

Инженеринг на изискванията

Извличане на изискванията

Лекция 4

Съдържание

- **Извличане на изискванията- същност, необходима информация**
- **Цели и процес на извличане на изискванията**
- **Начална информация за идентифициране на изискванията**
- **Техники за извличане на изискванията**
- **Проблеми при идентифициране на изискванията**

Кога възниква необходимост от ИИ

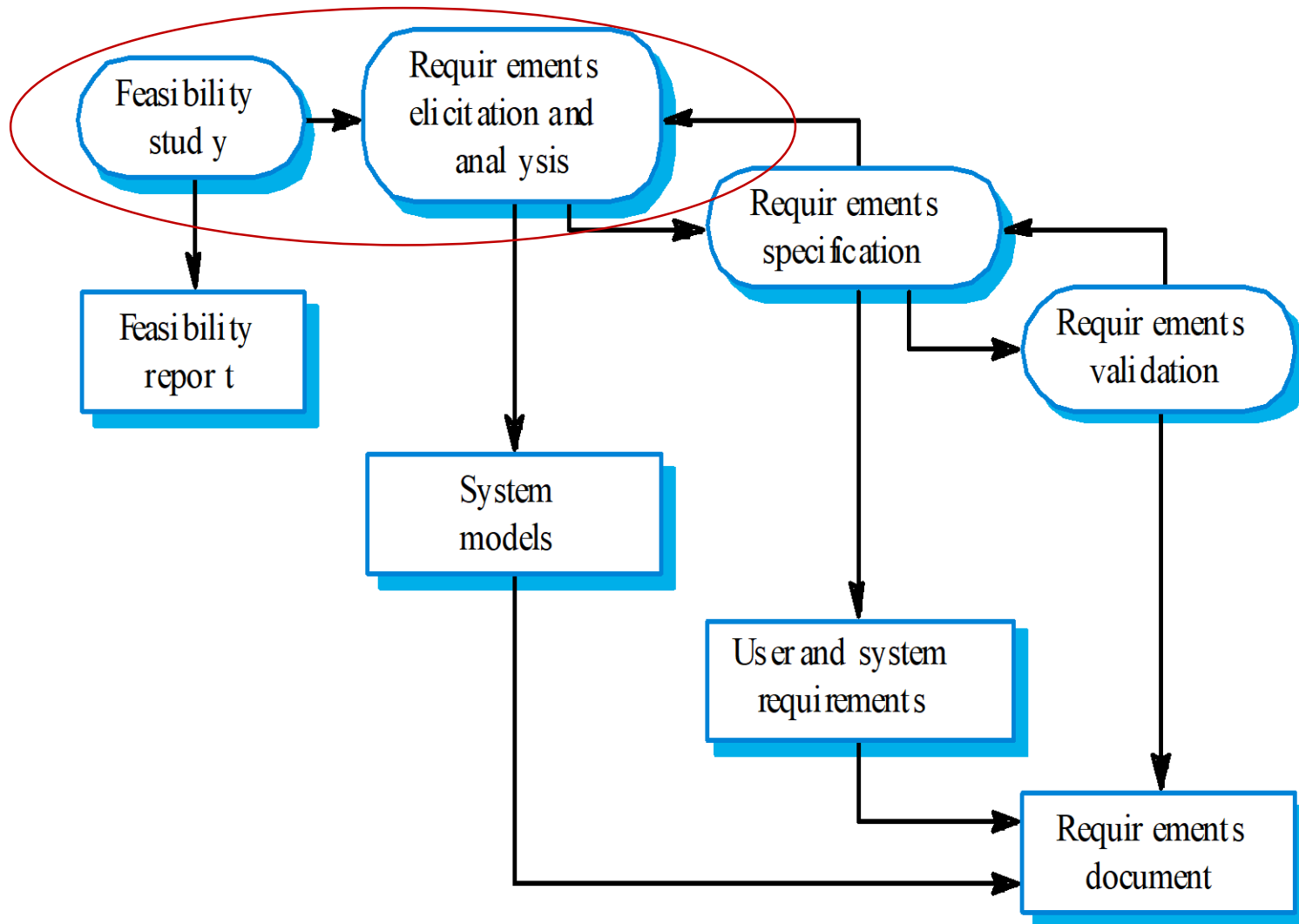
Начало е:

**Съществува “проблем”, който трябва да се реши,
защото съществува:**

- неудовлетвореност от текущото състояние
- нова възможност за бизнеса
- възможност за спестяване на разходи, време, ресурси и

др.

Процес на инженеринг на изискванията



Предпроектни проучвания /или проучвания за осъществимост/ (Feasibility studies)

Входна информация: Общо описание на системата; как ще бъде използвана

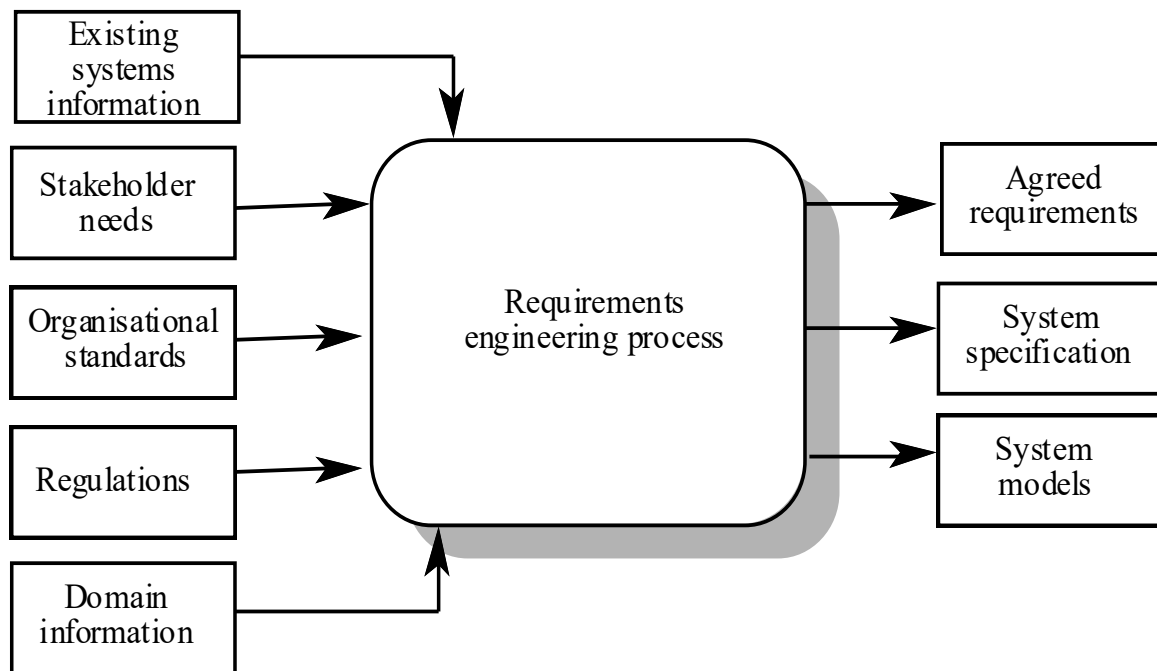
Исходна информация: Доклад-становище дали системата да бъде разработвана:

