THE EMULATE3D FRAMEWORK FOR THE EMULATION, SIMULATION, AND DEMONSTRATION OF INDUSTRIAL SYSTEMS

Ian McGregor

Emulate3D Ltd
Reading Enterprise Centre,
RG6 6BU, United Kingdom

ABSTRACT

Emulate3D industrial engineering products are designed to address the needs of several vertical markets. Pre-eminent among these are automated material handling in all its forms, and the airport baggage handling industry. Both are experiencing considerable changes due to the ongoing development of internet-driven buying patterns on the one hand, and the continuing rise of budget airlines on the other. The Emulate3D framework forms the technical base underpinning the growing range of Emulate3D industrial products. This framework facilitates the development of products to meet the requirements of different levels of users across the various stages of an automation project. Not only can Emulate3D developers create new generic features and functionality within the framework, but end users can also use it to modify the products to their particular company requirements.
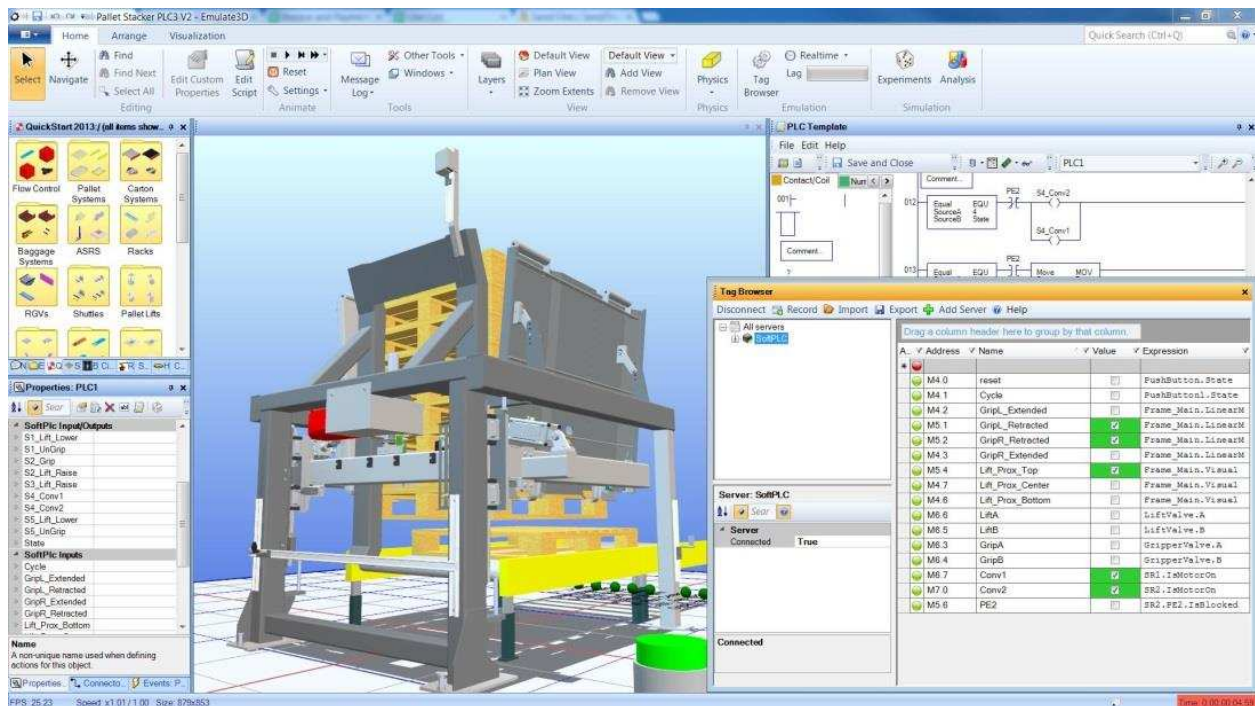
Figure 1: The Demo3D user interface.

1   INTRODUCTION

Changing buying patterns continue to generate shock waves throughout the postal system in its various forms, with the massive growth in household deliveries at the core. Distribution centers struggle to become more responsive, more flexible, and increase productivity while cutting operating costs, and managers look to automation to provide this feature mix. Automation system providers, integrators and consultants seek to develop more cost-effective and appealing solutions, demonstrate them to their clients, optimize their use, and ensure their safe and robust operation. Whether they need emulation, simulation, or demonstration capabilities, the virtual approach provides demonstrable advantages to achieving a good solution. Correctly used, simulation provides the most economical and repeatable way of exploring relevant options and narrowing down the field of possible solutions to a manageable number. As an impartial judge, simulation assists time-efficient decision-making to the satisfaction of all stakeholders, and models provide a credible digital representation of the state of the project.
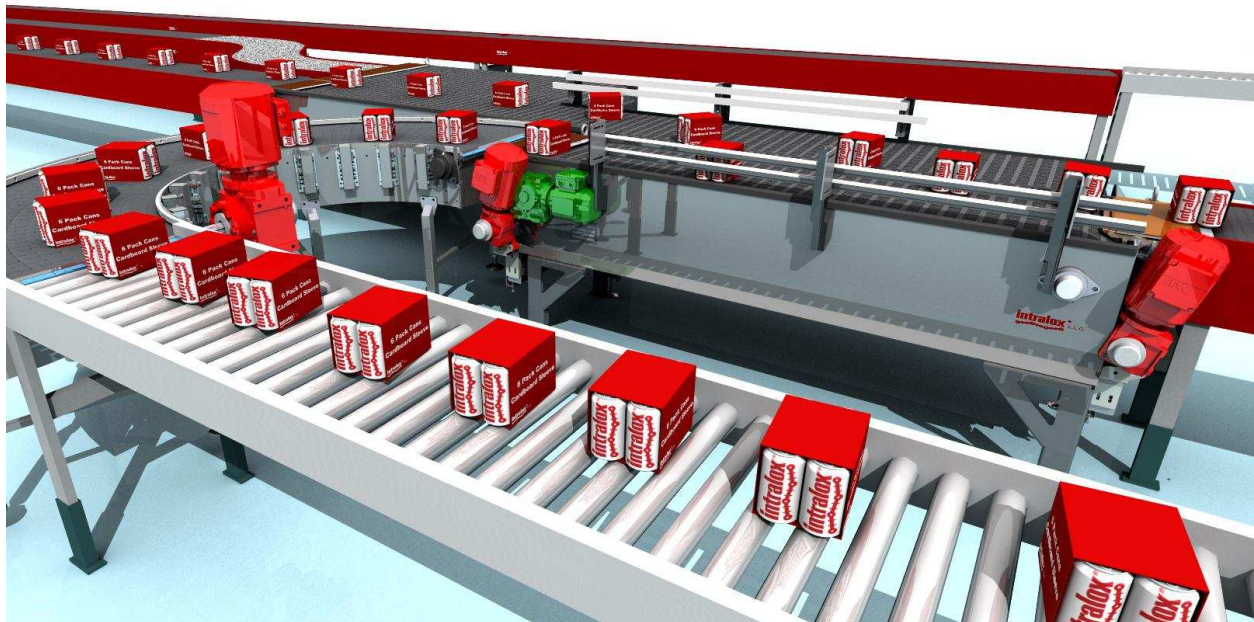

Figure 2: Emulate3D's unique environment allows production lines to be accurately modeled.

