

An Evaluation of the World-Views for the Grab-and-Glue Framework

Tillal Eldabi, Man Wai Lee and Ray J. Paul

Centre for Applied Simulation Modelling (CASM), Department of Information Systems and Computing, Brunel University, Uxbridge, UK

Creating simulation models based on the world-views such as the three-phase approach, the event-based approach, the activity-based approach and the process-based approach has been used as a standard practice for several decades. However, it is still not known whether they are suitable to be adopted or not when creating a simulation model by following the Grab-and-Glue framework. Hence, this paper aims at evaluating the suitability of accessing these world-views when building a simulation model based on the Grab-and-Glue framework. After an evaluation is conducted based on the proposed criteria, it is discovered that none of the world-views are suitable to be used. Hence, it is recommended that a new world-view is necessary to be developed for structuring the simulation model which is assembled based on the Grab-and-Glue framework.

Keywords: Grab-and-Glue, three-phase approach, event-based approach, activity-based approach, process-based approach.

1. Introduction

A programming source code or the software packages can be used as a tool for creating a discrete event simulation model. However, most simulation modellers prefer to use software packages rather than writing the source code from scratch, although it is still suffering from the problem of low flexibility and expensive software packages [1]. Eldabi et al. [1] aim at reducing these disadvantages by proposing to create a simulation model by using the Grab-and-Glue framework which was originally introduced by Paul [2]. The basic idea of this framework is to grab different scissions from the Web and to glue them together to form a simulation model (see Figure 1). “Scissions” is defined as “the act of the cutting or serving into

the divisions or fissions”. In computer science, this term is defined as “a piece of something for model construction” [3]. Scissions can be a piece of source code which is written in any programming languages, a component, or an object etc. After assembling a simulation model by using the grabbed scissions, it will be executed and tested immediately. If the problem owner satisfies the output result, life moves on; otherwise, the irrelevant scissions will be rejected and the new scissions will be grabbed and glued to the relevant positions. This process will be iterated until the output result can satisfy the problem owner’s requirement.

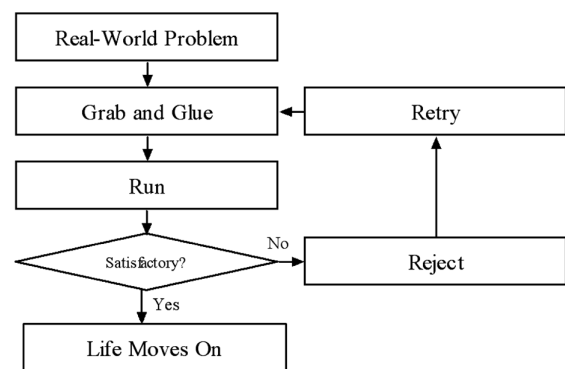


Fig. 1. Grab-and-Glue Framework.

Four widely used world-views have been introduced since the 1960s for building discrete event simulation models, which are the three-phase approach, the event-based approach, the activity-based approach, and the process-based approach [4, 5, 6]. The world-views allow a simulation modeller to structure a simulation model by using the programming source code.

However, they are not necessarily suitable because of the complexity of the nature of different systems [7]. Due to the fact that the idea of Grab-and-Glue is different from the classical simulation modelling method, it is still not sure whether the world-views are suitable to be followed for creating a Grab-and-Glue-based simulation model. Hence, this paper proposes some criteria which are based on the characteristics of the Grab-and-Glue framework. After that, the world-views will be evaluated based on the proposed criteria.

This paper is structured as follows. Section 2 proposes a number of criteria for selecting a suitable world-view for assembling a simulation model by following the Grab-and-Glue framework. Section 3 is the description of the world-views, and Section 4 is an evaluation of the reviewed world-views, based on the designed criteria in Section 2. Section 5 presents the conclusions of this paper and the information about future works.

2. Expected Criteria for Selecting a Suitable World-Views

The Grab-and-Glue framework aims at assembling a simulation model by using the existing scissions. Hence, the created model will have the characteristic of high flexibility [1]. Because of the limitation of the current simulation package, recent research suggests that the programming source code is preferable to be used for model development [3].

