

# GEMS Tabulation Database Design Issues in Relation to Voting Systems Certification Standards

Thomas P. Ryan<sup>1</sup> and Candice Hoke<sup>2</sup>

## Abstract

*This paper analyzes the Diebold Election Systems, Inc. election management software (GEMS) using publicly accessible postings of GEMS election databases. It finds that the GEMS architecture fails to conform to fundamental database design principles and software industry standards for ensuring accurate data. Thus, in election tabulations, aspects of the GEMS design can lead to, or fail to protect against, erroneous reporting of election results. Further, GEMS's dependence on Microsoft's JET technology introduces additional risks to data accuracy and security.*

*Despite these technical and systemic deficiencies, GEMS received approval as complying with Federal Voting System 2002 standards. Questions then arise concerning the adequacy of the 2002 and 2005 regulatory standards. The paper concludes that the standards structurally encourage and reward election system vendors for using less exacting database design standards.*

With unprecedented Federal funding available to States under the Help America Vote Act of 2002 (HAVA),<sup>3</sup> election administration has become highly reliant on computer technologies. While some continue to praise the new voting and tabulation technologies as a significant advance, the augmented computerization has introduced new possibilities for wide-impact election operational errors and may have opened new avenues for tampering with election results. Previous vulnerability analyses have focused on a direct-recording electronic (DRE) voting machine,<sup>4</sup> a paper ballot optical scanning device,<sup>5</sup>

computerized vote-tallying,<sup>6</sup> and a pilot test of internet voting.<sup>7</sup> But the systemic design features of currently utilized election tabulation databases have yet to be closely examined.

This paper analyzes the Diebold Election Systems, Inc. (DESI) election management software named Global Election Management System ("GEMS") using publicly accessible postings of GEMS election databases.<sup>8</sup> It finds that the GEMS architecture violates fundamental design principles and software industry standards for ensuring accurate data. When utilized for election tabulations, the GEMS design can lead to data errors, which in turn create a serious risk for generating erroneous election results. GEMS architectural design plus its use of Microsoft's JET technology,<sup>9</sup> introduces significant risk of data errors in elections administered using GEMS.

Either of these design aspects would be worrisome. For the GEMS database (DB) to have been structured with fundamental flaws at the levels of both system architecture and system technology, and yet still obtain Federal and State certification, raises questions concerning the adequacy of the existing regulatory standards. Thus the paper turns to ask what the relationship is between the regulatory standards and the technical database flaws. It argues the regulatory standards structurally encourage low DB design standards rather than promoting the use of tabulation system architecture that meet widely recognized industry standards for data accuracy and reliability.

This paper proceeds by briefly reviewing the DB design principles of the First and Second Normal Forms. In part II, the paper examines the GEMS DB in light of these fundamental design principles, concluding that GEMS does not satisfy even the most

---

<sup>1</sup> J.D., 2007; Technical Staff, Center for Election Integrity, Cleveland State University.

<sup>2</sup> Director, Center for Election Integrity and Associate Professor of Law, Cleveland State University. This paper was submitted to EVT/USENIX on April 23, 2007, accepted for publication on June 1, 2007, and will be presented at the EVT '07 Conference on August 6, 2007. A longer version will be available by August 1, 2007 (posted in the Working Papers section, Center for Election Integrity website, [www.urban.csuohio.edu/cei/](http://www.urban.csuohio.edu/cei/)) that is styled for the nontechnical audience. The Center initiated the Collaborative Public Audit of the November 2006 election in Cuyahoga County cited here, and its staff provided technical analysis for the audit.

<sup>3</sup> 42 U.S.C. §§ 15301 – 15545 (2006).

<sup>4</sup> Ariel J. Feldman et al., *Security Analysis of the Diebold AccuVote-TS Voting Machine*, (Sept. 13, 2006), at <http://itpolicy.princeton.edu/voting/ts-paper.pdf>.

---

<sup>5</sup> Hursti, Hari, *Critical Security Issues with Diebold Optical Scan Design*, (July 4, 2005), at <http://www.blackboxvoting.org/BBVreport.pdf>.

<sup>6</sup> Saltman, Roy G., *Accuracy, Integrity, and Security in Computerized Vote-Tallying*, NBS Special Publication 500-158, (August 1988), <http://www.itl.nist.gov/lab/specpubs/500-158.htm>.

