

개별화 학습 지원을 위한 학습객체 기본 구조 설계

홍지영⁰ 정영식 송기상

한국교원대학교 컴퓨터교육과

jipooh@chollian.net⁰, nurunso@hotmail.com, kssong@cc.knue.ac.kr

Design of LO's Basic Structure for supporting Individualized Learning

Jiyoung Hong⁰ Youngsik Jeong Kisang Song

Dept. of Computer Education, Korea National University of Education

요 약

e-Learning 콘텐츠 설계에 있어 객체지향기법에 근간한 학습객체 기본 설계에 많은 관심이 모아지고 있다. 학습객체는 기존의 콘텐츠가 하나의 커다란 덩어리로 이루어져 있어 동일한 내용에 관해서도 수많은 코스들이 생성되었던 재사용성의 문제를 해결하며 상호운용성, 접근성, 내구성 등의 잇점을 제시하고 있다. 이러한 학습객체는 레고모형에 비유되어 각각의 학습자마다 서로 다른 조합의 코스를 제공한다고 하지만, 현재의 시퀀싱된 형태는 CBT 수준의 분기수준에 머물러 있다. 본 연구에서는 개별화 학습을 지원할 수 있는 시퀀싱 설계를 위하여 학습객체 구조의 관점에서 접근하며, 이러한 학습 설계에 기초가 되는 학습객체의 기본 구조를 제안하고자 한다.

1. 서 론

e-Learning은 수많은 학습자를 대상으로 하기 때문에 개별화에 관한 고려가 필수적으로 요구된다. 그러나 현재의 대부분 e-Learning 콘텐츠는 학습자의 특성이나 지식을 염두에 두지 않은, 일반화된 집단을 목적으로 구성되고 있다. 즉, 교과서의 내용을 단순한 항목순으로 그대로 웹 상에 올려놓은 형태이다. 개별화에 관한 고려는 이미 이전부터 있어 왔지만 실제적인 구현이 부족했던 이유는 하나의 커다란 덩어리로 구성되어 있는 고정된 콘텐츠의 구조 때문이다.

이러한 문제의 관점에서 새롭게 대두된 콘텐츠 설계 요소가 바로 학습객체이다. 레고 모형에 비유되는 학습객체는 재사용 가능한 조각들을 학습자의 특성과 요구에 맞게 조합함으로써 개별화된 학습을 제공할 수 있는 가능성을 제시해 주고 있다. 그러나 현재 SCORM의 시퀀싱 기법에서는 재사용성의 관점에서 동일한 객체를 활용한 다양한 수업 설계를 보여주지는 있으나, 기존 CBT 수준의 제한적인 분기 형태를 지니고 있으며 진정한 개별화의 측면은 부족하다고 볼 수 있다.

본 연구에서는 학습객체 기본 콘텐츠 설계 프레임워크에서 개별화 학습을 설계하기 위하여 학습객체 관점에서의 접근을 통해 해결책을 찾아보고자 한다.

2. 개별화 학습과 학습객체

e-Learning의 2가지 근본적인 잇점은 시간과 거리의 장벽을 제거한다는 점, 그리고 학습자의 경험을 개별화 한다고 하는 것이다[1]. 그러나 개별화 학습을 제공하는 데 있어 현재의 일반적인 웹 콘텐츠는 다음과 같은 문제점을 갖는다.

그림 1은 2001, 2002년의 전국 교육용 소프트웨어 공모전에 출품된 코스웨어 중에서 불대수를 다루고 있는 웹 코스웨어를 조사한 것이다. 현재 고등학교 단계의 교과에서

불대수가 다루어지고 있는 교과는 그림에서와 같으며, 동일한 불대수의 학습내용 콘텐츠가 여러 교과 각 단원에서 중복되어 개발되고 있음을 알 수 있다. 하나의 코스웨어는 작은 수정을 거친 후 다른 교과에서도 충분히 재활용될 수 있음에도 불구하고 비슷한 내용의 코스웨어가 불필요하게 중복되어 제작되는 문제점이 발생한다.

