GTAG®

GLOBAL TECHNOLOGY AUDIT GUIDE

IPPF – Practice Guide

Auditing User-developed Applications

The Institute of
Internal Auditors

Global Technology Audit Guide (GTAG®) 14
Auditing User-developed Applications

Authors:

Christine A. Bellino

Douglas Ochab, CISA

Jeffery S. Rowland, CIA, CISA

June 2010

The information included in this document is general in nature and is not intended to address any particular individual, internal audit activity, or organization. The objective of this document is to share tools, resources, information, and/or other knowledge that is accurate, unbiased, and timely. However, based on the date of issuance and changing environments, no individual, internal audit activity, or organization should act on the information provided in this document without appropriate consultation or examination.

GTAG — Table of Contents

1. Executive Summary

User-developed applications (UDAs) typically consist of spreadsheets and databases created and used by end users to extract, sort, calculate, and compile organizational data to analyze trends, make business decisions, or summarize operational and financial data and reporting results. Almost every organization uses some form of UDAs because they can be more easily developed, are less costly to produce, and can typically be changed with relative ease versus programs and reports developed by IT personnel.

However, once end users are given freedom to extract, manipulate, summarize, and analyze their UDA data without assistance from IT personnel, end users inherit risks once controlled by IT. These risks include data integrity, availability, and confidentiality.

Data integrity risks exist because UDAs are neither subjected to structured manual balancing controls to validate the output nor to stringent application development and change management controls. Availability risks exist because UDAs can be stored on media (e.g., end users' computers or USB flash drives) that is easily lost or destroyed and may not be part of the IT department's automated periodic backup process. Confidentiality risks exist because a UDA and its data can easily be transmitted outside the company via e-mail. Data also can be stored without appropriate access controls.

Regulatory factors also must be considered as UDAs can have a significant impact on an organization's ability to comply with global regulations. Failure to properly control system development, changes, and logical access to critical UDAs can compromise an organization's compliance activities related to regulations such as the U.S. Sarbanes-Oxley Act of 2002, Japan's Financial Instruments and Exchange Law, the Gramm-Leach-Bliley Act, Payment Card Industry Data Security Standards (PCI-DSS), the Basel Committee on Banking Supervision's Basel II Framework, and other global financial and privacy regulations or standards. Considering the regulatory factors and the compliance requirements, internal auditors can play a strategic role as consultants to management on how best to develop, deploy, and maintain an effective UDA control framework. This role is in addition to the internal auditors' traditional assurance role in determining whether the internal controls over UDAs are properly designed and operating effectively. (*Note:* It is imperative for all parties to understand that development of the UDA control framework is management's responsibility, and internal auditors should not take ownership of the framework.)

Because management relies on UDAs, which can be a significant part of financial reporting and operational processes, as well as related decision making, the internal auditor should determine and review UDA risks and build an audit of UDAs into the annual internal audit plan as appropriate. The audit process includes a series of steps including identifying critical UDAs, evaluating the level of risk associated with each UDA, and testing the controls to determine whether they are sufficient to reduce associated risks to an acceptable level based on the organization's risk appetite and tolerance. Internal auditors should give special attention to the review of manual journal entries typically supported by UDAs as a source of potentially material spreadsheets. If the internal auditors do not have access to a management-generated inventory and risk ranking for UDAs, they would do well to look first at the UDAs that support the financial close and reporting processes as a basis for the audit's scope. In this situation, an internal auditor would likely immediately identify a control weakness due to the lack of a sufficient, management-driven, UDA control framework. Nonetheless, an audit could be performed based on the limited scope.

GTAG-14 *Auditing User-developed Applications* provides direction on how to scope an internal audit of UDAs. More specifically, it focuses the auditor on:

- Identifying availability of an existing UDA control framework that includes policies, procedures, UDA inventories, and a risk-ranking methodology that can be relied on for scoping purposes.
- Using the existing UDA control framework components to scope the UDA population to be included in the audit.

GTAG-14 also provides guidance for how the internal auditor's role as a consultant can be leveraged to assist management with developing an effective UDA control framework, including:

- Identifying the UDA population by using different discovery techniques.
- Assessing and ranking the risks associated with each UDA based on the potential impact and likelihood of risk occurrence.

Next, this GTAG outlines other considerations that internal auditors should address when performing UDA audits. These considerations can include management's concerns, the results of prior audits, tests of IT general controls, and consideration of best practices.

Finally, GTAG-14 provides a sample UDA process flow as well as a UDA internal audit program and supporting worksheets to help internal auditors organize and execute an audit. However, this document is not intended to provide extensive UDA internal audit tests and techniques; rather, it provides key considerations for auditing UDAs.

2. Introduction

In today's global economy, an organization's livelihood is impacted by how well the IT activity manages the availability, integrity, and confidentiality of the information and IT systems used to operate core business procedures.[1] However, UDAs can undermine the IT activity's ability to carry out its increasingly important responsibilities because they introduce unique challenges that traditionally are not addressed by standard IT processes. In addition, UDAs also present compliance challenges ranging from Sarbanes-Oxley to European laws. If not properly managed, UDAs have the potential to undermine controls even in the best managed and controlled organizations.

