

AN ALTERNATIVE APPROACH TO MODELING A PRE-SURGICAL SCREENING CLINIC

Philip Troy

Les Entreprises TroyWare
Cote Saint-Luc, QC H4W 2Z4, CANADA

Nadia Lahrichi

Dept of Mathematical & Industrial Engineering
Polytechnique Montreal
Montreal, QC H3T 1J4, CANADA

Lawrence Rosenberg

Department Of Surgical Services
The Sir Mortimer B. Davis Jewish General Hospital
Montreal, QC H3T 1E2, CANADA

ABSTRACT

Unable to find published material on how to model processes with multiple interacting flows on simulation platforms that use flow-chart like modeling paradigms, we applied Object Oriented Analysis concepts to such a platform. To do so, we identified all of the objects in the model. We then used the platform's queue objects to represent each state of each object, so as to systematize the modeling and make it possible to observe at each moment the number of each type of object in each state. We also modeled inter-object messaging and simulation events via calls to logic procedures, and modeled responses to the messaging and events via the logic in those procedures. The result is a model that was more readily understood, verified and validated, and a modeling approach that can facilitate development of pseudo-object based simulation models on non-object based simulation platforms.

1 INTRODUCTION

While there is a considerable amount of published material on how to use particular discrete event simulation platforms and on how to perform simulation studies (Banks et al. 2010) (Law 2009), there seems to be little published material on how to design, construct, and layout the models built with these platforms to perform those studies. This is particularly problematic for simulation platforms in which modeling is performed using flow chart like objects such as work item generators, queues, decision points, activities and work item sinks. This is because these platforms, when used naively, appear to be limited to flow chart like modeling, which makes them harder to use or inadequate for simulating complex processes, such as those having multiple independent but interacting flows, or entities that need to simultaneously be in multiple queues. This perceived limitation becomes very apparent when attempting to model clinics involving patients that are served by, and thus wait for, nurses, administrative staff, technicians and physicians, all of whom have their own scheduled arrival, break, lunch and departure times, tasks, goals and time changing patient interaction criterion.

To a large extent, the limitations to the use of flow charting for simulation modeling appear very similar to those of using flow charting for process analysis and software design and development. These limitations (Marrer 2009) include the difficulty in modeling complex logic without using multiple sheets of paper, the difficulty in organizing and laying out the chart, the difficulties in translating flow chart logic into a programming language, the need to minimize or eliminate crossed lines (Ackoff 1999), the frequent need

to redraw major portions of the flow chart when making modifications to it, and the difficulty in showing interactions between flows of different items.

As a result, simulation platforms in which modeling is performed using flow chart like objects often include second class constructs such as resources which are seized and released by the work items or other simulation constructs in the particular simulation platform. Unfortunately, these second class constructs generally cannot be managed as flexibly as the primary work flow items, or when they can, they often require more modeling effort to do so.

In the realm of process analysis and software design and development, which in many ways is closely related to simulation modeling, many of the limitations of using flow charting have been addressed by the use of Object Oriented Analysis (OOA). OOA is a systems analysis approach in which the entities, or objects, in a system are described as being members of classes (which often inherit characteristics from other classes). These objects typically have a state that can change over time, have the ability to communicate with other objects using some form of messaging, and have the ability to respond to incoming communications. As such, systems analysis approaches, and potentially simulation platforms, that use an object oriented approach to modeling processes might be expected to use these characteristics, and in particular the communication capabilities, to facilitate the modeling of processes with multiple independent but interacting flows.

In this paper, we discuss the design, construction and layout of a simulation model that required these types of capabilities. This model is of a Pre-Surgical Screening (PSS) clinic at the Sir Mortimer B. Davis Jewish General Hospital in Montreal. The purpose of the model was to help the hospital manage the clinic, which was designed to screen and then prepare patients via training and medication adjustments for the procedures recommended by their surgeons. In addition to needing to be able to determine the number of exam rooms for the clinic, the model also needed to allow analysts to individually schedule the arrival time, break time and lunch time for each nurse, blood taker, ECG technician, and administrative staff member, the arrival time of each physician, and the arrival time of each patient. The goal of this custom scheduling was to provide additional control mechanisms for minimizing a weighted sum of the cost of physician idle times, staff over time, and excessive patient waiting time. The model was developed using Simul8, a simulation platform which appeared at the time to be limited to flow chart like modeling. Because of the impact of the perceived limitation of using flow chart like modeling for complex processes having multiple independent but interacting flows, the authors attempted to and successfully adapted Simul8 to implement a few object oriented capabilities. Using these object oriented capabilities, the authors were able to successfully develop the needed model in a manner that made it easier to verify and validate.

