Exploiting the Complimentary Strengths of Computer Science and Information Systems in the Development and Implementation of Computing Curricula: A Case of an African University

Sunday O. Ojo[1]
Department of Computer Science, University of Botswana
Private Bag UB 00704, Gaborone, Botswana

Abstract

This paper presents how the complementary strengths of Computer Science and Information Systems are exploited in the development and implementation of computing curricula in an African university. This is an attempt at minimising the gap between the products of an earlier Computer Science program and market demand. The computing curricula evolved through an integrative and collaborative process. The evolutionary process started with the infusion of IS ideas into an existing Computer Science curriculum and ends with multi-stream collaborative Computing curricula. Key elements of the approach are presented. Some lessons drawn from our experience are presented.

Keywords: computing curricula, information systems, complementary strengths

1. INTRODUCTION

In the early conception of computing curricula it is common that the term 'Computing' is used to encompass the labels 'Computer Science' [and] 'Computer Science and Engineering' but specifically excludes programs in other computing disciplines such as information systems (CCCS 2001). This conception of computing as a discipline fails to address critical issues of computing in practice (Wilson, Greenleaf & Trenary 1989), hence, the various quests to create other computing disciplines, notably, Information Systems and Software Engineering (Denning et al 1989).

The discipline of computing continues to grow. All aspects of the computing field are facing rapid, continuous change. As a result, university level computing curricula need frequent updating to remain effective (CCCS 2002). A major change in the discipline over the past decade is the enormous broadening of the scope of the field (CCCS 2001; IS 2002). In the CCCS(2001) report, it is observed that the expansion of the field has a significant impact on the broad domain of the field of computing. In advocating for a broader view of the computing discipline, the report posits that the disciplines of Computer Science (CS), Computer Engineering (CE), Software Engineering (SE) and Information Systems (IS) taken together, represent much of academia's coverage of the discipline of computing.

Over the years, efforts are being made to

[1] ojoso@mopipi.ub.bw

develop these areas of computing into disciplines in their own right. These include IS 1997, ISCC 1999 (Lidtke D et al 1999), MSIS 2000, ACM/IEEE-CS 1999. Most recent effort is the Computing Curricula 2001 with component curricula including Computer Science (CCCS 2001), Software Engineering (CCSE 2003), Computer Engineering (CCCE 2003) and Information Systems (IS 2002). However, in practice, the split in the discipline of computing encourages the emerging disciplines to diverge while defining their own territories. A consequence of this is that important aspects of computing will be omitted from study in academic curricula (Wilson, Greenleaf & Trenary 1989).

Notwithstanding their differences, there is a close relationship between these academic fields of computing. This is with regards to their complimentary strengths, which needs be exploited in preparing computing graduates (IS 2002). For example, IS is typically strong in preparing students for the organisational environment. However, it has the challenge to maintain adequate depth of instruction in the technical aspects of computing. On the other hand, a CS program is typically strong in teaching technology and algorithmic processes, but organisational functions and systems may not be an area of emphasis. In IS 2002 report, it is observed that this high level perspective of complementary strengths suggests the existence of opportunities for courses designed for any of these computing fields meeting the needs of the others. Therefore, in designing a curriculum in any of these fields of computing, it is important that their complementary strengths be exploited rather than emphasizing their differences.

This is the thrust of the work being reported in this paper. More specifically, the complementary strengths of Computer Science and Information Systems fields of computing were exploited in the development and implementation of the Computing curricula in an African University (UB-BIS 2001;UBCS 2001;UB-GEC-2 2001;UB-RAGR 2000). It is an integrative and collaborative CS-IS curricula designed to address the problem of the gap between market demand and the products of an earlier CS curriculum. The curriculum development approach is evolutionary. It started with the infusion of IS ideas into a CS curriculum through to evolving a multi-stream integrative and collaborative CS-IS curriculum. It is integrative in that a number of common courses are shared to exploit the complimentary strengths of CS and IS. The collaborative aspect involves Departments of Computer Science, Library and Information Studies, and Accounting & Finance, which are located in the Faculties of Science, Humanities and Business, respectively.