이와 같이 현재의 콘텐츠 설계 형태에서는 학습 리소스 내부에 시퀀싱을 위한 코드를 포함하기 때문에, 각각의 리소스를 재사용하기 어려울 뿐만 아니라, 전체의 콘텐츠에 고정된 구조를 변경하기 위해서는 전체의 콘텐츠를 수정해야 하는 어려움이 있다. 이러한 방식은 모든 학생들로 하여금 동일한 선형의 순서를 따라 학습하도록 만든다. 일찍이 Crowder[2, 재인용]는 각 학생의 수업 순서는 그 학생의 응답에 대한 기록에 기초해야 하며, 수업은 교수과정을 통하여 여러 경로를 제공해야 한다고 했으나 이러한 아이디어

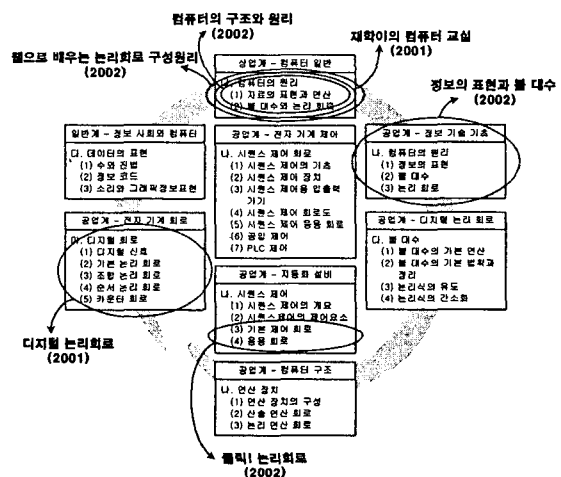


그림 1 현 코스웨어의 중복 문제

를 구현하고자 하는 Crowder의 시도는 실패했다. 프로그램이 다루기 어려울 정도로 커졌기 때문이다. 콘텐츠가 커다란 하나의 덩어리로 되어 있기 때문에 적응적인 학습을 위해서 여러 개의 코스를 만든다는 것은 불가능할 수밖에 없다.

최근 들어 e-Learning 콘텐츠 설계에서는 학습객체(Learning Object)에 기반한 콘텐츠의 설계와 제작에 많은 관심이 모아지고 있다. 레고 모형에 비유되는, 작은 규모로 제작되어 재사용될 수 있는 학습객체의 개념은 수정과 변형이 쉽지 않았던 기존의 코스웨어 개발에 대한 대체로써 인식되고 있다[3]. 학습 콘텐츠의 재사용성(reusability)과 상호운용성(interoperability)을 표준화 하기 위한 스펙인 SCORM(Sharable Content Object Model)에서는 표준화된 재사용 가능한 학습객체를 SCO(Sharable Content Object)로 보고 있다. SCO는 독립적인 학습단위이며, 패키지를 생성하기 위하여 블록(레고)을 구축하는 것으로써 사용될 수 있다. SCO는 수많은 다양한 요구를 충족시키기 위해 여러 가지 방법으로 재구성될 수 있으며, 여기에서 개별화, 적응형 학습의 가능성을 엿볼 수 있다.

3. 학습객체 기반 코스 설계의 문제점

학습객체가 개별화 학습을 위한 가능성을 제시하고 있으나 코스를 설계하는 데 있어 아래와 같은 문제점을 생각할 수 있다.