- *принос* за организацията, за която е предназначена
- *реалистичност* - може ли да бъде разработена с използване на наличните ресурси и ограничения
- възможност за *интегриране* с други системи.

Извличане на изискванията

Използва се информация от:

- съществуващи системи;
- заинтересованите лица;
- документи (за организационни стандарти, регулации, др.);
- информация за областта, наблюдения



Извличане на изискванията – процес и общи принципи (Requirements elicitation)

Дефиниция: Извличане на изисквания е процесът за търсене, откриване (идентифициране), обобщение на софтуерните изисквания от наличните източници.

- *Включва работа* за да се:
 - проучи приложната област
 - анализират внимателно исканията на заинтересованите лица и
 - ограниченията като бизнес процеси, законодателство и др.
- *Участници в процеса:* крайни потребители, мениджъри, инженери, които участват в поддръжката, експерти в областта, търговски съюзи, т.е. всички *заинтересовани страни*.

Деятности по идентифицирането на изискванията – четири измерения

- **Разбиране на приложната област**

Знанията за приложната област; Източници на информация

- **Разбиране на проблема**

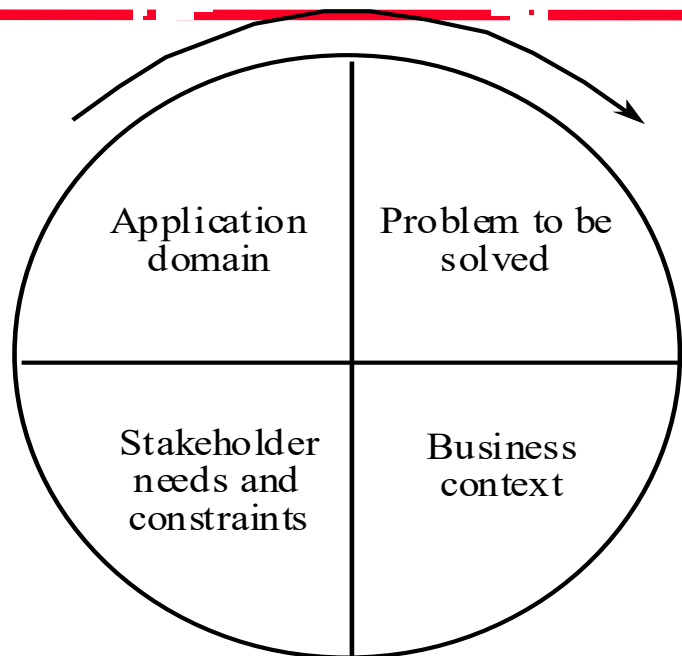
детайлите за специфичния клиентски проблем, предоставени от различни хора

- **Разбиране на бизнеса**

Трябва да разбере, как си взаимодействат системите и как спомагат за общите бизнес цели.

- **Разбиране на нуждите и ограниченията на заинтересованите лица**

Различните заинтересовани лица имат различни изисквания и могат да ги изразят по различни начини.



Идентифициране на изискванията – начална информация

Събиране на достатъчно информация за дефиниране на “проблема”

Кой проблем трябва да се разреши? (определяне на границите на проблема)

Къде е проблемът? (изясняване на контекста/проблемната област)

На кого е проблемът? (идентифициране на заинтересованите страни)

Защо е необходимо да се разреши? (определяне на целите на заинтересованите страни)

Как се проявява проблемът? (събиране на сценарии)

Кога трябва да се реши? (определяне на ограниченията върху разработката)

Какво може да попречи на разрешаването? (определяне на осъществимостта и риска)

The
journalist's
technique:
What?
Where?
Who?
Why?
When?
How?
(Which?)

Да станеш експерт в проблемната област (или просто съвети ☺)

Научете се как да се ориентирате бързо в нова проблемна област.

Използвайте своето първоначално незнание като извинение, за да задавате въпроси.

Признавайте експертните знания в областта на хората, с които разговаряте.

Идентифициране на изискванията

**Идентифициране на изискванията.
Изходна информация**

Начална информация за идентифициране на изискванията

Граници

- Определяне на обхвата на проблема

Заинтересовани страни/лица

- Определяне на носителите на (бизнес) изискванията (problem owners)

