Queen Mary University of London

EBU5608 Product Development and Management

Topic 9 – System Level Design

Queen Mary
University of London

Agenda

- Aim of Phase 2 System-level Design
- Product architecture – what is it?
- Modular and integral architecture
- Implications of product architecture
- Role of the product architecture team
- **4-step method** for establishing the product architecture
- Key outcomes

Queen Mary University of London

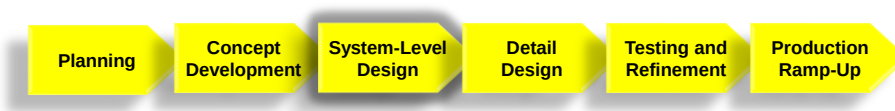
The product development process

- Product development **starts** with planning and **concludes** with product launch
- A generic product development process can be used as an example
- The process has six distinct phases
  - 0. **Planning**
  - 1. **Concept development**
  - 2. **System-level design**
  - 3. **Detail design**
  - 4. **Testing and refinement**
  - 5. **Production ramp-up**

Queen Mary
University of London

Phase 2 – System-level design

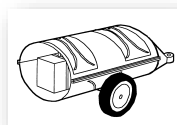
- During phase 2, the development of the **architecture** of the product continues from phase 1 - concept development
  - i.e. the **product specifications** identified in the Contract Book continue to become more **defined**
- Phase 2 includes the definition of the **product architecture** and the **decomposition** into functional & physical **elements**



Queen Mary University of London

Phase 2 – System-level design

- This definition and decomposition can be clearer for **physical** products, but still relevant for **software** etc
  - Think about the use of classes, subsystems and interfaces in **Java** program development, for example



Java

Queen Mary
University of London

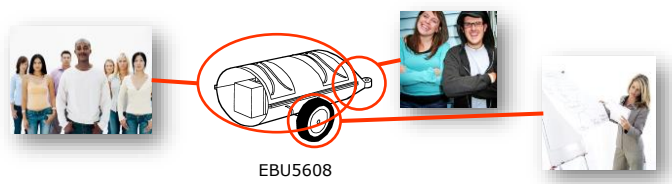
Product Architecture

- **Product Architecture** can be described as:
  - "the **scheme** by which the **functional elements** of the product are **arranged** into **physical chunks** and by which the **chunks interact**"
  - **Functional** – individual **operations** and **transformations** that contribute to its **overall** performance
  - **Physical** – **parts**, **components** and **subassemblies** that implement the product's functions

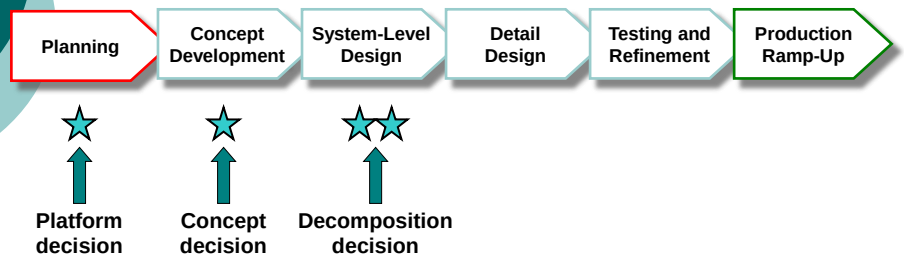
Queen Mary University of London

Product Architecture's Aim

- Architectural decisions allow the **detailed design** and **testing** of these physical blocks to be assigned to **teams**, individuals, and/or suppliers, so that the development of different portions can be carried out **simultaneously**



…in a Product Development Process

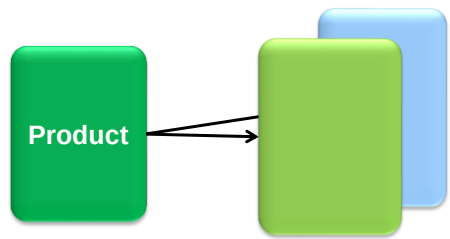


Product architecture is determined early in the development process.