2   OVERVIEW OF EMULATE3D

The Emulate3D range of products reflects the fact that automation projects pass through several identifiable stages with different sets of requirements. These phases are as follows:

- Sales phase (Demo3D and Layout3D – developing and presenting initial solutions)
- System development and build phase (Sim3D – throughput analysis, system refinement and improvement)
- Implementation and start-up phase (Emulate3D – controls testing and virtual commissioning)

2.1 Sales phase – The main requirement is to be able to present various solutions within a short time-frame, and respond rapidly to changes in the system specification. During this phase there are often several contenders for the business, so proposals are unpaid and therefore must be cost-effective. In order

to explain solutions and convince clients, layout fly-throughs, running demonstration models, videos, and rough costings are a great help, if not a necessity.

2.2 System Development and Build phase – A deeper analysis of proposals is carried out at this stage, often with more accurate data which becomes available progressively as decisions are taken and system choices arrived at. Demonstration models are refined and extended to become simulation models, and these are capable of providing a useful analysis of system response under various conditions, and predicting throughput. A range of operating conditions are often explored to determine system sensitivity to change, and to arrive at a robust overall solution, and this iterative "what – if?" approach is typical of this phase.

2.3 Implementation and Start-up phase – Traditionally the most difficult part of any project, this is the culmination of all the effort and investment, and the moment when the ideas and the development have to result in a working physical solution. Testing this solution is anything but straightforward, as real product has to be introduced to the system in a controlled and repeatable manner in order to test the control system. Any deviation from expected behavior results in the detailed analysis and repair of the system, followed by a restart of the test sequence to verify that the repair has not adversely affected any other part of the controls. The manual method of controls testing, as described, is notoriously hard to predict, and is often the reason good projects go bad, profitable projects turn into losses, and investors think twice before investing further. Virtual commissioning is a much safer off line approach, requires less investment, can be started earlier in the project cycle, and takes most of the testing off the project's critical path, which also reduces the time variability of the phase.

All development projects pass through the broadly recognized stages of an initial research among many possibilities, then a distillation down to a few reasonable ones, and then the selection of a preferred solution, and finally its implementation. Passing from one stage to the next in this process requires different levels of model detail, and the Emulate3D range reflects these. Different versions of the products in each of these ranges reflect the various user types and their feature requirements, and this allows different versions to be offered at various price points.

2.4   Emulate3D Approach

Simulation modeling consists of an initial model build phase, followed by verification and validation, to ensure that the model is not only accurate, but also fit for purpose. The resulting model may then be used for several applications, including demonstration, simulation analysis, and controls testing.

Clearly the main value of the model is extracted in the latter phase; the build and test phases represent costs. The use of tried and tested catalog items to build models presents users with two important methods to reduce these costs; firstly, the model build time is significantly reduced, and secondly, the amount of testing required is also minimized as the correct operation of the higher-level component parts is already known.

2.5   QuickStart – The Importance of Re-usable Catalogs

Emulate3D products are designed to allow users to progress past the model building phase as rapidly as possible, confident that their models are accurate and robust. At the core of Emulate3D technology is the notion of re-use, and so models are based on catalogs, or libraries of elements. The central catalog is based on a framework called QuickStart, and bears the same name.

Figure 3: A glimpse of the QuickStart catalog.

Catalog elements can be anything users need them to be, but the elements supplied with the products are based on material handling equipment, and include:

- Conveyors (carton, pallet, baggage, roller, belt, chain, injector, merge, divert, lifts, luffers, specialist, etc)
- Path systems (fork lift trucks, operators, automatic guided vehicles, shuttles, RGVs, automobiles, etc)
- AS/RS (racks, cranes, single/multiple product carriages, deep storage shuttle/slaves, miniload, etc)
- Sorters (drop tray, cross-belt, sliding shoe, chain lift, roller lift, turntable, pusher, etc)
- Robots (general purpose, palletizing, scara, gantry, delta, etc)
- Buildings (mezzanine, stairs, handrails, step overs, walls, etc)
- General equipment (pallet stacker/destacker, pallet wrapper, profile checker, slip sheet dispenser, etc)
- Controllers (vehicle manager, palletizer, depalletizer, TargetSelector, BasicProcess, Games Controller, Keyboard Controller, etc)
- Data Collection (Throughput, TimeInSystem, Counter, TimeInState, StateTracker, etc)
- Cameras (Camera points, cameras)
- Control Panels (test, panel, push button, text panel, switch before merge, etc)
- ...

Collections of static elements are also provided with the products, some of which are from third party 3D content providers such as Baumgarten™, FF-Digital™, Max-Realms™, and Cinema4D™. Users can also tap into the wealth of free 3D objects to be found in the Google Warehouse™.

2.6   QuickStart Framework

The QuickStart framework allows catalog objects to be snapped together intelligently to create viable systems consisting of elements which recognize each other and react accordingly. For example; fork lift

trucks can be connected to pallet conveyor which feeds into a shuttle system and then on to an AS/RS. Pallets from the AS/RS can be output to depalletizing robots which send loads to accumulation lanes from where they are built into orders by palletizing robots. Each of these pieces of equipment behaves, by default, just as you would expect it to, facilitating the rapid model build necessary to justify the broad use of a 3D modeling tool for sales presentations.

In order to match the model behavior to the client's requirements, each catalog element has a range of user-accessible properties which can be changed to modify its dimensions, appearance, and operation. More important than this, however, is the fact that all QuickStart elements are open, so trained users can fundamentally modify their behavior, or even create their own unique catalog items or material handling equipment systems.