<sup>7</sup> David Jefferson, Ari D. Rubin, Barbara Simons, David A. Wagner, *A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE)*, at <http://www.cs.berkeley.edu/~daw/papers/serverreport.pdf>

<sup>8</sup> See <http://www.equalccw.com/dieboldtestnotes.html> and <http://www.bbvforums.org/forums/messages/2197/44189.html>. This paper's GEMS assessment is perforce limited to examples of the end product but the design flaws are discernible at this level.

basic, essential precepts of the First Normal Form. Further, its use of JET technology renders GEMS susceptible to additional difficulties. Part III critically evaluates the federal regulatory structure and standards for certifying election management software. It concludes that the federal standards produce the unintended and injurious consequence of rewarding poor database designs with lower vendor research and development costs, and faster movement through less intensive certification reviews than if the DB design were more sophisticated.<sup>10</sup>

## I. Database Design Fundamentals

Any successful database (DB) must accurately and precisely store data without mixing values or losing information--an obvious essential in managing election results data. To diminish the incidence of anomalies which reduce the accuracy of DB contents, computer science and engineering have established fundamental DB design precepts, including "normalization."<sup>11</sup>

Normalization is a methodology of DB design that creates proper relations, removes redundant data, promotes efficient use of disk space,<sup>12</sup> and reduces the likelihood that accessing and manipulating data will result in anomalies. Normal form classification uses consecutive, progressive numerical titles (e.g., 1NF) to describe in shorthand whether a particular DB has satisfied the fundamental design precepts. If a DB design has not been normalized, the DB has been designed in a manner that fails to prevent avoidable errors and data corruption.<sup>13</sup> For example, when the DB design causes storage of specific data in multiple locations or tables, updates to that data can cause anomalies to occur. Failure to update the specific data in every location virtually simultaneously causes inconsistencies in the data between the two locations (an update anomaly). Normalized DBs also create correct dependencies<sup>14</sup> among data sets. Incorrect

dependencies can create errors when data is added or deleted from the DB.

### A. Normal Forms

#### 1. First Normal Form (1NF)

As Edgar Codd has outlined,<sup>15</sup> satisfaction of the first Normal Form requires a DB design to (a) eliminate repeating groups in individual tables (atomicity);<sup>16</sup> (b) identify each set of related data with a primary key;<sup>17</sup> and (c) create a separate table for each set of related data.<sup>18</sup>

Violations of the first Normal Form (1NF) include the flaws of repeating groups, the absence of unique identifiers, the inclusion of multiple meaningful values in a single field, and the inclusion of multiple columns representing the same type of atomic data. Data corruption is highly probable if any of these violations are found within the DB design.

#### 2. Second Normal Form (2NF)

The overarching purpose of the Second Normal Form (2NF) is to reduce the amount of redundant and duplicate entries within a DB. A DB table satisfies 2NF if (a) it conforms to 1NF and (b) each non-primary key element is dependent upon the primary key.<sup>19</sup> DB satisfaction of 2NF means tables with repeating information separate the repeating data and reference those records through the use of "integrity constraints." Integrity constraints provide a method to ensure data entry changes or updates do not result in a loss of data consistency.<sup>20</sup> The most common tool deployed is known as a foreign key

The first and second Normal Forms contain the most fundamental design principles for efficient and accurate DBs. Any DB that fails to satisfy the first two Normal Forms will suffer various failures upon deployment.

---

<sup>10</sup>The term "database" within this paper is limited to a modern relational database. Owing to limitations where proprietary software is protected from certain types of evaluative reviews the examination of GEMS DB design and implementation issues is not comprehensive. Instead, the paper seeks to serve as a starting point for future computer science, industry, and regulatory public policy analyses.

<sup>11</sup> Edgar F. Codd, *Normalized Data Base Structure: A Brief Tutorial*, Proceedings of 1971 ACM-SIGFIDET Workshop on Data Description, Access and Control, 1-21 (November 11-12, 1971).

<sup>12</sup> Ponniah, Paulraj, *Database Design and Development*, at 308-9 (3d ed. 2003).

<sup>13</sup>"Data corruption" as used in this paper and in computer science indicates a departure from the original or from what is pure or correct; the term need not import malevolent intent or an operator's deliberate intrusion to modify stored values.

<sup>14</sup> Dependencies are relationships between data, where one value depends on another. To credit the correct number of votes to candidate Joe Smith, one table may specify an identifier as '1234',

---

and then the identifier 1234 is defined in a separate table to be Joe Smith

<sup>15</sup> Codd, Edgar F., *A Relational Model of Data for Large Shared Data Banks*, 13 (6) Comm. of the ACM 13, 377-87 (June 1970) at <http://www.acm.org/classics/nov95/toc.html>.