Queen Mary University of London

Product Architecture - Chunks

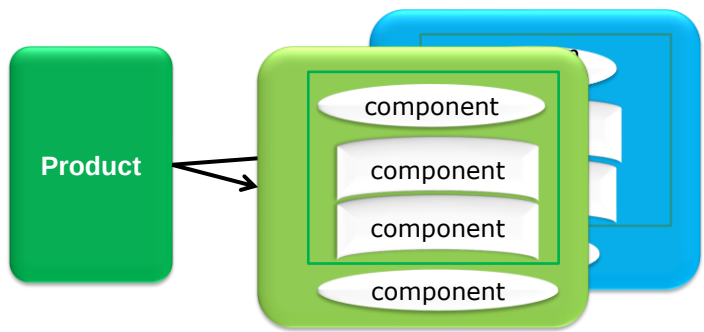
- The physical elements of a product are typically organised into several major building blocks – called **chunks**
- These chunks become the building blocks for the product or **family** of products



Queen Mary
University of London

Product Architecture - Chunks

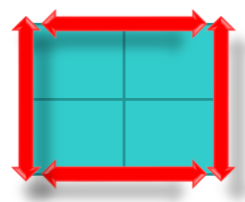
- Each chunk is then made up of a collection of **components** which implement the **functions** of the product



Queen Mary University of London

Product Architecture - Chunks

- The **architecture** of a product is therefore the **scheme** by which the **functional elements of the product are**
  - **arranged** into **physical chunks** and
  - by which the **chunks interact**



Queen Mary University of London

Mentimeter

Modular or Integral Architecture?

iPhone

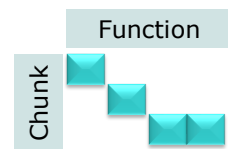
Rollerblade
In-Line Skates

Tesla
car

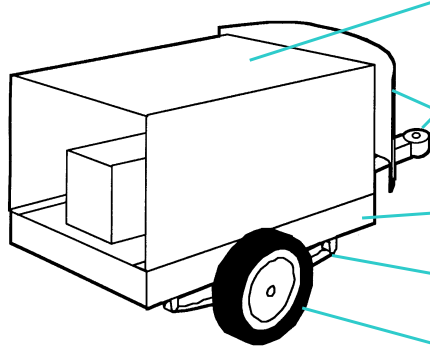
Queen Mary University of London

Modular & integral architecture

- A **key characteristic** of product architecture is the degree to which it is either:
  - **Modular**, in which
    - Each chunk implements **one** or a **few functions entirely.**
    - The **interactions** between chunks are **well defined.**
    - Modular architecture has **simplicity** and **reusability** for a product family or platform.



Trailer example: Modular Architecture

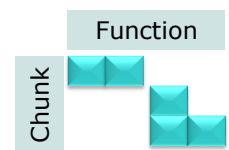


box	protect cargo from weather
tow bar	connect to vehicle
fairing	minimize air drag
bed	support cargo loads
springs	suspend trailer structure
wheels	transfer loads to road

Queen Mary University of London

Modular & integral architecture

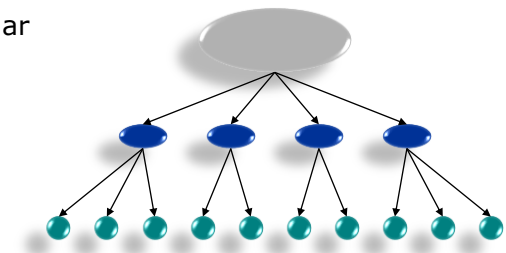
- **Integral**
  - **Functional elements** are implemented by **multiple** chunks, or a chunk may implement many functions
  - The **interactions** between chunks are **poorly defined**
  - Integral architecture generally increases performance and reduces costs for any specific product model.



Queen Mary
University of London

Concepts and levels

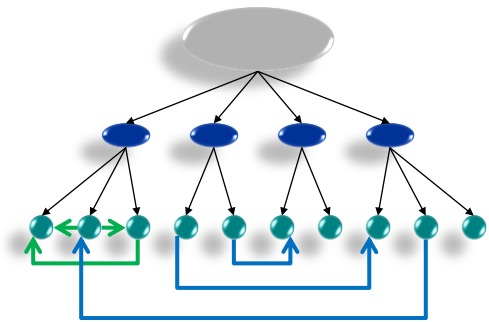
- The concepts of integral and modular apply at several levels:
  - System
  - Sub-system
  - Component



