Significant decisions: supporting social housing clients through *procession*'s artificially intelligent 3d information visualisation

Steve North
University College London
Virtual Reality Centre for the Built Environment
London WC1E 6BT
UK

Published in *Design & Decision Support Systems in Urban Planning: proceedings of the 5th international conference (Nijkerk, The Netherlands, 22nd-25th August 2000)*,
Harry Timmermans and Bauke de Vries, Editors,
The Netherlands: Eindhoven University of Technology, ISBN 90-6814-110-4, pp.215-228.

ABSTRACT

Procession is a 3d information visualisation software tool for social housing project clients. This client group are known as Registered Social Landlords (RSLs), which is the UK's catchall term for registered Housing Associations, Housing Co-operatives and Local Housing Companies. The focus of this research is specifically on client reporting during phase eight ('construction'- commonly known as the 'on-site' phase) of the University of Salford Generic Design and Construction Process Protocol (GDCPP) Group's Process Protocol Map. Procession's data surface is actually a 'carpet plot', generated by a scalar algorithm. The three dimensions are achieved from a two-dimensional set of points, which are warped by scalar values, in the direction of the surface normal. The amount of warping is controlled by the scalar value. A project with no deviations from the project baseline will produce a flat data surface. A fourth and non-spatial dimension is provided by colour mapping the data surface. This scalar value represents the significance of each deviation, to this specific client. By analysing the contribution of each deviation to the total deviation levels, Procession develops a set of artificial intelligence heuristics concerning each task's contribution to overall project deviation.This paper describes the development of the Significance of Deviance Algorithm. It is intended that this formula will calculate the contribution of an individual task, to the current total value of a deviation parameter, within a specific construction project.

1 INTRODUCTION

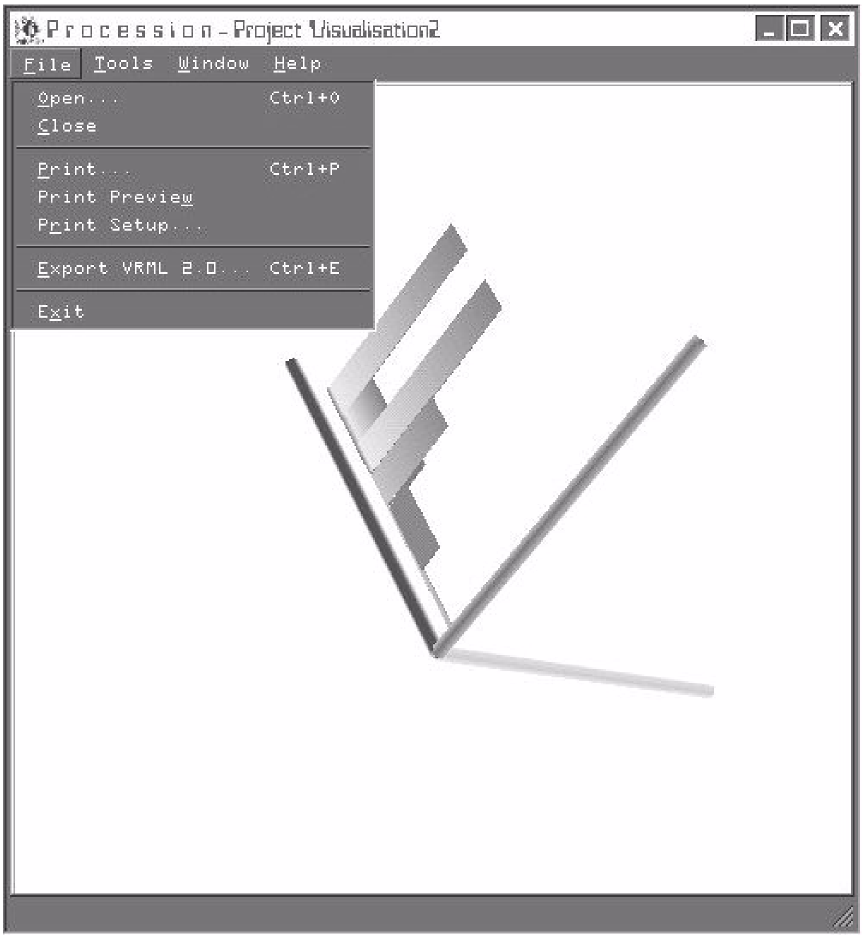
The conceptualisation, development and evaluation phases of the Procession software tool have been documented elsewhere (North 1999 and North 2000). Instead, this paper focuses on the technical approach underlying Procession's 'intelligent' elements. Before describing Procession's artificial intelligence features, a brief introduction to the background issues will now follow.

