

Negotiating Knowledge in Systems Engineering Curriculum Design: Shaping the Present While Struggling with the Past

Pieter W. G. Bots and Willem A. H. Thissen, *Member, IEEE*

Abstract—Designing a systems engineering curriculum is a complex process, not in the least because it involves a variety of academic professionals whose perceptions and interests rarely concur from the onset. The variety in stakeholders breeds variety not only in values and objectives, but also in supposed—and mostly tacit—views of an educational system. In such an ambiguous design context, models serve to make knowledge explicit and facilitate communication. This paper contains a description of curriculum design in systems engineering, policy analysis, and management (SEPA) at Delft University of Technology (DUT), with a particular emphasis on these models, and the way they are embedded in a systems approach to curriculum design.

Index Terms—Curriculum design, models, systems engineering education.

I. INTRODUCTION

WHILE a variety of educational programs in systems engineering have been developed (see [2], [7], [10], and [11]), little attention so far has been paid to issues in, and systemic approaches to, curriculum design. Traditionally, educational programs have been developed with strong if not dominant input from the educational entrepreneurs in the beginning stages. Subsequently, a process of adaptive evolution or “adaptive design” is seen, in which, gradually, changes are made at the individual course levels and with respect to the linkages between courses. The outcome of such processes is, to a large extent, determined by negotiations between the course providers, and guarantee neither transparency nor improvement at the curriculum level.

In an earlier paper [1], we advocated a more systemic design approach. An engineering curriculum may be “engineered” in a purposeful way by starting with a specification of curriculum objectives, as the functional requirements for the curriculum. Complex as these may be, these objectives may be further operationalized as educational goals, which then constitute the basis for the specification of course contents. Educational goals and course objectives are inputs for decisions regarding the working method for courses (traditional lectures, lab assignments, project course, etc.). We argued that parallel to such “top down” engineering of course content and form, mechanisms for quality control and evolution should also be put in place. Finally,

we emphasized the need for management of the design process, meaning deliberate decision making on the timing, purpose, and way of involvement of the different stakeholders, in particular the course providers and the students.

We do not claim that these guidelines constitute the beginning of some curriculum design theory, of some invariant order of steps to be taken in formulating a curriculum. Rather, we agree with Hansen [5] in that we want people to design particular curricula in intelligent ways. If our experiences with developing the Delft University of Technology (DUT) curriculum in systems engineering, policy analysis, and management (SEPA) suggest that “purposeful” curriculum engineering is feasible, they certainly confirm the need for management of the design process. In this paper, we focus on an issue that is central to such process management: the imperfect knowledge that stakeholders have of the system under design. The message we wish to convey is that models of the educational system are effective management instruments, provided that sufficient time and effort is invested in their institutionalization. After a summary of the SEPA curriculum design problem in Section II, we iterate over the models featured in the “top down” engineering approach that we presented in [1] in Section III. For each model type, we discuss the specific conditions that may facilitate or inhibit their effective use as a vehicle for communication between stakeholders in their learning and negotiation processes. In Section IV, we make an attempt to draw some more generic conclusions from our observations.

II. SEPA CURRICULUM DESIGN PROBLEM

The SEPA curriculum design problem was not a “green field” situation. Rather, the four year undergraduate curriculum developed between 1992 and 1996 was in need of redesign. This curriculum was developed in an adaptive design process, the four years of the curriculum being assembled one year at a time, using mainly (derivatives of) existing courses. This “bottom up” approach left much freedom to course providers, while time for course integration was scarce. Courses, and—albeit to a lesser extent—course integration, were improved gradually on the basis of teaching experience in the preceding year. In 1996, with all four years fully operational, deliberate redesign using a “top down” approach became an option. Expanding on earlier work, we reflect on five different aspects of the design problem that we consider to be critical to the success of a “top down” approach.

Manuscript received December 1, 1998; revised March 1, 1999.

The authors are with the Faculty of Technology, Policy, and Management, Delft University of Technology, NL-2600GA Delft, The Netherlands (e-mail: p.w.g.bots@tpm.tudelft.nl; <http://www.tpm.tudelft.nl>).

Publisher Item Identifier S 1094-6977(00)04792-1.

