

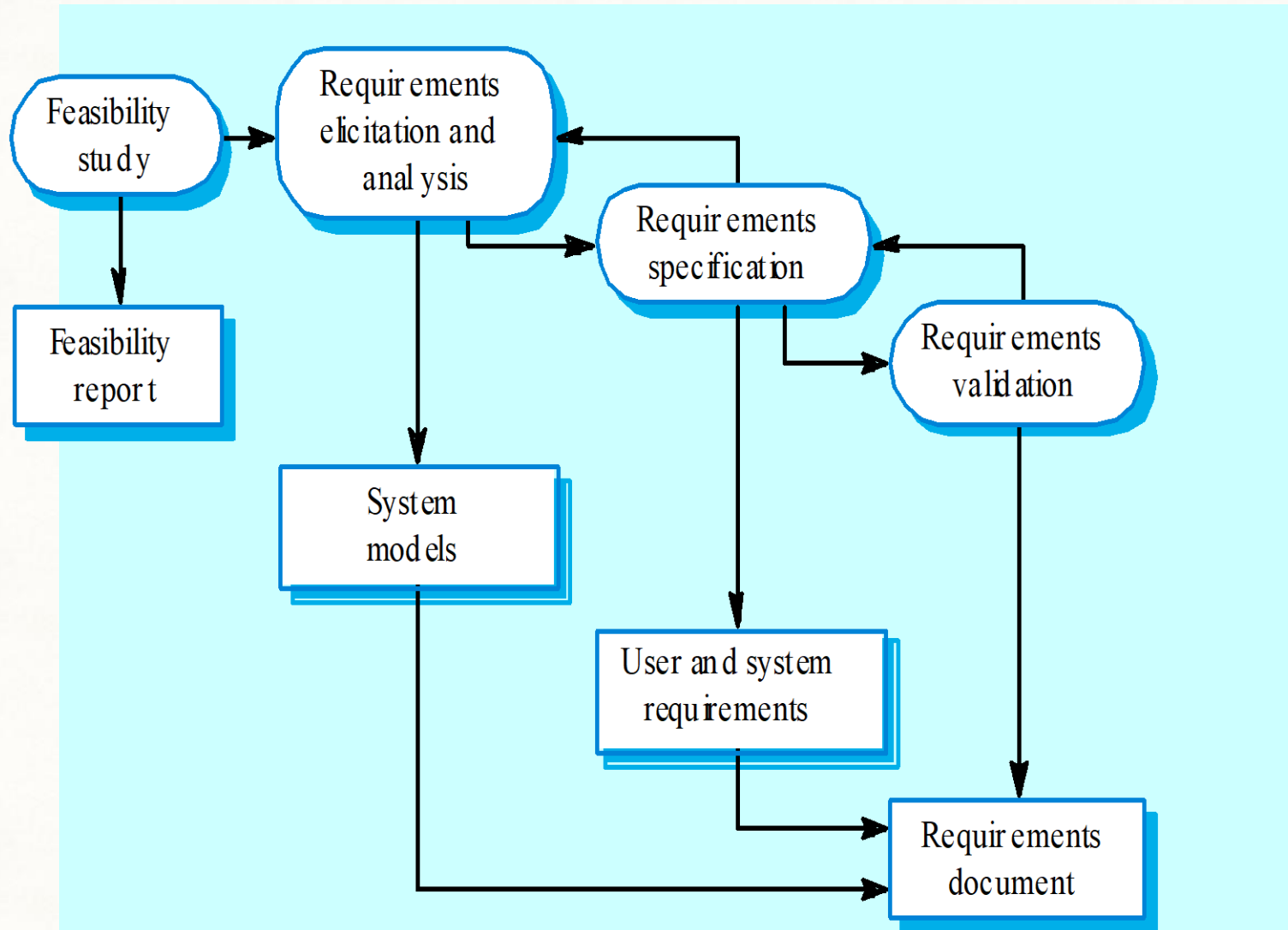
Валидиране на изискванията

Лекция 7

Съдържание

1. Валидиране на изискванията – същност и особености
2. Техники за валидиране на изискванията:
 - Преглед/рецензия на изискванията
 - Прототипиране
 - Валидиране на модел
 - Тестване изискванията

Процес на инженеринг на изискванията



Requirements validation

- Concerned with demonstrating that the requirements define the system that the customer really wants.
- Requirements error costs are high so validation is very important
 - Fixing a requirements error after delivery may cost up to 100 times the cost of fixing an implementation error.