Queen Mary University of London

Decomposition & interactions

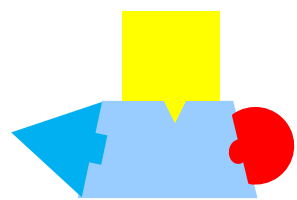
- Product Architecture is decomposition + interactions
- Decomposition
- Interactions
  - within chunks
  - across chunks



Queen Mary
University of London

Modular architecture

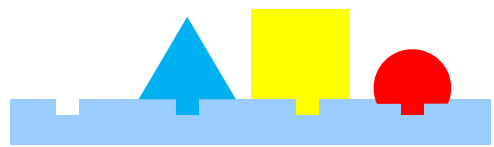
- There are three main types of modular architecture:
  1. **Slot-modular** (the most common type)
     - Each of the **interfaces** between chunks in a slot-modular architecture is of a **different type** from the others – therefore the various **chunks** in the product **cannot** be interchanged



Queen Mary University of London

Modular architecture

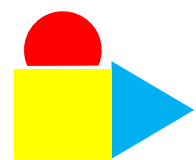
- There are three types of modular architecture:
  2. **Bus-modular**
     - There is a **common bus** to which the other chunks connect via the **same type** of interface



Queen Mary
University of London

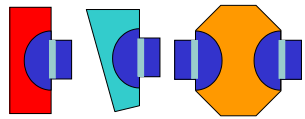
Modular architecture

- There are three types of modular architecture:
  3. **Sectional-modular**
     - All **interfaces** are of the **same type**, there is **no single element** to which all the other chunks **attach**
     - The assembly is built up by connecting the chunks to **each other** via identical interfaces

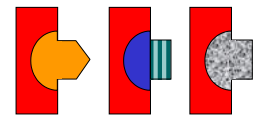


Queen Mary University of London

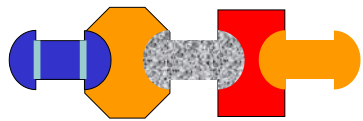
Other Types of Modularity



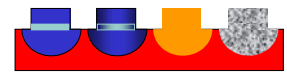
Slot Modularity



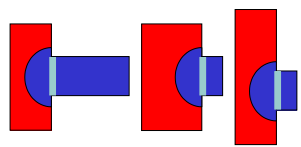
Sharing Modularity



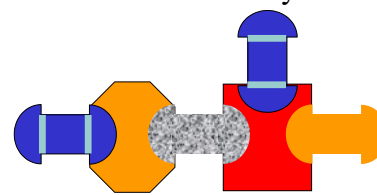
Sectional Modularity



Bus Modularity



Fabricate-to-Fit Modularity

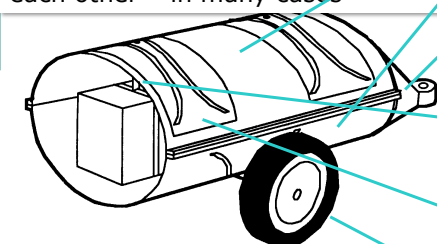


Mix Modularity

Adapted from K. Ulrich, "The Role of Product Architecture in the Manufacturing Firm", *Research Policy*, 1995.

Trailer example: Integral Architecture

Here we show a similar example to the modular architecture trailer, but the parts are now integral to each other – in many cases



upper half
lower half
nose piece
cargo hanging straps
strap slot covers
wheels

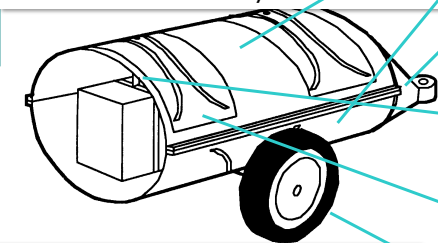
protect cargo from weather
connect to vehicle
minimize air drag
support cargo loads
provide trailer structure
transfer loads to road

One chunk can perform **many functions** – and **one function** can need **many chunks**

