Title: A METHOD AND SYSTEM FOR MAPPING BUSINESS PROCESSES

Abstract: A method and system for automatically mapping a business process employing a computer application such as a spreadsheet comprising analysing usage of the computer application by a user through categorising the instantaneous or historical usage by the user according to activities performed by the user in the computer application and quantifying the time spent by the user at each activity from the times recorded for each action performed by the user in the computer application, the times tracked for each action performed by the user being extracted from the event recording model of the computer application.
UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— with international search report (Art. 21(3))
— before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))
A Method And System For Mapping Business Processes

Introduction

This invention relates to a method and system for mapping business processes and, in particular, to a method and system for mapping business processes by automatically mining and quantifying previously invisible end-user computer application and system usage.

Background of the Invention

Business Process Mapping or modelling (BPM) is used to identify risk; to analyze and improve productivity and efficiency; and as a key part of planning for new software systems. Current BPM approaches are largely manual. The core principle is that an analyst meets with all members of staff and information is captured through structured interviews, questionnaires and data sampling.

Known BPM processes are time consuming, expensive and slow. This means that BPM analysis tasks are often only conducted at the start of major IT strategy projects. In a fast moving business context, a business process can evolve or change during large projects. In addition, due to the lack of automation, BPM analyses are prone to human error as employees or users are not always aware of what processes they follow or indeed what computer files they use. Even with the attention of an experienced analyst, many details of a business process can remain hidden, unknown and unquantifiable.
Critical to accurate BPM analyses is the manner in which computer applications and systems (hereinafter referred to collectively as computer applications) are used. As indicated above, in general, the types and duration of actions performed by end users of computer applications and systems in an organisation can be impossible to identify and quantify. This can be due to the large volumes of data involved, poor regulatory environments and inconsistencies in how such computer applications and systems are utilised and managed within organisations.

For example, spreadsheets are widely used dynamic computer applications in which data is stored and manipulated. Most companies, be they large or small, make heavy use of spreadsheets either in formal applications or as a supplement to conventional software to integrate disparate financial reporting packages. Spreadsheets are essentially a form of end user computing and are available in many formats such as, *inter alia*, Excel (Trade Mark), StarOffice (Trade Mark), Google Spreadsheets (Trade Mark) and Lotus 1-2-3 (Trade Mark).

Spreadsheets, like many applications, have the advantage of being easy to use, flexible and powerful. However, as a result of this ease of use and flexibility, spreadsheets also suffer from a number of well documented disadvantages. For example, spreadsheet usage is known to be susceptible to bugs whilst the widespread use of spreadsheets in organisations can be chaotic and uncontrolled giving rise serious error potential, poor data security, poor auditing and poor productivity by users of spreadsheets as a result of difficulties understanding complex, poorly structured spreadsheets and the entering of numerical information.
The risks associated with spreadsheets are well documented. For example, EUSPRIG, the European Spreadsheet Risks Interest Group, was established to specifically assist in quantifying and addressing the risks inherent in spreadsheet usage.

Cautionary examples of the risks posed by spreadsheet usage include multi-million Euro frauds committed at financial institutions by employees entering false data in spreadsheets or clerical errors in spreadsheets giving rise to false financial forecasting. Less sensational, but equally costly, is the poor user productivity associated with inefficient use of spreadsheets in organisations.

Two categories of systems have been used to address the aforementioned risks - enterprise spreadsheet management systems and spreadsheet auditing systems. Enterprise spreadsheet management systems are designed to take control of the spreadsheets within an organization and apply a level of discipline to their usage. Examples of enterprise spreadsheet management systems include ClusterSeven Enterprise Spreadsheet Management (Trade Mark) and Prodiance Spreadsheet Compliance (Trade Mark). Spreadsheet auditing systems are designed to look for errors in individual spreadsheets and to assist users in remediating them. An example of a spreadsheet auditing system is Spreadsheet Detective (Trade Mark).

US Patent Specification No. 7,231,593 describes a method for managing spreadsheets in which a standardized spreadsheet is employed and data statements are compared to identify changes made to a spreadsheet. US Patent Publication No. 2010/0049565 discloses a method of computing spreadsheet risk in which
spreadsheet metadata is collected and analysed to calculate the likelihood of error or risk associated with the spreadsheet.

In some instances, enterprise spreadsheet management systems and spreadsheet auditing systems can log changes to a spreadsheet into a separate spreadsheet and can track changes to a spreadsheet to cell level and record the time of the change and by whom. Enterprise spreadsheet management systems can also in some instances track spreadsheet status at moments in time - i.e. take "snapshots" of usage. However, such systems still fail to identify and categorise activities by users and the duration of such activities.

Accordingly, enterprise spreadsheet management systems and spreadsheet auditing systems fail to discover and address the costs associated with poor user productivity associated with inefficient use of spreadsheets in organizations i.e. the costs associated with the inefficient collecting, integrating and defining of data in the spreadsheet environment. In short, known systems are directed to containing and managing the risk of uncontrolled spreadsheet usage but fail to identify, target or address the cost of the underlying spreadsheet usage processes.

Similar problems are encountered in relation to mining and mapping usage of other computer applications/ systems such as email, word processing, database client — interface applications and database server systems.
Summary of the Invention

According to the invention there is provided a method for automatically mapping a business process employing a computer application comprising:

- analysing usage of the computer application by a user through categorising the usage by the user according to activities performed by the user in the computer application, and quantifying the time spent by the user at each activity from the times recorded for each action performed by the user in the computer application.

Preferably, the times recorded for each action performed by the user are extracted from the event recording model of the computer application.

In one embodiment of the invention, the business process is mapped by categorising and quantifying instantaneous actions of the user. Preferably, the usage of the computer application by a user is written to a usage data text file.

Suitably, the usage data text file is collected manually for categorisation. Alternatively, the usage data text file is collected automatically for categorisation.

Advantageously, separate usage data text files are consolidated into a consolidated usage data text file. Preferably, the consolidated usage data text file is manipulated to produce classified usage report data.
In an alternative embodiment of the invention, the business process is mapped by categorising, quantifying and comparing historical actions of the user. Preferably, the step of comparing the historical actions of the user is performed on two versions of a computer application data file.

Suitably, the step of comparing the historical actions of the user is performed on a summary version of the computer application data file.

Advantageously, the summary version of the computer application data file comprises an encoded string summary.

Preferably, the encoded string summary is prepared at regular intervals and the encoded string summary is written to a status text file.

Suitably, the status text file is collected manually for categorisation. Alternatively, the status text file is collected automatically for categorisation.

Preferably, separate status text files are consolidated into a consolidated status text file and, more preferably, the consolidated status text file is manipulated to produce classified usage report data.

In a preferred embodiment of the invention, the step of quantifying the time spent by the user at each activity comprises extracting the time of a first action from the event recording model of the computer application, extracting the time of a second action from the event recording model of the computer application, calculating the time
duration between the first and second events and allocating the time duration to an activity.

Preferably, the business process is mapped by a usage agent installed on a computer device to capture the activities performed by the user.

More preferably, the usage agent is installed on all user computer devices in a business.

Suitably, the computer device comprises a personal computer, a laptop, a tablet or a smartphone.

In an alternative embodiment of the invention, the usage agent is installed on a server on which computer application data files are stored.

Suitably, the usage agent is communicable with a management computer on which the captured user activities are analysed and, advantageously, the usage agent is communicable with the management computer via a Local Area Network, a Wide Area Network or the internet.

In one embodiment, the method of the invention further comprises the step of displaying the categorised usage and quantified time in textual, tabular or graphical form.