This GTAG will address:

- Defining UDAs.
- Benefits of UDAs.
- Risks associated with UDAs.
- Differences between UDAs and IT-developed and supported applications.
- Compliance challenges.
- The internal auditor's role in assisting organizations in managing and mitigating risks associated with UDAs.
- Scoping and considerations for an internal audit of UDAs.
- Development of a UDA internal audit program.

2.1. *Defining User-developed Applications*

UDAs are applications that are developed by end users, usually in a noncontrolled IT environment. Similar to traditional IT applications, UDAs automate and facilitate business processes. Although the most pervasive UDAs are spreadsheets, UDAs also can include user databases, queries, scripts, or output from various reporting tools. In general, a UDA is any application that is not managed and developed in an environment that employs robust IT general controls.

Even organizations with mature IT environments are highly likely to use and rely on UDAs in their day-to-day management activities. These UDAs range from simple calculations and information tracking to the use of complex macros that compile financial statements. Studies[2] have revealed that even large companies with mature IT environments use hundreds — sometimes thousands — of spreadsheets in the course of their daily business activities.

2.2. *Benefits of User-developed Applications*

Almost every organization uses some form of UDAs because they are:

- *Quicker to develop and use*. It may take several weeks and likely be expensive for IT personnel, who are following a rigorous system development and change management life cycle process, to create or modify a report that extracts information from a system in the format that a manager needs. That same manager often can extract and format the information within hours by using tools and utilities available to end users.
- *Readily available tools at a lower cost*. Commonly available tools, such as spreadsheets, offer users a way to automate business logic without going through a lengthy and costly software selection and/or system development and implementation process.
- *Configurable and flexible*. Compared to traditionally-managed IT applications, users have much greater flexibility to configure UDAs to fulfill business needs. For example, information in spreadsheets can easily be sorted and reformatted to allow additional analysis by users unfamiliar with structured programming languages and application development methodologies.

2.3. *Risks Associated With User-developed Applications*

Even though UDAs provide several benefits, they also pose risks to organizations, some of which could be significant. If the risks associated with UDAs are not properly managed and controlled, the integrity, availability, and confidentiality of UDAs can be compromised.

The most significant risk is the integrity of the data and information managed and reported. Management may assume that data contained in a report generated from a UDA is as reliable as information generated from an IT-developed and supported application. However, the nature of how UDAs are developed means this assumption may not be correct because UDAs typically do not follow a structured and controlled application development/change management life cycle.

Control breakdowns within UDAs may be traced to:

- *Lack of structured development processes and change management controls*. Lack of structure and controls around the development of and/or change to UDAs can lead to inaccurate calculations and data output. In all likelihood, the main factor of inaccurate data or reporting can be traced back to the lack of formal development processes and application change management controls.
- *Data download issues*. Lack of controls around the downloading of data from IT-developed or supported applications into the UDA can lead to use of

[1] IT Policy Compliance Group's 2008 *Annual Report: IT Governance, Risk and Compliance – Improving Business Results and Mitigating Financial Risk*

[2] Gartner's *Spreadsheet Controls Need a Boost*

inaccurate information. Similar issues also may occur for applications that rely on UDA output.

- *Increasing complexity*. The risk of UDAs becoming more complex over time than originally intended is often commonplace. Without adequate design or architecture, errors can occur in data manipulation and/or the resulting output.
- *Lack of developer experience*. UDA development by individuals who are unfamiliar with a particular application's functionality may cause them to use inefficient or ineffective development practices. For example, in designing a formula in a spreadsheet application, the developer of the UDA may "hard code" a particular number in a calculation rather than referencing the number from a field in the spreadsheet or using built-in application functionality.
- *Lack of version controls*. UDAs may be updated by many individuals, leading to various errors resulting from changes or corrections being deleted when older files overwrite a newer version.
- *Lack of documentation*. Lack of formal documentation of UDA design and functionality creates an environment that can lead to inaccurate information being input, processed, and eventually reported or used elsewhere. In addition, the lack of documentation makes it difficult to support and/or transition the use of the UDA to another employee or department.
- *Lack of support*. UDAs may be developed by an employee using a technology unfamiliar to others in the organization, which can create future support issues.
- *Limited input and output controls*. Lack of appropriate input and output controls, such as completeness checks, validity edits, and balancing routines, may result in data errors.
- *Lack of formal testing*. Failure to properly test a UDA's completeness and accuracy can lead to undetected errors.
- *Hidden data columns or worksheets*. UDAs may contain hidden data columns and worksheets that go undetected and untested.

In addition to data integrity, confidentiality also can be compromised by not taking advantage of security and access control mechanisms available within the UDA platform itself. UDAs typically are stored on less secure PCs, which can further increase the likelihood of a confidentiality breach.

