Modeling and Implementing Internal Control Automation in Compliance with the Sarbanes-Oxley Act

Seokjoo A. Chang*
University at Albany, State University of New York
Mohamed E. Hussein
University of Connecticut, Storrs
Patrick Wing Yin Leung
Caritas Institute of Higher Education, Hong Kong
Kinsun Tam
University at Albany, State University of New York

ABSTRACT
Accountants are responsible for evaluating internal controls, including those in automated information systems, under the Sarbanes-Oxley Act. Competence in IT (Information Technology) and internal control automation is necessary for such evaluation tasks. Accountants, however, typically have limited exposure to IT and automation in their professional training. To prepare accountants for the task of evaluating automated internal controls and to strengthen internal controls of accounting information systems, this paper discusses modeling and implementation of automated internal controls in Relational Database Management Systems. Strengthening of automated internal controls will benefit multiple disciplines including accounting, business management, and IT.

Key words: automated internal controls; constraints; relational algebra; structured query language
JEL: M42

*Corresponding author contact: Email: schang@albany.edu

I. Introduction

The era of digital disruption is rapidly transforming business models and changing the accountant's information environment. SEC (Securities and Exchange Commission) Commissioner Kara M. Stein goes as far as to suggest that accountants have a unique responsibility to move forward in the digital revolution and drive innovations to enhance transparency, accessibility, accountability, and integrity of the capital markets (SEC 2015). To keep pace with technological advancements and stay competitive, accountants need to upgrade the information technology (IT) component of their professional training.[1] The following word cloud summarizes IT skills relevant to future accountants.

Note: Web pages suggesting skills necessary for accountants pursuing an IT career path are downloaded from aicpa.org, careeraddict.com, indeed.com, and roberthalf.com. These pages are condensed into a word cloud, which highlights various software, programming, and data skills relevant to an IT career in accounting.

As a regulated profession, accounting must adapt to regulatory changes, one dimension of which is the accountant's statutory responsibilities in audit engagements. The Sarbanes-Oxley Act (SOA) imposes federal requirements for management to assess the effectiveness of internal control systems, and the accountant to attest to such assessments. SOA has important implications on automated accounting information systems. The SEC (2003), in Rule No. 33-8238 codifying SOA requirements, stipulates that management's assessment must be supported by evidential matter, including documentation, regarding both the design and testing of internal controls. For businesses using computerized accounting information systems, many internal controls such as input validation, reconciliation of subsystems, and generation of exception reports are automated. Automated elements enabled by IT are often embedded in computer programs to prevent, detect, and correct errors (American Institute of Certified Public Accountants or AICPA 2017). Compliance with Rule 33-8238 therefore requires management to assess automated internal control design and testing. To attest to management's assessment, the accountant must understand the design and testing of automated internal controls. Strengthening of IT training will enhance the accountant's capacity to assess these controls.

In response to changes in the information environment and statutory responsibilities, the U.S. Public Company Accounting Oversight Board (PCAOB 2017) asks the accountant to understand the client's use of information technology (IT) and automated controls. In addition, U.S. auditing standard AU-C Section 315 acknowledges the role of automated systems in initiating, authorizing, recording, processing, correcting, and reporting transactions supporting financial statement assertions, and asks the accountant to understand the related IT and IT procedures (AICPA 2017). Such moves to emphasize IT and automated control manifest a concerted international effort spearheaded by the International Auditing and Assurance Standards Board (IAASB 2010) whose International Standards on Auditing (ISA) No. 315 contains a similar language to AU-C Section 315. In fact, major economies such as Australia, Canada, China, Hong Kong, New Zealand, the U.K. and most European countries, as ISA adopters, all include a section in their respective auditing standards or guidelines acknowledging the role of IT and automated internal control in auditing.

Likewise, professional accounting organizations are suggesting changes to the accountant's training. In its specification for the Uniform Certified Public Accountant Examination, the AICPA (2015) asks CPA (Certified Public Accountant) candidates to be proficient in determining how IT affects the effectiveness of internal control. Similarly, the Institute of Internal Auditors (IIA 2016) and the Information Systems Audit and Control Association (ISACA 2012) call on accountants to be proficient in automated internal controls and the technologies underlying such controls in computerized business systems.[2] These calls echo the ongoing international effort to strengthen IT skills of future accountants. Professional accounting associations in Australia, Canada, China, Europe, New Zealand, as well as the rest of the world, are increasingly asking CPA candidates to horn their IT-related professional knowledge and skills for exercising controls over IT systems.[3]

According to the Committee of Sponsoring Organizations (COSO 2013), selecting and developing IT-related internal control activities is a fundamental internal control principle. There are both IT-related benefits and risks to consider in selecting and developing such activities. IT systems are inherently consistent, efficient, timely, and accurate, but they could suffer from program bugs, inaccurate data, unauthorized access to data and programs, failures in system update, accidental data losses, etc. (PCAOB 2017; AICPA 2017). Future accountants need to be proficient in IT benefits and risks to evaluate IT-based internal controls. Skills, knowledge and talents are necessary for mitigating risks in managing IT controls (Khan 2019). Continual computerization in the business environment makes it necessary for accounting professionals and internal auditors to improve IT training (Cangemi 2015). Cangemi (2016) encourages internal auditors to apply technologies to improve the effectiveness of internal audit.

However, accountants are traditionally not trained to become IT experts. Current technical training for accountants is often limited to basic accounting and office software, which is no longer enough (Meyer 2015). Many accountants are not equipped to address technically unfamiliar internal control aspects of SOA compliance audits, which require IT knowledge and technical expertise at a level well beyond what most internal and external accountants have been trained for (Edelstein 2004). Discussion of automated internal controls in traditional accounting information systems and auditing courses is almost non-existing. To update the accountant's training, this paper discusses modeling and implementation of automated internal controls in relational database management systems (RDBMS).

Accountants and auditors must keep abreast of the digital evolution to efficiently and effectively evaluate automated internal controls of information systems. Knowledge of automated internal controls will also enable accountants and auditors, when working as systems analysis and design team members, to communicate effectively with IT professionals in developing new business systems. By minimizing technical jargons and committing to an auditing context, this study customizes the discussion of automated internal control modeling and implementation for accountants and auditors.