A. Formulation of Objectives

A “top down” design approach implies that the system output, being the type of graduate the curriculum should lead up to, is specified in detail, and that all design decisions are made to best meet these specifications within given time and budget constraints. At a general level, there seems to be agreement on the key learning objectives of systems engineering education [9] which are to

- develop the ability to deal with a variety of complexities, including interdisciplinarity and dealing with a variety of stakeholders;
- develop versatility in both analysis and synthesis;
- develop the ability to implement solutions;
- acquire sufficient knowledge of systems methods and tools.

In addition to these general systems engineering characteristics, it is widely agreed that the acquisition of substantive knowledge of one or more specific systems product types and/or application domains is desirable. Depending on the primary focus, for example, the design of technological systems, of human-machine systems, of business systems, or of public policies and systems, a choice is made for depth with respect to specific types of systems, methodological, and managerial challenges.

In the case of SEPA at DUT, the choice was made to focus on the design of systems and policies where the public sector and the interaction between public and private sectors play an important part. This involves public policy analysis and design, the design of inter-organizational systems and processes, and strategies for change, typically in the domains of transport, information services, water management, industrial systems, and energy provision.

While these objectives provide a general direction to the curriculum design process, they leave ample opportunity for variety in interpretation, for example with respect to the level of detail of systems design, and with respect to the need for specific knowledge elements. As a consequence, the process of defining required knowledge elements still has a bottom-up character, consisting of identifying a range of possibilities and then making a selection on the basis of the views of those involved in the process.

B. Route to Knowledge and Skills Acquisition

When (or should we say “if”?) curriculum objectives have been formulated, the curriculum designers must turn to the issue of structuring the learning process. Design decisions regarding the sequence in which topics are taught and skills are trained require a basis of causal assumptions on how system engineering knowledge and skills are best accumulated. The most common, “traditional” structure is more or less sequential: First, focus on basic system theories and quantitative modeling methods, then gradually build up system complexity, and finally, in the last year(s) of the basic (undergraduate) program, broaden the student’s perspective to problem solving methodology in the context of which the theories, tools and techniques learned in the first years are applied. In a final capstone course, some management aspects are introduced typically with a focus on project management and cooperation skills.

Criticism of this traditional build-up has stimulated thinking about other structures. The criticism is basically two-fold. First, as students gain familiarity with a specific way of thinking and a specific set of tools or techniques, many develop a tendency to apply these as solutions to whatever problems are given to them, instead of developing a broad problem-centered attitude and approach, in which the characteristics of the problem situation guide the selection of techniques. Second, it may be far more motivating to introduce students right from the beginning to the type of real-world challenges for which they are being educated, and enable them to discover the need for methodologies, systemic theories and tools as well as social and behavioral knowledge and abilities.

From this perspective, a curriculum structure which starts right away with systems problem solving methodology using real-world cases as a core, followed by a more or less concentric build up of competence in dealing with relevant types of systems complexity and methodology may provide advantages over the traditional approach. Another issue is whether or not to introduce management complexities as well in the first year, in order to confront the students with this important aspect in as early a phase as possible.

C. Choice of Educational Forms

When subject matter has been established, curriculum designers must decide on the most appropriate educational settings for students to develop the desired insights and skills. Therefore, a second “body of knowledge” relevant to curriculum designers is the set of assumptions concerning the effectiveness of educational forms. Admittedly there is but limited empirical evidence in this area, but we can discern the following trends.

- Education in “hard” systems analysis and modeling¹ generally follows a traditional approach, in which basic theories and methods are studied using lectures and textbooks, supplemented with exercises to provide the necessary practical drills. The overall philosophy is to start simple, and gradually increase complexity.
- Education in methodology displays—in as far as it can be distinguished clearly—a mixed pattern of theoretical and problem-driven approaches. Basic concepts and principles are taught in a traditional classroom setting, but many curricula provide problem-oriented project courses in which students develop the ability to apply the methodology in a simulated problem-driven context.
- Education in management complexity—in as far as it is part of engineering education—generally is introduced through project work, in accordance with the trend that has developed in most business schools. In some cases, the management component exclusively focuses on project management and teamwork, in other cases, dealing with stakeholders and conflicting interests is also part of the challenge.

