

# 학습과제 유형별 유의미 연결을 통한 학습객체 기반 개별화 학습 시스템

홍지영<sup>†</sup> · 송기상<sup>††</sup>

## 요 약

개별화되고 적응화된 코스를 생성하기 위해서는 학습객체가 논리적 연관성을 가지고 연결되어 있는 기반구조를 갖추어야 한다. 이러한 학습객체간의 논리적 연관성, 그리고 개별 학습자를 고려한 다양한 링크를 통하여 학습의 각 시점에서 각각의 학습자는 서로 다른 학습경로를 제공받을 수 있게 된다. 본 연구는 학습과제 유형별 유의미 연결을 고려하여 학습객체 기반의 개별화 학습 시스템 구조를 설계하는데 목적이 있으며, 이를 위해 '관련성 요소 추출에 관한 연구', '학습목표 맵 구성에 관한 연구', '학습자의 인지상태 판단에 관한 연구'를 수행하였다. 학습객체 기반의 코스 설계가 단지 무의미한 객체들의 집합이라는 비판이 대두되는 시점에서, 본 연구의 학습객체간 관련성을 고려한 개별화 학습 시스템 모형 연구는 e-Learning 안에 유의미한 학습과 진정한 교육을 담고자 하는 시도가 될 것이다.

키워드 : 개별화 학습, 학습객체, e-Learning

## Individualized Learning System based on Learning Object, through Semantic Sequencing by Learning Task Types

Ji-Young Hong<sup>†</sup> · Ki-Sang Song<sup>††</sup>

## ABSTRACT

To generate individualized and adaptive course, it's required to have the foundation structure in which learning objects are connected with each other with logical relevances. Each learner can have peculiar learning path at each point of time of learning through the logical relevancy between those learning objects and various links, considering individual learner. The purpose of this study is to design a learning object-basis individualized learning system structure, considering semantic sequencing by learning task types. It is our understanding that the individualized learning system design model of this study, considering the relevancy between learning objects, can be a fresh trial to accommodate semantic learning and true educational spirits in e-Learning at this point of time when criticism, such as the learning object based course design is simply a collection of meaningless objects, etc., is becoming influential.

Keywords : Individualized Learning System, Learning Object, e-Learning

### 1. 서 론

e-Learning은 수많은 학습자를 대상으로 하기 때문에 개별화에 관한 고려가 필수적으로 요구된다. 그러나 현재 대부분의 e-Learning 콘텐츠는 학습자의 특성이나 지식을 염두에 두지 않은, 일

<sup>†</sup> 정 회 원: 한국교육개발원 교육통계정보센터 연구원  
<sup>††</sup> 총신회원: 한국교원대학교 컴퓨터교육과교수(교신저자)  
논문접수: 2004년 9월 30일, 심사완료: 2004년 11월 22일

반화된 집단을 목적으로 구성되고 있다. 즉, 교과서의 내용을 단원순 항목순으로 그대로 웹 상에 올려놓은 형태이다. 개별화에 관한 고려는 이미 이전부터 있어왔지만 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 학습객체를 기반으로 하는 SCORM 스펙[10]의 시퀀싱 기법을 적용한 코스[11]는 '각각의 학습객체들이 재사용 가능한 단위'라는 것을 제외하고는 기존의 코스와 많이 닮아 있다. 즉, 학습의 흐름은 선형적인 형태를 유지하며 크기는 '장-절-항'의 단원 중심의 일반적인 교과서 구조를 띠고 있다. 물론 교과서에서의 단원이라는 개념은 학습내용의 위계와 관련성을 고려하여 학습내용을 조직화한 것이라 볼 수 있지만, 학습의 어떠한 한 시점에서 각각의 학습자가 서로 다른 학습과제를 제공받아야 하는 개별화 학습 관점에서 생각해 본다면 기존의 트리 형태의 선형적인 코스 설계는 수정되어야 한다.

본 연구는 학습과제를 유형별로 구분하고, 그것들의 유의미한 연결을 고려하여 학습객체 기반의 개별화 학습 시스템 구조를 설계하는데 목적이 있으며, 이를 위해 ITS(Intelligent Tutoring System)의 지능형 기법 아이디어를 접목하고자 한다.

