Systems Engineering

l'ingegneria dei sistemi comprende:
- HW
- SW
- persone

=> e l'ingegneria del sw è un sottoramo

- Designing, implementing, deploying and operating systems which include hardware, software and people

What is a system?

- A purposeful collection of inter-related components working together towards some common objective.
- A system may include software, mechanical, electrical and electronic hardware and be operated by people.
- System components are dependent on other system components
- The properties and behaviour of system components are inextricably inter-mingled

sono dipendenti perché devono lavorare tutti insieme per far funzionare il sistema

Problems of systems engineering

- Large systems are usually designed to solve complex problems
- Systems engineering requires a great deal of co-ordination **across disciplines**
  - Almost infinite possibilities for design trade-offs across components
  - Mutual distrust and lack of understanding across engineering disciplines
- Systems must be designed to last many years in a changing environment

Software and systems engineering

- The proportion of software in systems is increasing. Software-driven general purpose electronics is replacing special-purpose systems
- Problems of systems engineering are similar to problems of software engineering
- Software is (unfortunately) seen as a problem in systems engineering. Many large system projects have been delayed because of software problems
- *Spesso viene richiesto al software di 'compensare' la 'rigidità' dell'hardware!*

Il SW non è l'unico elemento di un Sistema!!

- E' quindi necessario inquadrare l'attività di sviluppo del SW nell'ambito più vasto e generale del processo di ingegnerizzazione del Sistema in cui il SW sarà inserito.
- Bisogna quindi considerare lo scopo del Sistema, i requisiti (di business) cui va incontro, i ruoli delle varie componenti (varie tipologie di dispositivi HW, le persone e i processi coinvolti, altre procedure a DB coinvolti, …)
- Concentrarsi solo e immediatamente sul SW senza avere una vision più generale 'di Sistema' non è corretto!

proprietà che non vengono fuori guardando i singoli pezzi/sottosistemi/moduli ma nell'insieme complessivo (es. della bici, quando monto tutti i pezzi posso effettivamente vedere se funziona)

Emergent properties (proprietà *complessive*)

- Integrare la componenti di un sistema per costituire il sistema stesso dà origine a proprietà che non si possono utilizzare per caraterizzare le single componenti
- Properties of the system as a whole rather than properties that can be directly derived from the properties of components of a system
- Emergent properties are a consequence of the relationships between system components
- They can therefore only be assessed and measured once the components have been integrated into a system

Examples of emergent properties

- *The overall weight of the system*
  - This is an example of an emergent property that can be computed from individual component properties.
- *The reliability (affidabilità) of the system*
  - This depends on the reliability of system components and the relationships between the components.
- *The usability of a system*
  - This is a complex property which is not simply dependent on the system hardware and software but also depends on the system operators and the environment where it is used.

Types of emergent properties

- **Functional** properties
  - These appear when all the parts of a system work together to achieve some objective. They relate to the input/output relation of the system. For example, a bicycle has the functional property of being a transportation device once it has been assembled from its components. In SW, they relate to the input/output data relation of the system.
- **Non-functional emergent** properties
  - Examples are reliability, performance, safety (sicurezza, evitare comportamenti pericolosi, …) , and security (protezione, evitare accessi non autorizzati, …). These relate to the behaviour of the system in its operational environment (i.e., how the system is operating/functioning). They are often critical for computer-based systems as failure to achieve some minimal defined level in these properties may make the system unusable.