Clearly, an important design issue in curriculum design is how educational forms can be integrated in courses to maximize their potential synergy. A difficult issue, since what knowledge there

¹See, e.g., [6] for a distinction between “hard” and “soft” systems engineering.

is on the effectiveness of different educational forms resides with the teaching staff, who almost invariably have acquired this knowledge in one particular field (e.g., software engineering) from which it is difficult to generalize to other fields (e.g., civil engineering).

D. Tradeoff Between Quality and Quantity

Ideally, a systems engineering curriculum transforms 100% of the student body into 100% qualified engineers within the given time frame (four or five years in The Netherlands). In practice, 60% of all students entering a Dutch engineering program obtain the degree and need on average six years to finish their studies; and it is most difficult to define what would constitute a 100% qualified engineer. In other words, system performance in terms of quality and quantity of its output is difficult to measure. Performance is known to be affected in general by changes in the quality of the student inflow, the amount of effort invested by the student, teaching quality, examination level, and the school's organization. The system quickly adapts to changes, albeit within limits: student effort will increase as tests become more competitive (good for both quality and quantity), but above some unknown level, quantity will drop. Conversely, lowering standards will initially raise quantity, but cause student effort to drop. Many more such feedback mechanisms can be demonstrated to exist, but the concise "laws of conservation" that govern educational systems are unknown. Nonetheless, the SEPA redesign objective was to improve output quality while maintaining the current throughput, by increasing synergy within the curriculum and removing organizational barriers.

E. Politics

So far, we have addressed the substantive aspects of the curriculum design problem, and the lack of "hard" knowledge of the system to be designed is already obvious. Even without the intricacies of organizational politics, coping with the variety of unknowns, constraints, and degrees of freedom would constitute a staunch challenge; but politics *are* an important factor, since curriculum redesign implies a "brown field" situation with scores of professionals ready to defend their position and beliefs. Subject groups compete not only for the establishment of their vision of a particular area of the curriculum [8], but also to protect their territory in terms of both size and autonomy. The history of SEPA shows an academic staff buildup from four to 40 in three years. The staff has expanded too rapidly for the original kernel of "believers" to impress their vision—which is by necessity ambiguous—on the newcomers. At the same time, the brunt of the courses in the first two years of the curriculum (mathematics, economics, law, and technical subjects) was taught by staff of other faculties, who tailored their usual courses to meet specific SEPA objectives. New SEPA staff, specifically in the areas of technology and economics, wish to redesign these courses as they see fit, while other faculties wish to protect their investment in these courses. First-generation SEPA staff would like to see more of the SEPA-specific methodological insights acquired in the past four years incorporated in all courses.

The needs of curriculum management add to the variety of objectives and views: they stress transparency of curriculum struc-

ture, simplicity of regulations and efficiency. Efficiency requirements lead to a demand for more extensive forms of teaching, reducing teaching loads on the staff.

In such a setting, such imperfect knowledge on cause and effect, objectives, and constraints that is available, is likely to be interpreted and used in a variety of ways to further individual goals.

Summarizing, the SEPA curriculum design targets are increased integration of SEPA knowledge and skills while observing existing constraints with respect to curriculum length and orientation, and efficiency. This requires sharpening of curriculum objectives, and exploiting insight and experiences regarding knowledge build up and regarding the effectiveness of different educational forms.

III. MODELS IN CURRICULUM DESIGN

As we have shown in the previous section, curriculum practice is a multifaceted concept, constructed, negotiated, and renegotiated at a variety of levels and in a variety of arenas ([5], quoting from [4, p. 49]). In this section, we expand on our argument that models are indispensable in curriculum design. We introduced a specific set of models in [1] as tools in the different stages of a "top down" design approach. Presently—two years later—we can reflect on their effectiveness, looking for specific conditions that facilitate or inhibit their effective use as a vehicle for communication between stakeholders in their learning and negotiation processes.

A. Graduate Profile Definition (GPD)

The profile of a SEPA graduate has previously been described in [11] as that of "an engineer in the sense of problem solver/designer/integrator, knowledgeable in both technical and behavioral fields, and capable of communicating, organizing and decision making in technical environments characterized by multiple stakeholders, ambiguity and uncertainty." To an engineer, such a description would hardly seem to qualify as a model. But when extended to a two-page profile that further elaborates on the key words, profile definitions like these constitute verbal models that communicate a vision in the same way an "artist impression" conveys what an automated vehicle guidance system will look like. As a vision communicator, the GPD serves an important purpose.

