Proceedings of the 2011 Winter Simulation Conference
S. Jain, R.R. Creasey, J. Himmelspach, K.P. White, and M. Fu, eds.

LEAN+ MANUFACTURING PROCESS ANALYSIS SIMULATION (LPAS+)

Michael L. Gregg

The Boeing Company
P.O. Box 516, MC: S306-4130
Saint Louis, MO 63166-0516, USA

Sean M. Van Andel

The Boeing Company
13100 Space Center Blvd, MC: HB3-30
Houston, TX 77059-3556, USA

Steven E. Saylor

The Boeing Company
P.O. Box 3707, MC: 4C-73
Seattle, WA 98124-2207, USA

ABSTRACT

This paper presents an approach for modeling manufacturing process flows using a database-driven simulation design based on commercially available general purpose simulation software. The Lean+ Process Analysis Simulation (LPAS+) incorporates a work flow schedule to model cycle time and resource usage, accounting for task sequencing, task duration variability, resource (labor, tooling, position, etc.) requirements, maximum capacity, and contention. Advantages of the approach include rapid model development, 100 percent reusability, a database driven architecture, the incorporation of macros for automating the population of detailed input tables, and ease of end model use by non-simulation experts. The approach has been used successfully within Boeing to support analysis and cycle time reduction of aircraft and spacecraft production flows and resource requirements analysis including labor and equipment.

1 INTRODUCTION

By developing a general purpose application based on a proven commercially available simulation modeling language, ExtendSim, The Boeing Company has created a capability applicable to both military and commercial systems manufacturing that addresses a wide range of business issues including the ability to verify manufacturing plans, assess manufacturing resource requirements in advance, and perform trade studies associated with production rate, task scheduling, process improvement, resource loading and factory layout. ExtendSim was selected as the basis for development due to its open source coding environment, underlying messaging architecture and its tightly integrated database architecture (Diamond et al. 2010) that combined easily facilitate development of highly scalable database driven simulations. A requirement of the current effort was to develop a standardized reusable modeling toolset that could be easily configured via database imports for standalone operation or (as envisioned for the future) linked to a comparably designed supply chain simulation architecture (Saylor and Dailey 2010) to form the basis for an integrated design, manufacturing, and supply chain analysis toolkit.

2 KEY DESIGN FEATURES

Fundamentally, LPAS+ is a modeling approach for simulating scheduled task sequencing and execution. It is a framework that combines functional process flow model logic with direct database linkage. It's fo-

978-1-4577-2109-0/11/$26.00 ©2011 IEEE

cused for use in analyzing manufacturing process flow, but is general in that it can simulate any business process flow that can be characterized in terms of task sequences, predecessor and successor relationships, task duration and resource allocation. The LPAS+ simulation design shown in Figure 1 evolved out of a need to develop a systems engineering level modeling framework that is reusable across a very broad range of production programs.

