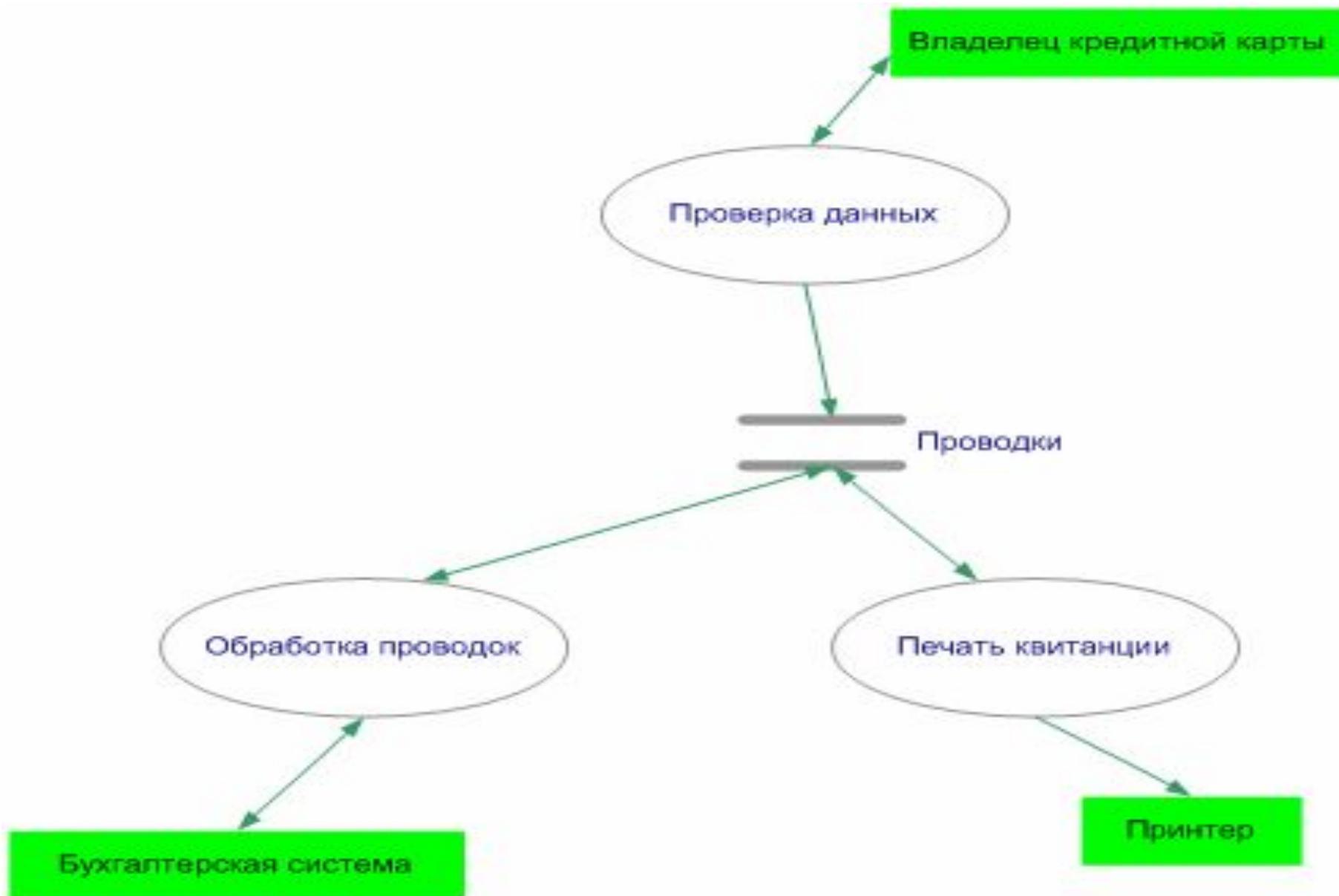
Диаграммы потоков данных (Data flow diagrams или DFDs) являются базовым понятием для многих традиционных методов моделирования. Диаграммы потоков данных имеют достаточно ограниченный набор графических элементов для представления структуры системы и ее интерфейсов. И, несмотря на то, что изначально диаграммы потоков данных предназначались для описания информации и информационных потоков, диаграммы этого типа могут использоваться для описания потоков любого типа, независимо от того базируется система на использовании компьютерной техники или нет. К сожалению, диаграмма потоков данных не отражает один важный поток информации– поток управления.

простой пример традиционного использования диаграммы потоков данных для моделирования информационной системы.

На диаграммах потоков данных используются следующие элементы:

- потоки данных (обозначаются стрелками с названиями);
- элементы преобразования данных (обозначаются окружностями или овалами);
- хранилища данных (отрезки горизонтальных параллельных линий);
- внешние сущности (прямоугольники).