Such a general profile definition, however, leaves a lot to be defined more precisely in terms of course contents. There is a risk of sticking to generic buzz-words that are acceptable to all, precisely because they allow freedom of interpretation.

Our experience indicates that this has, to a certain extent, indeed happened. This may have been prevented by deliberately investing in a careful choice of the key concepts and in intense communication of these to the staff to achieve sufficient inter-communication.

B. Knowledge Cluster Definition (KCD)

At its conception in 1991, the SEPA curriculum was outlined in terms of four knowledge clusters: mathematics, technology, administrative science and methodology. This clustering served mainly as a rough profile to show the relative weight of these

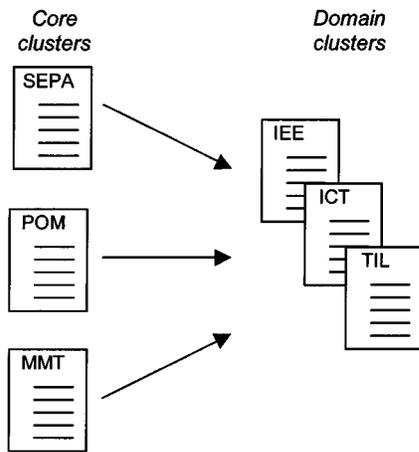


Fig. 1. Knowledge clusters in the SEPA curriculum.

four components. In 1996, the curriculum content could be defined in more specific terms. Doing justice to both the SEPA curriculum and the organization as it had developed, six knowledge clusters were defined as follows (see Fig. 1):

- 1) systems engineering and policy analysis (SEPA);
- 2) policy, organization, and management (POM);
- 3) mathematical methods and tools (MMT);
- 4) industrial systems, energy, and the environment (IEE);
- 5) information and communication technology (ICT);
- 6) transport, infrastructure, and logistics (TIL).

The first cluster contributes the methodological knowledge and skills for complex systems analysis and design in a technical/political context. The second cluster contributes knowledge from behavioral fields. The third cluster, plus one of the three domain-specific clusters students may specialize in, contribute to the student's technical background.

The definition of knowledge clusters reflects the classic design practice of complexity reduction by means of decomposition in subsystems. This practice requires that the relations between these subsystems also be defined. In the SEPA design, the three core clusters contribute basic principles, methods and tools that find their application in the respective domains. The decomposition also furthers the objective of some uniformity among the three domain clusters by requiring (without going into details) explicit coverage of domain-specific economics, law, and policy.

Like the GPD, the KCD is a verbal model. The specific contributions are elaborated typically in one page of text. Since the focus in this stage is on the engineering profile, knowledge areas are defined in terms of "knowledge components" that are explicitly linked to engineering skills, e.g., "the ability to choose an appropriate modeling methodology" for the MMT cluster, and "essentials of traffic engineering" for the TIL cluster.

The KCDs proved to be quite effective in the design process, since they gave the six individuals who were to act as "cluster coordinator" a standard to meet. A caveat of decomposition is that certain knowledge areas risk "falling between the cracks." For example, as information systems was not defined as a separate core cluster, this part of the original curriculum was scattered: introduction to computer science became part of MMT,

information systems design went to SEPA, and management information systems to POM. The disciplinary background of cluster coordinators may also introduce a bias in the components in a KCD: with a strongly policy and management-oriented cluster coordinator, the position of economics and law in the POM cluster stood the risk of being marginalized.

C. Specification of the Full Body of Knowledge and Assignment of Competence Levels

The principle of hierarchical decomposition is taken one step further by constructing a "competence tree" (see Fig. 2). In this model, the information contained in the GPD and KCDs is elaborated in more detail. For each knowledge component, knowledge elements are specified: theory or skills that the student is to master at a certain level of competence.

Competence levels may be operationalized on a scale similar to the one used by the DPMA [3] as follows:

- 1) no coverage;
- 2) *awareness*—introductory recall and recognition;
- 3) *literacy*—knowledge of framework and contents;
- 4) *concept*—comprehension as exemplified by translation, extrapolation, and interpretation of meaning;
- 5) *detailed understanding*—appropriate application of knowledge in a structured or controlled context;
- 6) *skilled use*—application using analysis, synthesis, and evaluation in new situations.

When competence levels have been assigned to all knowledge elements, the curriculum is fully specified. The general objectives worded in the GPD have been translated into operational goals.