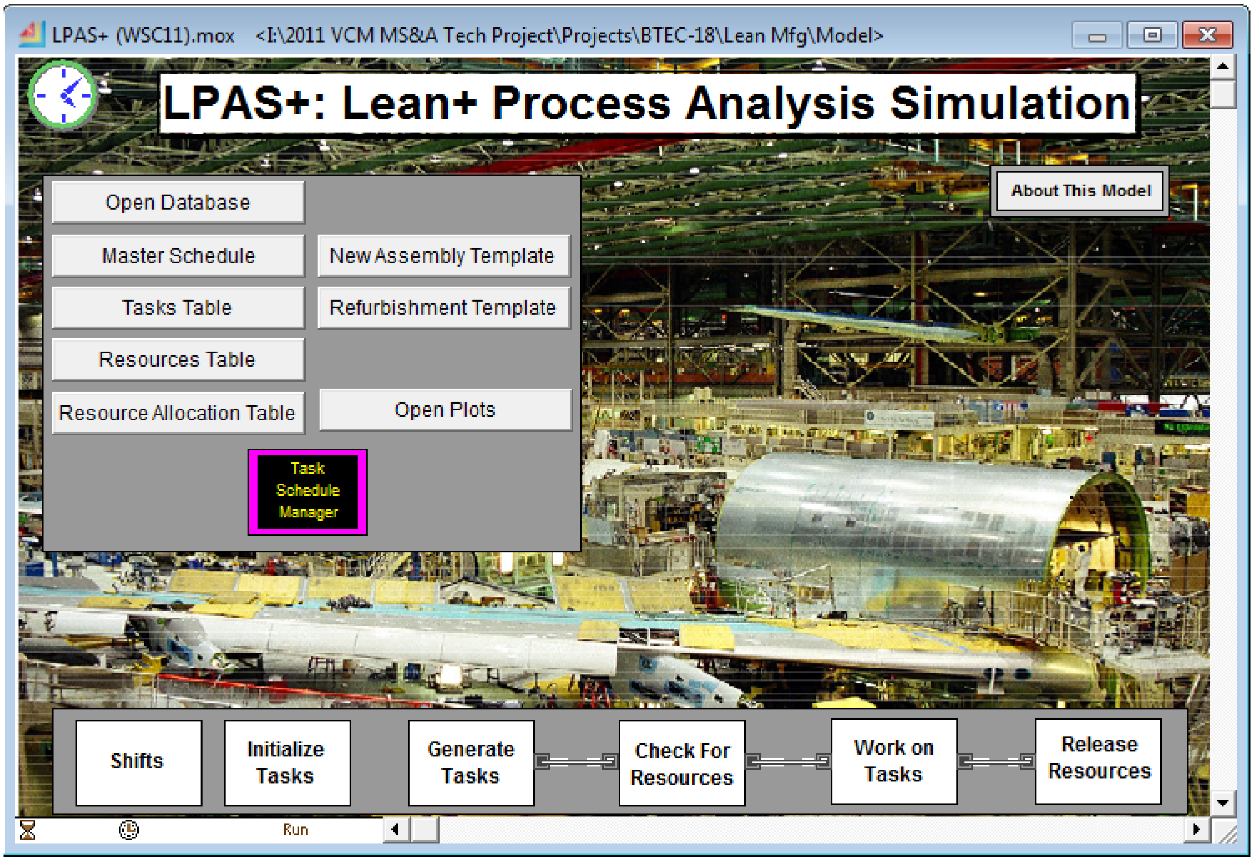


Figure 1: LPAS+ simulation

The principal behind LPAS+ is that the model itself needs no modification to be used from one production program to the next. Rather, the database inputs characterize the production flow being analyzed and the generalized data-centric approach allows the modeler to rapidly reconfigure the model to target a broad spectrum of production and assembly processes. The structure of the data architecture includes template tables that characterize a given set and sequence of tasks that each unit undergoes during a production, manufacturing, assembly, or refurbishment process. A master schedule table defines the scheduled start dates for each unit, or sub-unit by process type. At the start of each simulation run a task schedule manager macro reads the master schedule and correctly sequences each individual unit's set of tasks, be it assembly, refurbishment, or other process type as defined in the template tables. The model then runs the complete set of tasks, applies predecessor and successor task relationships, levies resource constraints, and outputs summary plots and detailed histories of each task activity for analysis.

The design takes advantage of several advanced ExtendSim features including:

- Rapid data access
- Database, table, field and cell to block linking (link alerts)
- Database aware modeling components

- Embedded distributions
- Database address attributes
- Database parent-child field relations

LPAS+ provides the ability to:

- Evaluate and verify process flows, task sequencing, and capacities.
- Identify and help mitigate potential schedule risks.
- Assess constraints such as task workload & cycle time, tooling & labor requirements, shifts, manufacturing day calendars, and system capacities.
- Perform trade studies on alternative manufacturing plans and assembly concepts.
- Evaluate impact of production rates changes.
- Analyze basic costing metrics.
- Provide justification for a recommended execution plan.

Basic Model Outputs include:

- Timestamps when each resource (labor or equipment) was seized and released
- Timestamps when each task began and concluded
- Equipment, Tooling, and Labor Usage Plots (user can modify and add at will)
- Recurring and non-recurring costs
- Summarized multi-run statistics

3 MODELING APPROACH

Boeing's choice to use a commercially available discrete-event simulation environment as the framework for LPAS+, ExtendSim, developed by Imagine That Inc., provided a well crafted set of modeling capabilities; most importantly, the capability to first design the model from a database perspective, organizing all input, runtime, housekeeping, and output data in an organized fashion both logically and hierarchically. Next, the model block structure and logic was designed that then leveraged the capability in ExtendSim to link and interface block structure with the internal database. Combined, these capabilities permitted Boeing to build a generic, reusable modeling approach that is being used internally by military aircraft, commercial aircraft, and spacecraft production programs with minimum to no modification across programs.

The database architecture used to support LPAS+ is shown in Figure 2. It is a compact and organized data structure that contains all information necessary to both set up and run the simulation. As denoted by the arrows linking data fields, parent-child relationships are used as necessary to logically relate child fields containing a common data element type across subordinate fields back to a parent reference in a parent field. Output data from the simulation may be easily exported to data processing tools for additional analysis and data reporting as required.

One of the key advantages of the database approach is the ability to provide direct traceability back to production program maintained data. Rather than creating a completely separate set of input that must be continually cross-checked with systems maintained by business functions, the value stream map and project schedule information including labor and tooling requirements, is downloaded from Production data sources and interfaced directly to the simulation database master schedule and template tables. As shown in Figure 3 under *MS Project Tasks and Manufacturing Capability*, Microsoft Project, shown as an example here, is used to populate the Master Schedule table. A macro custom coded block *Task Schedule Manager,* which runs during simulation initialization, reads the data from the Master Schedule table and Template tables to fully populate the Tasks table, sequencing each individual unit's production task across all units in accordance with the master schedule. The simulation then executes the task table.