The objective of this paper is to present salient features of the Computing Curricula to demonstrate how the complementary strengths of CS and IS are exploited in the development and implementation of the curricula. Key lessons learnt and their implications for computing curricula development are discussed.

2. THE CONTEXT

The National Context

The University of Botswana (UB) is at present the only university in Botswana. As such, the country depends mostly on the university to meet its computing human resources needs. Interaction with the various stakeholders both in the public and private sectors of the economy has revealed a need for strategy realignment in computing curricula provision in the country.

According to the report of a recent study (Kansichi 2001), there is a shortage of computing professionals in Botswana. This has culminated in the country depending mostly on expatriate computing personnel. The study further reveals that, 90% of computing project managers, enterprise systems implementers, and business analysts/consultants are expatriates, technical designers are 99% expatriate, while systems integrators are 60% expatriate. According to Gaborone (2001), these imported computing personnel as being artificially overprized. Given the global acute shortage of computing skills, Botswana cannot continue to depend on expatriate computing personnel. A deliberate program that will increase the critical mass of locally produced well-equipped computing personnel thus becomes an imperative.

The Botswana Government is the largest employer of computing professionals in the country. Over the years, the Government through its Government Computer Bureau has been discussing with the Department of Computer Science at the University, the need for a proper blending of computing theory and practice in the program offerings. The Advisory Board for the Department with membership drawn from the public and private sector of the computing industry, also concur with this need. The Directorate of Public Services Management, charged with the responsibility of recruiting public services personnel, recently underscored this need in its proposal to government (Ridderof 1999). In the proposal, two parallel cadres of computing personnel need within the government Ministries and departments are identified. That is, the Technical computing cadre and the Information Systems cadre. This is to respectively meet the technical-oriented and the business-oriented computing human resources needs in the public sector

Ramaribana (2001), reports that there has been an increased demand for a balanced technical cum business-oriented approach to the development of computing personnel. This is more so that the need for computerized systems to support business processes in government has increased during the first few years of the current National Development Plan period (i.e. 1997 to 2003).

The Government has responded by decentralizing the funding of computing projects and creating Computer Units in ministries and departments in order to enable them greater autonomy in the deployment of computerized systems to support their business. These deployment efforts are being hampered by an acute shortage of adequately skilled computing staff. The country needs to train many more computing personnel who are well equipped with technical and business-oriented skills, if the civil service is to remain efficient in comparison to similar services throughout the world, Ramaribana (2001) opines.

In addressing the needs, the government has been sending students and public officers abroad for a degree in IS. This has been a very costly venture for the government. There is therefore the challenge to the University to assist the government in providing academic programs targeted towards producing the required caliber of computing personnel at reduced cost.

It is important to note that out of over two hundred Computer Science graduates produced since the inception of the department of Computer Science at the University of Botswana, about 90 % have ended up in the industry, while less than 10% returned to taking up academic career. This is a reality that has to be borne in mind in the development and implementation of computing curricula in the country.

Generally, the country, like most other developing countries, is faced with operational, contextual and strategic constraints in effective exploitation of the computing technology for social progress and economic development. There is a need for computing curricula that enable the production of a calibre of computing personnel who are well equipped with operational, contextual, and strategic capabilities required for innovative and adaptive deployment of the technology in the country.

According to Ojo (1993), operational capability exists where the products of computing curricula are well equipped with necessary technical know-how and skills required for domestic IS development, maintenance and sustenance. Contextual capability is that which enables the products of a computing curriculum to have a clear understanding of the social, economic, cultural and organisational conditions of the local environments with respect to the utilisation of IS resources as well as the socio-cultural, organisational and economic biases of the IS technologies being adopted. It is also the ability to recognise these conditions as the critical issues in determining IS requirements and potential utility. Strategic capability deals with a combination of technical, managerial and socio-organisational know-how and skills required for innovative and adaptive use of IS technologies within the local context of its utilisation. It is also having competence in (a) recognising what can be utilised directly and what might be adapted for utilisation; (b) restructuring the context of utilisation to provide a more conducive environment for the technology if need be (Ojo 1993)

Given the foregoing, there is a need for computing curricula that enables the production of computing graduates who are adequately equipped with necessary 'hard' and 'soft' knowledge and skills for organizational information systems development and deployment.