Диаграмма потоков данных

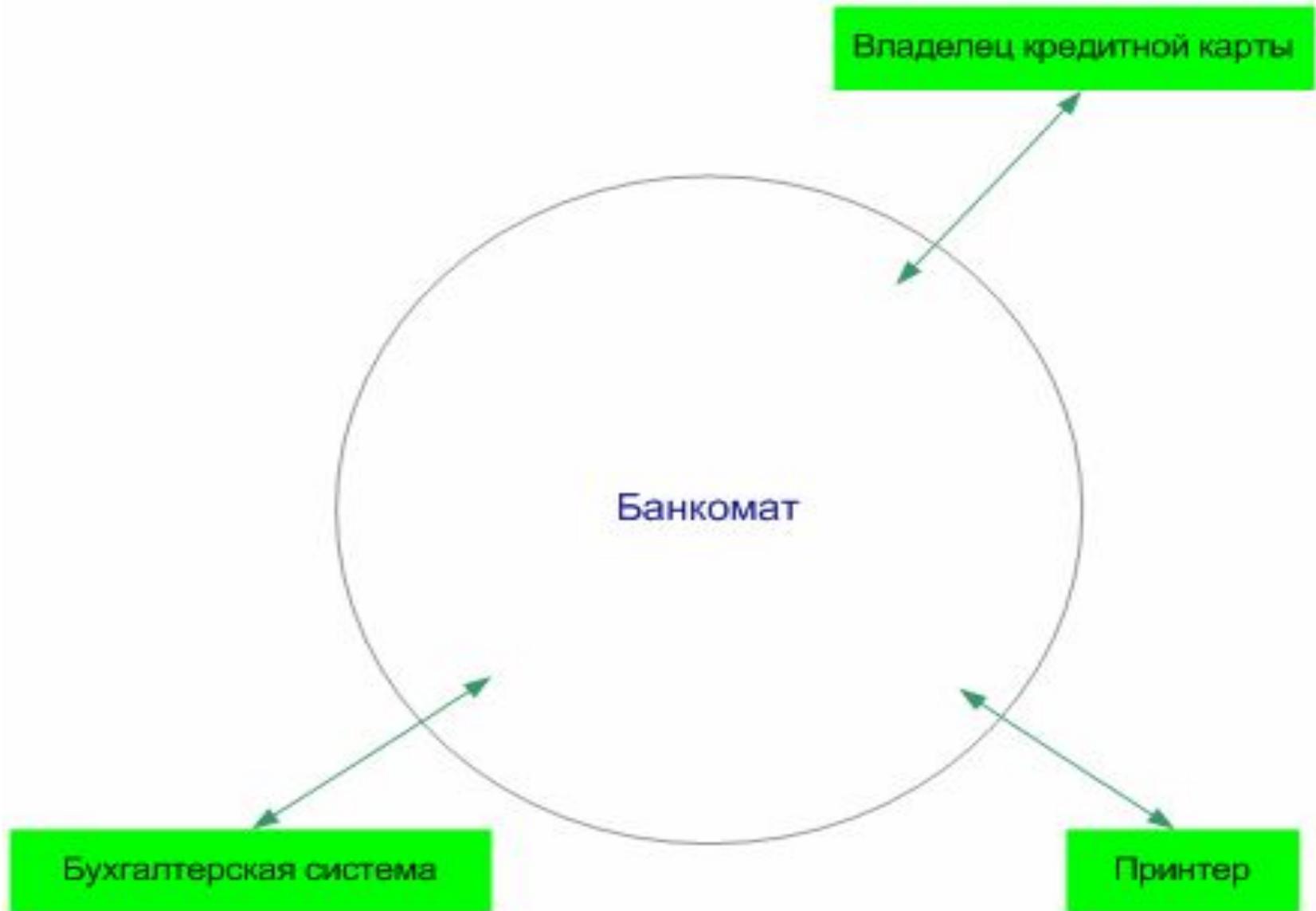


Потоки отображают либо информационный, либо материальный обмен между двумя элементами преобразования. В реальной жизни эти потоки могут быть непрерывными, запускаться по запросу, быть асинхронными и т.д. В соответствии с нотацией диаграмма должна сопровождаться текстовым описанием каждого процесса, хранилища данных и потока данных.

Для определения всех потоков и хранилищ данных используется словарь данных.

Каждый овал (окружность) определяет базовую функциональность, которую обеспечивает компонент системы. Эта функциональность описывается с помощью P-спецификаций (P-spec) или мини-спецификаций (mini-spec), которые представляют собой текстовое описание, зачастую написанное с помощью псевдокода.

Контекстная диаграмма - это диаграмма самого верхнего уровня модели, которая показывает все внешние системы, взаимодействующие с разрабатываемой системой



Овалы могут детализироваться при «движении» вниз от уровня к уровню (декомпозиция). При этом функциональное содержимое каждого овала может раскрываться с помощью отдельной диаграммы, которая, в свою очередь, также может содержать другие овалы и хранилища данных

Функциональная декомпозиция.

