A Database Management Assessment Instrument

Jeffrey P. Landry
jlandry@southalabama.edu

J. Harold Pardue
hpardue@southalabama.edu

Roy Daigle
RDaigle@southalabama.edu

Herbert E. Longenecker, Jr.
longeneckerb@gmail.com

School of Computing
University of South Alabama
Mobile, AL 36688, USA

Abstract

This paper describes an instrument designed for assessing learning outcomes in data management. In addition to assessment of student learning and ABET outcomes, we have also found the instrument to be effective for determining database placement of incoming information systems (IS) graduate students. Each of these three uses is discussed in this paper. We describe the use of a pre/post test, item validation, and correlation techniques for the purpose of validation and assessment. Although the instrument was developed for local assessment, its design is based on international information systems curriculum guidelines rendering it suitable for use in any program which incorporates database management in its curriculum.

Keywords: assessment, database, data management, exams, outcomes

1. INTRODUCTION

Universities are increasingly being required to demonstrate that student learning is occurring at their institutions in measurable, documented ways, and that these measurable results are being used to improve their educational programs. Assessment of learning has become a requirement of institutional and program accreditation. Many methods of assessment are possible, including internally/externally developed, direct/indirect measures of performance, and formative/summative indicators. Often these assessment approaches are developed for "local" use, i.e. they are not designed to be generalized for use by similar programs at peer institutions. This paper describes the development, validation, use, and results interpretation of a database exam—an internally-developed, direct assessment, formative indicator of student learning in a four-year information systems (IS) degree program—that we believe can be used for assessment in any program requiring a database management course. In the sections that follow, we describe the foundation for the exam, the approach taken for developing and verifying exam items, the approach taken for validating that the exam is a useful instrument for student outcomes assessment, and a discussion of the several uses that we have made of the instrument.

2. BACKGROUND

The exam was developed in the mid-2000's as an outgrowth of a national certification exam project, and for use at the co-authors' university—the University of South Alabama (USA), located in Mobile, Alabama. Available from the Center for Forensics, Information Technology, and Security, the USA-CFITS DB Exam consists of 25 multiple choice items, 16 of which appear on the IS 2002 exit exam, a national certification exam for information systems exit skills (Landry, Reynolds, & Longenecker, & 2003).

The original reason for creating the exam was to address a graduate program placement issue. Students admitted to the information systems master's program had traditionally been placed into the graduate data management course based on the prerequisite of having passed an undergraduate database course. Despite having transcript evidence of an undergraduate database management course at other institutions, some students were not prepared to succeed in our graduate database course. Since our undergraduate course was designed to satisfy course objectives consistent with learning units in IS 2002 and since graduate students who successfully completed our undergraduate database course also successfully completed the graduate database course, we concluded that a placement exam was needed to accurately determine when the undergraduate course should be a required prerequisite. Subsequently, the database placement exam was created to be given to incoming master's students, and used as a placement mechanism. Students making a passing score were admitted to the graduate data management course, while students making a failing score were advised to complete the undergraduate database course with a passing grade of 'C' or better.

Development and Validation of the Exam

The USA-CFITS DB Exam was originally designed to be a measure of data management knowledge and skills, one of the fundamental core areas of Information Systems curricula (Landry, Longenecker, Haigood, & Feinstein, 2000; Haigood 2001; Colvin 2008). The foundations for the exam are database-related learning units (LU) of IS curricula models, IS'90, IS'97, and IS2002 (Longenecker & Feinstein, 1991; Longenecker, Feinstein, Couger, Davis, & Gorgone, 1995; Davis, Gorgone, Couger, Feinstein, & Longenecker, 1997; Gorgone, Davis, Valacich, Topi, Feinstein, & Longenecker, 2003). The continuing relevance of database skills and knowledge in the IS curricula models is further supported by the results of two surveys—one targeting faculty and industry partners (Landry et al., 2000) and a second targeting IS professionals two to four years beyond graduation (Colvin, 2008).

