Developing a Business Simulation Game: Integrating Multiple Development Tools

Robert Williams & Des Klass
Curtin University of Technology, Perth, Australia

Bob.Williams@cbs.curtin.edu.au Des.Klass@gbs.curtin.edu.au

Abstract

In most cases today information systems development involves the use of multiple development tools. The developer has the choice of many competing vendor tools to choose from, and the choices to be made are complex. Will the various tools work together, and which tool should be chosen for which task? This paper describes the development of a business simulation game, and the role various development tools played in the process. The reasons, and how the tools were used, are also discussed.

Keywords: Software development tools, business simulation games.

Introduction

System development in the early twenty first century is a complicated task. Many technologies make up the development and operating environment. Mixtures of procedural and object oriented coding tools, and complex network, operating systems and World Wide Web applications make up this environment. It is most unusual that a complex system can be constructed with a single development tool and operate in a single production environment. Multiple tools and systems from multiple vendors generally are needed and used in practice (Beckham, 1991; Mahnke & Steiert, 2001; Shin & Wang, 2004).

This paper discusses the development of one such system – ServeSim. The system is a business simulation game for the service industry based on the business behaviours of the hotel industry. It was commissioned by a client for use in its industry training program. The system was developed using the Borland Delphi 6 programming tool, Borland Paradox database, Microsoft Excel 2000 spreadsheet, Microsoft Visual Basic for Applications (VBA), and the InstallShield Express installation tool. The system operates under the Microsoft Windows 2000 operating system in an IBM compatible PC environment.

Details of the Business Simulation

The ServeSim simulation was designed around the hospitality industry using a service quality framework. The simulation revolves around a number of key service management issues. These include: service strategy development, service quality, environmental analysis, the ability to integrate the various functional areas involved in senior level decision making and the role of leadership and team development in executive decision making.

Participants play the role of the executive management group, and develop and implement an overall strategy for one of the competing organizations. The success of this strategy depends on the participants' ability to think strategically and to contend with the slings and arrows of external environmental influences, including competitor strategy.

As each round of decision-making goes by, the outcomes of the teams judgment becomes evident in terms of both the performance of their organization and on their personal equanimity. Success depends on a number of factors including wise decisions by the team, as well as environmental factors and the serendipitous poor choices that might be made by the team's competitors.

All teams start from an identical position and will compete over several rounds of decision-making where each decision period equates to three months.

The key objective of the simulation is for participants to gain a thorough understanding of what drives success in terms of provision of services in the private sector – although many of these learnings will also be relevant to an increasingly market driven public sector. One of the things this simulation looks for is the overall strategy that participants adopt for their companies and how they go about implementing that strategy.

There are both 'hard' measures for success such as:

- Profitability
- Occupancy

As well as 'soft' measures such as:

- Observation of team dynamics
- Problem solving

As indicated earlier, in the simulation, teams of individuals form companies, which they then manage. A company faces a competitive environment in which other firms (up to six firms in the industry) operate similarly. All of this high-pressure activity takes place in a simulated world, with most of the attention focused on the island of Luann, a beautiful semi tropical landmass with three distinct regions.

Situated on the island are up to 12 resorts (depending on the number of teams competing), with half in Region One and the other half in Region Two. The resorts in both of these regions are clustered quite closely together with common walkways. At the start of the simulation all of the resorts are identical. At present there has been no development in Region Three. Originally all of the complexes were owned by a single Consortium – the Bainbridge Group, however a decision was made by the Bainbridge Group to withdraw from the hospitality industry and concentrate on their media interests. As a consequence the chain was broken up into equal parts and sold to separate companies. Each of these companies has one resort in Region One and one in Region Two. One year later the resorts have changed hands once again and this is where the team comes in. The team has just taken over one of these companies and must now decide what the future will bring.

The Island

Luann is a beautiful semi tropical island in the southern hemisphere that despite its dearth of natural resources has become quite a successful Mecca for a variety of travellers.

Region One has the advantage of unspoilt beaches, which seem to go on forever. These are stinger free and range from small to medium sized surf to calm water. Toward the east of Region One there also exists Wee Waters, a small protected cove that is perfect for very small children.

Region Two also possesses beautiful beaches but not to the same extent as Region One. What sets Region Two apart is its mountainous terrain and extensive walking trails. Region Two also has the old Seda Point Lighthouse. Built around 1850, it forms an important part of maritime history in the region. Opportunities for surfers also exist, with small to mid sized waves common along the coast.