Цели

- Определяне на критериите за успех

Граница на системата

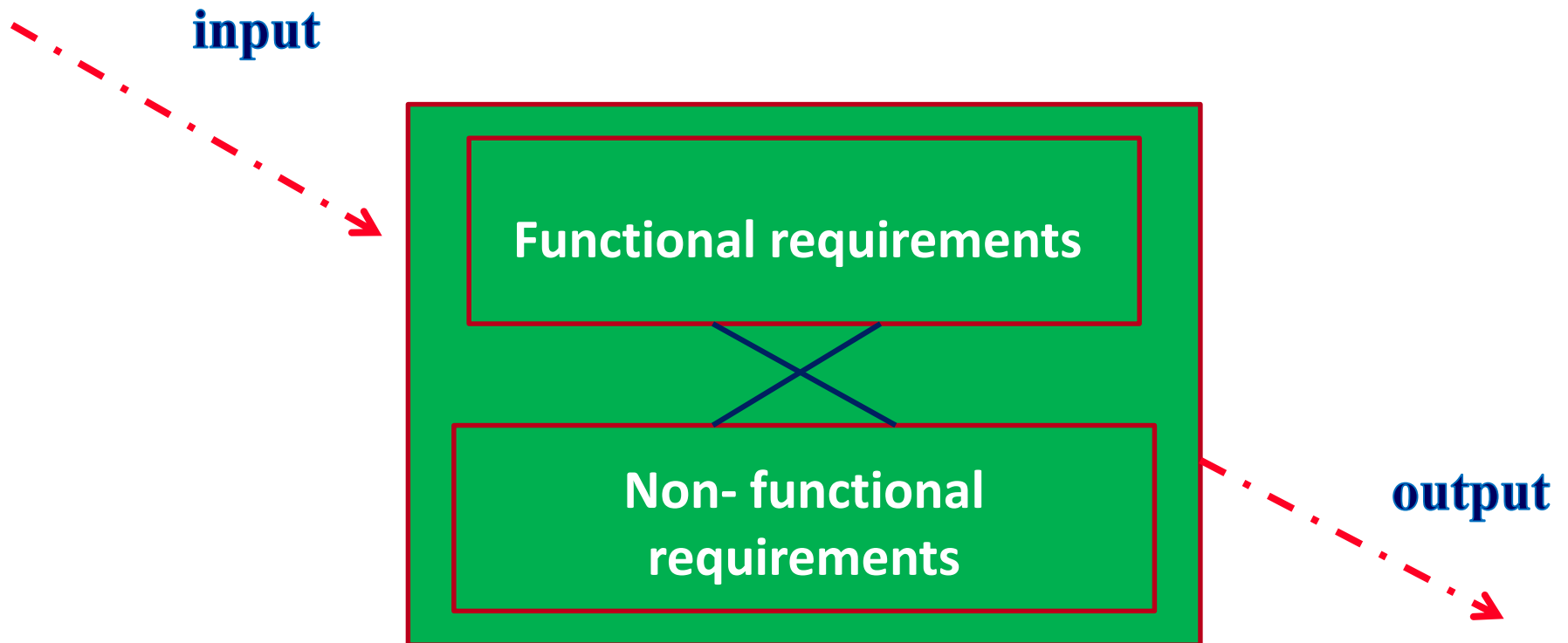
- **Границата на системата отделя планираната система от заобикалящата я среда.**
- **Обхватът на системата се определя от границата на системата.**
 - Продуктът (разработваната система) може да се моделира и да се променя по време на разработката.
 - Средата не може да се променя в рамките на този проект.

Контекст на системата - 1

- Границата на контекста *разделя частта от средата, съществена за планираната система от частта, която е без значение.*
- Контекстът на системата е от значение за изясняването и дефинирането на системата.
- Частта от света извън контекста е без значение в проекта.

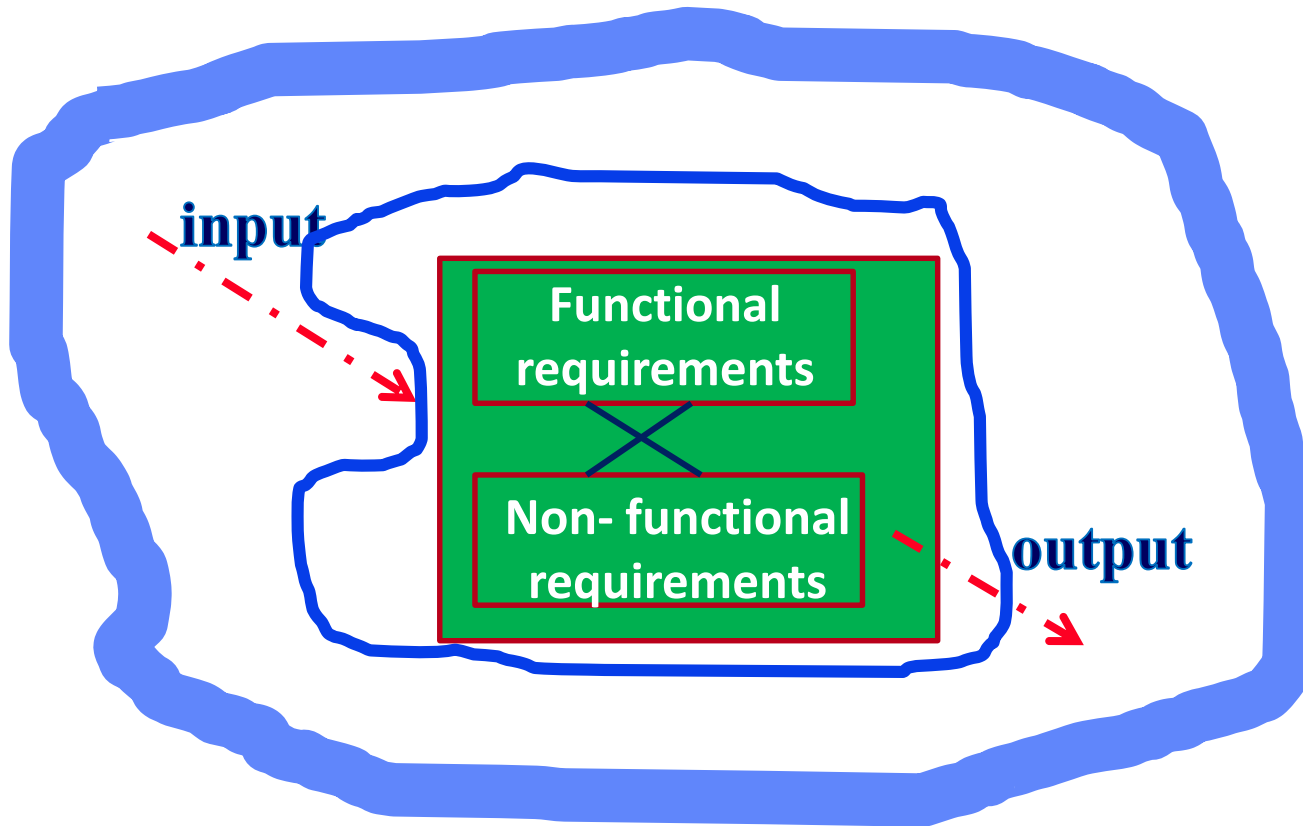
Контекст на системата - 2

Системата/продуктът и границата на системата са засегнати от контекста:



Определяне на границите

Границата на системата и границата на контекста могат да се променят по време на процеса на ИИ:



Boundary zones are fact of life.