Preferably, the computer application comprises an end user computer application.
More preferably, the end user computer application is selected from the group comprising spreadsheet applications, email applications, word processing applications and database client-interface applications.

Most preferably, the end user computer application comprises a spreadsheet application.

The computer application can comprise a database server system.

The invention also extends to a computing device comprising:

- a processor for controlling operation of the computer device;
- computer applications having corresponding event recording models, and
- a memory for storing computer executable instructions that cause the computing device to extract the times recorded for each action performed by the user from the event recording models of the computer applications.

Preferably, the computer executable instructions that cause the computing device to extract the times recorded for each action performed by the user from the event recording models of the computer applications are contained within an addIn usage agent.

More preferably, the usage agent captures and records instantaneous keystrokes of the user from the event recording model to determine each action performed by the user.
Suitably, the usage agent generates an encoded string of the computer application
data files and, advantageously, the usage agent is communicable with a
management computer. Suitably, the usage agent is communicable with the
management computer via a Local Area Network, a Wide Area Network or the
internet.

The invention also extends to a computer program comprising computer program
code means adapted to perform all the steps of the method hereinbefore defined.

Advantageously, the computer is embodied on a computer readable medium.

In the first embodiment of the invention, the usage agent captures and records
instantaneous keystrokes of the user from the event recording model to determine
each action performed by the user.

The invention also extends to a computer system incorporating a method for
automatically mapping a business process as hereinbefore defined.

In a further embodiment, the invention relates to a method for mining computer
application usage by a user comprising:

- analysing usage of the computer application by a user through
categorising the usage by the user according to activities performed by
the user in the computer application, and
quantifying the time spent by the user at each activity from the times recorded for each action performed by the user in the computer application, and displaying the categorised and quantified usage in textual, tabular or graphical form.

The method and system of the invention for mining and mapping computer application and system usage categorises and measures usage by separating and allocating the time a user interacts with a computer application or system into a set of defined activities to identify and highlight previously invisible or unseen non-productive or inefficient activities by users across an organisation so that targeted corrective action can be taken and the organisation's productivity improved. The method and system of the invention therefore enables the mapping of previously unseen, hidden and unknowable business processes associated with use of computer applications and systems.

Through the use of trigger responsive usage agents which can be installed on individual computer devices and/or servers as appropriate high level information on usage can be obtained based on the low level specific actions performed by users.

The use of a spreadsheet sensitive usage agent in accordance with the invention installed on all computing devices within an organisation facilitates the identification and monitoring of all spreadsheet usage e.g. even where spreadsheet files are only stored locally on a device, the usage agent will detect and identify activity on the spreadsheet for analysis. Accordingly, the usage agent overcomes problems
associated with informal approaches to the design, development, usage and storage of spreadsheet files including, firstly, where files are stored in a variety of locations: including personal computers, laptops, different physical locations and network drives, secondly, where file naming conventions are lacking or not enforced and, thirdly, where file versioning is ad-hoc, non-existent, or inconsistent.

Where the method and system of the invention is employed to analyse instantaneous actions, it is not necessary to analyse the large amounts of spreadsheet files (and indeed the potentially multiple versions of the same file) that can be stored in an organisation when analysis is required as the analysis is performed at event or activity level.

Moreover, the action or trigger responsive usage agents employed in the method and system of the invention can be employed with all spreadsheet formats to identify usage regardless of format.

The method and system of the invention is also adapted to associate navigation by a user through a spreadsheet with different categories of activities as required e.g. navigation associated with general reviewing of a spreadsheet or navigation associated with specific activities performed on the spreadsheet can be characterised appropriately i.e. relevant navigation can be associated with the relevant activity involving navigation before or after the activity. Accordingly, the method and system of the invention enables comprehensive business process mapping.
The system and method of the invention enables organisations and their advisors to quantify the time spent by users on collecting, integrating, defining, and analysing data. In general, end users and data analysts are paid to analyse data - not to collect, integrate and define it. Accordingly, the system and method of the invention facilitates the automatic mapping and improvement of business processes through the discovery, targeting and elimination of non-productive end user activities in an organisation giving rise to substantial cost savings and increased productivity.

In addition, the method and system of invention enables the mining of previously unobtainable data on computer application usage which enables the performance of computer application usage certification and risk assessment studies on the use of computer applications such as spreadsheets. The usage data enables businesses and organisations to identify the use of resources and to make appropriate changes in resource allocations to increase efficiency in the use of computer applications.

Moreover, the method and system for mapping business processes of the invention has particular utility in assisting in the elicitation and design of what is required from future or replacement information technology systems within organisations.

**Brief Description of the Drawings**

The invention will now be described, by way of example only, with reference to the accompanying drawings in which:
Figure 1 is a block diagram of a computing device in accordance with the invention;

Figure 2 is a block diagram of a networked system of the invention in which client laptops and/or PC's are in communication with a Usage Data Centre (UDC) via a Local Area Network (LAN), a Wide Area Network (WAN) or the internet;

Figure 3 is a topological diagram of a first embodiment of the method and system of the invention for the instantaneous discovering and mapping of spreadsheet usage in which the instantaneous spreadsheet actions of a user are captured and analysed;

Figure 4 is a topological diagram of the algorithmic process employed to generate the Usage Data Text (UDT) files described in Figure 3;

Figure 5 is topological diagram of the algorithmic process employed to generate the Classified Usage Report Data (CURD) table from the UDT files as outlined in Figure 3;

Figure 6 is a flow diagram of the algorithm employed in the generation of the UDT files described in the topological diagram of Figure 4;

Figure 7 is a flow diagram of a first part of the usage classification algorithm of Figure 3 employed to analyse the UDT files to create a Temporary Classified Data Report (TCDR) file;
Figure 8 is a flow diagram of a second part of the usage classification algorithm of Figure 3 in which data in the TCDR created in accordance with Figure 7 is manipulated to generate the CURD table;

Figure 9 is a topological diagram of a second embodiment of the method and system of the invention for discovering and mapping spreadsheet usage in which two versions of spreadsheet files are compared to create the CURD file;

Figure 10 is a detailed flow diagram of the generation of an encoded string of the spreadsheet files in the form of a Spreadsheet Status Text (SST) file by the usage comparison and classification algorithm described in Figure 9;

Figure 11 is a flow diagram of the analysis and manipulation of the SST file data to create a CURD file by the usage comparison and classification algorithm, and

Figures 12 to 17 are user interface screenshots from a computer system in accordance with the invention for discovering and mapping spreadsheet usage in which the method and system of the invention described in Figures 1 to 11 has been employed to map business processes via the analysis of spreadsheet usage in an organisation.
Detailed Description of the Invention

Although the following description relates generally to the use of the present invention with spreadsheet applications, it will be appreciated that the invention is also equally applicable to other computer applications and systems including but not limited to email, word processing, database client-interface applications and database server systems generally. Accordingly, the term "computer application" should be construed to include all such applications and systems.

Figure 1 is a block diagram of a typical computing device 100 for use with the systems and methods of the invention. The computing device 100 is made up of a processor 101, RAM 102, ROM 103, memory 104 and an input/output module 105 for operating the computing device 100.

Application data in accordance with the invention is stored in RAM 102 in the form of tasks or applications 106 when the computing device 100 is in operation and software tasks are being performed by the computing device 100. The computing device 100 can include various forms of computer readable media.

Applications 107, an operating system 108 and data 109 can be stored in the memory 104.