Specific knowledge and skill areas used to motivate item writing for the USA-CFITS DB Exam were drawn from prior work reflecting an intersection of academic and professional needs. Henderson, Champlin, Coleman, Cupoli, Hoffer, Howarth, Sivier, Smith, & Smith (2004) published a framework for Data Management curricula intended for postsecondary education and sponsored by a professional society, the Data Management Association (DAMA). Longenecker, Henderson, Smith, Cupoli, Yarbrough, Smith, Gillenson, & Feinstein (2006) studied this framework in detail and found that the skills were compatible with the IS2002 and IS2010 IS curriculum guidelines. Table 5 in the appendix reflects a synthesis of the DAMA framework, the IS model curriculum guidelines, and a job ad analysis (Landry et al., 2000; Haigood 2001).

In developing the USA-CFITS DB Exam to reflect both professional skills and curriculum guidelines, the authors wrote items that assessed the intersection of a data management sub-skill area and an IS 2002 learning unit. The learning objectives for each of the 25 items on the USA-CFITS DB Exam are as follows:

1. Given a piece of data to programmatically manipulate, choose the appropriate data type
2. Given a real-world application, determine appropriate fields to be stored in a file
3. Choose and defend the correct data type for representing a common data attribute
4. Differentiate between entities and attributes when developing an ERD
5. Recognize the need either for an intersection table in a M:N relationship or the need to revisit requirements to determine if there is a missing entity

6. Given a relational database description, evaluate the architecture
7. Given a system need, such as access control to a database, identify the necessary information
8. Differentiate among alternatives for enforcing data integrity constraints
9. Compare and contrast the processes involved in data modeling
10. Recognize the implication of a cascade delete
11. Recognize the notation of standard ER models
12. Recognize and describe a correct three-entity solution to a problem expressed as a many-to-many relationship between two entities
13. Recognize that many-to-many relationships require a third, linking table in a relational DB
14. Apply the knowledge of using a stored procedure to enhance the performance in a database environment
15. Given database design goals, identify correct techniques for implementation
16. Normalize (redesign) an unnormalized (poorly designed) table
17. Recognize correct syntax and correct use of views
18. Recognize the implication of using views in a client application
19. Recognize the advantages and disadvantages of implementation with stored procedures
20. Trace and debug SQL syntax
21. Recognize the correct formulation of a query
22. Differentiate normal forms as part of database design
23. Recognize which tasks are associated with discovering and eliciting database design requirements in the initial phase of requirements analysis
24. Recognize relevant factors involved in the purchasing decision of a major enterprise level DBMS package
25. Recognize properties of the Entity-Relationship Model, particularly the concept of minimum cardinality

Since the development of the USA-CFITS DB Exam, a revision of the information systems curriculum guidelines has been issued. IS 2010, available at http://www.acm.org/education/curricula, defines core course IS 2010.2 as *Data and Information Management*. All 25 USA-CFITS DB Exam items map to a stated course objective of the IS 2010.2 course. Of the 25 items, 13 of them map to course objectives 6, 8, and 12, dealing with conceptual data modeling, designing a high quality database, and various SQL commands, and 13 of the 21 course objectives are covered by at least one exam item.

The exam item objectives were also mapped to ABET student outcomes criteria (ABET, 2007, p. 14). The outcomes criteria, along with the number of exam items mapped to each, are shown in Table 1. See Table 5 in the appendix for a grand mapping of the 25 item objectives with IS 2002, IS 2010 and ABET.

Table 1 - Coverage of ABET Student Outcomes

	Student Outcomes that must be enabled	Number of associated exam item objectives
The program has documented measurable outcomes that are based on the needs of the program's constituencies	(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline	1
	(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution	5
	(c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs	12
	(i) An ability to use current techniques, skills, and tools necessary for computing	7

It is important that an internal exam designed for assessment be mappable into multiple assessment frameworks. Doing so strengthens the validity of the exam's content as being relevant outside of the local unit's needs. For more on the approach used to map multiple assessment frameworks, write items, and validate exams, see related papers (Landry et al., 2003; Landry , Daigle, Longenecker, &

Pardue, 2010; Reynolds, Longenecker, Landry, Pardue, & Applegate, 2004).