Furthermore, UDAs maintained on a PC may not be backed up or backup media may be kept in the same location where the original copy is stored, risking a loss of data if the PC is destroyed or becomes inaccessible. Software licensing violations may occur if the software used to create the UDA is not properly licensed to the organization. Also, duplication of efforts can occur as users develop UDAs with functionality similar to other UDAs or applications used within the organization. Finally, and in many cases, duties are not properly segregated between the person(s) who designed, developed, and tested the UDA. More often than not, the end user(s) who created the UDA is the same person using it. This lack of segregation can allow design, programming, and/or logic errors to exist without detection.

2.4. Differences Between User-developed Applications and IT-developed and Supported Applications

2.4.1. Development

UDA development follows a significantly different process than the life cycle of an IT-developed and supported application. Generally, for an IT-developed and supported application, there is a standard life cycle that encompasses a feasibility analysis that includes a risk assessment; requirements definition; a design phase, construction, and testing phase; and a post-implementation review to ensure that the final product meets the users' needs and is operating as designed. Throughout these stages, a representative from risk management, internal auditing, and/or information security (IS) should be part of the project management process to help ensure that the application is implemented with proper controls.

By contrast, UDAs often are developed on an ad hoc basis by individuals outside the formal IT roles and responsibilities, within a short period of time and often without the benefit of the internal controls provided by a structured application development and change management life cycle. A UDA typically is built with scant consideration to design and no appropriate approvals. Application controls are usually an afterthought, if considered at all. Frequently, there is little testing beyond a cursory review to make sure the information looks correct. If testing is performed, it often is done by the person who designed and is using the application. Finally, while system documentation is present with most IT-developed applications, UDAs are usually developed without any documentation explaining what the UDA does or how it works.

2.4.2. Deployment

When an IT-developed application is placed into operation, the programming code typically is stored in a source code management application to prevent the application from being accidentally or intentionally changed. If the code is changed, audit trails normally exist that can detail the changes. On the contrary, UDAs are exposed to the potential for data corruption as a result of the lack of security and versioning controls. UDAs typically are located in publicly accessible network folders that lack adequate logical security to protect against unauthorized changes. In addition, there may be few audit trails that record or track changes.

2.4.3. Operations

UDAs are not considered in typical IT governance processes since they are known to only the primary users and are not part of typical IT risk assessments or data classification schemes. Access controls, even when present, are usually weak and unless they are stored on a shared drive, UDAs may not be backed up.

2.5. Compliance Challenges

Given the frequent lack of structured controls over UDAs, their use can present organizations with several compliance challenges related to country- and industry-specific rules and regulations, such as the U.S.'s Sarbanes-Oxley Act, the Gramm-Leach-Bliley Act, the Health Insurance Portability and Accountability Act of 1996, Federal Financial Institution Examination Council guidance, the United Kingdom's Data Protection Act of 1998, Japan's Financial Instruments and Exchange Law, Germany's law on employee confidentiality, European privacy regulations, Basel II, and the PCI-DSS, to name a few.

Under Sarbanes-Oxley and Japan's Financial Instruments and Exchange Law, public companies must show that controls over financial reporting are designed and operating effectively. However, the lack of controls over the development and use of UDAs may make this difficult depending on their use with respect to external financial reporting. Internal controls should be designed and implemented to limit the risk associated with development and use of UDAs.

The Public Company Accounting Oversight Board (PCAOB) — which is responsible for monitoring external auditing firms' activities according to Sarbanes-Oxley and compliance with Auditing Standard No. 5 (AS5): An Audit of Internal Control Over Financial Reporting that is Integrated with an Audit of Financial Statements — issued a Report on the First-year Implementation of Auditing Standard No. 5. The report stated that: "Auditors tested certain controls without testing the system-generated data on which the tested controls depended; the auditors did not test controls over applications that processed financially significant transactions, including important manual spreadsheets."[3] This is a sign of future increased scrutiny over high-risk, financial-related spreadsheets.

Weaknesses in confidentiality controls over UDAs also can lead to compliance issues. To comply with the previously noted regulations in addition to privacy laws and standards, it is necessary for internal auditors to identify customer or patient information, its storage location, its use in UDAs, and the related computer systems storing the information.

The use of UDAs can make it difficult to identify, inventory, and control personally identifiable information (PII) such as customer credit and patient information because UDAs typically are not part of organizations' data classification schemes. Once internal auditors identify customer or patient information, the access controls over the UDA need to be sufficient to not only protect it against unauthorized updates but also view access. Because UDAs typically reside on a user's PC or removable media, it becomes increasingly more difficult to control access and adequately protect the information.

For U.S. financial institutions, the U.S. Federal Financial Institutions Examination Council (FFIEC) guidance requires standards to be in place for UDAs.[4] As outlined in FFIEC's *Information Technology Examination Handbook*, financial institutions should include procedures for managing internally developed UDAs in their application development standards. The FFIEC recognizes that formal controls and application change management procedures around the development of UDAs frequently do not exist. As a result, the financial institution needs to determine its level of reliance on the UDA in making business decisions, which will determine the extent to which formal development procedures, application change controls, and backup procedures are necessary.