Undeveloped to date, Region Three with its pristine environment is home to many marine and land creatures. Humpback whales pass close to the shoreline in Region Three en route to their breeding grounds. Turtles also have a special attraction to this part of the world and the island contains a substantial rookery. A declining world population in turtles, attributed to industrial fishing and over development of coastal regions means that the Luann Island rookery plays a significant role in the survival of the species. November through to March is a critical time as it covers the period from when the turtles come onshore to lay their eggs through to the perilous hatching period. Very few of those that emerge from the sand will ever grow to adulthood.

Each of the participating teams starts with two resorts, one in Region One and the other in Region Two. At the start of the simulation all resorts may be assumed to be identical in terms of architecture and features. Each complex has quite extensive grounds thus providing room for expansion if warranted, and includes a swimming pool, a restaurant and bar cum tearooms. As discussed earlier, the resorts in each region are clustered quite closely together and share common walkways between the resorts.

Each of the resorts has three levels of accommodation, forming a total of 50 suites i.e.:

Level One:	Premium accommodation	10 suites
Level Two:	Standard accommodation	30 suites,
Level Three:	Budget accommodation	10 suites

Each lodging type comfortably accommodates up to 4 people. Visitors have access to the pool, restaurant and bar regardless of level of accommodation. Use of the pool is included in the accommodation costs, however the other areas charge according to consumption.

There is a whole series of decisions that have to be made by the teams and some of these decisions include:

- Price
- Accommodation (New and/or reclassifications)
- Improvements (Buildings, facilities, Surroundings)
- Salary (Maintenance, Management, Service)
- Staffing levels
- Staff training and development
- Marketing (International /Local Promotions, Agent Discounts)
- Revenue Generating Facilities
- and much more…

The model is dynamic in that the outcome is determined by the decisions made by the teams. The successful decisions are ones that optimise on the decisions of the other teams within this competitive environment.

The Structure of the SERVESIM System

ServeSim has a number of data entry screens in which the user enters all the required data such as hotel company names, number of rooms in each category, number of staff, marketing budgets and so on. This data is then stored in database files. Computations for each period are performed by the spreadsheet, when the data from the database files is inserted into the appropriate cells in the spreadsheet. Reporting is performed by Excel macros that are triggered by the user from the main screen of the program. Understanding of the following sections of this paper will be easier if this model is kept in mind.

The Development Tools

The coding effort to develop the system was of the order of 500 person-hours, resulting in about 12,700 lines of Delphi code. The system that has resulted works very well, with excellent performance on a 1ghz AMD Athlon machine with 512K RAM. The nature of the game required a Windows program that could allow the user to enter many sets of data related to the running of hotel businesses in multiple markets. This data would then be used in complex calculations based on the behaviours of the hotel industry. Graphical and textual reports would then be required for examination at the end of a simulation period to help determine input data for the next simulation period. These requirements meant that a

- Windows development tool with good mathematical computation tools was needed.
- A Database tool to build the required data stores
- A graphical package to produce the required graphs
- A reporting tool to produce the required textual reports
- And an tool to build the final installation program

The products chosen to meet these needs were

- Borland Delphi - a Rapid Application Development (RAD) tool for Windows development
- Borland Paradox - a complete Database product that ships with Delphi
- Borland Database Desktop which understands Paradox and can be used to manipulate Paradox tables at design time
- Microsoft Excel is the industry standard spreadsheet software
- Microsoft VBA is the standard language for programming the Office suite of products, including Excel
- InstallShield Express builds Windows installation programs

Rationale for Choosing the Development Tools

Delphi 6 (the current version at the start of the project) was chosen as the primary development tool because of its suitability to the task at hand, and with the developers' previous successful experience with Delphi 4 for the development of the Allocate corporate resource allocation program, now used in over 10 organisations in Australia. Delphi 6 also has strong built-in support for interaction with Microsoft Office products, including Excel.

Paradox and Database Desktop were chosen for the database work because of their integration with Delphi.

Excel was chosen because of its ability to easily allow calculations to be programmed, its strong support for graphs, and its widespread use in industry.

VBA was chosen because of its association with Excel.

InstallShield Express was chosen because of a Delphi specific version that easily handles the installation of the Borland Database Engine (BDE).