The Institutional Context

The University is structured into academic units made up of Faculties of Agriculture, Business, Education, Engineering & Technology, Humanities, Science and Social Sciences. Each faculty is further divided into academic departments built around subject areas. The department of Computer Science that provides the anchor for the computing curricula in the University is situated in the Faculty of Science. The earlier Subject system adopted seems to encourage each department to be more or less self-contained with regards to the course modules required for an academic program. In the earlier CS program, for instance, all the required courses for the award of a degree in CS are contained within the CS curriculum. Hence, any non-traditional CS ideas such as those found in IS, can come into the syllabus only as course modules or integrated into existing course modules as topics (Ojo 1997).

Recently the University changed to a Semester Course system (UB-RAGR 2000) wherein students earn credits for courses they take based on level of performance. Each academic program offering requires students to take a number of credits of Core, Optional, Elective and General Education courses. For a given academic program in a department, a course in any of these categories can come from within or outside the department. An undergraduate academic program can be designated as a single major, combined major/major, combined major/minor, or combined multi-disciplinary degree program. The latter category enables a program to be developed out of more than two subject areas. This provides a good basis for exploiting the complementary strengths of related subject areas in curriculum development.

Each academic program with some exceptions, is designed to run for eight semesters. The requirement for program completion is to obtain a minimum of 120 credits distributed into 15 credits per semester. At least two thirds of these must be from core and optional courses while up to one-third from elective and general education courses. A typical 8-semester degree program is divided into four levels of study, which are levels 100, 200, 300, and 400. Each level is in turn divided into two semesters.

One significant shift in the University's approach to academic programming is the trend towards placing more emphasis on cross-disciplinary curriculum development. This discourages departmental silo mentality in curriculum development, which characterised the earlier Subject system, and so enables departments to collaborate in curriculum development. For instance, development of the Computing curricula being reported in this paper, involved collaboration between the Computer Science department and some other departments in the Faculties of Business and Humanities.

Further, in the present Semester system, flexibility in entry requirements is made possible so that students from varied disciplinary background can gain entry into a program. For example, candidates for the CS and IS programs in the department of Computer Science can come from students who completed first year Humanities, Business, Science, or Social Sciences, with the proviso that they include some Mathematics courses in their 1st year course selection.

Further more, in addition to the various discipline-based academic programs, the University makes provision for General Education program, which is divided, into seven thematic areas. The General Education courses provide students with breadth of knowledge to complement their knowledge in their respective subject specializations. This provides an avenue for meeting some of the knowledge and skills requirements for the computing students, which cannot be accommodated in the core curricula.

3. THE APPROACH

Complimentary Strengths of CS and IS

There is a close relationship between the various academic fields of computing with regards to their complementary strengths (IS 2002). For example, IS and CS have such relationship. The context for IS is an organisation and its systems. The IS academic field is typically strong in preparing students for the organisational environment. This strength is more when the IS program is within or closely tied to organisational or business studies. In contrast, the context of CS is algorithmic processes for information processing and associated technical and technology issues. These respective strengths of IS and CS are complimentary and can be exploited in a curriculum designed for preparing computing graduates. The exploitation of these complementary strengths provides opportunities for sharing courses be it core or optional courses (IS 2002).

The focus in this paper is on the exploitation of the complementary strengths of the CS and IS in computing curricula development. This involves a three-prong approach: a) Exploiting IS strength in a CS curriculum; b) Exploiting CS strength in IS curriculum, and c) Exploiting the strength of the General Education program in the Computing curricula. Before going into details of each of these and to provide reference points for the discussion it is important to show the general structure of the Computing curricula being discussed in this paper.