The construction industry already provides many visualisation tools for interpreting physical building models, but investigation has failed to identify anything similar for non-physical construction processes. Information of this type is usually known as project planning data and is concerned with the individual tasks (or processes) that make up construction projects. Particularly notable, is the absence of a planning visualisation tool providing decision support for construction clients. Procession's data surface is an abstract representation of three selected project dimensions. Its 3d progress reports provide construction clients with an 'at-a-glance'

Correspondence:
Steve North; Email: steve@stevenorth.com;
Telephone: +44 (0)20 72 63 50 09;
Surface Mail: Steve North, VR Centre For The Built Environment, University College London, 1-19 Torrington Place, London WC1E 6BT UK

indication of project 'health'. The peaks and troughs of the 3d data surface correspond to a snapshot of current project performance (or 'deviation') parameters, such as budget and schedule. The focus of this research is specifically on client reporting as

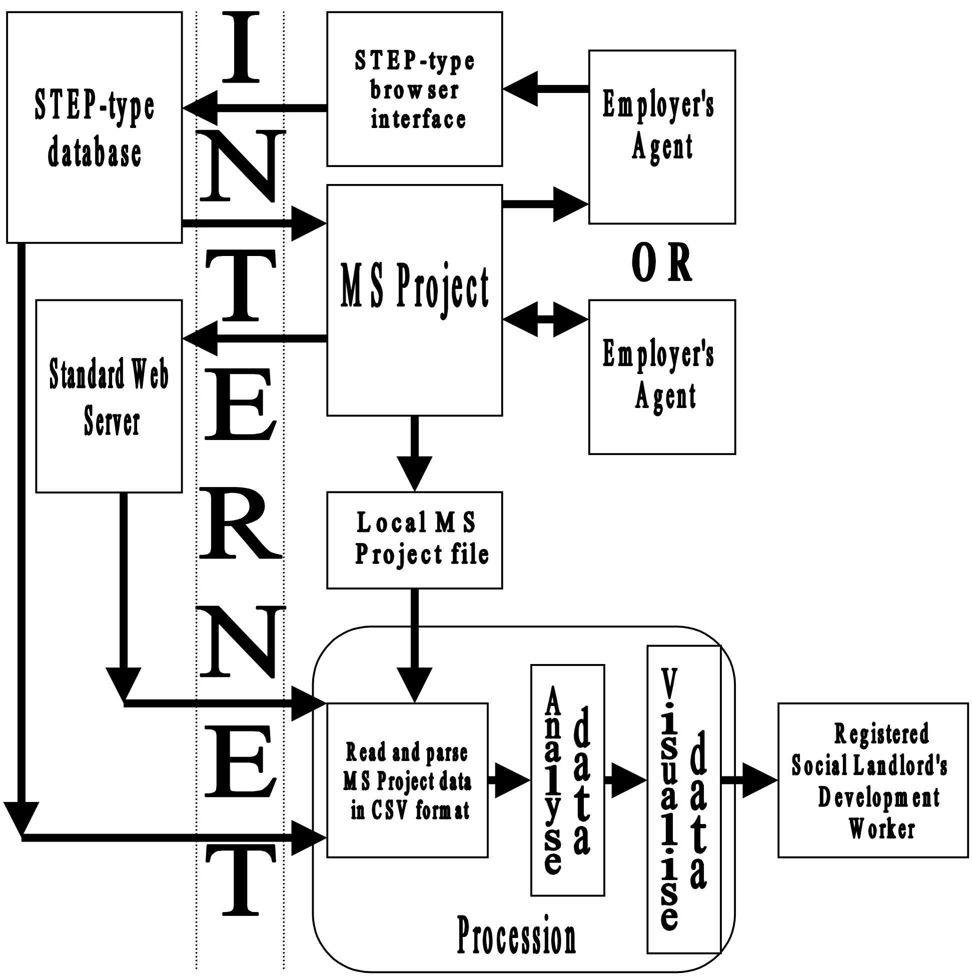
Figure 1. **Procession's Windows interface**



described by phase eight ('construction'- commonly known as the 'on-site' phase) of the University of Salford GDCPP Group's Process Protocol Map (Kagioglou et al. 1998). Procession is a stand-alone application developed for the Microsoft Windows 95/98/2000/NT platform (see Figure 1). It is capable of opening data files both from a local computer and from an Internet web server. Procession was developed in the C++ language, utilising Microsoft Foundation Classes (MFCs). Procession's 3d graphics functionality is provided by the Visualization ToolKit (VTK), an 'open source' system providing a very flexible C++ class library (Shroeder, Martin and Lorensen 1998).