Exam Construction

The multiple mappings established a useful foundation for item writing, which was carried out using these and other good practices in educational assessment (Hogan 2007; Crocker & Algina 1986). The writers wrote items and objectives in alignment with mapped frameworks. An item consisted of a stem with four possible answers with one correct answer. Good item writing was difficult, and multiple reviewers were utilized in the item review process. The entire item-writing and review process was supported by a web-based exam delivery system developed by the co-authors and their graduate students at the University of South Alabama. The candidate items were pilot tested, revised, and validated with statistical techniques, including test item statistics. See Section 3 – Validation below for details. A summary of recommended practices includes the following:

- Define objectives, and write items that target the objectives
- Map items into other outcomes for assessment value
- Don't write items that are too difficult
- Make sure items are based on knowledge
- Get multiple reviewers to rigorously review items, and correct
- Pilot test the exam
- Use test item statistics to validate
- Make exam easy to administer and score
- Select an appropriate passing score
- Develop good security policies

See Figure 1 for an overview of the item construction process.

A cut score for passing was set at 44% correct responses. The success rate of students in our graduate database course correlated with whether the student made at least a 44. A score of 44 correlated with a midrange 'C' performance in our undergraduate database course. While the score of 44 would seem low for a student who has taken a database management course, an explanation is that scores for this external exam are predictably lower than scores on internal assessments that reflect an individual instructor's preferences in instructional approach and topic emphasis. Furthermore, we designed the items on the exam to be discriminating, that is, to differentiate between those who know and those who don't, perhaps to a higher degree than instructors do in general.

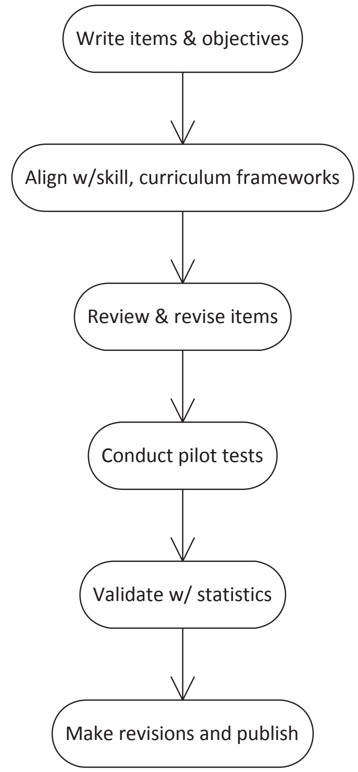

Figure 1 - Item Construction Process

Multiple Uses of the Exam

The faculty eventually found multiple uses for the exam in addition to graduate data management course placement. In the undergraduate database course, the exam is given as a pre-test at the beginning of the course and as a post-test incorporated as part of the final exam. This practice provides the capability of assessing the degree to which the undergraduate database course is achieving its intended learning outcomes, independent of instructor assignment (especially part-time instructors) and in different delivery formats (traditional, blended, fully online). This results are used as a formative program assessment method for both ABET and regional accreditation agencies (e.g. SACS).

3. VALIDATION

The results of using the exam over three years are described next. The first test described is a test using content experts. This test was intended as a face validity test, but also demonstrated content validity. The panel of experts, which consisted of professors from the university using the exam, took the test as a student would, in a proctored lab environment.

Overall, observations made by the experts included a perception that the test items are discriminating, that is, they are effective at discriminating between whether someone knew the answer or would have to guess. The perception among the content experts is testable. See discussion of item validation and pre/post testing below. Another positive reaction from an expert after taking the test was that "I knew what the item was about, but don't know if I got it right." This comment was interpreted as meaning the item was about a relevant database concept familiar to the expert, but that the item was also challenging. Another expert said that it was helpful that the exam had a consistent format of diagrams and tables that accompanied some of the items, as well as re-use of data in tables. Such consistency cuts down on the cognitive overload on takers. The eight items (of 25) that use tables or figures depict ER models, queries, or tables/views of data. One expert liked the "normalization item", another liked the item on "intersection tables" (which table gets the foreign key?").

