# CS 255 Model Application Short Paper

Daniel Loranger

Daniel.Loranger@SNHU.EDU

Southern New Hampshire University

## Process Model Application

A process model is a graphical representation of data workflows. Through general familiarity of typical LMS systems, some workflows are reasonably pre-defined unless there is a specific need to customize these. For example, accessing a course would first involve logging into the system, followed by locating the course in the catalog, then attempting to subscribe to the course. At this point a subprocess would trigger that checks for user access privileges and if insufficient may trigger a paid upgrade option to gain access. Having crossed that boundary, the subscription would be added and user could then access the course contents.

The customized workflows for this application would involve purchasing limited access resources (training time units), and then using these training time units to reserve physical resources (trainer and vehicle pair). Additionally, workflows would need to be created to handle accessing the DMV training materials, using the report generators, and dashboards. There are expected to be a lot of interwoven workflows, which are too numerous to detail in a written format. The key details to capture in these graphical representations will be data sources, sinks and datastores such as databases or local caches.

When capturing each process workflow, it is important to consider the applicable users that might use the tool and how their needs align with the workflow. These unique use cases may need to drive additional features, shortcuts, or other efficiency optimizations.

An example here would be the proxy user registrations. It would not make sense to have the proxy user be required to log-out and then back in as the end user for each change, rather there needs to be an alternate workflow that allows the proxy user to manipulate multiple user accounts while remaining logged in as themselves.

The interview promotes the need for additional tools that are completely distinct from the LMS and training system, but must however work with much of the same data sets. This is the reporting system that allows the leadership to review the courses taken by individuals and collate the data against the users test records to ensure their mission is being successful, as well as identify training effectiveness on a course-by-course basis.

## Object Model Application

An object model is a graphical representation of the objects (physical or software constructs). These graphical representations include the objects attributes, functions, and interconnectivity relationships including cardinality. A common version of this model is represented by a UML diagram.

A preliminary assessment would identify several types of objects that would include users (there are several more specific variants, but start with the generic), LMS data store, shopping cart, report generator, calendar handler, etc.

After a preliminary assessment, it would be prudent to go back to the process model and begin devolving each workflow to identify additional sources/sinks as well as datastores and work to map these to the existing objects, or define new objects as required.

Once the objects are well defined, the cardinality should be reviewed to understand the dependencies for what should exist and when. It is important to keep in mind both the creation relationships, as well as the destruction relationships, these may not be symmetrical. For example, a user is required to create an order, but the order can exist independent of the user once created (deleting a user shouldn’t destroy sales data).

Having the working model reasonably polished, the use cases should be reviewed to ensure that all object types have been accounted for and functionality defined for all object types and situations. For example, did we forget to account for the car maintenance technician being able to update the service records on the cars, or be able to take them out of the active objects while undergoing service which is outside of the 99% typical workflow models.

## 

## Process and Object Model Comparison

When comparing the two models, they perform interlinked, but distinctly different functions that must both be considered to gain a complete understanding.

The process model is used to establish the overall behavior and workflows of the systems which is essential to convey the behavior of the finished product. This model does not need to account primarily for the details of how things will work under the hood, but rather is concerned about showing the flow and look/feel of the system.

The object modeling is not useful for defining the user experience of the product, but is very helpful in defining how the various building blocks will work together to build the functionality of the product.

Without both of these models, you will have a difficult time conveying the complete product to be developed at a high level. This requires a reviewer to instead have to understand the entire architecture and implementation of the (code) system to provide feedback. This is impractical as many key stakeholders will not be technically oriented, or even just won’t have the time to deep dive and fully understand the product, and this can make changes happen late in the process costing time and expenses to make corrections that could have been caught in an earlier review.

In short, both models serve a unique and important role in the system development.