Financial institutions also should consider the Basel II standards related to operational risks, which are defined as "the risk of loss resulting from inadequate or failed internal processes, people, and systems or from external events."[5] The standard outlines criteria to be considered in measuring operational risks, such as the theft of information. The lack of controls over UDAs can contribute to operational risks and the loss of "proprietary and confidential" information.[6] Therefore, identification of controls over the development and use of UDAs is necessary to ensure compliance with the standards.

2.6. Internal Auditing's Role

The internal audit activity is in a unique position to understand management's reliance on UDAs, review the organization's use of UDAs, and evaluate the risk presented

[3] PCAOB Release No. 2009-006, Page 8

[4] FFIEC's *Information Technology Examination Handbook*, Development and Acquisition Booklet

[5] Basel II, Paragraph 644

[6] Basel II, Paragraph 819

to the organization. For high-risk UDAs, the internal audit activity is also in a position to recommend whether development, change management processes, and controls are required and how to implement them in the form of a UDA control framework.

Internal auditors can recommend and assist the organization in developing risk-based standards that can be used to trigger a more rigorous development process for UDAs. These standards also can be used to periodically evaluate existing UDAs because the design typically changes over time and may increase in complexity. However, internal auditors should remember to remain independent when assisting the organization in developing standards, procedures, or controls so as to not impair their objectivity. Refer to The IIA's International Professional Practices Framework's (IPPF's) Attribute Standard 1130: Impairment to Independence or Objectivity for more information.

The internal audit activity's involvement during the development of UDAs helps reduce the risk that problems and security and control weaknesses will not be identified until later in the UDAs' development life cycle. Changes made later in a UDA life cycle can be more costly than if they had been considered earlier in the development process, such as in the requirements phase.

Unfortunately, development of a UDA typically does not occur as a part of a formal project; therefore, internal auditors should help management raise awareness within the organization regarding the need for standards around the development of UDAs and the application of best practices. Internal auditors could assist management with developing an effective UDA program and help promote it throughout the user community. One of the primary challenges is creating a definition of when, for example, a spreadsheet becomes key to financial and operational reporting.

Internal auditors should review the organization's policies and procedures to determine whether they adequately address the development and protection of UDAs. Internal auditors also should test compliance with those policies and procedures as part of an internal audit. If UDA policies, procedures, inventories, and risk assessment procedures are deficient or nonexistent, those control weaknesses should be documented and reported to management as part of the audit.

During the course of conducting a review, internal auditors should provide management with an independent assessment of whether UDAs relied on for critical business decisions are adequately controlled and whether they provide complete and accurate information. Providing such an assessment would require an internal auditor to:

- Determine whether management has identified critical UDAs.
- Review management's risk assessment.
- Select the UDAs with the highest significance of risk.
- Evaluate the level of mitigating controls.
- Review documentation.
- Assess controls, where necessary, over:
    - Changes and modifications.
    - Backup and recovery.
    - Security.
    - Data integrity.

Details related to the above steps are provided in the following sections. A control weakness would exist if management has not completed an inventory and risk assessment related to UDAs.

3. Scoping a User-developed Application Audit

As emphasized in GTAG-11: *Developing the IT Audit Plan*, it is important to start with the right perspective when developing the internal audit plan: "An appropriate perspective to keep in mind is that technology only exists to support and further the organization's objectives and is a risk to the organization if its failure results in the inability to achieve a business objective." Throughout this GTAG, the term *technology* is simply replaced by the term *user-developed applications*. It is important to understand the risks presented by the use of UDAs to the organization's ability to achieve its operational, financial, and compliance objectives either as part of a focused analysis or as part of an integrated audit where the UDA component is considered within the business context of the review. Maintaining a clear focus on the business objectives and risks will assist the internal auditor in developing an audit plan that focuses audit resources on the risks and controls most significant to the organization. The internal audit activity can use the steps presented in the remainder of section 3 to either:

- Review the adequacy of an existing UDA control framework; or
- Assist management in developing or augmenting an effective UDA control framework.

Section 3.1 outlines the key elements required from an effective UDA control framework to allow the auditor to properly scope the review.

3.1. Defining What Constitutes a Key User-developed Application

Defining what constitutes a key UDA is critical to developing the internal audit scope. Because every newly created spreadsheet or database does not constitute a UDA, management must determine and define what constitutes a key UDA. As a reminder, for purposes of this GTAG, UDAs are any application that are not managed and developed in a traditional IT environment and under a formal development process. Spreadsheets used on an ad hoc basis — to provide lists of information or to quantitatively illustrate data available elsewhere — usually are not considered UDAs. A UDA is key if at least one of the following criteria are met:

- The UDA is used to initiate, accumulate, record, report, or monitor material financial reporting-related transactions and key operational management reports and/or meet regulatory compliance requirements.
- The UDA's use is inherent in performing key financial and/or operational control processes (e.g., account reconciliations and key performance indicator reports) so that if the spreadsheet or data was lost or corrupted, the loss would impact the control's effectiveness.