More critically, the experts thought that "four or five items need revisiting (more review)." Some jargon was recognized as being potentially confusing to students, including the use of United States zip codes on a data types item. The toughest items were believed to be those on triggers and constraints. The experts were skeptical of items that presumed a specific order of database life cycle activities. Another item asked about the "best way" to do something, and it was believed the item to be too normative.

The second set of tests we conducted was to run statistical analyses on the most recent set of test taker data. We calculated summary and item statistics, and conducted pre/post tests, and ran correlations of test vs. course performance.

Summary and Test Item Statistics

From January 2008 until May 2010, a total of 246 USA students, a combination of graduate and undergraduate students, English speaking and ESL students, took the USA-CFITS DB Exam. Over this period, 53.4 was the mean score with standard deviation of 14.6. This score is consistent with national norms for the information systems exit exam. The highest score was a 92, and the lowest score was a 16. Eight test takers, or a little more than 3 percent of all takers, scored below 25, or worse than guessing.

The KR20, which measures internal item consistency, was 0.62. The score is right above a minimally acceptable score of 0.60, which is recommended for tests in a subject domain taken by those trained in that domain.

Table 2 - Item Statistics

Pct Correct	Point Biserial
43	0.45
64	0.36
58	0.24
65	0.46
40	0.40
50	0.51
80	0.30
54	0.26
58	0.25
34	0.20
40	0.12
81	0.41
75	0.43
86	0.19
32	0.34
58	0.14
72	0.26
28	0.21
87	0.29
30	0.51
39	0.36
53	0.34
26	0.30
28	0.30
46	0.44

Some test item statistics are provided in Table-2 below. This table indicates the percentage of subjects getting each item correct, which varies from 26% to 87%, and the point biserial, which varies from .12 to .51. The percent correct scores indicate item difficulty on a 100-point scale, with a 100 representing the easiest (least difficult) item, that is, with 100% of takers answering it correctly. Higher point biserials are indicative of items that correlate well with the exam as whole, especially when values are 0.40 and higher.

Pre and Post tests

The purpose of a pre/post test is to demonstrate that learning took place between the two measurements. In our case, we gave the USA-CFITS DB Exam to incoming graduate students. Those (25 students) who failed to make a passing score were required to take an undergraduate database course, and three other students who barely passed also decided to take the database course.

Table 3 - Pre/Post Test Results

Taker #	**Pre-test score**	**Post-test score**	**Difference b/w pre & post**
1	24	52	28
2	32	48	16
3	36	56	20
4	28	52	24
5	16	56	40
6	40	56	16
7	28	60	32
8	36	68	32
9	40	76	36
10	48	68	20
11	44	68	24
12	32	44	12
13	24	44	20
14	40	48	8
15	40	48	8
16	20	40	20
17	40	48	8
18	32	32	0
19	64	72	8
20	24	56	32
21	40	68	28
22	36	36	0
23	32	48	16
24	32	44	12
25	40	52	12
26	40	60	20
27	40	56	16
28	36	44	8
# Failed	**25**	**3**	
# Passed	**3**	**25**	
Total takers	**28**	**28**	
Pct takers passed	**11%**	**89%**	
Mean score (0-100)	**35.1**	**53.6**	**18.4**

At the end of the database course, they again took the placement exam. These two sets of scores were compared using a paired t-test, using PASW Statistics. There were 28 students in the sample. The pre/post test scores are in Table 3 as follows.

By the end of the course the results were reversed. There were now 25 passing scores and three that were still below passing (although one of those improved by 20 points) for a pass rate of 89%. The pre-test mean was 35.1, compared to a post-test mean of 53.6. The mean difference was 18.4 points, and the result of a paired differences test was statistically significant at a .001 level (p=.000). Such a result is a strong indicator of learning taking place in the course. It was particularly remarkable that the increase in scores occurred despite the fact that many of the students in the sample had prior database experience and scored close to passing in the pre-test.