Procession's data surface is actually a 'carpet plot', generated by a scalar algorithm. The three dimensions are achieved from a two-dimensional set of points, which are warped by scalar values, in the direction of the surface normal. The amount of warping is controlled by the scalar value. Users can navigate the 3d data surface with a standard mouse. It is possible to 'examine' the data surface (left mouse button and drag), move forwards and backwards (right mouse button and drag) or slide in any direction (shift key and the left mouse button). Procession's menu options provide the following functionality: print current view, export a 3d report to Virtual Reality

Figure 2. **Data-flow between employer's agent and the registered social landlord's development officer, via MS Project, the Internet and Procession**



Modelling Language (VRML- version 2.0/97), take and save a 2d snapshot of a current view in the Windows bitmap (BMP) format, animate the data surface on a pre-set path and turn detailed progress alerts on or off. Procession's rendering can be switched from solid, which is the default, to stereoscopic (requiring red and blue coloured glasses) or wire-frame.

Figure 2, illustrates the three different ways that a development worker can use Procession. In the first scenario, the employer's agent exports a local data file from Microsoft Project and physically sends it to the development worker (postal service, email, by-hand etc.). The second method sees the worker up-load the data file from Microsoft Project to a traditional web server. This could be achieved with a standard

File Transfer Protocol (FTP) tool or possibly using Microsoft Project's built-in ability to 'publish' on the Internet. The Standard for The Exchange of Product model data (STEP) is defined in International Standards Organisation (ISO) standard 10303. The final approach is to make use of a STEP-type project database, such the University of Salford's Web-based IFC Shared Project EnviRonment - WISPER (Faraj et al. 1998).

As previously stated, Procession does more than transform planning data into 3d visual structures. It also has functions described as 'intelligent'. Primarily, Procession is capable of considering the impact of each performance deviation on the final project outcome. This is achieved by calculating the *expected* and *actual* contributions of each project task to the total deviation for each parameter (i.e. cost or work variances). By comparing these values, the Significance of Deviance Algorithm (see section 3) works out a significance value. If the task has already been encountered by Procession, it will have a corresponding stored weighting value. If it is a new task, a default weighting value will be assigned. The significance figure is used to increase or decrease the specified 'weighting' value. The adjusted weighting is used to colour the data-surface deviations, indicating the *relevance* of a deviation level, as discrete from its size. Finally, the adjusted weightings are saved back to the 'legacy archive' (see section 4), enabling Procession to 'learn' by its experience. Further sections of this paper describe the software structure that constitutes Procession, with specific reference to the techniques used for 'significance' and 'learning'.

2 THE SOFTWARE STRUCTURE

Figure 3 show's the route taken by raw text data, as it is converted to 3d visual structures by Procession's C++ classes (figure 4). Procession was designed as a partner application for Microsoft Project. Before Procession can process Microsoft Project data files, they must be converted to a different format.

Procession imports data in the Comma Separated Value (CSV) format. Files of this type have the extension .CSV and are commonly available as export options from database and spreadsheet applications, including Microsoft Project. CSV uses an American Standard Code for Information Interchange (ASCII) text encoding structure, with individual values followed by a comma. The following example shows comma-separated values exported from Microsoft Project, where each line represents a project task:

```
Plumbing,No,100%,100%,28,£0.00,"£36,308.48","-£36,308.48",£0.00,£0.00,0 days,0 hrs,
Electrical,No,67%,0%,28,£0.00,"£18,154.24","-£12,163.34",£0.00,£0.00,0 days,0,hrs,
```

A character used to separate data items is known as a 'delimiter' and commas are only one of the characters frequently used for this purpose. Tab-delimited text is also in common use. Early versions of Procession used a comma-delimiter, but when tested with a project data set consisting of larger values, a problem emerged. By default, Microsoft Project uses a comma in larger numbers, to separate thousands and hundreds. Unfortunately, there is no option to customise the format of exported

Figure 3. ***Procession's C++ procedural flow from 'raw' data to 3d visual structures***