우선 ITS의 한 분야인 적응형 하이퍼미디어 시스템(AHS; Adaptive Hypermedia System)에서의 시사점을 분석하고, 다음의 3가지 연구문제를 중심으로 연구를 수행하고자 한다. 첫째, 교사 관점에서 학습객체를 조합하여 코스를 구성할 때 어떻게 구성할 것인가에 관련된 문제로서, 학습과제를 유의미하게 연결하기 위한 '관련성 요소 추출에 관한 연구'이다. 둘째, 학습자가 학습할 준비가 되어 있는가를 판단하고 학습목표 달성 여부를 관리하기 위한 '학습목표 맵 구성에 관한 연구'이다. 셋째, 생성된 코스를 학습자에게 적용할 때 어떻게 적용할 것인가에 관한 학습자 관점

의 문제로서, 수준별 학습과제를 제시하기 위한 '학습자의 인지상태 판단에 관한 연구'이다.

## 2. 적응형 하이퍼미디어 시스템의 시사점

적응형 하이퍼미디어 시스템에서 적응형 학습 환경 구축을 위한 시사점은 다음과 같다[2, 17, 18].

첫째, 적응형 하이퍼미디어 시스템은 일반적인 ITS 시스템 구성요소인 도메인모델(Domain Model), 학습자모델(User Model), 적응모델(Adaptation Model), 적응엔진(Adaptive Engine)의 형태를 갖추고 있다. NetCoach의 '문서공간'에 해당되는 도메인모델은 개념들 간의 관련성을 통하여 정보가 어떻게 조직되어 있으며 어떻게 연결되어 있는지를 기술한다. 학습자모델은 하이퍼미디어 공간을 탐색하는 과정에서 학습자의 특성과 선호도 등을 유지하며, 이러한 학습자모델에 따라 도메인 지식을 연결해서 적응화된 환경을 구성하는 규칙들은 적응모델에 포함된다. 적응형 엔진은 실제로 하이퍼미디어 콘텐츠와 링크를 조작하여 적응화된 결과물을 생성하게 된다. 이와 비교해 볼 때, SCORM에서 채택한 IMS SS(Simple Sequencing)는 명백한 도메인모델이나 학습자모델을 포함하지는 않는다. 대신 시퀀싱 정의모델(Sequencing Definition Model), 추적모델(Tracking Model), 학습활동 상태모델(Activity State Model)을 통하여 학습진행을 위한 규칙을 기술하고 학습자와 콘텐츠와의 상호작용 결과를 체크하고 학습활동단위와 관련된 데이터를 저장한다. 추적모델 내부에 학습자모델을 포함하고 있다고 볼 수 있지만 적응형 하이퍼미디어 시스템과 비교하면 극히 단순한 형태이며 여기에서는 학습진행상태만을 기록하고 학습자의 선호도나 학습스타일은 구체적으로 다루지 않는다. SCORM에서는 도메인모델 대신 시퀀싱 규칙을 사용하여 교수설계자가 코스를 생성한다.

둘째, 적응형 하이퍼미디어 시스템은 적응화를 제공하기 위하여 몇가지 적응형 기법을 사용하고 있다. 일반적인 적응 기법으로는 적응적 탐색 지

원, 적응적 콘텐츠 제시, 적응적 콘텐츠 선택 등이 있다. 적응적 탐색 지원 기법으로는 적응적 링크 정렬, 적응적 링크 감추기, 적응적 링크 주석화, 맵 적응화, 직접 안내 등이 있으며, 적응적 콘텐츠 제시 기법에는 적응적 텍스트/멀티미디어 제시 등이 있다. SCORM은 시퀀싱 과정을 시퀀싱 규칙과 롤업 규칙을 이용하여 제어하는데, 시퀀싱 규칙은 학습활동 트리의 각 학습활동단위(activity)에서의 시퀀싱 동작을 기술하는 방법을 제시해주며 롤업은 지식단의 학습목표 수행여부가 부모단에도 영향을 미치게 하는 방법들을 명세하는 것이다.

셋째, 학습제공에 있어 기본 지식베이스를 구성하는 문서공간이 다양한 연결관계로 이루어져 있다는 것이다. 학습자에게 제시되는 문서 개념의 순서는 선행-관계로 연결되어 있으며, 어떤 문서 A를 학습했을 때 문서 B도 학습되는 관계로 추론되는 추론-관계로 연결되어 있다면 학습자는 이미 학습한 개념을 또다시 중복해서 학습

하게 되는 불필요한 노력의 낭비를 피하게 된다. 이와 같은 문서공간에서의 문서들의 연결 형태는 NetCoach 시스템 뿐만 아니라 다른 적응형 하이퍼미디어 시스템에서도 찾아볼 수 있는데, <표 1>과 같다.