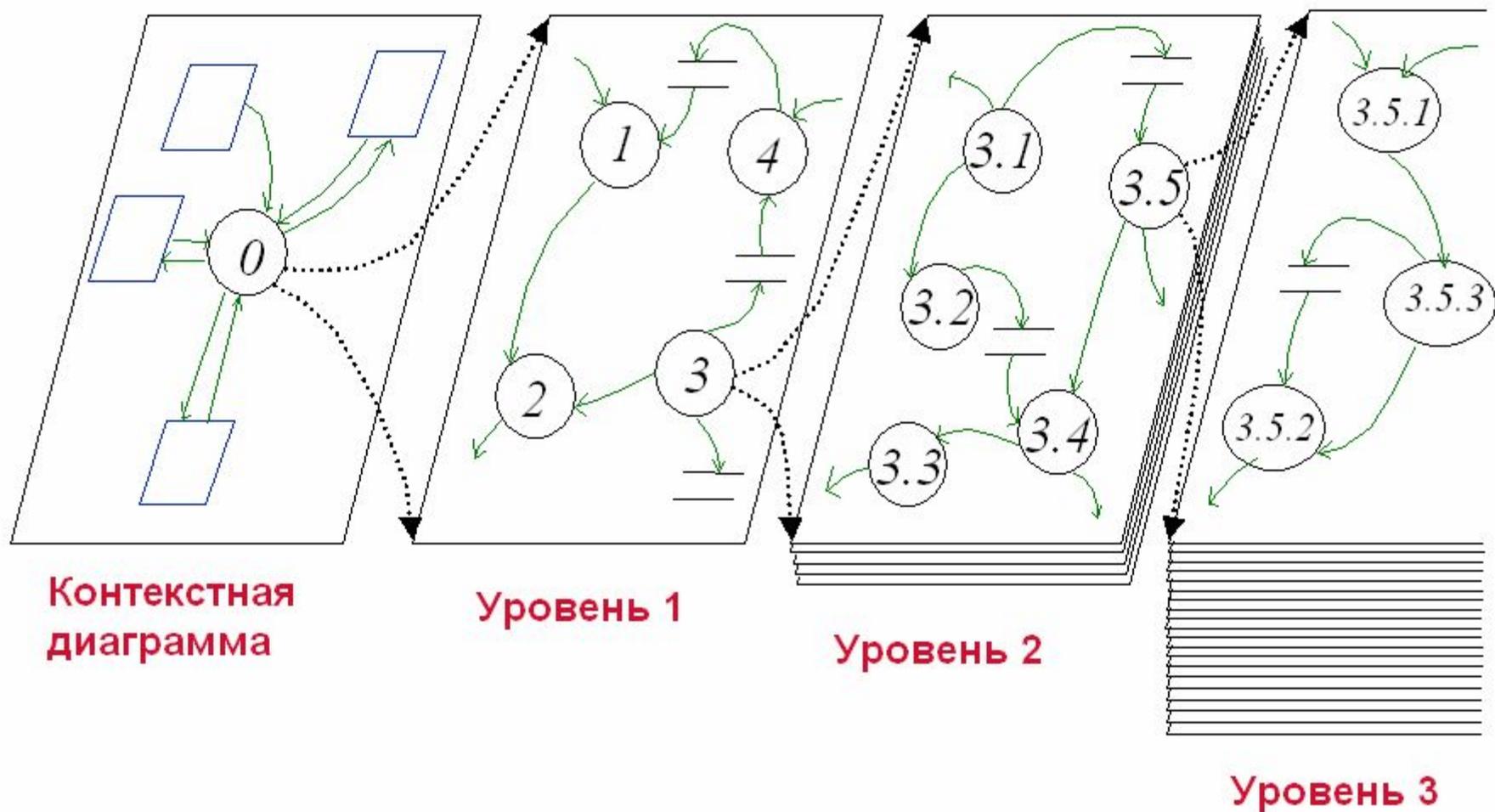
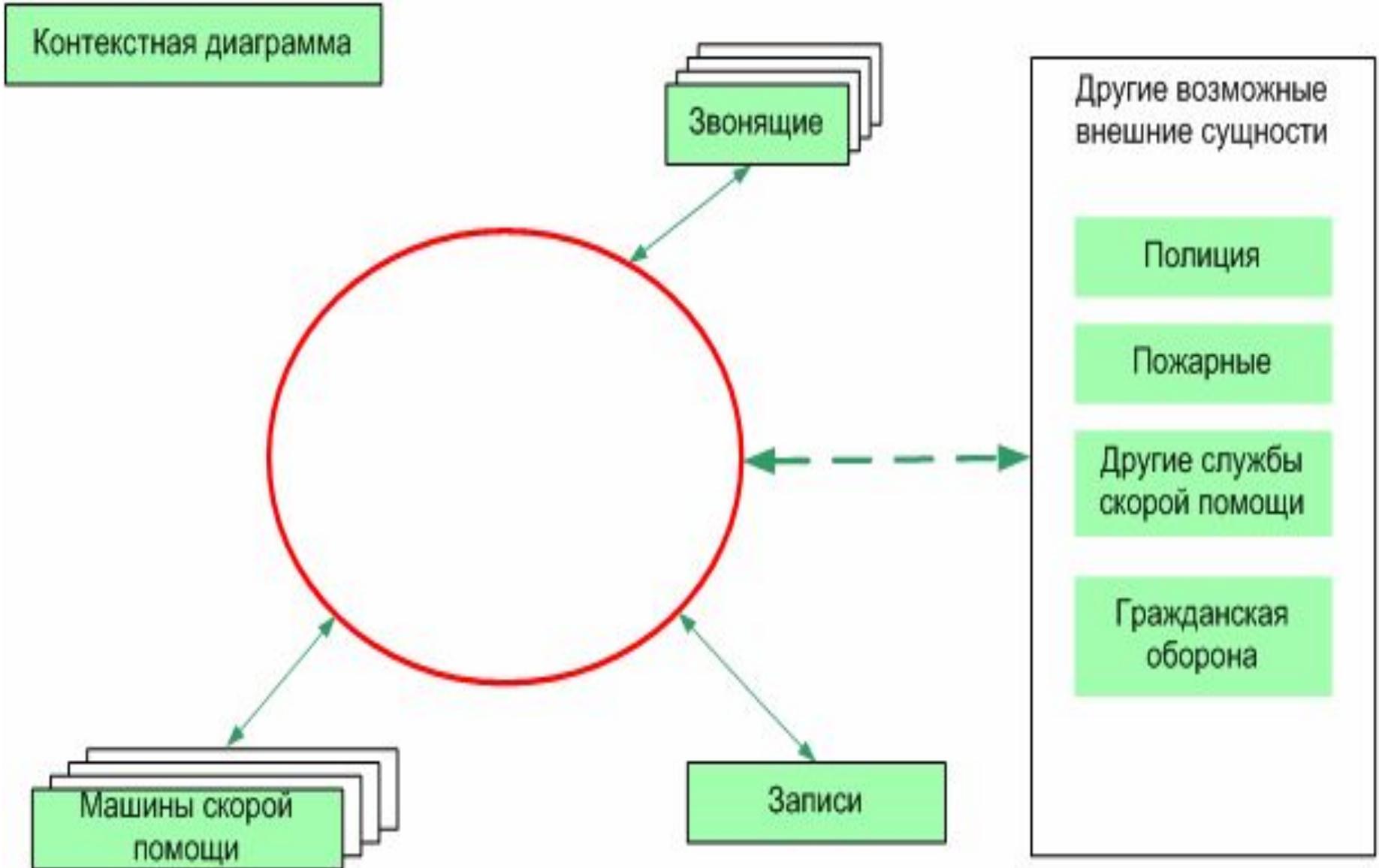


Иллюстрация применения диаграмм потоков данных пример контекстной диаграммы для системы управления и контроля скорой помощи



люди, которые звонят, чтобы сообщить об экстренных ситуациях, и машины скорой помощи, работой которых управляет система.

