The following items in ***bold*** *are requirements specified by an experienced simulation user who was evaluating new products. The Simio narrative responses are the parts not in bold.*

The requirements for this software are:

- **Software must be able to build, debug, run, and collect data from Discrete Event Simulation models**

The Simio modeling software lets you build and run dynamic 3D animated models of a wide range of systems – e.g. factories, supply chains, emergency departments, airports, and service systems. Simio is an object-oriented DES product that includes tightly integrated debugging (Trace, Watch, Break, and Step capabilities) and extensive data collection and analysis capabilities.

- **Software must run on standard Windows PCs**

Simio is built on and for Windows. It supports Windows XP, Vista and Windows 7. Minimum hardware requirements include:

- 1 GHz or faster processor
- 1 GB of RAM (2 GB or more recommended)
- Available hard drive space of 500 MB minimum for installation
- 1,024x768 display (higher resolution recommended)
- A 16-bit graphics card with 128 MB compatible with DirectX ver. 9

- **Software must be a production level software (no beta code)**
  - **Must be actively supported and maintained by the developer**
  - **Must be responsive to user requests for new features**
  - **Must be responsive to issues with the software**

Simio has already released 4 major versions and will soon release Simio Version 5. Simio is developed under an agile process which results in minor production releases (called “sprints”) every three weeks. These sprints provide very quick turn-around on feature requests and issues. The sprints are rich with features, each typically having 2-4 pages of release notes. Past release notes are available on our web site at http://www.simio.com/download/simio.htm

- **Software must use an object-oriented modeling framework**
  - **Must allow the model objects to be built, tested, and reused without programming**
  - **Must allow the model objects to be individually changed without affecting the rest of the model in which they reside**
  - **Must allow for the development of custom object libraries without programming**
  - **Must allow the sub-classing of objects**
  - **Must allow for sub-modeling**

- **Must allow object instances to be enhanced with add-on processes including time delays within the process**

Simio is an object-oriented application that provides a true object-oriented modeling environment to users. This object approach to modeling allows models to be built by combining objects that represent the physical components of the systems. You can build libraries of objects for your own purposes, or build libraries to share across your enterprise. You can also build commercial libraries that you market and sell to others.

One basic principle of Simio is the notion that any model can be an object definition. Models are used to define the basic behavior of an object. It is very easy to take a model that you have developed and then "package" it up for someone else to use as a building block for their models. It is also very easy to build sub-models within your own project, and then build your model using these sub-models as building blocks. One of Simio's unique advantages is the ability to give full access to object-building capabilities with no programming required. Instead of using C++ or Java to define objects, in Simio they are defined graphically using processes. Simio meets all of the above specified requirements.

- **Software must be scalable for application to various project sizes**

Simio uses a unique 3 tiered approach of Object Definitions, Object Instances, and Dynamic Object Runspaces. The extra tier allows better scalability than other object-oriented modeling packages because the objects that are duplicated the most can be much smaller.

- **Software must have a graphical user interface for use by model developers and users.**

Simio has a rich graphical user interface (GUI) that is state of the art. We were among the early adopters of the Microsoft Office 2007 GUI and we have already updated to the Office 2010 interface. In addition we are constantly soliciting user feedback and innovating new solutions to let our GUI help users model better. One such innovation is the use of extended tool tips. Since no one likes to read the documentation, we routinely build how-to-use instructions right into the GUI.

- **Software must allow for graphical model building only (no software coding required) – this application will be used by Modeling & Simulation professionals, not software professionals**

As noted above, no coding is required for basic or advanced modeling tasks. Our unique architecture allows full access to complete model-building and object-building capabilities through the use of graphical processes rather than coding.

- **Software must be developed using the Microsoft .NET framework to allow the use of an abundant amount of programming resources available to it (Microsoft Development Network)**
  - **Must be able to integrate with the Microsoft .NET languages to allow easy interfacing within a larger analysis system**

Simio is built with Microsoft Visual Studio and the Microsoft .NET Framework. While no coding is required for routine model building, Simio has a number of interfaces built in to support advanced users to extend the Simio product for example building custom Elements, Steps, User Rules, and IO functionality. These extensions can be added using any of the over 50 .NET languages. Simio also has an API to support interfacing with external programs and routines.

- **Software must be developed in an Object Oriented language to facilitate integration within a larger modeling and simulation framework**

Simio is built in C# using with Microsoft Visual Studio and the Microsoft .NET Framework. C# is an object-oriented language and the Simio platform supports integration with an external modeling and simulation framework.

- **Software must save models in an XML format**

Simio supports three file formats, all XML based. The first is an SPFX format, similar to what is used by Microsoft office products. It is written in XML, but then compressed to a binary for small file size. Simio also includes a straight XML format for the modeling components. The third format is a “simproj” format that outputs all the model components as a set of files that is most appropriate for saving in a source control or revision tracking system.

- **Software must support an Application Programming Interface (API) for integration with other models and software**

Simio has an API to support interfacing with external programs and routines and several example applications are included along with hundreds of pages of technical documentation.

- **Software must support concurrent 2D and 3D Animation**

Simio includes both 2D and 3D concurrent animation. By default Simio provides a top-down 2D view of the facility model. This is often a very convenient view for creating and editing your model. However even when working in 2D, Simio facility models are defined in 3D with no additional effort. You can easily switch between the 2D and 3D views in real time as well as select various facility viewpoints.

A traditional problem with 3D has been the effort and artistry required to create a 3D model. Simio eliminates the effort and difficulty by making 3D automatic. The artistry is addressed by not only supporting import of common graphic objects, but also incorporating a live link to Google Warehouse – an on-line library of over a million symbols available for free.