3.2 The Computing Curricula Structure

The Computing curricula comprise a BSc Computer Science and a three-streamed Bachelors of Information Systems (BIS) curriculum. The BIS curricula is streamed into BIS (Computer Information Systems), BIS(Information Management) and BIS (Business Information Systems), given the acronyms BIS (CIS), BIS (IM) and BIS (BIS), respectively.

The three-streamed BIS curriculum involves cross-departmental collaboration. The collaboration involves the Department of Computer Science in the Faculty of Science, the Department of Library & Information Studies in the Faculty of Humanities , and the Department of Accounting & Finance in the Faculty of Business. For implementation purposes, the three departments are responsible for the BIS (CIS), BIS (IM), and BIS (BIS) programs, respectively.

Each of the program streams gets intakes into year 2 from students who completed the 1st Year from the different faculties of the University. Candidates into the 2nd Year could also come from graduates of the Diploma in Computer Studies of the university. However, while intakes into the BSc (CS) and BIS (CIS) can come from students who complete 1st year from the Faculties of Science, Humanities, Business and Social Sciences, those for the IS programs in the Departments of Library and Information Studies (i.e. BIS(IM)), and the Department of Accounting and Finance (i.e. BIS(BIS)), are limited to completers of 1st year Humanities and Business, respectively. In addition to the core and optional courses in the core curricula, students take General Education courses and elective courses, which are extra curricula. The BIS program streams are integrated with the CS curriculum through sharing of some common courses.

Exploiting IS Strength in a CS Curriculum

Earlier, in Ojo (1997), we posit an approach to computing curricula development that balances computability and usability concerns of computing, through infusion of IS ideas into a CS curriculum. This involves a) infusing IS course modules into the CS syllabi and b) Infusing IS topics into existing CS course modules. Tables 1a and 1b summarise the earlier attempt in this direction. Table 1a shows some of the non-traditional CS courses included in the CS curriculum through this approach. Incorporation of IS topics into some existing CS course modules involved topics aimed at equipping students with ‘soft’ knowledge and skills, to complement the traditional ‘hard’ knowledge and skills in such course modules. Table 1b shows some of the courses and the nature of IS topics incorporated with the teaching of the courses from socio-organisational cum technical perspectives.

Table 1a: Non-Traditional CS Courses Infused into CS Curriculum

CS371 Information Systems & Organisations
CS291 Communication Skills for Computer Scientists
CS374 Management
CS471 Social Issues of Information Technology
CS473 Economics of Information Technology
CS 346 Systems Analysis and Design

Table 1b: Some of the CS Courses with IS Topics incorporated

CS Course Modules	Nature of IS Topics Incorporated
CS481: Database Systems	Incorporating socio-organisation and managerial issues; Information resources management, Use of Business-driven Information Engineering Methodology; Taught from socio-organisational and technical perspective
CS444: Software Engineering	Incorporating Professional and ethical responsibility issues, Software Safety and social vulnerability; Project management; Systems resources management.

However, according to Ojo (2002), this approach turned out to be a palliative measure that falls short of adequately addressing the target needs. Given this, in further exploitation of IS strength in CS curriculum, the CS curriculum was revised to infuse more IS courses. Also, the fourth (final) year courses were streamed into four areas of concentration, which includes IS stream. These streams are: Software Engineering, Knowledge-Based Systems, Scientific Computing, and Information Systems. This is to enable the students to be more focused in their selection of courses, with their future career interest in mind. As our concern in this paper is to show how we strengthen CS curriculum through incorporating IS courses, we shall only focus on the IS component of these streams of concentration.

Tables 2a and 2b summarise the key elements of the resulting changes from the revision of the CS curriculum. Table 2a shows the list of IS courses, which are not traditionally CS core, incorporated into the CS curriculum. Table 2b shows IS courses included in the IS concentration stream in the fourth (final) year of the CS programme with associated credit weighting. Out of the total of 34 course credits in the IS concentration stream, 22 are compulsory and must include the 4 credits from CSI461 (Networking course) and 6 credits from the two project courses (CSI403 and CSI405). Also, after selecting all core courses required for the CS program, all the IS courses infused into the CS curriculum are available either as core or optional courses to CS students.