The plan of this paper is as follows. In section 2 we briefly review the literature on simulation studies and modeling with respect to insights as to how to build and layout more complex simulation models like those needed for our problem. In section 3 we describe the specifics of the clinic and of the management goals and decisions that needed to be addressed. In section 4 we discuss the initial model that was built, the challenges faced building that model, how the model was rebuilt, the structure of the rebuilt model, and how that structure facilitated successful development of the needed model. Finally, in section 5 we state our final thought about this work.

2 LITERATURE REVIEW

While limited in detail, Banks et al. (2010) discusses how the “simulation analyst thinks in terms of processes” when working with a simulation platform that “supports a process-interaction approach”. In particular, they mention that the analyst defines the simulation model in terms of entities or objects and their life cycle as they flow through the system, demanding resources and queuing to wait for resources. Unfortunately this discussion is limited in that it does not address modeling of more complex processes in which different types of work items interact with each other in different ways.

In contrast, Graham (2008), while discussing the need for detailed process mapping, shows that Graham Process Mapping can map multiple work flows and their interactions. In essence, Graham Process Mapping

uses horizontal layers of process charts to represent individual flows, and narrow Vs or inverted Vs between the horizontal layers to represent interactions between the different flows. Unfortunately, the only simulation platform we've seen with anything that approaches this capability is ProcessModel.

With respect to building simulation models, Shannon (1998) very briefly discusses his model translation step as having the following model building tasks: "Divide the system into logical subsystems. Define the entities, which will flow through the system. For each subsystem, define the stations (locations where something is done to or for the entities). Define the basic flow patterns of entities through the stations using flow diagrams. Define alternative designs for the system, which are to be considered. Develop flow charts to show the routing logic for flexible paths." Separately, Pidd (1999) presents several useful principles for modeling, and also notes that the modelic process often appears chaotic.

Of particular interest to addressing the need for individual entities to be in multiple queues is the approach suggested by Tocher (1963). In this approach, simulation models that need for entities to simultaneously be in multiple queues create duplicate, or shadow, entities, that are linked together so that when one of them is removed from a queue to be served, the others are also removed from their queues. This approach was implemented by Davies et al. (1993) in a Pascal programming language simulation toolkit whose use was demonstrated by Brailsford and Hilton (2001).

Leung and Lai (1997) build on Yourdon's Structured Methodology to analyze simulation modeling needs and to then use that analysis to build simulation models. To the best of our knowledge, their work is the first to specify an approach to building simulation models from the results of a systems analysis, though they add an extra step by insisting that there be a difference between specifying what the model needs to do and how it does it. Going one step further, as part of his doctoral dissertation, Sonnessa (2004) points out the benefit of an object oriented approach to simulation modeling, and then adapts the Unified Modeling Language to support simulation modeling. He also points out that the implementation of agent based models and business process simulations can be reduced to a generic object-oriented modeling approach, i.e. that the output of an Object Oriented Analysis can be mapped almost directly into a simulation model when using an object oriented simulation platform. Troy and Rosenberg (2009) adapt these ideas in an effort to make it possible to analyze, build and document the simulation model at the same time. Relevant steps from their work are to "specify the system's entities and build them as entity objects, specify the attributes of those entities and build them as fields of the entity objects, specify the system's events and build them as event objects, specify the activities to be performed by entities in response to those events and build them as methods of the entity objects, . . ."

In context of this review, the approach documented in this paper is a partial implementation of object oriented simulation modeling concepts to a non-object oriented simulation platform. In particular, after modeling all entities directly as entities rather than as entities and resources, constructs of the non-object oriented simulation platform are used to represent the number of each entity in each state, and logic is used to emulate the processing that the objects need to perform in response to the events of the model.