Sommerville

Валидиране - същност

to validate - 1. потвърждавам, узаконявам, ратифицирам

- Удостоверява, че документът, описващ изискванията, е приемливо описание на системата.
- Документът на изискванията се проверява за:
 - Пълнота и последователност (completeness and consistency);
 - Конфликт в изискванията;
 - Неясни, двусмислени изисквания;
 - Съответствие със стандартите;
 - Грешки в моделите на системата;
 - Технически грешки
 - ...
- Участници в процеса на валидиране

Анализ и валидиране – общото и различното

- **Анализът** работи със суровите изисквания, директно извлечени от заинтересованите лица.
 - “Дефинирахме ли правилните изисквания /Have we got the right requirements?”
- **Валидирането** работи с крайния чернови вариант на формулираните изисквания т.е. с вече договаряни и съгласувани изисквания.
 - “Дефинирахме ли правилно изискванията / Have we got the requirements right?”

Основен проблем на валидирането на изискванията

- *Не съществува документ*, който да е база за сравнение за оценка на коректността;
- Критерият е:
 - 1) ясно описание и
 - 2) пълен отговор на всички изисквания на заинтересованите лица.

Валидиране на изискванията – основни проверки

- **Валидност:**

Does the system provide the functions which best support the customer's needs?

- **Консистентност:**

Are there any requirements conflicts?

- **Пълнота:**

Are all functions required by the customer included?

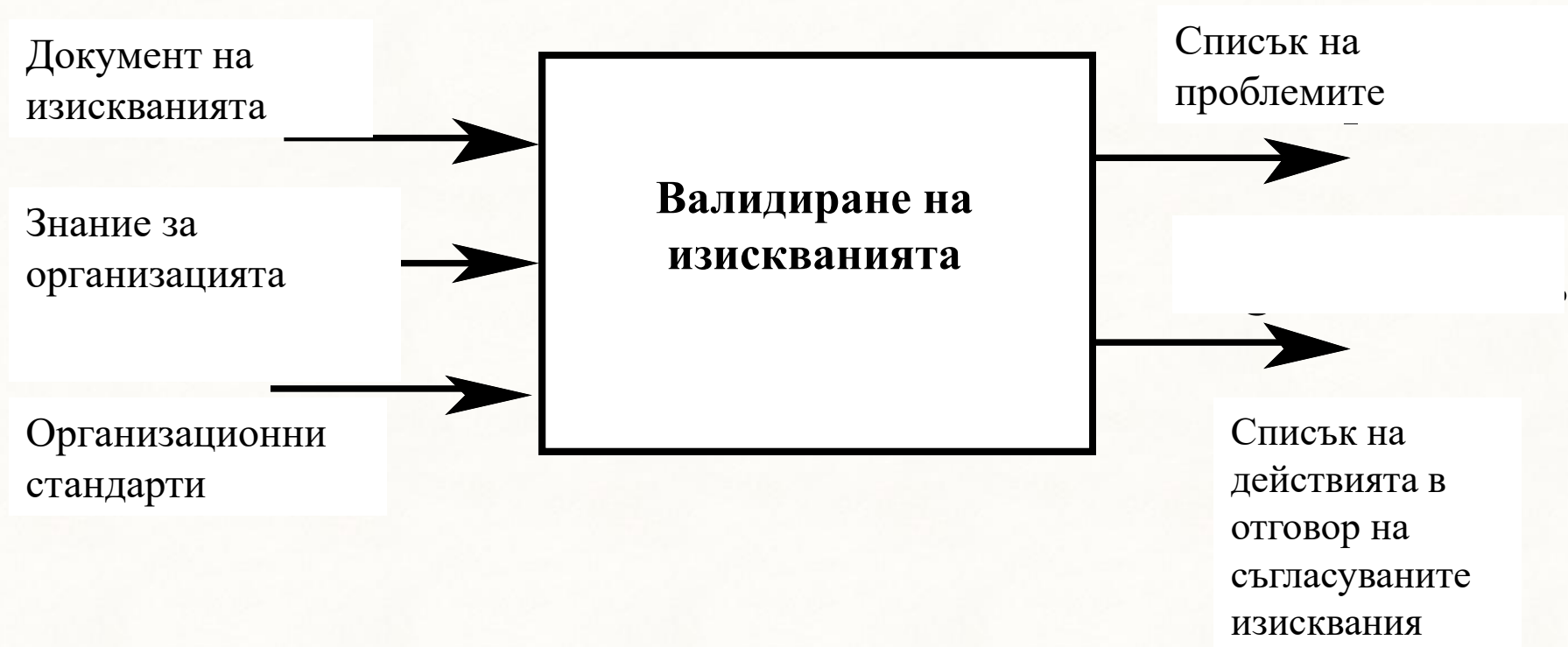
- **Реалистичност:**

Can the requirements be implemented given available budget and technology

- **Проверимост:**

Can the requirements be checked?