Table 2a: IS Courses in CS Curriculum

CSI292 Information Systems Fundamentals
CSI315 Web Technology and Applications
CSI371 IS Resources Management
CSI342 Systems Analysis and Design
CSI373 Economics of Information Technology
BIS302 Decision Support Systems I
CSI314 Decision Support Systems II
CSI472 Social & Professional Issues of Computing
CSI434 Knowledge Management Systems
CSI482 Information Systems Engineering
CSI484 National Information Systems Infrastructure

Table 2b: Courses in IS Concentration stream of CS Curriculum	**Credit**
CSI461 Computer Comm. Networks Management	4
CSI411 Operating Systems	3
CSI434 Knowledge Management	3
CSI481 Database Systems	3
CSI482 Information Systems	3
CSI484 National Information Systems Infrastructure	3
CSI471 Object Oriented Systems	3
CSI472 Social & Professional Issues of Computing	3
CSI416 Topics in Information	3
CSI403 Project I	2
CSI405 Project 2	4

Exploiting CS Strength in IS curricula

In exploiting the strength of CS in IS curricula, the IS programs have some common courses drawn from CS courses. These are meant to provide an avenue for balancing the business and organisational context of IS with the technical and technology context of CS. The IS curricula streams have some common core and optional courses with the CS curriculum.

Table 3 shows the CS courses that are shared with the IS curricula streams. It also shows whether the course is Core (C) or Optional (O) for the IS program. Optional (O) means the course is listed as one of those which students for the IS stream can take after satisfying the credits requirement for core courses. The CS courses shared with IS curriculum streams are essentially technology and methodology courses which are meant to address technical skills needs for IS graduates.

Table 3: CS Courses Shared with IS Curricula Streams

	CIS	IM	BIS
CSI231 Discrete Mathematics I	C		
CSI232 Discrete Mathematics II	C		
CSI241 Structured Programming (in Java)	C	C	C
CSI252 Operating Systems Concepts	C	C	C
CSI261 Machine Organisation	C	O	O
CSI272Computer Comm. Network Fundamentals	C	C	C
CSI315 Web technology and Applications	C	C	C
CSI341 Introduction to Software Engineering	C	C	
CSI342 Systems Analysis and Design	C	C	
CSI392 Human Computer Interaction	C	O	O
CSI372 Expert Systems	O		O
CSI414 Information Interfaces and Presentation	O	O	
CSI461 Computer Communications Network Management	C	C	C
CSI462 Distributed Systems	O		O
CSI481 Database Systems	C	C	C
CSI471 Object-Oriented Systems Development	O	C	O

Exploiting the Strength of General Education Program in Computing Curricula

The University makes provisions for a General Education program, which is divided, into seven thematic areas as shown in Table 4.

Table 4: General Education Thematic Areas.

Area 1	Communication and Study Skills
Area 2	Computer and Information Skills
Area 3	Modes of Inquiry and Critical Thinking
Area 4	Physical Education, Health and Wellness
Area 5	Science and Technology
Area 6	World Civilization
Area 7	World Economy and Business Skills

For a typical 4-year degree program, a student must earn up to 20 credits of General Education courses, out of the at least 120 credits required for graduation. Some courses in Areas 1, 2 and 3 are compulsory.

It provides an avenue for meeting some of the knowledge and skills requirements for the computing students, which cannot be accommodated in the core curricula. For example, GE Area 1 provides for skills requirements in Personal Productivity with IS Technology (as recommended in IS 2002), GE Area 2 provides for Communication Skills, GE Area 3 provides Analytical and critical thinking skills, GE Area 7 provides business skills. According to Gorgone et al (2002), all of these are required by computing graduates. With appropriate guidance, students select courses from the General Education program in such a way that they compliment their course selection in their core subject area with regards to their skills requirements not catered for in the latter. Thus, the strengths of the General Education program areas are exploited to compliment those of the CS and IS.