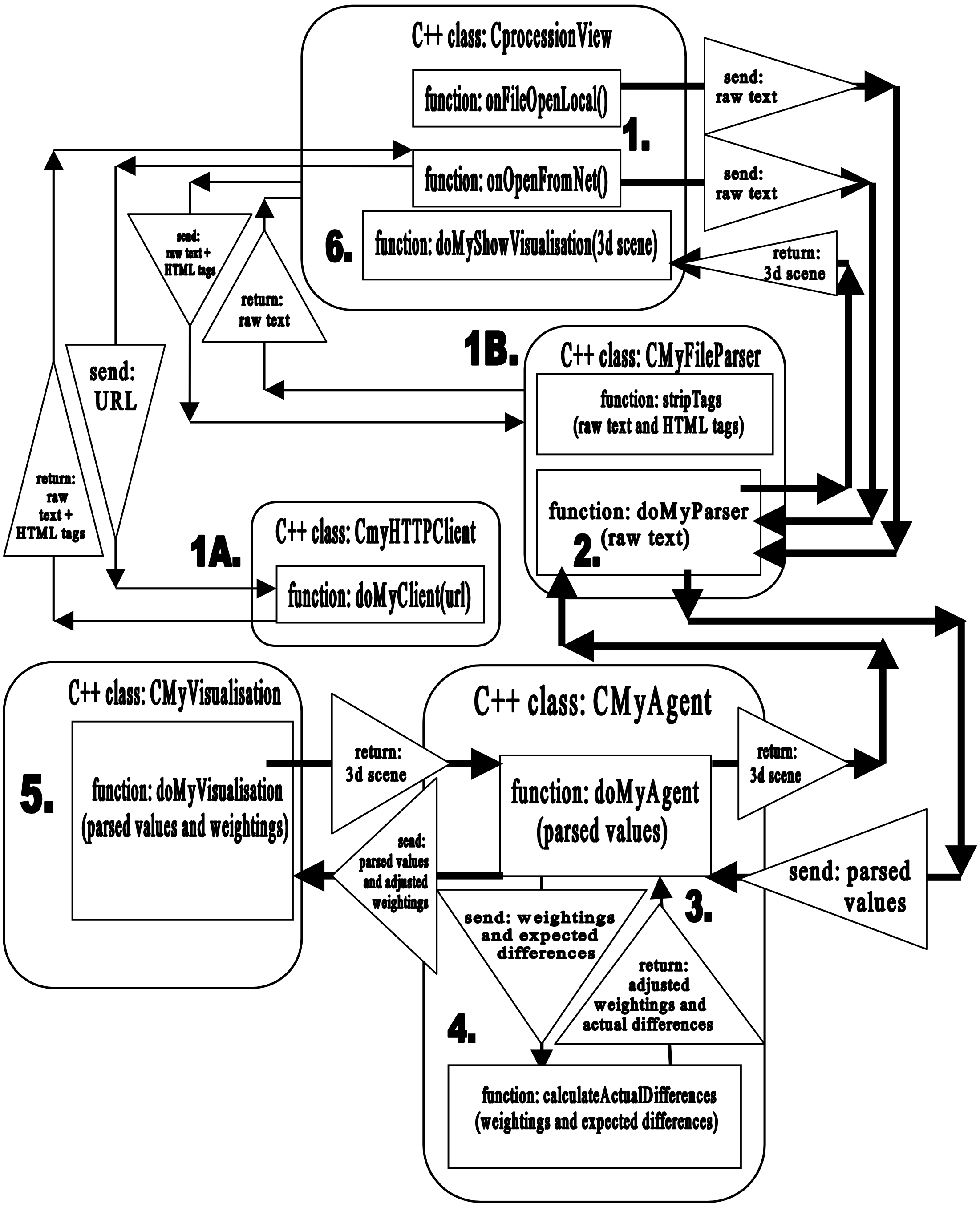
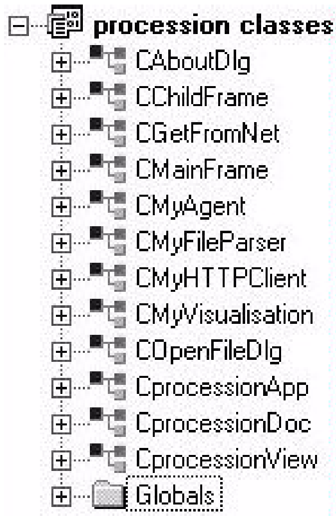


Figure 4. **Procession's C++ class structure**



numbers. Therefore, Procession's parser had to be adapted to filter tab-delimited data files. The previous data example is now repeated with a tab-delimiter:

Plumbing No 100% 100% 28 £0.00 "£36,308.48""-£36,308.48" £0.00 £0.00 0 days 0 hrs
Electrical No 67% 0% 28 £0.00 "£18,154.24" "-£12,163.34" £0.00 £0.00 0 days 0 hrs