If the test maps well to the objectives of the course, and the pre-test is given to those with little knowledge of the subject matter, a pre/post test design ought to detect whether learning is taking place. In this way, we can use the USA-CFITS DB Exam to verify that the undergraduate course is achieving its planned learning outcomes, over time, especially as the instructor changes. Once a pre/post relationship is established, it might be sufficient just to give the post-test, and compare the post test mean to historical post-test averages.

Correlations of test taker performance vs. database course performance

Over time (see Table 4), we determined that the scores on the exam correlated as follows:

Table 4 - Exam-Course Correlations

Score on USA-CFITS DB Exam (% correct)	Associated letter grade in the course
60-100	A
50-59	B
40-49	C
30-39	D
0-29	F

The grading scale on an exam like this is not the same as a typical 10-point scale used commonly in universities, with 90-100 A, 80-89 B, etc. The items on the exam, while representative of a first database course, are not particular to a specific institution's database course or its instructor.

We believe that instructors taught the database course in an unbiased manner towards the exam. It should be noted that that data includes scores from students in sections taught by two of the co-authors, one of whom also developed questions for this exam. The co-author's approach in teaching the course was not to teach to the test, nor use exam items elsewhere in the course. The other instructors had no access to the exam items before, during, and after the pre/post tests.

4. CONCLUSION

In summary, the benefits of using the exam are as follows:

- Maps to ABET outcomes
- Provides instructor-independent assessment of learning
- Can use as a placement exam for grad program or transfer students
- Useful for outcomes assessment for ABET accreditation
- Useful for course assessment

With the growing demand for more outcomes-based assessment in higher education, the use of this type of internally-developed exam, while becoming necessary, will offer many benefits. Among these are instructor-independent course and program outcomes assessment that supports multiple frameworks. We have shown that the USA-CFITS DB Exam is aligned with international curriculum models, ABET outcomes and job-related skills from two surveys (Landry et al., 2000; Colvin, 2008). With the specific exam being described, the USA-CFITS DB Exam, we have provided evidence that success in a first database course is most closely correlated with mastery of a specific subset of learning outcomes in data management. We described how we were able to converge on a cut score that predicted whether or not a graduate student needed to take a database prerequisite course. We provided evidence that post-test student scores parallel their local course performance, while trending lower than local scores for predictable reasons (i.e. exam is not specific to an instructor or the local course). All this made the exam useful for student placement and course assessment.

We believe that the need for more and better assessment helps make efforts like ours worthwhile. To inquire about use of the exam, contact the University of South Alabama Center for Forensics, Information Technology, and Security (USA-CFITS, http://www.usacfits.org).

5. REFERENCES

ABET Computing Accreditation Commission - ABET CAC (2007). Self-Study Questionnaire. ABET, Inc. August 2007. Baltimore, Maryland.

Colvin, R. (2008). Information Systems Skills and Career Success, Master's Thesis, University of South Alabama, School of Computer and Information Sciences.

Crocker, L. and Algina J. (1986). Introduction to Classical and Modern Test Theory. Holt, Rinehart and Winston. Orlando, Florida.

Davis, G., Gorgone J., Couger J., Feinstein D., and Longenecker H. (1997). IS'97: Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *ACM SIGMIS Database*, 28(1).

Gorgone, J., Davis G., Valacich J., Topi H., Feinstein D., and Longenecker H. (2003). IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *Data Base* 34(1).

Haigood, B. (2001). Classification of Performance Level Requirements of Current Jobs Within the Field of Information Systems, Master's Thesis, University of South Alabama, School of Computer and Information Sciences.

Henderson, D., Champlin B., Coleman D., Cupoli P., Hoffer J., Howarth L., Sivier K., Smith A., and Smith E. (2004). Model Curriculum Framework for Post Secondary Education Programs in Data Resource Management. The Data Management Association International Foundation, Committee on the Advancement of Data Management in Post Secondary Institutions, Sub Committee on Curriculum Framework Development.