3.2. Determining and Defining the User-developed Application Population

Management may call for a review of specific, known UDAs (e.g., those that support journal entries) or it may require the identification of all steps and tools used to support business processes. In either case, if management does not maintain a consolidated list of UDA applications, the auditor may, in the role of consultant, guide management on how to identify and inventory UDAs by evaluating business process documentation such as business process flows and procedural narratives. Other techniques that management may consider for identifying the UDA population include:

- The use of a search capability to identify spreadsheet and database file tags within all or specific file directories related to a business process.
- Use of purchased software tools to detect UDA populations. (See section 4.1 for UDA discovery tool attributes and capabilities.)
- Review of reports identifying manual journal entries, which likely are supported by a UDA.

3.3. Defining Risk Factors

When developing a UDA control framework, the process typically begins by interviewing key management and staff members. This is required to gain a complete understanding of who uses UDAs and how they are used as a part of business processes, reporting functions, compliance programs, or control structure. Establishing materiality guidelines will be critical during the risk assessment phase described later in this section.

Assessing the UDA's risk that is relevant to the organization's overall operational, financial, and compliance objectives presents the internal auditor with a considerable challenge. Using spreadsheets or other UDAs for accumulating and calculating critical operational and material financial information can present significant risk to the organization, including:

- Data integrity issues.
- Errors made during input, processing, and output, including interfaces and reports.
- Errors or intentional manipulation due to unsecured files or unmanaged change.[7]

The internal auditor can guide management on using risk assessment techniques to identify critical vulnerabilities pertaining to the organization's operational, financial

[7] ACL Services "Spreadsheets: A High-Risk Tool for Data Analysis". White Paper, Page 1.

reporting, and compliance requirements as required by the IPPF's Performance Standard 2120: Risk Management. Two factors should be considered in the evaluation: the potential impact of a failure and the likelihood of a failure.

At a minimum, the risk factors for identifying the impact of a failure in a UDA should include:

- *Financial, operational, and regulatory compliance materiality of the UDA*. The risk assessment process starts with review of the UDA inventory and the determination of whether a failure in the UDA's integrity represents a likely threat to the reliability of the financial statements, key operational management reports, and/or regulatory compliance requirements.
- *Expected life and frequency of use of the application*. If a spreadsheet or database is developed for repetitive or ongoing use on a regular basis, it may be a high-impacting UDA.
- *Number of users of both the application and the results*. If spreadsheets or databases are accessible to more than one user and are used to provide data to multiple recipients, they are more likely to be a high-impacting UDA.

At a minimum, the risk factors for identifying the likelihood of a failure in a UDA should include:

- *Complexity of obtaining inputs and generating desired outputs*. Spreadsheets with macros or links to databases or other spreadsheets, large amounts of data, use of data extracts, or complex calculations are more likely to be key UDAs.
- *Frequency of modification to the UDA*. As expected, the more change that occurs to the spreadsheet or database, the higher the risk that an uncontrolled change may occur, resulting in an error to the financial statements, management reports, or regulatory compliance reporting.

While the impact and likelihood risk criteria may be appropriate for some organizations, others may use UDAs so extensively that other relevant risk criteria may be needed to ensure the appropriate level of resources are expended to mitigate the risks associated with the use of UDAs. Additional risk criteria for determining impact may include:

- The number of business processes reliant on the UDA.
- The number of controls supported by the UDA.
- Alternative or independent sources of data and/or controls in place that would detect a UDA control failure.
- Alternative controls or data sources that would detect a UDA error or integrity issue.
- Sensitive information, such as PII, contained in the UDA.

Additional risk criteria for determining the likelihood may include:

- Relationship to other systems and their outputs. Spreadsheets that produce outputs that are easily verified to other reliable data sources are less likely to be considered high-risk UDAs. However, spreadsheets that depend on links to other data sources where outputs are not easily confirmed likely qualify as higher-risk UDAs. Consider the following IIA guidance that can be adapted for the use in determining in-scope UDAs:

  "If the normal operation of the manual portion of the control is sufficient to detect an error in the automated portion (e.g., the computer report), then the control can be considered entirely manual since no reliance is being placed on the computer application. For example, a bank reconciliation might use a report from the general ledger system of cash transactions; if the report was incorrect or incomplete, it would be detected by the bank reconciliation process." [8]
- Guidelines established and used during the design of input and output controls (e.g., data input area does not contain formulas or input data is in the same order as the source data).
- Logic guidelines established and followed during development of the UDA and when changes are made to existing UDAs (e.g., use of formulas that foot and cross-foot data, locking and protecting cells, placement of critical values in separate cells, etc.).
- Guidelines established and followed for testing and approvals of newly developed UDAs and modifications to existing UDAs.
- Prior control failures associated with the reliance on UDAs.
- Access guidelines established and followed that control access to UDAs (e.g., storage and limited to appropriate users).
- Knowledge of the staff responsible for creating and maintaining the UDA.
- Use of version control.
- Use of monitoring controls.

The result of establishing and assessing key risk factors associated with the UDA inventory will determine the criteria by which the risk assessment is performed.