2.7   QuickStart Build Approach

2.7.1   System Layout

Catalog elements are dragged into the build environment and will snap together, if it is appropriate for them to do so. Users often import a 2D DXF file of the layout to use as a basic template, and then QuickStart catalog elements are stretched and altered to fit the template. Parameters such as speeds or heights may be altered on single or multiple objects at a time, and where necessary, users can connect shuttles or fork lift trucks to input and output conveyor sections by dragging visual connectors.
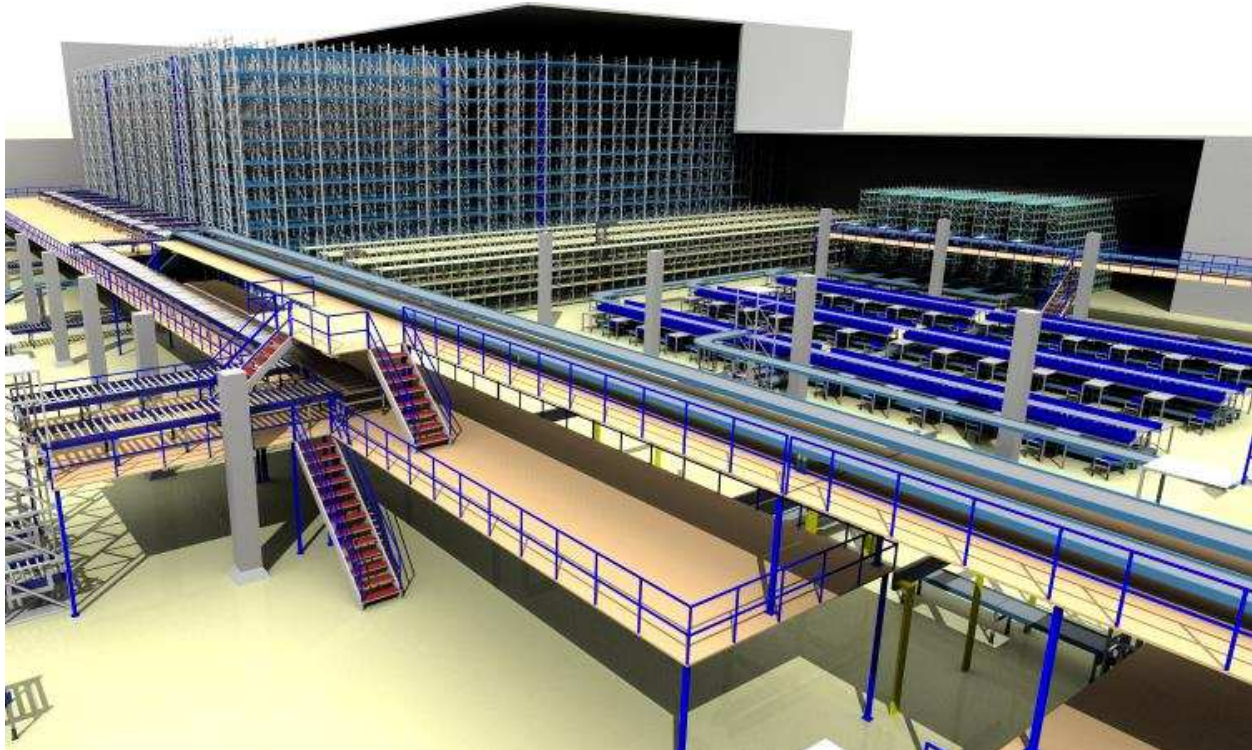

Figure 4: Layouts can be constructed rapidly.

2.7.2   Operational Logic

In many cases the default operation of the various pieces of QuickStart equipment is all that is needed; in the cases where it is not, two options are available to the user. The first is to use the supplied Control Blocks, created as a means of extending the QuickStart framework functionality in a user-friendly way, without resorting to scripting. Control Blocks are connected to the relevant pieces of QuickStart equipment using Visual Connectors, making it easy for the user to see where they have modified the default behavior, and greatly accelerating the model debugging process.

An example of a Control Block is a Target Selector, which can be used to separate a type of product from the others. Users can add filters into the Target Selector, so products of a certain type are sent one way, while all others go a different way. The Target may be many merges and diverts (or even systems) away from the initial "ListeningTo" conveyor section, but the underlying QuickStart takes care of the equipment-level decisions and allows the user to focus on defining product flows.

The second option available to trained users is to open the QuickStart object and edit the operational script to create the exact behavior required.
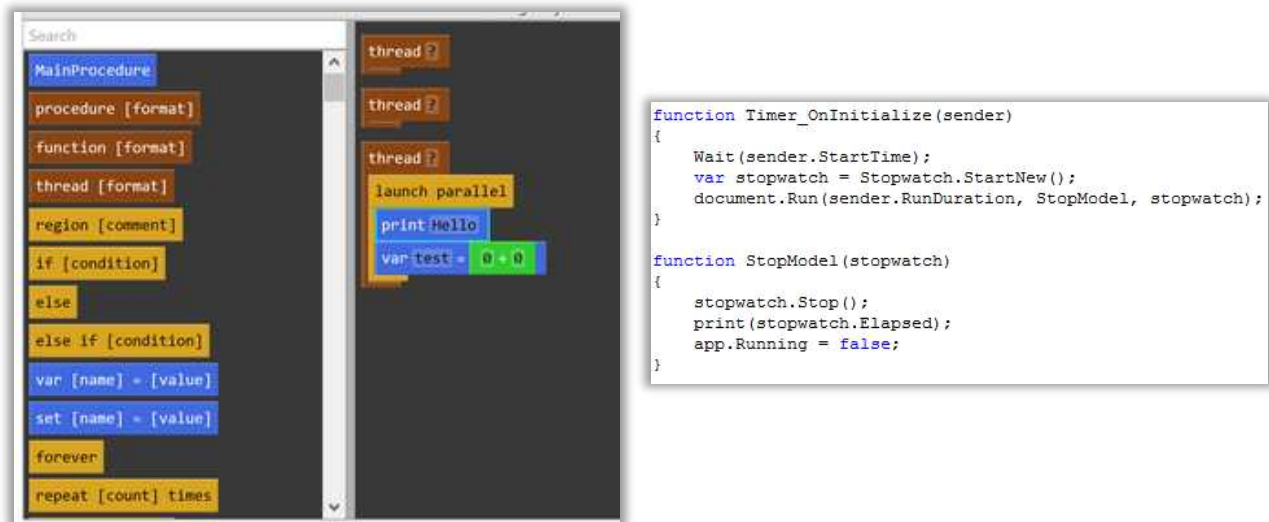

Figure 5: QuickStart elements can be opened and their logic modified.