Hogan, T. (2007). Educational Assessment: A Practical Introduction. John Wiley & Sons, Inc., Danvers, Massachusetts.

Landry, J., Daigle R., Longenecker H., and Pardue H. (2010). IS 2002 and ABET Accreditation: Meeting the ABET Program Outcome Criteria. Information Systems Education Journal, August 6, 2010, Volume 8, Issue No. 67, URL: http://isedj.org/8/67/.

Landry, J., Longenecker H., Haigood B., and Feinstein D. (2000). Comparing Entry-Level Skill Depths Across Information Systems Job Types: Perceptions of IS Faculty. Americas Conference on Information Systems (AMCIS 2000), August 2000, Long Beach, California.

Landry, J., Reynolds J., and Longenecker H. (2003). Assessing Readiness of IS Majors to Enter the Job Market: An IS Competency Exam Based on the Model Curriculum. Americas Conference on Information Systems (AMCIS 2003), August 2003, Tampa, FL.

Longenecker, H., and Feinstein D. (Eds.) (1991). IS'90: The DPMA Model Curriculum for Information Systems for 4 Year Undergraduates. Park Ridge, Illinois: Data Processing Management Association

Longenecker, H., Feinstein D., Couger J., Davis G., and. Gorgone J. (1995). Information Systems '95: A Summary of the Collaborative IS Curriculum Specification of the Joint DPMA, ACM, AIS Task Force. Journal of Information Systems Education, Volume 6, Number 4, pp. 174-187.

Longenecker, H., Henderson D., Smith E., Cupoli P., Yarbrough D., Smith A., Gillenson M., and Feinstein D. (2006). A Recommendation for a Professional Focus Area in Data Management for the IS2002 Information Systems Model Curriculum. In The Proceedings of the Information Systems Education Conference 2006, v 23 (Dallas): §2115. ISSN: 1542-7382

Reynolds, J. H., Longenecker H., Landry J., Pardue J., and Applegate B. (2004). Information Systems National Assessment Update: The Results of a Beta Test of a New Information Systems Exit Exam Based on the IS 2002 Model Curriculum. Information Systems Education Journal, May 1, 2004, Volume 2, Issue Number 24, URL: http://isedj.org/2/24/

Editor's Note:

This paper was selected for inclusion in the journal as a ISECON 2012 Distinguished Paper. The acceptance rate is typically 7% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2012.

APPENDIX: Table 5 - Grand Mapping of the USA-CFITS DB Exam

Skill					Skill Words				
1.1.3	Data Types and File Structures				analysis, design, development, debugging, testing, simple data structures (arrays, records, strings).				
#	**ABET Outcome**	**LU**	**IS2002 LU-Title**	**IS2002 LU-Goal**	**IS 2010**	**Outcome**	**Item Objective**	**% Correct**	**PtBiSer**	**Group Avg% Correct**
1	b Analyze	58	Problem Solving, with Files and Database	to present and ensure problem solving involving files and database representations	2.113	Implement a relational database design using an industrial-strength database management system, including the principles of data type selection and indexing.	**given a piece of data to programmatically manipulate, choose the appropriate data type**	0.43	0.45	**0.55**
2	b Analyze	42	Information Measurements/ Data /Events	to present the concept that data is a representation and measurement of real-world events	2.05	Apply information requirements specification processes in the broader systems analysis & design context.	**given a real-world application, determine appropriate fields to be stored in a file**	0.64	0.36	
3	b Analyze	58	Problem Solving, with Files and Database	to present and ensure problem solving involving files and database representations	2.11	Implement a relational database design using an industrial-strength database management system, including the principles of data type selection and indexing.	**choose and defend the correct data type for representing a common data attribute**	0.58	0.24	