The SERVESIM Database

There are 18 database tables and thousands of calculations involved in the execution of a simulation cycle in ServeSim, and the question of a naming convention for the database fields and program variables had to be considered. The prototype for ServeSim was a previously programmed Excel spreadsheet prototype, and it contained many calculations and Macros coded in terms of spreadsheet functions and cell addresses. A user friendly and user secure adaptation of this spreadsheet was required It occurred to the developers that perhaps using a variable naming convention based on individual cell addresses may be appropriate. E.g. A266, DE34 etc. This would enable the direct translation of the spreadsheet formulae into Delphi Pascal code. On reflection it occurred to the developers that maybe somehow the appropriate values from the database tables could be pumped directly into the spreadsheet itself, thus eliminating the need for any coding of calculations at all, and hence eliminating the need for the variables in the first place. As the base spreadsheet had already been substantially debugged, minimum debugging of the calculations components would thus be needed.

Research via the Web lead to the location of a number of articles from Delphi Informant magazine relating to the use of Excel with Delphi. It turns out that Delphi 6 has a very comprehensive set of components that enable seamless integration with Microsoft Office, including Excel. Delphi 6 can work with Excel in 3 ways

- Via ADO – ActiveX Data Objects
- Via ODBC – Open Database Connectivity
- Via Automation – Program Interaction

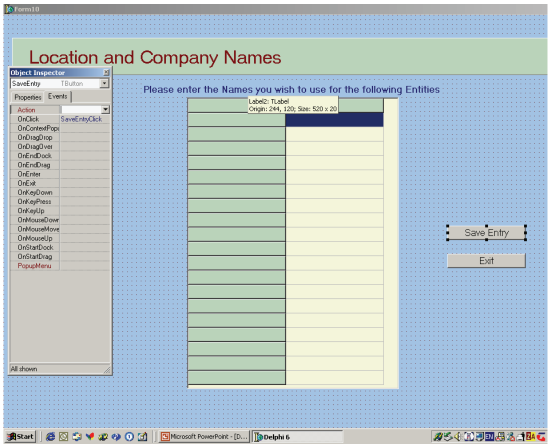
The developers chose Automation because it provides the most powerful facilities, and the requirements for the Excel interaction were unknown at the start of the project – we “hedged our bets”.

Delphi 6 implements Automation via Automation Server Components. These components are chosen from the component palete and dropped on a form. Typically we use an

- ExcelApplication
- ExcelWorkBook
- ExcelWorkSheet

The following screen shots show these components placed on a Delphi form at design time. Note the simplicity of the implementation of the complexities of interaction with Excel.

Event handlers are then written to manipulate the spreadsheet.



Form10

Location and Company Names

Object Inspector

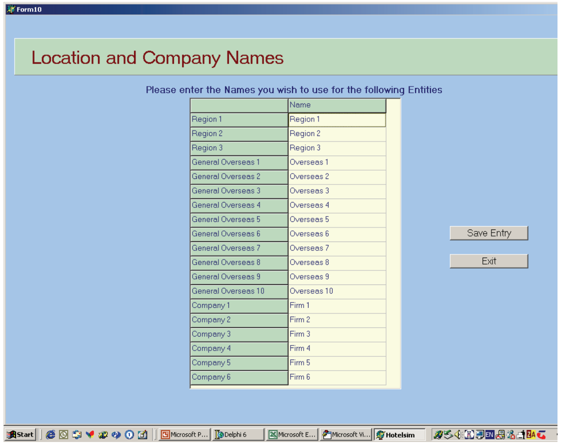
SaveEntry TButton

Action	
OnClick	SaveEntryClick
OnContextPopu	
OnDragDrop	
OnDragOver	
OnEndDock	
OnEndDrag	
OnEnter	
OnExit	
OnKeyDown	
OnKeyPress	
OnKeyUp	
OnMouseDowr	
OnMouseMove	
OnMouseUp	
OnStartDock	
OnStartDrag	
PopupMenu	