In order to export a delimited text file, it is necessary to prescribe a data map. This describes the fields to be exported and their order in the file. Each line in Procession's CSV export map consists of the following Microsoft Project fields:
Task Name, Summary (is this a summary/top-level task? YES/NO), %Complete (percentage of budgeted time expended), %WorkComplete (percentage of budgeted money expended), WBS (the ID code for each task), BCWS (an earned value analysis figure), Baseline Cost (the budget for this task), Cost Variance, Schedule Variance, Variance At Completion, Duration Variance and Work Variance. The last five of these represent the deviation parameters visualised in the 3d data surface. More detailed explanations of these now follow:

- COST VARIANCE, units £s. At this point in the project, is completed work on this task over budget?
- SCHEDULE VARIANCE, units £s. At this point in the project, how are we doing on this task compared with the estimated spend?
- VARIANCE AT COMPLETION, units £s. At the end of the project, what is the difference predicted to be between the current estimate of the total cost for this task and the original estimate?
- DURATION VARIANCE, units days. At the end of the project, what is the difference predicted to be between the current estimate of the total time to complete this task and the original estimate?

- WORK VARIANCE, units hours. At the end of the project, what is the difference predicted to be between the current estimate of total number of person hours to complete this task and the original estimate?

Familiarity with Procession's export map will allow a user to configure another Project Management application for Procession compatibility, or even to create data files from scratch. For the Microsoft Project user intending to provide progress reports with Procession, there is one factor that will assist the process. The distribution of Procession includes example template files for Microsoft Project. These contain the CSV export map and Visual Basic for Applications (VBA) Macros, for automatically exporting projects to Procession and publishing CSV files to the Internet. This is an example of one of the VBA macros, which prompts the user for a file name and then saves the file using the Procession CSV map:

```
Sub procession3dExporter()
' Macro procession3dExporter
' Macro Recorded Sun 19-03-00 by steve north.
' Guess at the name for the CSV file
suggestedName = Dir (".\*.mpp")
' Prompt user for CSV file name offering a suggestion based on current project
FileName = InputBox("Enter CSV file name(excluding .csv)", "Procession exporter",
suggestedName)
' Save the CSV using the Procession export map
FileSaveAs Name:="./" + FileName + ".csv", FormatID:="MSProject.CSV.8",
map:="procession "
End Sub
```

3 THE SIGNIFICANCE OF DEVIANCE ALGORITHM

Procession applies artificial intelligence to the problem of deciding the significance of an individual project task to the current total value for a specific deviation parameter in a specific project. The mathematical approach utilised is based on statistical methodologies, such as ANalysis Of VAriance (ANOVA), where the term 'deviate' is the actual value for an individual score X, minus the average score for this group of scores, i.e. Xi – Mx (Lowry 2000, chapter 13). In Procession, this value has been termed 'the significance of deviance', which is a current snapshot of significance. By using the current value to increase or decrease an on going record of previous significance (the legacy archive), Procession can 'learn' from its runtime experiences. Figure 5, shows the flow of data from CSV file to 3d visual structures, via significance calculation and up dating of the legacy archive.

Figure 5. **Procession's symbolic internal data-flow for calculating the significance of deviance and 'learning'**