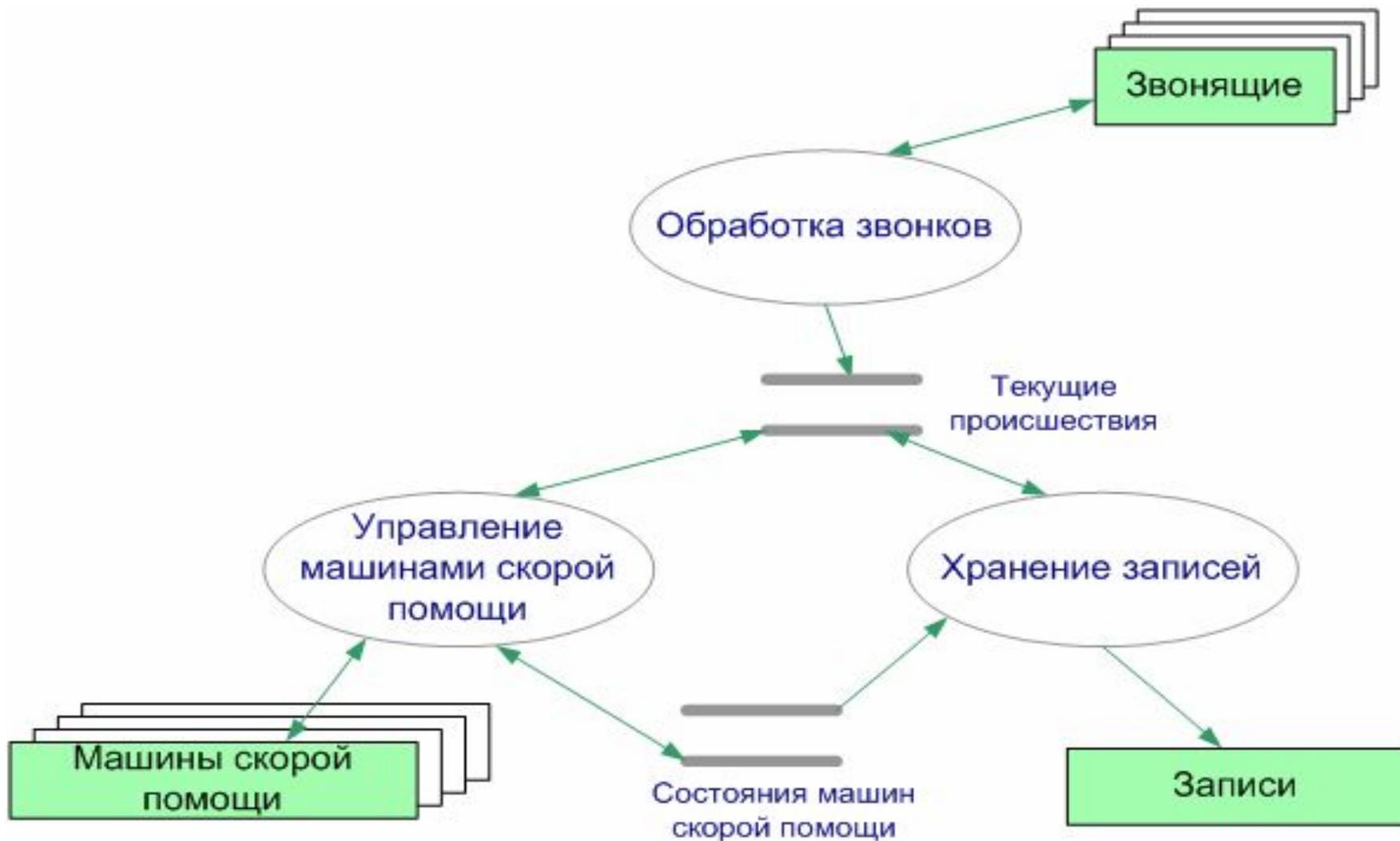
Отметим, что записи являются важным выходом системы (в реальной жизни это отражает требование законодательства) и не менее важным средством для измерения «производительности».

Другие внешние потенциальные сущности системы, представленные на диаграмме, тоже обязательны для реальной системы, но для простоты мы исключим их из рассмотрения в данном примере.

Следующим шагом является определение внутренних функций системы.