Входна и изходна информация за процеса на валидиране



Входна информация за процеса на валидиране

- **Документ на изискванията**
 - Би трябвало да е цялостна завършена версия на документа. Изготвянето и форматът на документа е съобразен със стандартите на организацията.
- **Знание за организацията**
 - Знание (често което се подразбира) за организацията – използвана терминология, умения на работещите, култура, структура ...
- **Организационни стандарти**
 - Трябва да бъде проверено спазването на исканите стандарти.

Изходна информация за процеса на валидиране

- **Списък на проблемите**, открити в описаните изисквания:
 - *Идеален вариант*: класификация на проблемите по тип.
- **Действия в отговор на съгласуваните изисквания**
Списък от действия, които да отговорят на откритите проблеми в дефинираните изисквания.

Забел.: Не е задължително 1:1 съответствие между проблеми и действия!

Техники за валидиране на изискванията

- **Преглед на изискванията (Requirements reviews)**
 - Систематичен ръчен анализ на изискванията.
- **Прототип**
 - Използване на изпълним модел на системата за проверка на изискванията.
- **Валидиране на модели**
 - Проверка за точността на моделите, представени в спецификацията
- **Генериране на тестови случаи**
 - Разработване на тестове за изискванията (to check testability).

Техники за валидиране на изискванията - 1

1. Преразглеждане / преглед / рецензиране (review) на изискванията

- **Най-често използваната техника** за валидиране на изискванията:

