Open Source Structural Modeling (OSSM) Tools

Requirements Development Document

Joseph J. Simpson

09/23/18

Introduction:

John Warfield created structural modeling in the 1970's and refined the techniques over a period of years. As structural modeling developed, the techniques associated with Unified Program Planning (UPP), Interpretive Structural Modeling (ISM) and Interactive Management (IM) were designed, refined and tested. These three modeling techniques (UPP, ISM and IM) are distinct individual methods and approaches. However, over a period of time, the distinction between these three methods has been lost. Reestablishment of the clear distinctions between and among these three approaches is a key objective of the OSSM Tools development.

Interactive Management:

IM is the last of the three structural modeling techniques to be developed and includes aspects of UPP and ISM. IM uses an overall holistic analysis approach that provides structured operations and templates to encapsulate the UPP and ISM methods to produce the IM products and outcomes.

There are two basic types of computer support activities associated with IM tasks. The first type is a collection of user management, content management and content presentation tasks. The second type is the application of logical rules to the detailed evaluation of a specific IM task or phase. This document's long term goal is the development and communication of the requirements for both of these basic computer support activities. The logical rule support activity component will be developed first and the other computer support activities will be addressed as time and resources allow.

Interactive Management Phases:

IM is organized around three phases 1) planning phase, 2) workshop phase and 3) followup phase. Each individual phase addresses group activities necessary to increase group communication, interaction and task focus. In general, IM is grounded on the the managerial functions of intelligence, design and choice. Based on these managerial functions, IM tasks produce definitions, designs and a preferred design approach.

The planning phase creates a organized, structured collection of tasks and activities focused on producing the required results with the minimum expenditure of resources. There are many tasks and activities that can be accomplished using computer based data templates, communication processes and databases. However, the ISM logic based computer system does not appear to have a role in the planning phase.

The workshop phase is executed based on the plans and preparations made in the planning phase. Organizational resources are minimized during the workshop phase using the current plan to generate ideas, clarify and evaluate these ideas, structure the ideas and interpret the ideas and their related structure. The ISM logic based computer system is used in the workshop phase.

The followup phase of IM can take many different forms depending on the IM activity topics and objectives. Specific types of followup phases are, definition project followup, design alternatives followup and a combination of a definition and alternatives followups. The ISM logic based computer system does not appear to be used in the followup phases. However, many standard system engineering decision making and alternative analysis computer programs appear to have a strong role

in the followup phase.

General Open Source Structural Modeling (OSSM) Tools Creation Approach:
The initial evaluation of the mathematics of structure and associated computer based algorithms was completed by Systems Concepts in 2015 and the work archived in the following Github repositories:

https://github.com/jjs0sbw/INCOSE_2015
https://github.com/jjs0sbw/smp

The next version of the OSSM Tools is based on Vue.js. This version will be based on the mathematical logic in the existing Github repositories and the requirement to support idea creation, evaluation, analysis and selection. Individuals are identified and assigned their own specific accounts. The requirements associated with tracking individual contributions including ideas and voting are part of the next software version.

The OSSMTools software architecture is modular and designed to allow incremental design, development and deployment of any requested system function. Appendix 2, "GMU ISM Software" of "A Handbook of Interactive Management" is used to guide the initial operational and functional requirements. Three functional commands are available in the GMU ISM Software to support a specific structuring application. These three commands are: 1) DOMODEL, 2) DOCLUS, and 3) DOPRIOR. The logic and mathematics behind these commands is extensively evaluated in the referenced Github repositories. These evaluations are not repeated in this document.

A wide range of natural language structuring relationships are addressed by the GMU ISM commands. The authors developed the augmented model-exchange isomorphism (AMEI) to provide a standard framework for relationship categorization and evaluation. The AMEI is available at: https://www.researchgate.net/publication/272238246_Augmented_Model-Exchange_Isomorphism_Version_11

Other relationship characteristics and attributes were developed by the authors to support a more rigorous and detailed natural language relationship analysis and evaluation approach. The publications and documents associated with the relationship characteristics and attributes development are located on ResearchersGate at: https://www.researchgate.net/profile/Joseph_Simpson3/contributions

The DOMODEL command is presented as the most general command which may be applied in any ISM structuring application. The authors have not successfully created an open source software DOMODEL function that has the capability to consistently reproduce the same outcomes. The new open source software, produced from this requirements document, will focus on providing an acceptable open source DOMODEL command.

The DOCLUS command is used when the natural language relationship contains a symmetric logical property. The authors created an open-source DOCLUS software program that consistently reproduces the same outcome.

The DOPRIOR command is used when the natural language relationship contains an asymmetric logical property. The authors created an open-source DOPRIOR software program that consistently reproduces the same outcome.

