An Adaptive Total Strict Partial Order:
A General System Structuring Approach

Joseph J. Simpson and Mary J. Simpson

March 18, 2016

Introduction

Structural modeling blends the formal information associated with mathematics and logic together with empirical information associated with a specific real world problem to create an outline of the problem structure. Structural modeling has three components (see also Figure 1):

1. Basic structural modeling that is focused on specific, formal mathematical constructs;
2. Interpretive structural modeling that is focused on the characteristics and attributes of a given 'real world' problem; and
3. Structural integration modeling that places the information acquired from the real world problem into proper alignment with the applicable formal mathematics.

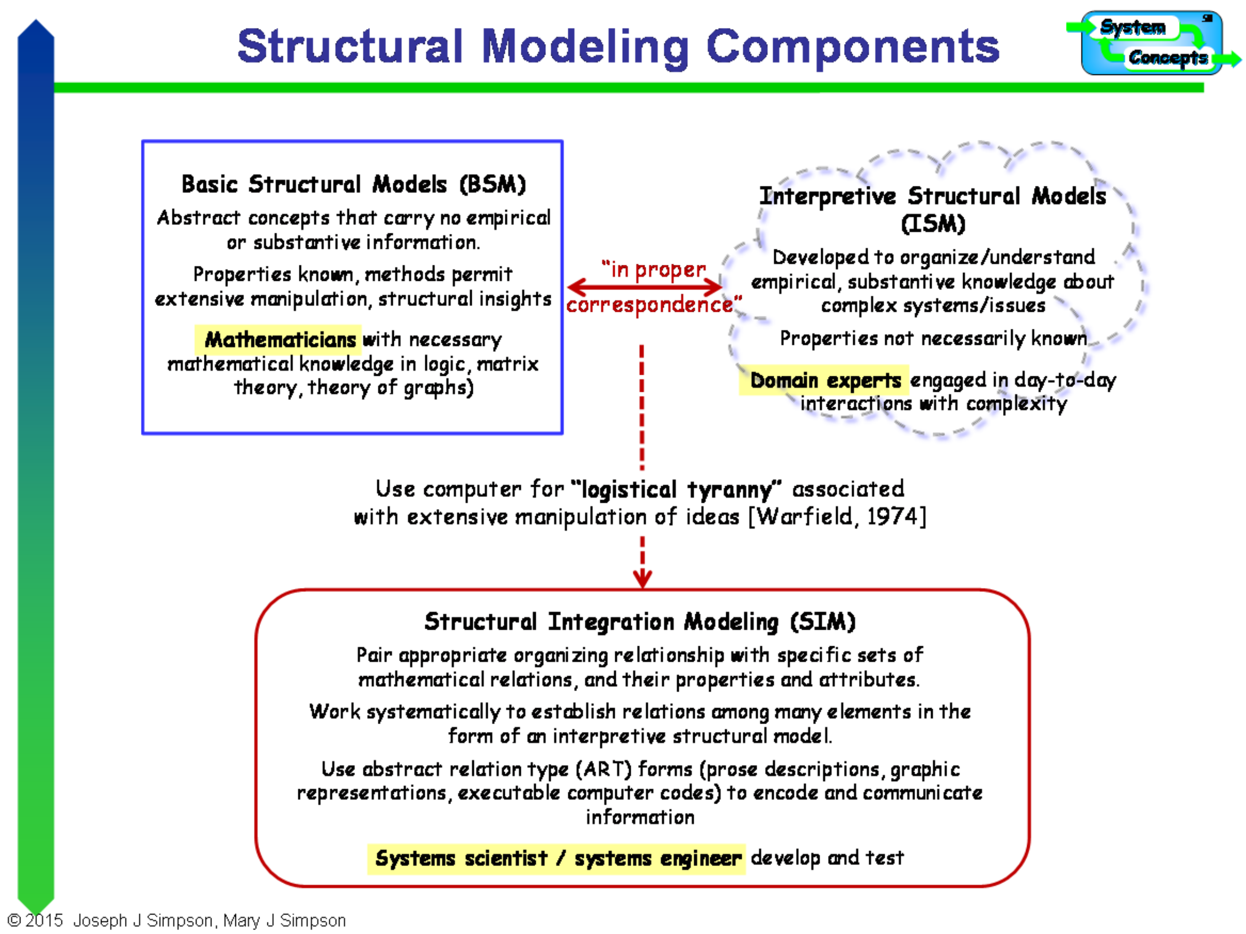


Figure 1 – Structural Modeling Components

This paper outlines some of the specific types of information that are necessary to effectively implement structural modeling, as well as some basic process and work flow considerations needed to create an acceptable structural model.

Systems, Relations, Logic and Structure

For the purpose of an adaptive total strict partial order in general system structuring, a system is defined as a relationship mapped over a set of two of more objects. This system definition places the organizing system relationship on the same level of importance as the objects that are members of the system. Every natural language relationship has a set of logical properties. Once the logical properties of a given natural language relationship are identified, then these properties may be used to create mathematical entities that are isomorphic to that particular natural language relationship. The set of logical properties contains real world information packaged in a manner that supports the creation and use of mathematical constructs that have the capability to greatly enhance our ability to identify previously unknown system structures.