As a cost-effective solution for high volume or recurring transactions, automated internal controls are an economically significant topic. The focus of this paper on automated internal controls resonates with the academic literature that documents significant economic consequences arising from weaknesses in IT-based internal controls. According to Haislip et al. (2016), IT-based controls extensively affect the quality of financial statements. Material weaknesses in IT-based controls are found to spread more widely, cost more to correct, impact the control environment more negatively, lead to less reliable financial information, and result in more adverse consequences than other internal control material weaknesses (Haislip et al. 2016; Weidenmier and Ramamoorti 2006; Canada et al. 2009). IT-related material weaknesses in controls bring about weak financial performance and create other internal control material weaknesses (Boritz et al. 2013; Klamm and Watson 2009).

Automated internal controls are relevant to a broad readership of professionals. Use of automated internal controls to enforce compliance with applicable policies and regulations is important to accountants, management, and regulators. Advances in computing and networking technologies have changed the environment of the accounting function. Now, even small firms use computer-based accounting systems and rely on electronic safeguards in such systems to achieve internal control objectives. To competently serve all categories of clients, accountants need to understand automated internal controls.

Moreover, internal controls permeate all areas of accounting and impact other disciplines. Understanding how automated internal controls improve the quality and integrity of accounting information and safeguard assets is relevant to financial accounting, auditing, and investor protection. Automated internal controls improve management of organizations by promoting operational efficiency and effectiveness. Implementation of automated internal controls in electronic systems is also of great value to the e-business and system security professions.

The focuses of this paper on IT and automated internal control are relevant to modern practice and research in both accounting and business. Organizations are seeking to gain a sustainable competitive advantage through business intelligence supported by computer software and hardware and seamlessly integrating IT with business. Public accounting firms are leveraging IT and automation in developing software audit tools (Zhang 2019). Methodologies developed for IT-based internal controls, per Geerts et al. (2013), could contribute to future research on continuous auditing. Capitalizing on internal control automation, continuous auditing engenders real time financial reporting (Rezaee at al. 2002), superior investigation of anomalies (Kogan et al. 2014), timely assurance over the effectiveness of risk management (Malaescu and Sutton 2015), and powerful fraud deterrence and detection (Gonzalez and Hoffman 2018).[4]

The remaining sections of this paper are organized as follows. Section 2 introduces relational databases as a platform for modeling and implementing computerized internal controls. Section 3 discusses the modeling and implementation of computerized internal controls relevant to SOA. Concluding observations and future research directions are presented in Section 4. Appendix A relates RDBMS and SQL to the Semantic Web.

2. Automated internal controls in databases

RDBMS is the dominant database technology capturing 80% to 90% of the database market (Yegulalp 2014; Swoyer 2016). Databases are closely tied to the accountant's job because data supporting financial statement assertions and audit opinions is typically stored in databases. Recognizing this close tie, the AACSB's (2014, p.3, 7) International Accounting Accreditation Standard A7 requires accounting degree programs to include "learning experiences that develop skills and knowledge related to the integration of information technology in accounting and business" and specifically refers to experiences relating to data creation, usage, and storage and information management in databases. Practitioners see the value of database skills as well. For instance, Edelstein (2004) remarks that accountants need detailed understanding of databases, scripts, applications, and electronic reports generation to analyze automated internal controls that have been programmed into accounting systems for data integrity checks, exception handling, and error tracking and reconciliation. Accountants working on forensic technology practices and post-merger accounting with complex databases will need even more extensive database skills (Meyer 2015).

This paper exposes accountants to the coding of automated internal controls in SQL as database constraints. According to an AICPA article, coding (1) is to "do things in sequence and create a clear yet creative road map for organizing [ideas] and problem-solving", which are promising qualities for accountants to have, and (2) helps accountants develop a logical mindset, which benefits all areas of accounting including audit and tax (Meyer 2015). Major CPA firms are increasingly viewing exposure to coding as valuable training for accountants. PwC recommends accounting students to take programming courses in Java or Python, while KPMG looks for double majors in accounting and technology when recruiting (Meyer 2015). The discussion of codes implementing various internal controls in RDBMS in the rest of this paper intends to contribute to the effort of cultivating logical thinking in accountants.

Database technology, when complemented by coding, enables internal control-related automation. Automated processing, including automated data updating systems, provides a simple, low-cost, low-error, and more efficient solution than manual systems (Kokina and Blanchette 2019). Automation of database operations, as in electronic verification between databases of transacting parties, can replace traditional manual assurance procedures (Moffitt and Vasarhelyi 2013). Remediation of internal control shortcomings through automation helps restore the external auditor's reliance on the internal control function (Farkas and Hirsch 2016). Although monitoring of internal control is financially burdensome (Blankley et al. 2019), automated approaches can mitigate the complexity and costs of internal control monitoring and enhance internal control efficiency (Giblin et al. 2006; Masli et al. 2010). Marshall and Lambert (2018) encourage future research into accounting task automation.

3. Modeling and implementing automated internal controls

As SOA obliges accountants to assess the design and testing of internal controls, this section will first discuss how to model internal control design in relational algebra and then implement them in SQL. Learning automated internal controls modeling and implementation in relational algebra and SQL will also help accountants logically express business rules.

Many internal controls can be modeled as constraints. Constraints define the conditions under which data is valid, and hereby specify the desired internal controls. Under the relational model, internal accounting controls are modeled as constraints in relational algebra. If data is stored in RDBMS, these constraints are then implemented in SQL code (Ullman and Widom 2002). Popular RDBMS vendors support a multitude of constraint implementations.

SEC Rule No. 33-8238 requires companies to maintain evidential matter, including documentation, to support management's assessment of the effectiveness of the company's internal controls. Control-related relational algebra expressions and SQL codes form a part of the documentation. Accountants can refer to relational algebra expressions and SQL codes in internal control documentation for precise specification of internal controls.[5] Knowing relational algebra and SQL codes also enables accountants to communicate with information system developers over technical design and implementation issues.

