What's in a Prerequisite

Roland Hübscher
Auburn University
Department of Computer Science and Software Engineering
107 Dunstan Hall, Auburn, AL 36849-5347, U.S.A.
roland@eng.auburn.edu

Abstract

It is important that all pedagogical design decisions in educational learning environments are made explicit to the educator. However, implicit assumptions affecting or even contradicting the "official" pedagogical method may unbeknownst slip into the design. We discuss a concrete problem in the area of adaptive hypermedia systems related to the notion of a prerequisite. Prerequisites are frequently used to describe the many different educationally effective paths on which learners may traverse a hypermedia system. However, closer look reveals that several different concepts are called prerequisite and confusing them may lead to flawed designs. We will make recommendations as how to avoid misusing the different notions of prerequisites in adaptive hypermedia systems.

1. Introduction

There will always be some implicit design decisions in a complex system, now matter how hard designers try to avoid them. Some of these decisions may be inconsistent with the pedagogy employed by the learning environment potentially hampering its effectiveness. We will discuss this issue in the area of adaptive hypermedia systems (AHS). Although a explicit pedagogical model is part of the general architecture, assumptions inconsistent with the pedagogical model may sometimes slip in through the backdoor. One of these backdoors is the concept of a "prerequisite," or, more correctly, the concepts *(sic!)* referred to as prerequisite.

Hypermedia has been recognized as having great potential in providing content to learners. Relationships between concepts can be made explicit with hyperlinks, and the same material can be organized along different dimensions presenting the material to be learned from different views [1].

Hyperlinks allow each individual learner to traverse and explore the content in a way that fits his or her interests and learning goals the best. However, this added flexibility, compared to books which are often read in a more or less linear fashion, can also cause problems seriously impacting the pedagogical benefits of a hypermedia system. It is quite easy for the learner to lose orientation that manifests itself in the learner not knowing how the current page fits into the big picture and what hyperlinked path to follow.

This is one of the problems that adaptive hypermedia attempts to solve. Adaptive navigation support and content organization is used to present the learner with the most effective paths through the hypermedia system given the learner's characteristics, skills, learning goals, and behavior.

This approach is also known as curriculum sequencing. For most systems using curriculum sequencing, this means that the pages are organized in a partial order which depends on the pedagogical model underlying the system and the content to be presented to the learner.

Often, the concept of a prerequisite is used to define this partial order. Although, different systems vary in the details of how prerequisites are implemented and used, we can assume that if page p_a is a prerequisite of page p_b then the content a described on page p_a must be learned before p_b describing content b is allowed to be visited. Without loss of generality we will assume that each concept x is described on exactly one page p_x.

As a common result of this approach, the pages (or concepts) are partially ordered such that if concept a is required to understand concept b, then p_a is a prerequisite of p_b resulting in the constraint that p_a must be visited before p_b.

In this paper, we will show that this decision already includes some strong implicit assumptions about the underlying pedagogy, even in systems with explicit pedagogical models. We will suggest how this problem can be solved resulting in systems whose adaptive mechanism and the pedagogical model are clearly separated.

2. General Architecture

We base the discussion of the general architecture of adaptive hypermedia systems on the work of De Bra and colleagues whose model is called AHAM [2, 3]. Al-

though other architectures could be used, AHAM is used because it explicitly refers to the pedagogical model.

The main elements of AHAM are the domain model, the user model, the teaching model, and the adaptive engine.

The domain model describes the content to be learned, that is, the concepts and how they are structured and related to each other.

The user model describes characteristics of an individual learner using the hypermedia system. For each user of the system, a specific model is built based on information gained through pretests, observed behavior, online quizzes, etc.

The teaching model (or, as we call it, the pedagogical model), describes how the user model and domain model are used to do the adaptation which is then actually executed by the adaptation engine.

The purpose of the pedagogical model is to make explicit how the system is adapted to the learner's needs. Thus, with respect to navigation, it describes what the pedagogically effective paths are for each individual learner. This is not a trivial task. It only takes 13 pages to provide a different path for every person of the earth's current population of six billion people.

3. Prerequisites

As mentioned earlier, prerequisites are often used in adaptive web sites to partially specify in what order the web pages need to be visited. The definition of a prerequisite in this context is as follows. If page p_a is a prerequisite of page p_b, then page p_a must be visited before page p_b may be visited. This definition of a prerequisite specifies a mechanism to partially order pages in an adaptive web site.

Switching over to an educational context, the concept of prerequisite is used in a pedagogical sense. For instance, if concept *a* is required to understand concept *b* then *a* is called a prerequisite of *b*. Alternatively, at a more coarse-grained level, if the material of course *a* is required for the material in course *b* to be mastered, *a* is declared a prerequisite of *b*.

Let's assume that *a* is required to understand *b*, that is, *a* is a prerequisite (in the pedagogical sense) of *b*. By definition of a prerequisite in adaptive web sites it follows that the page describing concept *a* must be visited before the page describing concept *b* may be accessed. In short, if *a* is required to understand *b* then *a* must be visited before *b*.

However, this conclusion is not correct for the general case and may contradict the specifications in the pedagogical model of the AHS.

