

EVALUATION CODE REVIEW IES

AUTHOR(S) : Peter Mazereeuw
DOCUMENT NUMBER : IES-Evaluation.Doc
VERSION : 1.0
SOURCE : Atos Origin
STATUS : FINAL
DOCUMENT DATE : 11 May 2003
NUMBER OF PAGES : 8

RELEASED BY : P. Mazereeuw

INITIALS:

Contents

| | |
|--|---|
| 1 Introduction..... | 4 |
| 1.1 Purpose..... | 4 |
| 1.2 Readers..... | 4 |
| 1.3 References..... | 4 |
| 1.4 Overview..... | 4 |
| 2 Background..... | 4 |
| 3 Evaluation Method..... | 5 |
| 4 Results..... | 5 |
| 4.1 Error Handling..... | 5 |
| 4.2 Referential Integrities..... | 6 |
| 4.3 Primary Keys with automatic numbering..... | 6 |
| 5 Conclusion..... | 7 |
| Appendix A – About..... | 8 |

List of changes

| VERSION | DATE | DESCRIPTION | AUTHOR |
|---------|-----------|-------------------------------------|-----------------|
| 0.1 | 2-4-2003 | Initial | Peter Mazereeuw |
| 0.2 | 16-4-2003 | Changes after internal review | Peter Mazereeuw |
| 1.0 | 11-5-2003 | Final report | Peter Mazereeuw |
| 1.1 | 6-6-2003 | Added references author and company | Peter Mazereeuw |

1 Introduction

1.1 Purpose

Powervote has assigned Atos Origin to perform an evaluation of the main issues raised by a Code Review on the application IES (“Integrated Election Software”) that was developed by Powervote for the Government of Ireland.

This report describes the results of that evaluation.

1.2 Readers

Readers of this report are Powervote, stakeholders from the Government of Ireland and John Pugh of “Nathean” as the last reviewer of the code. From the readers knowledge of the IES application is expected.

1.3 References

| Ref. | Titel | Version | Date | Author(s) |
|------|--|----------|----------|---------------------------|
| [1] | Code Review of the Powervote Electronic Voting System | | 14-12-01 | John Pugh, Robert Doherty |
| [2] | Evaluation of Integrated Election Software Development Environment | | 14-12-01 | John Pugh, Robert Doherty |
| [3] | Evaluation of Integrated Election Software Database | | 14-12-01 | John Pugh, Robert Doherty |
| [4] | Evaluation of Random Number Generation in the Powervote Electronic Voting System | | 14-12-01 | John Pugh, Robert Doherty |
| [5] | Code Review Guidelines for Powervote Electronic Voting System | | 14-12-01 | John Pugh, Robert Doherty |
| [6] | PMI's Software's Pseudo-code for Code Reviewing | | 14-12-01 | John Pugh, Robert Doherty |
| [7] | Code Review of IES build 2002-0096 | Rev. 1.0 | 20-09-02 | John Pugh (Nathean) |
| [8] | Analysis of developer comments | | 06-12-02 | John Pugh |
| [9] | Powervote's Responses to Architectural Review of IES | | dec 02 | Jan Janson |
| [10] | Powervote Responses to Code Review of IES build 2002-0096 | | dec 02 | Jan Janson |
| [11] | Technical Documentation IES (CD) | | 03-05-02 | Powervote |

1.4 Overview

The report starts in Chapter 2 with a summary of the scope and background of the review activities. Chapter 3 contains the method of the evaluation, which activities have been performed and which resources have been used and in Chapter 4 the results of the evaluation are described. Finally in Chapter 5 the conclusion of the evaluation is stated.

2 Background

Powervote sells already for years voting systems to the Dutch and German markets, called “ISS”. These systems are well performing and maintained by a professional staff. All systems have in common that they use comparable algorithms for the votings.

About three years ago the ISS system was also proposed to the Irish Government. As the Irish voting system differs significantly from those more current to ISS, it was decided to alter the program to support this different voting. This resulted in “IES” for the Irish market.

After completion of the first release the Irish Government asked for a code review on the new algorithms. This request was granted and carried out by John Pugh (see references). In the review [ref. 1] and a following review in 2002 [ref. 7] three major issues were addressed:

1. No proper error handling
 2. No referential integrity
 3. No use of auto numbered primary keys
-

These issues are not to be considered as showstoppers, but as they are reported as major issues, the report states to resolve them before the final roll-out.

According to Powervote the functionality is not influenced by these issues nor do they have effect on the quality of the system. Therefore, a second opinion has been asked to either confirm or deny the importance of resolving the issues on short notice.

3Evaluation Method

The issues raised by the Code Reviews are evaluated by Atos Origin on request of Powervote. After an initial interview with the programmer Atos Origin studied all available review reports as listed in § 1.3 and had all questions answered by the programmer in a thorough interview. Next a quick scan of relevant parts of the code was carried out. The results of the evaluation are stated in this report.

4Results

The evaluation focuses solely on the three major issues as they were stated in the Code Review Reports. In the sections below these issues are evaluated separately.