General System Structuring Approach

As briefly outlined above, the real world natural language relationship is used as a basis to create mathematical constructs that assist in the structuring of unknown systems. The structural modeling literature is filled with a wide array of mathematical methods and processes that may be used in different situations. Much of Warfield's published work on mathematical methods associated with structural modeling was used to communicate techniques and approaches to be used in specific situations. Warfield indicated that many of these techniques and approaches were at a ‘proof-of-concept’ level, and generally were disassociated from a complete, large scale, real world problem. These ‘proof-of-concept’ communications left an array of method, boundary, interface and implementation issues.

Simpson and Simpson created the abstract relation type (ART) and the augmented model-exchange isomorphism (AMEI) to assist in addressing these boundary, interface and implementation issues. Further, a simple, real-world problem based on ordering 19 cities was developed to use as a standard example to support detailed discussions. The 19 cities problem uses the 'north-of' relationship to order the cities in the problem. The natural language relationship by itself is not enough information to effectively structure the system. In the first 19 cities example, the real-world problem was also constrained to allow only one city at each level. This constraint created a strict order of the cities.

The ART form was developed to provide a standard format to capture all the information needed to effectively perform structural modeling. The AMEI was developed to organize the set of different system structure types based on their logical properties. While these formatting and organizing approaches are essential to support detailed discussions of system structuring, they are not enough to support the structuring of an unknown, unconstrained system by themselves. Unknown and unconstrained systems need an adaptable approach.

The 19 cities problem is used as an example of an unknown and unconstrained system structuring problem. If the structure of the 19 cities is truly unknown, then a number of different structural configurations are possible:

- Each city is at a separate, distinct latitude.
- All the cities are at the same latitude.
- There is a combination of individual cities and groups of cities in the final system structure.

In a truly unknown and unconstrained system structure, the structural modeling methods and techniques must be able to adapt to the system structure as it is discovered. The adaptable, total strict partial ordering (ATSPO) approach is the general approach that is able to address all of these possible system structural configurations.

The ATSPO approach does not allow the use of the antisymmetric logical term. The prohibition against the antisymmetric logical term is based on the concept of complexity reduction. The antisymmetric logical property contains an uncertain choice at the center of the logical property definition. This choice is between two completely different logical group types. Using antisymmetry introduces unnecessary complexity and confusion in the system structuring process. Many complexity issues associated with system structuring may be traced to the use of antisymmetry. Antisymmetry includes the logical properties for equality: reflexive, symmetric and transitive. An ordering relationship usually has a set of logical properties consisting of: irreflexive, asymmetric and transitive. This fundamental mismatch in logical properties generates a great amount of confusion in the practice of system structuring. A partial order on a set has the following logical properties:

- Reflexivity
- Antisymmetry
- Transitivity

Using only the concept of a partial order eliminates a number of valuable system configurations from consideration. If the logical terms for equivalence are used by themselves, then a wider variety of logical property sets may be used in the system ordering and/or structuring process. These logical property sets include:

Set 1: Irreflexivity
Asymmetry
Transitivity

Set 2: Non-reflexivity
Asymmetry
Non-transitivity

This wider set of logical properties enables a better matching of natural language relationships and their logical properties that describe the relevant mathematical relation. Web applications will be developed to support a wide range of AMEI logical property groups.

Conceptually the ATSPO has a two-step evaluation process. The first step creates equivalence classes of the system objects. The second step is the ordering of the equivalence classes. In practice, these two steps must be integrated during the system discovery process.

The 19 Cities Problem

When the ATSPO is used along with the current web application under development and named the general structural modeling grid, a few changes need to be made to the original 19 cities strict ordering web application. These changes include:

- Asking the subject matter experts if the cities are at the same level,
- Dynamically changing the size of the base matrix, and
- Dynamically creating equivalence classes.

These changes are now implemented in the new draft web application.

The 19 cities problem uses the 'north-of' natural language system structuring relationship to structure the unknown system. This natural language relationship has the following logical properties:

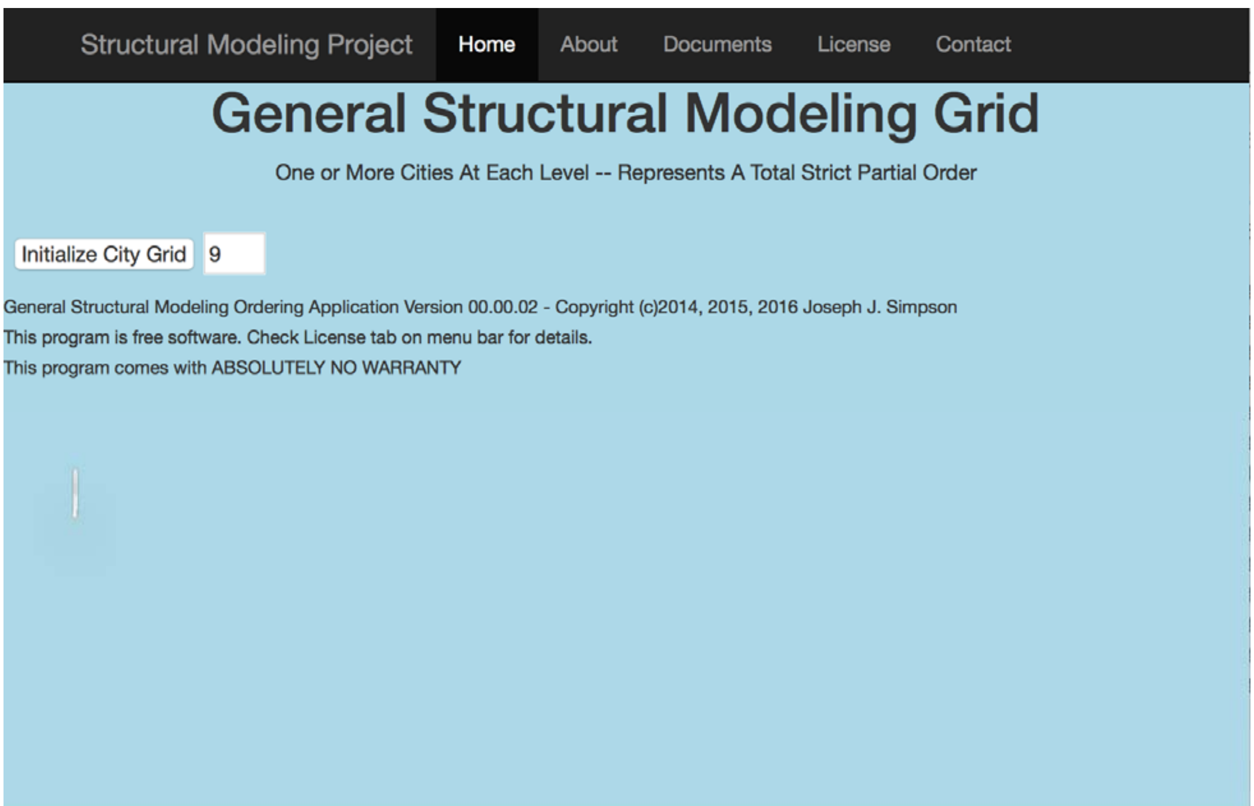
- Irreflexive (a city is not north of itself)
- Asymmetric (if city A is north of city B, then city B is not north of city A)
- Transitive (if city A is north of city B, and city B is north of city C, then city A is north of city C)

To reduce the size of this document the matrix size has been reduced to a 9 by 9 matrix that represents the evaluation of 9 cities.

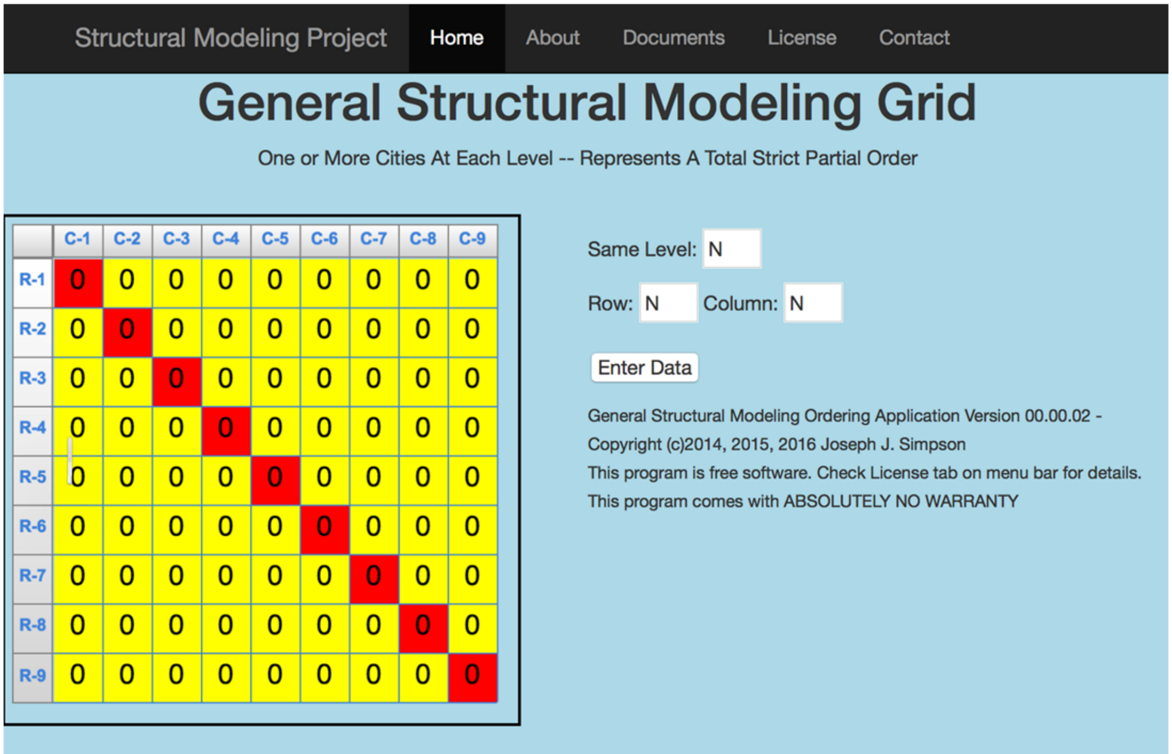
First Case – All Cities at same level

The first case presented is the case where all the cities are at the same level, producing only one equivalence class and nothing to order.