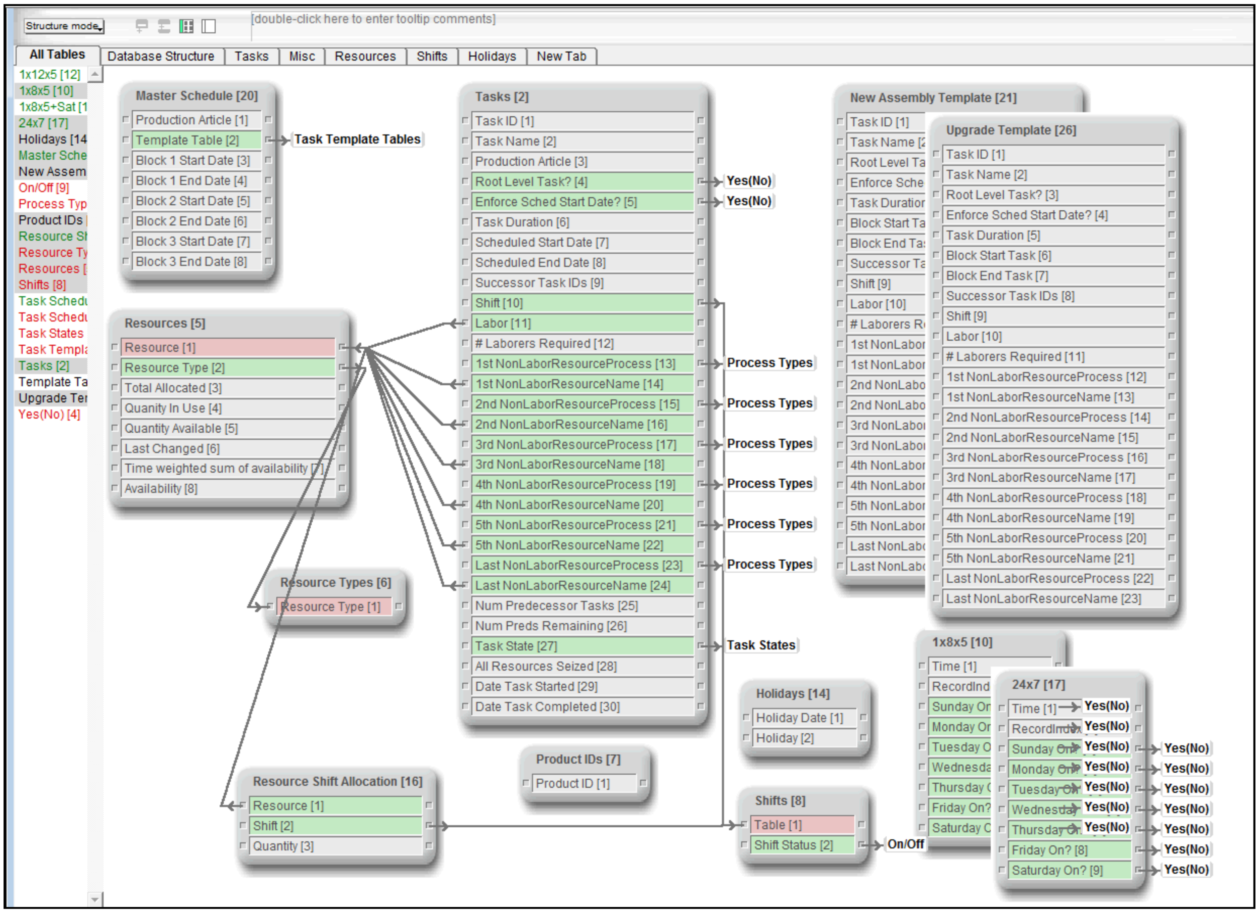


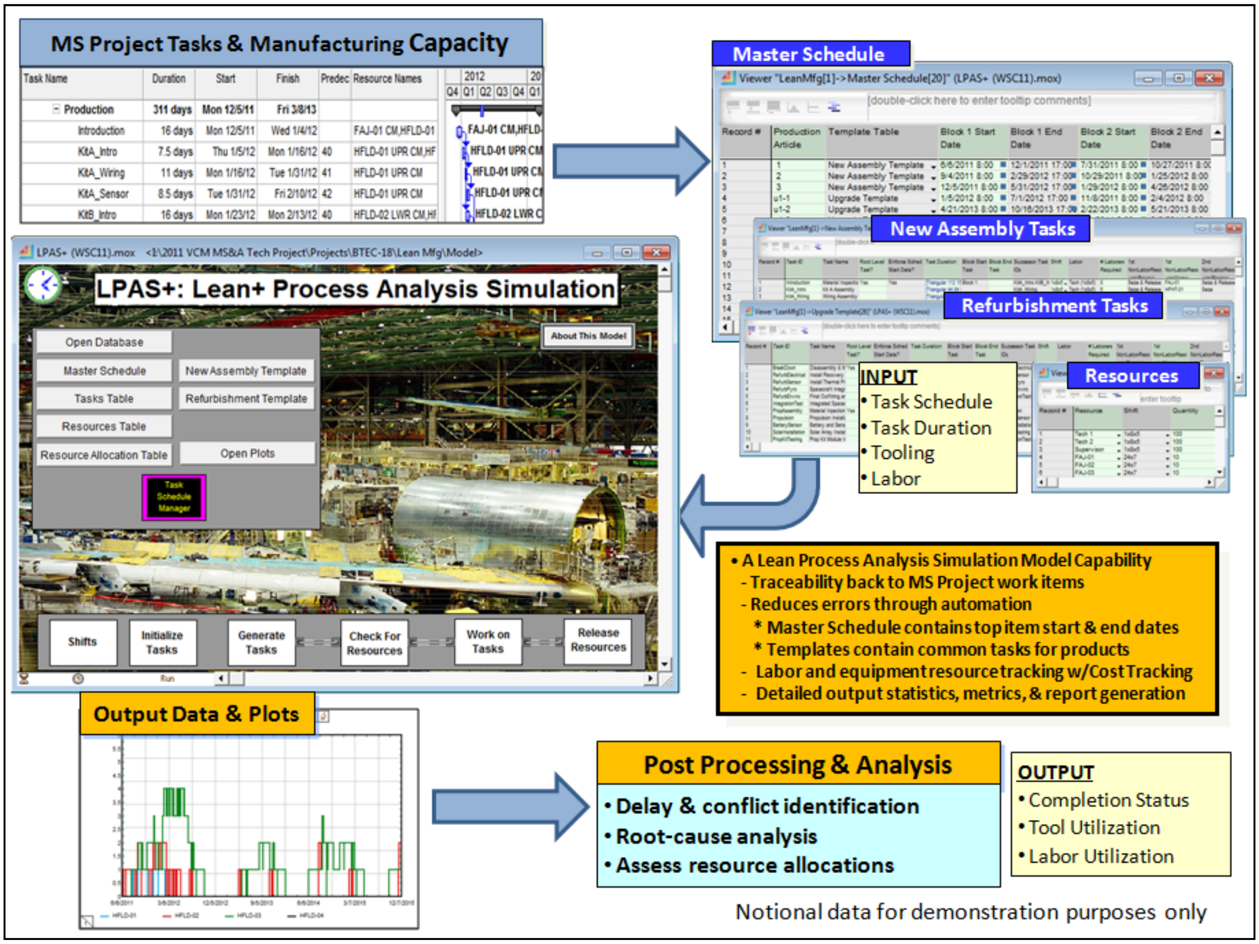
Figure 2: LPAS+ database architecture

As depicted in Figure 3, LPAS+ is a functional process flow model. It's designed to be compact and generic in that any set of tasks or activities that can be characterized in terms of a master schedule and individual unit's sequence of task activities may be simulated. The model itself is comprised of separate modules for the functions of *Generate Tasks, Check for Resources, Work on Tasks*, and *Release Resources*. Two additional modules are used to set up and control resource allocations across shifts and perform initialization functions associated with the Tasks tables.

The modules contain not only many of ExtendSim's standard blocks, but also a few hand crafted custom coded blocks. Many of the blocks contain custom equations written in ExtendSim's "C"-like language, ModL. Driven by the database, rather than data dispersed throughout the model or carried by individual items traveling through the model, this more automated approach to data tracking greatly reduces the potential for data related errors and greatly improves model scalability.

Using ExtendSim's built-in statistical and data tracking mechanisms, LPAS+ creates a set of data plots and data tables. Data such as *Completion Status*, *Tool Utilization*, and *Labor Utilization* can be analyzed by reviewing the output and can be directly exported to external tools such as Excel. Additionally, tools such as Minitab or JMP can be used upfront to perform design of experiments that can then be executed by the simulation.