Инженерингът на изискванията се ръководи от контекста на системата чрез съществуващите в него източници на изискванията:

- Хора
- Съществуващи системи
- Процеси
- Събития
- Документи

*В началото са идеи, желания, нужди, решения, предложения,
възможности*

Това трябва да се превърне в: цели и изисквания

Диаграма на контекста

Диаграмата на контекста се състои от:

- **Кутия** (box) (черна), представляваща системата;
- **Кутии с етикети** за всеки от външните източници на информация и за
- **Кутии с етикети за получателите на информация за системата**
- **Именуваните интерфейси между източниците и получателите на информация от системата.**

Никой от източниците не трябва да се забравя!

- Пропускането на някой от източниците може да се окаже фатално при идентифицирането на изискванията
- Фактори за избора на източник:
 - знаещ/осведомен човек
 - безопасност при събиране на информацията
 - Политически и/или икономически ограничения
- *Забележка:* Понякога не знаете какво не знаете – затова използвайте *списък и/или диаграма*.

Заинтересовано лице/страна

Заинтересованото лице е всеки, който по някакъв начин е засегнат или оказва влияние върху продукта/системата:

- човек
- организация
- група от хора, които имат обща заинтересованост

- Анализ на заинтересованите лица:
Идентифициране на ролите на заинтересованите лица.
- Примери на заинтересовани лица

Влияние на заинтересованите лица

- **Положително**

предоставят *изисквания, пари, знание ...*

- **Отрицателно**

противопоставят се на продукта и/или предоставят погрешни изисквания, спират ресурси, необходими за ИИ ...

Заинтересовани страни

- **Анализ на заинтересованите лица:**

Да се определят всички хора, с които ще се дават консултации по време за исканата информацията.

- **Примери на заинтересовани лица**

Потребители

за тях са важни възможностите и функционалността на новата система

Дизайнери

искат да изградят перфектна система или да използват съществуващ код

Системни аналитици

искат да “разберат изискванията правилно”

Персонал, който отговаря за обучението и поддръжката

искат да се уверят, че новата система е използвана и управляема

Бизнес анализатори

искат да се уверят, че “ние сме по-добри от конкуренцията”

(Технически) автори

те ще подготвят потребителски наръчници и друга документация за новата система

Мениджър на проекта

иска да завърши проекта навреме, в рамките на определения бюджет, и да бъдат постигнати всички подцели

“Клиентът”

Иска да получи най-доброто за инвестираните парични средства!

Техники за определяне на заинтересованите лица

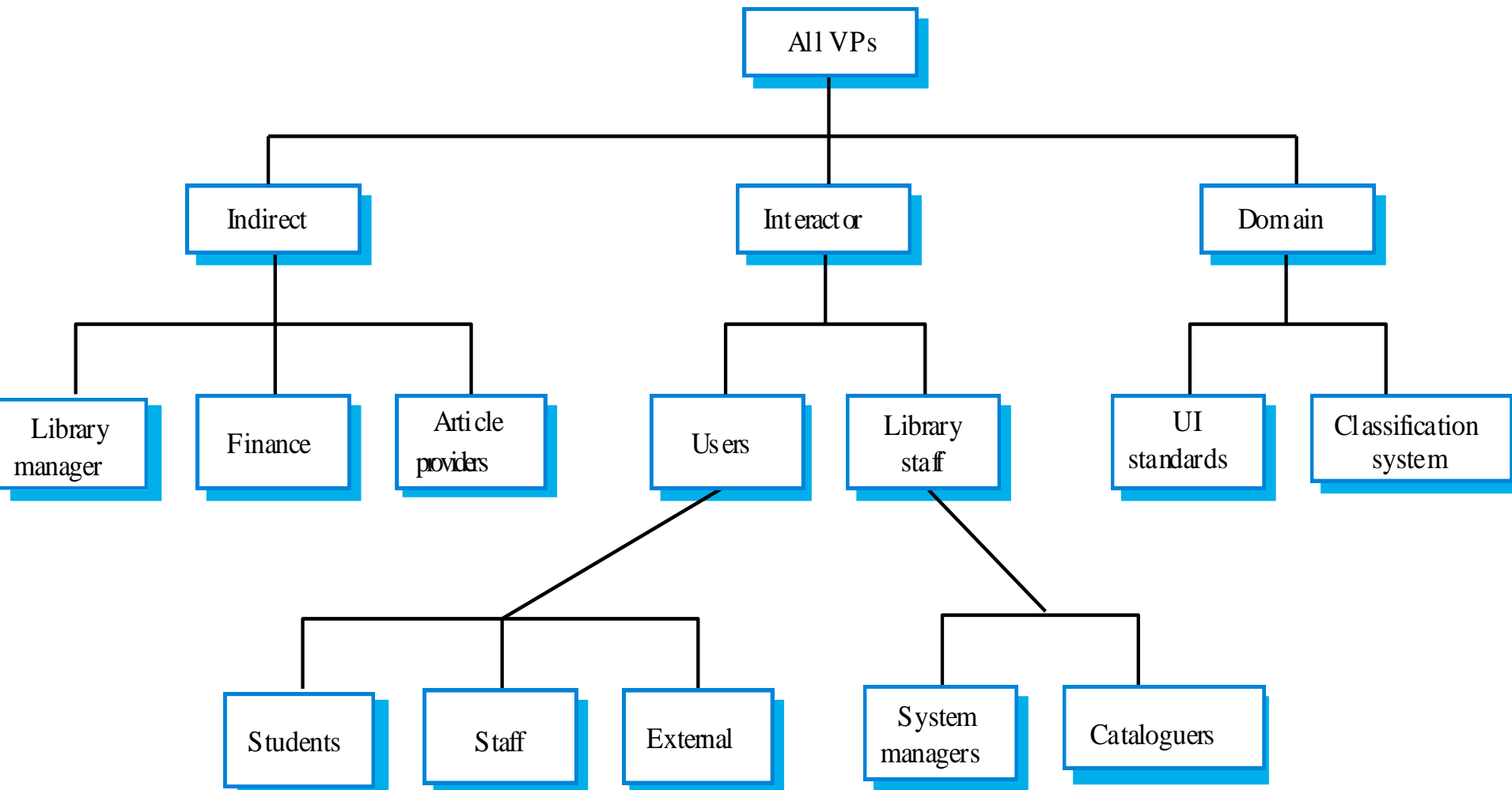
- *списъци*
 - въпроси: **кой** *знае, работи, чувства, получава полза или вреда ...*
- *класифициране* на заинтересованите страни и идентифициране на конкретен човек за отделните роли
- *мозъчна атака* (***brainstorm***) срещи за откриването на идеи, заинтересовани лица
- ***mind-maps*** за формиране и откриване на идеи.



