Software Systems Information Topology Modeling

Brian Iovino

August 2019

# Purpose

To describe in basic technical terms how the systems are connected from an information flow (data flow) perspective.

# Elements

All act on information (data)

* External Sources
* External Sinks
* Internal Sinks (*see Processes below*)
* Internal Sources
* Persistent Relays
  + Example: FTP
* Transient Relays – one to one.
* Distributor – one to many.
* Processes
  + Union
  + Subtraction
  + Addition
  + Transformation
  + Utilization – Internal Sinks
* Stores
  + Queues
  + Message Bus -Topics
  + Persistent
    - DB
    - File System
  + Transient
    - Caches

# SRP Violations

If some elements are of multiple types, then those will be flagged for Architectural review.

# Edges – Wires

The means for transport of information has no definition within this context. It could be a wire, network, printout to letter to keyboard entry, etc.

A basic analogy is to think of a 1-meter long wire connected to a light, the length of the wire, gauge, stranded or solid doesn’t matter from an electron flow (information) or EMF (force behind the information flow, relays). It doesn’t have to be a wire, it could be energy in the form of UV that is converted to light. All this to say, the substrate is not important to the context of this discussion.

We care about is the production, flow, consumption, distribution, manipulation, direction (sense) and storage of information.

# Benefit

High level model can be easily maintained for overall software systems topology.

This will assist with the following:

Discussions, strategic overlay, simplification, cost savings, understanding the complexity, running simulations, understanding the complex configuration space and how the systems will respond within this complex space, capacity modeling, stability modeling, and proactive failure projection/sensing.

This form can be represented in UML and OO class definitions for import to Enterprise Architecture software, engineering platforms for further study and a more effective ability to make the best recommendations on the target platform and resource, project requirements.

Maintenance will be quick and easy. Load test models can help to spot overall effects of topological changes.

# Question

How is this represented in UML as an official topological description?

Spotting loops and redundancies. Most likely will have to overlay schema for redundancies. This is starting to break out of the context of the view methodology described in this document.

How to add event and time domain-based simulation – Matlab, EA tools, custom?