The set of figures below illustrate details of some of the database tables utilized by LPAS+. The first table, Figure 4, shows an example of a Template table used to specify the set of tasks, and task sequencing associated with a given process. In this example the process is new assembly. Other processes include refurbishment, and/or other activities.   The task template table identifies the sequence in which tasks must be performed, task duration, and the resources required to perform each task. Note that data constructs are present to indicate if a resource is seized, seized and then released, or released. In this way we can model persistent seizing of resources across a sequence of tasks. A Successor Task field is used to indicate which task(s) require as a predecessor the completion of a given task. Template tables may be subdivided into blocks of tasks, wherein the initiating task for a given block is indicated as a root task. In this way, blocks of tasks may be configured with each block having its own schedule constrained start date to model parallel production operations.



MS Project Tasks & Manufacturing Capacity

Task Name	Duration	Start	Finish	Predec	Resource Names
Production	311 days	Mon 12/5/11	Fri 3/8/13		
Introduction	16 days	Mon 12/5/11	Wed 1/4/12		FAJ-01 CM,HFLD-01
KitA_Intro	7.5 days	Thu 1/5/12	Mon 1/16/12	40	HFLD-01 UPR CM,HF
KitA_Wiring	11 days	Mon 1/16/12	Tue 1/31/12	41	HFLD-01 UPR CM
KitA_Sensor	8.5 days	Tue 1/31/12	Fri 2/10/12	42	HFLD-01 UPR CM
KitB_Intro	16 days	Mon 1/23/12	Mon 2/13/12	40	HFLD-02 LWR CM,HF

Master Schedule

Viewer "LeanMfg[1]->Master Schedule[20]" (LPAS+ (WSC11).mox)

[double-click here to enter tooltip comments]

Record #	Production Article	Template Table	Block 1 Start Date	Block 1 End Date	Block 2 Start Date	Block 2 End Date
1	1	New Assembly Template	6/6/2011 8:00	12/1/2011 17:00	7/31/2011 8:00	10/27/2011 8:00
2	2	New Assembly Template	9/4/2011 8:00	2/29/2012 17:00	10/29/2011 8:00	1/25/2012 8:00
3	3	New Assembly Template	12/5/2011 8:00	5/31/2012 17:00	1/29/2012 8:00	4/26/2012 8:00
4	u1-1	Upgrade Template	1/5/2012 8:00	7/1/2012 17:00	11/8/2011 8:00	2/4/2012 8:00
5	u1-2	Upgrade Template	4/21/2013 8:00	10/16/2013 17:00	2/22/2013 8:00	5/21/2013 8:00

New Assembly Tasks

Refurbishment Tasks

Resources

INPUT
- Task Schedule
- Task Duration
- Tooling
- Labor

LPAS+ (WSC11).mox <I:\2011 VCM MS&A Tech Project\Projects\BTEC-18\Lean Mfg\Model>

LPAS+: Lean+ Process Analysis Simulation

- A Lean Process Analysis Simulation Model Capability
  - Traceability back to MS Project work items
  - Reduces errors through automation
    - * Master Schedule contains top item start & end dates
    - * Templates contain common tasks for products
  - Labor and equipment resource tracking w/Cost Tracking
  - Detailed output statistics, metrics, & report generation

Output Data & Plots

Post Processing & Analysis
- Delay & conflict identification
- Root-cause analysis
- Assess resource allocations

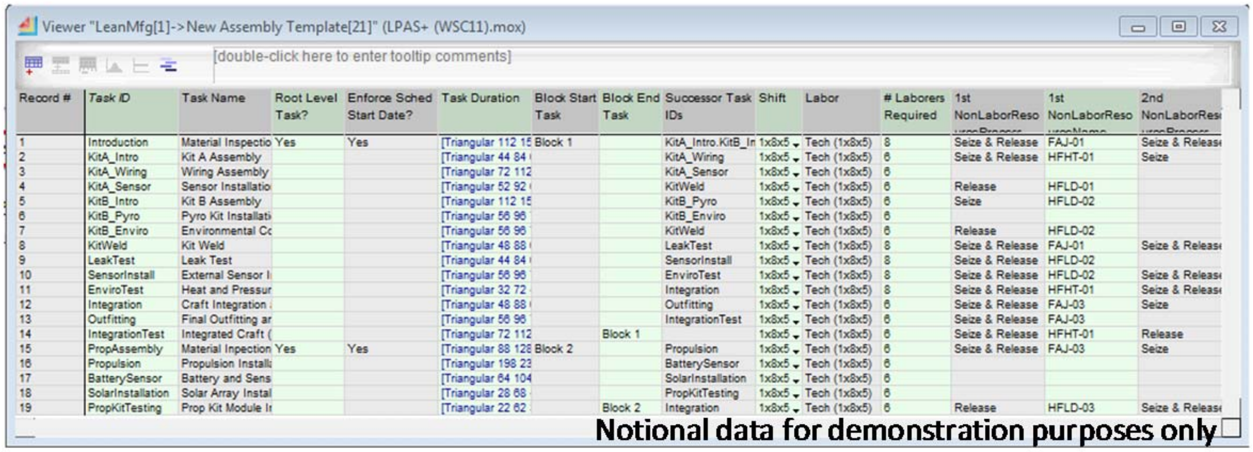
OUTPUT
- Completion Status
- Tool Utilization
- Labor Utilization

Notional data for demonstration purposes only

Figure 3: Analysis approach

The Master Schedule table shown in Figure 5 is used to set up the scenario and indicates the scheduled start dates for each production unit by process type. Each record in the master schedule is associated with a given production unit, and production units may appear multiple times within the table. The field Template Table identifies which of the template tables are associated with the given entry in the master schedule. In this way users can model the evolution of a given unit through production, into operation, and back to refurbishment or system upgrade. This provides a ready means to model the impact of operational usage rate wherein a production article comes back into the factory for major upgrade or overhaul potentially competing with new production resources.

The Master Schedule shown in Figure 5 represents the highest level data construct in the simulation and is used to trigger all succeeding events.