2.7.3   Presentation or Analysis

Once the model has been constructed and the logical flow of parts through it has been created and verified, it is ready to be used for presentation or analysis purposes. In the case of a presentation model, users can create camera paths and views in order to generate fly-through videos which take viewers through the system in an enlightening and memorable way. Still images of various formats can also be created easily, for inclusion in reports or for use as banners at trade shows, for example.

Models built for simulation are often data-driven, with external spreadsheets providing the data to populate the model and inform its operation.

2.8   Four Different Ranges – One Approach

The four different ranges of Emulate3D products reflect the different application areas the technology is applied to.

2.8.1   Layout3D

Layout3D allows users to build industrial layouts quickly from supplied equipment catalogs and imported CAD, present their proposals interactively to project stakeholders, then generate videos and multi-view plots to leave behind. Creating initial layout concepts often requires frequent modifications, so working with Layout3D is designed to be quick, easy, and cost-effective.
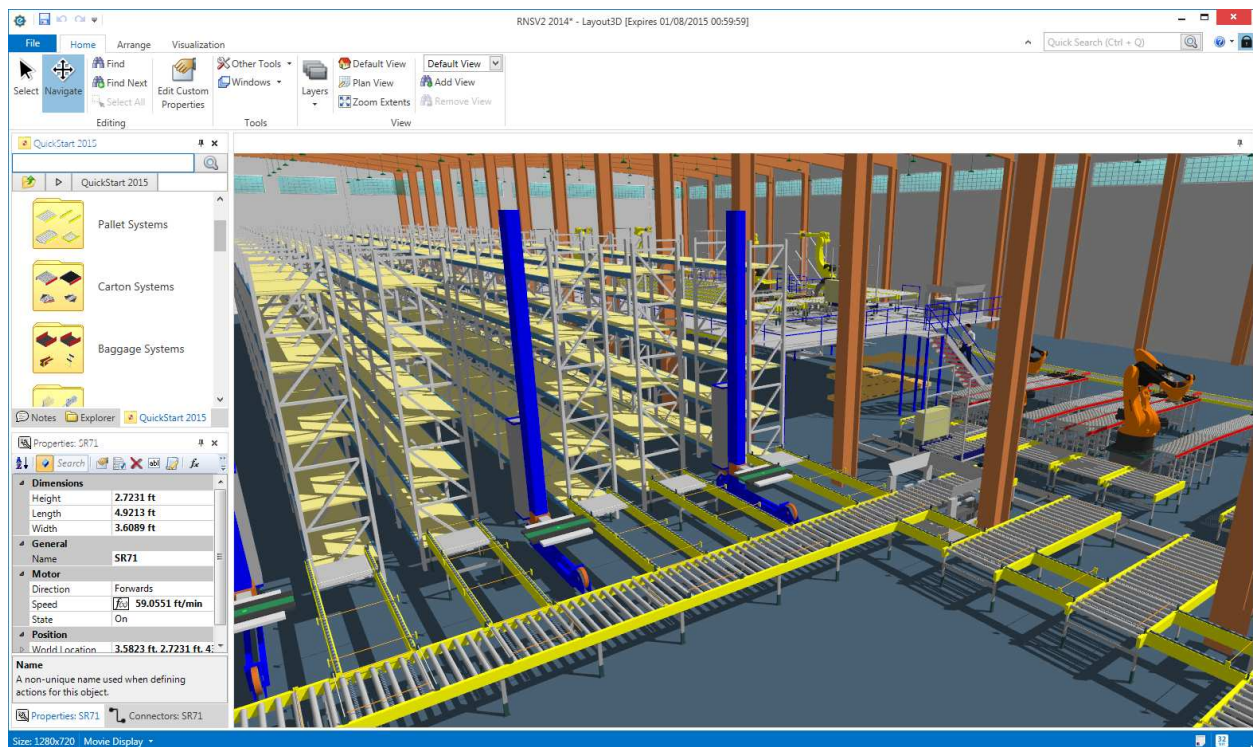

Figure 6: Layout3D features the same basic Emulate3D builder as all Emulate3D products.

Layout3D features the same core model building environment as all Emulate3D products, where equipment is selected from the supplied catalogs and dragged into the model. Many items are parameter-driven, and their dimensions can be numerically changed through the Properties panel or quickly modified by pulling on visual tabs in the model. More advanced equipment like Automatic Storage and Retrieval Systems offer an easy interface where multiple parameters can be defined including the number of shelves, rack dimensions, and aisle width, for example, and the system is generated automatically. Where appropriate, items (such as conveyor sections) will snap together to assist the build process.

Equipment may be grouped and copied multiple times to shorten build time, composite areas may be added to catalogs for later re-use, and there are numerous tools like Align, Distribute, and Offset which also help to get the job done accurately and fast.

Layout3D allows users to import CAD from a variety of sources in order to enrich the layout or to add precision. From a simple 2D DXF floor plan to a comprehensive 3D point cloud, visual reference materials help stakeholders imagine the proposed system more clearly, and minimize the risk of errors. Equipment and visuals may be placed in different layers, and can be made invisible when they are no longer needed.

Layout3D models, in common with all Emulate3D models, can be presented in a variety of ways:

- Navigate the model using a standard Microsoft X-box™ controller
- View it in interactive stereo 3D using Oculus Rift™
- View it on a 3D-enabled NVidia™ laptop
- View it in 3D stereo using Google Cardboard™