3.1 A Prerequisite is Not a Prerequisite

Although many books present concepts in such an order that a concept *a* required to understand concept *b* is presented before *b* is presented, there are important situations where this approach is inappropriate. Also, some important teaching methodologies may require different kinds of orderings. Next, we briefly discuss a few examples of such methodologies.

Problem-Based Learning (PBL) is a teaching/learning method developed for medical students [4]. One of the fundamental principles is that the students need to discover on their own that they are missing a certain understanding that they have to acquire before they can solve the case at hand. This approach is not restricted to highly specialized medical students. The same method has been used in middle school science curricula where the students are expected to discover that, in order to find the solution, they need to learn more about a certain concept [5]. This may result in a trip to the library, checking a web site (which is not adaptive yet), or talk to some expert, e.g., a student's parent. Learning by Design (LBD) is another approach based on the design process that requires the students to discover the need to learn certain concepts [6].

Of course, this may not be the approach of choice if you have to teach mathematics or programming. It would be hard to imagine the students discover all those theorems. However, a final example from teaching user interface design at the college level shows that some instructors may quite often use approaches consistent with LBD.

Assume you are teaching the user interface design process and you are talking about the need for not just one but alternative approaches or prototypes during the earlier design iterations. One way to go from here is to ask the students which one of the approaches is the best. Quite soon, they realize that they need something like a set of criteria, actually, a set of quantifiable and measurable criteria. If, instead, the students are simply told in advance that measurable criteria are needed, they may not be clear about why they are needed.

In terms of prerequisites, these examples can be described as follows. A concept *a* is required to understand concept *b*. Before concept *a* is discussed, concept *b* is presented to the student who needs to discover the need to learn about *a*. Therefore, the student will learn about *a* and then be able to fully understand *b*. This approach, where applicable, has the advantage that the need of *a* for *b* is more clear to the student.

It is now easy to see that we talk about two different concepts when referring to prerequisites.

- A prerequisite is a mechanism to partially order web pages in an adaptive web site.

- A prerequisite is a pedagogical relationship between two concepts.

The next two sections will investigate both prerequisite concepts in more detail.

3.2 Prerequisites as Ordering Mechanism

If we consider the definition of a prerequisite as an ordering mechanism and assume that p_a is a prerequisite of p_x and also p_b is a prerequisite of p_x, it follows that the conjunction of p_a and p_b is a prerequisite of p_x. Although this approach covers many situations, it imposes an unnecessary constraint on what kinds of prerequisites can be defined.

However, extending the expressiveness of prerequisites comes at a cost. For instance, adding disjunctive prerequisites of the form "the disjunction 'p_a or p_b' is the prerequisite of p_x", where at least one of p_a and p_b need to be visited before p_x may be visited, requires computationally expensive algorithms [7]. These extensions can be useful, though. For example, it can be used to make sure that a student sees the use of a certain concept or feature, say of a programming language, in at least two examples. This constraint can be easily expressed as a disjunction of conjunctions. Another example where disjunctions come in handy is where the instructor has several examples to explain an issue in the lecture notes. He wants to select just one example and make sure he is not going over other, redundant examples. Again, conjunctions alone cannot express such a relationship easily, but a disjunction of conjunctions can. For a more elaborate discussion of disjunctions, see [7].

Adding full prepositional logic, that is, adding negation can cause even bigger problems with non-terminating updating algorithms resulting in difficult-to-define semantics [2]. In addition, replacing the visited/not-visited dichotomy of pages with continuous variables and some more or less arbitrary thresholds may make it more difficult to interpret the meaning of these variables and thus, of the prerequisites.

It is therefore of utmost importance to clearly define the semantics of the mechanisms and formalisms used.

3.3 Pedagogical Prerequisites

A pedagogical prerequisite states the relationship between two concepts with respect to a learner. If a concept *a* is required to understand concept *b*, then *a* is a prerequisite for *b*. Again, we claim that this is an incorrect overgeneralization. Let's look at an example.

Imagine you are a programmer and need a good random number generator to drive your simulation. You look up your web-based manual and find an entry "random number generator" in the index.

In the first scenario, we assume that the web-based system is *not* adaptive, thus all the hyperlinks are available to all users all the time. Clicking on the link in the index gets you to the page with the title "Random Number Generator" and the page mentions that the generator, Mersenne Twister, is fast and has a period of $2^{19937-1}$. Knowing what a period of a pseudo-random number generator is you use it in your program without any problems and the resulting simulation runs well.

In the second scenario, we use an adaptive version of the same manual. You will only see links to concepts you are ready to learn. Again, you click on the link in the index and the system understands that you want to learn the concept on that page, namely about the Mersenne Twister random number generator. So it provides you with the following list of recommended links: primitivity of characteristic polynomials, inversive-decimation method, resolution-wise lattice method, etc. Probably, you are not ready to learn about the intricate details of the random number generator called Mersenne Twister. If you study hard enough, you may be able to get to the actual random number generator page in a few weeks and run the simulation soon thereafter.