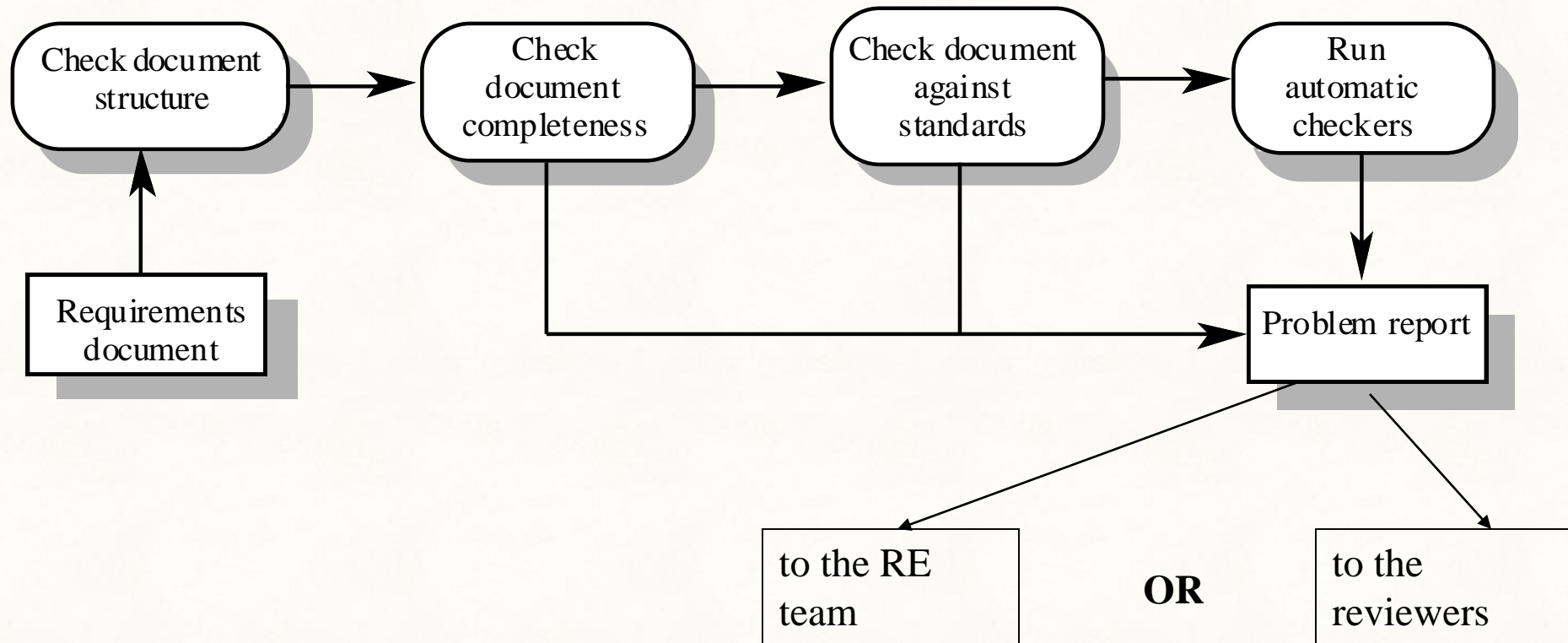
Формира се специална група, която да се

- 1) запознае с изискванията и да ги анализира;
- 2) формулира забелязани проблеми,
- 3) дискутира възможните действия за справяне с проблемите.

Предварителен преглед

- Разходът за прегледа на изискванията би могъл да бъде намален, ако се направи предварителна проверка от един човек. Търсят се явни пропуски - пълнота на документа, съгласуваност със стандартите, печатни грешки ...
 - Автоматични проверки
 - Техническо оформление
- Документът може да бъде върнат за коригиране или списък от проблеми да бъде разпространен между участниците в екипа.
- Кой осъществява предварителния преглед?
- Продължителност на предварителния преглед.

Процесът на предварителен преглед може да бъде схематично представен



Дейности на прегледа на изискванията (1)

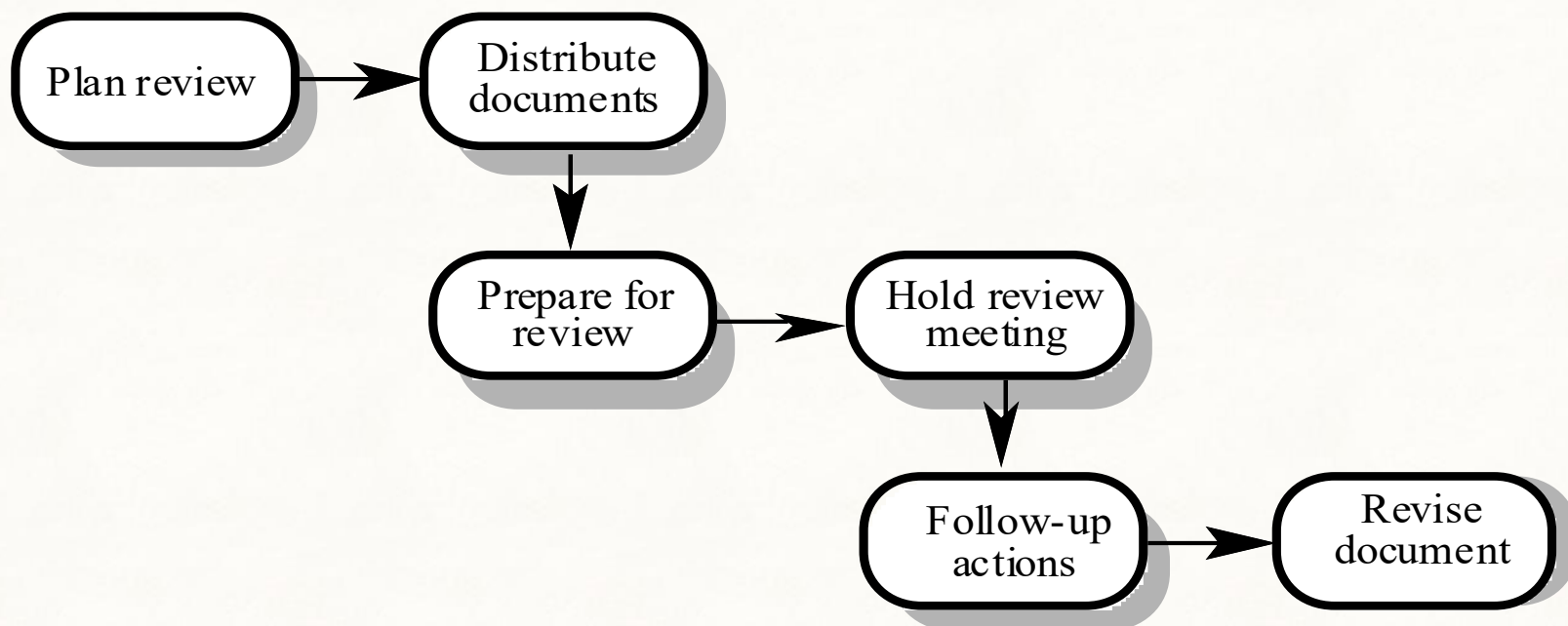
- **Планиране**
 - Определя се екип, ръководител и протоколчик; избира се време и място за срещите.
- **Разпространение** на документите за преглед до участниците в екипа.
- **Подготовка** за прегледа:
 - Всеки рецензент да прочете документа за изискванията и да набележи открити конфликти, пропуски, непоследователности, отклонения от стандартите, др. проблеми.