Figure 2 is a block diagram of a networked system 200 of the invention in which the computing devices 100 of Figure 1 in the form of client or user PC's 201 and/or laptops 202 adapted to implement the systems and methods of the invention as
outlined further below are in communication with a Usage Data Centre (UDC) 204 via a Local Area Network (LAN), a Wide Area Network (WAN) or the internet 203.

As will be appreciated by those skilled in the art, smartphones, tablets and other mobile devices can be used in combination with or in place of the PC's 201 and/or laptops 202. Moreover, spreadsheet usage can be accessed via browser technology with such devices. Moreover, the invention could be installed as an Addln to a browser for browser based spreadsheets in cloud computing environments.

Usage software in accordance with the invention, hereinafter referred to as a Usage Agent, communicable with the UDC 204 is installed on each PC 201 and laptop 202 to discover, categorise and map spreadsheet usage by the users of the PC's 201 and laptops 202.

In the second embodiment of the invention described further below in Figures 9 to 11, the (comparison) Usage Agent can be installed on a file server on which the spreadsheets are saved if desired as well as the user PC's 201.

Usage results are transmissible to a client management computer 205 also via the LAN, WAN or internet 203. As will be appreciated by those skilled in the art, the PC's 201 and laptops 202 can be located at disparate locations, e.g. London, Dublin and Paris as shown in Figure 2, within a client organisation so that global analyses of spreadsheet usage can be performed.
Figures 3 to 8 describe a first embodiment of the invention in which spreadsheet usage at the PC's 201 and laptops 202 of Figure 2 is categorised and measured by separating and allocating the time a user interacts with an individual spreadsheet into each of a set of defined activities to identify, target and highlight inefficient spreadsheet usage by users across an organisation so that the invention illuminates the process of usage detailing which users accessed files, when the files were accessed and for how long. Instantaneous actions of a user are analysed while for the purposes of the present embodiment an action is defined as any instantaneous interaction with a spreadsheet that results in a change to the spreadsheet. A change to the spreadsheet also includes any change in the position of the cursor and any cell or other object selected, highlighted or activated in the spreadsheet.

In the following description, periods of spreadsheet usage are segmented by actions in the following action categories although the invention can be adapted to segment actions into different categories as required:

- **A1. Review:** examination of the spreadsheet through selection and investigation of cells and other spreadsheet objects without changing the location, formatting or detail of the objects including navigation through the spreadsheet;
- **A2. Data Entry:** entry by keyboard, mouse or other means of numeric data to cell objects.
- **A3. Formula Development:** creation or alteration by keyboard, mouse or other means of formulas in cell objects, including naming of a cell;
- **A4. Object Development:** creation or alteration of any spreadsheet object other than a cell object;
A5. Design: Change in layout of spreadsheet through movement of cell, sheet or other object from one location to another;

A6. Formatting: Any change to a cell object other than a change to the value or the formula;

A7. Advanced Development: Creation or alteration of associated embedded software code;

A8. Output: Creation, saving, copying of spreadsheet files;

A9. Opening of a spreadsheet;


A11. Idle (periods of inactivity).

Action types can be labelled in accordance with the above action types from A1 to A11.

The above action types are collapsible into three general action categories as follows:

1. Review: A1 + A8;

2. Development: A3 + A4 + A7;


As will be appreciated by those skilled in the art, actions performed by users, such as key presses or mouse actions, trigger the various spreadsheet technologies referred to above e.g. Excel (Trade Mark), StarOffice (Trade Mark), Google Spreadsheets (Trade Mark) etc. to raise events in the software which are analysed.
to extract information regarding the user action. In the case of the spreadsheet technology used as an example here, namely Microsoft Excel (Trade Mark), there exists an extensive set of events, known as the Excel Event Model, which are exploited and analysed to determine information on actions.

Accordingly, a user action initially triggers events that are raised by the operating system to cover basic actions such as key presses and mouse actions. The system of events is usually, but not always, designed in a hierarchical fashion, and an event record generated by the operating system is passed in succession to each active computer application (such as Excel and applications such as word processors, email clients and database client-interface applications). Each aspect of the operating system software, and subsequent application program, on notification of the event by the operating system, can chose to act upon, or record these events. In turn, such events can then be handled by the spreadsheet application which can then raise a higher set of events, controlled and described through the Excel Event Model, which describe the characteristics of the spreadsheet interaction. Other computer applications (such as email, word processing and database client-interfaces), can also apply the notification of such events to their own internal event model and as such this invention could be applied to such systems as well.

Further computer systems have different models of event recording and management, for example database server systems, can trigger events based on the insertion, deletion, viewing, or modification of data held in tables or files. The method and system herein described can therefore be used in conjunction with such
systems to infer the behaviour and usage of users interacting with these system types to map business processes.

As indicated above, in the present embodiment, a usage agent in accordance with the invention is installed on user PC’s and laptops through add-in technology to, firstly, capture and record each keystroke and mouse action performed by the user on a spreadsheet and, secondly, categorise the actions to classify usage activity. It will be appreciated by those skilled in the art that where reference is made to Microsoft Excel (Trade Mark) in the following drawings and examples, other spreadsheet technologies can be substituted for Microsoft Excel (Trade Mark) without departing from the scope of the invention. As will be also be appreciated by those skilled in the art, various development environments and languages can be employed for the add-in technology without departing from the scope of the invention e.g. Microsoft Visual Studio, C# development language etc.

In the following description, multiple spreadsheets (or workbooks) may be open at any one time.

The entire time period of use of spreadsheets from the opening of the first spreadsheet to the closing of the last spreadsheet by a user is segmented by the actions defined above where the first action is a spreadsheet opening action A9. The algorithm of the invention records the time of the opening action or event A9 and then examines each action in turn based on the type of the current and previous actions and on the time, Tduration, between the previous action and the current action. The algorithm also uses three time parameters: Tcriticalidle, Tadd &
T_{critical} \leq T_{criticalldle}. In the algorithms described further below, T_{current} represents the time of the current action and T_{previous} represents the time of the previous action, thus T_{duration} = T_{current} - T_{previous}. The algorithm of the present embodiment may be summarised in two steps as follows, although the steps can be integrated if desired:

Step 1:

1. If T_{duration} \leq T_{criticalldle} then the time period from the previous event to the current one is assigned the activity associated with the type of the current event or action.
2. If T_{duration} > T_{criticalldle} then the period from T_{previous} to T_{current} - T_{add} is assigned as inactivity and the period from T_{current} - T_{add} to T_{current} is assigned the activity associated with the type of the current event.

Step 2:

1. The periods of review activity are examined in sequence, with consecutive periods aggregated to produce a (reduced) set of periods of review activity. If the time for an aggregated period is less than T_{criticalReview} then the aggregate period may be reassigned. If the period immediately following the aggregated period does not exist or the period immediately following is categorised as inactivity then the aggregated period remains categorised as review activity. Otherwise the aggregated period is assigned the activity of the period immediately following the aggregated period.
As indicated above, Figure 3 is a topological diagram of the first embodiment of the method and system of the invention 300 for discovering and mapping spreadsheet usage in which the instantaneous spreadsheet actions of the user are captured and analysed. As shown in the drawing, in step 301 the usage agent is first installed on each user PC 201 and laptop 202 (see Figure 2) running Microsoft Excel (Trade Mark) or other spreadsheet application. The system 300 is then configured 302 to select the users whose spreadsheet usage activity is to be collected and monitored. A folder and filename on the PC 201 or laptop 202 is selected in which the usage data is to be stored. The usage data is stored as a text document referred to as a Usage Data Text (UDT) file.