3.1 Modeling

RDBMS is widely popular because its basic modeling concept -- the relation -- is simple to understand and use. The relational model represents data as two-dimensional table (a.k.a. relation). For instance, a table (Employee Table) keeping track of attributes such as employee name, address, employee ID numbers, and withholding tax percentage may appear as follows:

Employee Table

ID	name	address	withholdingPCT
e10011	alice	5 Main St.	15
e20022	bob	10 West St.	20

Similarly, a second table (WagesPayable Table) with employee ID number and amount of wages payable as attributes could take the following form:

WagesPayable Table

employeeID	amount
e10011	1200
e20022	1100
e20033	1500

In the language of the relational model, the name of a table and the set of attributes of the table together constitute the schema (or relation schema) of that table. Therefore, the following schemas represent the structures of Employee Table and WagesPayable Table:

Employee (ID, name, address, withholdingPCT)
WagesPayable (employeeID, amount)

Tables are related to each other by shared data fields called keys. By applying operations to split up or combine existing tables, the relational model defines a new table as an answer to a given query. Relational algebra is a formal notation specifying these operations.[6] The following subsections illustrate how internal controls can be modeled as constraints in relational algebra.

3.1.1. Preventing payment of wages to bogus employees

SEC Rules No. 33-8238 defines "internal control over financial reporting" as a process designed to provide reasonable assurance regarding the reliability of financial reporting and the preparation of financial statements. The SEC then highlights specific internal control policies and procedures, one of which is to provide assurance that expenditures are made only with management authorization. In a RDBMS, check for authorization of payment can be automated. Payment of wages to bogus employees, for instance, can be prevented by automated internal control.

If the Employee table contains only all legitimate employees, it can be used to validate payees receiving periodic wage payments. The internal control that any employeeID value in the WagesPayable table must also exist as an ID value of the Employee table can be modeled as a referential integrity constraint in relational algebra. A referential integrity constraint requires that a value referred to by some object actually exist in the database.[7]

The control below dictates that every employeeID value selected from the WagesPayable table must exist or, in other words, must be traceable to a valid ID value of the Employee table. To illustrate, the last row of the WagesPayable Table storing wage information of employee e20033 constitutes a violation of this constraint because it cannot be traced to the Employee table. If proper constraints were in effect, the original attempt to add e20033 to the WagesPayable table would have been rejected by the RDBMS. This rejection of invalid data, in turn, helps guarantee data integrity.

$$\pi_{\text{employeeID}} (\text{WagesPayable}) \subseteq \pi_{\text{ID}}(\text{Employee})$$

3.1.2. Limiting the size of credit lines

SEC Rule No. 33-8238 requires assessing the effectiveness of the firm's internal controls over financial reporting. This assessment includes, *inter alia*, whether the internal control to prevent or detect material misstatements or omissions is effective. One way to prevent material misstatements is to set reasonable limits. For instance, if credit sales are to be within reasonable size, constraints on credit lines can be set for the following relation schema:

Customer (custID, custName, custAddress, creditLine, creditLineApprovedBy)

The internal control that no credit line can exceed 5 million dollars can be modeled as an attribute value constraint in relational algebra. An attribute value constraint requires that the value of an attribute must be drawn from a specific set or range of valid values. The attribute value constraint below dictates that the set of all creditLine values exceeding 5 million selected from the Customer table must be an empty set. In other words, valid creditLine values must be equal to or below 5 million.

$$\sigma_{\text{creditLine} > 5000000} (\text{Customer}) = \phi$$

3.1.3. Limiting the rank of credit department officer approving credit lines

SEC Rule 33-8238 specifically requires evaluating the operating effectiveness of internal control over transaction initiation. To control the initiation of credit sale transactions, management may create a constraint that only credit department officers with proper rank can grant credit line approval. Such as constraint can be set on the following two relation schemas:

Customer (custID, custName, custAddress, creditLine, creditLineApprovedBy) CreditDeptOfficer (officerID, rank, yearJoined)

If the CreditDeptOfficer table contains all legitimate officers of the credit department, it can help prevent improper approval of credit lines. The internal control that limits the rank of credit department officers approving credit lines to senior manager can be modeled as an attribute value constraint in relational algebra. This attribute value constraint dictates that every creditLineApprovedBy value selected from the Customer table must correspond to a credit department officer at the rank of senior manager.

$$\pi_{\text{creditLineApprovedBy}} (\text{Customer}) \subseteq \pi_{\text{officerID}}\, \sigma_{\text{rank= 'seniorManager'}} (\text{CreditDeptOfficer})$$

3.2 Implementation

When data is stored in a RDBMS, internal controls can be implemented as constraints in SQL. Just like constraints in relational algebra, a constraint in SQL specifies the conditions under which the data is valid. A constraint is called upon whenever values being constrained are modified (including insertions, deletions, or updates). The database software will reject modifications to the database that violate the constraint.

Just like constraints, assertions and triggers can safeguard the integrity of data and are widely supported by commercial RDBMS. Assertions and triggers complement constraints when the business rule to implement is more than what constraints can handle. The following discussion will first cover constraints, which is then followed by assertions and triggers.

3.2.1. Preventing payment of wages to bogus employees

The internal control that any employeeID from the WagesPayable table must also exist as a valid ID value of the Employee table can be implemented as a foreign key constraint in SQL when these tables are created.[8] The SQL statements for creating the tables are provided below. The foreign key constraint in SQL is the equivalent of referential integrity constraint in relational algebra. The foreign key constraint on the WagesPayable table is dependent on the primary key constraint on the Employee table, which must be created first. The foreign key constraint prevents payment of wages to bogus employees whose employeeID values do not match any value of the Employee table's ID attribute.

```
CREATE TABLE Employee
(
  ID CHAR(6) PRIMARY KEY,
  name CHAR(20),
  address CHAR(30),
  withholdingPCT real
)

CREATE TABLE WagesPayable
(
  employeeID CHAR(10),
  amount REAL,
```

```
  payDate DATE,
  FOREIGN KEY employeeID REFERENCES Employee (ID)
);
```