[Yu 1997]
[van Lamsweerde 2001]

Glinz and Wieringa 2007]

Пример: LIBSYS library system



Проблеми на идентифицирането на изискванията

- обобщение

- Заинтересованите лица (ЗЛ) *не знаят* какво точно искат.
- ЗЛ изразяват своите изисквания *на свой собствен език*.
- Различните ЗЛ могат да имат *противоречиви изисквания*.
- *Организационните и политически фактори* може да окажат влияние върху системните изисквания.
- Изискванията *се променят по време на процеса на анализ*. Може да се появят нови ЗЛ; бизнес средата се изменя.
- Знанията за приложната област трябва да се съберат от различни *източници* като наръчници, ръководства, хора
- *Няма достатъчно време*
- *Жаргоните и терминологията* варират за различните компании
- *Хората не искат да се променят*

Цели

Крайната цел е намерение относно свойствата и/или употребата на системата

- Дефинира се различно ниво на целите:
 - Високо ниво – крайна цел
 - Подцели – декомпозиция на крайната цел
(**refinement-усъвършенстване**)
- **Подцелите дефинират съответни сценарии**

Пример:

Цел: Да се разработи система за управление на банкова сметка.

Подцели: Внасяне, теглене, депозирание, прехвърляне ...

Видове техники за извличане на изискванията

- Питащи (задаване на въпроси)
- Колаборативни
- Изграждане и разиграване (на прототип, сценарий, story др.)
- Наблюдение

Техники за извличане на изискванията

1. **Сценарии**
2. **Интервюта, въпросници, анкети**
3. **Мозъчна атака**
4. **Семинари (срещи) за изискванията; FAST**
5. **Soft systems methods**
6. **Наблюдения и социален анализ**
7. **Прототипиране**
8. **Software Story**

Други: Повторно използване на изискванията

Сценарии и потребителски случаи

1. Сценарии - 1

- Сценариите са **примери за реалното използване на системата.**
- Точно определена последователност на взаимодействие между актьор и система
- Те трябва да включват
 - Описание на началното състояние;
 - Описание на редовната последователност на събитията;
 - Описание на това, какво може да се сгреша;
 - Информация за други успоредни дейности;
 - Описание на състоянието, когато сценарият завърши.

Пример:

Прием в болница – какво и как се случва по време на приемането?

Типичен отговор: “Вие, или човекът, който ви придружава, ще говори с лицето на рецепцията. Трябва да покажете своята здравна карта и да обясните кой ви е насочил към болницата. След това да опишете състоянието си. Да бъдете прегледан от лекар и ...”

Сценарии - 2

Сценарий:

Обикновено са кратки (3 и 7 стъпки)

Описват: *положителни* (необходимо поведение)

отрицателни (нежелателни взаимодействия)

Могат да бъдат *показателни* (описват текущата система) или *незадължителни* (как трябва да бъде)

Предимства

Много са *естествени*: обикновено заинтересованите лица ги използват спонтанно

Кратките сценарии са много добри за бързо илюстриране на специфични взаимодействия

Недостатъци

Липса на структура; Трудно се проверяват за пълнота

Пример: LIBSYS сценарий (1)

Първоначално предположение: Потребителят е вписан в системата LIBSYS и е открил документа, който съдържа копие на статията.

Нормален: Потребителят избира статията, която ще бъде копирана. След това системата изисква от потребителя да предостави информация за абониране за документа или да укаже как ще плати за статията. Други възможни методи за плащане са с кредитна карта или чрез предоставяне на номер на сметка. След това на потребителя се представя форма за авторски права, която той трябва да попълни и която запазва детайли за трансакцията, след което потребителят я изпраща на системата LIBSYS.

Формата за авторски права се проверява и ако е валидна, PDF версията на статията се зарежда в работната област на LIBSYS на потребителския компютър. Потребителят се уведомява, че статията е налична. От потребителя се иска да избере принтер и се отпечата копие на статията. Ако статията е била отбелязана като “само за принтиране”, тя се изтрива от потребителската система, след като потребителят потвърди, че принтирането е завършило.

Пример: LIBSYS сценарий (2)

Какво може да се обърка: Потребителят може да не успее да попълни правилно формата за авторските права. В такъв случай, формата трябва отново да се покаже на потребителя за извършване на корекции. Ако формата отново е погрешна, на потребителя се отказва исканата статия.

Заплащането може да не бъде прието от системата. На потребителя се отказва исканата статия.

Свалянето на статията може да прекъсне. Прави се нов опит до постигане на успех или потребителят прекратява сесията.

Може да не е възможно да се принтира статията. Ако тя не е отбелязана като “само за принтиране”, тогава тя се задържа в работното пространство на LIBSYS. В противен случай, статията се изтрива и потребителският акаунт is credited с цената на статията.

Други дейности: Едновременно сваляне на други статии.

Състояние на системата при завършването: Потребителят е вписан. Свалената статия е изтрита от работната област на LIBSYS, ако е била отбелязана само за четене.