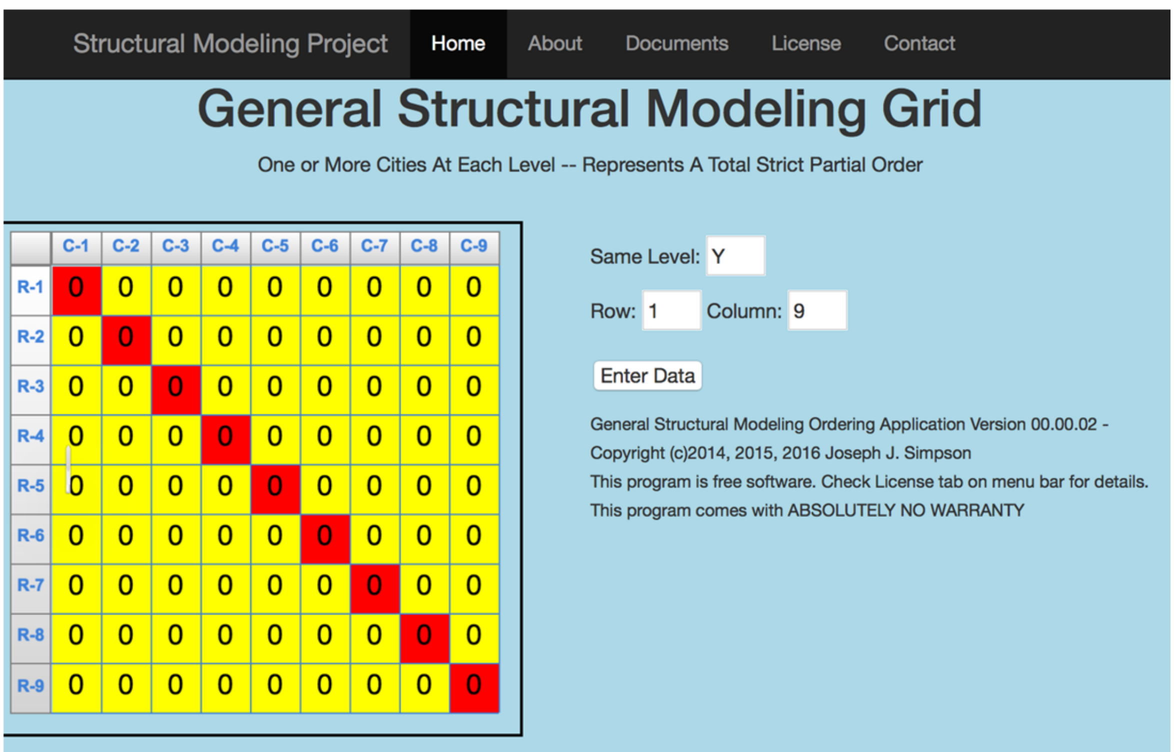
Step 1: The initial screen allows the user to designate the number of elements (cities). Entering a **9** (nine) produces the 9 x 9 matrix shown in Step 2



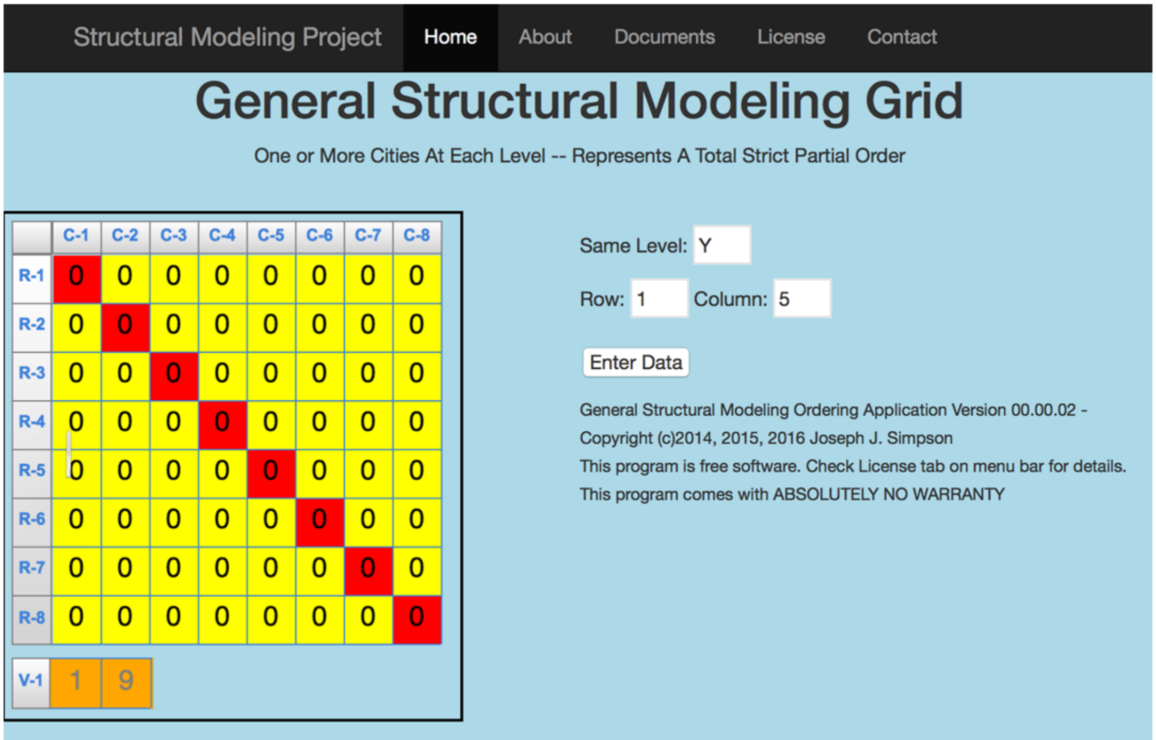
Step 2: Displays the 9 by 9 Matrix, showing an '**N**' in the 'Level', 'Row' and 'Column' boxes.



