Chapter IX

A Graphical Approach for Reducing Spreadsheet Linking Errors

Charles M. Morrison
University of Wisconsin–Eau Claire, USA

Joline Morrison
University of Wisconsin–Eau Claire, USA

John Melrose
University of Wisconsin–Eau Claire, USA

E. Vance Wilson
University of Wisconsin–Milwaukee, USA

ABSTRACT

Spreadsheet programs are deceptively simple tools that are widely used by end user developers in organizations. However, recent studies have shown that spreadsheets often contain significant, decision-affecting errors. One study that addressed “linking errors,” i.e., incorrect references to spreadsheet cell values on separate work areas, found these errors to be a major error source in complex spreadsheets that use distinct work areas spread across
multiple worksheets. This paper describes a code inspection approach that visually represents the structure of a linked spreadsheet and graphically identifies linked cells and their sources. We tested this approach in an experimental study where subjects created a complex spreadsheet. Results indicate that subjects who used the approach made significantly fewer errors and experienced no decrease in speed of spreadsheet production or satisfaction with the production process.

**INTRODUCTION**

Spreadsheets are widely used in business to develop simple “scratch pad” or throwaway applications, as well as to develop large and complex applications that guide critical corporate decisions (Panko, 1998; Panko, 2000). Managers typically create spreadsheets rather than personal databases for large, data-intensive applications because they perceive spreadsheets as being easier to use and because they are more familiar with spreadsheets.

When spreadsheets were first introduced, developers solved complex problems using a single large worksheet. An elaborate worksheet might involve hundreds or even thousands of columns and rows, and often the spreadsheet developer found it difficult to keep track of what was going on in distant regions of the spreadsheet. As spreadsheets evolved, vendors enabled users to partition large spreadsheets by creating inter-linked worksheets that allow results from one worksheet to act as inputs for calculations in another worksheet. Today, related worksheets are combined into sets called workbooks. Links between different worksheets are called inter-worksheet links, and workbooks with links among worksheets are called **three-dimensional** spreadsheets. Every major spreadsheet vendor now supports three-dimensional spreadsheets with inter-workbook links. These links allow users to decompose complex spreadsheets into smaller worksheets that are more easily understood. Unfortunately, three-dimensional spreadsheets also introduce potential for inter-worksheet linking errors (Janvrin & Morrison, 2000).

For example, suppose a user mistakenly thinks one or more cells on one worksheet are linked to another worksheet, when in fact the actual data values were copied. If data values are changed on the source worksheet, the values on the destination (target) worksheet do not change, and incorrect data appears on the target sheet. Similarly, a user might delete a cell or a range of cells on a source worksheet that are linked to a second target worksheet without realizing that the cells were linked to the target sheet. If the worksheet uses relative addressing, the data values that are now located where the deleted cells were located are used in
Application Controls for Spreadsheet Development
www.igi-global.com/article/application-controls-spreadsheet-development/55669?camid=4v1a