In reality, it turned out to be tremendously difficult to construct a competence tree, for a number of reasons. First, domain specialists had trouble relating their contributions to the overall SEPA objectives. Many often lacked sufficient overall SEPA knowledge to invest effectively in a more integrated set up. They experienced the approach as overly restrictive and tended to limit themselves to safeguarding their course based on their own perspective of knowledge requirements. Second, staff has an interest in earning the returns on previous investments in mono-disciplinary course materials. Third, staff from other faculties lack incentives for change as there will be no rewards whatsoever from within their own faculty.

D. Course Design

Assuming that the curriculum content has been defined as knowledge elements in a competence tree, tensions between content, sequence and form must be resolved. The sequence must meet precedence constraints: learning to apply the laws of thermodynamics, for example, requires specific mathematical competence. The form can vary as long as it stays within institutional conventions, such as a four year program and five teaching periods in a year, and the credit system. The design objective is to provide a smooth path for students to acquire knowledge and use this knowledge to achieve the desired competence levels.

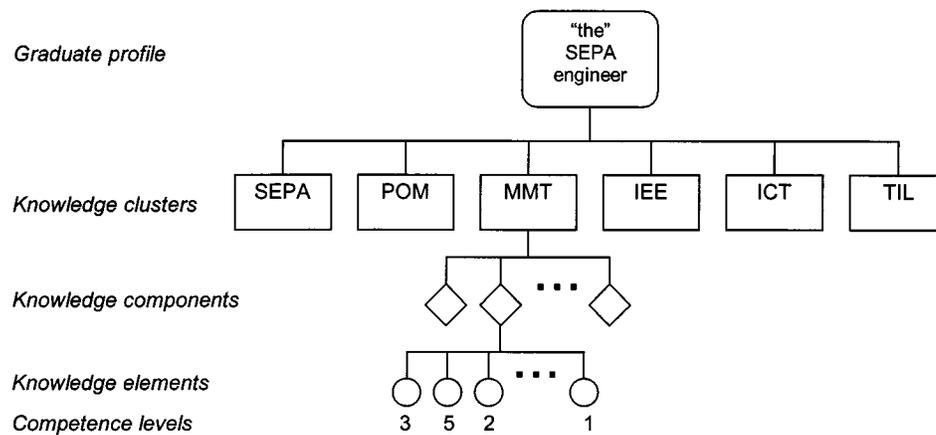


Fig. 2. Structure of a “competence tree.”

Cluster	Year 1	Year 2	Year 3	Year 4	Total
SEPA	6	3	6	3	18
POM	6	10	9	3	28
MMT	14	13	3	-	30
domain	6	6	8	8	28
projects	10	10	10	22	52
choice	-	-	6	6	12
	42	42	42	42	168

Fig. 3. Credit table for the SEPA curriculum.

This “top down” course design is facilitated by using *credit tables*, *year schemas* and *course patterns* to resolve the tension between content, process and form at different levels of detail.

In a credit table, credits are allocated in “lump sum” fashion to clusters for each year of the program. Fig. 3 shows the credit table for the four-year, 168-credit SEPA program, one credit equaling one week (40 h) of study. The rightmost column with row sums shows the relative “weight” of the clusters in the curriculum. The table reflects some basic choices: an early, solid mathematical training, gradually more emphasis on the POM cluster, continual attention for the chosen application domain (IEE, ICT, or TIL), strong emphasis on project work, and some freedom in selecting courses from the full range taught at DUT in the last two years. The 22 project credits in the fourth year constitute the student’s graduation project, typically six month full-time.

Fig. 4 shows how the credits from the first and second year in Fig. 3 are distributed using year schemas. The basic structure of a year schema consists of two theory tracks with three-credit “slots” and a project track with two-credit “slots.” In this way, 40 credits can be allocated evenly over the five study periods used at present at DUT. This leaves two “floating” credits. The rationale is that this structure strikes a balance between two objectives: maximize variety and minimize fragmentation. Each teaching period students are involved in two theory modules

and one project module. For the first year, the “floating” credits are allocated to the MMT cluster, and the project track is organized into three small and one larger project. In the second year, the “floating” credits are allocated to the POM and the MMT cluster.