A number of world-views have already been developed since 1960s [4]. The expected criteria for selecting a suitable world-view for developing a Grab-and-Glue simulation model are created and shown as follows:

1. Time Issue

This criterion is the most important factor for deciding whether a framework is suitable to be adopted or not. The term “Time” mentioned here is the simulated time. “Time” is an important factor because it determines the state of different events at different periods of time. Hence, this factor cannot be ignored, even though a simulation model is formed by following the Grab-and-Glue framework – assembling scissions together. If the “Time” fails to work as

expected, the usability of the created simulation model will be low.

2. Simplicity

This criterion is important because it affects the time required for model development. If the implementation of an approach is difficult, the required time of model development will be longer, and hence, simulation modeller may be reluctant to adopt it because of the time-consuming problem. The Grab-and-Glue framework aims at creating a simulation model in a fast speed. If the implementation of an approach is easy, the model development time will be shorter and the required resources will be lower, which is the purpose of using the Grab-and-Glue framework.

3. Flexibility

This criterion is also an important factor because it affects the usability of the constructed model directly. The term “flexibility” is defined as “able to change”. If a model has low or even no flexibility, it will become useless if the problem owner thinks that the output result is not what he/she wants. On the other hand, if the model has high flexibility, the irrelevant parts of the model can be changed easily to satisfy the problem owner’s requirement, which is one of the characteristics of the Grab-and-Glue framework.

4. Model Running Speed

This criterion needs to be considered because it affects the running speed of the model. If the executive speed of a simulation model is slow, the time required for model testing will be high, and hence the required resources such as the cost of model development will be increased. Grab-and-Glue is aimed at developing simulation models in a faster speed and relatively lower cost when compared with creating a simulation model which is based on the classical simulation modelling framework. A model with slow running speed may lead to the requirement of high resources, which is against the idea of the Grab-and-Glue framework.

5. Suitability

This criterion is the summary of the above designed criteria. After evaluating different

world-views by using the above criteria, this factor will be used to determine whether it is suitable to be adopted for the Grab-and-Glue framework or not.

3. Review of Existing World-Views

After designing the required criteria for selecting a suitable world-view for model development by following the Grab-and-Glue framework, this section presents four most widely used world-views, which are the three-phase approach, the event-based approach, the activity-based approach and the process-based approach, coupled with the advantages and disadvantages of each of them.

3.1. Description of Existing World-Views

The Three-Phase Approach

The introduction of the *three-phase* approach for building a discrete event simulation model was invented by Tocher in 1963 [4, 8, 9]. It is assumed that there are two events in this approach, namely B-event (Bs) and C-event (Cs) [4, 6]. According to Pidd [4], the definition of an event is “a state change that occurs at an instance of time”. The Bs is the bound event or the book-keeping event. It will be executed whenever the scheduled time arrives. The Cs is the conditional or the co-operative event. It will be executed if the specific conditions within the simulation are satisfied [6, 7]. The specific condition here indicates that the activity cannot be scheduled in advance, because it can only be executed if the required resources or the state of other simulation entities are ready [10]. Figure 2 presents the framework of the three-phase executive. The “three” in the three-phase approach implies the number of phases which are executed in each cycle of the simulation [11]. The first phase is the “A phase”, which is the time scan for finding out the due time of the next event and determining which Bs are then to occur [6]. The simulation clock will then be advanced to that particular time [12]. The second phase is the “B phase”. All the Bs which are scheduled in advance in the due list identified at the A phase will be executed [6, 7, 8]. If the execution of the Bs needs to wait until a C

event is ready, this phase will not be executed, instead, this executive will be put into a queue. The third phase is “C phase”, which will test each of the Cs in turn and execute those conditions which are satisfied [6]. The execution of this phase will be repeated until no more Cs can be started [4, 6, 7]. The sequence of the execution depends on the location of the Cs inside the simulation model, as a result, a higher priority of Cs will be executed first [8]. The characteristic of this approach is that it contains a C phase which can prevent the deadlock when running the simulation model. Even though the event’s execution time is due, it cannot be started until the C event is ready.