All shown

Please enter the Names you wish to use for the following Entities

Label2: TLabel
Origin: 244, 120; Size: 520 x 20



Form10

Location and Company Names

Please enter the Names you wish to use for the following Entities

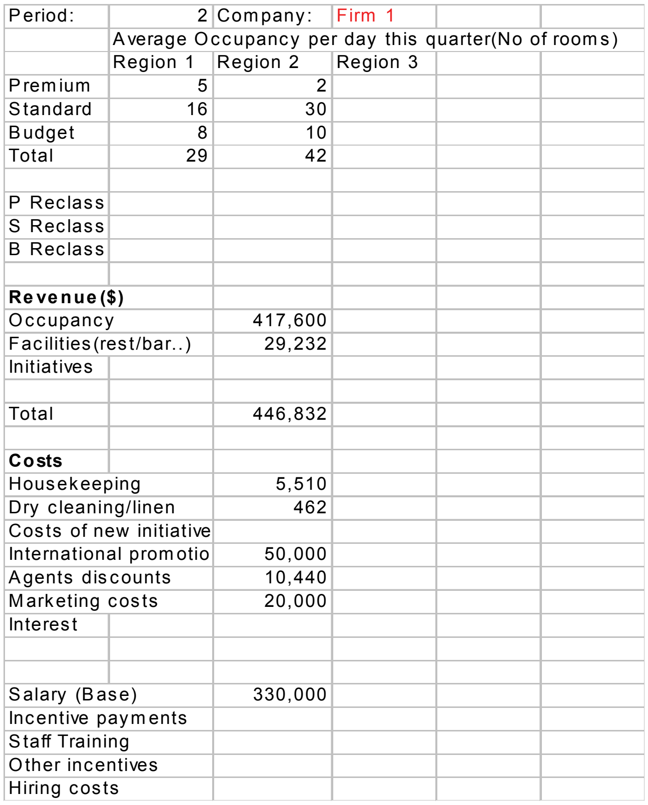
	Name
Region 1	Region 1
Region 2	Region 2
Region 3	Region 3
General Overseas 1	Overseas 1
General Overseas 2	Overseas 2
General Overseas 3	Overseas 3
General Overseas 4	Overseas 4
General Overseas 5	Overseas 5
General Overseas 6	Overseas 6
General Overseas 7	Overseas 7
General Overseas 8	Overseas 8
General Overseas 9	Overseas 9
General Overseas 10	Overseas 10
Company 1	Firm 1
Company 2	Firm 2
Company 3	Firm 3
Company 4	Firm 4
Company 5	Firm 5
Company 6	Firm 6

Figure. 1. Design time user input screen

Figure 2. Data Entry screen at run time

Figures 1 and 2 show the design and run time screens for entering Hotel Location and Company Names into the system.

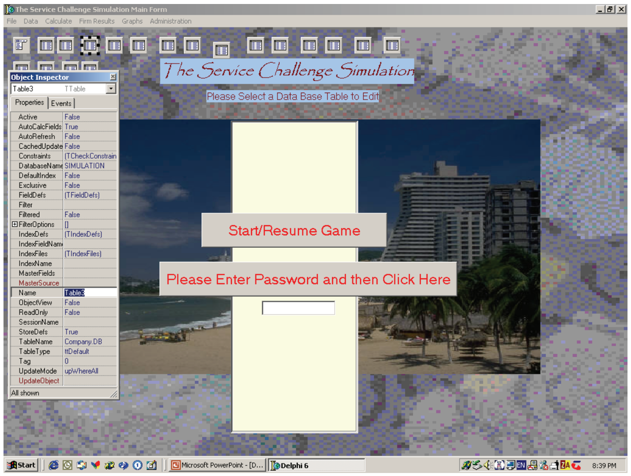
The relevant range of the spreadsheet for Company 1 is shown in Figure 3.



Period:	2	Company:	Firm 1		
	Average Occupancy per day this quarter(No of rooms)				
	Region 1	Region 2	Region 3		
Premium	5	2			
Standard	16	30			
Budget	8	10			
Total	29	42			
P Reclass					
S Reclass					
B Reclass					
Revenue($)					
Occupancy		417,600			
Facilities(rest/bar..)		29,232			
Initiatives					
Total		446,832			
Costs					
Housekeeping		5,510			
Dry cleaning/linen		462			
Costs of new initiative					
International promotio		50,000			
Agents discounts		10,440			
Marketing costs		20,000			
Interest					
Salary (Base)		330,000			
Incentive payments					
Staff Training					
Other incentives					
Hiring costs					

Figure 3: Spreadsheet for Company 1

Figure 4 shows the main screen at design time with a number of Database components placed on it. Table3 is highlighted as the component that has focus, and the Object Inspector box to the left shows some of the design time properties for Table 3 and their values.