### 3. 학습객체 기반 개별화학습 모형설계

학습자에게 개별화된 학습내용을 제시하고자 하는 '시퀀싱'의 관점에서, 본 연구는 현재 트리 기반의 선형적인 시퀀싱을 제공하는 SCORM에 ITS의 지능형 기법 아이디어를 접목하고자 한다.

학습객체 기반 개별화 학습 시스템의 프레임워크 설계를 위하여, 학습과제를 유의미하게 연결하기 위한 '관련성 요소 추출'에 관한 연구, 학습자가 학습할 준비가 되어 있는가를 판단하고 학습목표 달성 여부를 관리하기 위한 '학습목표 맵 구성'에 관한 연구, 수준별 학습과제를 제시하기 위한 학습자의 '인지상태 판단'에 관한 연구문제를 선정

<표 1> 적응형 하이퍼미디어 시스템의 문서공간에서의 연결관계

시스템	문서공간에서의 연결관계
NetCoach	$preq(D_i, D_j)$ (선행학습 지식) $infer(D_i, D_j)$ (문서 $D_i$ 를 학습했을 때 학습되었을 것이라 추론되는 문서 $D_j$ ) $succ(D_i, D_j)$ (문서제시 순서) $part\_of(D_i, D_j)$ (단원의 구조) $terminal\_flag(D_i)$ (종속 문서가 존재하지 않는 말단 문서임을 표시) $criterion(D_i, Value)$ (문서 $D_i$ 학습을 완료하는데 반드시 필요한 평가항목의 수) $test\_assignment(D_i, X), X \in \{Testgroup, Testitem\}$ (문서를 평가그룹이나 평가항목과 연결)
ELM-ART II	$preq(D_i, D_j)$ (선행학습 지식) $out(D_i, D_j)$ (문서 $D_i$ 를 학습했을 때 학습되었을 것이라 추론되는 문서 $D_j$ ) $related(D_i, D_j)$ (저작자가 정의한 문서들 간의 관계) $successor(D_i, D_j)$ (문서제시 순서) $part\_of(D_i, D_j)$ (단원의 구조) $terminal\_flag(D_i)$ (종속 문서가 존재하지 않는 말단 문서임을 표시) $test\_assignment(D_i, X), X \in \{Testgroup, Testitem\}$ (문서를 평가슬롯이나 평가항목과 연결)
InterBook	$preq(D_i, C_j)$ (선행학습 지식) $out(D_i, C_j)$ (문서 $D_i$ 를 학습했을 때 학습되었을 것이라 추론되는 문서 $D_j$ ) $succ(D_i, D_j)$ (문서제시 순서) $terminal\_flag(D_i)$ (더이상의 종속문서가 존재하지 않음을 표시) $part\_of(D_i, D_j)$ (단원의 구조)
KBS Hyperbook	$keyword(D_i, C_j)$ (문서에서의 주요 개념) $depends(C_i, C_j)$ (개념 간 학습 의존성) $role(D_i, X), X \in \{Course, Goal, Lecture, Example, etc.\}$ (문서 $D_i$ 유형 구분) $role(C_i, X), X \in \{Introduction, Concept\}$ (개념 $C_i$ 유형)

하였다.

### 3.1 관련성 요소 추출

본 연구에서 제안하는 시스템은 '교사 관점'과 '학습자 관점'으로 나누어 볼 수 있는데, 관련성 요소 추출에 관한 연구는 교사 관점에서 학습객체를 조합하여 코스를 구성할 때 '어떻게 구성할 것인가'에 관련된 문제이다. 독립적인 학습객체 단위의 메타데이터에서 벗어나 학습과제인 학습객체를 연결하는 '링크'에 유의미한 속성을 추가하고자 하는 것이다. 이를 위하여 학습과제를 유형별로 구분하여 이러한 학습과제 유형을 링크의 속성으로 적용하고자 하였으며, '학습소개/선행/후속/연습/예시/도전/평가'의 7가지 관련성 요소를 추출하였다. 이러한 관련성 요소를 이용하여 생성된 코스에서 학습의 주된 흐름은 '후속' 링크로 연결되는 학습객체 리스트이며, 이러한 주 흐름에서의 학습객체는 학습목표 달성 여부를 체크하기 위하여 각각 평가객체를 반드시 수반하도록 하였다. 이러한 모델은 선행학습이 부족한 학습자에게는 '선행' 링크로 연결된 학습과제를 제시하고 학업 성취도가 높은 학습자에게는 '도전' 링크로 연결된 학습과제를 제시하는 등 개별화되고 적용화된 학습 제공을 가능하게 하는 기반 구조를 생성하게 된다.