The DOCLUS and DOPRIOR command logic may be combined to produce an application that

addresses the creation of an ordered sequence of clusters. The authors documented this types of open-source program creation and application in the material found on ResearchGate and Github.

The new open-source software architecture development process contains several concurrent activity threads. These concurrent activity threads are:

- Requirements development
- Software architecture design
- Software code production
- Software unit, integration and functional test

Requirements for the DOCLUS, DOPRIOR and DOMODEL are addressed in the following sections, in that order.

Open Source DOCLUS Command Requirements:
The DOCLUS command name and title was changed to "Cluster Objects Into Classes," in the open-source software available on Github. The initial screen for the software is shown in Figure 1.

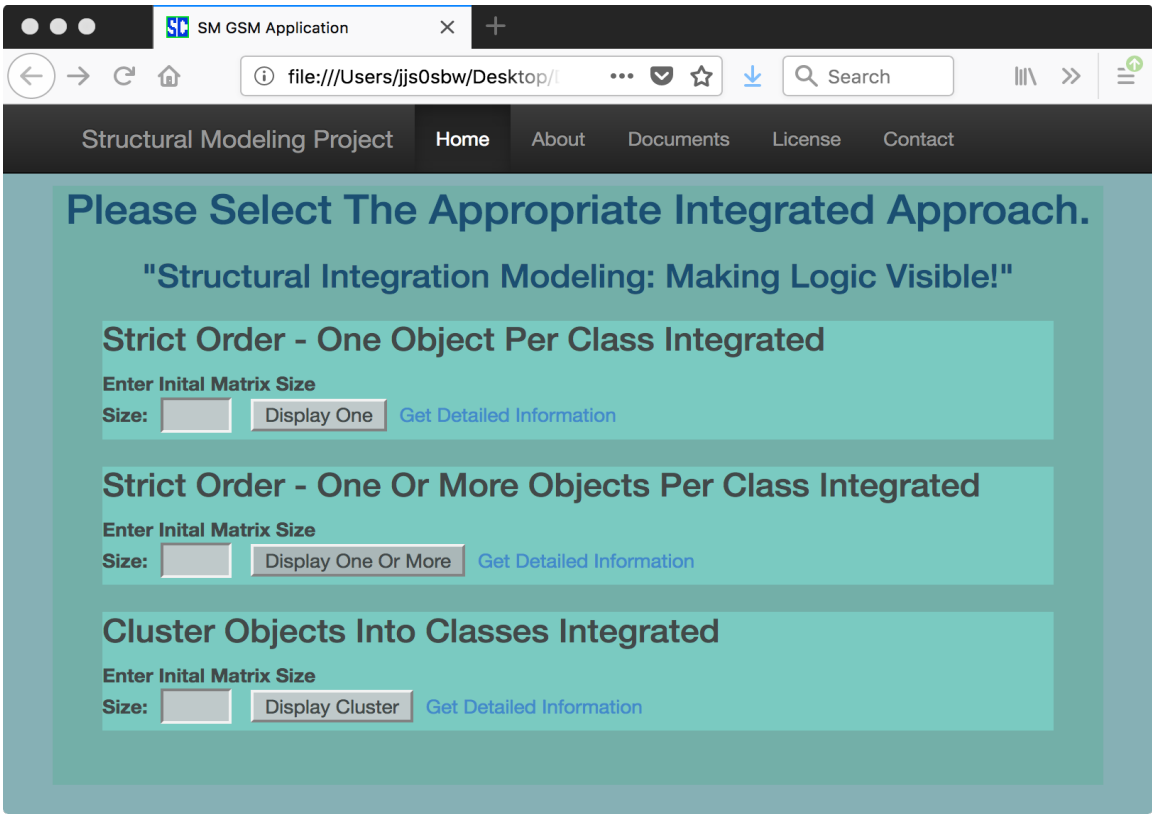

Figure 1, Existing Initial Screen

The OSSMTools initial screen will have similar content but the text boxes for entering the initial matrix size and the display buttons will be removed and replaced with a start button. Once the Cluster Objects Into Classes start button has been selected a new screen that allows the entry, editing and deletion of the objects of interest will be displayed. The information needed to proceed with this specific structuring process is associated with the naming and identification of each individual object of

interest. This collection of objects was developed previously, in Phase Two. Each object may have a large array of associated information. For instance, information about the person who first identified the object as an object of interest, the original reason for the object inclusion, comments about the object and other data may be available in OSSMTools database.

A system structuring event is data intensive and any software-based, system-structuring tool needs a database component. Sqlite3 has been chosen as the initial database component for the OSSMTools software suite. All information and all persistent system state data will be placed in the Sqlite3 database. IM Phase Two has six general processes shown in Table 1.