Figure 7: The Oculus Rift™ is just one way users can view Emulate3D models in stereo 3D
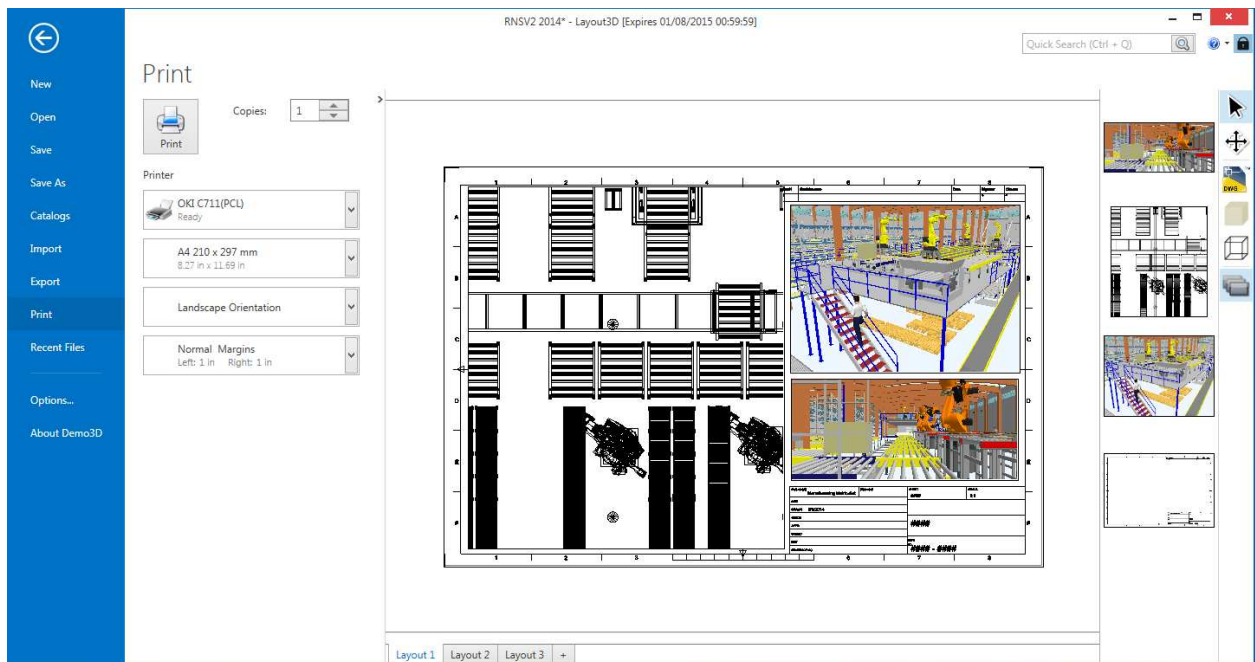

Figure 8: Prepare and print different views

Print and Plot - Create printer-ready multi-view images with the Layout3D Print interface - add a scaled outline, then select and add 2D and 3D wireframe or solid views of the model. Users can navigate within each of these views until they are satisfied with the result.

Straightforward Video Creation - Layout3D also offers a range of video formats, including animated PDF files which can be embedded into PDF documents. Creating videos is straightforward - there's a default value for everything, and taking the extra steps to modify the camera path and video duration is easy.

2.8.2 Demo3D

Demo3D is the range of products designed for presenting solutions to project stakeholders, and for prototyping new systems. Models can be based on QuickStart catalog elements or not, according to the requirements of the project. Demo3D models exhibit realistic load movement based on friction constraints and the notion of gravity. Loads may bump into each other or even fall from conveyors or other handling equipment, and their resulting movement is emergent, as opposed to deterministic. Models not built with QuickStart are constructed with Demo3D primitives and all behavioral logic must be created by the user. Users may also be trained to create custom-made QuickStart catalog elements of specific or non-standard equipment which conform to the QuickStart framework.
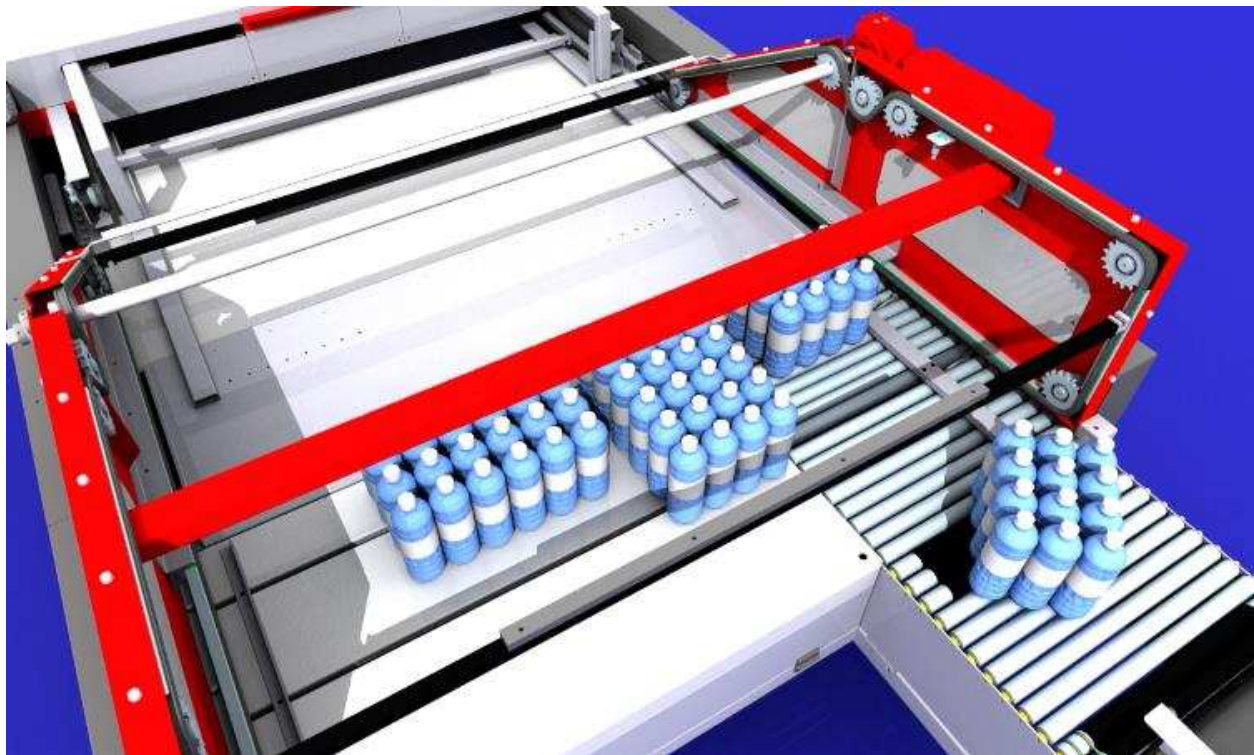

Figure 8: Demo3D's friction-based world creates emergent load movement.

2.8.3   Sim3D

Sim3D models may also be built from QuickStart components, but the primary aim is to generate outcomes as fast as possible in order to simulate, analyze, and improve systems. QuickStart components are designed to operate realistically in both Demo3D and Sim3D modes, but whereas loads and other parts of Demo3D models display emergent behavior, Sim3D models are deterministic and consequently can run considerably faster.

Emulate3D technology includes the notion of three different levels of physics – Volumetric, Planar, and Linear. Volumetric mode is necessary to create "visually realistic" models, with loads bouncing, tipping, and being blocked by each other. Planar models are less computationally demanding, and many Sim3D models are run in Planar mode. Bottling lines of many types and single-file accumulation models may be run in Linear mode, and this is the fastest way to execute a model.