Дейности на прегледа на изискванията (2)

- **Среща по прегледа**
 - Дискутират се коментари и проблеми от отделни участници от екипа и се дефинира набора от възможни действия за справяне с проблемите.
- **Проследяване на действията**
 - Ръководителят на екипа проверява дали формулираните действия са изпълнени.
- **Ревизиране на документа**
 - Документът на изискванията се редактира, за да отрази съгласуваните действия. На този етап той може да бъде предаден като окончателен вариант или върнат за нов преглед.

Кой може да бъде ръководител на екипа по преглед на изискванията?

Модел на процеса на преглед на изискванията



(Възможни) Действия за решаване на всеки отделен проблем

- **Изясняване на изискването**
 - лошо представено или да има случайно изтървана информация.
Презаписване на изискването.
- **Липсваща информация**
 - Отговорност на инженерите по изискванията е да *открият* тази информация от съответен източник.
- **Конфликт в изискванията**
 - Ако съществува значим конфликт в изискванията, то трябва да се *преговаря*, за да достигне до съгласие.
- **Нерелистични изисквания**
 - Изискването не би могло да бъде реализирано, защото има технологични и/или друго ограничение върху системата. Необходимо е клиентите да се консултират, как да направят изискването *по-реалистично*.

Време за преглед: 2-4 изисквания за час!

Каква е разликата от справяне с програмни грешки?

....

Участници в екипа на прегледа

- Част от рецензентите трябва да включват **клиенти** с различни специалности, за да се
 - използват различни умения и знания;
 - разбере по-добре необходимостта от промяна на изискванията;
- В екипа трябва задължително да се включи:
 - *експерт от специфичната област* на създаваната система, както и поне един *краен ползвател на системата*.
- В екипа е добре да бъдат включени:
 - *Клиент*
 - *Разработчик на системата*
- Брой на експертите, формиращи екипа - от 3-10 човека;
 - Организиране на работата на екипа: компромис с екипа, но по-малко с времето.

Преглед/ревью на изискванията

- Ревютата трябва да се организират регулярно
- Екипът на клиента и екипът на изпълнителя трябва да участват в ревютата.
- Ревютата могат да бъдат:
 - формални (with completed documents)
 - неформални.

Въпроси на прегледа

| | |
|--|---|
| <ul style="list-style-type: none"> • Единствено формулиране / идентифициране на дадено изискване? | Проследимост; Съответствие със стандартите |
| <ul style="list-style-type: none"> • Даден ли е речник на специализираните термини? | Разбираемост |
| <ul style="list-style-type: none"> • Разбираемо ли е само за себе си изискването или е необходимо да бъде и разгледано и друго, за да бъде разбрано даденото? | Разбираемост; Пълнота |
| <ul style="list-style-type: none"> • Използван ли е последователно един и същи термин в описанието на изискването? | Неяснота |
| <ul style="list-style-type: none"> • Изисква ли се една и съща услуга при различни изисквания? • Има ли противоречия в информацията при тези изисквания? | Консистентност; излишество |
| <ul style="list-style-type: none"> • Ако изискване се отнася и до други характеристики, те описани ли са в документа? | Пълнота |
| <ul style="list-style-type: none"> • Групирани ли са заедно изисквания, които са свързани? Ако не - взаимно отнасят/насочват ли се? | Организираност; Проследимост |

Проблеми в описаните изисквания - пример

- “4. EDDIS will be configurable so that it will comply with the requirements of all UK and (where relevant) international copyright legislation. Minimally, this means that EDDIS must provide a form for the user to sign the Copyright Declaration statement. It also means that EDDIS must keep track of Copyright Declaration statements, which have been signed/not-signed. Under no circumstances must an order be sent to the supplier if the copyright statement has not been signed.”