Obviously, we don't want to learn every pedagogical prerequisite even though it exists. Whether we want to learn a prerequisite *a* of *b* depends on why and for what purpose we need to learn about *b*. Do we need a deep understanding of *b*, do we need to be able to use *b* in non-standard situations, or are we just going to apply it in the most common cases?

This problem cannot be solved satisfactorily by just including those concepts that the user is required to learn. This would only work if every learner had to learn all the concepts in the web site. This is not a very interesting case for adaptive web sites that are supposed to cater to each learner individually. Consider an adaptive web-based manual. Most users will read parts of it only on demand and the random number generator example shows that the intended use of the concept influences what prerequisites are to be learned and which are not.

By definition of a prerequisite, a prerequisite that is not required is not a prerequisite. A potential approach may be using something like levels of understanding as proposed by Merrill's Component Display Theory [8]. In the context of adaptive hypermedia systems, this idea has also been mentioned by Murray [9]. Loosely based on Merrill's work, a concept could be used in various ways: remember, use, and understand. Remember: no need to learn the concept in any way, just being aware of its existence. Use, meaning applying the concept in normal situations: no need to understand the underlying principles of the concept. Understand: required to understand the underlying principles of the concept, thus, also being able to apply it in exceptional situations.

Of course, adding this kind of differentiation may complicate things and it may become less and less clear what the status of a specific prerequisite is. Furthermore, it is not clear who would specify the prerequisites for all the different situations, for instance, for remember, use, and understand. And finally, who would know whether the learner cares deeply about a concept or not? Maybe he really wants to know more about the Mersenne Twister, just because of its cool name.

3.4 Preconditions and Prerequisites

We have discussed two main problems with the use of prerequisites.

First, these are really two different concepts that happen to have the same name and look similar at their surface. As an anonymous reviewer has once suggested, we could call the prerequisites used as ordering mechanisms "preconditions." The pedagogical prerequisites should possibly be called exactly that, pedagogical prerequisites, to emphasize their purpose and origin. After all, in non-educational applications of AHS we still would need preconditions but probably no pedagogical prerequisites.

Second, a pedagogical prerequisite does not necessarily have to be learnt at all, which is not a newly discovered issue. However, it should be taken more seriously in many AHS.

It will be impossible to completely avoid having implicit design decisions slip in that contradict the underlying pedagogical model. We propose to clearly mark the separation between those parts (the set *P* for *P*edagogy) in the AHS that are related to pedagogy and those that are not (the set *NP*). Clearly, the intersection of *P* and *NP* must be empty. Furthermore, the parts in *NP* must *implement* those in *P*, that is, the pedagogical model of the AHS must be fully described by the elements in *NP*.

4. Conclusions

An educational system must make its underlying pedagogical method explicit. The notion of prerequisites shows that there is always the danger that some implicit assumptions inconsistent with the "official" approach may slip in.

With respect to AHS, keeping a separate pedagogical model is important. However, every design decision needs to be understood in the light of the pedagogical method employed. In practice, this can be very difficult if not impossible. Almost every design decision will have some impact on the learner. However, if we try to keep pedagogical parts and those implementing them as strictly apart as possible, we may reduce the number of harmful, implicit design decisions.

5. References

[1] R. J. Spiro, P. J. Feltovich, M. J. Jacobson, and R. L. Coulson, "Cognitive Flexibility, Constructivism, and Hypertext: Random Access Instruction for Advanced Knowledge Acquisition in Ill-Structured Domains," *Educational Technology*, vol. May, pp. 24-33, 1991.

[2] H. Wu, P. De Bra, and G.-J. Houben, "Adaptation Control in Adaptive Hypermedia Systems," presented at Adaptive Hypermedia and Adaptive Web-Based Systems, Trento, Italy, 2000.

[3] P. De Bra, G.-J. Houben, and H. Wu, "AHAM: a Dexter-based reference model for adaptive hypermedia," presented at Proceedings of the 10th ACM Conference on Hypertext and Hypermedia, 1999.

[4] H. S. Barrows, *How to Design a Problem Based Curriculum for the Preclinical Years*. New York, NY: Springer Verlag, 1985.

[5] R. Hübscher, C. E. Hmelo, N. H. Narayanan, M. Guzdial, and J. L. Kolodner, "McBAGEL: A Shared and Structured Electronic Workspace for Problem-Based Learning," presented at Second International Conference on the Learning Sciences, Evanston, IL, 1996.

[6] R. Hübscher, S. Puntambekar, and M. Guzdial, "A Scaffolded Learning Environment Supporting Learning and Design Activities," presented at AERA, Chicago, 1997.

[7] R. Hübscher, "Logically Optimal Curriculum Sequences for Adaptive Hypermedia Systems," presented at Adaptive Hypermedia and Adaptive Web-Based Systems, Trento, Italy, 2000.

[8] M. D. Merrill, "Component Display Theory," in *Instructional-Design Theories and Models: An Overview of their Current Status*, C. M. Reigeluth, Ed. Hillsdale, NJ: Erlbaum, 1983, pp. 279-333.

[9] T. Murray, "A Model for Distributed Curriculum: From Tutor-Centered to Topic-Centered Representations," presented at AIED-99 Workshop: Ontologies for Intelligent Educational System, Le Mans, France, 1999.