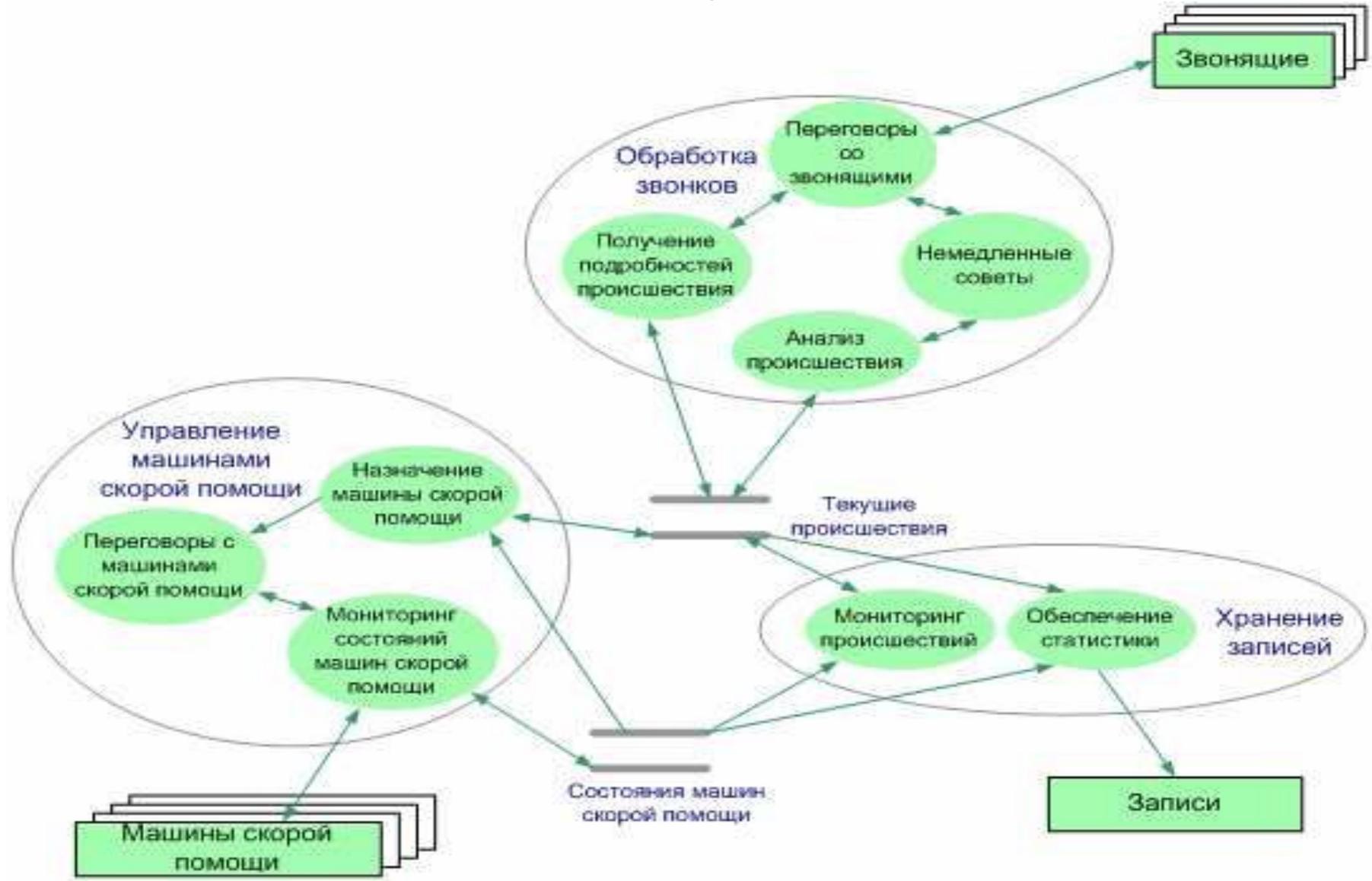
Обычно вначале отображают функции для работы с каждой из внешних сущностей, получая, таким образом, минимальную функциональную декомпозицию системы. После этого отображают основные данные, которые должны транслироваться между этими функциями верхнего уровня.

Модель системы управления и контроля скорой помощи.



На следующем этапе функции верхнего уровня разбиваются на более мелкие составляющие функции

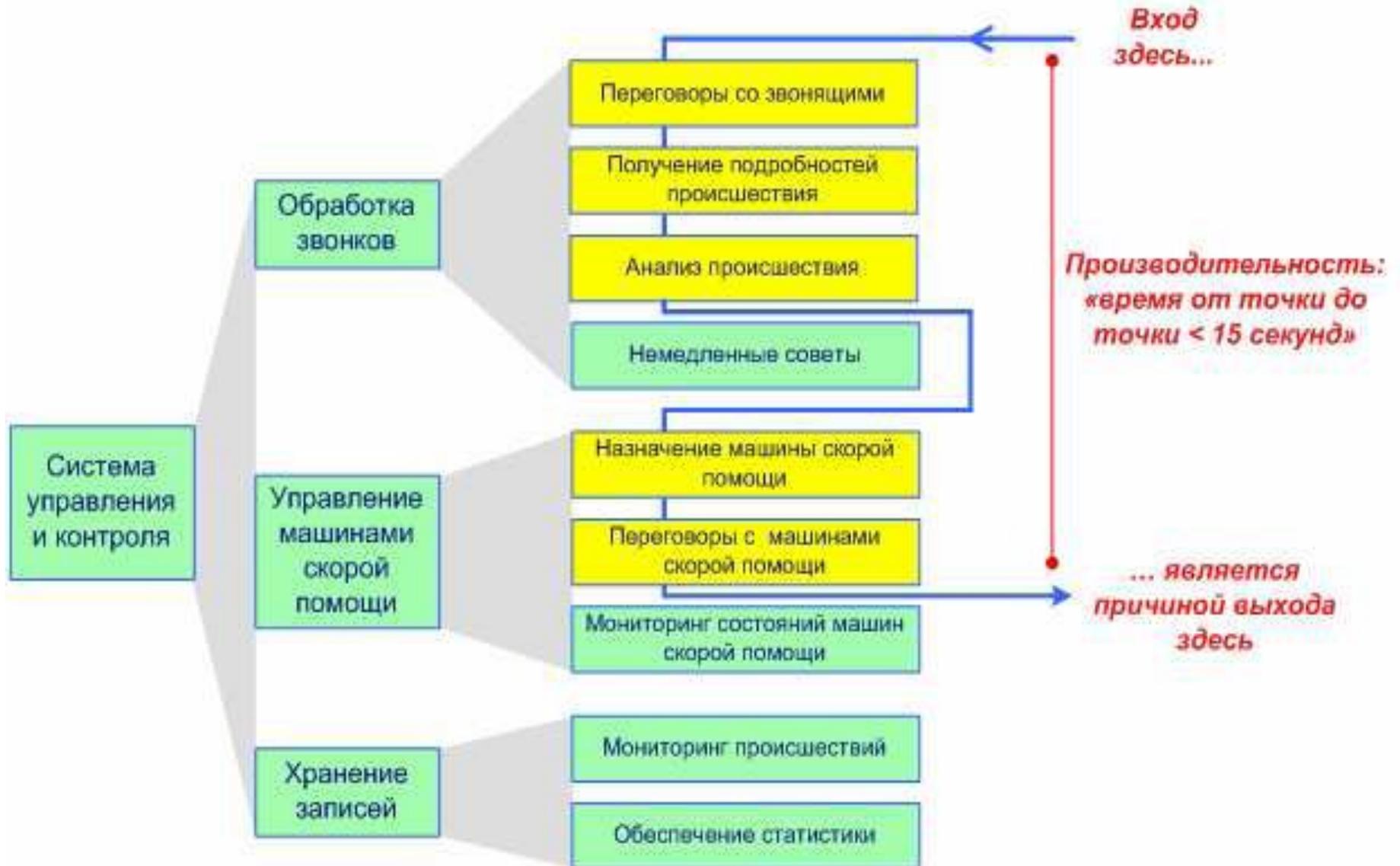
Детализированная модель системы управления и контроля скорой помощи.