Sim3D features an Experiment and Analysis Manager which allows users to define parameter sets for different model runs to identify robust solutions.

Experiment	Fixed Time	Start Time	Run Duration	Reset	Status	VM1.NumVehiclesEnabled
1	☑	00:05:00	00:10:00	☑	Running	1
2	☑	00:05:00	00:10:00	☑	Ready	2
3	☑	00:05:00	00:10:00	☑	Ready	3
4	☑	00:05:00	00:10:00	☑	Ready	4
5	☑	00:05:00	00:10:00	☑	Ready	5

Figure 9: The Experiment and Analysis utility in Sim3D

2.8.4   Emulate3D Controls Testing

Emulate3D Controls Testing is used to reduce the time and costs associated with control system development and testing by taking much of the task off the project's critical path. Without virtual commissioning control systems can only be tested properly once the real equipment they control is in place, complete with products. As this is inevitably at the end of the project time line, any accumulated lateness presses heavily on the demanding task of verifying the control system. By using a model to replace the real equipment, the controls can be developed and tested in parallel with other parts of the project, and carried out in an office environment rather than in a factory or a warehouse. The result is a more thoroughly tested control system, delivered on time, and at a lower cost.
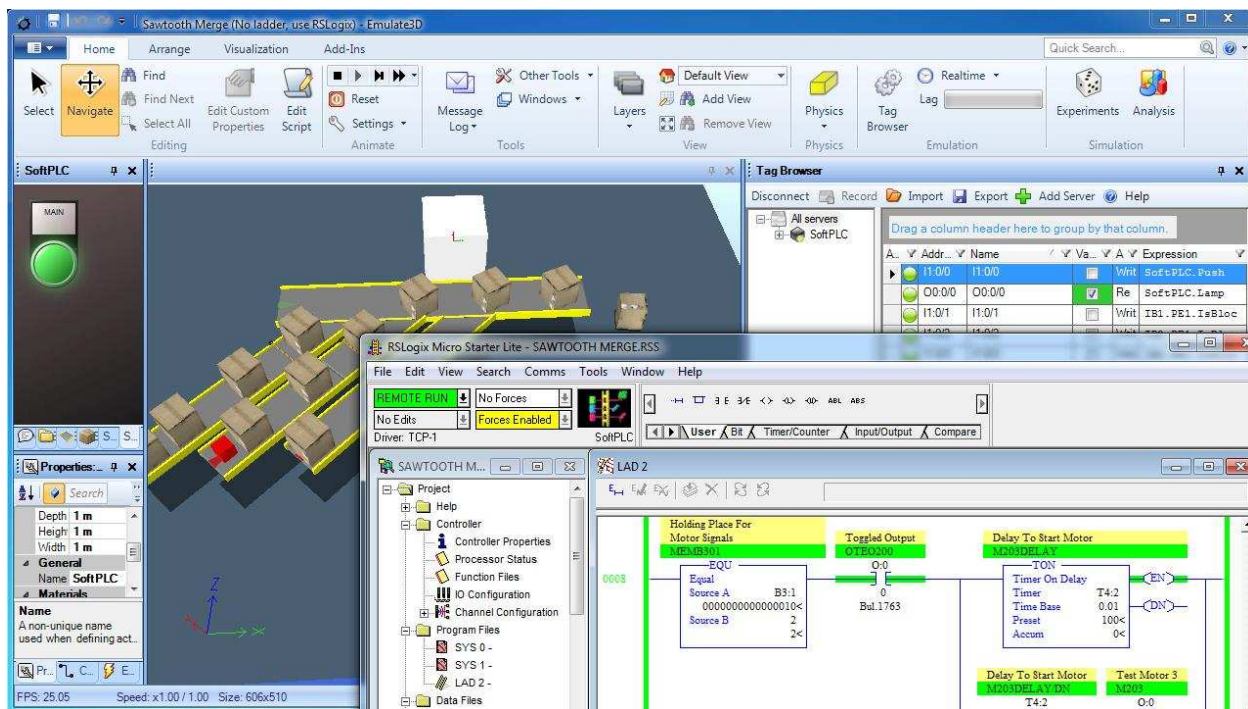
Figure 10: The Emulate3D Controls Testing user interface

As controls testing models replace the actual material handling equipment with their virtual equivalents, models are equipment-centric and load behavior is again emergent, as it is with Demo3D models. Load movement results from the friction caused between the conveyor belt material and the load material, for example. Emulate3D virtual commissioning models are connected to external control systems, usually residing on programmable logic controllers or PLCs. Models may also be connected to any kind of control or data acquisition system using a wide range of methods including SOAP, TCP/IP, UDP, and sockets.

3 WORKING WITHIN THE EMULATE3D FRAMEWORK

Emulate3D products are open and easily extended, and many users are trained in how this is accomplished. Companies wishing to add their equipment to a catalog can import their 3D CAD via a wide range of industry standard formats and use it to skin existing elements, or may create more complex behaviors by adding their own custom operational logic.

Catalogs are separable from the main product and may be used by others who can simply read them into their Demo3D license. Catalogs can also be locked so they may contain elements which can be included in models but not opened, so their internal operation remains protected. A further safeguard is available in the form of a time-limited operation for catalog elements – once past this date, any model containing a time bombed element will no longer work.

3.1 Flexible Modeling Framework and Multiple Control Languages

Emulate3D technology is unique among industrial modeling systems in offering a choice of control methods to suit the different requirements of its broad range of users. An overriding development goal is

