



## **Fall 2020 Sessional Assignment : Software Design & Architecture**

**Presented to: Mam Aasma Khan**

**Presented by: Sajjad younas**

**Id : 13850**

**Date : 9/6/2020**

---

### **“Software Design And Architecture”**

**Q 1:- Choose an software architectural style of your choice and give its explanation that must cover the below given components of an architecture style:**

>Element/component

>Connector

>Constraints

>Attribute

**Ans:-Software Architecture:-**

Software architecture is an abstract representation, or model, of a software system in terms of a structure that consists of a collection of elements together with the relationships among them to achieve software design purposes and to manifest a certain set of design properties of the system.

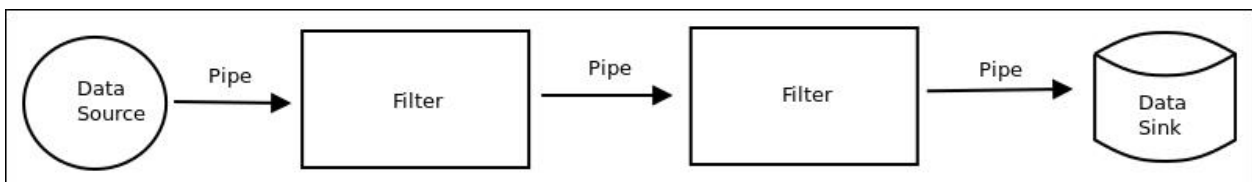
I select pipe and filter architecture style and their explanation is given below

### **Pipe and Filter architectures:-**

Pipe and Filter is a simple architectural style that connects a number of components that process a stream of data, each connected to the next component in the processing pipeline via a Pipe.

The Pipe and Filter architecture is inspired by the Unix technique of connecting the output of an application to the input of another via pipes on the shell.

The pipe and filter architecture consists of one or more data sources. The data source is connected to data filters via pipes. Filters process the data they receive, passing them to other filters in the pipeline. The final data is received at a **Data sink**.



Many systems are required to transform streams of discrete data items, from input to output. Many types of transformations occur repeatedly in practice, and so it is desirable to create these as independent, reusable parts, Filters.

The filter transforms or *filters* the data it receives via the pipes with which it is connected. A filter can have any number of input pipes and any number of output pipes.

The pipe is the connector that passes data from one filter to the next. It is a directional stream of data, that is usually implemented by a data buffer to store all data, until the next filter has time to process it.

The pump or producer is the data source. It can be a static text file, or a keyboard input device, continuously creating new data.

The sink or consumer is the data target. It can be a another file, a database, or a computer screen.

**Example:-**

- Unix programs. The output of one program can be linked to the input of another program.
- Compilers. The consecutive filters perform lexical analysis, parsing, semantic analysis, and code generation.

1)**Element and component** called filters, apply local transformations to their input streams and often do their computing incrementally so that output begins before all input is consumed.

A component is a unit of software that performs some function at run-time.

**Examples:** programs, objects, processes, clients, servers, databases

2)**Connector:-** called pipes, serve as conduits for the streams, transmitting outputs of one filter to inputs of another filter.

A connector is a mechanism that mediates communication, coordination, or cooperation among components. Connectors describe the interactions among components.

**Examples:** shared variable access, procedure calls, remote procedure calls

3)**Constraint:-** Constraints on composition of elements in this style dictate that pipes must connect output ports to input ports. Specializations of the pipe-and-filter style may also impose other constraints, such as that the architectural graph be acyclic or that the configuration define a linear sequence, or a pipeline.

4)**Attributes:-**

**>Advantages:**

Simplicity – Allows designer to understand overall input/output behavior of a system in terms of individual filters. Maintenance and reuse .Concurrent Execution –Each filter can be implemented as a separate task and be executed in parallel with other filters.

**>Disadvantages:**

Interactive transformations are difficult – Filters being independent entities designer has to think of each filter as providing a complete transformation of input data to output data. No filter cooperation. Performance – may force a lowest common denominator on data transmission-parse and unparsed latency.

**THE END**

\*\*\*\*\*