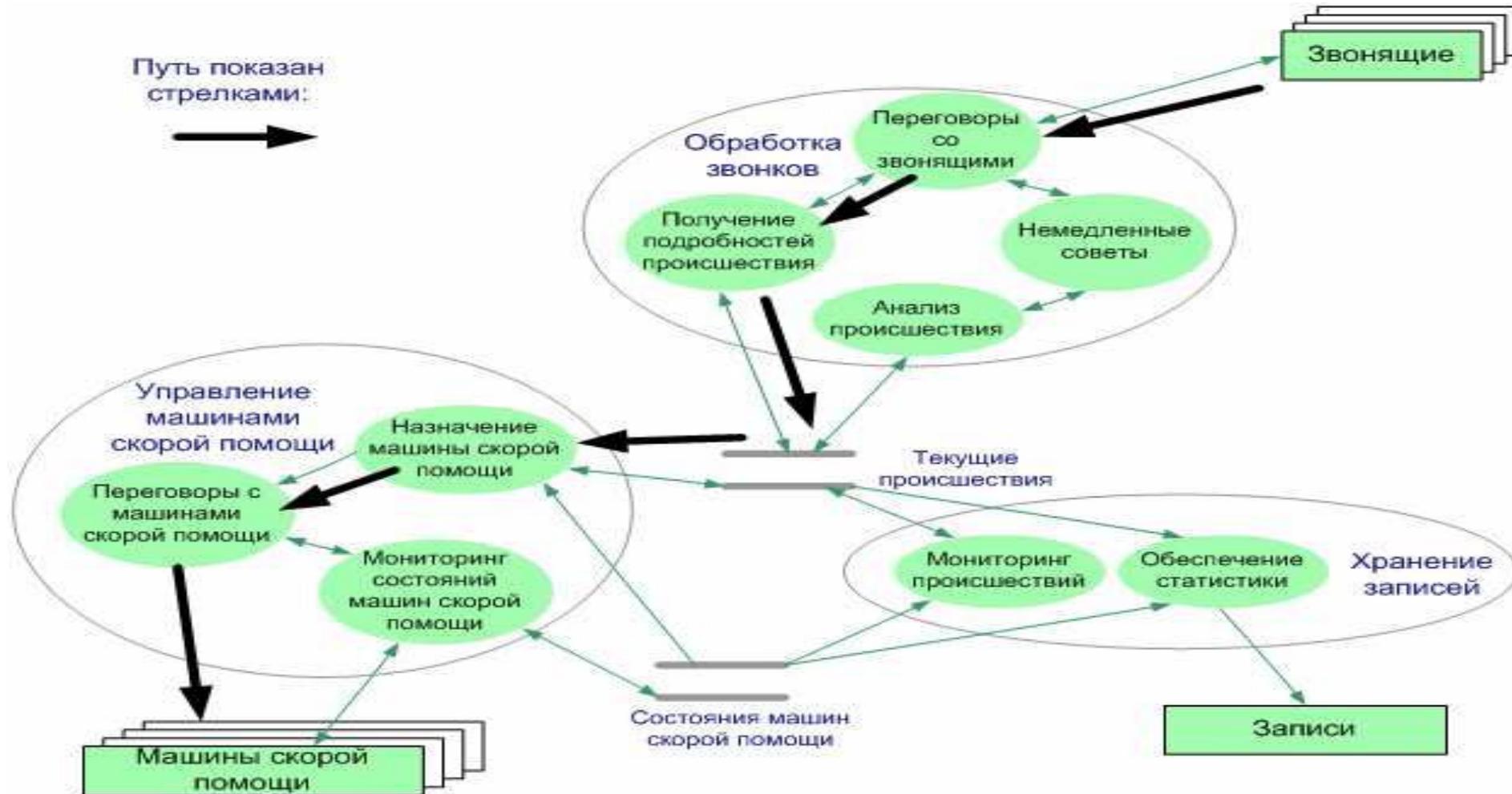
Функциональная структура системы управления и контроля скорой помощи.



Представленная иерархия и интерфейсы могут хорошо описывать модель компонентов модели, но дают плохое представление о тех действиях внутри системы, которые заключены между точкой входа и точкой выхода (системные транзакции) (transactions)
Системные операции.



Отсюда возникает необходимость рассматривать системные транзакции с разных точек зрения– пути, который они проходят внутри системы; времени, которое требуется для их выполнения; ресурсов, которые они задействуют. Альтернативой динамическим моделям, показывающим пользовательские требования в действии и иллюстрирующим основные операции системы, является отображение системных транзакций на диаграммах потоков данных, где путь системной транзакции показан жирными стрелками.



Диаграммы потоков данных хорошо подходят для представления структур, но они недостаточно «аккуратны». Они гораздо менее точны, чем текстовое представление разрабатываемой системы. На диаграмме потоков данных линии связи могут быть неоднозначны, любое слово диаграммы может обобщать что угодно и иметь массу значений. Кроме этого, на диаграммах потоков данных нельзя корректно отображать ограничения.

Диаграммы потоков данных достаточно хорошо отображают функции и интерфейсы.