to continue to suit the products to the skill sets of the users, and this is reflected in the fact that users can choose between several control languages and scripts.
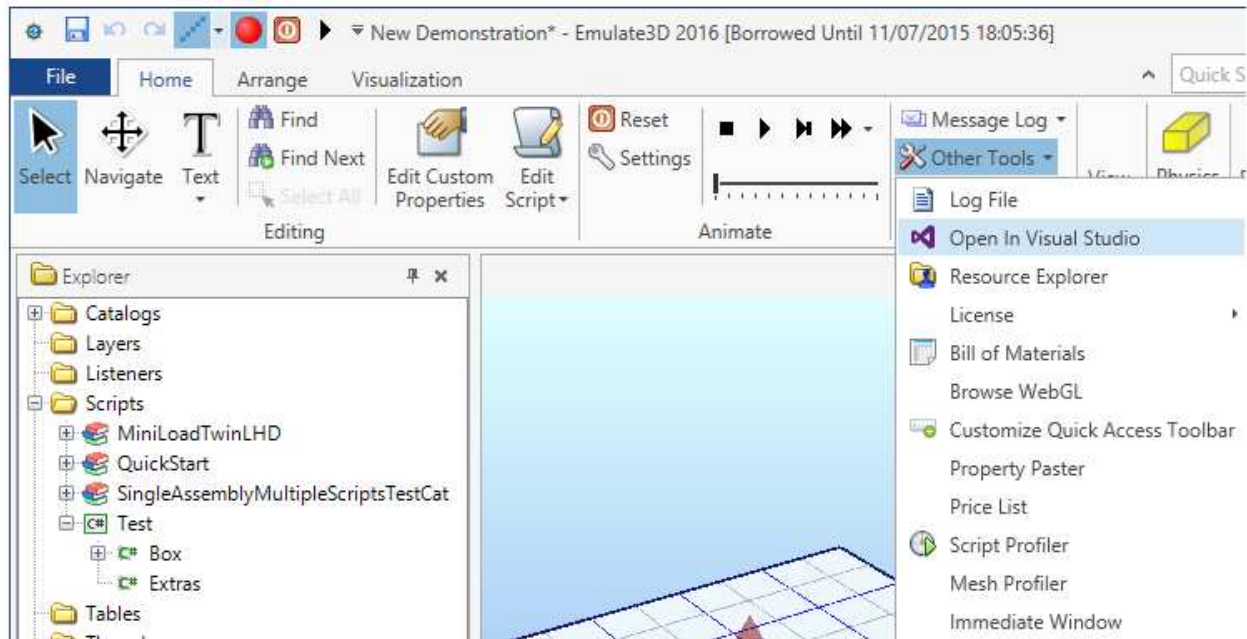

Figure 11: Catalog developers can create C# controlled objects within Visual Studio

3.1.1 Emulate3D Control Methods

Control methods fall into two categories – those which are integrated within the products, and which are therefore operational at the model clock speeds, and those which are external to the products, run in real time, and communicate with the models using OPC, sockets, Modbus, or other product-specific communications.

- C# - with integrated Visual Studio editing (internal to the product)
- Microsoft Jscript (internal to the product)
- QuickLogic drag and drop (internal to the product)
- Ladder Logic (internal or external to the product)
- Instruction List (internal to the product)
- Sequential Function Charts (Grafcet – external to the product)
- Programmable Logic Controllers (external to the product)
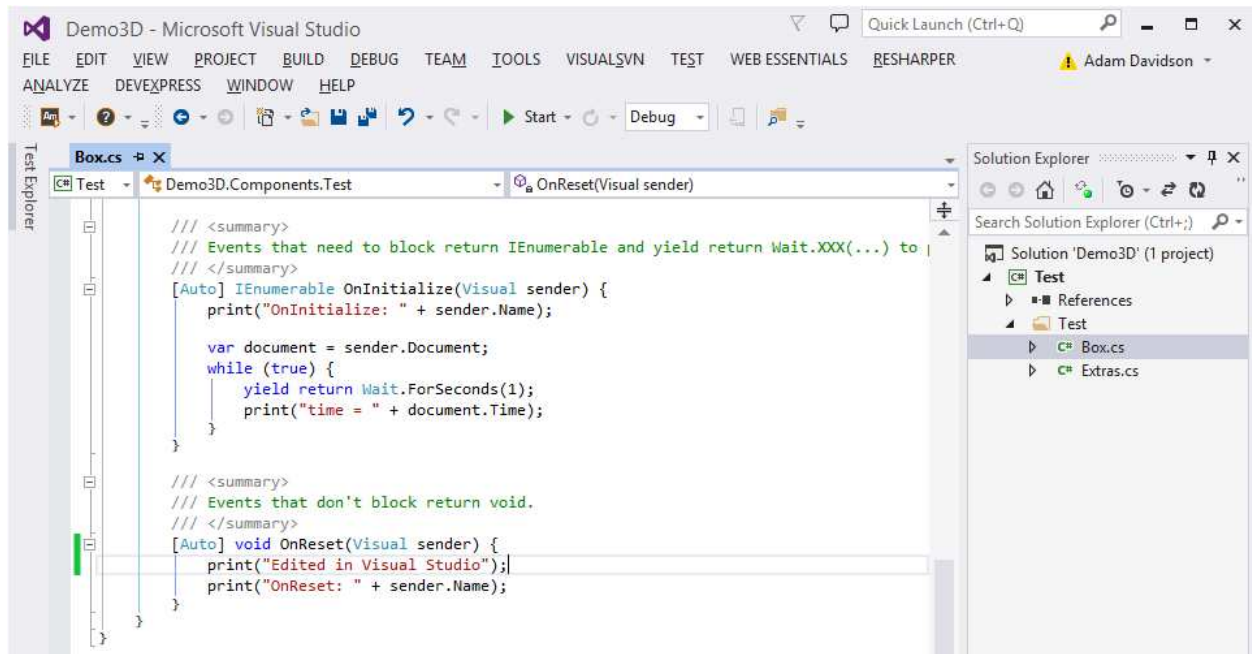- …