Problems

- Не пълно (**Incompleteness**)
 - What international copyright legislation is relevant?

Recommendation:

....

- Не ясно (**Ambiguity**)
 - Signiture?

Recommendation:

...

- Съответствие със стандартите (**Standards**)

...

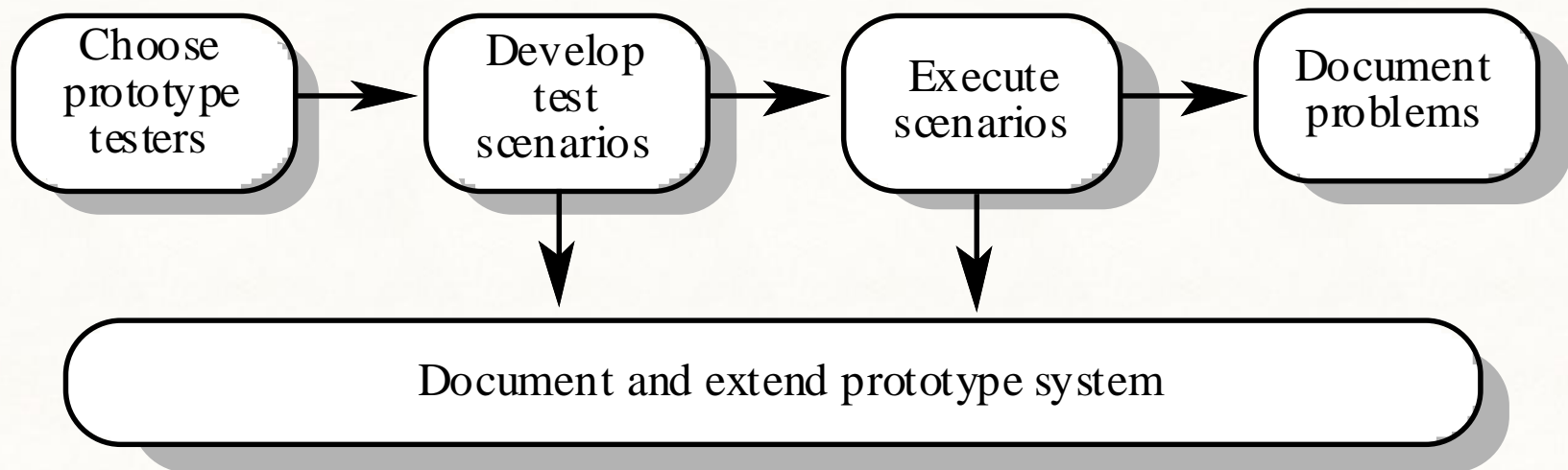
2. Прототипиране

Важно! Техниката на прототипиране в процеса на валидиране заслужава да се прилага **само** в случай, че вече е разработен прототип за целите на извличане на изискванията.

- Прототипите, предназначени за валидиране на изискванията, имат за цел да **демонстрират изискванията** и да **помогнат на заинтересованите лица да открият проблеми**.
- Трябва да бъдат достатъчно **пълни, ефикасни и робастни**, за да се оцени използването на исканата система.
- Валидирането може да започне с непълен прототип, но задължително системата трябва да бъде допълнена и довършена по време на валидирането.
- **Документация** и ръководство за ползване *трябва* да бъдат предоставени.

Прототипиране за целите на валидирането

Когато прототипите, създадени в процеса на извличане, са с непълна функционалност, то прототипът се доработва по време на валидирането:



Действия при прототипирането: успоредни процеси

- **Избор на подходящ екип за тестване на прототипа**
 - Потребители с *умерен* опит и които са “отворени” към използването на нови системи.
 - Крайни потребители с *различен характер на извършваната дейност*
- **Разработване на тестови сценарии**
 - Внимателното планиране предполага формулиране на достатъчно *пълен набор* от тестови сценарии.
- **Изпълнение на сценариите**
 - При изпълнение на сценариите потребителите е добре *да работят самостоятелно*, като по този начин симулират и реалната ситуация на използването на системата.
- **Документиране на проблемите**
 - Съставяне на електронен или хартиен *документ* с откритите проблеми.