Process Name	Persistent Data	Uses Logical Software	Track Users
Idea Generation	Yes	No	Yes
Idea Clarification	Yes	No	Yes
Idea Adjustment	Yes	No	Yes
Idea Structuring	Yes	Yes	Yes (Votes)
Interpret Structures	Yes	No	Yes (Votes)
Adjust Structures	Yes	No	Yes (Votes)

Table 1 – Phase Two Structural Modeling Processes

As shown in Table 1, all six processes generate persistent data and only one process uses the structural modeling logical software. Each of the six processes will be explored next in more detail.

Ideawriting:
Based on the information and data collected in Phase One, the specific triggering question may be used in Phase Two to guide the generation of ideas using the Ideawriting process. All group participants are given 30 minutes to generate and document relevant ideas. The ideas are entered into the database.

Nominal Group Technique:
The Nominal Group Technique (NGT) may be used to 1) generate ideas, 2) adjust ideas and 3) classify ideas. NGT processes generate data to be persisted in the database as well as guiding the group participants toward documented, structured outcomes.

Interpretive Structural Modeling:
Interpretive Structural Modeling (ISM) may be used to 1) structure ideas, 2) interpret idea structures, and 3) adjust idea structures. ISM generates a significant amount of data that needs to be persisted in the Sqlite3 database.

Initial Software Architecture:
The initial OSSMTools software architecture development will focus on Phase Two processes and the creation of a embedded database to support the IM processes.

Initial Software Design:
The initial software system is based on Node.js, Vue.js and Sqlite3. The basic software module contains a database as well as functions to register users. Each individual user has access to their own material. The individual editing functions need to be analyzed and evaluated to determine the proper

collection of editing functions.

A database administrative role is needed to provide access to the complete collection of information in the database.

The entries into the database can be modified but not deleted. Records are copied before modification and keep in the same database table. Need to further evaluate the 'information history' concept and function. An open source solution may be available for this type of edit history function.

These functions will need further elaboration.

Conceptual Relationship Representations:
The augmented model-exchange isomorphism (AMEI) presents an isomorphic semantic representation of a relationship in three language types: prose, structured graphics and mathematics. A relationship needs to be carefully evaluated to determine the quality of the semantic alignment across all three language types. Types of numbers and measurement scales used in the mathematical expression are of special concern when evaluating the semantics of a real-world situation expressed in terms of the AMEI framework.

There are four measurement scales:

- Nominal
- Ordinal
- Interval
- Ratio

The nominal scale includes Boolean values: true and false. Warfield extended Boolean values to the ordinal scale by adding the > (greater than) and < (less than operators) to Boolean algebra. This is a specific example of carefully extending a measurement scale to provide insight into system structure. While human beings have a difficult time assigning an interval scale to preferences and values, many areas of system interest have been measured using either an interval or ratio measurement units. These existing measurements are now part of the common, accessible knowledge base. This existing system knowledge may be used in the structural modeling process, if correctly applied.

The effect of the measurement scale on the relationship type is an important but relatively unexamined area of structural modeling. Examples in Appendix 2 – GMU ISM Software of "A Handbook of Interactive Management," used the nominal scale and the ordinal scale. The clustering example used the participants collective knowledge to determine which objects belonged in a specific group. The ordering example used the participants collective knowledge to determine the order of objects in specific collection.

If the ordering example was based on individuals estimates of weight, then the ordinal scale is the most appropriate. However, if the ordering example was based on measured weights, then it is possible the use the interval scale, if necessary. The "is-heavier-than" relationship creates an ordered collection of objects. If the weights are known by careful measurement, then this ordered collection may be placed on an interval scale. If an interval scale is used, then the use of "scoped transitivity" is a possibility. The concept of scoped transitivity needs further description and development.

The relationship "near" is on example of a relationship that may have scoped transitivity. The

relationship near may used to place items in an ordered collection. However, the ordering capability of the relationship is dependent of the specific interval measure assigned to near. A collection of objects may contain objects that are not near to each other or the collection may contain objects that are near one or more objects in the collection. In this specific case, the relationship is dependent on an interval measure. If interval knowledge and information associated with the near relationship is available to support the system structuring activity, then scoped transitivity may be useful in some cases.

In addition to adding an ordinal scale to Boolean algebra, Warfield also add an interval scale to a binary matrix by adding weights to the binary matrix edges. In this manner, a basic nominal scale (True, False), has been extended to an ordinal and interval scale using the structural modeling techniques developed by Warfield.

Proposed New Process Interfaces And Representations:
The careful design of measurement scales, question types and inquiry approaches presents the opportunity to expand computer based support associated with structural modeling tasks. These new structural modeling interfaces are based on the availability of extensive, ubiquitous online information.