<sup>16</sup> A repeating group is one that is not atomic, that is, holds more than one meaningful entry per data block.

<sup>17</sup> A primary key is a unique identifier; most commonly the table will start with the integer 1 and increase entry by a factor of one. e.g. 1,2,3,4, etc.

<sup>18</sup> Related data is data with such a strong relationship that it should not be separated. A common example is the elements of an individual's address. (i.e. Street number, street, city, state, zip).

<sup>19</sup> See Codd, note 12.

<sup>20</sup> Abraham Silberschatz et al., *Database System Concepts*, 193 (3d ed. 1999).

## II. GEMS Database Design Flaws

This paper's analysis of the GEMS DB design is based on review of publicly available GEMS election DBs that are publicly accessible via the internet.<sup>21</sup>

### A. System Architecture Design Flaws

Analysis of the GEMS DB architecture<sup>22</sup> demonstrates that it violates both 1NF and 2NF.

#### 1. Violations of 1NF

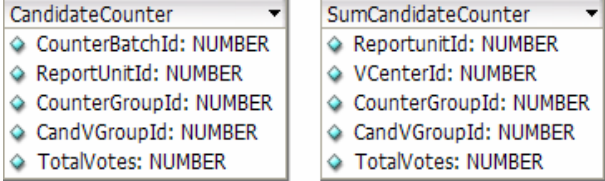
The GEMS DB design violates fundamental principles of DB architecture for it fails to conform to several 1NF principles. Consequently, GEMS is susceptible to the common errors and anomalies that 1NF seeks to eliminate. Most troubling, nonconformity with 1NF can cause erroneous data to be entered into the DB through normal operation of the system. System failures can then occur without an operator knowing or having any indication that the system is failing because the DB lacks essential design constraints in place to prevent invalid data.

**First 1NF Violation:** GEMS's Race table violates 1NF because it has multiple columns representing the same type of atomic data. The Race table is structured for two columns to contain the same type of atomic data, VGroup1Id and VGroup2Id, but the purpose of having two columns cannot be distinguished from examining the table alone. Through normal operation, the GEMS design creates unnecessary processing and uses DB storage inefficiently. GEMS thus violates one of the main purposes of the 1NF: eliminating duplicative columns from the DB.<sup>23</sup>

**Second 1NF Violation:** GEMS includes multiple meaningful values within a single field as demonstrated by the VCenter table of GEMS. The VCenter table holds information regarding polling locations but because of the column "Label," the VCenter table violates 1NF: multiple meaningful values are held within the same field. Combining data in this manner makes it difficult to query voting locations and allows for numerous entries for the same polling location.

**Third 1NF Violation:** Several GEMS tables lack a unique identifier, a failure demonstrated by review of Figure 1. Within the two Counter tables, the third

and forth entries of both Counter tables are indistinguishable from one another other than by their position in the table.



CandidateCounter	SumCandidateCounter
CounterBatchId: NUMBER	ReportunitId: NUMBER
ReportUnitId: NUMBER	VCenterId: NUMBER
CounterGroupId: NUMBER	CounterGroupId: NUMBER
CandVGroupId: NUMBER	CandVGroupId: NUMBER
TotalVotes: NUMBER	TotalVotes: NUMBER

Figure 1: GEMS tables CandidateCounter and SumCandidateCounter

These tables additionally attempt to avoid data duplication via deployment of MS Access-specific functions, a strategy which has proved to be unreliable.<sup>24</sup>

**Other 1NF Violations** The design of a number of other GEMS tables crucial to producing accurate election results reports violate 1NF principles, with flaws similar to those inventoried above.<sup>25</sup>

#### 2. Violations of 2NF

The GEMS DB design reveals little if no attempt to conform to 2NF principles. GEMS lacks data constraints that ensure data integrity,<sup>26</sup> and omits referential constraints that ensure data consistency.<sup>27</sup> GEMS also replicates the same data throughout numerous locations and tables.