1.3.1		**Modeling and design, construction, schema tools, DB systems**				Data modeling, SQL, construction, tools -top down, conceptual, logical and physical designs; scripts; bottom up designs; schema development tools; desk-top/enterprise conversions; systems: Access, SQL Server/Oracle/Sybase, data warehousing & mining; scripts, GUI tools; retrieve, manipulate and store data; tables, relationships and views				
13	a Basics	89	ADTs: Database Models and Functions	to develop awareness of the syntactical and theoretical differences between database models	2.11	Implement a relational database design using an industrial-strength database management system, including the principles of data type selection and indexing.	**recognize that many-to-many relationships require a third, linking table in a relational DB**	0.75	0.43	
23	b Analyze	111	IS Requirements and Database	to develop requirements and specifications for a database requiring multi-user information system	2.07	Link to each other the results of data/information modeling and process modeling.	**recognize which tasks are associated with discovering and eliciting database design requirements in the initial phase of requirements analysis**	0.26	0.30	
25	b Analyze	111	IS Requirements and Database	to develop requirements and specifications for a database requiring multi-user information system	2.08	Design high-quality relational databases.	**recognize properties of the Entity-Relationship Model, particularly the concept of minimum cardinality**	0.46	0.44	**0.50**
6	c Build	81	IS Database Applications Development	to develop application skills for implementing databases and applications by operating and testing these databases	2.06	Use at least one conceptual data modeling technique (such as entity-relationship modeling) to capture the information requirements for an enterprise domain	**given a relational database description, evaluate the architecture**	0.50	0.51	
8	c Build	81	IS Database Applications Development	to develop application skills for implementing databases and applications by operating and testing these databases	2.08	Design high-quality relational databases.	**differentiate among alternatives for enforcing data integrity constraints**	0.54	0.26	

10	c Build	88	IS Data Modeling	to develop skill with data modeling which describe databases	2.06	Use at least one conceptual data modeling technique (such as entity-relationship modeling) to capture the information requirements for an enterprise domain	**recognize the notation of standard ER models**	0.34	0.20
11	c Build	88	IS Data Modeling	to develop skill with data modeling which describe databases	2.06	Use at least one conceptual data modeling technique (such as entity-relationship modeling) to capture the information requirements for an enterprise domain	**recognize and describe a correct three-entity solution to a problem expressed as a many-to-many relationship between two entities**	0.40	0.12
12	c Build	88	IS Data Modeling	to develop skill with data modeling which describe databases	2.15	Understand the basic mechanisms for accessing relational databases from various types of application development environments.	**compare and contrast the processes involved in data modeling**	0.81	0.41
16	c Build	90	IS Database and IS Implementati on	to develop skill in application of database systems development and retrieval facilities needed to facilitate creation of information system applications	2.10	Design a relational database so that it is at least in 3NF.	**normalize (redesign) an un-normalized (poorly designed) table**	0.58	0.14
21	c Build	92	IS Database Application Implementati on	to develop skill with application and physical implementation of database systems, using a programming environment	2.12	Use the data definition, data manipulation, and data control language components of SQL in the context of one widely used implementation language.	**recognize the implication of using views in a client application**	0.30	0.51
4	i Tools	58	Problem Solving, with Files and Database	to present and ensure problem solving involving files and database representations	2.06	Use at least one conceptual data modeling technique (such as entity-relationship modeling) to capture the information requirements for an enterprise domain	**differentiate between entities and attributes when developing an ERD**	0.65	0.46

