Defining the Content of the Undergraduate Systems Analysis and Design Course as Measured by a Survey of Instructors

Timothy J. Burns
tburns1@ramapo.edu
Anisfield School of Business,
Ramapo College of New Jersey
Mahwah, New Jersey 07430, USA

Abstract

There are many factors that make the undergraduate systems analysis and design course somewhat enigmatic in its purpose, and therefore equivocal in its delivery. The purpose of this research is to learn, specifically, what instructors are teaching in their systems analysis and design courses. This paper reports the results of a survey and follow up interviews that were administered to instructors of the course located in colleges and universities around the world. Results indicate that there is a fair amount of consensus among instructors as to the course content.

Keywords: IS research toward educators, pedagogy, IS undergraduate curriculum, Teaching Systems Analysis and Design

1. INTRODUCTION

In their 2006 article introducing a special issue on teaching systems analysis and design (SA&D), Harris et al. made the statement that "If you were to assemble 50 IS professors into one room you would likely get 50 different opinions on how best to teach SA&D" (Harris, Lang, Oates, & Siau 2006). The purpose of this research is to test that statement, but on a slightly larger scale. Over 1500 Information Systems (IS) professors were asked their opinion on how best to teach SA&D. Although only 172 chose to answer, it is felt that this is a large enough sample to get a good idea as to the heterogeneity (and conversely the homogeneity) of the course content of undergraduate SA&D courses across colleges and universities around the world. In essence, we want to know what instructors are teaching in their SA&D courses.

This article is an attempt to go further in depth and breadth than a similar article written in 2005. In that article, course materials were looked at to determine what textbooks and system development methodologies were being covered in system development courses (Burns & Klashner 2005). This article goes further in breadth in that it looks at all materials covered in the courses. It goes further in depth in that it delves deeper into instructor backgrounds, teaching methods, topics covered, etc.

2. BACKGROUND

As instructors of an SA&D course, we are constantly asking ourselves how best to conduct the course. How can we deliver the best product to prepare and educate students for a career in information systems development? Gorgone et al. define the purpose of an IS curriculum to be to "produce graduates equipped to function in entry level information systems positions with a strong basis for continued career growth" (Gorgone, Davis, Valacich, Topi, Feinstein, & Longenecker 2002). Gorgone et al. also provide a fairly comprehensive list of what

should be covered by the IS curriculum and what skills and perspectives IS graduates should have (Gorgone et al., 2002). They say that IS graduates should have strong analytical, critical thinking, interpersonal, and team skills. Furthermore, they should have broad business and real world perspectives. Finally, they should have strong ethical principles and must be able to design and implement IT solutions that enhance organizational performance (Gorgone et al., 2002).

The SA&D course is an essential part of the IS curriculum (Harris et al., 2006). According to Gorgone et al., it should provide experience in determining system requirements and developing a logical design. Furthermore, students should work in teams to analyze problems and design and implement information systems (Gorgone et al., 2002).

These seem like clear cut goals, but there are many factors that make the SA&D course somewhat enigmatic in its purpose, and therefore equivocal in its delivery. A literature review of what SA&D instructors are researching, writing, and teaching in their SA&D courses shows that there are many issues that raise debate. There appear to be two predicating issues. First, is the issue of the volume of material that could be potentially covered in the course. SA&D encompasses a large and ever expanding field of material. How does the instructor choose what is important to cover? This raises a paradox between the volume of the material available to deliver and the amount of material that can be successfully delivered to the students. Many students get bored or overwhelmed with the material. Where on the spectrum between breadth and depth should the course content land?

The second predicating issue is the applied nature of the IS discipline and the ever changing dynamic of the field. Should we be teaching fundamental, tried and true concepts, methods, tools, and techniques, (many of which are 30 or more years old)? Or, should we be teaching the latest and most current of the concepts, methods, tools, and techniques being introduced in industry or by academics, even if they have not yet been completely vetted?

There are many other areas of contention when discussing the content of SA&D courses. For instance, currently there are two general overarching approaches to systems development; the traditional (or structured approach) and the object-oriented approach. Many instructors feel that one should be covered over the other (Rob 2006) or that perhaps the traditional approach is outdated. However, there is research that shows that teaching IS students the traditional approach is still a viable vehicle for SA&D instruction (van Vliet & Pietron 2006).