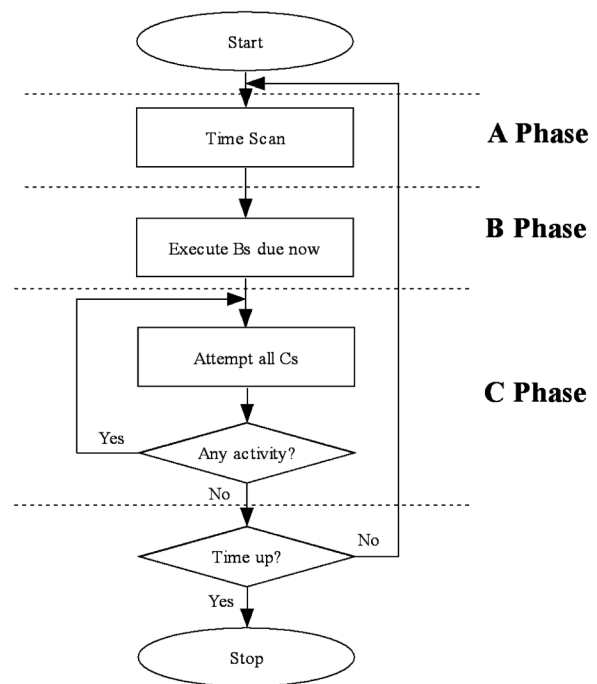


Fig. 2. The Three-Phase Executive [4].

The Event-Based Approach

The *event-based* approach was the most commonly used simulation modelling approach from 1960 to 1980 because it was embedded within one of the most widely used simulation programming languages, SIMSCRIPT, during that period of time [4, 7]. An event-based model is considered as event routines, and its execution has only two phases [4]. According to Pidd [4], an event routine is defined as “a set of statements, in some programming language, which

capture the entire set of logical consequences that can flow from an event". Figure 3 presents the framework of an event-based executive. A time scan of finding out the next due event from the event calendar will be started, and the simulation clock will be moved to that time, as well as all the events that were scheduled to be executed at that time will be moved to the current events list. The simulation clock will be kept constant, and the finishing time of the current activities, coupled with the time for the new activity's arrival time, will be scheduled. In this approach, simulation clock will be kept moving between the start of the scheduling time for the scheduled event and the end of scheduling time of that event until the simulation is finished [4].

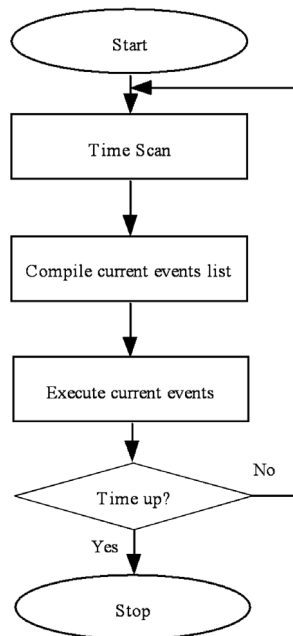


Fig. 3. The Event-Based Executive [4].

The Activity-Based Approach

The *activity-based* approach originally came from the three-phase approach. The characteristic of this approach is that it is not necessary to maintain an event calendar. Instead, it must be ensured that all the executives can be detected during the simulation time period [4]. Also, it is not required to identify which activity is due at next simulation clock time [7]. The only consideration is that the scheduled activities must be started and stopped when the simulation clock

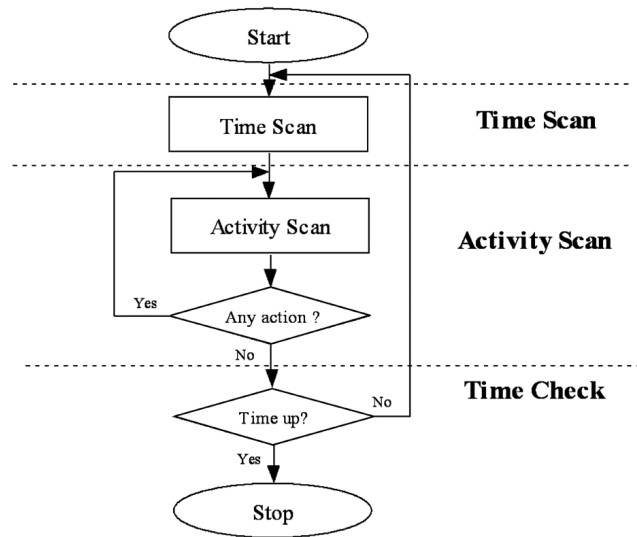


Fig. 4. The Activity-Based Executive (Adapted from Pidd [4]).

reaches the activities' starting or stopping time [4]. Figure 4 presents the framework of the activity-based executive. The starting time for the next event will be checked and the simulation clock will be moved to that particular time. All the activities will be scanned in turn (every time the clock moves) to find out whether any activity is due. The scanning process will be repeated until no more activities need to be executed. These processes will be iterated until the simulation is over [4, 7].