The most troubling violation of the 2NF involves the duplication of data between two tables holding vote tallies. In Figure 1, for instance, both CandidateCounter and SumCandidateCounter tables hold total vote data for a candidate -- which is identified by the CandVGroupID column. This dual depositing scheme for election results data can easily generate update anomalies — otherwise known as inconsistencies in election results data for a given candidate and race. After an update anomaly has occurred, it is impossible to rectify the inconsistency without identifying when the anomaly occurred, or starting again from the beginning point when data processing began. If an election ended and the tables held different numbers in each table, the question

<sup>21</sup> See note 8 above.

<sup>22</sup> DESI's GEMS software includes components for electronic ballot creation and other tasks but this paper restricts its scope to the GEMS database design for interacting with JET to manage election tabulations and the reporting of results.

<sup>23</sup> Kent, W., *A Simple Guide to Five Normal Forms in Relational Database Theory*, 26 Comm. of the ACM 120-25 (1983).

<sup>25</sup> The longer version of the paper contains this discussion. See note 2, above.

<sup>26</sup> Data integrity constraints ensure the data type allowed is the correct one for the field, such as only allowing positive integers for a vote total field.

<sup>27</sup> Foreign keys are not utilized correctly, if at all.

becomes which vote total should be accepted as the correct value.<sup>28</sup>

Eliminating the opportunity for these types of anomalies is precisely the objective of 2NF. GEMS's design thus violates 2NF in tables essential to producing accurate and consistent election results reports.

Additional exemplars of GEMS' departures from 2NF can be identified. DB architects use data integrity constraints to ensure the type of data is correct for a given field. These limits ensure a program cannot input letters where numbers are expected or a negative number where only positive numbers should be allowed. The SumCandidateCounter table (see *Figure 1*), holding vote tally information, violates 2NF in allowing negative values to be entered into the table.

VCenterId is the polling location where votes were registered, and thus should always have a positive value associated with it.<sup>29</sup> A table that permits incorrect negative values, such as *Figure 2*, vitiates one identifier of DB corruption.

Reportunit Id	VCenter Id	Counter GroupId	CandV GroupId	Total Votes
829	-1	0	1	246
829	-1	0	2	45
829	-1	0	3	231
829	-1	0	4	51
829	-1	0	5	252
829	-1	0	6	56
829	-1	0	7	230
829	-1	0	8	49
829	-1	0	9	231

Figure 2, Sample Data from GEMS SumCandidateCounter

Allowing negative values into the VCenter column means tracking the origin of votes cannot occur<sup>30</sup> and the value '-1' might, but does not necessarily, indicate the table and/or DB is corrupted.<sup>31</sup> Placing data constraints upon the VCenter table would force the values to remain positive integers.

Integrity constraints provide the foundation for managing data that resides in several interrelated tables. The election data management code within the

<sup>28</sup> The Cuyahoga County (Ohio) Collaborative Public Audit (of the November 8, 2006 General Election) Final Report, which was issued after this paper had been drafted, mentions the inconsistent tables containing election results data as a troubling feature for the accuracy and reliability of its election data. See [http://urban.csuohio.edu/cei/public\\_monitor/cuyahoga\\_2006\\_audit\\_report.pdf](http://urban.csuohio.edu/cei/public_monitor/cuyahoga_2006_audit_report.pdf) at 34-36 (April 19, 2007; hereafter Cuyahoga 2006 Election Audit Report).

<sup>29</sup> The VCenter table has ids ranging from 1 to 302. There is no entry for -1.

<sup>30</sup> Votes may not be able to be tracked at all if there are duplicate entries for all values in the table.

<sup>31</sup> Repeating values, such as the value "-1," can indicate corruption for some database engines such as Microsoft JET.

GEMS software lacks this essential foundation for data accuracy. Although the GEMS software holds some form of data management, without integrity constraints it is only a matter of time before major problems surface. While modifying the GEMS architecture to include integrity constraints would not completely solve the problem, it would assist in ensuring data accuracy.

## B. System Technology Flaws: Use of JET

Microsoft's Joint Engine Technology (JET) is a basic DB engine<sup>32</sup> technology that is appropriate for personal computing and very small scale applications requiring DB technology. Commercially known as Microsoft Access®, JET is a file-sharing DB that can support DBs with sizes up to 2 gigabytes.<sup>33</sup> JET is often considered ideal for small DB deployments with very few concurrent user/processes,<sup>34</sup> and can also be used by custom programs to access the data through the Microsoft Data Access Components Application Programming Interface (MDAC API).