Another area of contention surrounding the SA&D course is whether the content should be delivered in one course or two. Given the volume of material a two course model seems to makes sense. Many institutions have adopted this two course model. However, a two course paradigm may not fit in a curriculum where there are a limited number of IS specific course credits available. This is particularly true if the IS program is housed in a business program where many of the available credits are eaten up by core business courses.

One solution to fitting two SA&D courses into the curriculum is to make the second course part of the final capstone course in the program. Using this approach eases the burden of trying to deliver the large volume of course content, and it solves some other issues as well. In particular, it addresses the issue of where in the sequence of courses in an IS curriculum the SA&D course should lie. There are many who feel that even if the SA&D content is delivered as one course, it should be near the end of the IS course sequence. That is because the SA&D course draws upon the material presented in other IS core courses, such as database and network design, programming, project management, etc.

With two SA&D courses, one focused on basic theoretical foundations can be placed earlier in the course sequence, and one focused on the application of the knowledge gained can be placed near the end. The capstone course can be used to simulate a real world project where the students solve real business problems and work with real users.

This brings up yet another point of contention surrounding the SA&D course. Should the course include case studies and course projects that allow the students to apply their knowledge? Or should SA&D course content be delivered primarily through the traditional lecture? Studies have shown that traditional lecturing is by far the most common method

for delivering course content and yet, is one of the least effective in terms of how much of the material that the students retain (Griffiths & Oates 2003).

Introducing case studies and projects (simulated or real world) into the course has several advantages. Using case studies gives meaning and real world context to the material (Avison & Cole 2006). Other techniques such as the use of "assumption/implication" debates, where the instructor states an assumption and then asks the students the implications, can provide a greater depth of understanding to the students while at the same time making the course material less boring (Avison et al., 2006).

3. RESEARCH METHODOLOGY

This research was conducted using a "grounded theory" approach. Grounded theory was developed by the sociologists Barney Glaser and Anselm Strauss in the 1960's. In the grounded theory approach, conclusions are drawn and theories are produced by analyzing a body of data. In essence, the theories that are produced are "grounded" in the data (Glaser & Strauss 1967).

For this study, the process began by analyzing the current body of literature on teaching the SA&D course. This allowed the researchers to create a survey instrument that would be used to ask questions about the delivery of the SA&D course and the demographical background of the instructors and institutions that delivered those courses. The survey included mostly closed end questions (which are listed in the results section below) and a few open ended questions shown in the appendices at the end of the paper. The appendices show a summary of the responses to the open ended questions.

A list that contained approximately 1500 names of IS instructors was compiled and an email was then sent to every person on the list. The email explained the purpose of the study and provided a link that the subject could click on to complete the questionnaire. Approximately 172 people chose to participate in the study. Once the initial results had been tabulated, a follow up email was sent to all of the participants in order to gain a deeper understanding of their responses.

4. RESULTS

In this section, the data that was collected is summarized and presented as a series of tables. The survey questions are included to provide additional clarity.

Question 1: How do you determine what subjects and material to cover in your Systems Analysis & Design course? (Multiple Answers Allowed)

Table 1 How Instructors Choose their SA&D Course Content

Most often chosen combination:

Based on industry experience, feedback, or trends, the textbook, and academic literature

Based on industry experience, feedback, or trends	83%
Based on the textbook	70%
Based on academic literature	45%
Based on academic suggested course outline	24%
Mandated by college or department	12%
Other	13%

Question 2: What textbook(s) do you use in your course?

Table 2 Textbooks Used in SA&D Courses By Percentage of Respondents

Whitten & Bentley, "Systems Analysis and Design Methods"	18%
Dennis, Wixom, & Roth "Systems Analysis & Design"	12%
Shelly, Cashman, & Rosenblatt, "Systems Design & Analysis"	11%
Satzinger, Jackson, & Burd: "Systems Analysis and Design in a Changing World"	6%
Dennis, Wixom, & Tegarden: "Systems Analysis and Design with UML"	6%
Hofer, George, & Valcich: "Modern Systems Analysis and Design"	6%
Valacich, George, & Hoffer: "Essentials of Systems Analysis and Design"	6%