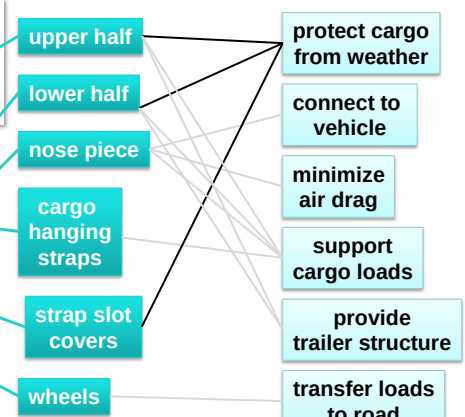
Queen Mary University of London

Trailer example: Integral Architecture

Here we show a similar example to the modular architecture trailer, but the parts are now integral to each other – in many cases

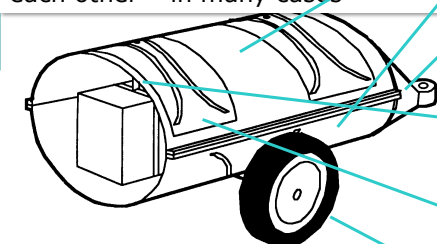


One chunk can perform **many functions** – and **one function** can need **many chunks**



Trailer example: Integral Architecture

Here we show a similar example to the modular architecture trailer, but the parts are now integral to each other – in many cases



upper half
lower half
nose piece
cargo hanging straps
strap slot covers
wheels

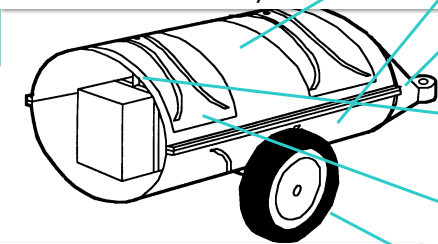
protect cargo from weather
connect to vehicle
minimize air drag
support cargo loads
provide trailer structure
transfer loads to road

One chunk can perform **many functions** – and **one function** can need **many chunks**

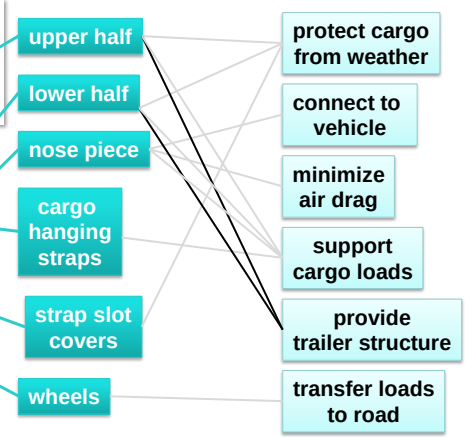
Queen Mary University of London

Trailer example: Integral Architecture

Here we show a similar example to the modular architecture trailer, but the parts are now integral to each other – in many cases

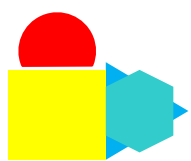


One chunk can perform **many functions** – and **one function** can need **many chunks**



Implications of product architecture

- There are **implications** to the decisions you make about product architecture
- **Product change**
  - **Modular** chunks allow **changes** to be made to a **few isolated** functional elements of the product without necessarily affecting the design of **other chunks**
  - **Motives** for change include – upgrades, add-ons, adaptation, wear, consumption, flexibility in use, reuse



Queen Mary University of London

Implications of product architecture (cont.)

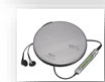



- **Product variety**
  - Variety refers to the **range** of product models the firm can produce within a particular **time** period in response to market **demand**
  - Products built around **modular** product architectures can be **more easily varied** without adding tremendous complexity to the **manufacturing** system
    - e.g. mobile phone handset design, portable audio

Queen Mary University of London

Implications of product architecture – product variety example

- Sony is a major consumer and commercial electronics manufacturer
- Sony sells **54 models** of portable audio equipment in the UK, including
  - MP3
  - Personal CD, minidisc & tape
  - Portable CD+radio
  - Portable radio
- Uses **modular architecture** and **flexible manufacturing** techniques to
  - Introduce new **models**
  - Meet changes in **demand**



Queen Mary University of London

Implications of product architecture (cont.)

- **Component standardisation**
  - The use of the **same** component or chunk in **multiple** products
  - If a chunk implements only one or a few **widely useful** functional elements, then the chunk can be **standardised** and used in several **different** products
  - Example – standard batteries

Implications of product architecture (cont.)