The 'shall-not' properties

- Properties such as performance and reliability can be measured
- However, some properties are properties that the system should not exhibit
  - Safety (sicurezza) - the system should not behave in an unsafe way
  - Security (protezione)- the system should not permit unauthorised use
- Measuring or assessing these properties is very hard

Human and organisational factors

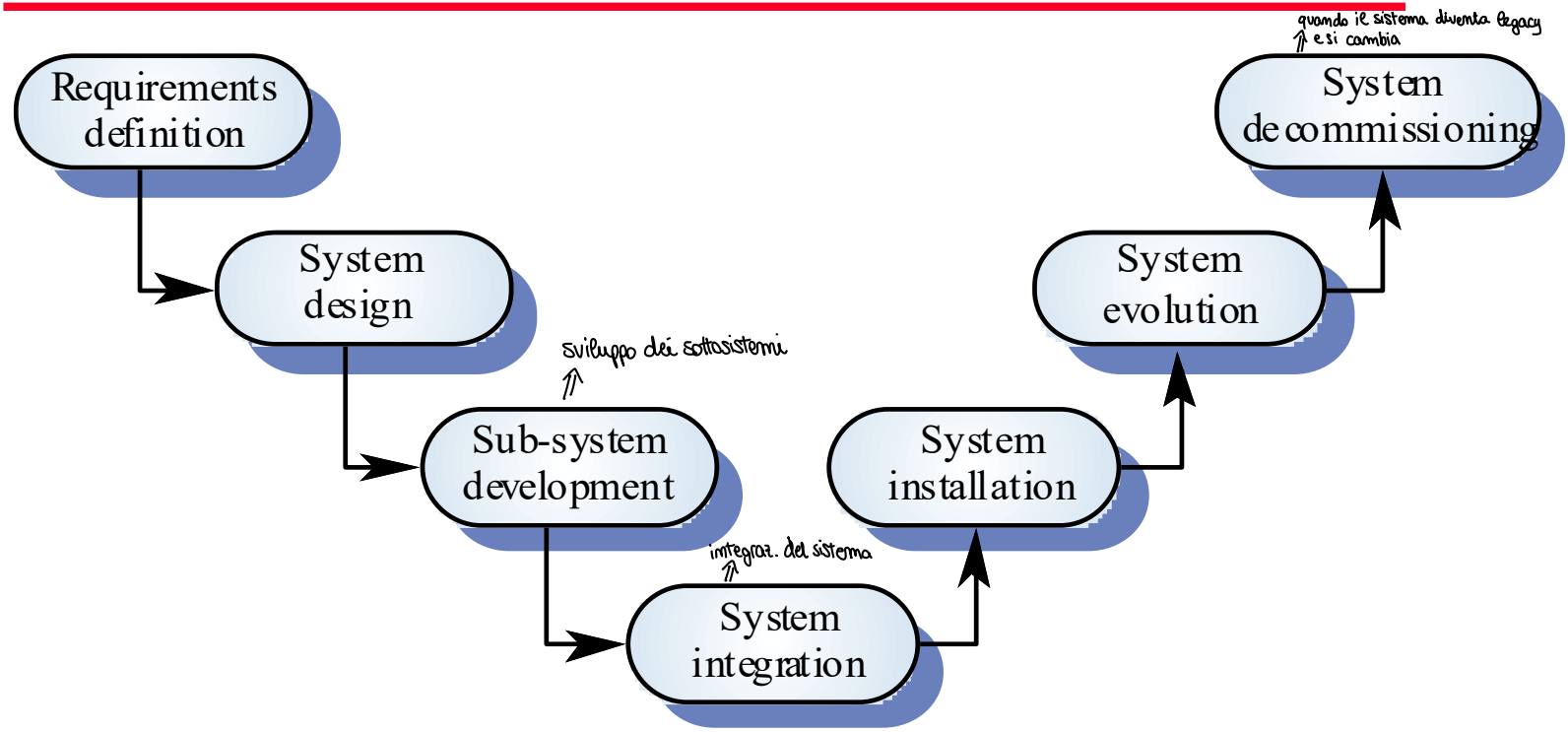
Fattori che rappresentano criticità da affrontare:

- *Process changes*
  - Does the system require changes to the work processes in the environment?
- *Job changes*
  - Does the system de-skill the users in an environment or cause them to change the way they work?
- *Organisational changes*
  - Does the system change the political power structure in an organisation?