- **Software must include a library of standard modeling objects, including but not limited to**
  - **Resources**
  - **Entities**
  - **Transporters**
  - **Conveyors**
  - **Paths (should be able to be single and bi-directional)**
  - **Probabilistic and Conditional Routing Logic**
  - **Object Creation and Termination**

Simio's Standard Library consists of 15 objects that include all of the above. It also includes Workers (e.g. resources that are mobile and can carry entities), very sophisticated conveyors and vehicles, several types of paths, and batching/combining. The definitions of all of these objects are open. While their functionality is already comprehensive, if you desire additional or different functionality, you can derive a custom object from Simio objects, or simply use add-on processes to add behavior.

- **Software must include a method of setting up and running a set of experiments and comparing the results**

Simio's has exceptional experimentation capability tightly integrated within the Simio software itself, not as a separate afterthought component. With almost no effort you can set up and execute a proper statistical analysis. Simio provides automatic Ranking algorithms to quickly determine which scenarios are the best candidates based on replications that have been run. Simio also provides Ranking and Selection algorithms based on the latest research that will compare scenarios and iteratively run additional replications to reduce the variability enough that the best alternative can be reliably determined.

Simio has taken advantage of state-of-the-art research in output analysis to build in a graphical tool to help you analyze and compare scenarios. We have extended recent developments by Dr. Barry Nelson to generate SMORE Plots (Simio Measure of Risk & Error) to display both the expected value of a scenario and multiple levels of variability and risk analysis behind the expected value.

- **Software must include real time displays and graphs of simulation output**

In addition to the post-run analysis described above, Simio provides a rich set of real-time output (e.g. output generated while a model is running). Simio offers status labels, status plots, Status pie charts, circular gauges, linear gauges, and interactive buttons. These can be placed in one or more separate "Dashboard" windows or can be placed directly in the facility window. They can even be attached to an object. When attached to a dynamic object (e.g. an entity or transporter), the feature will move with the dynamic object. This latter capability provides high model clarity because you can have an arbitrary amount of graphical information (like a due date or work time remaining) traveling with an entity.

- **Software must support optimization of response parameters**

Simio includes the capability to set up and run a series of scenarios and do manual optimization. It also includes both built-in features and free add-ins to do statistically valid ranking and selection. Simio has the ability to run user-written add-ins that can be used to support advanced experimentation, including optimization. An optional Simio component is OptQuest for Simio, which is the leading optimization technology in the DES world.

- **Software must be able to take advantage of multi-core processing to simultaneously make multiple replications**

Simio's Experiment processing takes full advantage of all available processors. For example, on a common quad-core machine up to 4 scenarios will be run concurrently and if running fewer than 4 scenarios, up to 4 replications will be executed concurrently. This allows the experimentation phase of a project to proceed significantly faster. Simio Team Edition also allows distributing runs across other computers in your local work group with no extra licenses required.

- **Software must include a post-run simulation hierarchical output report that can be readily reorganized**

In addition to standard reports, Simio provides output in a pivot table (similar to what you might know from Excel or data analysis products). Pivot tables provide a powerful "data mining" tool that is especially good for applications with large amounts of data such as simulations. The Pivot Table organizes the data by columns. Each column can be sorted to prioritize the data, filtered to reduce the volume of data, or rearranged to change the structure and hierarchy of the report.

Simio also supports exporting the detailed or summary results into many different forms including CSV (for further processing in Excel and many other applications), PDF, HTML, and others.

- **Software must allow object entities to follow pre-defined processing routing sequences**

Simio includes sequence tables which not only allow a pre-defined processing sequence, but also support adding any number of additional data items that can be used at each location (e.g. a processing time at each location). The sequence can be used intermittently if desired to do special processing or tasks that are “off sequence”. The sequence can be changed at any time for example when an entity gets escalated or perhaps needs to follow a repair cycle. Sequences can be imported and exported to external databases. Sequences, like all tables, can be part of a relational database which supports easy integration with external data.

- **Software must allow for a minimum of two-dimensional with a preference of up to ten-dimensional data array tables**

Simio supports assignable states in scalar, vector, matrix, and up to 10 dimensions. Matrix arrays can be associated with a table so they can be automatically read, dimensioned and initialized from external data.

- **Software must allow adding behavior to entities and resources so objects can interact and make decisions**

Because Simio is object-oriented (OO) and its objects are open, custom behavior can be added to any objects including entities and resources. But Simio is unique among OO products in that it makes the full power of object building available without requiring coding like Java or C++. Objects can be built and enhanced using graphically defined process logic. In addition, most objects (including entities and resources) have add-on process triggers to make it particularly easy to add behavior to interact with other objects – to change object behavior without actually changing the object definition.

- **Software must support multiple entity types with different characteristics**

Simio offers an unlimited number of entity types. Each entity type can have its own distinctive properties, states and built-in behavior. Each entity can have not only unique animation, but also sets of animation symbols.

- **Software must support resources which can be seized and released and moved over paths to a location where needed**

In addition to the extremely powerful basic Resources, Simio has Vehicle and Worker objects. Both of these can be controlled explicitly (e.g. Seize, Release, and Move to a specific location) as well as controlled automatically (e.g. I need a pickup by a vehicle that can carry at least 10 tons).

- **Software must include sample models to illustrate the concepts of the software objects and techniques in model building and use**

Simio comes with a set of example models that are full models illustrating a complete operation (e.g. a Mine, or Airport). Simio also contains a large set of "SimBits" which are small, very well-documented models that illustrate how to do a specific task. SimBits are very helpful in learning Simio. Simio also provides a number of short training videos as well as a full 8 hour training course on-line at no charge.