Upon opening of an Excel (Trade Mark) file by the user, operation 303 of data capture is commenced. More particularly, the UDT file is opened and usage information is collected, categorised and recorded 304 by the Usage Agent on a continuous basis and written to the UDT file also on a continuous basis. The connection to the UDT file is closed when the Excel (Trade Mark) application is closed by the user.

The UDT files of users are then collected 305, either manually, or more usually, automatically via a network script where the PC’s 201 or laptops 202 are in communication with the UDC 204 via LAN/WAN/internet 203 as described in Figure 2.

Usage categorisation 306 of the data contained within the UDT files is then performed at the UDC 204 by a categorisation application. Firstly, data contained
within the UDT files is consolidated by appending the UDT files to create a single
UDT file for analysis. An algorithmic usage categorisation process as outlined above
and in more detail below is then performed on the consolidated UDT file in which the
data is examined and manipulated on a row by row basis to generate a Classified
Usage Report Data (CURD) table 307 for reporting usage.

The CURD table is then subjected to analysis 308 where the CURD data is
summarised and visualised 309 through textual, tabular and graphical methods as
outlined further below and as illustrated in the screenshots of Figures 12 to 17.
Examples of the information extracted from the CURD data can include the amount
of time spent at each activity A1 to A11 for each spreadsheet opened by each user.
The analysis can also show when this activity took place and the sequence in which
the users conducted the activities on the files examined. The analysis can be
generated by client management via the management computer 205 as shown in
Figure 2 or by external consultants as required.

The completed analysis is then reported 310 to relevant personnel so that remedial
action can take place within the organisation to address poor productivity as a result
of improper or inefficient usage of spreadsheets within the organisation and also
reveals the business processes within the organisation in terms of the sequence and
timing of activities by users on files.

Figure 4 outlines in more detail the process employed in the generation of the Usage
Data Text files 304 of Figure 3. As shown in the drawing, as the user performs
spreadsheet operations 400, the user interacts with the spreadsheet 401 so that the
spreadsheet application raises software events in response to the actions of the user in conventional fashion.

The user action initially triggers events that are raised by the Operating System to cover basic actions such as key presses and mouse actions. These events can then be handled by the spreadsheet application which can then raise a higher set of events, controlled and described through the Excel Event Model, which describe the characteristics of the spreadsheet interaction.

The triggered events contain information, *inter alia*, on the type of event, the time of the event, the username of the current user and the spreadsheet workbook and the worksheet in use. Typical examples of such events in Microsoft Excel (Trade Mark) include:

- SheetPivotTableUpdate - triggered when a pivot table in Excel (Trade Mark) is altered;
- SheetSelectionChange - triggered when the user selects a new cell;
- Workbookopen - triggered when a workbook is opened.

The events raised by the software in response to actions performed by the user are monitored by the installed usage agent i.e. the characteristics of events are analysed by the usage agent to determine activities. Relevant triggered events are then exploited and analysed by the usage agent as they occur 402. Irrelevant events that do not contain information of relevance are ignored although the invention is also
adapted to watch for the disabling of certain options in spreadsheet applications by
the user which may be indicative of certain activities.

According to the event type, the usage agent can then either write information
contained in the event directly to the UDT file 403 or examine the spreadsheet to
determine if any change has been made to a cell or cells 404 before writing the
information to the UDT file. For example, Type 1 events such as a
SheetPivotTableUpdate event is written directly to the UDT file and includes:

- type of event;
- name of user;
- workbook name;
- worksheet name;
- time of event.

Type 2 events such as SheetSelectionChange event are examined to determine if
changes are made to cells. Accordingly, the type of event is identified according to
the change to the cell and the following information is also written to the UDT file:

- type of event;
- name of user;
- workbook name;
- worksheet name;
- time of event.
As previously described and as outlined further below, the events can then be collapsed as required into categories such as Review, Development and Data Entry.

The generated and consolidated UDT file 304 includes a row for each event captured by the usage agent which is then processed as outlined in Figure 5 to generate the Classified Usage Report Data (CURD) 307 table from the consolidated UDT file. As shown in Figure 5, the UDT file used to generate the CURD table consists of multiple rows corresponding with the events as outlined above. Each row comprises the following information:

- type of event;
- name of user;
- workbook name;
- worksheet name;
- time of event.

The information is parsed as follows to calculate usage times. The opening event is ignored and each subsequent event is then examined in turn based on the current and previous events and on the time, Tduration, between the previous event and the current event. Three time parameters are employed as previously described - TcriticalIdle, Tadd and TcriticalReview where Tadd <= TcriticalIdle. Tcurrent represents the time of the current event and Tprevious represents the time of the previous event so that

\[
T_{duration} = T_{current} - T_{previous}.
\]
As indicated above, the time data is manipulated and transformed into usage data as follows to create the CURD table 307:

Step 1:
- If \( T_{\text{duration}} \leq T_{\text{criticalIdle}} \) then the time period from the previous event to the current one is assigned the activity associated with the type of the current action.
- If \( T_{\text{duration}} > T_{\text{criticalIdle}} \) then the period from \( T_{\text{previous}} \) to \( T_{\text{current}} - T_{\text{add}} \) is assigned as inactivity and the period from \( T_{\text{current}} - T_{\text{add}} \) to \( T_{\text{current}} \) is assigned the activity associated with the type of the current event.

Step 2:
The periods of review activity are examined in sequence, with consecutive periods aggregated to produce a (reduced) set of periods of review activity. If the time for an aggregated period is less than \( T_{\text{criticalReview}} \) then the aggregate period may be reassigned. If the period immediately following the aggregated period does not exist or the period immediately following is categorised as inactivity then the aggregated period remains categorised as review activity. Otherwise the aggregated period is assigned the activity of the period immediately following the aggregated period as described in more detail in Figures 7 and 8 below.

The created CURD table 307, like the UDT file comprises rows of data consisting of the following information in each row:
- type of activity;
- name of user;
- workbook name associated with activity;
- worksheet activity associated with activity;
- duration of activity.

Figure 6 is a detailed flow diagram of the algorithm employed in the generation of the Usage Data Text files 304 described in Figures 3 and 4. As shown in Figure 6, upon running a spreadsheet application 600, the installed usage agent is woken or activated and the UDT file opened 601. The usage agent monitors and captures software events triggered by the user 602 on a continuous basis as described above. The usage agent constantly queries more events 603 and where no further events are detected the UDT file is closed 604.

However, where further events are detected, the usage agent queries whether the event is a relevant triggering event 605. Where the event is not a relevant triggering event, the process returns to query any further events 603. However, where the event is a triggering event, the event type is categorised according to whether the event is a Type 1 event 606 as described at 404 in Figure 4 or not. Where the event is a Type 1 event, the event information is written to a row in the UDT file 610 i.e.

- type of event;
- name of user;
- workbook name;
- worksheet name;
However, where the triggering event is not a Type 1 event (i.e. a Type 2 event), the cell contents are recorded 607 and the next event awaited and the cell contents compared 608 to deduce the activity based on a comparison of the cells.

If desired, the usage agent can determine the menu options available to the user during actions to assist in the analysis of the actions being performed by the user. Such a determination can be performed intermittently or at brief intervals.

The deduced activity and associated information is then written 609 to a row in the UDT file as described above.

Figure 7 is a detailed flow diagram of a first part of the usage classification algorithm 307 of Figure 3 employed to analyse the UDT files to create a Temporary Classified Data Report (TCDR) file before creation of the CURD table as outlined below in Figure 8. The creation of the TCDR file is included in the following description in the interests of clarity to clearly demonstrate the processes of the invention. However, as will be appreciated by those skilled in the art, the two-step algorithm as described in Figures 7 and 8 can also be performed in a single step thereby obviating the requirement for the creation of the TCDR file as outlined in Figures 3 and 4.