Потребителски случаи - структуриран метод за идентифициране и анализ на изискванията

- **Потребителските случаи са техника на структурно представяне, базирана на сценарии, която идентифицира *актьорите* в едно взаимодействие и която *описва самото взаимодействие*.**
- **Наборът от всички потребителски случаи за една система-описва всички възможни взаимодействия със системата.**
- **Диаграмата на потребителския случай се допълва с:**
 - **Диаграма/и на последователността** добавя детайли към потребителските случаи като показва последователността на обработката на събитията в системата.
 - **Диаграма на дейностите**

Потребителски случаи - допълнение

Потребителските случаи са предложени в Objectory method (1993). Те групират един главен сценарий със съответните алтернативни сценарии и изключения.

Деф.: Спецификацията от последователности от действия, включително варианти от последователности и последователности от грешки, които една *система*, *подсистема*, или *клас* могат да извършват като си взаимодействат с външни обекти, за да предоставят услуга или стойност.

Диграма на потребителските случаи

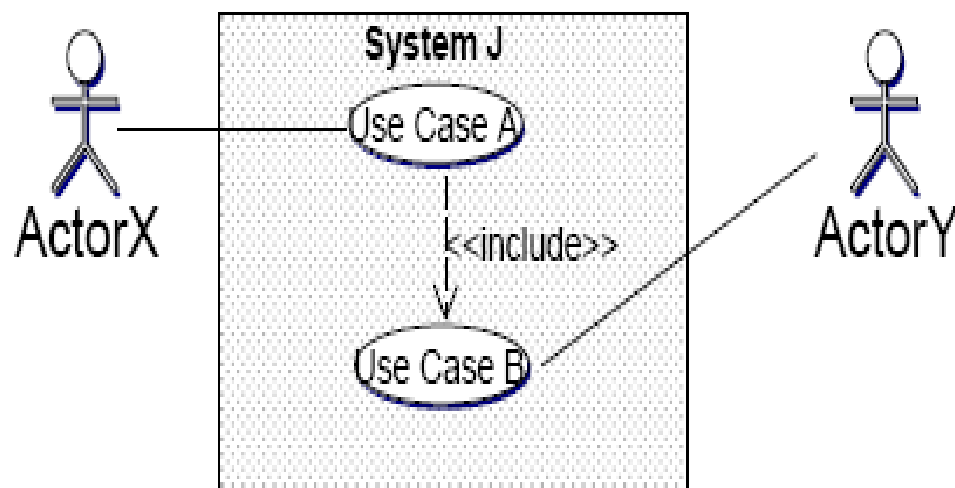
Диаграмата на потребителските случаи за една система представя:

- *Границата* на системата;
- *Потребителски случаи* от високо ниво в системата (генерализации);
- *Актьори*: хора или системи от контекста
- *Връзки* между потребителските случаи и актьорите;
- *Връзки* (ако има такива) между потребителските случаи и/или между актьорите.

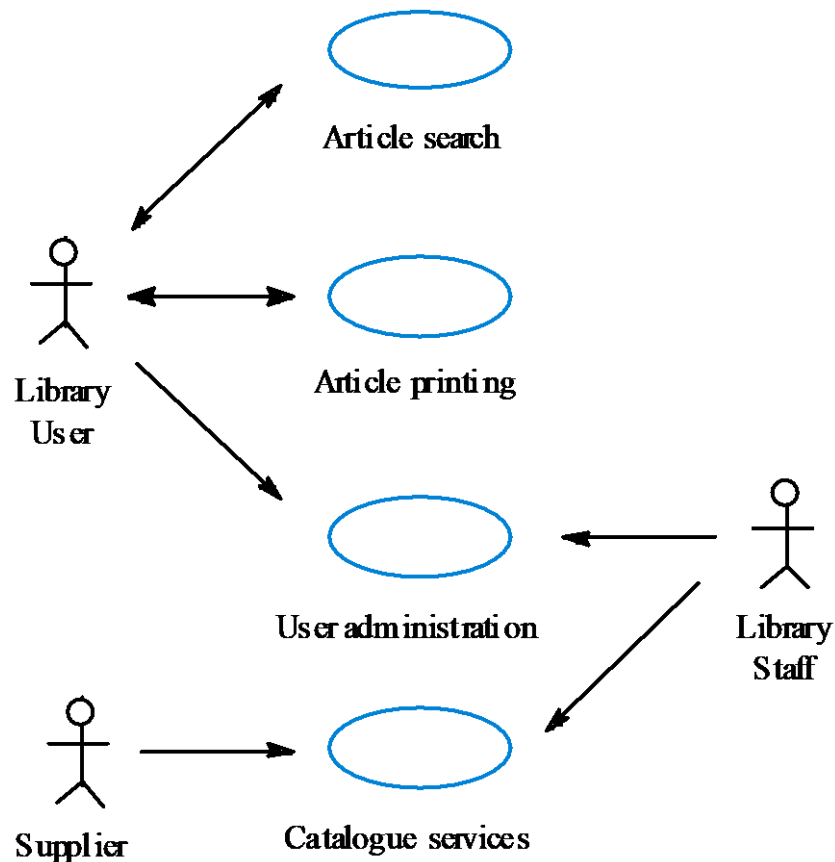
Потребителски случаи в UML

- UML предоставя графично представяне на потребителските случаи, наречено *Диаграма на потребителските случаи (Use Case diagram)*.
- То позволява графично изобразяване на:
 1. Актьори
 2. Функции
 3. Връзки
 4. Зависимости: include, extend, генерализации (обобщения)

