In ST2, we can apply a style, and then must justify why that style applies. (must identify what problem exists to match it’s styles.)

**Architecture (Architectural style is in the design model)**

**Software Architecture (Know the definition** **Word for Word NB)**

The reason behind a decision may be small, but can have large ramifications.

**Why is it important?**

Provides a representation of for communication between al the stakehoulders.  
Models how the system is structed, and how the components work together

**This is important when choosing your design for st2**

**Organisation and Refinement**

Establish a set of design criteria to assess the architectural design (**ST2 Q, What are the considerations when making your design criteria**)

What are the things we need to consider before choosing an architectural style:  
  
-control  
-data

**Ask yourself**  
What is control, and how is it managed.

Control is: Power/influence – directing course of events. Control of the software and the manner in which it is built. How its executed. How it responds to inputs from data/other systems.

**Are you able to manage control? (Exam or ST2 test Q)**

Hierarchy: Which components have the most control  
Transfer: How does the system share control.  
Topology: how arranged  
Synchronised: is it in harmony.

**When asked for this Q, we have to say the 5 questions, then give answers to them (NB)**

**Data**

How data moves in the system, to make a complete system.

How is data communicated?   
Is the flow continuous?   
Mode of transfer?  
Data components  
Passive or active  
…

What need to be asked before  
What considered during the choosing:  
Don’t confuse the 2: one if before, one is during. (**More exam)**

**Architectural patters**

We wont be asked for a pattern in the **ST2  
-These are for specific problems?**

Patterns are used with styles to shape the overall system.

**Architectual Styles (NB slide after aladin image (ST2 Q))**

Components: **Not just take** the subsytems already there (in use case diagram), but also add ones that make sense eg Verification (does not just need to be part of the subsystem) (**NB)**  
Constraints: Where is the system going ot struggle  
Properties: What are the properties of the system

**Data-Centered Architecture**

Single data point, and components that connect  
Promotes integration :can intergrate with multiple components (different facets of the system)  
**When the system needs you to integerate multiple components together.**The components use the data center to perform their required operations.

-Components are independent

**Problem:**

Data stored is passive: Components often don’t know about changes.  
-Only when a component accesses the data center will it know a change has been made from another component (only if that other one has stored the new change in the data center).  
  
Therefore: when a system requires constant updates this system would be very bad.

**Single point** of failure/entry (backups fix this)  
-Not distributed

**Data Flow Architecture**

Batch or sequential architecture.  
Where input data is supposed to be transformed/manipulated in a series of connected components  
  
-Easily go from design to implementation fase  
Does not need to have a program – execution unpredictable  
Great when need Continuous flow of data.  
Bad: One component goes down, all goes down.  
-Components are not independent

**Main Program / Subprogram (probs wont use this)**

Some type of control hierarchy

Decomposes function into

Main program invokes sub program.  
  
A system where one subsystem creates another subsystem.  
  
Create component as data flows through system.

**In ST2, don’t just give the diagram, but give it with the components from the case study.**

**Object oriented design**

Applicable: when use case presents itself in object oriented manner.  
Objects can pass data to other objects so they can perform their functions. (similar to data flow)

Eg Login system passes to validation, passes to DB

(Subsytems can be broken down into multiple components)

Layered Architecture (We can go through this ourselves)

**Sirs advise  
How to answer:**

Asked to choose an architectural systle, and motivate why: I choose an object oriented, why its applicable, and examples. (Q3.2 2022)  
  
When asked diagram: 4 marks for drawing correctly, then describe each component and how it is working in the architecture that chosen, and how eg advantages, disadvatnages. Just discuss key points. (8mark Q) (Q3.3 2022)