The Process-Based Approach

The *process-based* approach firstly appeared in SIMULA [4, 13], and it was the most frequently used approach. According to Ellision and Tunnicliffe Wilson [5], a process is defined as "a collection of happenings detailing the history of an entity as it progresses through the simulation". A process-based executive needs to know the whereabouts of each entity in its process and needs to have some control on the starting and the ending of the entity's movement during its process [4]. Figure 5 shows the framework of a process-based executive. A future events scan will be executed first for determining the time for the next event, and the simulation clock is advanced to that new time. The next step is to move those events from the future lists whose re-activation time equals the new clock time to

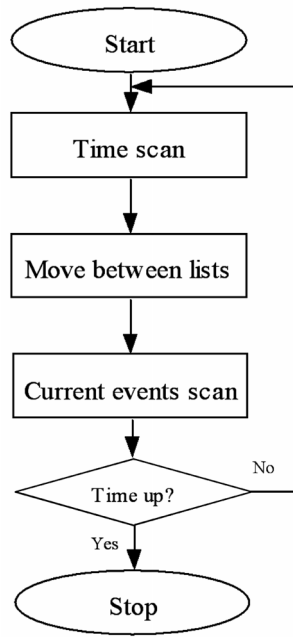


Fig. 5. The Process-Based Executive [4].

the current events list. After that, the executives will manage those entities on the current event lists which are scheduled later, either to complete or to halt, which depends on the conditional or the unconditional delay. The unconditional delay occurs when the progress of an entity is halted for a time period which can be determined in advance. The entity will be restarted once the simulation time has passed. On the other hand, the conditional delay occurs when an entity’s movement through its process

is halted until the specific conditions in the simulation are satisfied. Entity needs to remain at this delay point with a fixed condition until it is instructed to move on. If it is an unconditional delay, the records will be moved to the future events list, and the next re-activation will be set [4].

3.2. Comparison of the World-Views

In the above subsection, different world-views for building a simulation model have been presented. Based on each of their characteristics, this subsection discusses the advantages and disadvantages of each of them.

Table 1 shows a comparison of the advantages and disadvantages of different world-views. The three-phase approach contains a C event, which helps to prevent the deadlock of the simulation model. However, it is more complex when compared with the event-based approach or the activity-based approach, because both of them contain two phases only. The execution of the event-based approach is simpler than the three-phase approach, because it is only concerned with the scheduling event routines and has no sequencing tasks to perform. The execution speed is also faster than the model which is developed by following the three-phase approach, because the logical consequences are held in the appropriate event routines. Also, it is not necessary to perform a full scan of all the Cs events

	Advantages	Disadvantages
Three-phase approach	<ul style="list-style-type: none"> • Contains Cs events which prevent deadlock 	<ul style="list-style-type: none"> • More complex than event-based or activity-based approach
Event-based approach	<ul style="list-style-type: none"> • Simpler than three-phase approach • Execution speed faster than three-phase approach 	<ul style="list-style-type: none"> • Difficult to construct model in complex system
Activity-based approach	<ul style="list-style-type: none"> • Simplicity 	<ul style="list-style-type: none"> • Low execution speed
Process-based approach	<ul style="list-style-type: none"> • Faster running speed compare with activity-based approach • Construction of initial program simpler than event-based approach 	<ul style="list-style-type: none"> • Difficult to ensure that no deadlock occurs in the complex system

Table 1. Comparison of different world-views.

at every event which is required in the three-phase approach. However, it may be difficult to ensure that all possible outcomes are accounted for within the event routines in the complicated system. A simulation model which is built by following the activity-based approach has the advantage of simplicity, because it is not necessary to maintain an event calendar.

However, the speed of execution is slower than running a model which is developed by following the event-based approach, because it needs to scan all the activities in each event repeatedly, even though it is not due at that time. Creating a simulation model following the process-based approach has the advantage of fast running speed when compared with a model based on the activity-based approach; also the construction of initial program is simpler when compared with the event-based approach, because it is not necessary to think through all possible logical consequences of an event [4].

4. Evaluation of Existing World-Views with Respect to Grab-and-Glue

4.1. What Grab-and-Glue Wants?

The expected characteristic of a simulation model which is assembled based on the Grab-and-Glue framework is fitness for purpose, high flexibility, and low model development time [1]. The development of a simulation model is to help the decision maker to make decisions,

or to help the problem owner to get a better understanding of his/her problem [14]. As a result, it may not be worth spending a large amount of resources to form a simulation model which may not satisfy the requirements from the problem owner. Therefore, the selected world-view should allow the modeller to build a model which is fast and easy, and with high flexibility for changing the irrelevant scissions.