3 PRE-SURGICAL SCREENING CLINIC

3.1 The Purpose Of The Clinic

The Sir Mortimer B. Davis Jewish General Hospital is a McGill University affiliated teaching hospital that serves a large and diverse population in Montreal. As such it provides a broad range of inpatient and outpatient services, has a large cancer center, has tertiary and quaternary cardiac, neurosurgery and robotic surgery programs, and performs approximately 14,000 operative procedures per year. To ensure that its surgical patients (PA) are healthy enough to undergo elective surgical procedures, the hospital is in the process of converting its pre-admission testing clinic into a more comprehensive pre-surgical screening clinic. In this new clinic, approximately 35 patients a day will need to do some or all of the following activities: register for the clinic (R1); see a pharmacist or a pharmacy technician to have their medications reconciled (PH); have blood taken (BT); enter an exam room (ER), change into a gown in that exam room,

have their electrocardiogram (ECG) taken in that exam room, be seen by a general practitioner (GP) or internist (IN) in that exam room, get dressed again in that exam room, watch a training video (DVD), receive individual training provided by a nurse (RN), and submit insurance information (R2). Some but not all of these activities have to be done in a particular order, e.g. patients must register before doing any of the other activities, whereas submitting insurance information can be done at any time the patient is not in an exam room or otherwise busy with another activity.

3.2 Management Challenges

In planning for the new clinic, hospital management had to address several challenges including the randomness of time needed for each task for each patient, the different sets of tasks needed by different groups of patients, physicians who were adamantly opposed to being idle even for a minute, limited space for the clinic, a desire to preclude the need for patients to move between rooms while in gowns, the need to minimize or preferably eliminate staff overtime, and the need to minimize patient overflow in the hallways of the clinic as well as patient dissatisfaction with excessively long waits once they arrived to the clinic. To address these challenges, hospital management commissioned the building of a simulation model of clinic daily activities, and the use of that model to evaluate different policies and decisions. These included the number of exam rooms, the scheduled arrival times of each of the 35 patients, the arrival times, break times, lunch times, and departure times of 3 nurses, 1 electrocardiogram technician, 1 pharmacist technician, 2 administrative staff, 1 phlebotomist, and 2 administrative staff, and the arrival times of two general practitioners and one internist. The goal of this process was to minimize a weighted sum of the cost of physician idle time, staff overtime costs, and an estimate of the cost of excessive patient waiting time, which was defined to be the excess over one hour of the total amount of time individual patients waited in the clinic as opposed to being served.

4 THE MODEL

4.1 The Initial Model

In the initial development effort of the model, an attempt was made to build the model using a flow chart like approach, using the Simul8 simulation platform. Thus an effort was made to have patients flow through the model from activity to activity, and to treat physicians, exam rooms, nurses, administrative and technical staff as resources. As can be seen in Figure 1, just modeling patient flows using a flow chart like approach, without even attempting to model the flows of the other entities (or resources) involved in the process, leads to a model which violates the guideline suggested by Ackoff (1999) of not crossing flow chart lines; simplification of the model by pooling queues would have very significantly affected the performance of the model, and the complexity of the logic underlying the model, as it would have necessitated searching queues for the appropriate type of patients to be served.

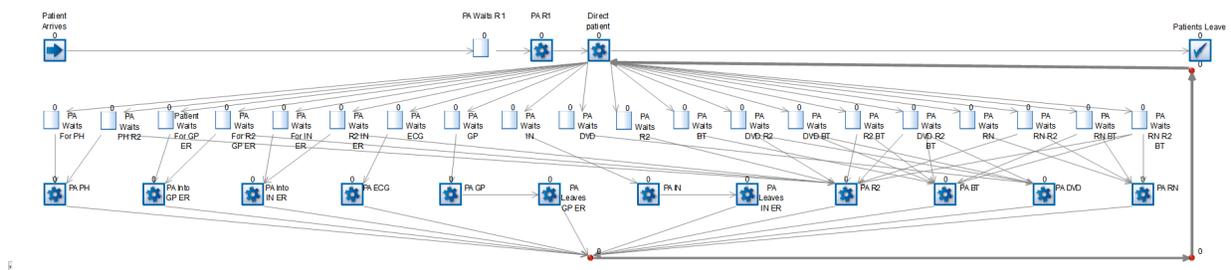


Figure 1: Patient only flow chart like simulation model.

This initial model development effort ran into several additional and more difficult challenges. These challenges included modeling the complexity of individually controlling each resources' arrival, break,

lunch and departure times, particularly when they were affected by the number of other similar resources that were in the clinic at the time, as well as the remaining workload they faced for that day. The ability to individually control each resource’s arrival, break, lunch and departure times was desired so as to meet the project goals of simultaneously minimizing physician idleness, staff overtime and excessive patient waiting by optimizing those times. This was a challenge because to the best of our knowledge, the ability to control the arrival, break, lunch and departure times for each individual resource is generally not available in flow-chart oriented simulation platforms.