첫째, 일반적인 웹 콘텐츠에서처럼 학습객체를 기반으로 하는 설계에서도 획일화(one-size-fits-all)의 문제가 제기된다. 교사는 관련된 학습객체를 확인하고 그것을 코스로 구성해 나갈 때 일반적인 학습에 관해 생각을 하게 된다. 하지만 그 학습의 학생들은 다양한 관심사, 지식, 학습배경, 그리고 다양한 학습 스타일을 갖는다. 교사가 주의 깊게 선택한 몇가지 내용이 어떤 학생들에게는 쓸모 없는 것을 수 있으며 단지 그들을 출뜨릴 수 있다. 심지어 특정 학습자들에게 중요한 몇가지 자료들이 선택되지 않을 수도 있다. 어떠한 범주의 학생들에게 이익을 주는 내용 조직은 다른 범주의 학생들에게는 장애물로 작용할 수도 있다.

둘째, 기술적 관점에서 문제점을 살펴보면 다음과 같다. 현재의 SCORM은 그림 2에서와 같이 제한적인 분기를 통한 교정학습을 제공한다. SCORM에서 말하는 학습목표(objectives)라고 하는 것은 LMS로 하여금 SCO 간의 상태값들을 연결시키도록 하는 전역변수를 말한다[3]. 이 학습목표 변수에는 개별적인 학습목표에서의 점수, 합격/불합격 상태, 완료상태, 그리고 각 과제에서 소비한 시간 등

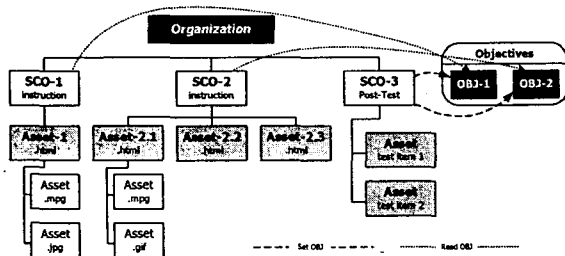


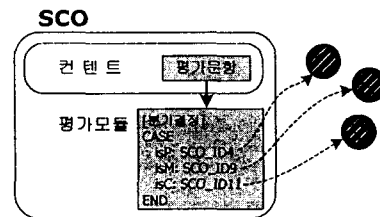
그림 2 SCORM에서의 교정분기

단순한 값들이 저장되어 참조된다. 즉, 그림에서 보면 콘텐츠 구조를 구성하고 시퀀싱을 하는 패키징 파일(manifest) 파일 안에 각 SCO가 참조하는 전역변수인 학습목표 ID를 기록하여 콘텐츠를 담고 있는 SCO와 평가문항을 담고 있는 SCO 간의 맵(map)을 구성하게 된다. 만일 SCO-3의 평가에서 OBJ-1이 'failed' 되었다면, 그 학습목표와 연결되어 있는 SCO-1이 교정학습으로서 학습자에게 제공된다. 그러나 SCORM의 현재 버전에서 LMS는 단지 그 평가SCO 수준에서만 추적이 가능하다. 때문에 만일 10개의 항목을 갖는 평가를 만들고 특정 평가문항에 관한 학습자의 반응을 알고자 한다면, 하나의 aggregation 내부에 분리된 SCO로서 각각의 평가 아이템을 만들어야 한다[3,4]. 그러므로, 개별화 학습을 위하여 각 지식모듈 간의 관련성을 고려한 분기를 위해서는 각각의 SCO를 고려한 개별화 학습 요소별 평가SCO가 매번 뒤따라야 하며, 복잡한 참조 연결이 설계되어야 하는 문제가 발생한다.

4. 개별화 학습 지원을 위한 학습객체 설계

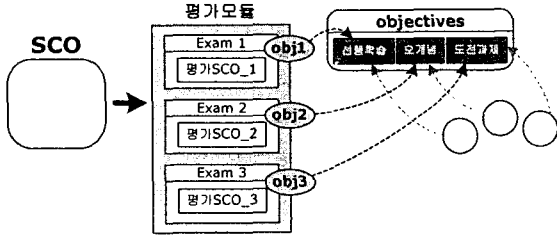
학습객체 기반의 코스에서 개별화 학습을 지원하기 위해 몇가지 시나리오를 통해 해결방법을 찾아나가고자 한다. 각각의 SCO를 연결하여 코스를 구성하여 나갈 때, 기존의 단원순 항목순이 아닌, 학습객체의 '지식 간 연관성'을 고려하여 연결하고자 하는 것이다. 이러한 연관 요소로는 선행학습이 부족한 경우, 오개념을 갖는 경우, 도전과제가 요구되는 경우.. 등으로 생각해 볼 수 있다. 아래 그림에서의 'P'는 선행학습, 'M'은 오개념, 'C'는 도전과제를 나타낸다.