3.2.2. Limiting the size of credit lines

The internal control to ensure that no credit line exceeds 5 million dollars can be implemented as an attribute value constraint in SQL. The following attribute value constraint prohibits any modification of the value of creditLine to above 5000000.

```
CREATE TABLE Customer
(
  custID CHAR(10) UNIQUE NOT NULL,
  custName CHAR(30),
  custAddress VARCHAR(255),
  creditLine REAL CHECK (creditLine <= 5000000),
  creditLineApprovedBY CHAR(10)
);
```

3.2.3 Limiting the rank of credit department officer approving credit lines

The internal control that allows only senior managers of the credit department to approve credit lines can be implemented as an attribute value constraint in SQL. The following attribute value constraint rejects any attempt by any officer other than a senior manager of the credit department to grant or modify credit lines.

```
CREATE TABLE Customer
(
  custID CHAR(10) UNIQUE NOT NULL,
  custName CHAR(30),
  custAddress VARCHAR(255),
  creditLine REAL,
  creditLineApprovedBy CHAR(10),
  CHECK
  (
    creditLineApprovedBy IN
    (
      SELECT officerID FROM CreditDeptOfficer
      where rank = ‘seniorManager’
    )
  )
);
```

3.2.4 Limiting the rank of credit department officer approving credit lines

Internal controls can also be implemented as assertions. An assertion is a form of constraint involving multiple tables to assure that business rules or policies are observed. An assertion is evaluated whenever a table named in the assertion is changed. Conditions specified in an assertion must always be true. Any database modification whatsoever that violates the assertion will be rejected.

As a more general version of constraint, assertions can replace any constraint discussed above. For instance, the internal control based on an attribute value constraint to limit the rank of credit department officer approving credit lines as in 3.2.3 can be rewritten as an assertion. The assertion controls the rank of credit department officer approving credit lines by rejecting any attempt by any officer other than a senior manager of the credit department to grant or modify credit lines.

```
CREATE ASSERTION CheckApproval
CHECK
(
  NOT EXISTS
  (SELECT creditLineApprovedBy from Customer)
  EXCEPT
  (SELECT officerID from CreditDept
     where rank='SeniorManager')
);
```

3.2.5 Enforcing a depreciation policy

SEC Rule 33-8238 specifically requires evaluating the design and testing of controls over selection and application of appropriate accounting policies. One example of accounting policies is depreciation policy, which is illustrated here with the help of the following Equipment relation schema.

Equipment (eID, cost, category, usefulLife)

Assertions guarantee consistent application of a given accounting policy by enforcing functional dependency. A functional dependency between two sets of attributes A and B exists on a table R if "whenever two rows agree on A, they must also agree on B". This dependency is interpreted as "A functionally determines B". Some business rules (those tied to not a single value but a range of possible values) are too abstract to be implemented as attribute value constraints, and require more versatile expressions such as assertions and functional dependencies.

For instance, a firm's depreciation policy may specify that equipment of the same category must have the same useful life. Suppose all computers are depreciated over 3 years, trucks over 6 years, copiers over 6 years, and buildings over 30 years, etc. Under this policy, the equipment category attribute is said to functionally determine the usefulLife attribute because whenever two rows agree on category, they must also agree on usefulLife. If management desires to enforce this policy, the corresponding internal control on the Equipment table can be modeled as the following constraint in relational algebra:

$$\sigma_{E1.eID>E2.eID \text{ and } E1.category=E2.category \text{ and } E1.usefulLife <> E2.usefulLife} (\rho_{E2}(\text{Equipment}) \times \rho_{E1}(\text{Equipment})) = \phi$$

In RDBMS, the internal control that two pieces of Equipment of the same category must agree on the usefulLife attribute is implemented as an assertion in SQL that specifies the corresponding functional dependency. This assertion is evaluated whenever the Equipment table is modified. It will reject any modification that causes two rows with the same value on the category attribute to disagree on the usefulLife attribute.

```
CREATE ASSERTION categoryDetermineUsefulLife
CHECK
(
  NOT EXISTS
   (
    SELECT * FROM Equipment AS E1, Equipment AS E2
      WHERE E1.eID>E2.eID and E1.category = E2.category and
  E1.usefulLife <> E2.usefulLife
   )
);
```

3.2.6 Reporting non-routine changes

SEC Rule 33-8238 requires evaluating the design and testing of internal controls over the initiation and processing of non-routine and non-systematic transactions. One way to monitor unusual activities is to set up an internal control procedure to log any transaction determined to be non-routine for subsequent review by management or the accountant. In RDBMS, such a control procedure can be automated. For instance, to monitor unusual changes to credit lines in the Customer table of 3.1.2, a trigger can be implemented to generate an exception report. Current U.S. auditing standard AS 2401 (PCAOB 2017) specifically requires the accountant to understand the design of automated controls for generating exception reports.

Similar to constraints and assertions, triggers are a means to automate internal controls. While constraints and assertions prohibit violating transactions, triggers perform follow-up actions. A trigger is invoked when certain pre-specified database events (usually insert, delete, or update) occur. It then tests a condition. If the condition is satisfied, the action associated with the trigger, such as generating an exception report, is performed by the RDBMS.

The following trigger implemented in SQL produces a log of unusual changes in credit lines. When the attribute "creditLine" is increased more than 10 times its original value, a record specifying the customer's ID number, the officer approving the change, the old credit line, and the new credit line will be added to the BigCreditChanges table, which is monitored by management or the accountant. This log alerts management or the accountant to non-routine activities.

```
CREATE TRIGGER CreditLineTrigger
AFTER UPDATE of creditLine on Customer
WHEN (OLD.creditLine*10 <NEW.creditLine)
  INSERT INTO BigCreditChanges
  VALUES (NEW.custID, NEW.creditLineApprovedBy,
     OLD.creditLine, NEW.creditLine)
FOR EACH ROW;
```

4. Conclusions

Moffitt and Vasarhelyi (2013) describe current accounting and auditing methods as in danger of becoming anachronistic. As firms increasingly see data as the most significant asset (KPMG 2019), strengthening the internal controls of database systems is a relevant research topic.

Agrawal et al. (2006) propose applying database technologies towards implementing internal controls over financial reporting as required by SOX. Extending Agrawal at al. (2006), this study demonstrates to accountants and other interested readers how automated internal controls are modeled and implemented.