4.2 The Revised Model

The revised model was motivated by an attempt to apply object oriented analysis to the modeling effort, so as to obtain some of the benefits of object oriented analysis when dealing with interactions between different types of objects. To do so, we built a list of the different types of objects to be modeled (see Table 1). We also determined the attributes for each type of object which included arrival, break, lunch and departure times for staff, arrival times for physicians, and arrival times and clinic needs for the patients. We further determined the states each type of object could take on (see the cells of each row in Figure 2, which shows a screen shot of the simulation’s animation at a particular instance of time); this included being out of the office, idle, on break, at lunch or performing work activities for each staff member, and waiting for each activity and performing each activity for patients. We then interpreted the different types of events that could occur to each object as being messages to those objects, interpreted those messages as calls to logic subroutines, and interpreted responses to events and messages as the logic in those subroutines.

Table 1: Objects

Abbreviation	Description
PA	Patients
AS	Admissions staff used to register patients (R1) and collect insurance information (R2)
PH	Pharmacists or pharmacy technicians
BT	Blood takers
GP	General practitioners
IN	Internists
ECG	Electrocardiogram technicians
ER	Exam rooms
DVD	DVD players (for watching training videos)

To implement the model on the simulation platform, which did not at the time support object orientation (it has since added some agent based modeling capabilities), we started by creating different types of work items, with the attributes mentioned above, for each object. We then modeled, conceptually and visually, the number of each type of object in each state by using storage bins (queues) for each state. Because the storage bins are animated, as the simulation is run one can see the number of each type of object in each state the object can be in; because this approach to visual modeling does not show the flow between states, it removes the complexity of crossed lines. (See Figure 2.) To model activity completion events we used the built in events in the Simul8 platform for expiry of time in storage bins (which Simul8 conveniently included) to include logic for handling those events; we also took advantage where useful of the storage bin on entry and on exit events, particularly for tracking the time different objects were in each state. For other events we developed logic that could be called by the appropriate objects.

Of particular interest is the approach used to assign resource like objects and patients to interact with each other. By treating both the patients and the resource like objects as objects, we were able to significantly simplify the logic so that when a patient enters a queue it checks the availability of the resource like objects it needs to proceed with any of the activities it currently needs to perform. If they are available, a subroutine is called indicating to the object and the patient that they should start performing the activity together. In

Admission Staff	AS Not In PSS 0	AS In Transition 0	AS Idle 1	AS Idle Needs Break 0	AS Idle Needs Lunch 0	AS Idle Needs To Leave 0	AS In Bathroom 0	AS On Break 0	AS At Lunch 0	AS Register 1 0	AS Register 2 1									
Nurse	RN Not In PSS 0	RN In Transition 0	RN Idle 3	RN Idle Needs Break 0	RN Idle Needs Lunch 0	RN Idle Needs To Leave 0	RN In Bathroom 0	RN On Break 0	RN At Lunch 0	RN Call Patient 0	RN Train Patient 0	RN Train Group 0	RN 1st PA Chart Review 0	RN 2nd PA Chart Review 0						
Pharmacy Technician	PT Not In PSS 0	PT In Transition 0	PT Idle 0	PT Idle Needs Break 0	PT Idle Needs Lunch 0	PT Idle Needs To Leave 0	PT In Bathroom 0	PT On Break 0	PT At Lunch 0	PT Interview Patient 0	PT Patient Follow Up 0									
ECG Technician	ECGT Not In PSS 0	ECGT In Transition 0	ECGT Idle 0	ECGT Idle Needs Break 0	ECGT Idle Needs Lunch 0	ECGT Idle Needs To Leave 0	ECGT In Bathroom 0	ECGT On Break 0	ECGT At Lunch 0	ECGT Taking ECG 1										
Blood Taker	BT Not In PSS 0	BT In Transition 0	BT Idle 0	BT Idle Needs Break 0	BT Idle Needs Lunch 0	BT Idle Needs To Leave 0	BT In Bathroom 0	BT On Break 0	BT At Lunch 0	BT Taking Blood 1										
General Practitioner	GP Not In PSS 0	GP In Transition 0	GP Idle 0	GP Idle Needs Break 0	GP Idle Needs Lunch 0	GP Idle Needs To Leave 0	GP In Bathroom 0	GP On Break 0	GP At Lunch 0	GP Seeing Patient 2										
Internist	IN Not In PSS 1	IN In Transition 0	IN Idle 0	IN Idle Needs Break 0	IN Idle Needs Lunch 0	IN Idle Needs To Leave 0	IN In Bathroom 0	IN On Break 0	IN At Lunch 0	IN Seeing Patient 0										
Changing Room		CR In Transition 0	CR Idle 0							CR In Use 2										
Exam Room		ER In Transition 0	ER Idle 3							ECGT ER In Use 1	GP ER In Use 2	IN ER In Use 0								
DVD Player		DVD Player In Transition 0	DVD Player Idle 4							DVD Player Training 0										
Patient	PA Not In PSS 5	PA Register 1	PA PT 0	PA Into CR And Gown 2	PA Into GP ER And Gown 0	PA Into IN ER And Gown 0	PA Into ECGT ER 0	PA ECGT 1	PA Into GP ER 0	PA GP 2	PA Into IN ER 0	PA Internist 0	PA Out Of Gown 0	PA Into CR And Out Of Gown 0	PA DVD Training 0	PA Group Training 0	PA Individual Training 0	PA Register 2 2	PA Blood Taker 1	PA PSS Process Completed 0
		PA Waits R1 0	PA Waits PT 0	PA Waits CR And Into Gown 0	PA Waits GP ER Gown 0	PA Waits IN ER Gown 0	PA Waits ECGT ER 3	PA Waits ECGT 0	PA Waits GP ER 7	PA Waits GP 0	PA Waits IN ER 0	PA Waits Internist 0	PA Waits CR And Out Of Gown 0	PA Waits DVD BT 0	PA Waits DVT 0	PA Waits GT 0	PA Waits IT 0	PA Waits R2 0	PA Waits BT 0	PA Waits Consult 0
			PA Waits PT R2 3	PA Waits CR And Into Gown R2 3	PA Waits GP ER Gown R2 0	PA Waits IN ER Gown R2 0								PA Waits DVD BT 0	PA Waits DVT R2 0	PA Waits GT BT 1	PA Waits IT BT 0	PA Waits BT R2 0		
														PA Waits DVD BT R2 0	PA Waits GT R2 0	PA Waits IT R2 5	PA Waits IT BT R2 0			
														PA Waits DVD BT R2 0	PA Waits GT BT R2 0	PA Waits IT BT R2 0				