- **Product performance**
  - How **well** a product implements its intended **functions**
  - Product performance **characteristics** include speed, efficiency, life, accuracy and noise
- **Manufacturability**
  - The product architecture also directly affects the ability of the team to design each chunk to be **produced** at **low cost**

Queen Mary University of London

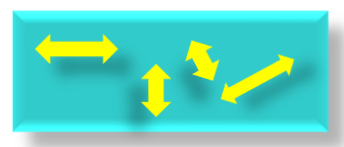
Implications of product architecture (cont.)

Mentimeter

- **Product development management**

Responsibility for the detail design of each chunk is usually assigned to a relatively **small group** within the **firm** or to an outside **supplier**

Chunks are assigned to a single individual or group because their design requires careful resolution of **interactions** among components within the chunk



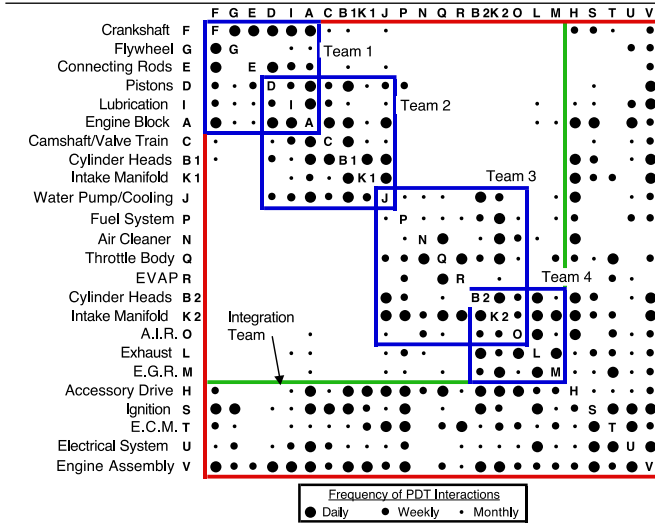
Queen Mary University of London

Role of Product Architecture Team

- The architecture should be established by a **cross-functional** effort by the development team as it will have implications for
  - subsequent product **development** activities, and
  - the **manufacturing** and **marketing** of the completed product

Queen Mary University of London

System Team Assignment Based on Product Architecture



From "Innovation at the Speed of Information", S. Eppinger, *HBR*, January 2001.

Queen Mary
University of London

Summary

- The **architecture** of a product is therefore the **scheme** by which the **functional elements of the product** are arranged into **physical chunks** and by which the **chunks interact.**
- Two main types of architecture
  - **Modular** (slot, bus, sectional) in which each chunk implements **one** or a **few functions entirely** and the **interactions** between chunks are **well defined.**
  - **Integral**, in which **functional elements** are implemented by **multiple** chunks, or a chunk may implement many functions, and the **interactions** between chunks are **poorly defined.**
- Implications of product architecture: product **change**, **variety**, **performance**, **manufacturability**, **components standardisation**, **product development management**

Queen Mary University of London

The 4 step method

- Ulrich and Eppinger recommend a **4-step method** for establishing the product architecture
  - Step 1 – Create a **schematic** of the product
  - Step 2 – **Cluster** the elements of the schematic
  - Step 3 - Create a rough **geometric layout**
  - Step 4 – Identify the fundamental and incidental **interactions**