Figure 12: C# is the preferred solution for power users

3.1.2 Event Based Behavior

Occasional users are generally most comfortable and productive using the default behaviors of the QuickStart elements, modified using Control Blocks. Specific behaviors or prototype systems can often be conveniently controlled using the drag and drop QuickLogic elements, which allow logic to be expressed without entering into the bureaucracy of a full programming language. QuickLogic encourages a focus on logical operation, as opposed to language structure. Catalog element builders, and those who are most comfortable with a full scripting language can use Microsoft C# for performance, or Jscript, another robust industry standard if they prefer. Trained users can create QuickStart-compatible catalogs of their proprietary equipment, task-specific Control Blocks, or complete systems of their own.
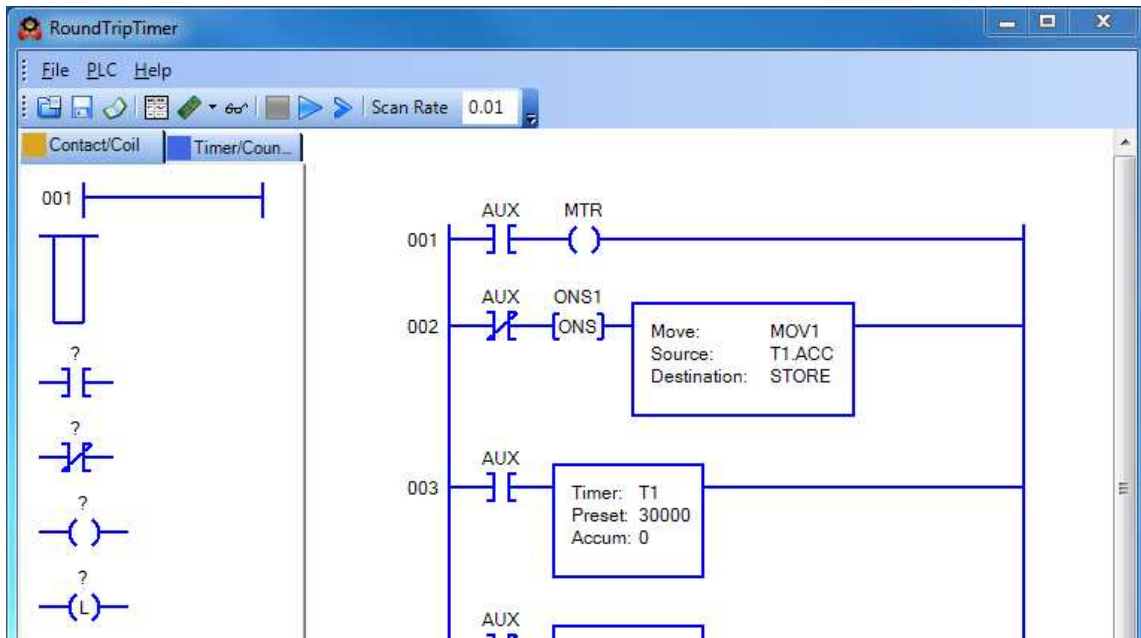
Figure 13: State-based ladder logic is an internal control option.

3.1.3 State Based Behavior

Emulate3D Controls Testing users are not often familiar with creating control in event-based languages like QuickLogic or Microsoft Jscript, and for them the Emulate3D environment offers several state-based control options, including an internal soft PLC which can be operated with Relay Ladder Logic or Instruction Lists. In addition to these two, we also added Sequential Function Charts (SFC – a standardization of Grafcet) as an external stand-alone application connected via the Emulate3D Controls Testing tag browser. These represent three of the five programmable controller programming languages defined by the IEC 1131 standard.
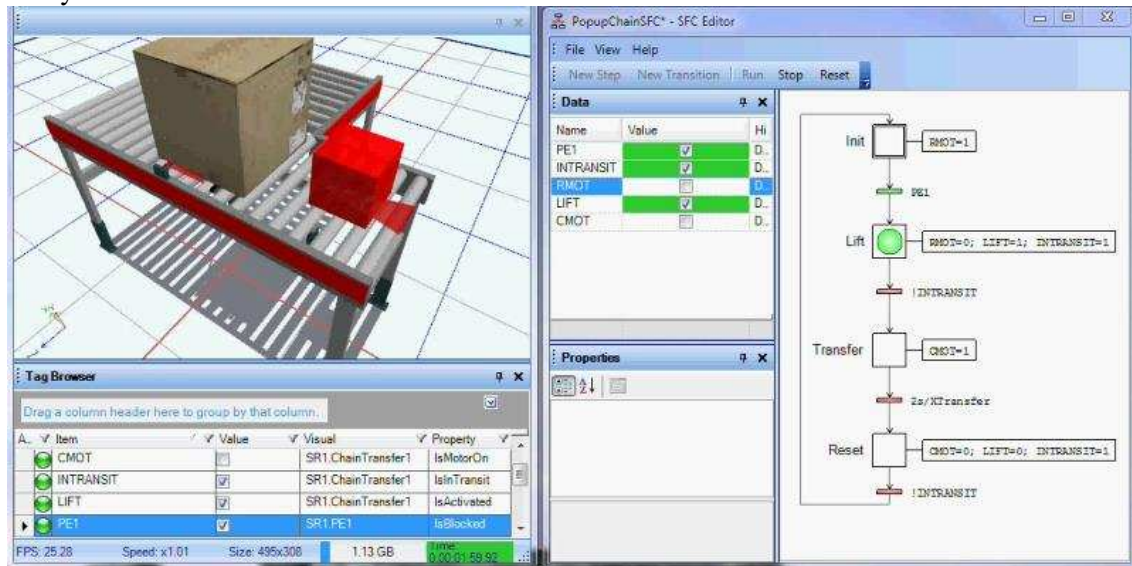

Figure 10: Sequential Function Blocks are an external control option.

4 SUMMARY

Emulate3D technology addresses the requirements of industrial engineers faced with the daunting task of conceiving, developing, and delivering working automation solutions on time, and within budget. The Emulate3D range of products recognizes that there are different sets of dynamic modeling requirements at different stages on the project time line, and that these needs are best met by specific products which may run the same base model. The ability to use the same model under different conditions, and to enhance it with relevant data as the project progresses minimizes errors, and accelerates the workflow.

The use of industrial standards to create an accessible framework facilitates the adoption of the technology within those companies that see the benefit of customizing the product to make it the solution they need, rather than the solution being offered. The wide range of input formats and export options means Emulate3D technology can use existing assets, and produce useable data for other products.

Emulate3D's unique breadth of logic creation offerings makes the products accessible to a wide range of different users. Developers can create control interfaces for occasional users, and many companies have their own catalogs of company-specific equipment.

Emulate3D Ltd develops cost-effective industrial modeling solutions which provide support to engineers working in a wide range of automation industries. Emulate3D continues to invest in the development of new features and functionalities that meet the diverse needs of the growing user base.

REFERENCES

Mueller, G., Using Emulation to Reduce Commissioning Costs on a High Speed Bottling Line, WSC 2001.
Svensson, P. Å., IEC 60848 Ed. 2 Specification language GRAFCET for sequential function charts, IEC Committee Draft.

AUTHOR BIOGRAPHY

IAN W. MCGREGOR is a director and co-founder of Emulate3D Ltd since 2005. Prior to that he worked as Simulation Business Development Manager for AutoSimulations since joining them in 1996, and was posted in Singapore, Japan, and Utah. He has an MSc in Computer Integrated Manufacturing from Cranfield Institute of Technology in England, a Diplome d'Ingenieur from the Universite de Technologie de Compiegne in France, and a BSc in Production Engineering from Kingston Polytechnic in England. He is a Chartered Member of the Institute of Mechanical Engineers and served as Registration Chair for WSC 2001. His email address is `ian.mcgregor@demo3d.com.`