Figure 2: Simulation screen showing the number of objects of each type in each state at a particular time.

the listing immediately below, the subroutine searches for a general practitioner from one of four states, idle (GP Idle), idle needing break (GP Idle Needs Break), idle needing lunch (GP Idle Needs Lunch), or idle needing to leave (GP Idle Needs To Leave). If found, the subroutine changes the general practitioner’s state to seeing the patient, and the patient’s state to being seen by the practitioner. Likewise when a resource like object becomes available, a similar subroutine (not displayed) is called to check appropriate queues for available patients that need that resource like object; if such a patient is found the subroutine changes the patient’s state to show its interaction with the resource object, and the resource object’s state to show its interaction with the patient. Thus all the resource like objects in transition states, and all the patient queue objects on entry logic, have very similar logic; this in turn makes it possible to focus verification efforts on the differences.

```

VL SECTION: PA Waits GP On Entry Logic
    Call _TryToAssignClinicResourceToPatient_MultipleResourceSources taskGPExam,PA Waits GP,PA GP,GP Seeing Patient,4,GP Idle,
        GP Idle Needs Break,GP Idle Needs Lunch, GP Idle Needs To Leave
    
```

4.3 Benefits Of The Revised Model

The revised modeling approach provided several benefits. The primary benefit was that once it was decided to use this approach, the process of creating the model became more straightforward, in that it mostly followed the steps involved in any object oriented analysis of identifying objects, their attributes, the events that occurred to each object, and the activities that objects needed to perform in response to those events. A second benefit of this approach was that the ability to visually identify the exact state of the system with

respect to idle resource objects and waiting patient objects, and to visually identify state transitions, made the model easier to verify and validate.

5 FINAL THOUGHT

One of the reasons for using simulation platforms that use flow chart like modeling paradigms is that they make it easy to build simple simulations. We think that this is both good and not so good, This is good because it makes it easy to introduce simulation to students, and it makes it easy to build simple yet useful models. This is not so good because it can make it harder to build, verify and validate more complex models than it would be when using simulation platforms with object oriented modeling paradigms.

In context of this thought, our sense is that in addition to the work presented in this paper being a stop-gap measure for addressing the limitations of simulation platforms that use flow chart like modeling paradigms, it also has the possibility of pointing the way to extending these platforms to include more object oriented capabilities and thus facilitate building more complex models in a simpler manner.