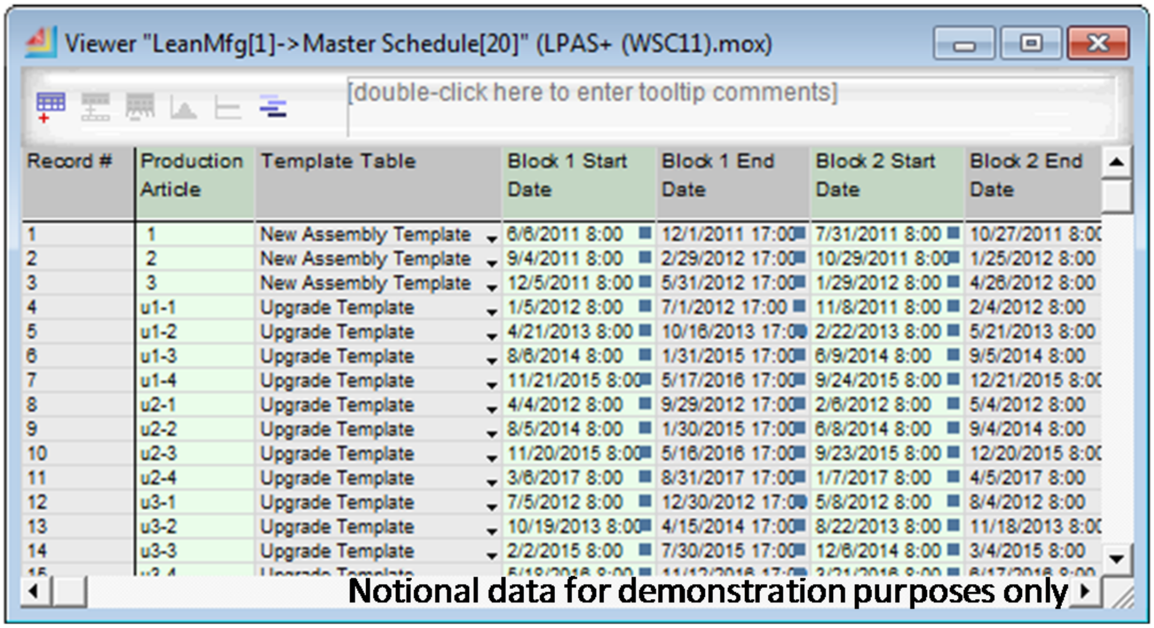
Viewer "LeanMfg[1]->New Assembly Template[21]" (LPAS+ (WSC11).mox)

[double-click here to enter tooltip comments]

Record #	*Task ID*	Task Name	Root Level Task?	Enforce Sched Start Date?	Task Duration	Block Start Task	Block End Task	Successor Task IDs	Shift	Labor	# Laborers Required	1st NonLaborReso	1st NonLaborReso	2nd NonLaborRes
1	Introduction	Material Inspectio	Yes	Yes	[Triangular 112 15	Block 1		KitA_Intro.KitB_In	1x8x5	Tech (1x8x5)	8	Seize & Release	FAJ-01	Seize & Release
2	KitA_Intro	Kit A Assembly			[Triangular 44 84			KitA_Wiring	1x8x5	Tech (1x8x5)	6	Seize & Release	HFHT-01	Seize
3	KitA_Wiring	Wiring Assembly			[Triangular 72 112			KitA_Sensor	1x8x5	Tech (1x8x5)	6			
4	KitA_Sensor	Sensor Installatio			[Triangular 52 92			KitWeld	1x8x5	Tech (1x8x5)	6	Release	HFLD-01	
5	KitB_Intro	Kit B Assembly			[Triangular 112 15			KitB_Pyro	1x8x5	Tech (1x8x5)	6	Seize	HFLD-02	
6	KitB_Pyro	Pyro Kit Installati			[Triangular 56 96			KitB_Enviro	1x8x5	Tech (1x8x5)	6			
7	KitB_Enviro	Environmental Cc			[Triangular 56 96			KitWeld	1x8x5	Tech (1x8x5)	6	Release	HFLD-02	
8	KitWeld	Kit Weld			[Triangular 48 88			LeakTest	1x8x5	Tech (1x8x5)	8	Seize & Release	FAJ-01	Seize & Release
9	LeakTest	Leak Test			[Triangular 44 84			SensorInstall	1x8x5	Tech (1x8x5)	8	Seize & Release	HFLD-02	
10	SensorInstall	External Sensor I			[Triangular 56 96			EnviroTest	1x8x5	Tech (1x8x5)	8	Seize & Release	HFLD-02	Seize & Release
11	EnviroTest	Heat and Pressur			[Triangular 32 72			Integration	1x8x5	Tech (1x8x5)	8	Seize & Release	HFHT-01	Seize & Release
12	Integration	Craft Integration			[Triangular 48 88			Outfitting	1x8x5	Tech (1x8x5)	6	Seize & Release	FAJ-03	Seize
13	Outfitting	Final Outfitting ar			[Triangular 56 96			IntegrationTest	1x8x5	Tech (1x8x5)	6	Seize & Release	FAJ-03	
14	IntegrationTest	Integrated Craft (			[Triangular 72 112		Block 1		1x8x5	Tech (1x8x5)	6	Seize & Release	HFHT-01	Release
15	PropAssembly	Material Inpection	Yes	Yes	[Triangular 88 128	Block 2		Propulsion	1x8x5	Tech (1x8x5)	6	Seize & Release	FAJ-03	Seize
16	Propulsion	Propulsion Install			[Triangular 198 23			BatterySensor	1x8x5	Tech (1x8x5)	6			
17	BatterySensor	Battery and Sens			[Triangular 64 104			SolarInstallation	1x8x5	Tech (1x8x5)	6			
18	SolarInstallation	Solar Array Instal			[Triangular 28 68			PropKitTesting	1x8x5	Tech (1x8x5)	6			
19	PropKitTesting	Prop Kit Module I			[Triangular 22 62		Block 2	Integration	1x8x5	Tech (1x8x5)	6	Release	HFLD-03	Seize & Release

Notional data for demonstration purposes only

Figure 4: Task Template Table



Viewer "LeanMfg[1]->Master Schedule[20]" (LPAS+ (WSC11).mox)

[double-click here to enter tooltip comments]