(1) CASE 1



첫번째 경우로 생각해 볼 수 있는 형태는 위의 그림과 같다. 현재의 SCO에서 선행학습에 해당되는 SCO는 무엇이고, 오개념을 가졌을 때 해당되는 SCO는 무엇이고, .. 등으로 현재의 SCO에서 링크되는 학습객체들이 고정되어 있는 경우로서 SCO 안에 평가모듈을 포함하는 구조이다. 학습객체 개발자는 SCO를 설계하는 단계에서 개별화 학습을 고려한 SCO의 ID를 명세한다. 이를테면 IEEE LOM 메타데이터의 '7. relation' 항목에서 관련성 있는 객체의 ID를 적어주는 것과 비슷한 의미로 생각할 수 있다. 이러한 형태는 관련성 있는 SCO들을 명시적으로 기술함으로써 구현 문제를 간단히 해결할 수 있으나, 이것은 하나의 SCO에서 외부로의 고정된 연결을 포함하게 된다. 즉, "실행 시 하나의 SCO는 다른 SCO를 호출해서는 안된다"고 하는 설계 규칙을 위반하게 되는 것이며, 재사용을 위해서 독립적이어야 한다고 하는 학습객체의 기본 규칙에 위배된다.

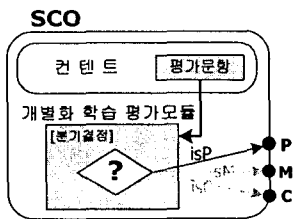
(2) CASE 2



독립적인 SCO를 유지하기 위하여 CASE 2에서는 평가모듈을 분리해 보았다. 이러한 구조는 그림 2에서와 같이 현재 SCORM에서 이루어지고 있는 교정분기 구조와 유사한데, 각각의 평가 문항은 학습목표(objective)와 연결이 되어 있으며, 각각의 학습목표를 통과하지 못했을 경우, 학습목표와 연결되어 있는 SCO로 다시 분기하여 학습자에게 제공하는 형태이다. 즉, 각각의 평가SCO에 연결된 학습목표의 달성 유무에 따라 그 학습목표와 링크되어 있는 교정 학습 내용을 제시해 주는 방식과 비슷한 형태라고 볼 수 있다.

이러한 구조에서 개별화 학습을 설계하기 위해서는 각각의 SCO마다 다음으로의 분기를 위하여(선행학습이 부족한 경우, 오개념을 갖는 경우, 심화학습을 제공하는 경우 등) 현재의 SCO를 끝낸 후 반드시 평가모듈을 거치게 되며, 물리적으로는 분리되어 있으나 각 SCO에서의 개별화 학습 항목을 평가할 수 있는 각각의 평가문항을 갖는 종속적인 평가모듈이 매 SCO 뒤에 반드시 뒤따라야 한다.

(3) CASE 3



CASE 1과 CASE 2의 문제점을 해결하기 위하여 본 연구에서는 CASE 3의 학습객체 기본구조를 제안하고자 한다.

여기에서의 학습객체 구조는 평가모듈을 포함하고 있으나, CASE 1에서와 같이 다른 SCO로의 분기를 위해서 관련성으로 연결되어 있는 SCO들의 ID를 명시하지는 않는다. 즉, '독립적'이라고 하는 학습객체의 속성을 유지한다. 그러나 그 평가모듈을 통하여 SCO 안에서 평가가 이루어 지든, 아니면 다른 방법으로 체크를 하든간에 현재 그 학습객체 안에서 분기를 위한 판단이 이루어진다. 간단한 진행 과정을 살펴보면 다음과 같다.