REFERENCES

- Ackoff, R. 1999. *Re-creating the Corporation: A Design of Organizations for the 21st Century*. Upper Saddle River, New Jersey: Oxford University Press.
- Banks, J., J. S. Carson, B. L. Nelson, and D. M. Nicol. 2010. *Discrete-Event System Simulation*. 5th ed. Upper Saddle River, New Jersey: Prentice-Hall, Inc.
- Brailsford, S., and N. Hilton. 2001. "A comparison of discrete event simulation and system dynamics for modelling health care systems". In *Planning for the Future: Health Service Quality and Emergency Accessibility*, edited by J. Riley, Volume Glasgow. Glasgow Caledonian University.
- Davies, R. M., R. M. O'Keefe, and H. T. O. Davies. 1993. "Simplifying the Modeling of Multiple Activities, Multiple Queuing, and Interruptions: A New Low-Level Data Structure". *ACM Transactions on Modeling and Computer Simulation* 3:332–346.
- Graham, B. (2008). "A Common Ground for Business and Development". Available via <http://www.worksimp.com/articles/detail%20process%20charts%20-%20acommonground.htm> [accessed April 15, 2013].
- Law, A. M. 2009. *Simulation Modeling And Analysis*. 4th ed. New York: McGraw-Hill, Inc.
- Leung, J. W. K., and K. K. Lai. 1997. "A Structured Methodology To Build Discrete-Event Simulation Models". *Asia-Pacific Journal of Operational Research* 14:19–37.
- Marrer, G. 2009. *Fundamentals of Programming*. Available via <http://laptoppress.garymarrer.com> [accessed April 16, 2013].
- Pidd, M. 1999. "Just Modeling Through: A Rough Guide To Modeling". *Interfaces*:118–132.
- Shannon, R. 1998. "Introduction To The Art And Science Of Simulation". In *Proceedings of the 1998 Winter Simulation Conference*, edited by D. Medeiros, E. Watson, J. Carson, and M. Manivannan, 7–14. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Sonnessa, M. 2004. *Modelling and simulation of complex Systems*. Ph.D. thesis, University of Turin, Raleigh, North Carolina. Available via eco83.econ.unito.it/dottorato/michele_sonnessa/sonnessa_thesis.pdf.
- Tocher, K. D. 1963. *The Art of Simulation*. London: English Universities Press.
- Troy, P., and L. Rosenberg. 2009. "A Java Class Library For Simulating Peri-Operative Processes". In *Proceedings of the 2009 Winter Simulation Conference*, edited by M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, 1927–1936. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

Philip Troy is the principal consultant for Les Entreprises TROYWARE, a consulting and software development corporation focusing on health care operations management; as such, he is also part of the Health Care Operations Research team at the Sir Mortimer B. Davis Jewish General Hospital. He received his Ph.D. in Operations Research from Yale University in 1992. His research interests include simulation modeling, simulation modeling platforms, and admission control policies for service organizations. He is a member of INFORMS, SHS and IIE. He is also an Adjunct Professor of Surgery at McGill University, and teaches Quantitative Decision Modeling and Simulation Modeling and Analysis at McGill University's School of Continuing Studies. His email address is phil@philtroy.com and his web site is www.philtroy.com.

Nadia Lahrichi is an Assistant Professor of of Mathematics and Industrial Engineering at Polytechnique Montreal and co-responsible for the health care systems engineering program. She is also part of the Health Care Operations Research team at the Sir Mortimer B. Davis Jewish General Hospital. She holds a Ph.D. in operations research, and her research focuses on the application of operational research tools to healthcare in general and patient flow (scheduling, sequencing ...) and resource optimization (space requirements, nurse scheduling ...) in particular. She is a member of the Society for Health Systems (Society of the Industrial Engineering Institute) and the INFORMS Health Applications Society and her email address is nadia.lahrichi@polymtl.ca.

Lawrence Rosenberg is a Professor of Surgery and Medicine and Director of the Division of Surgical Research at McGill University, A.G. Thompson Chair of Surgical Research at the McGill University Health Center, and Chief of Surgical Services at the Sir Mortimer B. Davis Jewish General Hospital. He obtained his M.D.,C.M. From McGill University in 1979, trained in General Surgery at McGill, completed his surgical residency and Ph.D. in Experimental Surgery in the area of pancreatic cell differentiation in 1985, and undertook post-doctoral studies and fellowship training in transplantation surgery at the University of Michigan. His email address is lawrence.rosenberg@mcgill.ca.