2.1. Разработване на ръководство на потребителя

- Писането на ръководство стимулира детайлния анализ и допринася за изясняване на изискванията. (effect of requirements rewriting)
- Информация, която трябва да се съдържа в ръководството
 - Описание на функционалността и как тя е реализирана;
 - Коя част на системата не е изпълнена
 - Как да се излезе/възстанови нормалния статус при наличие на конфликт или проблем.
 - Как да се инсталира и стартира системата
- Ръководство на потребителя
 - Чернови вариант (дори и без наличие на прототип)
 - Чернова за създаване на крайния вариант на документацията на системата.
 - Цената за писане на ръководството да се включи към разходите за валидиране.

3. Валидиране на модел

Проверката на точността на създадените модели на системата е съществено в процеса на валидиране.

- **Цел на валидирането на модела**
 - *моделът е логичен, съдържа цялата необходима информация, няма конфликти (self consistent)*
 - всеки модел *точно представя реалните изисквания* на клиента
 - моделите за една система са *вътрешно и външно съвместими (internally and externally consistent)*
- Някои проверки са възможни с **автоматични** софтуерни инструменти
 - CASE tools support. (Какво може да бъде проверено?)
- **Перифразирането** на модела е ефективна техника за проверка

А) Перифразиране - Шаблони

Пример: DFD -> перифразирано описание

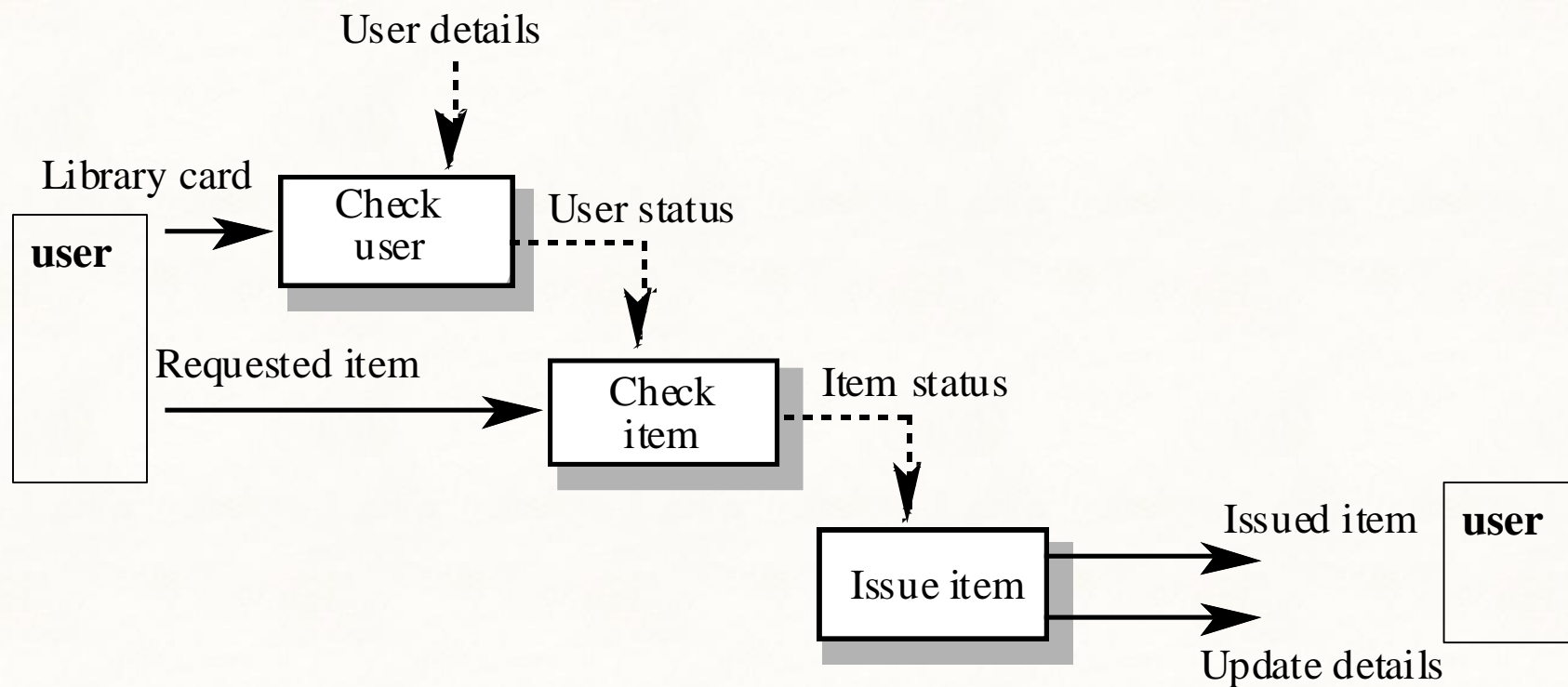
1. Име на осъществяваната трансформация (функция)
2. Източник и входна информация
3. Функционалност на трансформацията
4. Изходна информация и за къде е предназначена
5. Изключения или управляваща информация