[8] *Sarbanes-Oxley Section 404: A Guide for Management by Internal Control Practitioners*, Page 34

can enhance the organization's ability to create the inventory in a sustainable manner that can easily be repeated as new UDAs are developed. In a more complex business environment, there can be hundreds — if not thousands — of UDAs that change on a daily basis. Manual inventory processes can be extremely time intensive, taking up to two or three months to complete in some cases. During that time, additional UDAs likely will have been added and deleted, making the effort even more challenging.

The use of technology will assist efforts to obtain an inventory that is complete, accurate, and relevant to the intended user. Because there typically is not time to expend the effort to conduct a manual discovery process, automated tools not only have the ability to assist organizations in the discovery of UDAs, but also can perform a risk ranking based on predefined materiality and complexity. Also, automation allows for the implementation of a continuous monitoring program in support of the organization's UDA control framework. An additional benefit to this approach is that the internal auditor can then focus more effectively on the true risks presented through the organization's use of UDAs.

Best-of-breed tools not only assist in the discovery, inventory, and risk ranking of UDAs but also provide the following capabilities:

- Perform diagnostics to identify UDA mechanical or logical errors, including errors of omission, and report on those errors for remediation purposes.
- Provide ongoing UDA management capability to ensure integrity from creation to storage and destruction.
- Provide a workflow with electronic signature capability.
- Enhance ability to manage linked spreadsheets.
- Provide documentation management capability for UDA key inputs, calculations, and outputs.
- Provide a structured process for UDA change management, data integrity, and version and access controls.
- Provide visibility for continuous monitoring to support maintenance of the control environment.

The reference to technology within this section can be as simple as running a script to capture all files located on the network that contain, for example, .xls or.mdb file extensions. On the other end of the spectrum, there are off-the-shelf software packages that can perform all of the capabilities discussed within this section.

4.2. Best Practices for Controls Over User-developed Applications

These guidelines illustrate best practices for the development, maintenance, and use of UDAs. As the complexity and criticality of any given UDA increases, there becomes a greater need to formalize how changes are applied. While not all of these guidelines are required, adoption and implementation of these principles facilitates control implementation and leads to improved quality and consistency of UDAs.

Access Guidelines

- Limit access to spreadsheets and other end user systems stored on a network server on a need-to-know basis according to job responsibilities.

Source Data Guidelines

- The data input area generally should not contain formulas. "When each cell contains both key data and the complicated assumption-laden algorithms to be applied, confirming the results are appropriate or reasonable may be virtually impossible — even if calculated correctly. It is a better practice to separate the data from the algorithms and assumptions being applied to the data."[9]
- When possible, data input — manual or interfaced — should be in the same order as the source data to facilitate review and minimize input errors.
- Lock formulas.

Source Output Guidelines

- Do not use the same worksheet and only change the assumptions and variables while leaving no baseline or trail of what has been changed during the "what if" analysis. "The best way to compare and review results from different combinations of variables are (a) to copy the original data sets and calculations into a separate spreadsheet tab, and (b) to build a comparison spreadsheet tab, which presents and contrasts the original."[10]
- Consider what the final presentation format needs to look like. Avoid the need to manually retype the output into other formats and tools, causing errors.[11]
- Identify authorized users for each report that is output as well as data storage and retention guidelines.

Testing Guidelines

- Make sure that changes to highly complex or critical UDAs are formally requested, documented, and tested.
- Task someone other than the spreadsheet's user or developer with testing complex or critical calculations and logic.
- Use analysis and reasonableness reviews to detect errors in calculations and logic.

[9, 10, 11] "Spreadsheet 'Worst Practices,'" CFO.com

Logic Guidelines

- Place critical values in a separate cell and refer to this cell in the formula rather than incorporating the number in a formula in one or more cells.
- Incorporate batch totals and control totals.
- Use formulas that foot and cross-foot data.
- Ensure data integrity by locking or protecting cells to prevent inadvertent or intentional changes to static data or formulas.
- Include expected results where possible to compare and monitor the reasonableness of UDA output.

Version, Backup, and Archiving Guidelines

- Use unique folder and file naming conventions that include the month, quarter, and year to help ensure that only current and approved versions of UDAs are used. Consider using check-in and check-out software to manage version control.
- Ensure data backup by storing spreadsheets and other UDAs on a network server that is backed up daily.
- Store historical files and databases not in use in a segregated, read-only folder to avoid mistakenly using them.

Documentation Guidelines

- Document the purpose and use of each critical UDA and update accordingly. The documentation should include the business objective, inputs, outputs, and sequence of execution for multistep processes.
- Create a consistent layout for spreadsheets and other UDAs to simplify use and testing. The areas for data input, calculations, and output should be distinct and separate.
- Label files, data sets, worksheets, key fields, rows, columns, and data for easy identification.
- Inventory all key spreadsheets and other UDAs impacting financial statement preparation.
- Clearly document assumptions applied and leveraged to generate data or perform calculations.