4.2. Evaluation of Existing World-Views

As mentioned in Section 2, the timing issue, simplicity, flexibility and model running speed are the criteria for selecting a suitable modelling world-view. It is not a necessity for a simulation modeller to select an appropriate world-view when building a simulation model by using a software package, because the simulation packages have already made the decision itself. However, the development of a simulation model following the Grab-and-Glue framework depends on using code as a tool [3]. Hence, choosing a suitable world-view becomes an important issue.

The design of the world-views is to help simulation modellers to structure simulation models, which allows them to write the required programming code [7]. An evaluation on different existing world-views has been conducted, based on the designed criteria in Section 2 (See Table 2).

	Three phase approach	Event-based approach	Activity-based approach	Process-based approach
Time Issue	Simulation clock moves at A phase, and keeps constant until the execution of due events is finished	Simulation clock moves between activities starting and ending points	Simulation clock moves with an increment	Simulation clock moves by scanning events in future events list
Simplicity	Difficult	Moderate	Easy	Difficult
Flexibility	Low	Low	High	Low
Speed	Fast	Fast	Slow	Fast
Suitability	No	No	No	No

Table 2. Comparison between different modelling approaches.

Time Issue

In the three-phase approach, “time scan” is executed at the A phase (see A phase in Figure 2). The due-time for the next event is then determined, and the simulation clock moves to that particular time. After that, simulation clock becomes constant until the execution of the due events is finished (B phase and C phase in Figure 2). Although this approach can be used for model development, it is not suitable for creating a simulation model by following the Grab-and-Glue framework, because the model will be affected easily, once any scission is changed.

In the event-based approach, the starting time of the first event is scanned and the simulation clock advances to that particular time. The ending time of the first event and the starting time of the second event will be scheduled. Simulation clock will move to the ending time of the first event. The characteristic of the “time” in the event-based approach is that the simulation clock keeps moving between an event’s starting time and its ending time until the simulation is finished.

In the activity-based approach, the simulation clock will keep increasing by one time unit. After an increment of time, all the activities will be scanned again to find out whether any event is due to happen or is finished (see activity scan part in Figure 4). The increment of time and the scanning process will be repeated until the simulation is over (Time check in Figure 4).

In the process-based approach, the first event is scanned (Time scan in Figure 5). The simulation clock moves to that time and finds out any event that is due to happen and moves that event to current event list (Move between lists in Figure 5). The simulation clock will stop and the execution of the executives will be performed (Current event scan in Figure 5). If a delay occurs, the executive will be moved back to the future-events list. The simulation clock will then scan for the next event. This process will be iterated until the simulation process is finished (Time up in Figure 5).

All the proposed world-views can be used for model development, however, not all of them are suitable to be used for assembling a Grab-and-Glue simulation model. If a model is created by using the three-phase approach, the

event-based approach or the process based approach, the structure and hence the “time” of the model may be affected if a scission is added or deleted. A model created by following the activity-based approach may not have this problem, because it scans all the activities of the model before any execution.

Simplicity

Having a C phase in the three-phase approach helps to prevent the deadlock of the execution of a complex model, however, the execution of this approach is difficult because it contains three phases in each executive cycle. The event-based approach and the activity-based approach are much simpler than the three-phase approach, because both of them contain two phases only. However, the event-based approach is difficult to be used when the simulation modeller is dealing with a complex simulation model. The process-based approach is also difficult to be used because it contains two event lists. The process-based approach is difficult to be used, because events in the future event list will be moved to the current event list once the due time has arrived and it will be moved back to the future event list if the activity needs further process [4].

Flexibility

Building a simulation model by using the three-phase, the event-based or the process-based approach have low flexibility because the development of the model is in sequential way and its execution consists of the whole model, starting by scanning the model and finding out whether an event is due to start or not. It is difficult to modify the model if any event is required to change. Flexibility of the activity-based approach is relatively high, because the maintenance of an event calendar for the execution is not necessary [4].

Speed

The time required to execute a simulation model which is created based on the activity-based approach is extremely long if compared with a

model which is built by following the three-phase, the event-based, or the process-based approach [4]. The main reason is that the execution in the activity-based approach needs to keep scanning all the activities to find out if any activity is due to happen or finish, even though they are not ready to be executed.