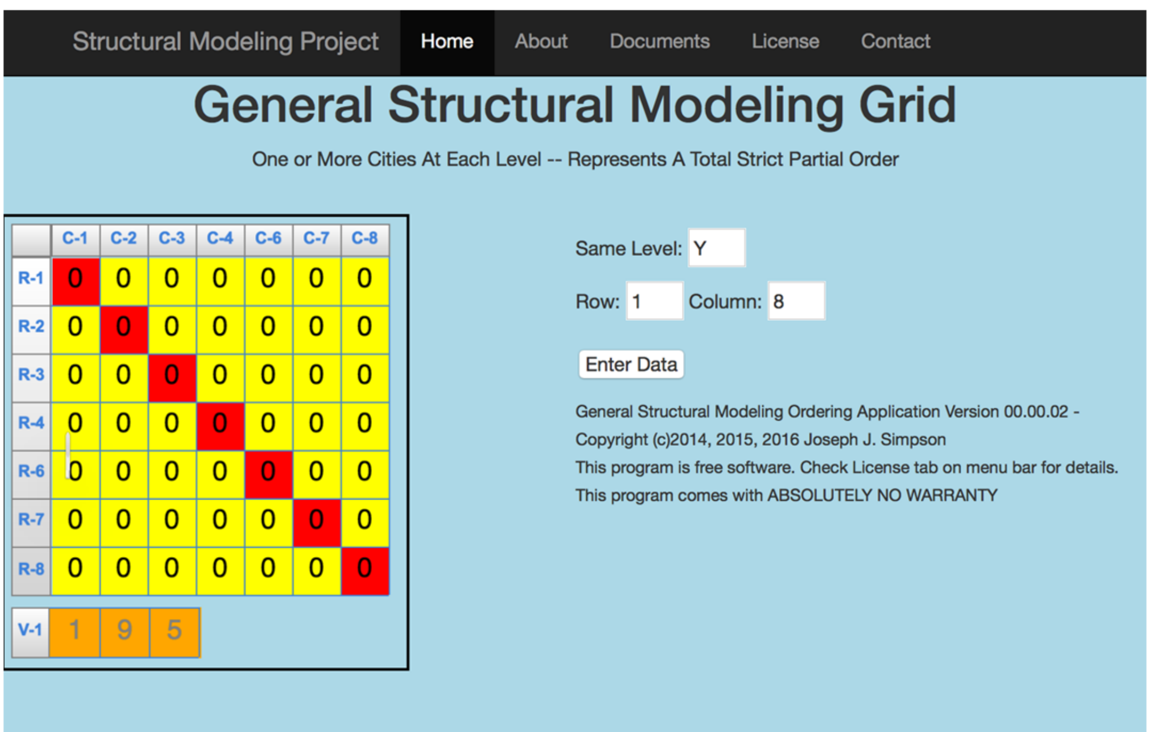
Step 3. Enter a '**Y**' in the 'Same Level', a '**1**' in 'Row' and a '**9**' in 'Column' (as the cities 1 and 9), and **click on 'enter data'**.