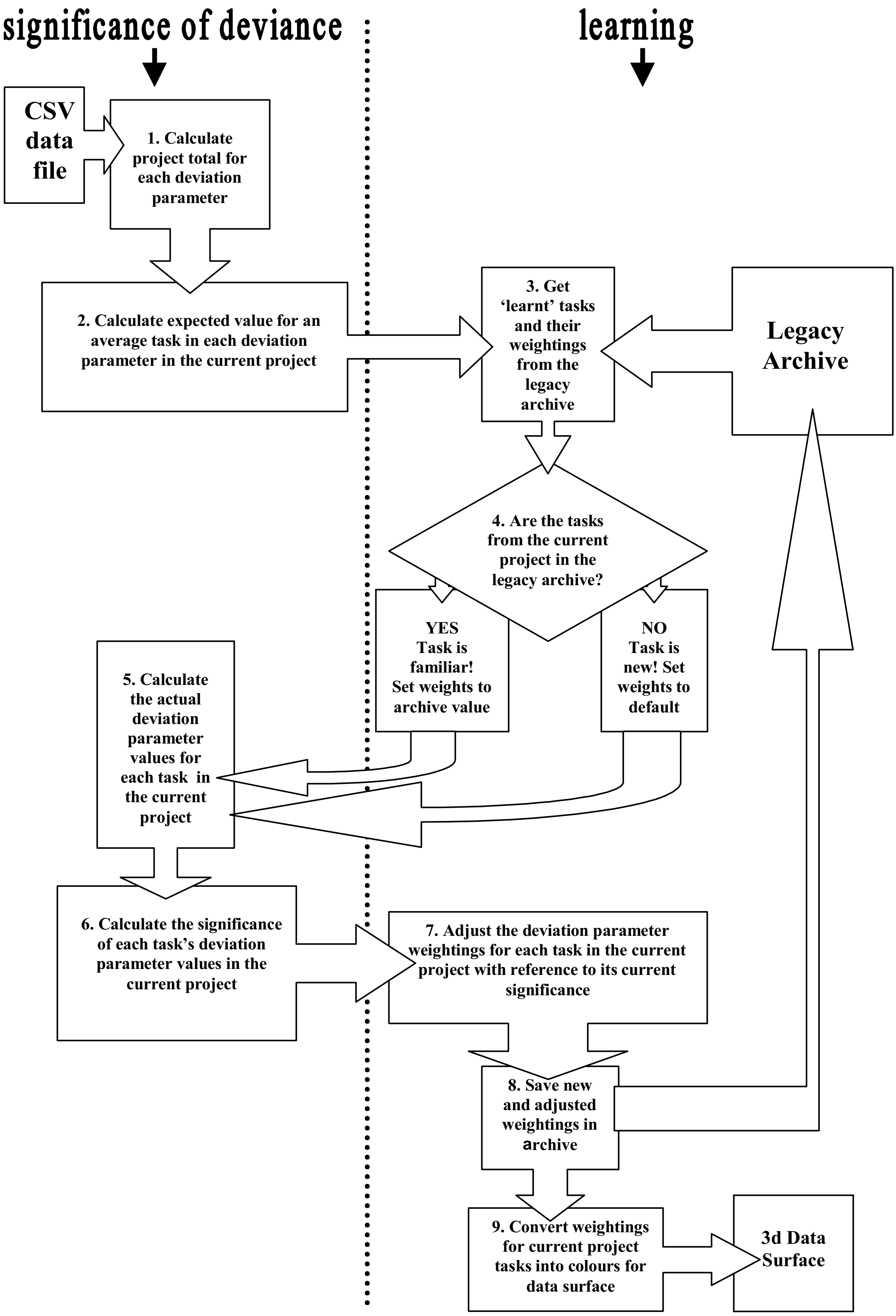
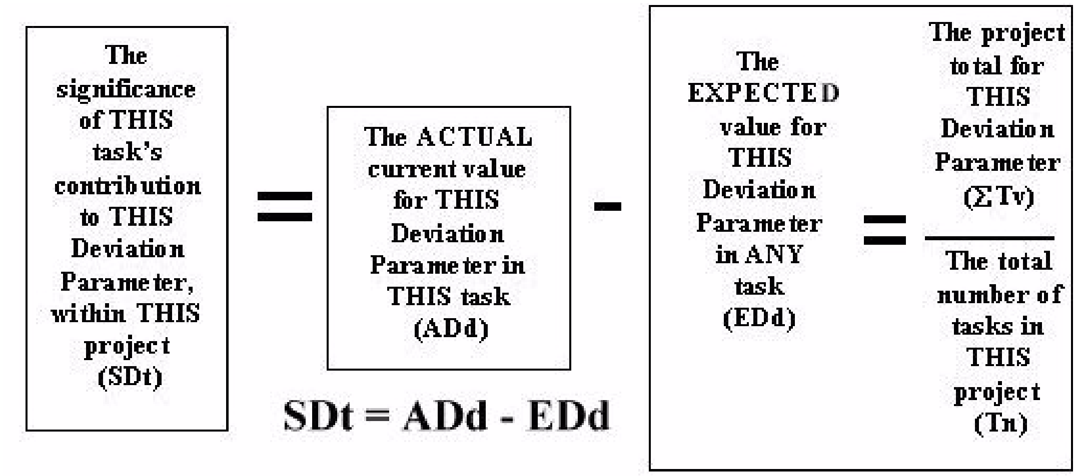


Figure 6. **Procession's Significance of Deviance Algorithm**



This paper will now provide a more in depth explanation of the significance calculation. Figure 6 provides the formula for Procession's Significance of Deviance Algorithm. The following definitions have been assumed:

SDt = The significance of a specified task's (t) contribution to a specified deviation parameter, within a specified project
d = a specified deviation parameter
p = a specified project
t = a specified task
Tn = the number of tasks in a specified project
Tv = the current value of a specified deviation parameter value (d) for a specified task (t) in a specified project (p)
ADd = The actual current value for a specified deviation parameter in a specified task
EDd = The expected value for a specified deviation parameter (d) for any task (t) in a specified project (p)
EDd = $\sum$(Tv) / Tn

The resulting significance value can be interpreted as follows:

- A positive result value indicates that this task is making a greater than average contribution to the total for this deviation parameter. Therefore, the weighting should be increased.