Automated internal controls define the conditions under which data is valid, thus enabling desired business rules to be specified. Automated internal controls can be implemented as constraints, assertions, and triggers, which are supported by RDBMS. An attribute value constraint expresses the requirement that the value of an attribute must be drawn from a specific set of values or lie within a specific range. A referential integrity constraint (foreign key constraint) requires that a value referred to by some object actually exists in the database. Assertions can be used to enforce (1) internal controls affecting multiple tables and (2) abstract business rules (e.g. functional dependencies) such as "equipment category functionally determining equipment useful life". A trigger is a means to implement a certain class of internal controls that require follow-up actions. A trigger tests a condition. If the condition is satisfied, the action associated with the trigger is performed by the RDBMS.

This paper aims at strengthening the accountant's understanding of these automated controls. It examines both modeling and practical implementation of internal controls in RDBMS. SOA requires public accounting firms to attest to management's assessment of internal control effectiveness. Accountants need to understand internal control modeling and implementation to be able to attest to management's assessment of internal control design and testing as required by the SEC.

Automated internal controls are almost never discussed in the accounting literature. On the other hand, constraints, assertions, and triggers discussed in the computer science literature are usually jargon-ridden and lacking a meaningful auditing or internal control context. To open this topic to accountants, this paper highlights the relevance of automated internal controls to accounting and auditing, and discusses automated internal controls in simple examples which are closely tied to SOA.

Our discussions explain how constraints work, and how they are modeled and implemented. While these discussions alone will not convert an accountant to an IT expert, they illustrate critical concepts of automated controls and help the reader gain an understanding of the internal control role of databases.

A unique contribution of this paper is helping the non-technical audience comprehend technical concepts underlying automated internal control. To make this paper palatable to accountants, we have taken care to stay within the accounting and auditing context, choose the most straightforward illustrations, and introduce technical concepts systematically. To the extent possible, jargons are either replaced by or immediately followed by non-technical descriptions. Nevertheless, because of the inherent technical nature of relational algebra and SQL, we acknowledge that some readers may still find the materials challenging.

Constraints, which are powerful safeguards against writing bad data into RDBMS, become less meaningful for read-only mobile applications. Typically normalized to achieve optimal table design, RDBMS may need costly table-joining operations which contradict the low latency expectation of mobile communication. Designed for structured data with predefined schemas, RDBMS is not agile enough to scale-up to high volume of unstructured and semi-structured data in the Big Data era. Therefore, we acknowledge the limitations of RDBMS and RDBMS-based automated internal control. Future research will need to extend automated internal control studies to mobile computing and big data platforms such as JSON (JavaScript Object Notation), NoSql (Not only SQL), document-oriented databases, etc.

Moreover, an assortment of automated and other controls are used in practice. Therefore, integrating automated internal controls with manual and physical controls is a relevant research topic. Future research may investigate how economic factors such as firm size, industry type, the extent of customer support and interaction, etc. affect the optimal balancing of automated versus other internal controls. In addition, high uncertainty-avoidance cultures are expected to prefer more predictable outcomes (Hofstede 2001, p.148). The influence of cultural characteristics on the adoption of automated internal controls, which behave programmatically and thus also predictably, is an interesting topic for future research.

IT comes with both benefits and risks. Future research on IT-based internal controls can help the accounting profession better leverage these controls. Such research can address the following questions: Where are automated controls most needed? Where are they most difficult to introduce? Can these controls be circumvented? What are the resistances to automated controls? What automated controls are needed for extended enterprises? What are the difficulties in retraining accountants for IT-based controls? Findings from such research should guide future adoption and evaluation of IT-based internal controls.

Appendix A

The World Wide Web Consortium (W3C) is an international community that develops open standards to support future growth of the Web (Source: w3.org). RDBMS is relevant to W3C's Semantic Web vision (Wikipedia 2021). To highlight RDBMS's role in the World Wide Web's extension, we illustrate how RDBMS data can be shared electronically across the Semantic Web as RDF (Resource Description Framework) data.

RDBMS, Semantic Web, and RDF

The Semantic Web (aka Web of data) is to extend the World Wide Web by making data machine-readable and accessible across a network, which in turn facilitates trusted data sharing between remote computer systems (W3C 2015). RDF, written in XML, provides a common format to describe information from multiple data sources (i.e. including RDBMS) to be shared digitally and remotely (W3C 2005; W3Schools 2021). The mapping of RDBMS data to RDF extends the reach of RDBMS data, enabling RDBMS users to share data in RDF format with collaborators through the internet (W3C 2012).

SPARQL is a language for querying RDF data. SPARQL supports querying across diverse data sources, including RDBMS sources distributed over a network (W3C 2008). The results of SPARQL can be further refined by filtering, which is a form of Term Constraint. SPARQL filters apply specific restrictions to narrow down the solution set to those results consistent with the filtering expression (W3C 2008). To illustrate the relevance of RDBMS to the Semantic Web and the potential of sharing RDBMS data across a network, the following examples exhibit the conversion of RDBMS to RDF, the validation of RDF data, the tabular and graphical representations of RDF data, and querying of RDF data with SPARQL.

A.1 Employee table converted into RDF

ID	name	address	withholdingPCT
e10011	alice	5 Main St	15
e20022	bob	10 West St.	20

The Employee table, as introduced in Section 3.1, is the starting point of this illustration. W3C (2009) describes various projects initiated, and tools developed, to convert RDBMS data to RDF format. Converting the given Employee table to RDF format produces the following RDF instance document.