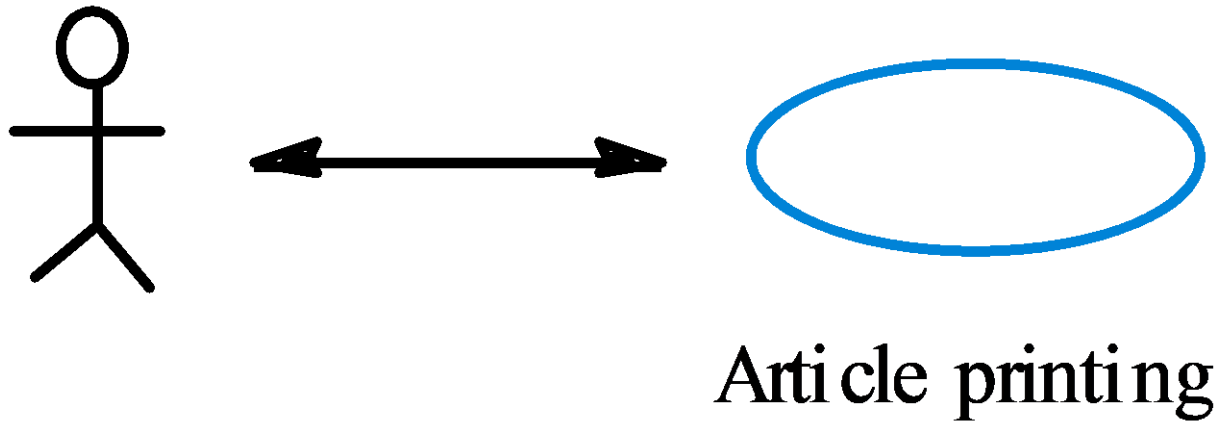
Дефинира границата на системата



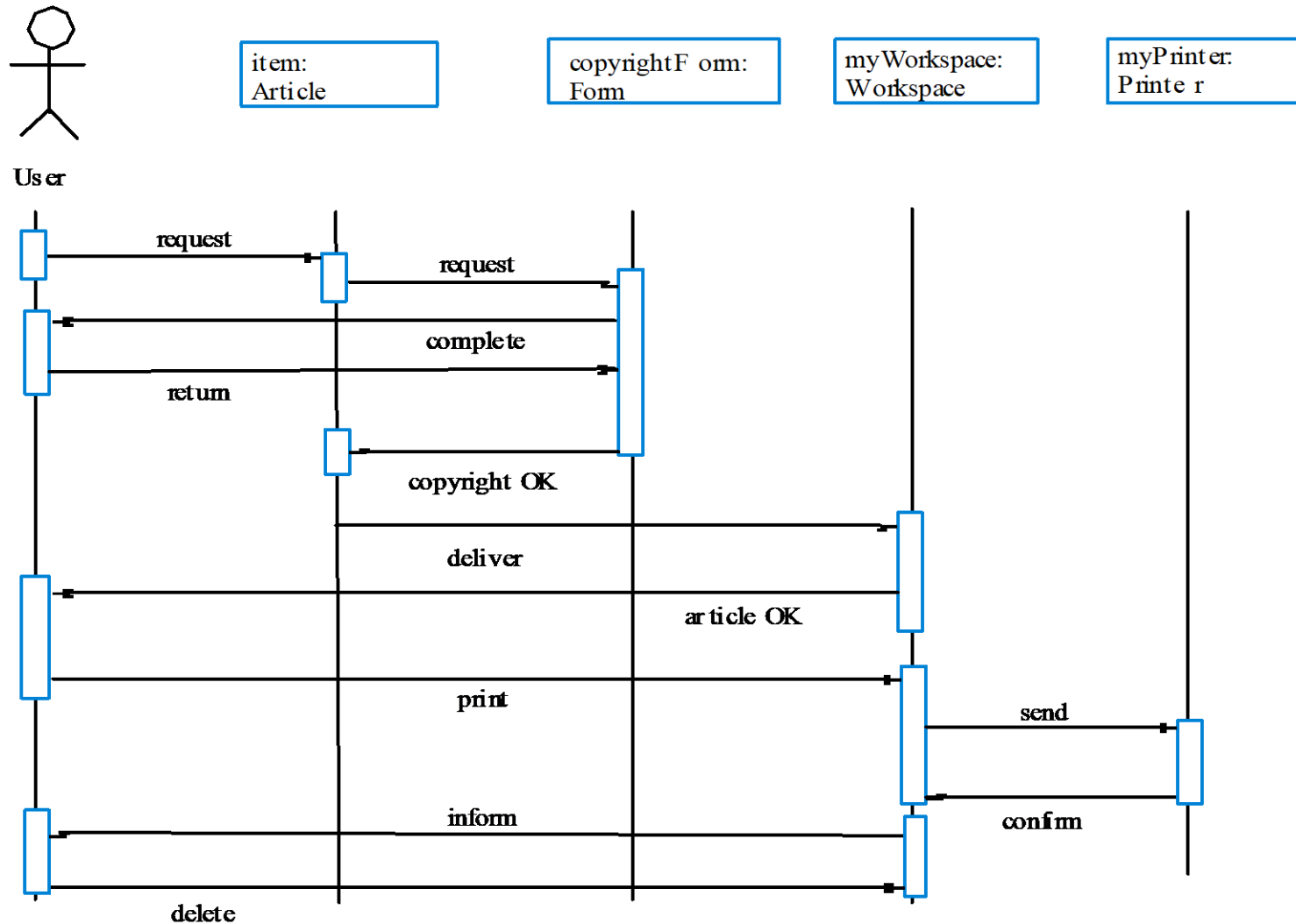
Пример: Потребителски случаи в LIBSYS



Пример: Потребителски случай за принтиране на статия



Пример: Диаграма на последователност за случай “принтиране на статия”



2. Интервюта -1

- Интервюирането е основен източник на информация за изискванията
- Интервюирането изисква:
 - търпение,
 - изслушване,
 - разбиране и
 - задаване на разнопосочни въпроси.

Интервюта - 2

Видове интервюта:

- **Затворени интервюта** - търсят се отговори на предварително подготвени въпроси
- **Отворени интервюта.** Няма предварително подготвени въпроси. Изискванията се обсъждат в диалог.

.... but blurred ...

Съществени елементи на интервюирането

- Интервюиращите трябва да бъдат *непредубедени* и да не провеждат интервюто с предварително формирани идеи за това какво се изисква.
- На ЗЛ *трябва да се даде стартова точка* за дискутиране.
- Трябва да се има предвид организационната политика – някои от реалните изисквания може *да не се обсъждат с конкретно ЗЛ* заради определени (политически в общия смисъл) ограничения.

Характеристики на интервюирането

Интервюто се фокусира върху въпроси за:

Бизнес случай за проекта

Функционални изисквания за проекта

Нефункционални изисквания

Рискове

Ограничения

Заинтересовани лица

Какво не може да се открие по време на интервю?