Own Material	5%
Whitten & Bentley: "Introduction to Systems Analysis and Design"	4%
Kendall & Kendall: "Systems Analysis and Design"	4%
George, Batra, Valacich, & Hoffer: "Object-Oriented Systems Analysis and Design"	4%
Shelly & Rosenblatt: "Systems Analysis & Design"	3%
Marakas: "Systems Analysis & Design: An Active Approach"	3%
DeWitz: "Systems Analysis and Design and the Transition to Objects"	2%
Harris, "Systems Analysis and Design for the Small Enterprise"	2%
Larman: "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development"	2%
Other	6%

Question 3: How did you determine what textbook(s) to use in your Systems Analysis & Design course? (Multiple Answers Allowed)

Table 3 How Respondents Determined What Textbook to Use

Based on what I feel the course should cover	73%
Based on industry experience, feedback, or trends	39%
Suggested by a colleague	19%
Based on an academic suggested textbook	12%
Mandated by college or department	4%
Use my own materials	3%
None of the above	2%
Authored the book	2%

Most often chosen combination:

Based on industry experience, feedback, or trends and on what I feel the course should cover

Question 4: How is your Systems Analysis & Design Course delivered?

Table 4 How SA&D Course is Delivered

Traditional classroom	78%
Hybrid (part classroom/part online)	11%
Online	7%
Some sections online and some in traditional classroom	2%
Other (learner centered approach, videos, etc)	2%

Question 5: What phases of the systems development life cycle are covered in your Systems Analysis & Design course?

(Multiple Answers Allowed)

Table 5 Phases Covered in SA&D Course

Initiation	85%
Planning	92%
Analysis	98%
Design	93%
Implementation	75%
Maintenance	52%
None of the above	0%
Other (testing, project management, non-traditional)	10%

Most often chosen combination:

Initiation, Planning, Analysis, Design, Implementation, and Maintenance

Question 6: What system development approaches do you cover in your Systems Analysis & Design Course?

Table 6 Approaches Covered in SA&D Course

Both traditional and object oriented	53%
Traditional	25%
Object Oriented	15%
Traditional, object oriented, and other (Agile, RAD, JAD, etc.)	5%
Other (Method Engineering, Short life cycle, prototyping)	2%

Question 7: What system development methodologies or models do you cover in your

Systems Analysis & Design Course? (Multiple Answers Allowed)

Table 7 Methodologies Covered in SA&D Course

Waterfall	80%
Boehm's Spiral	24%
Prototyping	81%
Object Oriented	66%
Rapid Application Development	75%
Extreme Programming	35%
Scrum	12%
None of the above	0%
Other	10%

Most often chosen combination:

Waterfall, Prototyping, Object Oriented, and Rapid Application Development

Question 8: What project feasibility measurement concepts and techniques do you cover in your Systems Analysis & Design Course? (Multiple Answers Allowed)

Table 8 Feasibility Concepts Covered

Economic Feasibility	84%
Technical Feasibility	89%
Organizational/Cultural Feasibility	71%
Resource Feasibility	56%
Scheduling Feasibility	63%
Cost/Benefit Analysis	77%
Return on Investment	62%
None of the above	7%
Other	5%

Most often chosen combination:

Economic Feasibility, Technical Feasibility, Organizational/Cultural Feasibility, Resource Feasibility, Scheduling Feasibility, Cost/Benefit Analysis, Return on Investment

Question 9: What project management tools/techniques do you cover in your Systems Analysis & Design Course? (Multiple Answers Allowed)

Table 9 Project Management Tools/Techniques Covered

Microsoft Project	44%
Work Breakdown Structures	30%
GANTT Charts	66%
PERT Charts	56%
Critical Path	51%
None of the above	17%
Other	7%

Most often chosen combination:

Microsoft Project, GANTT Charts, PERT Charts, Critical Path

Question 10: What information gathering techniques do you cover in your Systems Analysis & Design Course? (Multiple Answers Allowed)

Table 10 Information Gathering Techniques Covered

Interviews	94%
Questionnaires	84%
Observation	80%
Heuristic Analysis	17%
Protocol Analysis	17%
Document Review	77%
JAD	58%
None of the above	2%
Other	7%

Most often chosen combination:

Interviews, Questionnaires, Observation, Document Review, JAD

Question 11: What diagramming techniques do you cover in your Systems Analysis & Design Course? (Multiple Answers Allowed)

Table 11 Diagramming Techniques Covered in SA&D Course

E-R Diagrams	77%
Data Flow Diagrams	81%
Flowcharts	30%
Structure Charts	39%
Database Diagrams	34%
UML Class Diagrams	52%
UML Use Case Diagrams	54%
UML Activity Diagrams	34%
UML Communication/Collaboration Diagrams	23%
UML State Machine Diagrams	22%
Package Diagrams	9%
None of the above	1%
Other	6%

Most often chosen combination:

E-R Diagrams, Data Flow Diagrams

Question 12: What other system development concepts and techniques do you cover in your Systems Analysis & Design Course?