Suitability

After comparing all the widely used world-views, it is discovered that all of them are not suitable to be adopted for developing a simulation model by following the Grab-and-Glue framework, although all of them can be used for model development. The main reason is either not simple enough to be used, low flexibility or low execution speed.

5. Recommendations

This evaluation shows that the widely-used world-views are not suitable to be used for assembling a simulation model based on the Grab-and-Glue framework. One of the reasons is because the simulation time of the created model will be affected if one of the scission changes. Apart from that, other reasons are that the existing world-views are too difficult to be used, low flexibility, or low model running speed.

Because of these, a world-view should be developed based on the characteristics of Grab-and-Glue framework, coupled with the disadvantages of the existing world-views. One proposed method is to structure a simulation model by gluing all the scissions together. However, the problem of “time” should not be affected if any of the scissions is changed. The world-view should also be easy to use with a running speed that enables fast development and testing of the model.

6. Conclusion and Future Works

This paper evaluates four widely used world-views by using five proposed criteria, which are designed based on the characteristics of the Grab-and-Glue framework. After conducting an evaluation on the world-views, it has been

discovered that all of them are not suitable to be used for developing a simulation model by following the Grab-and-Glue framework, although all of them can be used for model development. The main reasons are the time problem and the simplicity problem. Because of this, it is recommended that a new world-view should be developed based on the characteristics of Grab-and-Glue, as well as the disadvantages of the existing world-views.

In the future, a suitable world-view which aims at structuring a Grab-and-Glue simulation model will be introduced, as presented in Section 5. The proposed world-view should have sufficient flexibility to enable simulation model modifications to be timely, whilst retaining ease of use.

References

- [1] T. ELDABI, M. W. LEE, R. J. PAUL, A Framework for Business Process Simulation: The Grab and Glue Approach, in *Proceedings 15th European Simulation Symposium: Simulation in Industry* (A. Verbraeck, V. Hlupic Ed.), (2003), Society for Computer Simulation International, Delft, pp. 291–296.
- [2] R. J. PAUL, The Internet: An End to Classical Decision Modelling? In *Internet Management Issues: A Global Perspective* (J.D. Haynes, Ed.), (2002), Idea Group Publishing and Information Science Publishing, Hershey, pp. 209–219.
- [3] T. ELDABI, M. W. LEE, R. J. PAUL, Examining the Feasibility of Constructing Simulation Models Using the Web-Based ‘Grab-and-Glue’ Framework, in *Proceedings of the 2004 Winter Simulation Conference* (R. G. Ingalls, M. D. Rossetti, J. S. Smith, B. A. Peters, Ed.), (2004) Association for Computer Machinery, New York, pp. 1494–1501.
- [4] M. PIDD, *Computer Simulation in Management Science*, John Wiley & Sons, Chichester, (1998).
- [5] D. ELLISION, J. C. TUNNICLIFFE WILSON, *How to Write Simulations Using Microcomputers*, Mc-Graw-Hill Book Company Limited, London, (1984).
- [6] R. J. PAUL, D. BALMER, *Simulation Modelling*, Brunel University, Uxbridge, (1998).
- [7] C. HIRATA, J. KRAMER, An Investigation of Time Warp, *Simulation Practice and Theory*, 5 (1997), pp. 515–533.
- [8] R. A. CASSEL, M. PIDD, Distributed Discrete, Event Simulation Using the Three-Phase Approach and Java, *Simulation Practice and Theory*, 8 (2001), pp. 491–507.

