

Pipe-filter architectural patterns - useful approach for analytics based on daisy chain

A very simple, yet powerful architecture, that is also very robust. It consists of any number of components (filters) that transform or filter data, before passing it on via connectors (pipes) to other components. The filters are all working at the same time. The architecture is often used as a simple sequence, but it may also be used for very complex structures.

Gliffy Macro Error

You do not have permission to view this diagram.

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 source is also called as pump or producer. 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.

Reference use cases of Pipe-filter architectural pattern are compiler, DNA information analysis and etc.

Pipe-filter architectural pattern has following strong points

- Easy to add filters
- Has flexibility
- Easy to reuse filters
- Easy to recompose given filters for creating another pipeline.

Bad points of Pipe-filter architectural pattern will be a specific filter operational performance could cause poor efficiency, so overall performance could be worse. In worst case, it could cause deadlock state when the filter operation does not finished as planned.