Microsoft's white paper on spreadsheet compliance emphasizes the importance of developing a long-term spreadsheet development and maintenance methodology as well as how Microsoft Office 2007 can help address compliance challenges.[12]

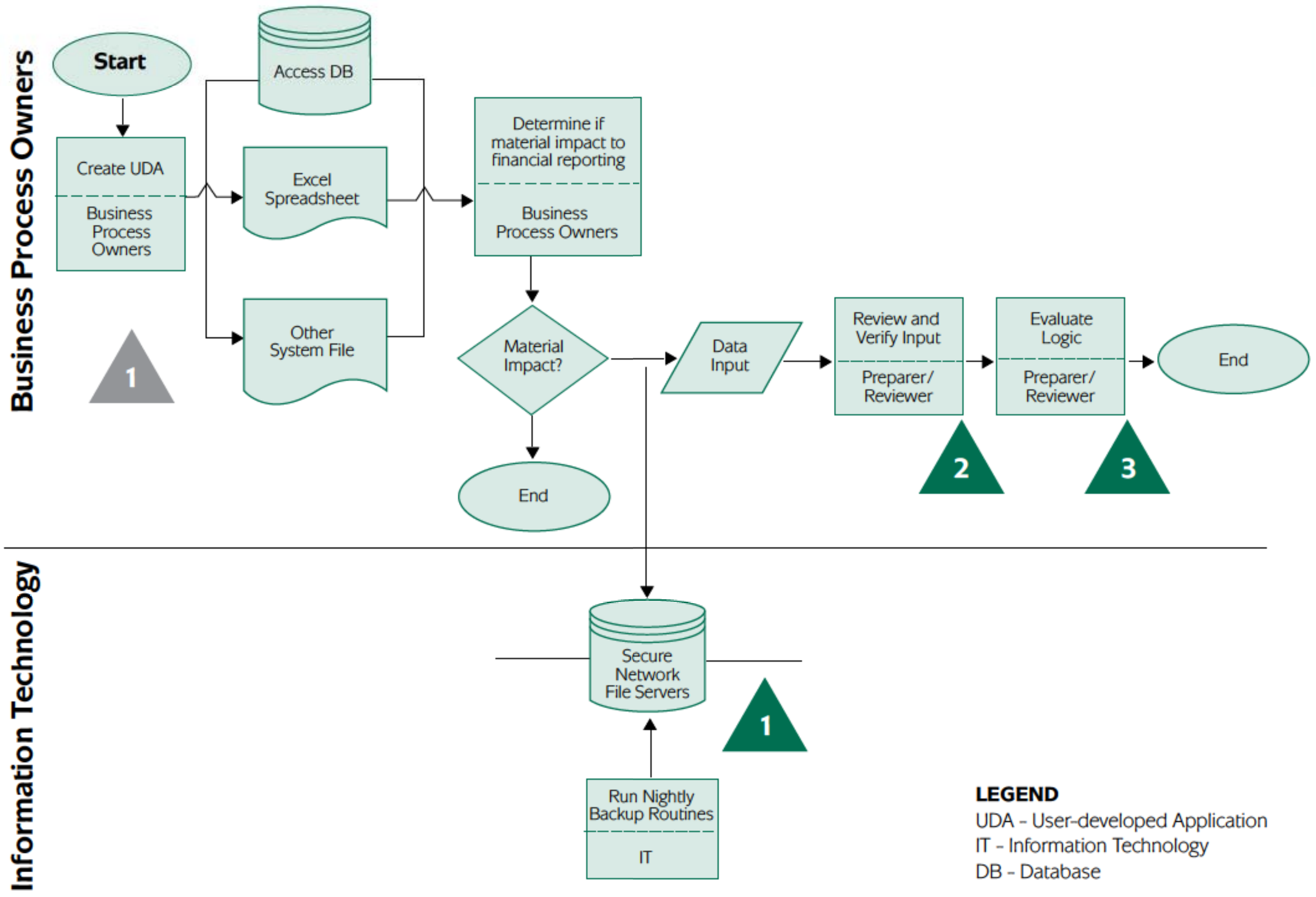
5. Developing the Audit Program

Section 5.1 outlines a sample UDA audit program. The overall intent in providing the audit program is to provide something that could be used for a more complex environ-

[12] "Spreadsheet Compliance in the 2007 Microsoft Office System" White Paper

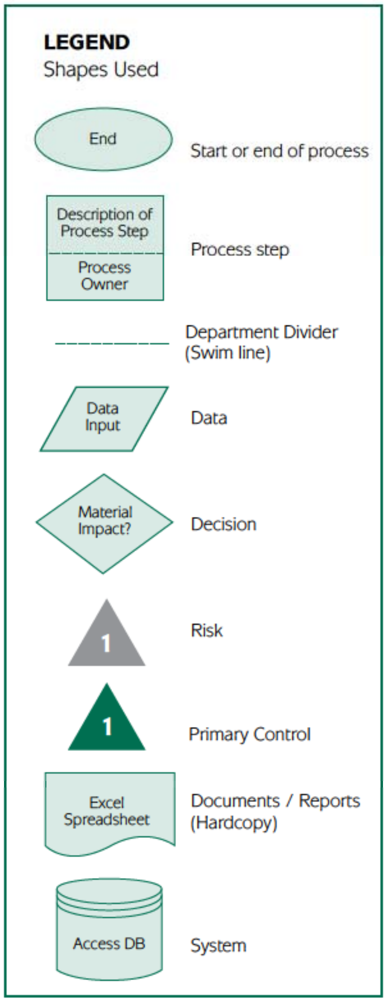
7. Appendix: Sample User-developed Application Process Flow