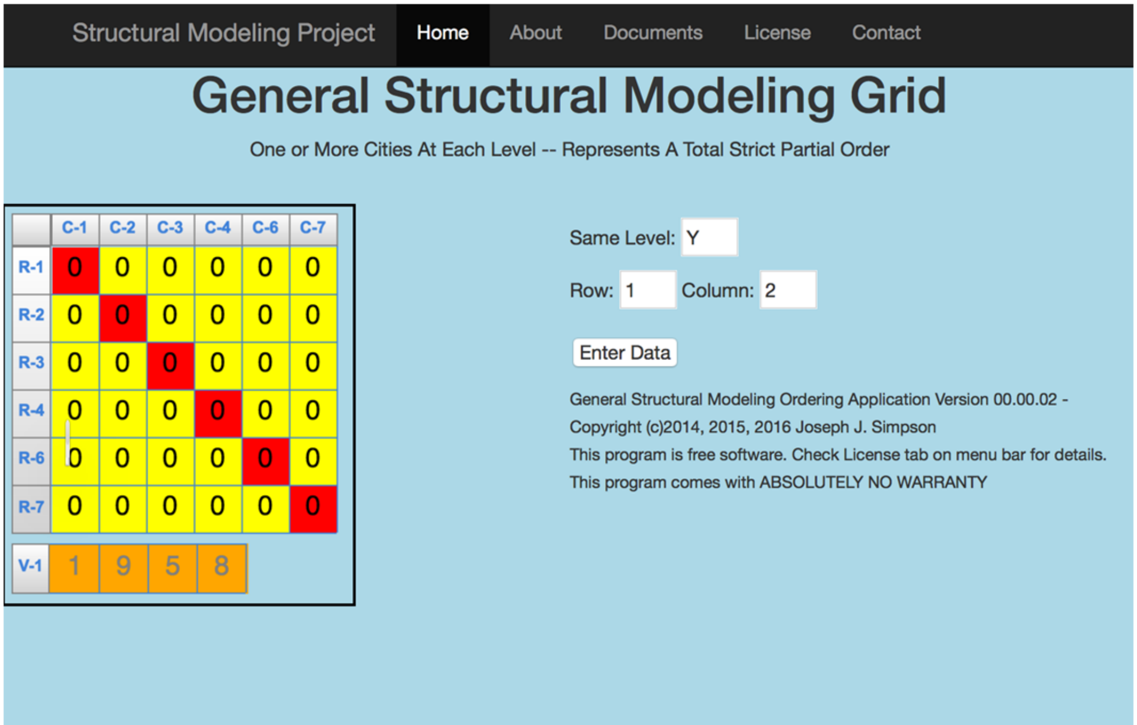
Step 4. The equivalence vector with **1 and 9** appears under the matrix. Now enter a '**Y**', the cities '**1**' and '**5**' and **click on 'enter data'**.



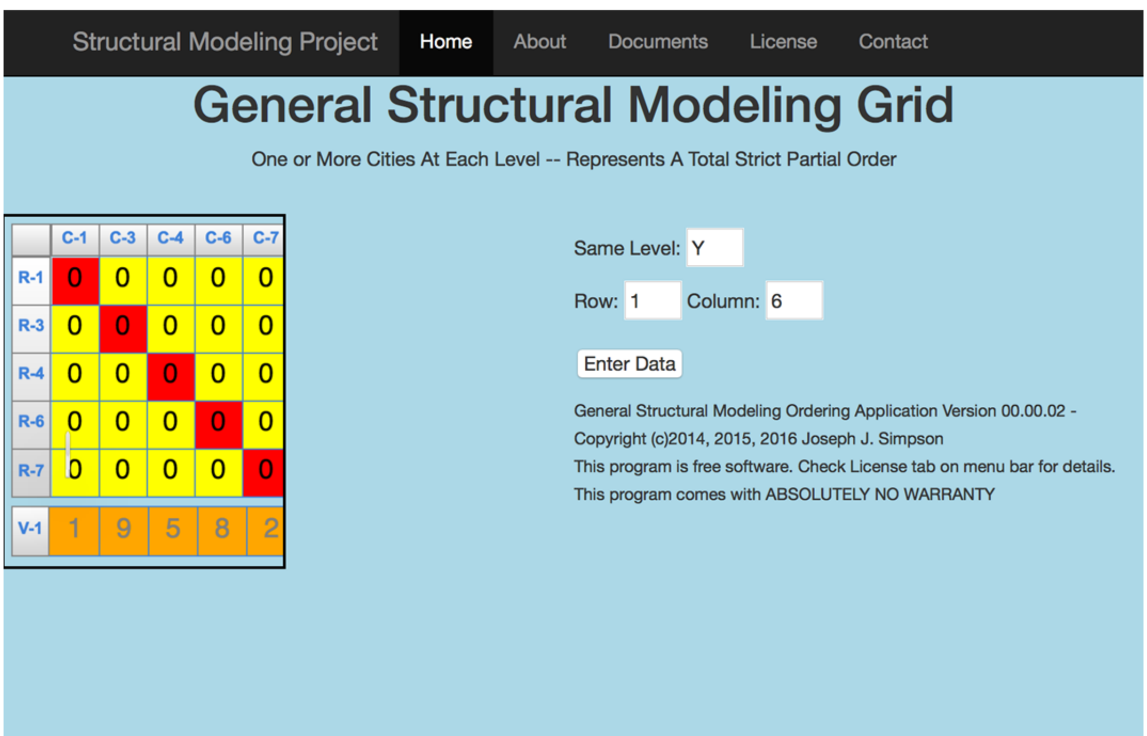
Step 5. See below that **City 5** has been added to the equivalence vector. Now enter a '**Y**', the cities '**1**' and '**8**' and **click on 'enter data'**.