- A negative result value indicates that this task is making a smaller than average contribution to the total for this deviation parameter. Therefore, the weighting should be decreased.

Table 1. **Example values for two deviation parameters**

	DEVIATION PARAMETERS	
TASK NAME	COST VARIANCE	DURATION VARIANCE
Plumbing	£4.00	1 day
Electrical	£2.00	3 days

Table 1, shows an example of a progress report from a simple two-task project. The only deviation parameters under consideration here are 'cost variance' and 'duration variance' (see section 2). For both of the deviation parameters, the total value is calculated across all tasks (∑(Tv)). In this case, the total cost variance is six and the total duration variance is four. Next, the total number of tasks contributing to each deviation parameter is determined (Tn): cost variance has two tasks and duration variance also has four. If all tasks were contributing equally to the total deviation values, you would expect each task to have an individual cost variance of three (six divided by two tasks) and a duration variance of two (four divided by two tasks). These are the expected deviations for cost variance and duration variance in this project, at this time (EDd). The following C++ code exert demonstrates Procession's calculation for expected deviation:

```
// first calculate project deviation parameter totals
for(v=0;v<theNumberOfDeviationParameters;v++)
{
	for (i=0; i<totalSubTasks; i++)
	{
	totalProjectDeviations[v]=totalProjectDeviations[v]+subTaskParameters[i][v];
	}
	// now calculate expected difference factors for each deviation parameter
	expectedProjectDifferenceFactors[v]=
	totalProjectDeviations[v]/(float)totalSubTasks;
}
```

However, it is unlikely that all tasks will contribute equally to the deviation parameters. They will probably have quite different values. The real cost variance and duration variance values for each task are called the actual deviations (ADd). For each deviation parameter, the difference between each task's expected deviation value and its actual deviation value, is the significance of deviance (SDt). The value of a significance of deviance indicates the significance of this task to the current total value for this deviation parameter in this project. The difference between the expected and actual deviations (the significance of deviance) is calculated by subtracting the expected from the actual.

The following C++ code exert demonstrates Procession's calculation for significance of deviance and its application to the weightings:

```
for(i=0; i<totalSubTasks; i++)
{
        for(v=0; v<theNumberOfDeviationParameters; v++)
        {
                // adjust deviation parameter weightings by current significance
                subTaskWeightings[i][v]=subTaskWeightings[i][v]
                + subTaskParameters[i][v] -
                expectedProjectDifferenceFactors[v]);
        } //close deviations loop
} //close subtasks loop
```

Thus, the calculations for the example in table 1 would be as follows:

Plumbing:
Cost 4 (ADt) – (6/2= 3 (EDd)) = a significance (SDt) of 1 (therefore increase weighting)
Duration 1 (ADt) – (4/2= 2 (EDd)) = a significance (SDt) of -1 (therefore decrease weighting)

Electrical:
Cost 2 (ADt) – (6/2= 3 (EDd)) = a significance (SDt) of –1 (therefore decrease weighting)
Duration 3 (ADt) – (4/2= 2 (EDd)) = a significance (SDt) of 1 (therefore increase weighting)

Procession would draw the following conclusions from these calculations and colour the data surface accordingly:

- Plumbing is currently making a greater than average contribution to total cost variances and its weighting value for this deviation parameter will be increased by the significance value.
- Plumbing is currently making a lower than average contribution to total duration variances and its weighting value for this deviation parameter will be decreased by the significance value.
- Electrical is currently making a greater than average contribution to total duration variances and its weighting value for this deviation parameter will be increased by the significance value.
- Electrical is currently making a lower than average contribution to total duration variances and its weighting value for this deviation parameter will be decreased by the significance value.

4 THE LEGACY ARCHIVE

The Legacy Archive file contains a set of deviation parameter weightings for each task. On running procession for the first time, all of the weightings are set to a default value. The total number of significance weighting values corresponds to the total number of tasks multiplied by the total number of deviation parameters. Weighting values are increased or decreased by Procession, according to the relevant significance values (see last section). A positive significance value indicates that a task is making a greater than average contribution to the total for a given deviation parameter. Therefore, the corresponding weighting is increased. A negative significance value indicates that a task is making a smaller than average contribution to the total for a deviation parameter. The weighting is then decreased.