Позволява да се открие лесно липсваща информация.

Б) “Автоматизирани” шаблони – отчети в Case tools. Проблеми.

В) Математическо доказателство при използване на формални спецификации и модели.

Пример: DFD за услугата заемане на библиотечно издание (item) от читател (user) на библиотека



Пример: Перифразиране на Data-flow диаграма за заемане на библиотечно издание

| | |
|-------------------------|---|
| Check user | |
| Inputs and sources | User's library card from end-user |
| Transformation function | Checks that the user is a valid library user |
| Transformation outputs | The user's status |
| Control information | User details from the database |
| Check item | |
| Inputs and sources | The user's status from Check user |
| Transformation function | Checks if an item is available for issue |
| Transformation outputs | The item's status |
| Control information | The availability of the item |
| Issue item | |
| Inputs and sources | <i>None</i> |
| Transformation function | Issues an item to the library user. Items are stamped with a return date. |
| Transformation outputs | The item issued to the end user Database update details |
| Control information | Item status - items only issued if available |

4. Тестване на изискванията

- За всяко изискване трябва да може да бъде *съставен тест/ове за проверка* дали изискването е дефинирано добре.
- Невъзможността да се създаде тест говори за липсваща или неясна информация в описанието на изискванията.
- **Всяко функционално** изискване в документа на изискванията трябва да бъде *анализирано с подходящ тест*. (*requirements testability*)
- *Тест за изискването или тест на системата ?*

Тестови случаи за изследване на описаните изисквания

Въпроси, подпомагащи съставянето на тест за изискванията:

- Каква *употреба* ще се тества? (*контекст на сценария*)
- Съдържа ли изискването *достатъчна информация*, за да позволи разработване на тест? (други изисквания, зависимости между тях?)
- *Един* или повече тестове са нужни, за да се направи изследването? (**Какво евентуално може да показва необходимостта от повече тестове?**)
- Може ли изискването да бъде *преформулирано*, за да бъдат тестовите случаи по-ясно дефинирани?

Проектиране и попълване на формуляр на тест за всяко изискване (Test record form)

- **Идентификатор** на изискването
- **Свързани изисквания**
- **Описание на теста**
 - Кратко описание на теста и обяснение защо е обективен за даденото изискване; описание на входната и изходна информация на системата.
- **Проблеми**
 - Описание на проблемите, които правят теста труден или невъзможен.
- **Коментари** и препоръки
 - Съвети за евентуални решения на откритите проблеми.

Пример: Формуляр на тест за изискване 10.iv на EDDIS

10.(iv) When users access EDDIS, they will be presented with web pages and all the services available for them

Requirements tested: 10.(iv)

Related requirements: 10.(i), 10.(ii), 10.(iii), 10.(vi), 10.(vii)

Test applied: For each class of user, prepare a login script and identify the services expected for that class of user.

The results of the login should be a web page with a menu of available services.

Requirements problems: We don't know the different classes of EDDIS user and the services which are available to each user class. Apart from the administrator, are all other EDDIS users in the same class?

Recommendations: Explicitly list all user classes and the services which they can access.

Трудности при проектиране на тест за някои типове изисквания

- **Изисквания за системата**

- Това са изисквания, прилагани за системата като *цяло*. Те са най-трудните за валидиране независимо от използвания метод. Тестове, които *не се изпълняват*, не могат да тестват нефункционални, общи системни характеристики като например използваемост.

- **Изключения**

- Изисквания, които *изключват* специфично поведение. Например: изискването формулира, че системните повреди *никога не трябва* да допускат повреда на базата данни. Такова изискване *не би могло да бъде тествано изчерпателно*.

- **Някои нефункционални изисквания**

- Някои нефункционални изисквания като надеждност могат да бъдат тествани само с *голям брой тестове*.