But JET's limitations have led Microsoft (MS) to state that JET is inappropriate for systems that require data integrity, security, and transaction logs and rollbacks.<sup>35</sup>

Microsoft JET ... was not intended (or architected) for the high-stress performance required by 24x7 scenarios, ACID transactions, or unlimited users, that is, scenarios where there has to be absolute data integrity or very high concurrency.<sup>36</sup>

An election management system obviously requires both "absolute data integrity" and in many urban jurisdictions if not all, a "very high concurrency" of users. Thus, the GEMS' architects' choice of inexpensive JET as the DB engine places the entire election tabulation process at very high risk.

<sup>32</sup> A database engine is the underlying software that creates, retrieves, updates, and deletes information from the database.

<sup>33</sup> Luke Chung & Dan Haught, *When to Migrate from Microsoft Access to Microsoft SQL Server*, (2005) at [http://download.microsoft.com/download/5/d/0/5d026b60-e4be-42fc-a250-2d75c49172bc/when\\_to\\_Migrate\\_from\\_Access.doc](http://download.microsoft.com/download/5/d/0/5d026b60-e4be-42fc-a250-2d75c49172bc/when_to_Migrate_from_Access.doc).

<sup>34</sup> Concurrent users and processes are those attempting to use the database at the same time. In the election tabulation context, concurrent usage could include the uploading of election results from scores of DRE units operating simultaneously, or from dozens of optical scanners.

<sup>35</sup> *Microsoft Access or SQL Server: What's Right in your Organization?*, (2005) at <http://www.microsoft.com/sql/solutions/migration/access/compare-access.msp>

<sup>36</sup> *Using Microsoft JET with IIS*, (Rev. 6.1 2007) at <http://support.microsoft.com/default.aspx/kb/222135>.

## 1. Capacity Limit of 2 Gigabytes

GEMS exacerbates JET's limitations and can lead to DB failure. Its 2 gigabyte limit can easily be exceeded in large turnout urban elections, especially where central count scanning is utilized.<sup>37</sup> If the capacity limit is exceeded, database corruption is highly probable.

## 2. Data Corruption During Normal Operation

As Microsoft documentation has stated:

When Microsoft JET is used in a multi-user environment, multiple client processes are using file read, write, and locking operations on a shared database. Because multiple client processes are reading and writing to the same database and because Jet does not use a transaction log (as do the more advanced database systems, such as SQL Server), *it is not possible to reliably prevent any and all database corruption.*<sup>38</sup>

Because this is a file-locking DB system,<sup>39</sup> the operating system (Windows) could function as a "user" that locks the DB file. Corruption of JET DBs can occur from hardware conflicts from peripherals,<sup>40</sup> software conflicts,<sup>41</sup> multi-user access,<sup>42</sup> and an overall poor DB design.<sup>43</sup>

## 3. Multi-User Access Limitation

Software systems that utilize a DB typically have multiple clients or users that attempt to access the data at the same time but the JET DB engine is not designed to manage such simultaneous requests.

---

<sup>37</sup>A Microsoft spokesperson confirmed the Cuyahoga Audit Committee's finding that Microsoft recommended a different system for operations as large as Cuyahoga County's. See Bob Driehaus, *Audit Finds Many Faults in Cleveland's '06 Voting*, *N.Y. Times* Section A, Page 16 (April 20, 2007). The GEMS-JET database can be compressed and backed up but each operation introduces additional risks of database corruption.

<sup>38</sup>*How to Troubleshoot and to Repair a Damaged Access 2002 or Later Database*, (Rev. 6.1 2006) at

<http://support.microsoft.com/default.aspx?scid=kb;en-us;283849>

(emphasis added); see also *NY Times*, note 37 above ("Scott Massey, a Microsoft spokesman, said any file-based database was subject to corruption if a connection was lost while a transfer was in progress"); and Cuyahoga 2006 Audit Report at page 67, cited in note 28 above.

<sup>39</sup> When a process is accessing the database, it prevents all other concurrent access. This "locks out" all other processes until the first process has completed its tasks.

<sup>40</sup> Hardware conflicts occur between hardware devices such as two network cards in use on one machine.

<sup>41</sup> Software conflicts occur between software programs both accessing the database.

<sup>42</sup> Multi-user access conflicts occur when multiple users are using the same program, each accessing the database concurrently.

<sup>43</sup> A poor design includes the lack of normalization described previously within this paper.