The Service Challenge Simulation Main Form
File
Data
Calculate
Firm Results
Graphs
Administration
The Service Challenge Simulation
Please Select a Data Base Table to Edit
Start/Resume Game
Please Enter Password and then Click Here
Start
Microsoft PowerPoint - [D...
Delphi 6
8:39 PM

Object Inspector

Table3 TTable

Properties | Events

Active	False
AutoCalcFields	True
AutoRefresh	False
CachedUpdate	False
Constraints	(TCheckConstrain
DatabaseName	SIMULATION
DefaultIndex	False
Exclusive	False
FieldDefs	(TFieldDefs)
Filter	
Filtered	False
⊞FilterOptions	[]
IndexDefs	(TIndexDefs)
IndexFieldNam	
IndexFiles	(TIndexFiles)
IndexName	
MasterFields	
MasterSource	
Name	Table3
ObjectView	False
ReadOnly	False
SessionName	
StoreDefs	True
TableName	Company.DB
TableType	ttDefault
Tag	0
UpdateMode	upWhereAll
UpdateObject	

All shown

Figure 4. Main screen at design time showing database components

VBA Macros for Reports and Period Rollover

The spreadsheet has a number of macros for end of period processing, report and graph production. These are manipulated from Delphi as in the following examples. Figure 5 shows how a particular graph can be printed.

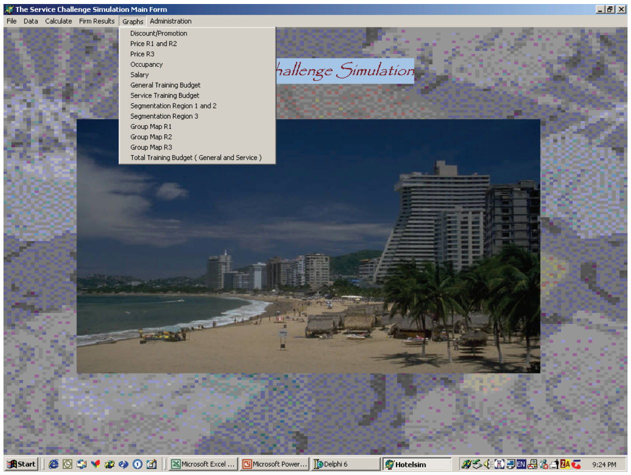
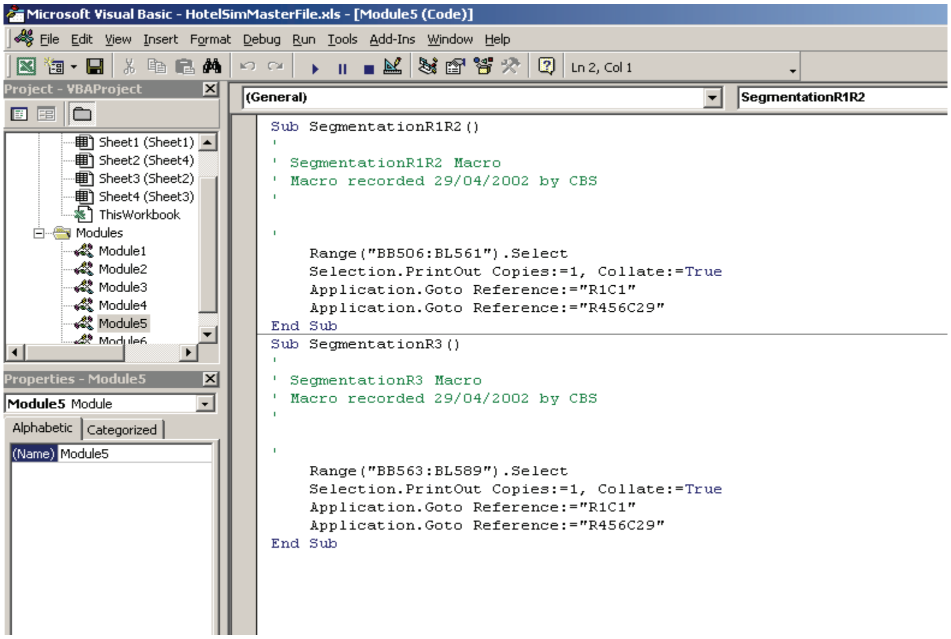