(Multiple Answers Allowed)

Table 12 Other System Development Concepts and Techniques Covered

Systems Development Life Cycle	91%
Interface Design	68%
Forms Design	55%
Database Design	58%
Network Design	21%
Buy vs. Build	63%
Object and Class Design	42%
Use Case Descriptions	62%
UML	39%
Modular Concepts (cohesion and coupling)	33%
People and Resistance Issues	54%
Scope Creep	58%
Pseudo code Techniques	20%
Structured English	28%
None of the above	1%
Other	7%

Most often chosen combination:

Systems Development Life Cycle, Interface Design, Forms Design, Database Design, Network Design, Buy vs. Build, Object and Class Design, Use Case Descriptions, UML Modular Concepts (cohesion and coupling), People and Resistance Issues, Scope Creep, Pseudo code Techniques

These next tables represent the answers given to a series of follow up questions that were administered to the survey respondents.

Question 13: Is your course delivered in one course or two**?**

Table 13 Number of Courses

One	76%
Two	24%

Question 14: Do you have a course project?

Table 14 Respondents With Course Project

Yes	96%
No	4%

Question 15: Do you use a real world or simulated project?

Table 15 Real Or Simulated Project

Real	58%
Simulated	42%

Question 16: If real world, how do you find the projects?

Table 16 How Projects Are Found

Instructor finds projects	40%
Students find projects	60%

Question 17: Do you split students into groups or do all students work on one project

Table 17 How Students Collaborate On Project?

Split into groups	88%
All work together	7%
Students work individually	5%

Question 18: Does the course project extend beyond the course and one semester

Table 18 Does Course Extend Beyond One Semester?

Yes	21%
No	79%

Question 19: In your SA&D course, do you use more lecture or hands-on activities?

Table 19 Lectures Or Hands On

Lecture	22%
Hands on	9%
About Equal	69%

Question 20: Do you feel that the purpose of a SA&D course should be to give students practical experience or theoretical foundation?

Table 20 Instructors Perception of the Purpose of the SA&D Course

Practical experience	4%
Theoretical Foundation	2%
Mostly Practical	31%
Mostly Theory	22%
Even Split	40%

5. CONCLUSION

The results of this research seem to indicate that, contrary to the opinion of Harris et al., for the most part there is a consensus on how best to teach the undergraduate systems analysis and design course. There are overwhelming majority answers to almost all of the survey questions.

However, there are some results that warrant additional discussion. First is the question as to what system development approaches are covered in the course. A majority of the respondents (58% total) cover both the traditional and object-oriented approaches. This lends credibility to the instructors' faith in the importance of teaching both approaches. Surprisingly, 25% of the respondents teach only the traditional approach and 15% teach only the object-oriented approach in their course. This is surprising given the popularity of both the traditional and object-oriented approaches in industry (Satzinger, Jackson, and Burd 2009).

Question two, which asks what textbook the instructors' are using in their SA&D course shows an area of heterogeneity. While Whitten & Bentley, "Systems Analysis and Design Methods" is the most often used book by respondents in this survey, it is clear that there are many popular books. It is interesting that five percent of the respondents chose to use no book and just use their own materials.

Another question that warrants some discussion is question four, which asks how the course is delivered. Although the online format has gained some traction, the overwhelming majority (78%) still deliver their SA&D course in a traditional classroom setting. So, depending on your viewpoint, that 78% number may be surprisingly high, surprisingly low, or just about right.

It appears that three quarters of the respondents deliver all of their course material in one class (although many of the respondents commented that they wished they had two). Almost all of the respondents (96%) have a course project although they were split as to whether to use a real world project or a simulated project (such as a case study).

When asked if they used more lectures or more hands on activities in their courses, the vast majority of respondents (69%) said that they used both about evenly. Given the statistics on the commonality of lecturing (Griffiths and Oates 2003), it was a bit surprising that only 22% said that they more often utilized a lecture oriented approach.