Elaboration of course patterns for each of the courses in the year schemas proceeds “inward” in the sense that entry and exit competence levels are specified first. The knowledge elements for which competence is to be enhanced during the course are detailed further by specifying for each week the topics to be covered. These specifications form the basis for checking coherence and precedence as well as assessing whether the student work load is proportional with the allocated credits.

E. Maintenance

Recognized as crucial for the “adaptive” curriculum design approach taken, formal procedures for course evaluation were implemented in 1992, when the SEPA curriculum was taught for the first year. Using extended course descriptions (in fact similar to course patterns, but without the formal specification of competence levels) as points of reference, courses are evaluated systematically (see Fig. 5). A further elaboration of the quality control system and experiences so far can be found in the paper by Van Peppen and Van der Ploeg in this issue.

IV. EXPERIENCES AND CONCLUSION

The preceding sections show that a systemic approach to curriculum design can be defined and implemented. The soundness of this approach is yet to be demonstrated. As the curriculum reengineering process is still in an early stage, the first experiences reported here are impressions rather than conclusions.

- Specification of knowledge elements is indeed time consuming, and in particular difficult where it comes to choosing the appropriate level of detail.
- The concept of competence levels is elegant, yet difficult to operationalize. There is a tendency to use courses and students in mono-disciplinary programs as a reference. This is found to be counterproductive when the objective is to synthesize new courses covering a variety knowledge elements.

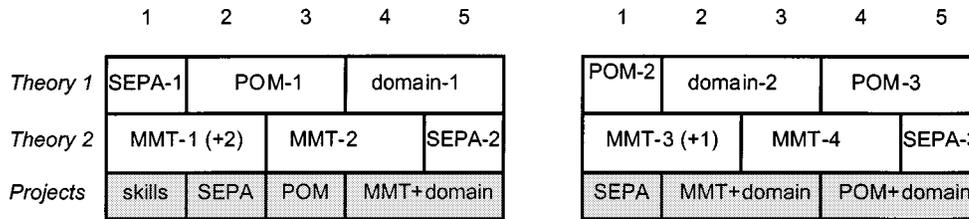


Fig. 4. Two year schemas for the SEPA curriculum.

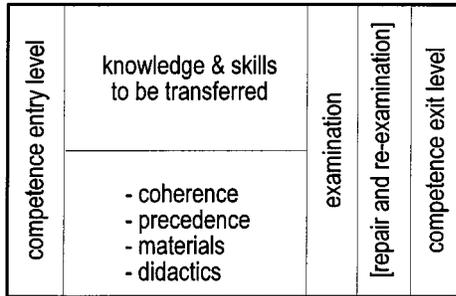


Fig. 5. Possible structure of a course pattern.

- Competence levels as cited in Section III relate to knowledge rather than skills. This makes the 0–6 scale unsuited to express gradual competence buildup in e.g. presentation skill or scientific report writing skill.
- While discussing curriculum designers tend to switch rapidly between aggregation levels, e.g., when countering arguments concerning allocation of credits to clusters with arguments concerning the content of a specific course. The models presented in Section III seem to be effective as structuring tools.
- Uncertainty is a major disincentive for course providers to develop new courses. Making choices “of principle” regarding the number of credits per cluster and course will increase the momentum of the design process. The design approach must nonetheless allow room for renegotiation.
- Generalizing the previous observation: although presented as such that way, the “top down” design approach is not linear. Fully consistent with experiences in other forms of systems design, iterations over stages are needed.
- The main advantage of the systemic approach is its rationality. Explicit formulation of objectives, means-ends, relationships, and mechanisms for evaluation and improvement create a transparent development process which favors system optimization. Drawbacks of this approach are the risk of failure in goal formulation, the complexity of design, and a lack of commitment from course providers that see their professional freedom taken away from them.
- Decomposition into clusters, combined with a hierarchy of responsibilities (a design process manager, cluster coordinators, course coordinators) is an effective means to cope with the complexity of a “grand design” approach.

- The internal and external communication concerning the design process must be carefully managed. Specifically, information about important design decisions must reach all actors involved synchronously to avoid confusion.
- The designers must stay aware of potential overlap with courses taught by other departments. A market focus by choosing for specific application domains may have relations to market segments already served by other groups at the university!
- In the process of specification of course contents, staff appear to be strongly biased by the content of existing courses. Often, their professional (mono-disciplinary) specialization hampers them in relating their design to the overall curriculum contents and philosophy.