User-developed Application - Process Flow



LEGEND
UDA - User-developed Application
IT - Information Technology
DB - Database

User-developed Applications - Risk and Controls



Process Risks

▲ 1 Inaccuracies in end-user systems result in financial reporting misstatement.

Process Controls

▲ 1 All spreadsheets and other end-user systems are protected from unauthorized access. Spreadsheets and other end-user systems are saved in secure directories on secure network file servers where access privileges are limited to appropriate people or business groups.

▲ 2 To ensure data is input correctly and completely, the input data is reviewed and verified for reasonableness by both the preparer and reviewer of the spreadsheet or other end-user system.

▲ 3 Changes to the logic or mechanics of the end-user system are reviewed and verified by both the preparer and the reviewers of the spreadsheet or other end-user system.

8. References and Resources

References

- *2008 Annual Report: IT Governance, Risk and Compliance – Improving Business Results and Mitigating Financial Risks*, The IT Policy Compliance Group, 2008.

- ACL Services, Ltd., "*Spreadsheets: A High-Risk Tool for Data Analysis*" White Paper, http://www.treasuryandrisk.com/SiteCollectionDocuments/wp_spreadsheetrisks.pdf (accessed April 14, 2009).

- GTAG-11: *Developing the IT Audit Plan*, The Institute of Internal Auditors, Altamonte Springs, Fla., USA, 2008.

- *Information Technology Examination Handbook*, Development and Acquisition Booklet, The Federal Financial Institutions Examination Council.

- *The International Professional Practices Framework*, The Institute of Internal Auditors, Altamonte Springs, Fla., USA, 2009.

- Microsoft Corp., "Spreadsheet Compliance in the 2007 Microsoft Office System" White Paper, http://www.microsoft.com/downloads/details.aspx?FamilyID=79619EF8-AEA0-40B6-BC8D-74249793DEEF&amp;displaylang=en&displaylang=en (accessed April 14, 2009).

- Report on the First-year Implementation of Auditing Standard No. 5, *An Audit or Internal Control Over Financial Reporting That Is Integrated With an Audit of Financial Statements* (Release No. 2009-006), The Public Company Accounting Oversight Board, 2009.

- *Sarbanes-Oxley Section 404: A Guide for Management by Internal Controls Practitioners*, 2nd Edition, The Institute of Internal Auditors, Altamonte Springs, Fla., USA, 2008.

- Shaid Ansari and Richard Block, "Spreadsheet 'Worst Practices,'" CFO, May 14, 2008, http://www.cfo.com/article.cfm/11288290 (accessed April 14, 2009).

- *Spreadsheet Controls Need a Boost*, ID Number G00166352, Gartner Inc., May 2009.

Resources

Federal Financial Institutions Examinations Council, www.ffiec.gov

The Institute of Internal Auditors, www.theiia.org

IT Policy Compliance Group, www.itpolicycompliance.com

Public Company Accounting Oversight Board, www.pcaobus.org

9. Authors and Reviewers

Authors:

- Christine A. Bellino
- Douglas Ochab, CISA
- Jeffery S. Rowland, CIA, CISA

Reviewers:

The IIA thanks the following organizations and individuals who provided valuable comments and added great value to this guide:

- The American Institute of Certified Public Accountants
- Brad Ames, CISA
- Douglas J. Anderson, CIA
- Ken D. Askelson, CIA, CITP
- David F. Bentley
- Denny K. Beran, CIA, CCSA
- Lawrence P. Brown, CIA, CISA
- Angelina Chin, CIA, CCSA
- Jeanot de Boer
- Steven Hunt, CIA, CISA, CGEIT
- The IIA's Professional Practices Advisory Council:
  - Advanced Technology Committee
  - Board of Regents
  - Committee on Quality
  - Ethics Committee
  - Internal Audit Standards Board
  - Professional Issues Committee
- IIA–South Africa
- IIA–UK and Ireland
- Rune Johannessen, CIA, CCSA, CISA
- David S. Lione, CISA
- Jacques Lourens, CISA, CGEIT
- Michael J. Lynn
- Peter B. Millar
- Fernando Nikitin, CIA, CCSA, CISA, CISM, CGEIT, CISSP
- James Reinhard, CIA, CISA
- Edward Rusiecki, CISA
- Donald E. Sparks, CIA, CISA
- Johannes Tekle, CIA, CFSA
- Archie R. Thomas, CIA
- Karine F. Wegrzynowicz, CIA, CISA
- David Williams, CISA