As shown in the drawing, upon activation of the usage classification algorithm 700, the system is instructed 701 by a user manager or external consultant on which UDT files require analysis. Each required UDT file is opened in turn 702 and the
information contained within each row for each event is input into an opened TCDR file 703 for output.

The algorithm queries more events 704 and where no events are detected the TCDR file is closed 705. However, where further events are detected, the relevant row of data is reviewed and the time of the event is stored as Tprevious 707 and subsequent events checked 704. Where no additional rows of information are present Tduration is calculated 708 as

$$T_{duration} = T_{current} - T_{previous}$$

i.e. event data is converted to activity data based on the categories of activity previously defined.

$T_{duration}$ is then compared with $T_{critical}$ 709 and where $T_{duration} < T_{criticalIdle}$ the activity data row is written to the TCDR 710 file including the following data:

- type of activity;
- name of user;
- workbook name;
- worksheet name;
- time of activity;
- duration of activity

and the process repeated as required.
Where $T_{duration}$ is not $< T_{criticalIdle}$, two rows are written to the TCDR file as follows:

Row 1: All data from event data (type of event/name of user/workbook name/worksheet name/time of event). $Duration = T_{add}$.

Row 2: Data from event data except time of event (type of event/name of user/workbook name/worksheet name). $Time = T_{current} + T_{add}$. $Duration = T_{add}$.

The process is also repeated as required.

Figure 8 is a detailed flow diagram of a second part of the usage classification algorithm in which data in the TCDR file created in accordance with Figure 7 is manipulated to generate the CURD table 307. As shown in the drawing, the TCDR file created as outlined in Figure 7 is first opened 800 while a CURD table 801 is also opened for later output. The activity rows within the TCDR file are checked 802 and, where no rows are present, the CURD table is closed and the algorithm ended.

Where additional activity rows are present in the TCDR file, the row activity type is examined 803 and where the activity is a non-review type activity, the activity data row is written to the CURD table 806 to include type of activity/name of user/workbook name/worksheet name/time of activity/duration of activity.
Where the activity type is identified as a review type activity, the algorithm continues to examine subsequent activity rows until a non-review activity is identified. The cumulative review times, TR, are calculated based on individual durations.

If \( TR > T_{\text{criticalReview}} \), the activity data to include type of activity/name of user/workbook name/worksheet name/time of activity/duration of activity is written to the CURD table based on the characteristics of the first review row with Activity Type defined as Review and Duration = TR.

If \( TR \) is not > \( T_{\text{criticalReview}} \), it is determined whether type of first non-review activity is idle.

Where the non-review activity is identified as lying idle, the activity data to include type of activity/name of user/workbook name/worksheet name/time of activity/duration of activity is written to the CURD table based on the characteristics of the first review row with Activity Type defined as Review and Duration = TR. However, where the non-review activity is not lying idle, the activity data to include the activity data to include type of activity/name of user/workbook name/worksheet name/time of activity/duration of activity is written to the CURD table based on the characteristics of the first review row with Activity Type defined as the type of first non-review activity and Duration = TR.

Figures 9 to 11 show a second embodiment of the system and method of the invention in which spreadsheet usage is categorised and measured by comparing two versions of the spreadsheet file to generate the CURD table.
More particularly, Figure 9 is a topological diagram of the second embodiment of the method and system of the invention 900 and describes a process broadly similar to that described in Figure 3. As shown in the drawing, a Comparison Usage Agent (CUA) is first installed 901 on each user PC 201 or laptop 202 as previously described. Alternatively, the CUA can be installed on a server on which spreadsheets are saved.

The CUA is then configured 902 to select users and the folders and/or files to be monitored. A Spreadsheet Status Text (SST) file is then set up in an analogous manner to the UDT file of Figure 3.

During operation 903, the CUA is activated upon opening of an Excel (Trade Mark) spreadsheet file to be analysed during use. A connection to the SST file is opened and at regular intervals, the spreadsheet files are analysed and a summary representation in the form of an encoded string which stores information on the cells in the spreadsheet is created 904. As with the UDT file of the first embodiment, the CUA writes information to include user name/workbook name/time/encoded string summary on a row by row basis to the SST file on a continuous basis. When all spreadsheets are closed by the user, the connection to the SST file is also closed.

The SST files are then collected 905 either manually or automatically using a network script as before where the user PC’s or laptops etc. are connected via a LAN or via the internet.
Usage categorisation 906 is then performed on the collected SST files analogous to the categorisation performed on the UDT files. Firstly, the collected data is consolidated employing a comparison and categorisation application in which the collected SST files are appended to create a single text file for analysis by the comparison and categorisation application. The consolidated SST files are then subjected to an examination and manipulation on a row by row basis by a usage and comparison algorithm as outlined further below 907 to produce the CURD table containing the CURD data.

In the analysis process 908, the data contained within the CURD table is summarised and presented in textual, tabular or graphical form as required 909 and as illustrated in Figures 12 to 17 below while the outcome of the analysis is finally reported 910 to relevant personnel in the organisation so that remedial action can be taken where required to improve productivity.

Figure 10 is a detailed flow diagram of the algorithm employed to generate an encoded string of the spreadsheet files in the form of a Spreadsheet Status Text (SST) file as outlined in step 904 of Figure 9. The initial steps of the process are broadly similar to steps 600 to 604 employed to generate the UDT file of Figure 6.

As shown in Figure 10, upon running a spreadsheet application 1000, the installed CUA is woken or activated 1001. A CUA timer triggers a spreadsheet analysis at regular intervals 1002 and an SST file is opened. The CUA constantly queries whether the spreadsheet application is still in use by the user 1003 and where
spreadsheet usage has ceased, the SST table is closed 1004 and the CUA process is terminated.

Where the spreadsheet application continues in use, the CUA queries at regular intervals whether it is an appropriate time to perform an analysis 1005.

Where analysis is appropriate, an encoded string (or HashMap) for the spreadsheet is generated in the SST table to include the following:

- encoded string representing ordered data values in cells of each worksheet in workbook (DataString);
- encoded string representing ordered formula values in cells of each worksheet in workbook (Formula String);
- encoded string representing ordered text values in cells of each worksheet in workbook (Text String);
- encoded string representing ordered object values in cells of each worksheet in workbook (Object String);
- encoded string representing the 'focus' cell address of each worksheet in workbook (Focus String).

The encoded strings, namely DataString, Formula String, Text String, and Non-Cell Object String are then combined or concatenated into single Spreadsheet String 1007 and the Spreadsheet String is written to the SST table 1008 together with

- name of user
- worksheet name
- focus string
- time of event.

Figure 11 is a detailed flow diagram of the usage comparison and classification algorithm outlined in step 907 of Figure 9 employed to analyse the SST file to create the CURD file. The preliminary steps in the usage comparison and classification algorithm are broadly similar to the preliminary steps of the usage classification algorithm as outlined in Figure 7. As shown in the drawing, following activation of the usage comparison and classification algorithm 1100, the usage comparison and classification algorithm is instructed on which SST files to analyse 1101 and the CURD text files are opened in turn and each row of data inputted into the CURD text files 1102. The CURD data file is then opened 1103 for output and the algorithm queries further rows of data 1104. Where there are no further rows of data in the SST files, the CURD file is closed 1105 and the process is terminated 1106.

However, where there are additional rows of event data, each row is analysed in turn 1107 and the time of the event is stored as the previous event time and the spreadsheet string is stored as Previous_string 1108.