Record #	Production Article	Template Table	Block 1 Start Date	Block 1 End Date	Block 2 Start Date	Block 2 End Date
1	1	New Assembly Template	6/6/2011 8:00	12/1/2011 17:00	7/31/2011 8:00	10/27/2011 8:0
2	2	New Assembly Template	9/4/2011 8:00	2/29/2012 17:00	10/29/2011 8:00	1/25/2012 8:00
3	3	New Assembly Template	12/5/2011 8:00	5/31/2012 17:00	1/29/2012 8:00	4/26/2012 8:00
4	u1-1	Upgrade Template	1/5/2012 8:00	7/1/2012 17:00	11/8/2011 8:00	2/4/2012 8:00
5	u1-2	Upgrade Template	4/21/2013 8:00	10/16/2013 17:0	2/22/2013 8:00	5/21/2013 8:00
6	u1-3	Upgrade Template	8/6/2014 8:00	1/31/2015 17:00	6/9/2014 8:00	9/5/2014 8:00
7	u1-4	Upgrade Template	11/21/2015 8:00	5/17/2016 17:00	9/24/2015 8:00	12/21/2015 8:0
8	u2-1	Upgrade Template	4/4/2012 8:00	9/29/2012 17:00	2/6/2012 8:00	5/4/2012 8:00
9	u2-2	Upgrade Template	8/5/2014 8:00	1/30/2015 17:00	6/8/2014 8:00	9/4/2014 8:00
10	u2-3	Upgrade Template	11/20/2015 8:00	5/16/2016 17:00	9/23/2015 8:00	12/20/2015 8:0
11	u2-4	Upgrade Template	3/6/2017 8:00	8/31/2017 17:00	1/7/2017 8:00	4/5/2017 8:00
12	u3-1	Upgrade Template	7/5/2012 8:00	12/30/2012 17:0	5/8/2012 8:00	8/4/2012 8:00
13	u3-2	Upgrade Template	10/19/2013 8:00	4/15/2014 17:00	8/22/2013 8:00	11/18/2013 8:0
14	u3-3	Upgrade Template	2/2/2015 8:00	7/30/2015 17:00	12/6/2014 8:00	3/4/2015 8:00

Notional data for demonstration purposes only

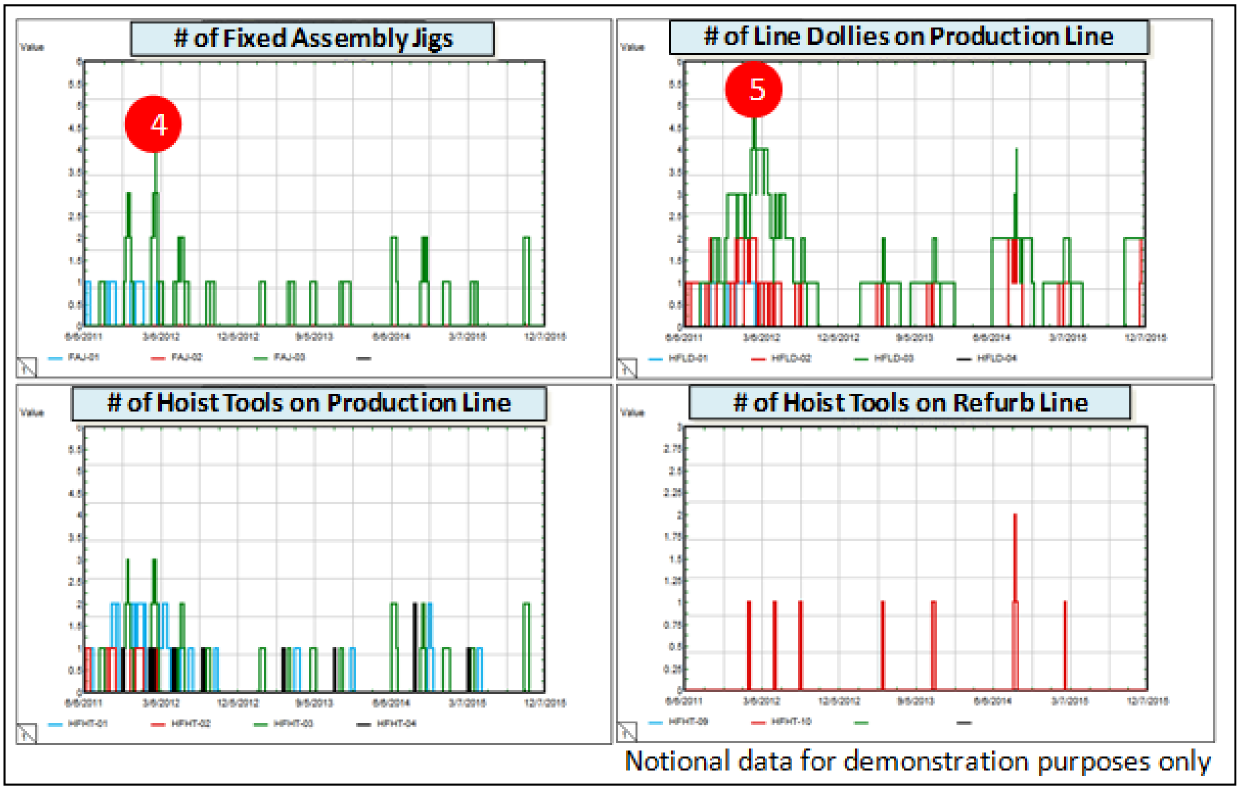
Figure 5: Master Schedule Table

4 USER APPLICATION CASE STUDY

Boeing chose to use the LPAS+ model for a small fixed-priced development program to evaluate production capacity and estimate resource requirements based on a customer-provided delivery schedule. The program consisted of a limited number of production units, each of which would be refurbished/upgraded after use and flown again for a total of ten cycles each. The production schedule was imported into the LPAS+ Master Schedule, and both initial production and refurbishment tasks were baselined in the task template tables. The Task Schedule Generator's auto-build capability populated the entire production and refurbishment task schedule within seconds, including estimated dates and durations for tooling and labor resource usage.

Once the LPAS+ model was validated and debugged, it was turned over to the program for use in the real world. Novice users were able to obtain meaningful output data and charts for presentation to program management. Multiple runs were performed to establish confidence levels that would assist program management in making procurement decisions. After two short weeks of learning the model, the program users were able to perform schedule vs. resource trades that would improve the program's affordability.