Они также могут использоваться для определения системных транзакций типа «начало-конец», хотя и отражают их только косвенно. В идеале, конечно, было бы неплохо иметь возможность так просматривать диаграммы и «разворачивать» элементы структуры прямо на диаграмме, чтобы видеть контекст каждой диаграммы, каждого уровня декомпозиции. Кстати, некоторые CASE средства позволяют это делать.

Таким образом, диаграммы потоков данных:

- отображают в целом потоки данных и функциональность системы;
- определяют функции, потоки и хранилища данных;
- определяют интерфейсы между функциями;
- обеспечивают базу для получения системных требований;
- имеют инструментальные средства для работы с ними (специальное ПО);

Диаграммы сущность-связь

Часто бывают ситуации, когда важной необходимостью является моделирование информации, которая хранится в системе, например, планы полетов, информация о наземных объектах и ориентирах, записи базы данных. В этом случае диаграммы сущность-связь (Entity-relationship diagrams или ERDs) как раз и предоставляют средства для моделирования сущностей системы и связей, существующих между ними.

В таких диаграммах часто используется принятая терминология.

Сущность это объект, который может быть четко классифицирован и определен, например, заказчик, поставщик, деталь или продукт.

Свойство (или атрибут) это информация, которая описывает сущность.

Связь характеризуется множественностью, которая описывает тип связи между сущностями (один к одному, один ко многим, многие ко многим).

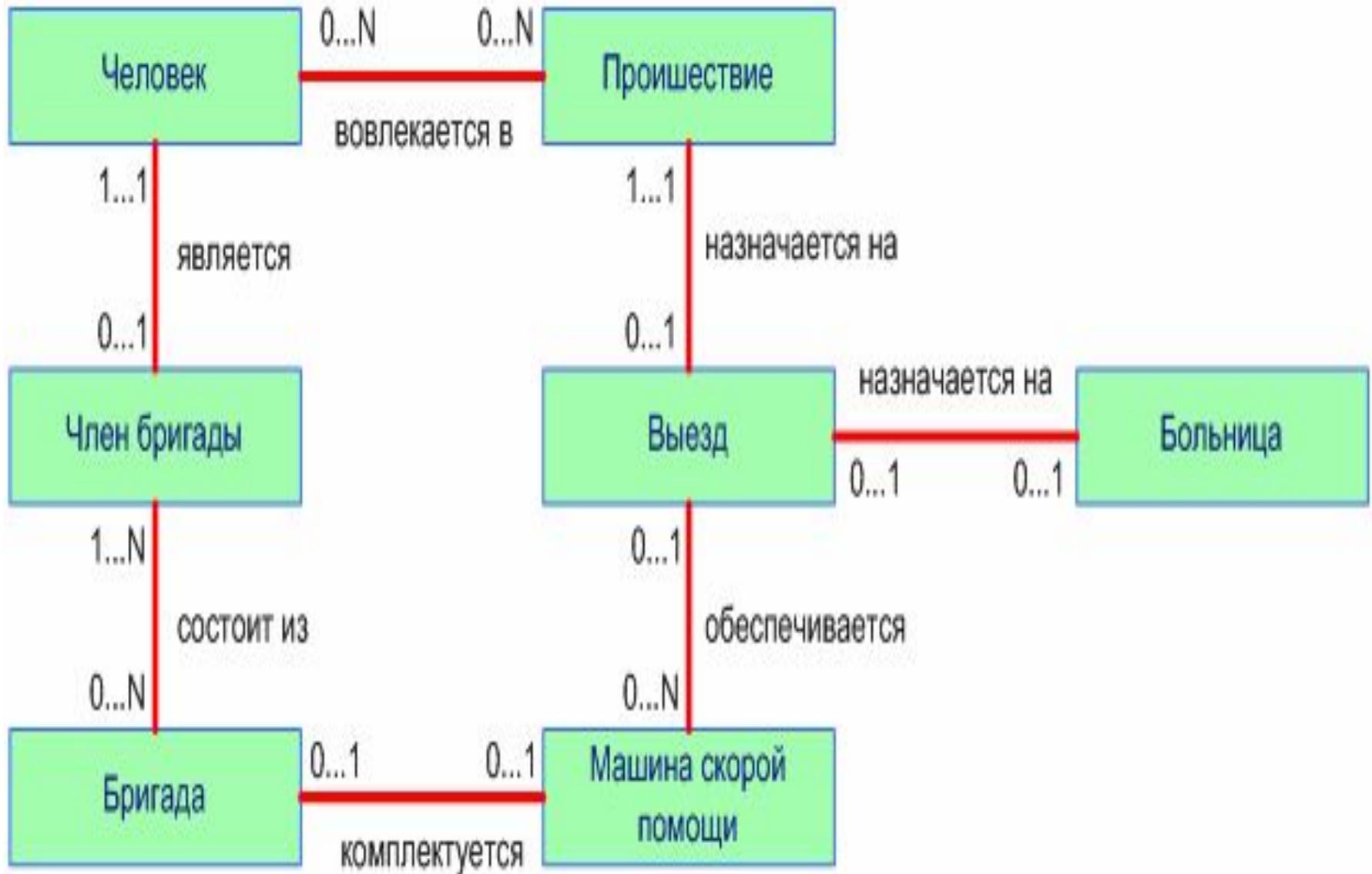
Подтипом называется подмножество объектов сущности (например, тип X является

Подтипом Y, если любой член множества X принадлежит к Y).

Диаграммы сущность-связь формируют модель, которая лишь частично описывает

систему, поскольку определяет только сущности внутри системы и связи между ними. Эта модель является независимой по отношению к протекающим в системе процессам, которые требуют создания или использования информации. Однако такая модель является идеальным средством для абстрактного моделирования на этапе разработки системных требований.

Диаграмма сущность-связь для системы управления и контроля скорой помощи.