In the event that the result at step 1107 is in the negative,

\[ T_{\text{duration}} = T_{\text{current}} - T_{\text{previous}} \]

The current and previous strings are then compared 1110 and where the strings are identical the following data is written to the CURD file 1111:
activity type = Idle
- name of user
- workbook name
- start time
- Tduration

However, where differences are detected between the current and previous strings, the differences between the strings are classified in the present embodiment as follows:

- difference only in cell focus. Type = Review;
- difference in text string, data string, and possibly cell focus. Type = Data Entry;
- difference only in formula string and object string. Type = Development;
- Difference in more than one category. Duration equally split.

The time of the event is then stored as the previous event time and the spreadsheet string is stored as Previous_string 1108 and the process repeated until there are no further data rows identified at step 1104 and the CURD file 1105 is closed and the algorithm ends 1106.

Figures 12 to 17 show examples of user interface screenshots from a computer system in accordance with the invention for discovering and mapping spreadsheet usage in which the method and system of the invention as described above in
relation to Figures 1 to 11 has been employed to analyse spreadsheet usage in an organisation.

As will be appreciated by those skilled in the art, the method and system of the invention can be adapted as required to monitor various activity types. The spreadsheet usage for the organisation described in Figures 12 to 17 was classified in time blocks of usage at a detailed level including time spent editing cells, creating formulae, editing formulae, saving, printing, charting, etc. The particular spreadsheet being used as well as information about the spreadsheet was also recorded. The user ID of the person using the spreadsheet was also recorded. The detailed usage information was then abstracted into six groupings as follows:

- Advanced Spreadsheet Development (Such as Macros & Data Extracts);
- Standard Spreadsheet Development (Creating formulas and similar);
- Data Entry;
- Output (Analysis and Data Exploration);
- Formatting and Restructuring;
- Review (Reviewing the spreadsheet and trying to understand or remember what a spreadsheet is supposed to be doing including navigation through the spreadsheet).

This information was represented through a series of graphics with controls to explore the usage of spreadsheets (in hours) by spreadsheet, user and activity.
Figure 12 shows a screenshot of a typical user interface 1200 provided with a dialog box 1201, a shortcuts box 1202 with shortcuts to data compiled and displayed relating to Time Costly Spreadsheets, Excessive Spreadsheet Usage, Data Entry Alert and Most Shared Spreadsheets. A filters box 1203 so that spreadsheet data can be selected according to User, Spreadsheet and Activity is also provided on the interface 1200 while in the present interface spreadsheet data is displayed in the form of a bar chart 1204 with each selected spreadsheet ID displayed on an X-axis and time units 1206 displayed on a Y-axis so that the cumulative time expended on individual spreadsheets is displayed i.e. Time Costly Spreadsheets can be identified.

A facility is also provided, indicated by the reference numeral 1207, whereby data can be displayed in the form of stacked and percentage bar charts.

The screenshots in the following Figures are broadly similar to the screenshot of Figure 12 and like numerals indicate like parts.

Figure 13 is a screenshot of an interface displaying a bar chart 1204 in which the cumulative time expended by individual users on spreadsheet activity is displayed with the user ID 1300 being displayed on the X-axis of the bar chart to highlight Excessive Spreadsheet Usage by users.

Figure 14 is a screenshot of a stacked bar chart 1204 in which the bar chart is stacked according to activity. A list 1400 of each activity is listed on the interface while each spreadsheet is identified on the X-axis and time periods identified on the Y-axis as in Figure 12. The activities stacked in Figure 14 include Review,
Formatting, Development, Data-Entry, Output and Advanced Development with Data Entry being displayed.

Figure 15 is a screenshot of a Most Shared Spreadsheets bar chart in which the X-axis of the bar chart identifies spreadsheets and the bars for each spreadsheet are subdivided according to colour-coded user ID's 1300 displayed on the interface. The Y-axis indicates the level of sharing as a %.

Figure 16 is a screenshot of Time Costly Spreadsheets in which the bars associated with each spreadsheet are subdivided by colour according to Review, Formatting, Development, Data-Entry, Output and Advanced Development with Data Entry activities listed 1600 on the interface.

Finally, Figure 17 is a screenshot of Time Costly Spreadsheets in which the bars corresponding to individual spreadsheets are subdivided according to usage by each user ID.

The data available from the screenshots of the type displayed in Figures 12 to 17 could be applied in the following scenario to identify previously unrecognised and unquantified wastage of resource in an organisation and enable effective, significant and accurate cost savings. For example, if it assumed that a finance department in a mid-size company has 10 people in it ranging from a Financial Director on a significant six figure salary of €150K per annum, two accountants on a salary of €60K or€70K per annum and a number of lower level employees such as Accounts Payable clerks and credit controllers on salaries of €35 or€40K and it is further
assumed that the 10 people have an average salary of €50K and that the burdened cost of employing them is 100%, then the cost of the finance department is €50K \times 2 \times 10 \text{ people} \text{ i.e. } €1 \text{ million per annum. All employees are spreadsheet users. However, if, using the method and system of the invention, it is determined that the non-productive spreadsheet usage of the employees is 50\% of their total time, then the non-productive spreadsheet usage cost is €500K per annum.}

The method and system of the invention can be adapted to generate other graph types from which business process data can be extracted, including, \textit{inter alia}, time spent by users on spreadsheets by day over a defined period (e.g. weeks/months), total time spent by users for each hour over the course of a day using spreadsheets and when and by which user spreadsheet files were used by date, time and activity.

The invention provides a powerful tool which facilitates the analysis of heretofore invisible business processes through the identification of the nature and type of spreadsheet usage giving rise to the non-productive usage costs so that any necessary reforms can be implemented.

As illustrated above, the invention measures the time and associated costs, of spreadsheet usage within an organization by spreadsheet, by user and by activity to enable significant productivity gains by highlighting inefficiencies and rendering previously unidentified or invisible work processes visible.
The invention also provides an early warning system for unexpected spreadsheet use and quantifies and mitigates the risk of security, errors, fraud and data loss associated with spreadsheet use.

The invention is not limited to the embodiments herein described which may be varied in construction and detail without departing from the scope of the invention.
Claims

1. A method for automatically mapping a business process employing a computer application comprising:

   analysing usage of the computer application by a user through categorising the usage by the user according to activities performed by the user in the computer application, and quantifying the time spent by the user at each activity from the times recorded for each action performed by the user in the computer application.

2. A method as claimed in Claim 1 wherein the times recorded for each action performed by the user are extracted from the event recording model of the computer application.

3. A method as claimed in Claim 1 or Claim 2 wherein the business process is mapped by categorising and quantifying instantaneous actions of the user.

4. A method as claimed in Claim 3 wherein the usage of the computer application by a user is written to a usage data text file.

5. A method as claimed in Claim 4 wherein the usage data text file is collected manually for categorisation.

6. A method as claimed in Claim 4 wherein the usage data text file is collected automatically for categorisation.
7. A method as claimed in any of Claims 4 to 6 wherein separate usage data text files are consolidated into a consolidated usage data text file.

8. A method as claimed in Claim 7 wherein the consolidated usage data text file is manipulated to produce classified usage report data.

9. A method as claimed in Claim 1 or Claim 2 wherein the business process is mapped by categorising, quantifying and comparing historical actions of the user.

10. A method as claimed in Claim 9 wherein the step of comparing the historical actions of the user is performed on two versions of a computer application data file.

11. A method as claimed in Claim 10 wherein the step of comparing the historical actions of the user is performed on a summary version of the computer application data file.

12. A method as claimed in Claim 11 wherein the summary version of the computer application data file comprises an encoded string summary.