4. LESSONS LEARNT

Our experience in developing and implementing the Computing Curricula being discussed in this paper brought out the following lessons:

a) CS and IS are two sides of the same coin with complementary strengths that can be exploited in Computing curricula development and implementation. This is necessary to minimise the gap between graduates and market demands. The rapid changes in the technology of computing and the needs in the environment of its use, demand paradigmatic flexibility that enables this exploitation. This requires appropriate blending of "hard" and "soft" sciences. Holding a doctrinaire belief in the purity of "hard" science of computing to the extent that it should not be 'adulterated' with the 'soft' science of the socio-cultural and organisational context of applications of computing, would not help in evolving computing curricula which are appropriate for a context.

b) Cross-disciplinary collaboration facilitates the exploitation of the complimentary strengths of the fields. In our case, the collaboration between the Department of CS and the other two departments in the faculties of Humanities and Business facilitated the exploitation of the IS strengths in CS curriculum. Such collaborative efforts should not be limited to curricula development; rather, it should extend to cross-disciplinary research that enables evolution of paradigm of common understanding among the partners.

c) Exploitation of complimentary strengths of IS and CS in Computing curriculum development enables curricula rationalisation. With curricula rationalisation, unnecessary cross-curricula course module duplication is avoided. This results in implementation cost saving.

d) The process of cross-departmental collaboration in curriculum development could be a path strewn with socio-political considerations, rather than pure rationality. In particular, our experience in the emergence of the 3-streamed Bachelors of Information Systems is worth noting. It is a product of almost acrimonious turf war with political and interest-balancing considerations. Notwithstanding, the fact that such an academic program emerges that cuts across traditional departmental/Faculty boundaries is seen as a 'feat' in cross-disciplinary collaboration. In fact, it is at present considered as a reference model in the University when encouraging other departments to collaborate in academic program development.

e) The level of resources availability imposes a constraint on the growth of computing curricula in responding to the changing needs. The Department of Computer Science depends on the collaborating department to offer the business-oriented courses to its students. However, those departments turned out unable to offer this service due to resource constraints. CS department services the IS programs in other departments with the technical courses. This is over stretching the computing and human resources of the department. Due to this situation, not all the optional courses are offered each academic year. According to Tatnall (1997) for growth of curricula to occur there has to be an educational infrastructure – physical resources, central support facilities, teaching laboratories and human resources. He further reports that a study of the curriculum history of business computing reveals a complex interrelationship between computer technology, educational needs, institutional facilities, and the on-going development of curricula. This is a reality that needs be borne in mind.

f) Evolving an appropriate strategy for continuous interaction with the industry stakeholders is essential in developing Computing curricula that are relevant and hence minimise the gap between the graduates and the market demands. The involvement of a program Advisory Board which includes representatives of the industry stakeholders, has been of tremendous in the evolutionary development of the curricula towards meeting the target needs.

g) The level of flexibility inbuilt into the institutional academic environment for the development and implementation of curricula has a bearing on the extent to which the CS and IS strengths can be exploited. Where the academic units are in silos with more emphasis on subject areas and departments, rather than programs, evolving Computing curricula that is responsive to the changing needs, through exploitation of the complimentary strengths of CS and IS, is made all the more difficult if not impossible. The flexibility which the University built into the Semester course system that replaced the subject system greatly facilitated the responsiveness of our Computing curricula to the needs in the national context.

5. CONCLUSIONS

In this paper, we have presented how we exploited the complementary strengths of CS and IS in the development and implementation of Computing curricula in an African university. We have also drawn some lessons learnt in the process.

The socio-economic context of the computing adoption in Africa raises problems of a new sort that demands a re-alignment of strategy in our approach to Computing curricula development and implementation. It demands not just simply applying ready-made curricula models, rather finding new or adapted models and approach in order to appropriately address the challenges presented in the productive deployment of the science and technology of computing in Africa.