The system engineering process

- Le fasi del processo per lo sviluppo sono le stesse che a livello più specifico vengono utilizzate per il SW
- Usually follows a 'waterfall' model because of the need for parallel development of different parts of the system
  - Little scope for iteration between phases because hardware changes are very expensive. Software may have to compensate for hardware problems
- Inevitably involves engineers from different disciplines who must work together
  - Much scope for misunderstanding here. Different disciplines use a different vocabulary and much negotiation is required. Engineers may have personal agendas to fulfil

The system engineering process



System evolution

- Large systems have a long lifetime. They must evolve to meet changing requirements
- Evolution is inherently costly
  - Changes must be analysed from a technical and business perspective
  - Sub-systems interact so unanticipated problems can arise
  - There is rarely a rationale for original design decisions
  - System structure is corrupted as changes are made to it
- Existing systems which must be maintained are called **legacy systems**

Legacy Systems

Def.:

I sistemi legacy (o sistemi ereditati) sono sistemi informatici socio-tecnici sviluppati in passato, impiegando tecnologie che col tempo sono diventate vecchie o obsolete; questi sistemi non includono solo hardware e software, ma anche procedure, lavorazioni, vecchi modi di fare, cose difficili da cambiare perchè ereditate, e modificare una parte di questi sistemi richiede inevitabilmente la modifica di altri Componenti.

Legacy Systems: i problemi

1. Componenti non più supportati dai produttori
2. Scarse conoscenze, mancanza di documentazione
3. Scarsità o mancanza di competenze sul mercato o in azienda
4. Scarsa espandibilità, per i motivi precedenti
5. MTTR – Mean Time To Repair* troppo alto
6. Scalabilità scarsa, ossia difficoltà di elaborare masse più elevate di dati ed altre tipologie di dati
7. Costi elevati di gestione e manutenzione

(*) tempo medio di riparazione, `e il valore atteso del tempo di ripristino a seguito di un guasto

Legacy Systems: perché rimangono?

1. Costi elevati per sostituirli

2. Operatività: non è accettabile un fermo del vecchio sistema

La valutazione corretta dovrebbe includere anche:

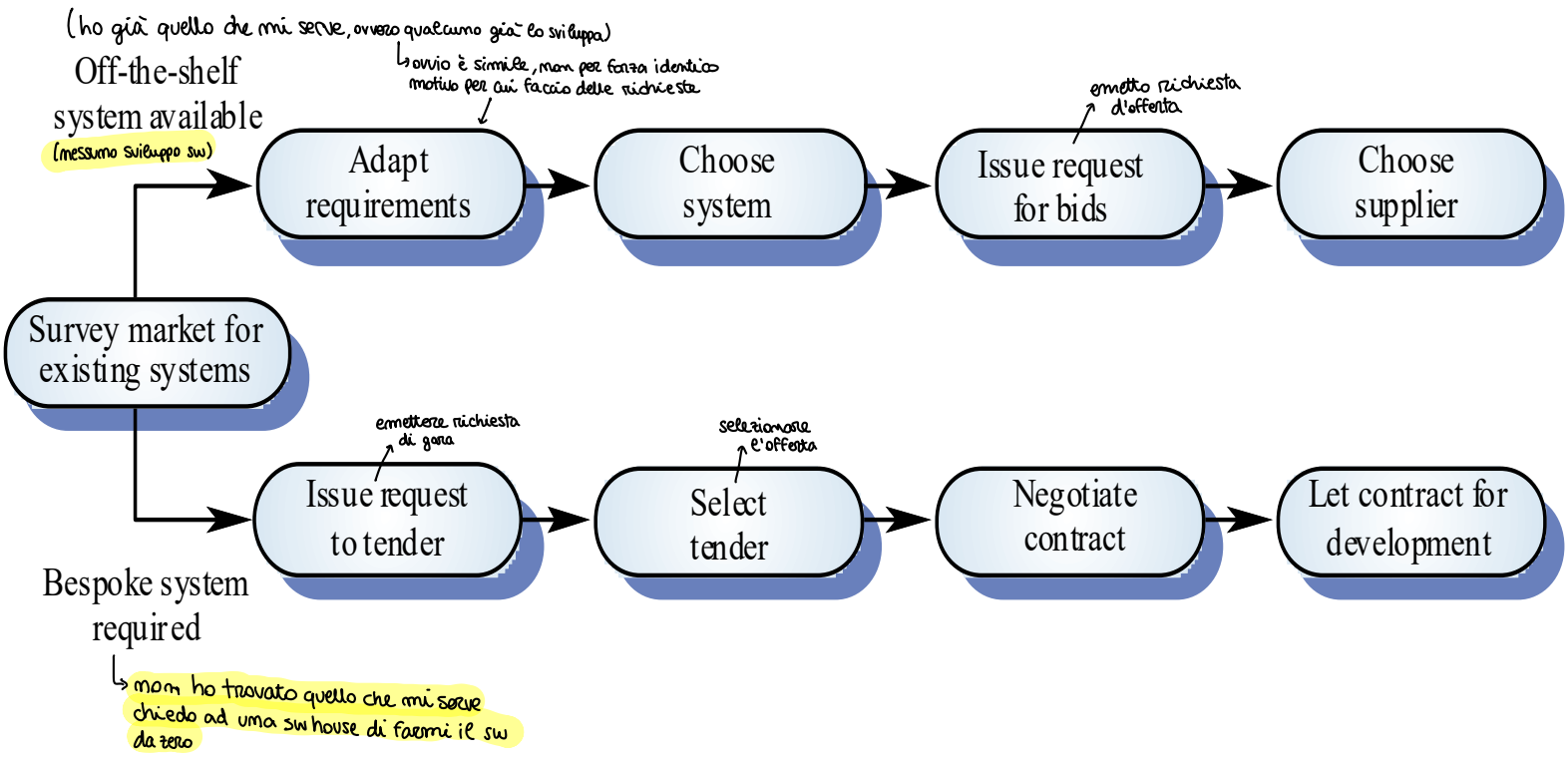
- Tutti i costi che il legacy provoca
- I vantaggi dell'investimento che è necessario per sostituire il legacy