13. A method as claimed in Claim 12 wherein the encoded string summary is prepared at regular intervals.

14. A method as claimed in any of Claims 9 to 13 wherein the encoded string summary is written to a status text file.
15. A method as claimed in Claim 14 wherein the status text file is collected manually for categorisation.

16. A method as claimed in Claim 14 wherein the status text file is collected automatically for categorisation.

17. A method as claimed in any of Claims 14 to 16 wherein separate status text files are consolidated into a consolidated status text file.

18. A method as claimed in Claim 17 wherein the consolidated status text file is manipulated to produce classified usage report data.

19. A method as claimed in any of Claims 1 to 18 wherein the step of quantifying the time spent by the user at each activity comprises extracting the time of a first action from the event recording model of the computer application, extracting the time of a second action from the event recording model of the computer application, calculating the time duration between the first and second events and allocating the time duration to an activity.

20. A method as claimed in any of Claims 1 to 19 wherein the business process is mapped by a usage agent installed on a computer device to capture the activities performed by the user.

21. A method as claimed in Claim 20 wherein the usage agent is installed on all user computer devices in a business.
22. A method as claimed in Claim 20 or Claim 21 wherein the computer device comprises a personal computer, a laptop, a tablet or a smartphone.

23. A method as claimed in Claim 20 wherein the usage agent is installed on a server on which computer application data files are stored.

24. A method as claimed in any of Claims 20 to 23 wherein the usage agent is communicable with a management computer on which the captured user activities are analysed.

25. A method as claimed in Claim 24 wherein the usage agent is communicable with the management computer via a Local Area Network, a Wide Area Network or the internet.

26. A method as claimed in any of Claims 1 to 25 further comprising the step of displaying the categorised usage and quantified time in textual, tabular or graphical form.

27. A method as claimed in any of Claims 1 to 26 wherein the computer application comprises an end user computer application.

28. A method as claimed in Claim 27 wherein the end user computer application is selected from the group comprising spreadsheet applications, email applications, word processing applications and database client-interface applications.
29. A method as claimed in Claim 28 wherein the end user computer application comprises a spreadsheet application.

30. A method as claimed in any of Claims 1 to 29 wherein the computer application comprises a database server system.

31. A computing device comprising:
   a processor for controlling operation of the computer device;
   computer applications having corresponding event recording models, and
   a memory for storing computer executable instructions that cause the computing device to extract the times recorded for each action performed by the user from the event recording models of the computer applications.

32. A computing device as claimed in Claim 31 wherein the computer executable instructions that cause the computing device to extract the times recorded for each action performed by the user from the event recording models of the computer applications are contained within an addIn usage agent.

33. A computing device as claimed in Claim 32 wherein the usage agent captures and records instantaneous keystrokes of the user from the event recording model to determine each action performed by the user.

34. A computing device as claimed in Claim 33 wherein the usage agent generates an encoded string of the computer application data files.
35. A computing device as claimed in any of Claims 31 to 34 wherein the usage agent is communicable with a management computer.

36. A computing device as claimed in Claim 35 wherein the usage agent is communicable with the management computer via a Local Area Network, a Wide Area Network or the internet.

37. A computer program comprising computer program code means adapted to perform all the steps of any of claims 1 to 30.

38. A computer program as claimed in claim 37 embodied on a computer readable medium.
Fig. 2
START

Installation

Install Usage Agent on each Machine running Microsoft Excel or other Spreadsheet Application

Configuration

Configure setup to select particular users whose activity is to be collected

Configure setup to select folder and filename for text document used to store the usage data on the local machine (referred to as the Usage Data Text file)

Operation

Excel Application Opened by User

Connection to Usage Data Text file opened

Usage information collected, categorised and recorded by Usage Agent on a continuous basis

Usage Agent writes information to Usage Data Text file on a continuous basis

Excel Application Closed

Connection to Usage Data Text file closed

Collection

Usage Data text Files collected either manually or automatically through possibility a network script, if machines are connected through a local area network, or using web services (such as htps) if machines have access to an internet connection.

Usage Categorisation

Consolidation of Data: Through a Software Application (referred to as the Categorisation Application), separate Usage Data Text Files are appended to create a single text file for analysis

The Categorisation Application, through the Usage Classification Algorithm, then examines and manipulates the consolidated Usage Data Text file on a row by row basis. A data table, referred to as the Classified Usage Report Data, containing the classified usage is returned for reporting.

Analysis

Information in the Classified Usage Report Data is summarised and presented through textual, tabular and graphical methods. Examples of information returned include amount of time spent at each activity for each spreadsheet for each user. This analysis can be performed by one of the users or by an external consultant.

Reporting

The outcome of the analysis stage is reported to relevant personnel in an organisation. Using the information processes and systems within the organisation may be improved or replaced.

FINISH
User Performing Spreadsheet Operations

As User interacts with the Spreadsheet, the spreadsheet application generates (triggers) software events in the background in response to the actions of the user.

The events contain information including the type of event, the time of the event, the username for the current user and the workbook and worksheet that the user is currently working on.

Examples of types of events in Microsoft Excel (2007 and 2010) are:
1. SheetPivotTableUpdate, triggered when a pivot table in Excel is changed.
2. SheetSelectionChange, triggered when the user selects a new cell.
3. WorkbookOpen, triggered when a workbook is opened.

These system-generated events are monitored by the Usage Agent and the relevant ones analysed by the Usage Agent as they occur.
(Not that many events are ignored as they do not contain information of relevance)

For some events (for example SheetPivotTableUpdate) the Usage Agent writes information contained in the event directly to the Usage Data Text file.
The information written includes:
1. Type of the event,
2. Name of the user,
3. Worksheet name
4. Workbook name
5. Time of the event

For some events (eg the SheetSelectionChange) the event prompts the Usage Agent to examine the spreadsheet to determine if any change has been made to a cell or cells. This yields the type for the event.
Again, the following information, taken from the event, is written to the text file:
1. Type of the event,
2. Name of the user,
3. Workbook name
4. Worksheet name
5. Time of the event

Generates Usage Data Text File
(This file includes a row for each event.)

Fig. 4
Parse the Usage Data Text file
Consists of rows with event information with following information in each row:
1. Type of the event.
2. Name of the user.
3. Workbook name
4. Worksheet name
5. Time of the event

The algorithm ignores the opening event and then examines each event in turn based on the type of the current and previous events and on the time, Tduration, between the previous event and the current event. The algorithm also uses three time parameters: TcriticalIdle, Tadd & TcriticalReview. Note that Tadd<=TcriticalIdle.
The algorithm uses Tcurrent to represent the time of the current event and Tprevious to represent the time of the previous event, thus Tduration=Tcurrent – Tprevious.

There are two steps to the algorithm:

Step 1:
- If Tduration<=TcriticalIdle then the time period from the previous event to the current one is assigned the activity associated with the type of the current action.
- If Tduration>TcriticalIdle then the period from Tprevious to Tcurrent-Tadd is assigned as inactivity and the period from Tcurrent-Tadd to Tcurrent is assigned the activity associated with the type of the current event.

Step 2:
- Events of Review type are examined in sequence, with consecutive periods aggregated to produce a (reduced) set of periods of review activity. If the time for an aggregated period is less than TcriticalReview then the aggregate period may be reassigned. If the period immediately following the aggregated period does not exist or the period immediately following is categorised as inactivity then the aggregated period remains categorised as review activity. Otherwise the aggregated period is assigned the activity of the period immediately following the aggregated period.