Product Architecture Example: Hewlett-Packard DeskJet Printer

Queen Mary University of London

Step 1 – Create a schematic of the product

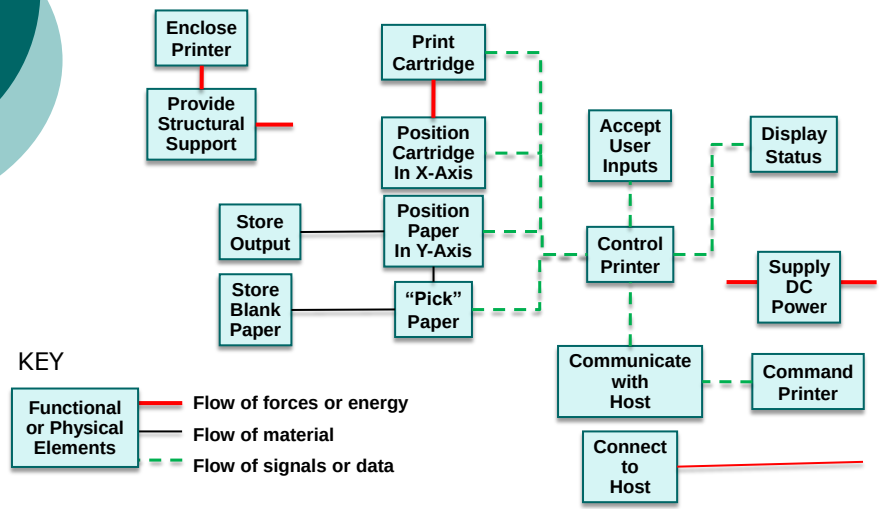
- A schematic is a diagram representing the team's understanding of the **component elements** of the product
  - An example can be seen in a later slide
- At the **end** of **Phase 1** – Concept Development
  - some of the **elements** in the schematic are **physical** concepts
  - some correspond to **critical components**
  - but some are still only described **functionally**

Step 1 – Create a schematic of the product

- The schematic should reflect the **best understanding** of the state of the product, it does not have to contain every detail
- More **detailed** functional elements are finalised **later** in Phase 2
- A good **guideline** is for there to be **no more than 30 elements** in the schematic
- Usually **more than one** alternative schematic is developed and the team spend time **selecting** the most appropriate

Queen Mary University of London

Architecture example - DeskJet Printer Schematic



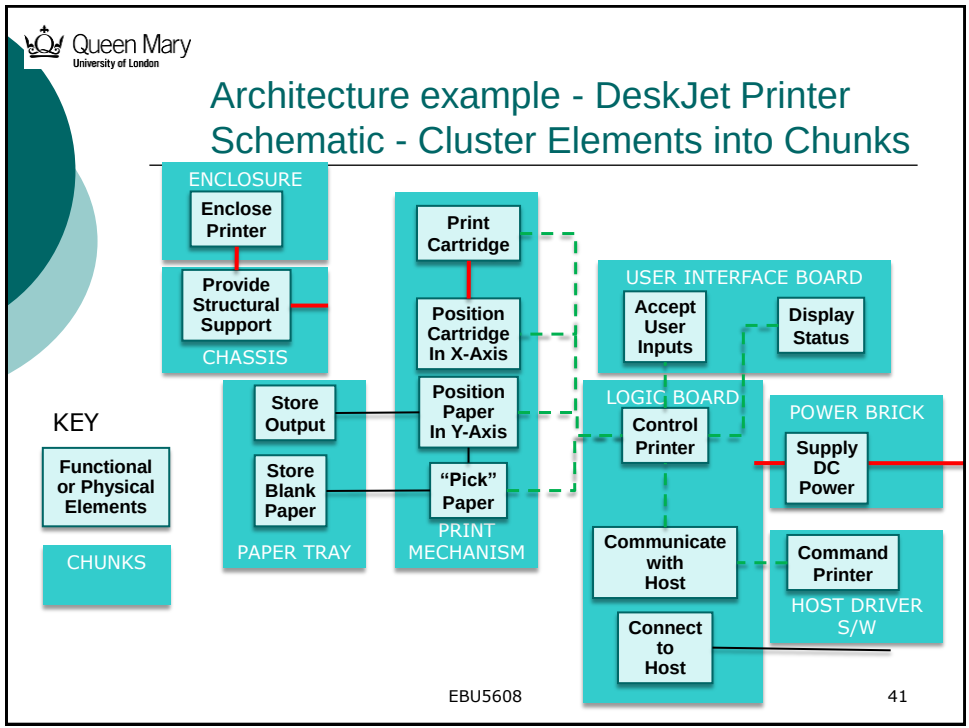
Queen Mary University of London

Step 2 – Cluster the elements of the schematic

- This step requires **assigning** and/or **clustering** each of the **elements** of the schematic to a **chunk**
- An **example** of this clustering can be seen in the next slide