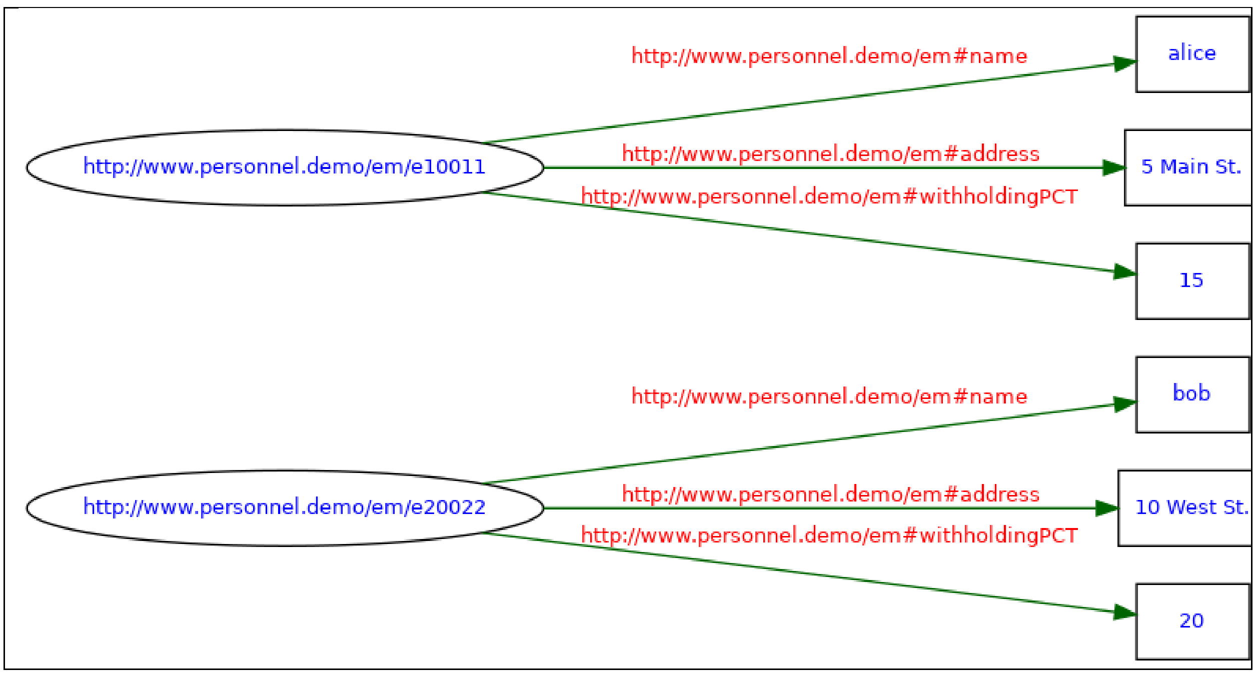
```
<?xml version="1.0"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:em="http://www.personnel.demo/em#">
   <rdf:Description
      rdf:about="http://www.personnel.demo/em/e10011"
      em:name="alice"
      em:address="5 Main St."
      em:withholdingPCT="15" />
   <rdf:Description
      rdf:about="http://www.personnel.demo/em/e20022"
```

```
      em:name="bob"
      em:address="10 West St."
      em:withholdingPCT="20" />
</rdf:RDF>
```

A.2 Tabular view and graphical view of subject–predicate–object triples

W3C's (2006) RDF validation services at https://www.w3.org/RDF/Validator/ parse the RDF instance document, check its syntax, and generate corresponding tabular and graphical views. The tabular and graphical views express RDF data in triples of the form "subject–predicate–object". The graphical view highlights the potential of ultimately representing all RDF data (all triples) in the world as directed labeled graphs. The tabular and graphical views derived from the given RDF instance document are as follows.

Number	Subject	Predicate	Object
1	http://www.personnel.demo/em/e10011	http://www.personnel.demo/em#name	"alice"
2	http://www.personnel.demo/em/e10011	http://www.personnel.demo/em#address	"5 Main St."
3	http://www.personnel.demo/em/e10011	http://www.personnel.demo/em#withholdingPCT	"15"
4	http://www.personnel.demo/em/e20022	http://www.personnel.demo/em#name	"bob"
5	http://www.personnel.demo/em/e20022	http://www.personnel.demo/em#address	"10 West St."
6	http://www.personnel.demo/em/e20022	http://www.personnel.demo/em#withholdingPCT	"20"



A.3 SPARQL query and result set

Just as SQL enables queries against RDBMS, so SPARQL (SPARQL Protocol and RDF Query Language) supports queries against RDF data from diverse data sources across a network (W3C 2008, 2013). In addition, constraints can be specified as filters in SPARQL (W3C 2008, 2013). The following SPARQL query applies a FILTER requiring `withholdingPCT` to be less than 18. As a consequence of this filter, the result set contains only `ID` of "`e10011`" and `withholdingPCT` of "`15`".

```
PREFIX  em:  <http://www.humanResources.demo/em#>
SELECT  ?ID ?withholdingPCT
WHERE   { ?x    em:withholdingPCT   ?withholdingPCT .
          FILTER (?withholdingPCT < 18)
          ?x    em:ID               ?ID . }
```

Result set:

ID	**withholdingPCT**
e10011	15

References

AACSB International. (2014) *AACSB International Accounting Accreditation Standard A7: Information Technology Skills and Knowledge for Accounting Graduates: An Interpretation*. Tampa, Florida, September, 2014.

Agrawal, R., Johnson, C., Kiernan, J. and Leymann, F. (2006 April). Taming compliance with Sarbanes-Oxley internal controls using database technology. In 22nd *International Conference on Data Engineering* (ICDE'06) (pp. 92-92). IEEE.

American Institute of Certified Public Accountants. (2015) *Content and Skill Specifications for the Uniform CPA Examination*. Ewing, NJ: AICPA.

American Institute of Certified Public Accountants. (2017). AU-C Section 315. *Understanding the Entity and Its Environment and Assessing the Risks of Material Misstatement*. Durham, North Carolina: AICPA.

Blankley, A., Hurtt, D. and MacGregor, J. (2019) An exploration of the choice to comply voluntarily with SOX section 404 (b). *Managerial Auditing Journal*, 35(1) pp. 93-110. https://doi.org/10.1108/MAJ-10-2018-2023

Boritz, J.E., Hayes, L. and Lim, J.H. (2013) 'A content analysis of auditors' reports on IT internal control weaknesses: The comparative advantages of an automated approach to control weakness identification', *International Journal of Accounting Information Systems*, 14(2), pp.138-163.

Borthick, A.F. (2012) Designing continuous auditing for a highly automated procure-to-pay process. *Journal of Information Systems*, 26(2), pp.153-166.