When developing Procession, two different strategies were tried for calculating the legacy archive weightings. In the first version, there was not a graduated correlation between the significance value and its corresponding significance weighting. Changes to the weighting were only of one unit, in response to the sign (+ or -) of the significance value. The weightings did not change in proportion to the significance value. This meant that two deviation values, both above the expected deviation but to widely varying degrees (i.e. a small and large peak in the data-surface), were being assigned the same weightings. The calculation for this was implemented as follows:

```
if (deviance significance (SDt) is a positive value)
{
weighting++
}
else
weighting--
```

The prototype of Procession used for evaluation featured an improved calculation, where the weighting changed in response to the size of the significance value:

```
weighting = weighting + deviance significance (SDt)
```

The legacy archive is an ASCII text file, stored in the root of the primary Windows hard drive (usually C:\ ). Each line in the legacy archive file is space-delimited and has the format: task name, weighting one (cost variance), weighting two (schedule variance), weighting three (variance at completion), weighting four (duration variance) and weighting five (work variance).

For the project example in table 1, the legacy archive might look like this:

```
#Legacy Archive file created by
#P r o c e s s i o n application
#1999/2000
#Do Not Edit!
#file format:
#<TASK NAME> <PARAMETER 1 WEIGHTING>
# <PARAM 2 WEIGHTING>...etc.
#Last Updated: Fri Mar 20 15:00:42 2000
#------START DATA-----
Plumbing 46 57 32 48 51
Electrical 50 56 35 67 77
```

5 CONCLUSIONS

The next phase of the research will focus on implementing the software evaluation methodology. It is hoped that the mean RSL satisfaction scores, will be seen to increase while moving from their current reporting methods and through the Procession prototypes. In addition, it would be very positive to report types of informational deductions, made possible by Procession, that were not present with clients' traditional reporting strategies.

The planning data from a real social housing project has been obtained. Risk analysis techniques, such as Monte Carlo simulations, have been applied to the data set. This has identified specific tasks associated with high levels of 'unacceptable risk'. Using this analysis, project scenarios have been simulated, which will be used with RSL volunteers to evaluate Procession's reporting capabilities. Hypotheses have been decided for the experiment and the RSL interview protocols will be analysed using the statistical 't-test for the significance of the difference between the means of two groups'.

The Significance of Deviance Algorithm is an initial attempt to simulate an extremely complex relationship between project tasks and resources. Further work might consider other contributory factors. For example; the interactions between specific named tasks, the relationship between deviations and project outcome, the temporal location of tasks within the project and the relative sizes of tasks (in terms of both cost and duration).

6 ACKNOWLEDGEMENTS

This work was supported by a UK Office of Science & Technology Foresight Challenge Award. The Virtual Reality Centre For The Built Environment is an interdisciplinary initiative between University College London, Imperial College and a consortium of UK construction industry partners.

7 REFERENCES

Faraj, I., Alshawi, M., Aouad, G. et al. (1998) The Implementation of the IFC in a Distributed Computer Integrated Environment in *Product and Process Modelling in the Building Industry, the proceedings of ECPPM '98*, R. Amor (ed.), Building Research Establishment, UK, pp.187-198.

Kagioglou, M., Aouad, G., Cooper, R. and Hinks, J. (1998) The Process Protocol: Process and IT modelling for the UK construction industry in *Product and Process Modelling in the Building Industry*, the proceedings of ECPPM '98, ed. R.Amor, Building Research Establishment, UK, pp.267-276.

Lowry, R. (2000) *Concepts and Applications of Inferential Statistic* s, available from: faculty.vassar.edu/~lowry/webtext.html, [accessed April 2000].

North, S. (1999) Enter the Virtual Building Site, *Housing*, April 1999, The Chartered Institute of Housing, UK, p.40.

North, S. (2000) Procession: using intelligent 3d information visualisation to support client understanding during construction projects in *Visual Data Exploration and Analysis VII*, Robert F.Erbacher, Philip C. Chen, Jonathan C. Roberts, Craig M. Wittenbrink (editors), Proceedings *of SPIE* (The International Society for Optical Engineering) *Vol. 3690 (24th-26th January 2000, San Jose, USA)*, ISSN 0277-786X, ISBN 0-8194-3578-3, pp.356-364.

Shroeder, W., Martin, K., Lorensen, B. (1998) *The Visualization Toolkit*, Prentice Hall, USA.