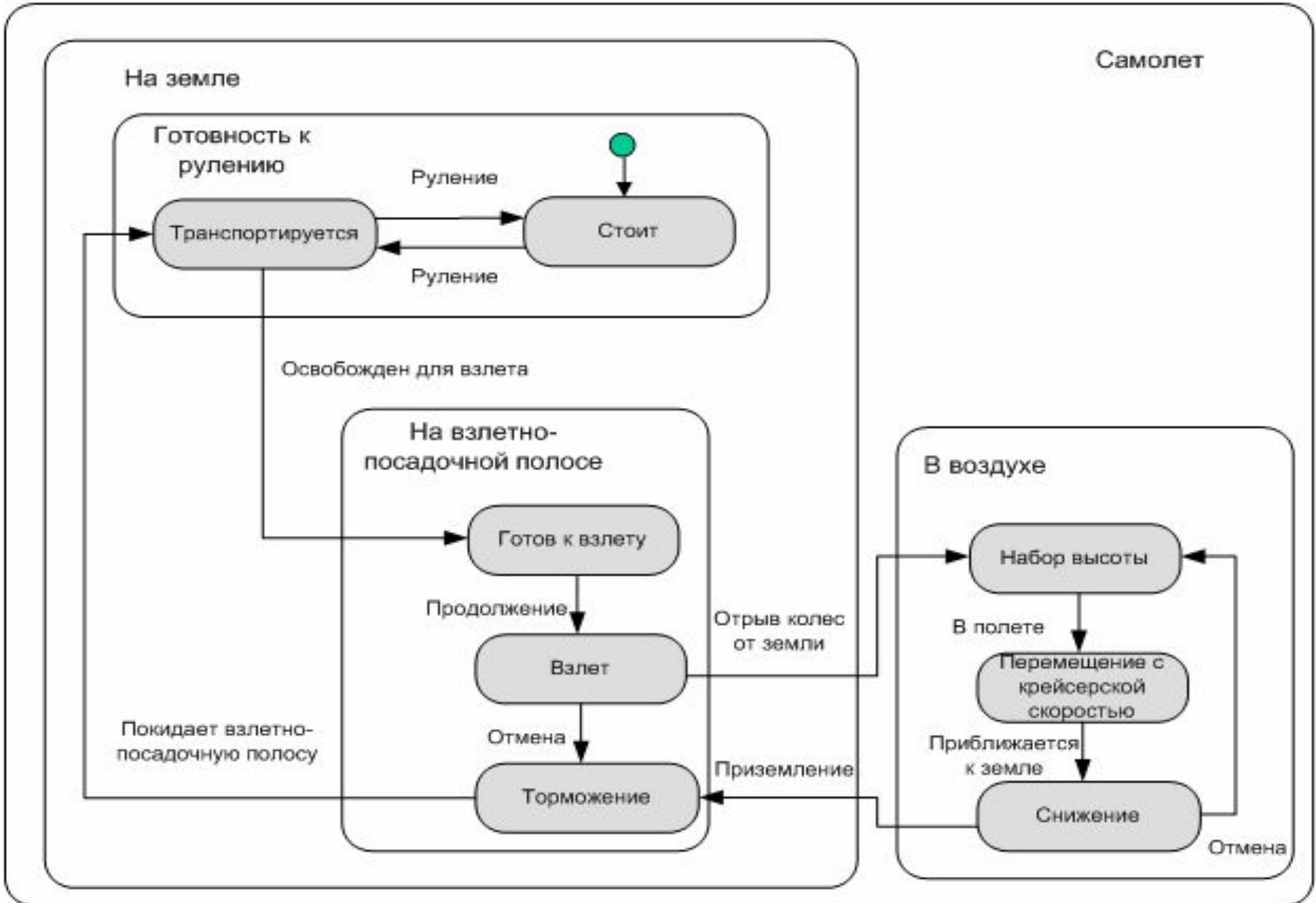
Диаграммы состояний

Для определения требований недостаточно иметь только функциональные модели и модели потоков данных. Необходимо также рассмотреть поведение системы в различных ситуациях, имея в виду, что при этом система может находиться в конечном числе возможных состояний.

Внешние события для системы могут служить триггерами, которые, срабатывая, переводят систему из одного состояния в другое. Для того чтобы рассмотреть систему в данном аспекте необходимо наглядно представить возможные состояния системы и ее реакцию на внешние события в разных состояниях. Обычно для этого используются диаграммы состояний (statecharts) Харела (Harel).

Диаграммы состояний обеспечивают описание поведения системы. В рамках одной диаграммы состояний может быть показана вся иерархия состояний системы и последовательность переходов системы из одного состояния в другое. Тем не менее, диаграммы состояний могут также использоваться и для описания систем, внутри которых возможно протекание параллельных процессов. На диаграмме состояний прямоугольником с закругленными углами обозначается состояние. Иерархия состояний отображается путем включения одного состояния в другое и направленными стрелками, причем подписи к стрелкам являются описаниями событий, символизирующими переход между состояниями.

Диаграмма состояний для полета самолета.



Два состояния верхнего уровня это– «на земле» и «в воздухе», между которыми определены переходы. Внутри состояния «в воздухе» существует три других независимых состояния, а внутри состояния «на земле» существуют состояния «готовность к рулению» и «на взлетно-посадочной полосе». При этом внутри состояний «готовность к рулению» и «на взлетно-посадочной полосе» также имеются собственные вложенные состояния.

Система переходит в состояние «в воздухе» когда колеса самолета отрываются от земли, а когда колеса касаются земли, система возвращается в состояние «на земле». Далее эти состояния детализируются в иерархическом порядке.

Диаграммы состояний имеют еще одну важную составляющую нотации– историю (history).

Здесь имеется в виду следующее.

Если имеется некое состояние, которое на диаграмме помечается специальным значком (H), то при возвращении системы в это состояние все вложенные состояния внутри этого H-состояния также считаются перешедшими в это первичное состояние.

Спасибо за внимание!