Figures 6 and 7 below show LPAS+ output charts for tooling types and quantities required to support production and refurbishment over the entire program schedule. As seen in Figure 6, early in the schedule there was a need for four Fixed Assembly Jigs and five Line Dollies on the production line. Review of the data showed that the Unit 1 refurbishment schedule overlapped with Unit 4 production, therefore requiring use of the same tools for a very limited time. These resources would only be needed for one week to accomplish a single build task, so the expensive tools would either be stored or scrapped after this limited use. Figure 7 shows how a simple adjustment of start and end dates for the refurbishment tasks eliminated the need for the additional tools and saved the program significant expense in non-recurring capital investment.

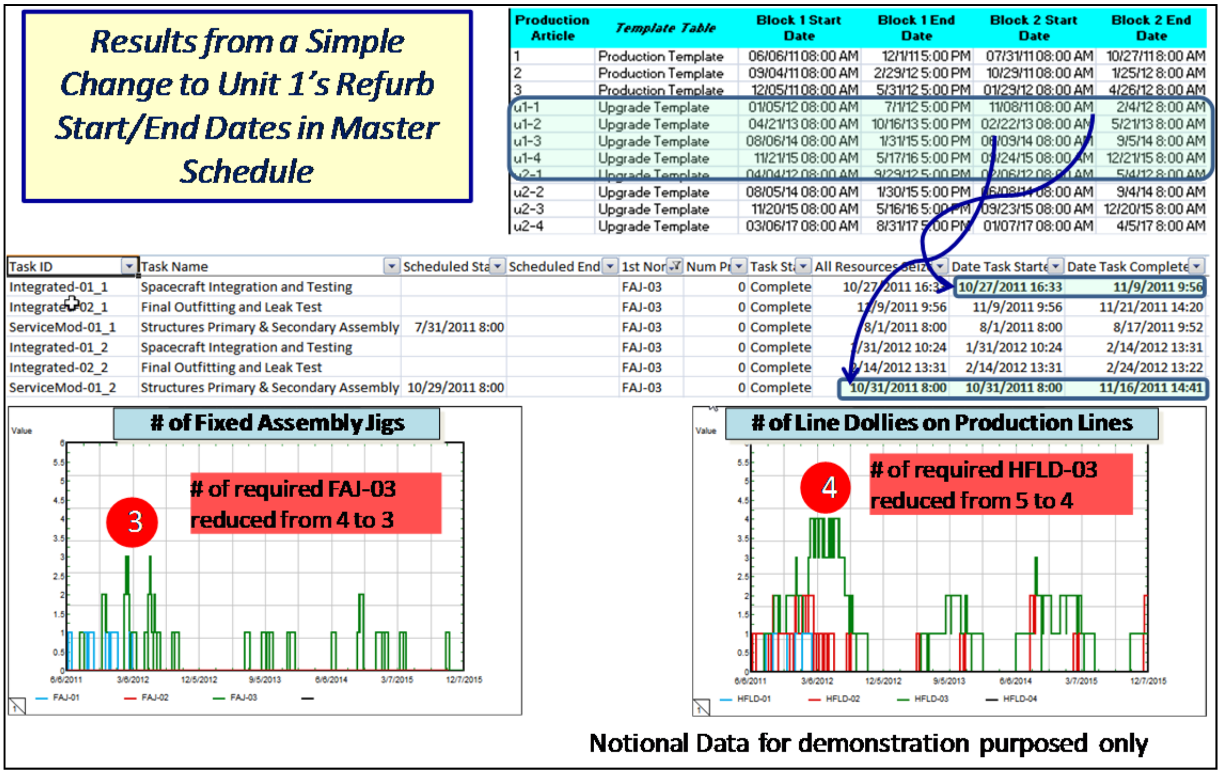


Notional data for demonstration purposes only

Figure 6: Running Model and Analyzing Results

5 MODEL EVOLUTION

As with most simulation projects, there are a number of challenges and a spectrum of disadvantages ranging from incorrect model specification through an incomplete understanding of what data is available. Most stem from program immaturity. A version of the old adage applies, "the results are only as good as the input data." Particularly, for a model of an assembly line to produce a new, untried, cutting-edge product there are additional challenges. Schedules slip, priorities change, and as the design become more mature, some of the early assumptions will have to be abandoned and a new design iteration will need to be dealt with. In situations where the product design is evolving even while construction of the production facility is underway, the simulation model must be flexible enough to be quickly modified to allow product engineers to evaluate rapidly changing production layouts, equipment and labor resource requirements, processing times, and cost assumptions.



Results from a Simple Change to Unit 1's Refurb Start/End Dates in Master Schedule

Production Article	*Template Table*	Block 1 Start Date	Block 1 End Date	Block 2 Start Date	Block 2 End Date
1	Production Template	06/06/11 08:00 AM	12/1/11 5:00 PM	07/31/11 08:00 AM	10/27/11 8:00 AM
2	Production Template	09/04/11 08:00 AM	2/29/12 5:00 PM	10/29/11 08:00 AM	1/25/12 8:00 AM
3	Production Template	12/05/11 08:00 AM	5/31/12 5:00 PM	01/29/12 08:00 AM	4/26/12 8:00 AM
u1-1	Upgrade Template	01/05/12 08:00 AM	7/1/12 5:00 PM	11/08/11 08:00 AM	2/4/12 8:00 AM
u1-2	Upgrade Template	04/21/13 08:00 AM	10/16/13 5:00 PM	02/22/13 08:00 AM	5/21/13 8:00 AM
u1-3	Upgrade Template	08/06/14 08:00 AM	1/31/15 5:00 PM	06/09/14 08:00 AM	9/5/14 8:00 AM
u1-4	Upgrade Template	11/21/15 08:00 AM	5/17/16 5:00 PM	09/24/15 08:00 AM	12/21/15 8:00 AM
u2-1	Upgrade Template	04/04/12 08:00 AM	9/29/12 5:00 PM	02/06/12 08:00 AM	5/4/12 8:00 AM
u2-2	Upgrade Template	08/05/14 08:00 AM	1/30/15 5:00 PM	06/08/14 08:00 AM	9/4/14 8:00 AM
u2-3	Upgrade Template	11/20/15 08:00 AM	5/16/16 5:00 PM	09/23/15 08:00 AM	12/20/15 8:00 AM
u2-4	Upgrade Template	03/06/17 08:00 AM	8/31/17 5:00 PM	01/07/17 08:00 AM	4/5/17 8:00 AM