Canada, J., Sutton, S.G. and Randel Kuhn Jr, J. (2009) 'The pervasive nature of IT controls: An examination of material weaknesses in IT controls and audit fees', *International Journal of Accounting & Information Management*, 17(1), pp.106-119.

Cangemi, M.P. (2015) Staying a Step Ahead–Internal Audit's Use of Technology. *The Global Internal Audit Common Body of Knowledge (CBOK)*, 16.

Cangemi, M.P. (2016) Views on internal audit, internal controls, and internal audit's use of technology. *The EDP Audit, Control, and Security Newsletter (EDPACS)*, 53(1), pp.1-9.

Chartered Professional Accountants of Canada. (2017) *Annual Report* 2016-2017. https://www.cpacanada.ca/en/the-cpa-profession/about-cpa-canada/annual-reports/2016-2017

Chinese Institute of Certified Public Accountants. 2007. Guidance on Competence of Chinese CPAs. Beijing, China: CICPA. http://www.cicpa.org.cn/BNIE/201611/W020161104386825001095.pdf

Committee of Sponsoring Organizations of the Treadway Commission. (2013) *Internal control – Integrated framework*. Durham, NC: COSO.

CPA Australia and Chartered Accountants Australia and New Zealand. 2015. Professional accreditation guidelines – Australia and New Zealand. Appendix 2: Competency Areas and Learning Outcomes (Detailed). Australia, Melbourne: CPA Australia. https://www.cpaaustralia.com.au/academics/accreditation-guidelines-for-higher-education-programs/professional-accreditation-guidelines/appendix-2

Edelstein. S. (2004) Sarbanes-Oxley Compliance for Nonaccelerated Filers: Solving the Internal

Farkas, M.J. and Hirsch, R.M. (2016) The effect of frequency and automation of internal control testing on external auditor reliance on the internal audit function. *Journal of Information Systems*, 30(1), pp.21-40.

Federation of European Accountants. 2014. Opening a discussion: The Future of Audit and Assurance. Brussels, Belgium: FFE (a.k.a. Accountancy Europe).

https://www.accountancyeurope.eu/wp-content/uploads/Future_of_Audit_and_Assurance_Discussion_Paper_1402.pdf

Geerts, G.L., Graham, L.E., Mauldin, E.G., McCarthy, W.E. and Richardson, V.J. (2013) Integrating information technology into accounting research and practice. *Accounting Horizons*, 27(4), pp.815-840.

Giblin, C., Müller, S. and Pfitzmann, B. (2006) From regulatory policies to event monitoring rules: Towards model-driven compliance automation. *IBM Research Zurich, Report RZ,* 3662.

Gonzalez, G.C. and Hoffman, V.B. (2018) Continuous auditing's effectiveness as a fraud deterrent. *Auditing: A Journal of Practice & Theory*, 37(2), pp.225-247.

Haislip, J.Z., Masli, A., Richardson, V.J. and Sanchez, J.M. (2016) 'Repairing organizational legitimacy following information technology (IT) material weaknesses: executive turnover, IT expertise, and IT system upgrades', *Journal of Information Systems*, 30(1), pp.41-70.

Hardy, C.A. (2014) The messy matters of continuous assurance: Findings from exploratory research in Australia. *Journal of Information Systems*, 28(2), pp.357-377.

Hofstede, G. (2001) *Culture's consequences: Comparing values, behaviors, institutions and organizations across nations*. Thousand Oaks, CA: Sage publications.

International Auditing and Assurance Standards Board. 2010. *International Standard on Auditing 315 Identifying and Assessing the Risks of Material Misstatement Through Understanding the Entity and Its Environment*. New York, NY: IAASB. http://www.ifac.org/system/files/downloads/a017-2010-iaasb-handbook-isa-315.pdf

Information Systems Audit and Control Association. (2012) *ISACA Model Curriculum for IS Audit and Control, third Edition*. Rolling Meadows, IL: ISACA.

Institute of Internal Auditors. (2016) *International Standards for the Professional Practice of Internal Auditing (Standards)*. Lake Mary, FL: IIA.

International Federation of Accountants. (2017) *Handbook of International Education Pronouncements* 2010 Edition. New York, New York: IFAC.

Khan, S. (2019). Mitigating risks in managing cloud-based IT controls. *International Journal of Business Information Systems*, 32(4), pp.425-437.

Klamm, B.K. and Watson, M.W. (2009) 'SOX 404 reported internal control weaknesses: A test of COSO framework components and information technology', *Journal of Information Systems*, 23(2), pp.1-23.

Kogan, A., Alles, M.G., Vasarhelyi, M.A. and Wu, J. (2014). Design and evaluation of a continuous data level auditing system. *Auditing: A Journal of Practice & Theory*, 33(4), pp.221-245.

Kokina, J. and Blanchette, S. (2019) Early evidence of digital labor in accounting: Innovation with Robotic Process Automation. *International Journal of Accounting Information Systems*, 35, p.100431.

KPMG (2019) Capitalize on data's near limitless value to exploit its power. https://assets.kpmg/content/dam/kpmg/xx/pdf/2019/10/data-as-an-asset.pdf

Malaescu, I. and Sutton, S.G. (2015) The reliance of external auditors on internal audit's use of continuous audit. *Journal of Information Systems*, 29(1), pp.95-114.

Marshall, T.E. and Lambert, S.L. (2018) Cloud-based intelligent accounting applications: accounting task automation using IBM Watson cognitive computing. *Journal of emerging technologies in accounting*, 15(1), pp.199-215.

Masli, A., Peters, G.F., Richardson, V.J. and Sanchez, J.M. (2010) Examining the potential benefits of internal control monitoring technology. *The Accounting Review*, 85(3), pp.1001-1034.

Meyer, C. (2015) *Should accounting students learn to code?* July 14, 2015. https://www.aicpa.org/interestareas/accountingeducation/newsandpublications/should-accounting-students-code.html

Moffitt, K.C. and Vasarhelyi, M.A. (2013) AIS in an age of Big Data. *Journal of Information Systems*, 27(2), pp.1-19. https://doi.org/10.2308/isys-10372

Public Company Accounting Oversight Board. (2017) *Auditing Standards of the Public Company Accounting Oversight Board as of December 15, 2017*. Washington, DC: PCAOB.