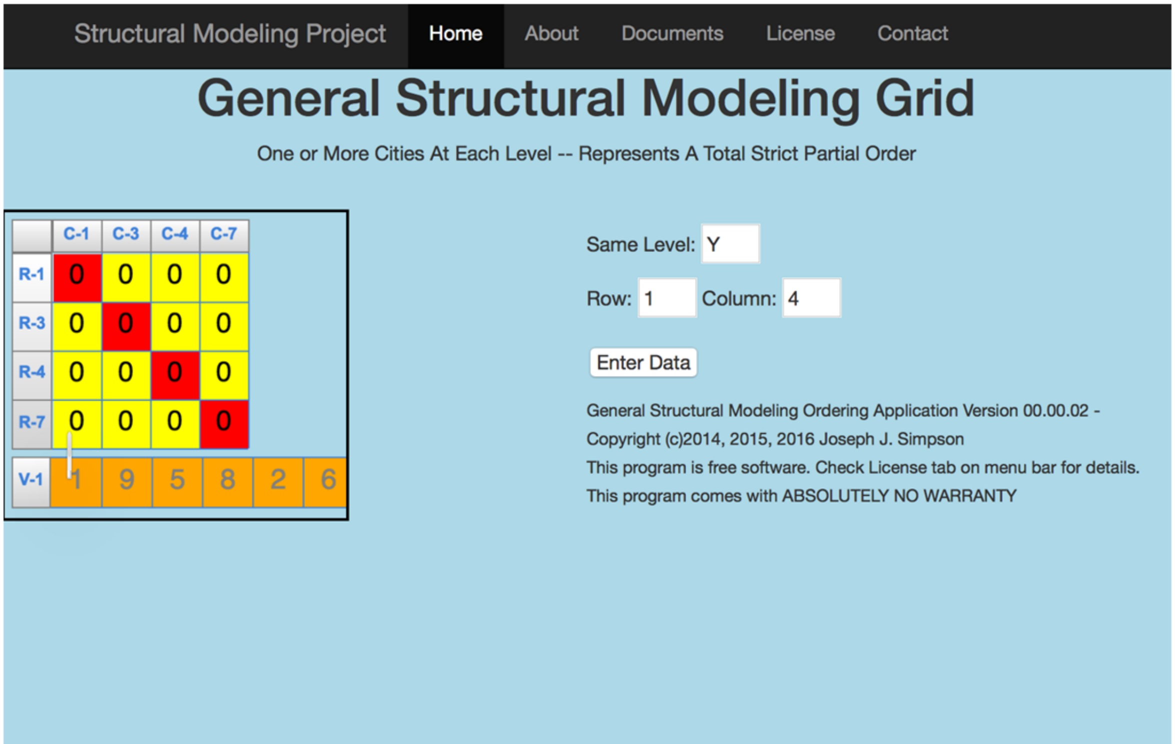
Step 6. See below that **City 8** has been added to the equivalence vector. Now enter a '**Y**', the cities '**1**' and '**2**' and **click on 'enter data'**.



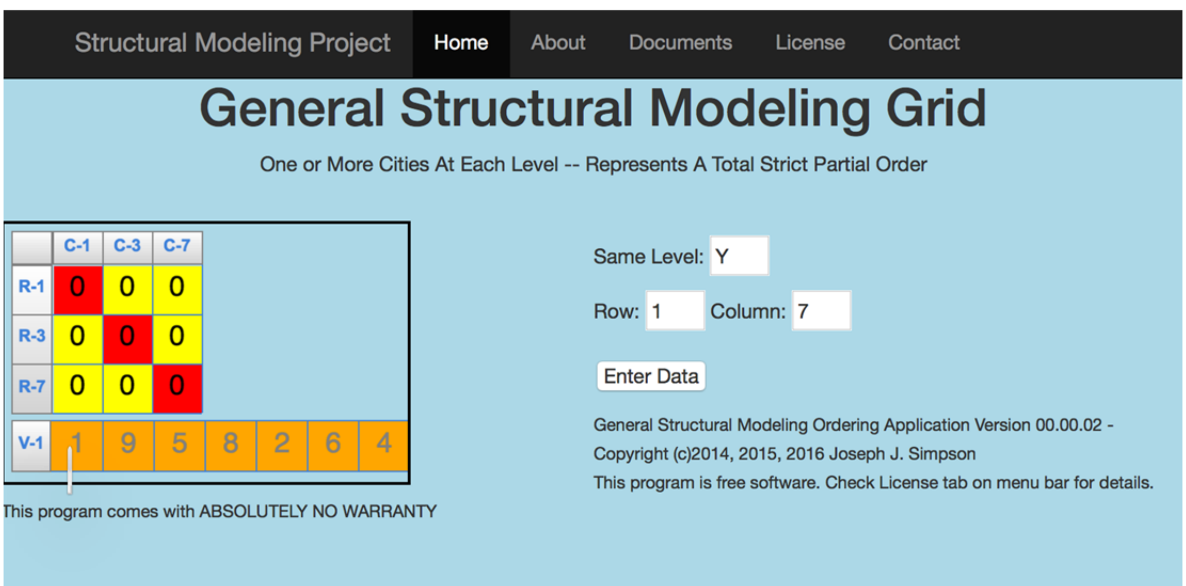
Step 7. See below that **City 2** has been added to the equivalence vector. Now enter a '**Y**', the cities '**1**' and '**6**' and **click on 'enter data'**.