As demonstrated in the case presented, there is a need for an approach to university computing curricula that adequately exploits the complementary strengths of the apparently competing fields of computing. This definitely requires an appropriate integration of 'hard science' and 'soft science' paradigms into a single whole, to evolve a more flexible and comprehensive paradigmatic framework that gives equal treatment to technical and socio-organisational aspects of computing. This in turn requires cross-departmental collaboration. Evolving such a framework does pose an enormous challenge in the light of the lessons drawn from the case considered in this paper. It is hoped that this paper provides some direction in this respect.

6. REFERENCES

ACM/IEEE-CS 1999: Software Engineering Coordinating Committee. Guide to the Software Engineering Body of Knowledge (SWEBOK). Stone Man Version 0.5. Joint project of the ACM/IEEE-CS. **http://www.swebok.org/stoneman/version05/**

CCCE 2003, Computing Curricula: Computer Engineering. ACM/IEEE-CS. http://www.eng.auburn.edu/ece/CCCE/

CCCS 2001, Computing Curricula 2001 Volume II, Computer Science. ACM/IEEE-CS. http://www.computer.org/education/cc2001/index.htm

CCSE 2003, Computing Curricula: Software Engineering Curriculum. ACM/IEEE-CS. http://www.eng.auburn.edu/ece/CCSE/

Denning Peter J 1999, Our seed corn is growing in the commons. Information Impacts Magazine, March. **http://www.cisp.org/imp/march_99/denning/03_99denning.htm**

Gaborone O. M, 2000, Opening Speech, International Conference on Software Practice & Experience – Issues and Challenges for the 21st Century, Gabion, June.

IS 2002, Model curriculum and guidelines for undergraduate degree programs in information systems. ACM/AIS/AITP http://192.245.222.212:8009/IS2002Doc/Main_Frame.htm

IS 1997, Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems ACM/AIS/AITP http://www.is2000.org/rev/review1.html

Kansichi R, 2001, "Proposal for ICL Center of Excellence in IT Skills." A presentation to the Computer Science Department, University of Botswana.

Lidtke K. Doris, Stokes Gordon E., Haines Jimmie, and Mulder Michael C. 1999, ISCC '99 - An information systems-centric curriculum '99, July. **http://www.iscc.unomaha.edu**.

MSIS 2000, Model curriculum and guidelines for graduate degree programs in information systems. ACM/AIS. **http://cis.bentley.edu/ISA/pages/documents/msis2000jan00.pdf**

Ojo, Sunday O., 2002, "On the Conception of Computing Curricula in an African University." In Proceedings of ISECON 2002. Colton D, Payne M.J & Bhatnagar N (Eds) Vol. 19, CD-ROM 2002/351c.

Ojo, Sunday O., 1997, "Integrating Information Systems Ideas into the Computer Science Curriculum – A Case of Two African Universities." In Proceedings of ISECON'97, Orlando, Florida, October.

Ojo, Sunday O., 1993, "Appropriate Information Technology Human Resources Development in Developing Countries". In Ojo, Parsons & Age (eds.), Information Technology Utilisation in Developing Countries: A Need for Cooperation and Coordination, Gaborone, 1993. Printing & Publishing Co. Pp.32-45.

Ramaribana, A. 2001, A Correspondence from Government Computer Bureau on Computing Program in the University of Botswana.

Ridderof A. 1999, A Proposal for 2-Cadre of IT Human Resources in Public Services. Internal Document, Directorate of Public Services Management.

Tatnall A., 1997, "The Evolution and Re-Invention of Information Systems Curricula in Australia." In Albin M(Ed), Proceedings of ISECON '97, Orlando, Florida, October p.131-136.

UB-BIS 2001: Bachelors of Information Systems Program, Internal Document, and University of Botswana.

UB-CS 2001: Revised Undergraduate Computer Science Program, Internal Document, and University of Botswana.

UB-GEC-2, 2001, GEC Area 2: Computer & Information Skills Program, Internal Document, and University of Botswana.

UB-RAGR, 2000, Revised Academic General Regulations Internal Document, University of Botswana.

Wilson J, Greenleaf Newcomb, and Trenary R. 1989, "Algorithms and Software: Integrating Theory and Practice in Undergraduate Computer Science Curriculum."