Microsoft has recommended that fewer than ten clients concurrently access the DB,<sup>44</sup> but single users have also created concurrency errors.<sup>45</sup>

In a GEMS election tabulation, Windows can be one of the processes accessing the DB. In Ohio, during election tabulations the DB is monitored by GEMS as well as a State-mandated security program, DigitalGuardian (DG). Thus, at a minimum, in Ohio GEMS is faced with mediating three potential concurrent clients of the DB. In addition to these three programs, during uploading and processing of election data, GEMS is accessing the DB at a very high rate.

Further, GEMS must mediate a large number of concurrent data requests of the DB during election tabulations. At some points, data is simultaneously being uploaded to the DB from multiple sources (for instance, 30 memory cards); snapshot election results reports are requested (generating data analysis requests from the DB); and software audit logging is occurring (both Windows events logging and GEMS audit logging).

The context of election tabulations ineluctably presents GEMS with a high rate of data concurrency and throughput -- exactly the situation Microsoft has warned can cause DB corruption in its Microsoft JET technology. GEMS cannot be an exception to JET's core deficiencies. This constellation of issues raises very serious questions on whether GEMS is capable of managing and producing accurate election tabulations and other data reports.

## 4. Microsoft Deprecation of JET Components

"Deprecation" is a term used by software companies to notify end users and software developers that a portion of a product line or Application Programming Interface (API)<sup>46</sup> will not be supported in future releases. Microsoft has decided to deprecate MS Data Access Components (MDAC); future releases of JET will not include the MDAC

---

<sup>44</sup> "Jet can support up to 255 concurrent users, but performance of the file-based architecture can prevent its use for many concurrent users. In general, it is best to use Jet for 10 or fewer concurrent users." Fitzgerald, James, *Microsoft Data Engine (MSDE) for Microsoft Visual Studio 6.0: An Alternative to Jet for Building Desktop and Shared Solutions*, (2002) at <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmsde/html/msdeforvs.asp>

<sup>45</sup> *Single User Concurrency Issues with ADO and JET*, (2001) at <http://support.microsoft.com/kb/216925/EN-US>.

<sup>46</sup> Application Programming Interfaces allow programmers to reuse code, such as the code used to communicate to the JET database engine. APIs allow programmers to use the functionality of established code, such as the JET engine, without "reinventing the wheel."

components.<sup>47</sup> The GEMS software utilizes MDAC to communicate with the JET DB engine.<sup>48</sup>

It remains unclear whether DESI was marketing GEMS after MS published the deprecation notice yet omitted disclosure of the point and its consequences to prospective purchasers, election administrative jurisdictions. Software that utilizes the JET MDAC components, such as GEMS, will likely need to be re-written to utilize a different DB technology, and potentially at a high cost for any jurisdictions transitioning to a new GEMS product.

## II. Compliance with Federal Voting System Standards

Section 222(e) of HAVA declared the 2002 Voting System Standards (VSS) to be HAVA's first set of voluntary voting system technical standards. Via section 221, Congress authorized a Technical Guidelines Development Committee (TGDC)<sup>49</sup> to develop recommendations for improvements to the Voting System Standards of 2002,<sup>50</sup> in the expansive time frame of nine months. The TGDC's 2005 prolix recommendations address voting system performance standards (Volume I) and testing standards (Volume II).<sup>51</sup> As a matter of federal law, the 2005 VSS remain voluntary rather than compulsory on VS manufacturers although some States have mandated VS equipment be certified by federally approved independent testing laboratories.

Volume I of VSS 2005 incorporates much of the early FEC standards from 2002.<sup>52</sup> The new Security section is written in highly technical language and adds some substantial overdue protections for voting systems technical security. But whatever its improvements for security and other issues, Volume I of the 2005 VSS omits a requirement that qualifying election tabulation databases must satisfy 1NF and 2NF.<sup>53</sup>

---

<sup>47</sup>Microsoft has stated "Starting with version 2.6, MDAC no longer contains Jet components. In other words, MDAC 2.6, 2.7, 2.8, and all future MDAC releases do not contain Microsoft Jet, Microsoft Jet OLE DB Provider, or the ODBC Desktop Database Drivers." Shirolkar, Prash, *Data Access Technologies Roadmap*, (2004) at [http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmdac/html/data\\_mdacroadmap.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmdac/html/data_mdacroadmap.asp)

<sup>48</sup> *GEMS User's Guide – Version 1.17.15*, (Rev. 3, 2001) available at <http://freespeech.metacolo.com/pimaupgrade.zip>.