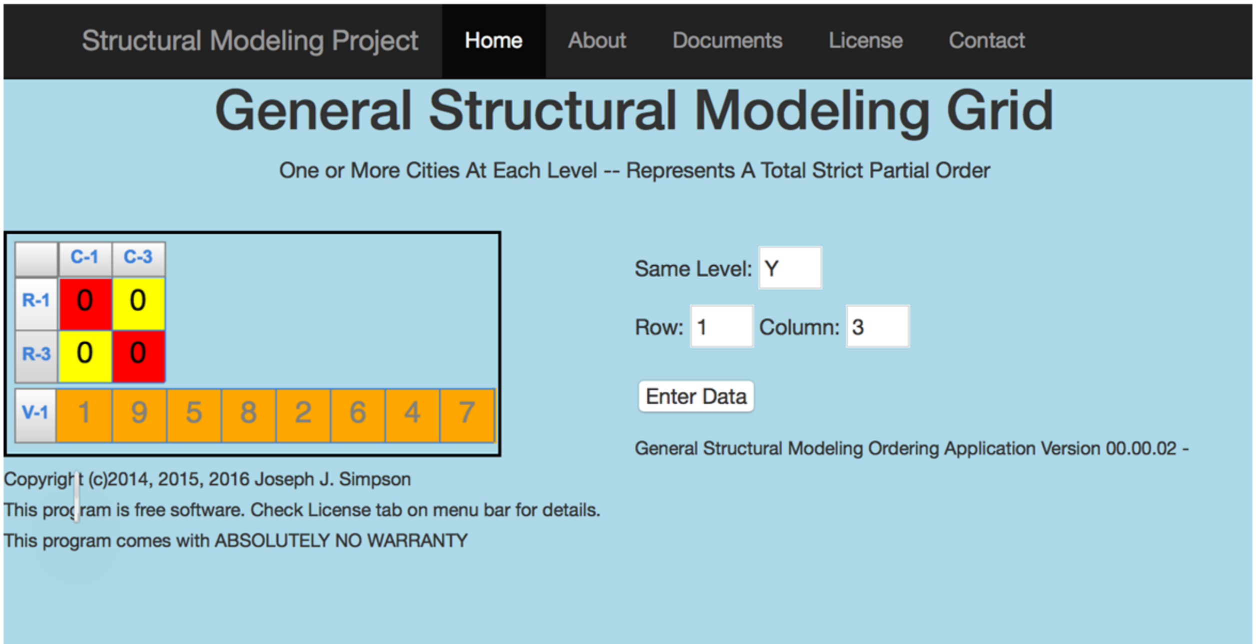
Step 8. See below that **City 6** has been added to the equivalence vector. Now enter a '**Y**', the cities '**1**' and '**4**' and **click on 'enter data'**.



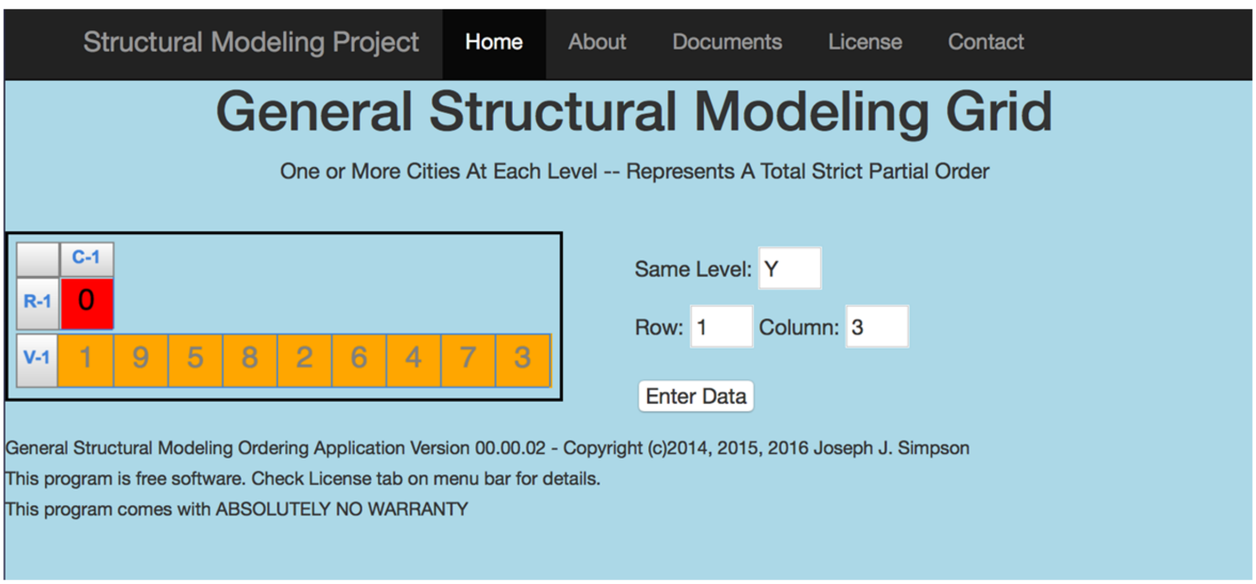
Step 9. See below that **City 4** has been added to the equivalence vector. Now enter a '**Y**', the cities '**1**' and '**7**' and **click on 'enter data'**.



Step 10. See below that **City 7** has been added to the equivalence vector. Now enter a '**Y**', the cities '**1**' and '**3**' and **click on 'enter data'**.



Step 11. See below that **City 3** has been added to the equivalence vector.



The Process is now complete. Notice that **all information is empirical** information; **no inference can be used in these cases**.