- [9] P. R. HARPER, A. K. SHAHANI, A Decision Support System for the Care of HIV and AIDS Patients in India, *European Journal of Operational Research*, 147 (2003), pp. 187–197.
- [10] M. PIDD, R. A. CASSEL, Using Java to Develop Discrete Event Simulations, *Journal of the Operational Research Society*, 51(4) (2000), pp. 405–412.
- [11] H. I. ODHABI, R. J. PAUL, R. D. MACREDIE, Making Simulation More Accessible in Manufacturing Systems through a ‘Four Phase’ Approach, in *Proceedings of the 1998 Winter Simulation Conference* (D. J. Medeiros, E. F. Watson, J. S. Carson, M. S. Manivannan Ed.), (1998), pp. 1069–1075, Association for Computer Machinery, New York, available online via <http://www.informs-cs.org/wsc98papers/146.PDF> [accessed June 19, 2004].
- [12] M. PIDD, R. A. CASSEL, Three Phase Simulation In Java, in *Proceedings of the 1998 Winter Simulation Conference* (D. J. Medeiros, E. F. Watson, J. S. Carson, M. S. Manivannan Ed.) (1998), pp. 367–371, Association for Computer Machinery, New York, available online via <http://www.informs-cs.org/wsc98papers/050.PDF> [accessed June 19, 2004].
- [13] O. DAHL, K. NYGAARD, SIMULA – An ALGOL-Based Simulation Language, *Communication of the ACM*, 9(9) (1966), pp. 671–678.
- [14] R. J. PAUL, S. J. E. TAYLOR, What Use is Model Reuse: Is There a Crook at the End of the Rainbow? In *Proceedings of the 2002 Winter Simulation Conference* (E. Yücesan, C. H. Chen, J. L. Snowdon, J. M. Charnes Ed.), (2002), pp. 648–652, Association for Computer Machinery, New York, available online via <http://www.informs-cs.org/wsc02papers/084.PDF> [accessed January 26, 2003].

Received: July, 2004
 Revised: August, 2004
 Accepted: August, 2004

Contact address:

Tillal Eldabi
 Centre for Applied Simulation Modelling (CASM)
 Department of Information Systems and Computing
 Brunel University
 Uxbridge
 Middlesex, UB8 3PH
 UK
 e-mail: tillal.eldabi@brunel.ac.uk

Man Wai Lee
 Centre for Applied Simulation Modelling (CASM)
 Department of Information Systems and Computing
 Brunel University
 Uxbridge
 Middlesex, UB8 3PH
 UK
 e-mail: manwai.lee@brunel.ac.uk

Ray J. Paul
 Centre for Applied Simulation Modelling (CASM)
 Department of Information Systems and Computing
 Brunel University
 Uxbridge
 Middlesex, UB8 3PH
 UK
 e-mail: ray.paul@brunel.ac.uk

TILLAL ELDABI is a lecturer at the Department of Information Systems and Computing at Brunel University, UK. He received a B.Sc. in econometrics and social statistics from the University of Khar-toum. He received his M.Sc. in simulation modelling and his Ph.D. from Brunel University. His research is in aspects of healthcare management and the intervention of simulation and his main research also concentrates on the economy of healthcare. He is trying to exploit the means of simulation on to wider information systems management area to assist in problem understanding. Dr. Eldabi's email and web addresses are [<tillal.eldabi@brunel.ac.uk>](mailto:tillal.eldabi@brunel.ac.uk) and <http://www.brunel.ac.uk/~cssrtte>, respectively.

MAN WAI LEE is a Ph.D. student in the Centre for Applied Simulation Modelling and the VIVID Research Centre at the Department of Information Systems and Computing, Brunel University, UK. He is now under the supervision of Professor Ray J. Paul and Dr. Tillal Eldabi. He received a B.Eng. from the Department of Mechanical Engineering at The University of Hong Kong and a M.Sc. (with distinction) in building services engineering from Brunel University. His main research concentrates on fast simulation process, the new Grab-and-Glue modelling technique. His email address is [<manwai.lee@brunel.ac.uk>](mailto:manwai.lee@brunel.ac.uk).

RAY J. PAUL is a Professor of Simulation Modelling, Director of the Centre for Applied Simulation Modelling, founder of the Centre for Living Information Systems Thinking, all at Brunel University, UK. He received a B.Sc. in mathematics, and a M.Sc. and a Ph.D. in operational research from Hull University. He has published widely, in books, journals and conference papers, many in the area of simulation modelling and software development. He has acted as a consultant for a variety of United Kingdom government departments, software companies, and commercial companies in the tobacco and oil industries. He is the editor of the Springer-Verlag Practitioner book series. His research interests are in methods of automating the process of modelling, and the general applicability of such methods and their extensions to the wider arena of information systems. He is currently working on wider aspects of simulation, in particular in Web-Based Simulation and the new Grab-and-Glue modelling technique. His email and web addresses are [<ray.paul@brunel.ac.uk>](mailto:ray.paul@brunel.ac.uk) and <http://www.brunel.ac.uk/~csstrjpp>. Professor Paul has Parkinsonism, but insists on working part-time because he enjoys it.