Figure 5. Main screen at run time showing menu items for graphs

Figures 6 and 7 show this VBA code at design time in the VBA editor, and the resultant graph.



```
Sub SegmentationR1R2()
'
' SegmentationR1R2 Macro
' Macro recorded 29/04/2002 by CBS
'

'
    Range("BB506:BL561").Select
    Selection.PrintOut Copies:=1, Collate:=True
    Application.Goto Reference:="R1C1"
    Application.Goto Reference:="R456C29"
End Sub
Sub SegmentationR3()
'
' SegmentationR3 Macro
' Macro recorded 29/04/2002 by CBS
'

'
    Range("BB563:BL589").Select
    Selection.PrintOut Copies:=1, Collate:=True
    Application.Goto Reference:="R1C1"
    Application.Goto Reference:="R456C29"
End Sub
```

Figure 6. VBA macros at design time

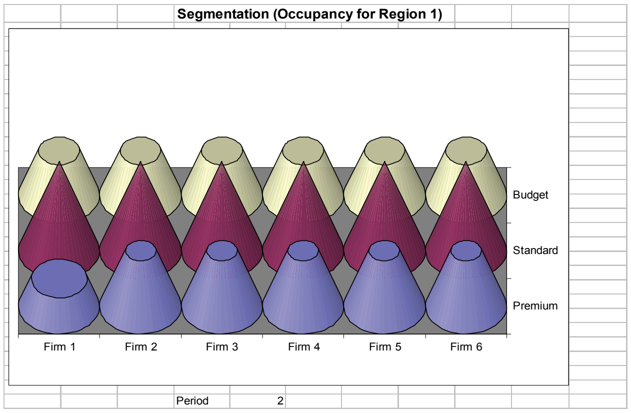


Figure 7. Graph produced for segmentation

Discussion

The integration of the many products described in this paper went reasonably smoothly. Prior to the implementation, the authors were familiar with Delphi, Excel, Paradox, VBA and Install-Shield, but not with the interaction between Delphi and Excel. Some research was required as to this could be done efficiently. Once this mechanism was understood, there were no major problems encountered with programming our specific requirements.

The authors feel that the experience outlined in this paper will become more the norm than the exception in future projects, because each individual development tool has it’s own strengths, but generally a single tool does not cover all the functionality required today. The ability to integrate multiple tools when building systems will become a necessary requirement for success in the future.

Better systems result, as there is a synergistic effect on the evolving system through the development process, as each tool contributes its strength to the process.

References

Beckham, B. (1991). A Scheme for Little Languages in Interactive Graphics. *Software – Practice and Experience*, 21, 2, 187-207.

Mahnke, W. & Steiert, H. (2001). To a Man with an ORDBMS Everything Looks Like a Row in a Table, *Third International Symposium on Cooperative Database Systems for Advanced Applications,* Beijing.

Shin, K. & Wang, S. (2004) Performance Modeling for Software Integration, *Position Paper,* Department of Electrical Engineering and Computer Science, The University of Michigan. URL: http://www.itrd.gov/iwg/sdp/vanderbilt/position_papers/kang_shin_performance_modeling_for_software.pdf Accessed 13 December 2004.

Biographies

Robert Williams has over 25 year's experience in the Information Systems industry, as a practitioner, researcher and lecturer. He currently is a lecturer in the School of Information Systems at Curtin University of Technology in Perth, Western Australia. He has extensive experience in systems analysis and design, and programming, on a variety of mainframe, mini and personal computers, and a variety of operating systems and programming languages. Applications he has worked with include mathematical, statistical, bridge and road engineering, financial, corporate resource allocation, business simulation and educational systems. He has published a number of articles on system users' personalities and satisfaction, decision support systems, and automated essay grading systems. In 2001 he led a team of researchers in the School of Information Systems at Curtin University of Technology which conducted what is believed to be the first trial in Australia of an Automated Essay Grading system. Robert holds a Bachelor of Arts degree with double majors in Mathematics and Economics from the University of Western Australia, a Graduate Diploma in Computing from the Western Australian Institute of Technology, and a Master of Information Systems degree from Curtin University of Technology.

Dr. Des Klass is an Associate Professor at the Graduate School of Business at Curtin University. Dr. Klass has published extensively and is the co-developer of the GDAM (Generalised Decision Assurance Methodology) facilitation process and the “Team Decision Framework” instrument that identifies the base level of decision intelligence inherent in decision groups. Dr Klass consults in several areas which include strategy and policy development and review, executive development programs, the development and application of business simulation games, improving resource allocation decisions and processes and enhancing decision quality. His research interests include Group Decision Support Systems, Computer aided facilitation, Corporate Strategy, Decision Analysis and Decision Quality.