- изясняване на областта на системата
- знание за организацията

Рядко са напълно достатъчни за инженеринга на изискванията

3. Мозъчна атака (Brainstorming)

Метод за генериране на идеи в група

Brainstorming е техника за работа в екип, която **предизвиква откриването на знания** от всеки участник - надграждането на идеите на другите, освобождаване на мисленето и разглеждане на проблема от различни гледни точки.

- Помага при разглеждането на *алтернативи* и правене на правилни избори
- Включва не само дискусия, но и сесия за *съвместно* управление на *знания*.
- Едно от най-добрите средства да се определи дали получените изисквания са правилни или не.
- Продължителност и участници

4. Работни срещи (Requirements workshops)

Аналитикът **събира** (основни) заинтересовани страни, за да анализира системата и да разработи решението. **Изискванията се идентифицират и дискутират съвместно.**

В идеалния случай се провеждат в *контролирана среда* чрез обмен на знания.

Ръководителят на срещата води процеса/дискусията *фокусирано*, а *протоколчик* документира дискусията.

Често са необходими *множество срещи*, за да приключи процесът успешно.

Недостатъци: Може да се окаже трудно да се съберат всички нужни заинтересовани страни едновременно.

/Facilitated Application Specification Technique (FAST) – вид работна среща

Има за цел да покрие празнината между мисленето на разработчиците и желанието на клиентите.

Всички заинтересовани страни се считат за част от екип, за който се организира следното:

- Среща на неутрално място
- Правила за подготовка и участие
- Дневен ред – формално + неформално
- Водещият управлява срещата
- Механизъм на дефинирането (начин на провеждане на срещата)
- *Идентифициране на проблем, предлагане на решение, уговаряне на подходи, определяне на решение чрез набор от изисквания.*

Каква е разликата между работните срещи мозъчна атака?

5. Soft Systems methods (SSM)

- ✓ Чрез тях се получават *неформални модели* на цялостната социално-техническа система.
- ✓ SSM обръщат внимание на това, че софтуерните системи са вградени в един по-широк човешки и организационен контекст.
- ✓ **Това не е техника за детайлно идентифициране на изискванията. Те са по-скоро начин да се разбере даден проблем и неговия контекст в организацията.**
- ✓ (Задължително) се съчетават с други техники на ИИ.

Етапи на SSM

1. Оценка на проблемната ситуация
2. Описание на проблемната ситуация
3. Абстрактна дефиниция на системата от избрани **гледни точки**
4. Концептуално моделиране на системата (human activity models)
5. Сравнение модел/реален свят
6. Идентификация на промяната
7. Препоръки за действие

Пример: Система за оповестяване на наводнения

6. Наблюдение и социален анализ

- Значимост на техниката – защо и кога се прилага; исторически корени.
- **Етнографът прекарва известно време в наблюдение на хората по време на работа и така си изгражда представа за това, как се извършва работата.**
- **Етнографията може да се използва за извличане на:**
 - а) социални изисквания, но за конкретен краен потребител (Пример: Социална мрежа)**
 - б) организационни изисквания (Пример: Обслужване на клиент)**

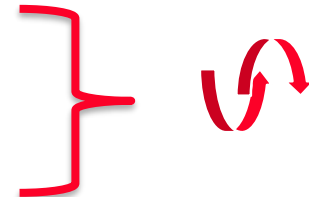
Подход и характеристики на етнографското изследване

Етнографските проучвания са ефективни, когато *работата е по-обширна, сложна и динамична и е невъзможно представяне с опростени системни модели.*

- ❖ Хората не трябва да обясняват или да говорят за своята работа, а да бъдат наблюдавани.
- ❖ Ефективността на етнографските проучвания зависят от:
 - личния характер на етнографа,
 - типа на процеса и
 - на участващите ЗЛ.

Насоки за провеждане на етнографското изследване

- Приемете, че хората *са добри* във вършенето на своята работа. Търсете нестандартните начини за работа.
- Отделете време *да опознаете хората* и *да установите връзка на доверие*.
- Водете си *подробни бележки* за всички практики. Анализирайте ги и правете заключения въз основа на тях.
- Видео и аудио записи могат да са полезни, ако не са много дълги.
- Организирайте редовно *разбор (debriefing session)*, при което етнографът говори с хора извън процеса
- Комбинирайте етнографията с *други техники за идентификация (защо?):*
 - Комбинирайте с отворени интервюта
 - Комбинирайте с прототип



Етнографски перспективи на описание – механизъм за организиране и структуриране на етнографския архив

➤ **Обстановката на работата**

Описва се *контекстът и физическото местоположение* на работата и как хората използват обектите, за да изпълняват задачите си.

Пример: проучване за описание на *helpdesk*.

➤ **Социална и организационна перспектива**

Разкрива *какво се случва ежедневно* в работата така, както го виждат *различните участници*. Тази гледна точка се опитва да *обедини (различието)* на възприятията на отделните участници.

➤ **Процеса на работа**

Представя работата като поредица от дейности, при което информацията тече от една дейност към друга.

7. Повторно използване на изискванията (Requirements reuse)

- Повторното използване включва вземане на изискванията, които са били разработени за една система и използването им за друга.
- *Спестява време и усилия*, тъй като изискванията вече са били анализирани и валидирани в други системи.
- Повторното използване на изискванията *е неформален процес*, но **по-систематично** използване може да доведе до по-голямо спестяване на разходите (*Пример: Design patterns*).

Възможности за повторно използване

- Когато изискването се отнася до *предоставяне на информация за приложната област (ограничения на системата)*.
- Когато изискването се отнася до *стила* на представянето на информацията (*Пример: характеристики на интерфейса*). Повторното използване води до съгласуваност на стила между приложенията.
- Когато изискването отразява *политиките на компанията*, като например *политиката за сигурност, за съхранение на лични данни, етика на диалога с потребители ...*

Повече от 50% от изискваният може да попаднат в тази група!

Възможни са и проблеми (какви?).

Други изследвания

Идентифицирането на изискванията може да включва и *проучване на документи*:

- Индустриални стандарти, закони, и/или наредби
- Литература за продукта (собствени или на конкуренцията)
- Документация на процеси и инструкции за работа
- Заявки за промяна, доклади за проблемите или помощни доклади
- Научени уроци от предишни проекти и изработени продукти
- Отчети и други резултати от съществуващи системи