4.1 Error Handling

It is good practice to enclose code that is vulnerable to runtime errors into a *try – except* clause, such as where file access is needed. This makes it possible to handle any or specific errors in a managed way. According to the Code Review error handling is not implemented everywhere in the application which can lead to system crashes or unclear error messages.

The current version of the code already includes quite a lot of error handling coverage, but – as stated – not everywhere. The locations in the code mentioned in the reports however are not explicitly vulnerable points for runtime errors so error handling can be covered on a higher level, i.e. carried out in the calling methods. Including any extra clauses is not needed to be of a high priority in these cases.

On the point of logging and/or presenting any runtime errors Powervote suggested, and already implemented in some cases, the use of EurakaLog, an extensive logging tool. It is recommended to extend the use of this tool throughout the whole system. But, as it does not affect the functionality or the current stability, it is not necessary and not obligatory for getting the system in production.

Listing 1 shows the use of the *Try – Except* clause in the IES application. This listing is a good representation of how error handling is implemented in most of the modules of the system. It also shows that any error statement is presented in a very raw format. EurakaLog will improve this significantly.

```
procedure Calculate_Count_Referendum;
var
...
begin
  Screen.Cursor:=crHourGlass;
  Try
    frmChild := (frmMain.ActiveMDIChild as tfrmChild) ;
    With frmMain,frmChild.dmElection do begin
      Try
        .....

      Finally
        Gauge_End;
        Screen.Cursor:=crDefault;
      end;
    end;
  Except
    On E:Exception Do
      Raise EMathError.createfmt("Error in Unit Calc_Count_Referendum, Procedure Calculate_Count_Referendum: %s: %s",
[E.ClassName,E.Message]);
  end;
end;
end.
```

Listing 1. Code snippet of IES showing *Try – Except* and *Try – Finally* clauses

4.2 Referential Integrities

The third major issue raised by the Code Review is that, instead of using the integrated possibilities of MS Access, relations between tables and (thus) referential integrities are implemented directly in the code.

Relations and also referential integrities are used to ensure consistency of data, to prevent orphan records and minimize redundancy. MS Access offers a nice, quick and stable way to implement these relations and integrities, but it is not mandatory to use these features when using the database in any system.

Due to historical decisions Powervote implements the relations and integrities in the code instead of in the database. This lowers maintainability and increases risks of failures during development. But as IES is already operational and stable and maintained by Powervote self, it is recommended to keep the existing code.

4.3 Primary Keys with automatic numbering

In MS Access primary keys can have the so called “Autonumber” type. This means that on insertion of records Access gives that record an automatic generated and unique number in that field. IES does not use this feature; unique numbers for the primary keys are generated in the code. According to the Code Review the use of the autonumber field type is a common practice and the lack of it is qualified as a major issue.

The use of the “Autonumber” type is widely used in application, but it cannot be considered as best practice. A set back of this feature for example is that you do not have any control over the value and/or format. The use of a custom unique value for the primary key does give this control and should be used in all cases any control over the primary key field is necessary. As far as discovered in the quick scan IES does not use any custom format for her counters, but even a simple custom counter does not improve nor degrade any application. Alterations on this issue are therefore not necessary for either quality attribute and therefore not recommended to implement.

5Conclusion

The Code Review of the IES application mentions three major issues as remarks on the IES application. Based on the quick scan and the current state of the application none of these remarks must be regarded as blocking issues and are therefore no obstacles for the application to go into production.

It is possible that a pinpointed extensive review of the code will raise more issues, but as the application is proven stable and covers all required functionality, resolving these issues will most probably not improve the quality of the software enough to validate the costs for such a review.

Appendix A – About

About Atos Origin

Atos Origin is an international information technology services provider. Its business is turning client vision into results through the application of consulting, systems integration and managed operations, including outsourcing and on-line services. In August 2002, Atos Origin acquired KPMG Consulting in the UK and The Netherlands, trading as Atos KPMG Consulting. The company generates annual revenues of EUR 3 billion and employs 28,000 staff in 30 countries. The Group's client list includes major companies such as ABN AMRO, Akzo-Nobel, Alstom, BNP Paribas, BP, Euronext, Fiat, ICI, ING, KPN, Lucent, Philips, Renault, Royal Bank of Scotland, Saudi Aramco, Shell, UBS Warburg, Unilever, Vivendi Universal, Vodafone and Wolters Kluwer.

For more information, please visit the company's web site at <http://www.atosorigin.com>

About ir. Peter Mazereeuw

Peter Mazereeuw is a senior ICT consultant and architect at the Department of Business Application Solutions of Atos Origin. He is specialized in evaluating software architectures and implementing development processes and is a technical authority on the .NET and J2EE frameworks. Peter is working in ICT from 1982, first in the field of Applied Physics at the University of Groningen where he also lectured physics and ICT didactic skills. From 1994 till 1999 he worked as teacher and IT consultant in the Netherlands Antilles and is employed at Atos Origin since 2000.

Peter can be contacted through email: peter.mazereeuw@atosorigin.com