Queen Mary University of London

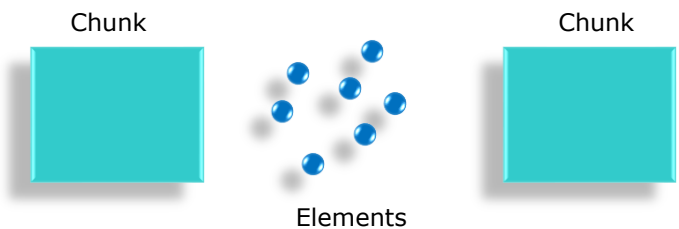
Architecture example - DeskJet Printer Schematic - Cluster Elements into Chunks



Queen Mary University of London

Setting the clusters

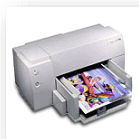
- The clustering role is undertaken by the **team** to ensure that the right decisions are taken in terms of **functionality split**



Queen Mary University of London

Setting the clusters

- When deciding where to set the clusters, the following factors should be considered:
- **Geometric integration and precision**
  - Assigning elements to the **same** chunk allows a **single** individual or group to control the physical **relationships** among the elements
  - Elements requiring **precise location** or **close geometric integration** can often be best designed if they are part of the **same** chunk
    - e.g. paper handling in a printer



Queen Mary
University of London

Setting the clusters

- **Function sharing**
  - When a **single** physical **component** can implement **several functional** elements of the product, these functional elements are best clustered **together**
  - For example, an integrated control panel on a car

Queen Mary University of London

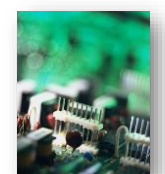
Setting the clusters

- **Capabilities of vendors**
  - A **trusted vendor** may have specific capabilities related to a project
  - To best take advantage of such capabilities a team may choose to cluster those elements about which the vendor has **expertise** into **one chunk**

Queen Mary
University of London

Setting the clusters

- **Similarity of design or production technology**
  - When **two or more** functional elements are likely to be implemented using the **same design** and/or **production technology**, then incorporating these elements into the **same chunk** may allow for more economical design and/or production
  - A common strategy, for example, is to **combine** all functions that are likely to involve **electronics** in the **same chunk**. This allows the possibility of implementing all of these functions within a **single circuit board**



Queen Mary University of London

Setting the clusters

- **Localisation of change**
  - When a team expects there to be a **great deal of change** in an element, it makes sense to **isolate** that element into its own modular chunk
  - In that way, any necessary changes to the element can be carried out without **disrupting** any of the other chunks

Queen Mary
University of London

Setting the clusters

- **Accommodating variety**
  - Elements should be clustered together to enable the firm to **vary** the product in ways that will have value for **customers**
  - Example – a power supply needs to handle different mains supplies in different countries

230V 50Hz

120V 60Hz

100V 60Hz

120V 50Hz

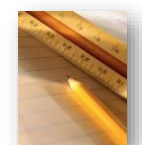
Queen Mary University of London

Setting the clusters

- **Enabling standardisation**
  - If a set of elements will be useful in **other products**, they should be clustered together into a single chunk
  - This allows the **physical elements** of the chunk to be produced in **higher quantities**
  - Example – cartridge in ink jet printer

Step 3 - Create a rough geometric layout

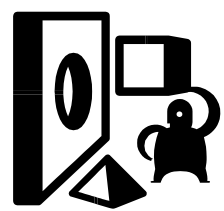
- A geometric layout can be created in **2 or 3 dimensions** or as **physical** models
  - an example can be found on a later slide
- Creating a geometric layout forces the team to consider
  - whether the **geometric interfaces** among the chunks are **feasible** and
  - to work out the basic **dimensional relationships** among the chunks



Queen Mary University of London

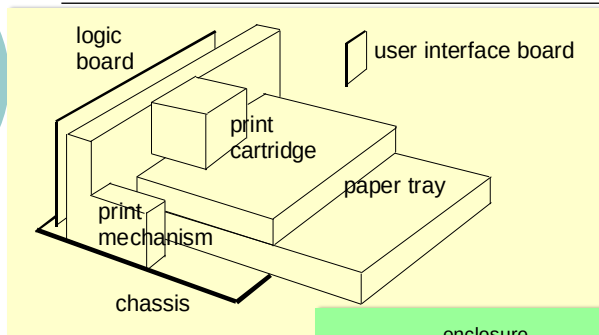
Step 3 - Create a rough geometric layout