System procurement

- Acquiring a system for an organization to meet some need
- Some system specification and architectural design is usually necessary before procurement
  - You need a specification to let a contract for system development
  - The specification may allow you to buy a commercial off-the-shelf (COTS) system. Almost always cheaper than developing a system from scratch

2° modello dei PROCESSI

The system procurement process



nel caso di prima sarà la SW house che ci offre il suo servizio

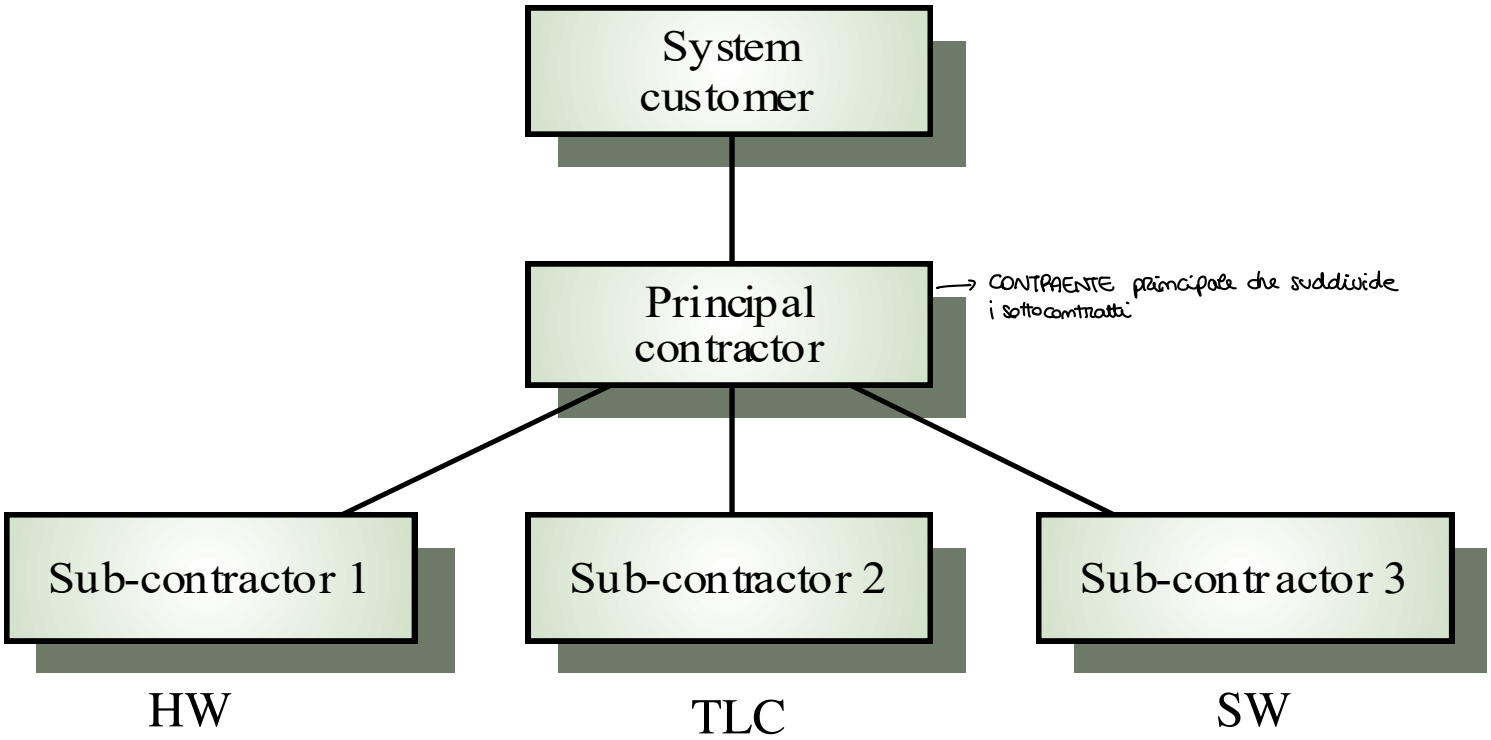
CONTRATTO

=> se il progetto SW è grande è possibile che il CONTRAENTE (SW house con cui ho fatto il contratto) decide di dare a sua volta un sub-contratto ad altre aziende di lavoro per suddividere meglio il carico => ovviamente il cliente deve esserne al corrente ed essere d'accordo

Contractors and sub-contractors

- The procurement of large hardware/software systems is usually based around some principal contractor
- Sub-contracts are issued to other suppliers to supply parts of the system
- Customer liases with the principal contractor and does not deal directly with sub-contractors

Contractor/Sub-contractor model



Key points

- System engineering involves input from a range of disciplines
- Emergent properties are properties that are characteristic of the system as a whole and not its component parts
- System architectural models show major sub-systems and inter-connections. They are usually described using block diagrams

Key points

- System component types are sensor, actuator, computation, co-ordination, communication and interface
- The systems engineering process is usually a waterfall model and includes specification, design, development and integration.
- System procurement is concerned with deciding which system to buy and who to buy it from

Conclusion

- Systems engineering is hard! There will never be an easy answer to the problems of complex system development
- Software engineers do not have all the answers but may be better at taking a systems viewpoint
- Disciplines need to recognise each others strengths and actively rather than reluctantly cooperate in the systems engineering process

Set 2 - Cosa ricordare: concetti, motivazioni, conseguenze, relazioni fra concetti, ecc.

- Concetto generale di Sistema. I sistemi che ci interessano includono HW SW e aspetti organizzativi (persone, processi).
- Problematiche dell'ingegneria dei sistemi e relazioni con l'IS.
- Proprietà emergenti. P. funzionale e p. non funzionali. Shall-not properties.
- Cambiamento dei fattori umani/organizzativi e loro importanza.
- Processo di systems engineering e fasi principali.
- Evoluzione e legacy
- System Procurement: due possibilità, modello dei processi
- Modello Contractor/subcontractor