Rezaee, Z., Sharbatoghlie, A., Elam, R. and McMickle, P.L. (2002) Continuous auditing: Building automated auditing capability. *Auditing: A Journal of Practice & Theory*, 21(1), pp.147-163.

Securities and Exchange Commission. (2003) *Final Rule: Management's Reports on Internal Control Over Financial Reporting and Certification of Disclosure in Exchange Act Periodic Reports. Release Nos. 33-8238; 34-47986*. Washington, DC: SEC.

Securities and Exchange Commission. (2015) *Accountants and Capital Markets in an Era of Digital Disruption: Remarks to the Institute of Chartered Accountants in England and Wales and BritishAmerican Business by Commissioner Kara M. Stein*. https://www.sec.gov/news/speech/remarks-inst-chartered-acctnts.html

Swoyer, S. (2016) *Relational Technology Has Staying Power*. May 17, 2016. https://tdwi.org/articles/2016/05/17/relational-staying-power.aspx

Ullman, J., and J. Widom. (2002) *A First Course in Database Systems*. Upper Saddle River, NJ: Prentice Hall.

W3C. (2005). A Survey of Current Approaches for Mapping of Relational Databases to RDF https://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf

W3C. (2006). Validation service. https://www.w3.org/RDF/Validator/

W3C. (2008). SPARQL Query Language for RDF: W3C Recommendation 15 January 2008. https://www.w3.org/TR/rdf-sparql-query/#restrictNumber

W3C. (2009). Rdb2RdfXG/StateOfTheArt. https://www.w3.org/wiki/Rdb2RdfXG/StateOfTheArt#Proof_of_Concept_Projects

W3C. (2012). A Direct Mapping of Relational Data to RDF. W3C Recommendation 27 September 2012. http://www.w3.org/TR/2012/REC-rdb-direct-mapping-20120927/

W3C. (2013). SPARQL 1.1 Query Language. https://www.w3.org/TR/sparql11-query/

W3C. (2015). Semantic Web. https://www.w3.org/standards/semanticweb/

W3Schools. (2021). XML RDF. https://www.w3schools.com/xml/xml_rdf.asp

Weidenmier, M.L. and Ramamoorti, S. (2006) 'Research opportunities in information technology and internal auditing', *Journal of Information Systems*, 20(1), pp.205-219.

Wikipedia. (2021). Resource Description Framework. https://en.wikipedia.org/wiki/Resource_Description_Framework

Yegulalp, S. (2014) Not so fast, NoSQL -- SQL still reigns. *InfoWorld*. Jun 18, 2014. https://www.infoworld.com/article/2607910/database/not-so-fast--nosql----sql-still-reigns.html

Zhang, C. (2019) Intelligent Process Automation in Audit. *Journal of Emerging Technologies in Accounting*, 16 (2): 69–88.

ENDNOTES

[1] Auditors are a kind of accountants. Statements made about accountants automatically apply to auditors. To avoid unnecessary distractions, this paper does not explicitly differentiate accountants from auditors.

[2] Recognizing automated procedures as a part of the control framework, the Institute of Internal Auditors (IIA 2016) calls on internal auditors to acquire sufficient knowledge of IT risks, IT controls, and IT-based audit techniques. The Information Systems Audit and Control Association (ISACA 2012) encourages accountants to gain proficiency in automated controls and the technologies underlying internal controls in computerized business systems.

[3] The Chartered Professional Accountants of Canada (CPAC 2017) has reviewed its competency framework to ensure all new CPAs are empowered with appropriate information technology skills for the future. The Chinese Institute of Certified Public Accountants (CICPA 2007) emphasizes the need for CPAs to use, evaluate, design, and manage IT systems and exercise controls over such systems. Acknowledging the huge impact of IT and automation on auditing, the Federation of European Accountants (FEE 2014) is reviewing its education and training models. FEE (2014) reminds accountants to maintain IT-related professional knowledge and skills at the required level. Professional accreditation guidelines issued by CPA Australia and Chartered Accountants Australia and New Zealand require future CPAs to be competent in information system controls and RDBMS. The International Federation of Accountants (IFAC 2017) asks professional accountants to be competent in analyzing the adequacy of IT-based general controls and application controls.

[4] Admittedly, realizing the benefits of continuous auditing takes more than an automated control environment. Implementation of continuous auditing is not straightforward (Hardy 2014). Auditors need related experience to effectively implement continuous auditing (Borthick 2012).

[5] Relational algebra and SQL are only two of many means to document automated internal control. The English language is also frequently used in internal control documentation. However, plain English cannot unambiguously express a concept, but a formal language such as relational algebra or SQL can. Unambiguous documentation is needed for computer systems, which are in general ambiguity-intolerant.

[6] In the relational model, a row of data is called a tuple. Relational algebra defines set operations (union, intersection, difference) and special relational operations (selection, projection, join) as follows:
R1 union ($\cup$) R2 produces a table containing all tuples that appear in R1, R2, or both.
R1 intersect ($\cap$) R2 produces a table containing all tuples that appear in both R1 and R2.
R1 difference (-) R2 produces a table containing all tuples of R1 that do not appear in R2.
The selection (σ) operation selects tuples from a table whose attributes meet the selection criteria,
The projection (π) operation chooses a subset of the columns in a table, and discards the rest.
The join operation ($\bowtie$) pairs up tuples from two tables that agree in whatever attributes are common.
R1 $\subseteq$ R2 is true if R1 is a subset of R2.
ϕ denotes the empty set.
The Cartesian product (x) of two sets A and B is the product set containing all possible combinations of one element from each set.
ρ denotes renaming.

[7] A simple example can help explain this referential integrity concept. If an author forgets to include a table in a research paper, reference in the text of the paper to the table will constitute a violation of referential integrity.

[8] Alternatively, constraints can be added subsequent to table creation with "ALTER TABLE" statements:
ALTER TABLE Employee ADD CONSTRAINT idAsKey PRIMARY KEY (ID);
ALTER TABLE WagesPayable ADD CONSTRAINT employeeIdAsForeignKey FOREIGN KEY (employeeID) REFERENCES Employee (ID);