Few of these experiences are surprising. Enhancing rationality in the curriculum design “arena” with its many actors and their perceptions and interests remains a challenge.

REFERENCES

- [1] P. W. G. Bots and W. A. H. Thissen, “Issues in systems curriculum design,” in *Proc. 1996 IEEE Int. Conf. Systems, Man, Cybernetics*, 1996, pp. 3144–3149.
- [2] G. Dauphin-Tanhuy, P. Deshayes, and N. Qualyle, “Educational project and educational content: Integration and interaction,” in *Proc. 1995 IEEE Int. Conf. Systems, Man, Cybernetics*, 1995, pp. 4346–4349.
- [3] Information Systems: The DPMA Model Curriculum for a Four Year Undergraduate Degree (1990). [Online]. Available: <http://negaduck.cc.vt.edu/DPMA/committees.html#Curriculum>
- [4] I. Goodson, “Studying curriculum: Toward a social constructionist perspective,” in *Qualitative Educational Research Studies: Methodologies in Transition*, I. Goodson and M. Mangan, Eds: Res. Unit Classroom Learn. Comput. Use Schools (RUCCUS), Faculty Education, Univ. Western Ontario, London, ON, Canada, 1991, pp. 49–90.
- [5] R. E. Hansen, “Five principles for guiding curriculum development practice: The case of technological teacher education,” *J. Ind. Teach. Ed.*, vol. 32, no. 2, pp. 30–50, 1995.
- [6] M. C. Jackson, *Systems Methodology for the Management Sciences*. New York: Plenum, 1992.
- [7] J. L. Kays and M. L. McGinnis, “A historical perspective of undergraduate systems engineering at the united states military academy,” in *Proc. 1995 IEEE Int. Conf. Systems, Man, Cybernetics*, 1995, pp. 4356–4360.
- [8] R. McCormick, “The evolution of current practice in technology education,” in *Proc. NATO Advanced Research Workshop: Integrating Advanced Technology into Technology Education*, Eindhoven, The Netherlands, Oct. 1990.
- [9] A. P. Sage, *Systems Engineering*. New York: Wiley, 1992.
- [10] W. T. Scherer and D. E. Brown, “Real design problems and real clients: A systems engineering senior design program,” in *Proc. 1995 IEEE Int. Conf. Systems, Man, Cybernetics*, 1995, pp. 4341–4345.
- [11] W. A. H. Thissen, R. R. Meinsma, and P. G. J. Twaalfhoven, “Project learning in the TU Delft curriculum in systems engineering, policy analysis, and management,” in *Proc. 1995 IEEE Int. Conf. Systems, Man, Cybernetics*, 1995, pp. 4350–4355.



Pieter W. G. Bots received the M.Sc. degree in computer science (specializing in compiler construction and computer linguistics) from Leiden University, The Netherlands, in 1985, and the Ph.D. (with honors) from Delft University of Technology (DUT), in 1989, where he studied design methodologies for decision support systems.

Until 1991, he was Assistant Professor with the Information Systems Group, DUT. Since 1992, he has been an Associate Professor with the Faculty of Technology, Policy, and Management, DUT. His current research within the Policy Analysis Group focuses on methods and tools to support multistakeholder policy and decision making processes, in particular dynamic actor network analysis (DANA).



Willem A. H. Thissen (M'80) received the M.S. degree in engineering physics in 1973 and the Ph.D. degree in systems and control engineering in 1978, both from Eindhoven University, The Netherlands.

He was with the Department of Systems Engineering, University of Virginia, Charlottesville, and The Netherlands Ministry for Transport and Public Works before joining Delft University of Technology, Delft, The Netherlands, in 1986, to become one of the Founders of the new School of Systems Engineering, Policy Analysis, and Management, and

Head of its Systems and Policy Analysis Department. His research and teaching focuses on interdisciplinary systems approaches to the analysis and resolution of policy problems, in particular in water resource management, transport and infrastructure policy, risk and safety policy, and technology policy. He has published on a variety of subjects, including modeling and simulation, policy analysis methodology, and systems engineering education.

He was a Co-Guest Editor of the May 2000 Special Issue on Systems Education for the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS and is a member of the Executive Committee of the Systems, Man, and Cybernetics Society.