21	i Tools	92	IS Database Application Implementation	to develop skill with application and physical implementation of database systems, using a programming environment	2.12	Use the data definition, data manipulation, and data control language components of SQL in the context of one widely used implementation language.	**recognize correct syntax and correct use of views**	0.39	0.36	
1.3.2		**Triggers, Stored Procedures, Audit Controls: Design / Development**				Triggers, audit controls-stored procedures, trigger concepts, design, development, testing; audit control concepts/standards, audit control Implementation; SQL, concepts, procedures, embedded programming (e.g. C#)				
5	c Build	81	IS Database Applications Development	to develop application skills for implementing databases and applications by operating and testing these databases	2.06	Use at least one conceptual data modeling technique (such as entity-relationship modeling) to capture the information requirements for an enterprise domain	**recognize the need either for an intersection table in a M:N relationship or the need to revisit requirements to determine if there is a missing entity**	0.40	0.40	**0.57**
15	c Build	90	IS Database and IS Implementation	to develop skill in application of database systems development and retrieval facilities needed to facilitate creation of information system applications	2.11	Implement a relational database design using an industrial-strength database management system, including the principles of data type selection and indexing.	**given database design goals, identify correct techniques for implementation**	0.86	0.19	
15	c Build	90	IS Database and IS Implementation	to develop skill in application of database systems development and retrieval facilities needed to facilitate creation of information system applications	2.14	Understand the concept of database transaction and apply it appropriately to an application context.	**apply the knowledge of using a stored procedure to enhance the performance in a database environment**	0.32	0.34	

17	c Build	92	IS Database Application Implementation	to develop skill with application and physical implementation of database systems, using a programming environment	2.12	Use the data definition, data manipulation, and data control language components of SQL in the context of one widely used implementation language.	**recognize the advantages and disadvantages of implementation with stored procedures**	0.72	0.26	
18	c Build	92	IS Database Application Implementation	to develop skill with application and physical implementation of database systems, using a programming environment	2.12	Use the data definition, data manipulation, and data control language components of SQL in the context of one widely used implementation language.	**trace and debug SQL syntax**	0.28	0.21	
19	c Build	92	IS Database Application Implementation	to develop skill with application and physical implementation of database systems, using a programming environment	2.12	Use the data definition, data manipulation, and data control language components of SQL in the context of one widely used implementation language.	**recognize the correct formulation of a query**	0.87	0.29	
22	c Build	95	IS Database Conceptual/Logical Models	to show how to design a conceptual relational database model and logical data base model, convert the logical database designs to physical designs, develop the physical database, and generate test data	2.09	Understand the purpose and principles of normalizing a relational database structure.	**differentiate normal forms as part of database design**	0.53	0.34	

1.3.3	**Administration: security, safety, backup, repairs, replicating**	monitoring, safety -security, administration, replication, monitoring, repair, upgrades, backups, mirroring, security, privacy, legal standards, HIPAA; data administration, policies

24	b Anal yze	111	IS Requirement s and Database	to develop requirements and specifications for a database requiring multi-user information system	2.01	Understand the role of databases and database management systems in managing organizational data and information.	**recognize relevant factors involved in the purchasing decision of a major enterprise level DBMS package**	0.28	0.30	**0.54**
7	c Build	81	IS Database Applications Development	to develop application skills for implementing databases and applications by operating and testing these databases	2.17	Understand the key principles of data security and identify data security risk and violations in data management system design	**given a system need, such as access control to a database, identify the necessary information**	0.80	0.30	
1.3.6		**Data Quality: dimensions, assessment, improvement**				Data Accuracy, Believability, Relevancy, Resolution, Completeness, Consistency, Timeliness; Data definition quality characteristics, Data model / requirements quality characteristics; Data clean-up of legacy data, Mapping, transforming, cleansing legacy data; Data defect prevention; referential integrity; Data quality employee motivation, Information quality maturity assessment, gap analysis				
9	b Anal yze	88	IS Data Modeling	to develop skill with data modeling which describe databases	2.18	Understand the core concepts of data quality and their application in an organizational context.	**recognize the implication of a cascade delete**	0.58	0.25	**0.58**
							Average % Correct ---->	**0.53**		

Note: The table is organized by sub-skills. Each row of the table shows the item number, the mapping of the item to the ABET program outcomes, IS 2002 Learning Unit (LU) number, LU Title and LU Goal statement followed by and IS 2010 learning outcome from IS2010.2 course. The item objective (in bold) was mapped to the IS 2010 learning outcome. The last three fields show the percent correct, and the point bi-serial correlation coefficient, and the average of percent correct for each sub-skill. Test items (not shown) were derived by first developing the Item Objectives (while studying the sub-skill and LU data) and then the Test Item was written.