Task ID	Task Name	Scheduled Sta	Scheduled End	1st Nor	Num Pr	Task St	All Resources Seiz	Date Task Starte	Date Task Complete
Integrated-01_1	Spacecraft Integration and Testing			FAJ-03	0	Complete	10/27/2011 16:33	10/27/2011 16:33	11/9/2011 9:56
Integrated-02_1	Final Outfitting and Leak Test			FAJ-03	0	Complete	11/9/2011 9:56	11/9/2011 9:56	11/21/2011 14:20
ServiceMod-01_1	Structures Primary & Secondary Assembly	7/31/2011 8:00		FAJ-03	0	Complete	8/1/2011 8:00	8/1/2011 8:00	8/17/2011 9:52
Integrated-01_2	Spacecraft Integration and Testing			FAJ-03	0	Complete	1/31/2012 10:24	1/31/2012 10:24	2/14/2012 13:31
Integrated-02_2	Final Outfitting and Leak Test			FAJ-03	0	Complete	2/14/2012 13:31	2/14/2012 13:31	2/24/2012 13:22
ServiceMod-01_2	Structures Primary & Secondary Assembly	10/29/2011 8:00		FAJ-03	0	Complete	10/31/2011 8:00	10/31/2011 8:00	11/16/2011 14:41

Notional Data for demonstration purposed only

Figure7: Schedule trade study – example

Using the model presented in this paper, accounting for these ongoing design changes can be readily handled. The database architecture eliminates the need to dig into the model to find task durations, resources, costs buried somewhere inside. Changes can be accomplished at the user-interface level through modification of the database, for example the task templates, such as those shown in Figures 4 and 5, will need to change (e.g., tasks may need to be added or reordered, times modified, resources added or deleted). However, the simulation model architecture that is being continually enhanced allows the customer to determine the impacts resulting from changes in the design early enough to detect problems and act on them with limited production interruption.

Based on the data presented in this paper, there were two general categories of improvements requested by the customer; 1) evolve model based on ongoing updates to the Value Stream Map and schedule and 2) develop recurring cost analysis capability and methodology. Upon completion of items identified in the first category, the model will:

- Provide needed granularity of task resources and cycle times.
- Allow resource constraints to be clearly identified and process impacts identified (available manpower, tooling, work zones).
- Facilitate higher fidelity simulations and trades (line balancing and optimization, impacts to design or schedule changes).

The second category, develop recurring cost analysis capability and methodology, is intended to allow customer to:

- Assign material and labor cost estimates to each task which will allow the work breakdown structure to be mapped to each new build and refurbishment task.
- Post-process analysis of costs in Excel providing ability to rack and stack costs by unit, year, month, mission, and subsystem.
- Use recurring cost data to baseline refurbishment cost total productive maintenance.

- Use budget constraints in lieu of estimates as program matures; this provides capability to perform numerous "what-if" budget exercises.

These enhancements are planned to aid the customer and allow them to use the model as their program evolves and more information about the product becomes available while maintaining the database structure of the simulation model architecture. Once completed, the customer will be able to perform ad hoc trade studies entirely on their own. There will be no need to modify the simulation model to add a resource or modify a fixed cost buried in a block somewhere deep inside the model. The data that drives the model is visible at the database level and under customer control at all time.

6 SUMMARY

LPAS+ is a general purpose simulation designed to assist in production system development and proposal support. LPAS+ is used to identify production resource usage and line capacity for capital planning purposes. Cost related metrics generated during model run time are used to support recurring cost analysis to identify and manage risk. Its highly reusable architecture significantly reduces model development time and cost and its compact data-driven design approach have proven highly beneficial to non-simulation experts using the model accurately to conduct trades.

Incorporation of the Task Schedule Manager capability allows highly scalable modeling capability and greatly reduces the time required to set up a complete scenario with direct traceability back to qualified data sources. It's been proved to be well-suited to smaller, low-volume production programs and is cost-effective and easy to use.

LPAS+ provides analytical insight into production and refurbishment processes to assist in concept maturation and proposal development.

REFERENCES

Diamond, B., D. Krahl, A. Nastasi, and P. Tag. 2010. "ExtendSim Advanced Technology: Integrated Simulation Database." In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, 32-39. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc. Accessed May 1, 2011. http://www.informs-sim.org/wsc10papers/004.pdf.

Saylor, S., and J. Dailey. 2010. "Advanced Logistics Analysis Capabilities Environment." In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, 32-39. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc. Accessed May 1, 2011. http://www.informs-sim.org/wsc10papers/197.pdf.

AUTHOR BIOGRAPHIES

MICHAEL GREGG is involved in a number of modeling efforts through Boeing's MS&A Tech (Modeling, Simulation and Analysis Technology) department. Currently focusing on enhancing MSAT's Lean+ Process Analysis Simulation (LPAS+) tool, Mike insures that this standard Boeing offering incorporates refinements and additional features requested by users across Boeing. He holds a MS in Chemical Engr. from UC Berkeley and a MS in Systems Engineering from UM Rolla. His email address is michael.l.gregg@boeing.com.

SEAN VAN ANDEL serves as a Support Planning & Management Engineer for a number of programs, within Boeing's Space Exploration division. His primary responsibilities include supporting proposals and leading IRAD projects to ensure that new and legacy Human Spaceflight Systems are sustainable. Currently he is the acting lead of BDS Product Support for the Space Exploration Division. Sean has been a user of ExtendSim for the past 5 years and has led the development of several models in support of new

business. He holds a MS in Operations Management from Embry Riddle Aeronautical University. His email address is sean.m.vanandel@boeing.com.

STEVEN SAYLOR is an Associate Technical Fellow at The Boeing Company within Boeing Research & Technology (BR&T). His area of focus is in the development of modeling and simulation capabilities for use across the Boeing enterprise. He holds a B.S. degree in Aerospace Engineering from San Diego State University. His email address is steven.e.saylor@boeing.com.