Perhaps the one item that demonstrates that there is some fragmentation of the instructors' beliefs about the systems analysis and design course is the last question that asks if the purpose of a SA&D course should be to give students practical experience or theoretical foundation. Although a slight majority felt that the focus of the course should be evenly split between both, there were a large number of respondents who felt that the course should be

exclusively oriented towards either practical experience or theoretical foundation.

This paper was intended to be an introductory seminal work. Future research will focus on two questions. First, does the demographic background of the instructor (in terms of industry experience, years teaching, etc.) affect the course content of the SA&D course? Second, and perhaps more importantly, is the content of the typical undergraduate SA&D course consistent with the skills, tools, and knowledge required in industry?

6. REFERENCES

Avison, D. and Cole, M. (2006), "Reflections on Teaching Information Systems Analysis and Design: From Then to Now!," *Journal of Information Systems Education* (17:3), pp. 253--256.

Burns, T., Klashner, R. (2005), "A Cross-Collegiate Analysis of Software Development Course Content", Proceedings of the 6th Conference on Information Technology Education**,** Newark, NJ, USA, pp. 333-337. 2005.

Glaser, B. G., & Strauss, A. L. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research.* New York: Aldine Publishing Company.

Gorgone, J., Davis, G., Valacich, J., Topi, H., Feinstein, D., & Longenecker, H., (2002), "IS 2002 : Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems," *Communications of the AIS*, Vol11, No 1.

Griffiths, G., Oates, B., (2003) "Lecture-free Teaching or Systems Analysis: An Action Research Study", *Informing Science*

Harris, A., Lang, M., Oates, B., & Siau, K. (2006) "Systems Analysis and Design: An Essential Part of IS Education". *Journal of Information Systems Education*, 17(4): 241-248. (Special Issue on "Systems Analysis and Design Education")

Rob, Mohammad A. (2006), "Dilemma between The Structured and Object-Oriented Approaches to Systems Analysis and Design", *Journal of Computer Information Systems*. Vol. 46, Issue 3, Spring 2006, pp. 32-41.

van Vliet, P. J. A. and Pietron L. R. (2006), "Information Systems Development Education in the Real World - A Project Methodology and Assessment", *Journal of Information Systems Education*, 17, 3, 285--293.

Appendix I

Respondents' answers to the question: "What other concepts, methods, models, approaches, tools, and techniques do you cover that have not been mentioned in this survey?"

Accounting for advanced analytic applications and their impact on data warehousing & Mart design; dimensional modeling and conversion of ERD to dimensional models
ARIS enterprise model
Case based approach
Case tool: visible analyst
Critical requirements analysis, requirements traceability, presentation skills, documentation
Crud matrix, application architecture design, strategic information planning, Zachman framework
Data management (meta data, data analysis, data governance and stewardship)
Data normalization, unit testing, system testing, usability testing, acceptance testing
Dialog diagrams
Each project is different and one must design an approach to use to complete a project
Error, Fault, Failure and how to identify and classify IS problems. Report Design, Structured Walkthroughs
Ethical analysis
Higher abstraction of SAD: meta-modeling and method engineering
I focus on Requirements analysis using a use case approach. I cover CRC cards as a method to design system architecture - what classes are needed and what are their responsibilities.
Logical process modeling; logic modeling
Management issues
Management of Change, ANT, Socio-technical, Design of experiments
MS Visio
Naive inquiry, action methods
Participative approach and even EUC
Physical architecture design, systems controls
Process modeling
Project monitoring using earned value measurement, service oriented architecture (but these are covered in different modules of our programme)
Reducing risk by downloading, testing, and perhaps modifying open source software.
RUP process, Analysis patterns, Design Patterns (Larman, GOF, Buchmann), Arquitectural issues (horizontal/vertical partitioning, architectural patterns)
Selling projects, tradeoff triangle, team roles and conflicts
Soft skills - written and oral communication, team skills, presentation skills
Soft systems methodology (checkland)
Systems analysis is a people sport--we spend time on Maxwell's leadership, on the works of Stephen Covey, on team development skills, meetings, and on organizational development and reengineering concepts.
Systems thinking, Added value
Team-based, real-world projects with local clients
Teamwork and experiential learning, group issues, experts not wanting to make theselves obsolete
Testing, support and maintenance, usability, user centered design techniques, navigation diagrams, training, change management
The real world, Office Space
The students are required to develop a sample system analysis & design document based on instructor developed material
Vendor scorecards, estimating techniques, creating interview outlines
Writing requirements, Planning Analysis, Analyst skills and role, Stakeholder analysis, Enterprise Analysis, Estimation, CHAOS study, Quality Assurance

Appendix II

Respondents' answers to the question: "How do you balance the challenge of delivering the volume of material required for a typical SA&D course and keeping students from being overwhelmed/bored?"