첫째, 학습객체 개발자는 그 학습객체에서의 선행학습은 무엇이며, 거기에서 발생할 수 있는 오개념은 무엇인지, 그리고 잘 하는 학생에게 던져줄 수 있는 도전과제는 무엇인지를 '학습객체 개발단계'에서 명시하게 되며, 각각의 개별화 학습 항목을 판단할 수 있는 평가문항을 생성한다.

둘째, 교수설계자는 코스를 설계하는 과정에서 각

gateway에 해당하는 객체를 연결한다. 이것은 학습객체 개발자가 의도한 내용과 달라질 수 있는데, 즉 학습객체 개발자는 선행학습을 SCO-a로 생각했는데 교수설계자는 SCO-b로 생각하여 연결할 수 있다.

셋째, 학습하는 과정에서 어디로 분기를 하여야 할 것인가가 결정된다. 예를 들어, 선행학습(Prerequisite) : isP = true; 오개념(Misconception): isM = false;와 같이 만일 학습자가 이 학습객체 부분에서 선행학습이 부족하다고 판단되면, 위 그림에서와 같이 선행학습(P) 게이트웨이의 링크가 활성화 되며 그 학습객체의 '선행학습'에 해당하는 학습객체를 던져주게 된다.

이렇게 SCO 안에 평가모듈을 포함하며 분기를 위한 판단이 SCO 내부에서 이루어지므로 CASE 1과 비교해 볼 때 '독립된 객체'임을 알 수 있으며, CASE 2와 비교해보면 부가적인 평가모듈-SCO가 필요 없게 되므로 설계와 개발에 있어 부담을 줄일 수 있다. 정상경로는 존재한다고 보며, '판단'은 평가모듈을 사용하거나 학습자의 학습 추적에 근거할 수 있다.

5. 결론

본 연구에서는 지식 간의 관련성을 고려한 개별화 학습을 지원하기 위하여 그 기본 구성요소가 되는 학습객체의 기본 구조를 생각해 보았다. 웹 상에서 이루어진다 하더라도 수업이라는 것은 일반 정보의 제공 형태와는 구별이 되어야 한다. 수업, 학습이라고 하는 것은 지식 간의 관련성이 고려되어 제시되어야 하며, 각각의 학습자에게 맞는 학습의 경로를 제공함으로써 개별화될 수 있어야 할 것이다.

여기에서 살펴본 학습객체의 구조는 개략적인 형태에 불과하며 이러한 학습객체의 실제적인 설계를 위해 고민해야 할 많은 항목들이 존재한다. 첫째는 판단을 어떻게 할 것인가, 평가문항을 통해서 인가 아니면 LMS에 제공되는 학습자의 정보를 통한 추적인가, 둘째는 지식 간 고려할 수 있는 관련 요소들은 구체적으로 어떠한 것들이 있는가 하는 요소 추출의 문제이다. 또한, 다른 학습객체로 이동했다가 다시 복귀하는 문제나 평가모듈에서의 판단이 여러 개인 경우, 즉 선행학습도 부족하고 동시에 오개념도 부족한 경우 등도 해결해야 할 문제들이다.

6. 참고문헌

- [1] P. Drucker, "Need to Know: Integrating e-Learning with High Velocity Value Chains," A Delphi Group White Paper, 2000.
- [2] E. R. Steinberg, "Computer-Assisted Instruction," Lawrence Erlbaum Associates, 2002.
- [3] Carnegie Mellon Learning Systems Architectures Lab., "SCORM Best Practices Guide for Content Developers, Ver. 1.8," Carnegie Mellon University, 2002.
- [4] IMS Global Learning Consortium, Inc., "IMS Simple Sequencing Best Practice and Implementation Guide, Ver. 1.0 Final Specification," 2003.