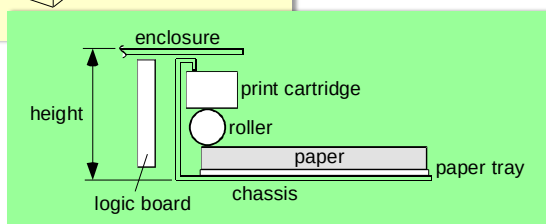
- In some cases, the team may discover that the clustering derived in step 2 is **not** geometrically feasible
- If this happens, some of the elements have to be **reassigned** to other chunks



Queen Mary
University of London

Geometric Layout Example





Queen Mary University of London

Step 4 - Identify the fundamental and incidental interactions

- It is most likely that a **different** person or group will be assigned to design **each chunk**
- Because the chunks interact with one another in both **planned** and **unintended** ways, these different groups will have to **coordinate** their activities and exchange information
- To **manage** this coordination process better, the team should **identify** the known **interactions** between chunks during the system-level design phase

Step 4 - Identify the fundamental and incidental interactions

- There are two categories of interaction
  - **Fundamental**
    - Those which correspond to the lines on the **schematic** that **connect** the chunks to one another
    - These are the **fundamental** interactions of the systems operation

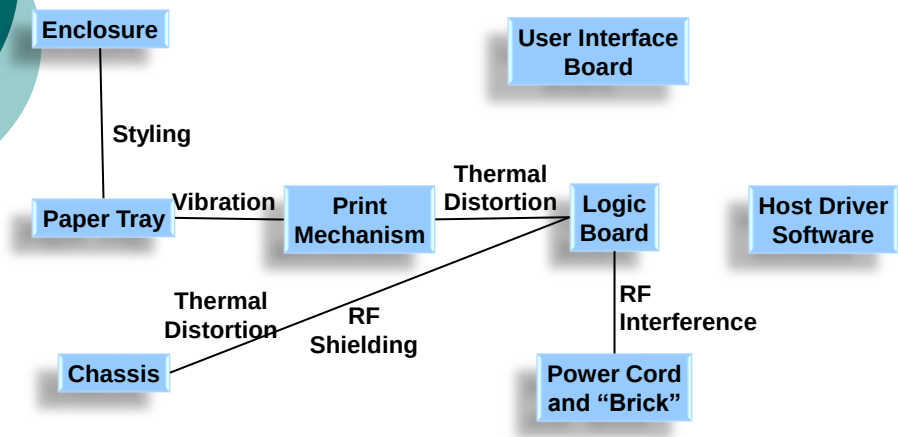
Queen Mary University of London

Step 4 - Identify the fundamental and incidental interactions

- **Incidental**
  - Those that arise because of
    - the particular physical implementation of functional elements, or
    - because of the geometric arrangement of the chunks
  - An **incidental interaction graph** is used to document this type of interaction, see next slide

Queen Mary University of London

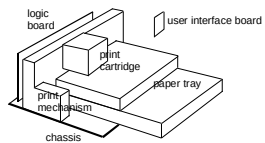
Incidental interactions example



Queen Mary University of London

Key Outcomes

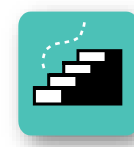
- The **end results** of this 4-step method for establishing the product architecture are:
  - an approximate **geometric layout** of the product
  - **descriptions** of the major chunks
  - documentation of the key **interactions** among the chunks



Queen Mary
University of London

Summary

- **4-step method** for establishing the product architecture
  - Step 1 – Create a **schematic** of the product
  - Step 2 – **Cluster** the elements of the schematic
  - Step 3 - Create a rough **geometric layout**
  - Step 4 – Identify the fundamental and incidental **interactions**



Queen Mary University of London

In-class Exercise (15 mins):

1. Name 3 different types of modular design in a PC.
2. From system-level design perspective, explain the 3 major differences between Apple keyboard design 1983 and that of 2024.

Queen Mary
University of London

Reading

- **Core Textbook** (Ulrich & Eppinger, 7th Edition)
  - Chapter 10. Product Architecture.

Queen Mary University of London

References

1. Product Design and Development, Karl T Ulrich and Steven D Eppinger, 7th Edition, 2020, McGraw-Hill, chapter 10.