It is a challenge since the concepts are abstract and student experience is minimal.
Try to make them work in teams, discuss among them, talk to people about real projects
Make the project the focus of the course, eliminate duplicate assignments, not tests
Best balance is to make sure they get some time in the lab to work on their projects; the hands-on components.
Make class interesting; give exams that require demonstration of learning how to apply concepts.
2 semesters help
They really get into their projects
After 25 years I think I balance it well. Hands on activities prevent boredom.
Keeping the students engaged / involved with discussion and hands on activities helps.
I decided to go with a textbook that covered the SDLC but with only 10 chapters.
Use of animation in presenting modeling, use of real world example, guest speakers, and hands on practice
By making them read at home and applying the knowledge in class students learn fast and efficient.
Selecting only relevant chapters from textbook, not discussing others.
Mixture of course delivery methods – exercises, practical examples, class participation etc.
This is a big challenge. I try to use just-in time learning, so they learn how to do something in class and then apply that technique for their client. The project is divided into 6 milestones: team establishment, problem id, current system processes, proposed system, final report, final presentation/poster. This is crucial.
I try to be judicious about what I cover. Based on my discussions with colleagues/adjuncts, I try to cover the material that is currently important. Over the course of two semesters, I think I get most of the important stuff. Are they overwhelmed? Maybe. Are they bored? Well, life isn't always fun, is it?
In the first course I concentrate totally on the Planning and Analysis phases. I emphasize the importance of project management (work plan, cost benefit analysis, breakeven) and on business modeling using UML. I focus on the Use Case, Class and Activity diagrams. I do not pursue design topics until the second course.
By keeping them involved and sometimes be relaxed on some issues like deadlines.
What works best is to scale the project down to a manageable level, this also allows for a discussion on scope and leads to the class developing a statement of scope
I give up some course content via lectures to focus on the project.
I view the course as a design course.
This is the reason for split into 2 courses
I have taught the course for 12 years and it has been a learning experience. I develop a schedule that is planned out for each class period. I don't try to cover the entire text book. I pick topics that I feel will help them the most in the real world. There is a link to the course outline above.
Keep a good balance of conceptual and practical/hands-on
The hands on activities focus most students.

Cut down on the volume. Cannot cover everything and do it well. Better to do less
Explain the material and then let them do a small in-class assignment in a group
30 years of program manager experience
I have found that carefully selected smaller chunks gone over in depth is far more effective than a voluminous broad overview.
The lectures are very short, maybe 5-10 mins followed by discussion or hands-on.
By engaging them in discussion and activities
We focus on the deliverables each week

Appendix III

Respondents' answers when prompted to: "List some hands-on activities you do in your SA&D Course"

Gathering requirements, designing a solution, coding, testing
Research project: best analysis and design practices in real companies: which problems they find while developing software, how do they solve/try to prevent them, etc.
Lots of exercises on design patterns. Working in groups, they present the solutions to a selection of problems.
Each of the deliverables in the SDLC is explored with the final outcome a deliverable; might use project management software, CASE tool, etc.
Perform and document varíous analysis and design tasks. Program, test and document (parts) of the system
They use Microsoft Visio to draw the dfds; VB for the screenshots; Word for the narratives; and Excel for the reports.
Draw diagrams on the board in class; have students evaluate each others' work; practice modeling and defending answers for choices include in models; in class discussions to provide support for why one way of modeling a particular problem provides a better or more useful approach than another; CRC cards; act out CRC exercise;
Practice with case tool- Visible Analyst, make decision table, calculate cash flow, find solutions to cases
Dfd, erm
I have a hands on activity for most of the important topics. Sometimes I use an activity from the back of the book, other times I create something.
Uml modeling
DFD, interviewing, reviewing each other project ideas (small groups)
In class exercises on Process and data modeling, in class Lab time to work on projects
Work on project during class meetings
Theory, exercises or case studies on different types of models, cost/benefit exercises etc.
Modeling problems, interview role playing, mini-cases
I have them work on the milestones for their projects (data flow diagrams, data models, requirements analysis plans, etc.
Comprehensive group project throughout the semester going through the all the activities of SDLC, three-to-four presentations on key areas of learning/SDLC (problem definition, process modeling, data modeling and program design), use of tools such as Microsoft Project, Visio, Visible Analyst, as well as develop all documents necessary to go through the SDLC activities. Also write two research papers, and sometimes in-class case studies from the book. Most emphasis is on data-flow diagrams that students only learn from this class.
Activity Diagrams, Data Flow Diagrams, ERD's, interface prototypes and implementation recommendations.
In addition they interview user(s) from the project and conduct a needs analysis, they also present their projects to the department faculty
Project, interviews, data collection, analysis, data flow diagrams, project design, meetings with clients
The students learn to use CASE tools.
SAD modeling methods
Modeling problems in class: dfd, erd, data dictionary. Process specs, use-case diagramming.
The in-class project described above is the main in-class hands on.