Classified Usage Report Data
Consists of rows of usage activity information with following information in each row
1. Type of the activity.
2. Name of the user.
3. Workbook name associated with activity
4. Worksheet name associated with activity
5. Duration of activity
Start: User runs Spreadsheet Application

Wakes up Usage Agent and Opens Usage Data Text File

Usage Agent Monitors Individual Software events Triggered by User manipulating spreadsheet

More Events to Analyse

N

Event of relevance to Usage Analysis

Y

Event of Type 1

Write Event Details to Usage File. Details include:
1. Type of the event.
2. Name of the user.
3. Workbook name
4. Worksheet name
5. Time of the event

Record Current Cell Contents

Await next event and compare Cell contents. Derive Activity based on Comparison

N

Y

End: Usage Data Text File Closes

Fig. 6

600

601

602

603

604

605

606

607

608

610

609
START: USAGE CLASSIFICATION ALGORITHM

Application receives instructions on which Usage Data Text Files to analyse.

Open Usage Data Text files in turn and input each row of data (event information) one by one.
(1. Type of the event, 2. Name of the user, 3. Workbook name, 4. Worksheet name, 5. Time of the event)

Open Temporary Classified Data Report (TCDR) File for Output

More Events to Check

Store time of event as previous event time

Close TCDR File

Calculate Tduration = Tcurrent – Tprevious.
Convert Event to Activity based on Categorisation

Row 1

Tduration<=TcriticalIdle

Write Activity data row to TCDR file including
(1. Type of the Activity, 2. Name of the user, 3. Workbook name, 4. Worksheet name, 5. Time of the activity, 6. Duration of the Activity)

Write 2 rows to TCDR file
Row1: Data(1-5) from Event data. Duration = Tadd
Row2: Data(1-4) from Event data.
Time = Tcurrent + Tadd
Duration = Tadd
Open TCDR File with Activity details.

Open Classified Usage Report Data (CURD) File for

More Activity Rows from TCDR to Check

Y

Row Activity Type = Review

Examine activity rows until reach non-review activity. Calculate cumulative review times (TR) based on individual durations.

TR>TcriticalReview

Y

Type of First Non-Review Activity = Idle

Write Activity data row as above to CURD file based on characteristics of first review row with Type = Type of First Non-Review Activity and Duration = TR

N

N

Write Activity data row as above to CURD file including:
(1. Type of the Activity, 2. Name of the user, 3. Workbook name, 4. Worksheet name, 5. Time of the Activity, 6. Duration of the Activity)

Y

Write Activity data row as above to CURD file based on characteristics of first review row with Type = Review and Duration = TR

N

Close CURD file & END

800

802

804

806

808

809

801
START

Installation
-
Install Comparison Usage Agent (CUA) on each machine running Microsoft Excel or on the server where the spreadsheets are saved.

Configuration
-
Configure setup to select particular users and for folders and files that are to be monitored.

Configuration
-
Configure setup to select folder and filename for text document used to store the usage data on the local machine or on the server (referred to as the Spreadsheet Status Text file).

Operation
-
Excel File under Monitor Opens – CUA activates.
-
Connection to Spreadsheet Status Text file opened.
-
At regular intervals spreadsheet files are analysed and a summary representation in the form of an encoded string which stores information on the cells in the spreadsheet.
-
Usage Agent writes information on a row by row basis to the Spreadsheet Status Test file on a continuous basis. This data includes the User Name, Workbook Name, Time, Encoded String Summary.

Operation
-
All spreadsheets close.
-
Connection to Spreadsheet Status Test file closed.

Collection
-
Spreadsheet Status Text file text file. Text files collected either manually or automatically through possibly a network script, if machines are connected through a local area network, or using web services (such as https) if machines have access to an internet connection.

Collection
-
Consolidation of Data: Through a Software Application (referred to as the Comparison and Classification Application), separate Spreadsheet Status Test files are appended to create a single text file for analysis.

Usage Categorisation
-
The Comparison and Classification Application, through the Usage Comparison and Classification Algorithm, then examines and manipulates the consolidated Spreadsheet Status Text file on a row by row basis. A data table, referred to as the Classified Usage Report Data, containing the classified usage is returned for reporting.

Analysis
-
Information in the Classified Usage Report Data is summarised and presented through textual, tabular and graphical methods. Examples of information returned include amount of time spent at each activity for each spreadsheet for each user. This analysis can be performed by one of the users or by an external consultant.

Reporting
-
The outcome of the analysis stage is reported to relevant personnel in an organisation. Using the information processes and systems within the organisation may be improved or replaced.

FINISH
Start: User runs Spreadsheet Application

Wakes up Comparison Usage Agent (CUA)*

- Usage Agent Timer triggers a spreadsheet analysis (at regular intervals)
- Open spreadsheet status file

Spreadsheet application still running

END: Spreadsheet Status File Closes & CUA closes

Appropriate time to analyse spreadsheet

Generate encoded string (also known as HashMap)
Including:
- Encoded String representing ordered data values in cells of each worksheet in workbook (DataString).
- Encoded String representing ordered formula values in cells of each worksheet in workbook (Formula String).
- Encoded String representing ordered text values in cells of each worksheet in workbook (Text String).
- Encoded String representing ordered object values in cells of each worksheet in workbook (Object String).
- Encoded String representing the ‘focus’ cell address of each worksheet in workbook (Focus String).

Combine (Concatenate) DataString, Formula String, Text String, and Non-Cell Object String into single String (Spreadsheet String)

Write: Spreadsheet String, Name of User, Worksheet Name, Focus String, and Time of Event to Spreadsheet Status file.
START: Comparison Usage Classification Algorithm

Application: Spreadsheet focus Text Files

Open CURD text files in turn and input each row of data

Open classified Usage Report Data File (CURD) for output

More rows to check

N

Close CURD file

End

Y

Row 1

N

Calculate: Tduration = Tcurrent - Tprevious

Y

Previous String = Current String

N

Write to CURD file:
1. Type Activity = Idle
2. Name of user
3. Workbook name
4. Start time
5. Tduration

Note: 2, 3, 5, & 6 from input file

Classification of difference between previous spreadsheet string and current spreadsheet string to determine type.
- Difference only in Cell focus. Type = Review.
- Difference in Text String, Data String, and possibly Cell focus. Type = Data Entry
- Difference only in Formula string and Object string. Type = Development
- Difference in more than one category. Duration equally split.
INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06F11/34 G06Q10/00

ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F G06Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database consulted during the international search (name of database and, where practicable, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category*</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
</table>

Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:

A” document defining the general state of the art which is not considered to be of particular relevance

E” earlier application or patent but published on or after the international filing date

L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

O” document referred to in an oral disclosure, use, exhibition or other means

P” document published prior to the international filing date but later than the priority date claimed

*” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

*Y” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

*Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

&” document member of the same patent family

Date of the actual completion of the international search

27 November 2012

Date of mailing of the international search report

04/12/2012

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2

NL - 2280 HV Rijswijk

Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016

Authorized officer

Bozas Ioannis
<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>the whole document</td>
<td></td>
</tr>
<tr>
<td>Patent document cited in search report</td>
<td>Publication date</td>
<td>Patent family member(s)</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>US 2008133287 Al</td>
<td>05-06-2008</td>
<td>NONE</td>
</tr>
<tr>
<td>US 2005183143 Al</td>
<td>18-08-2005</td>
<td>NONE</td>
</tr>
<tr>
<td>us 2005235267 Al</td>
<td>20-10-2005</td>
<td>NONE</td>
</tr>
<tr>
<td>us 2008222286 Al</td>
<td>11-09-2008</td>
<td>NZ 553157 A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2008222286 Al</td>
</tr>
</tbody>
</table>