<sup>49</sup> 42 U.S.C § 15361 (2006); the TGDC reports its recommendations to the U.S. Election Assistance Commission.

<sup>50</sup> 42 U.S.C § 15361(b)(2) (2006).

<sup>51</sup> Voluntary Voting System Guidelines Version 1: Initial Report, (2005) at <http://vote.nist.gov/VVSGVol1&2.pdf>. (hereinafter VSS 2005).

<sup>52</sup> The major new sections of this volume include the Human Factors section and the Security section.

<sup>53</sup> Volume I includes basic functions that an election DB must manage, including "identify contests, candidates and issues"; "define ballot formats and appropriate voting options"; "accumulate vote totals at multiple reporting levels as indicated in the system

But DB design issues fall easily within the scope of voting systems (VS) technical standards and within the TGDC expertise. If the TGDC is able to create and develop detailed standards regarding highly technical security concerns, it would appear also to possess the regulatory scope and technical resources to develop election tabulation DB design and implementation standards.

Volume II of the 2005 VSS is primarily concerned with testing standards for the "qualification" or certification process. It focuses upon the specific details for Independent Testing Authorities (ITAs, now renamed), vendors, and election officials in the qualification process. Like Volume I, Volume II also adopts and reaffirms a vast amount of the 2002 VSS testing standards. The TGDC significantly updated the standards, however, to include changes to reflect the U.S. Election Assistance Commission's process for certification of voting systems and HAVA's usability and accessibility requirements.

In its description of VS testing requirements, Volume II offers no new references regarding election tabulation DB design specifications or testing procedures. Thus, the DB testing standards remain as they were in 2002 without specific requirements or constraints on the designs for an effective and reliable DB.

Volume II lists the documentation that must be provided to the Independent Testing Authority ("ITA") before the election management software (including tabulation functions) can be qualified. In detailing the required vendor DB documentation, the VSS provides the most specific standards for DB design requirements.<sup>54</sup> This Volume II section, however, only requires substantial DB documentation to be provided to the ITA if the specifically listed DB design paradigms were utilized. Those vendors whose DB designs are not reflected in specified paradigms—and thus less likely to be soundly designed—are not required to supply the additional DB documentation. If a vendor chooses to design a DB using paradigms such as entity relationships, or security and privacy constraints, it then must submit substantial documentation to the ITA.

The upshot of this regulatory approach is that the VS vendor who offers a poorly designed tabulation DB that can still meet the minimum requirements set forth in Volume I, Section 2.2.6, can likely reach the testing/certification phase faster than the vendor seeking to market a better designed DB. Moreover,

---

documentation"; "generate the post-voting reports required by Section 2.5" but omits any reference to satisfaction of fundamental DB design requirements. See Vol. I of VSS 2005, section 2.2.6 (cited in note 46). Software standards found in Volume I, section 4, similarly sidestep DB design precepts.

<sup>54</sup> VSS 2005, Volume II, section 2.5.8 (see note 46).

this low horizon vendor will experience substantially lower costs for DB design and VSS required documentation. More documentation requires a larger financial investment for the vendor.

The VSS 2005, and especially Volume II, section 2.5.8, therefore creates a disincentive for election system vendors to design DBs that adhere to sound well, established design paradigms. This incentive runs exactly counter to the overwhelming public interest in accurate and reliable election tabulations. This regulatory inversion regarding DB design standards suggests that the entire VSS 2005 should be analyzed to identify other sections that may inadvertently create incentives that undermine the public interest in accurate, secure elections, and to provide pointers for the next VSS revision.

## **Conclusion**

The GEMS DB has not been designed to adhere to fundamental normalization principles that can permit a DB to reach high standards of accuracy and reliability. But the federal regulatory apparatus has also failed, for it imposes differential documentation requirements and financial burdens on vendors seeking certification for their election DB software. Those vendors who attempt to achieve higher design standards face far greater burdens and costs, including possibly more delays, than vendors who settle for DB designs with lower horizons. None of the 2005 VSS standards constitute a mandatory federal floor for voting systems to be deployed in federal elections. The certification of the GEMS software notwithstanding the significant demonstrable design flaws, offer a clear demonstration of the inadequacy of the current certification regime.

## **Acknowledgements**

The authors appreciate the significant assistance of Andrew W. Appel, David Kettyle, George H. Taylor, and David A. Wagner, and the advice of Aaron Burstein and Joe Hall in preparing this paper. Errors are the responsibility of the authors alone.