Dfds, interviewing
Give class time for students to work on their projects and work on the interfaces between their various projects, also have several in-class worksheets for dfds
Games, mini cases, exercises
Practicing modeling techniques, doing presentations and walk-throughs
Create project documents
Erds, dfds, Data models
Assignments on Visible Analyst and Access, plus they work on their projects
Data flow, project mgt
Requirements classification "game"
Design the worst interface you can
Mock client requirements interview
Break into a website (application security)
We do some sample parts of the projects on a simplified case study so they are aware of pitfalls and issues that may arise.
Project request, project plan, use case model, class diagram, sequence diagrams, object state charts, normalized data model
Data-flow diagramming, project management software, report development
Create a data model, create a process model, Read current articles that pertain to the topic of the week, and disucss them in class.
Class discussions and exercises, homework assignments on dfds, ERD, problem definition, proposed design

Appendix IV

Respondents' answers to the question: "Do you have any other techniques, comments, or concerns that you would like to share with this study?"

The flow of the book should be such that a student can logically progress from business case to a simple design document. My suggestion will be to deemphasize database design, interface design etc since there is no time for these topics. Database design should be done in a database course anyway. Include risk management
Since the course is so heavy on team work ensure that there is a evaluation rubric for team participation; everyone evaluates the time/effort put forth by each member of their team, will eliminate the social loafer from earning the same grade as the rest of the team.
I think that a 2-semester sequence would be very helpful
I would like to see Agile (scrum) featured in a textbook. Even though some businesses would not go with this methodology I think in a web class my students could get through some aspect of a project using Agile in a single semester. With students living in diverse locations a collaborative environment could work with a Agile project.
Business rules are also extremely important in our class,
Blend of standard textbooks with modern examples (facebook, Nintendo Wii, Twitter, iphone etc) works very well because of the deep familiarity of the students.
Use many examples from industry in class discussions.
Interacting with a real client is invaluable.
I think the key is to make the material as practical as possible. I come from industry, and I have lots of war stories, which seem to help. I also bring in guest speakers from industry to back up and extend what I talk about. I get good feedback from those classes.
I have my student-groups develop almost every document that I discuss in the class and submit at the end of the semester as a portfolio.
The course projects are completed in groups with every attempt made to get a cross section of majors in each group
Our Advisory Board saw a shortfall in our previous program – their observation was that students need to work in a large systems environment during some part of their coursework – not just small-scale stand alone projects.
We are fortunate to approach the course sequence as team of faculty rather than using a silo-based approach to the courses in the MIS major.
At another school at which I taught this was spread over 2 semesters and it worked much better.
This is our capstone sequence and fulfills many of our ABET outcomes (we are ABET accredited) so it's a very important sequence and we put our best teachers into it
The only way students learn the techniques is to do them
If the course is in a business school for students who seek to become managers, being able to recognize data and how it traverses the company is more important than creating a design that is ready for the programmers. Hence, the use of object modeling may not be the most effective. Data flow diagramming is one technique that is both useful and can be easily understood by others in presentations. If the sa&d course is directed to cs students, then object notation might be best.
I mix theory and hands-on in every class to make it interesting. Only theory makes it boring.
The main problem I encounter is convincing the students of the relevance and importance of the material -- often they seem disinterested
While doing a real project allows the students to see the difficulty of getting requirements with clients, it is difficult to control.