### 3.2 학습목표 맵 구성

일반적인 코스웨어에서는 선행학습이 제대로 되어있는가 하는 학습자의 '준비도' 판단을 위하여 '사전평가'를 실시한다. 제안 모델에서는 학습목표 맵을 이용하여 학습자의 학습 여부를 체크하고자 하였다. 학습객체가 관련성 링크를 통하여 유의미한 연결로 코스를 구성하였다면, 적절한 판단을 통하여 학습자의 경로를 개별화하여야 한다. 여기에서 가장 기본이 되는 것은 '학습자가 다음으로 어떤 학습을 진행해야 하는가'에 관한 근거를 제시할 수 있는 '학습목표 맵'이라고 볼 수 있다. 이러한 학습목표 맵을 참고하여 선행학습이 이루어지지 않은 학습자에게는 '선행' 링크로 연결된 학습과제를 제시하며, 선행학습 준비

가 되어 있는 학습자에게는 해당 학습과제를 제시하도록 하였다. 본 연구에서는 학습목표 요소를 '지식/적용/문제해결'로 구분하여 학습목표 맵을 구성하도록 하였으며, 학습목표 맵의 어느 요소와 관련이 있는지를 명시하도록 하였다. 이로써 어떤 학습과제 단위에서 학습자는 그 부분의 지식만을 갖고 있는지, 적용할 수 있는 능력을 지니고 있는지, 실제 상황으로의 문제해결 능력을 획득했는지를 판단하게 된다.

### 3.3 인지상태 판단

학습자의 인지상태 판단은 시스템의 '학습자 관점'으로서, 생성된 코스를 학습자에게 적용할 때 '어떻게 적용할 것인가'에 관한 문제이다. 제안모델의 학습자모델에서는 적용적으로 다음 학습과제를 선택하고 전달하기 위하여 2가지 접근을 수행하는데, 첫째가 학습목표 맵의 이용이다. 학습자들은 각자 선택한 코스의 교과 학습목표 맵을 갖게 되며, 평가를 통해 각 학습목표의 달성여부를 체크하게 된다. 이러한 학습목표 맵은 다음 코스 학습 시에도 적용되어 학습자가 이전에 학습한 개념은 건너뛸 수 있는 근거를 제시해주게 된다. 둘째는 학습자모델 안에서 인지상태를 판단하는 과정이다. 학습자의 학습수행 관찰, 평가점수 결과 등 학습수행 관련 데이터를 통하여 학습자의 인지상태가 현재 어느 수준인지를 동적으로 체크하도록 하였으며, 학습자의 수준에 맞는 적절한 난이도 단계를 지닌 수준별 학습과제를 제시하기 위하여 학습자의 수준을 판단하는 적용 규칙을 생성하였다.

## 4. 시스템 구성: FOL(First-Order Logic) 표현

FOL(First-Order Logic)의 논리적 표현[17, 18]을 통하여 제안모델의 시스템 구성을 표현해보았다. 본 제안모델을 ILS(Individualized Learning System)로 명명하여 표현한 식은 다음과 같다. 시스템은 문서공간, 학습자모델, 상호작용 관찰, 적용규칙 항목으로 구성되어 있다.

$ILS = (DOCS_{ILS}, UM_{ILS}, OBJ_{ILS}, AC_{ILS})$   
 ·  $DOCS(DOCUMENT\ Space)$  : 문서 공간  
 ·  $UM(User\ Model)$  : 학습자모델  
 ·  $OBS(OBSERVATIONS)$  : 상호작용 관찰  
 ·  $AC(Adaptation\ Component)$  : 적용 규칙

(1) 문서공간(Document Space)

제안모델의 문서공간은 코스(C; Course), 각각의 독립적 학습객체인 학습과제(LT; Learning Tasks), 평가객체(TO; Test Object), 학습목표(OBJ; Objective)로 구성된다.

$C_1, \dots, C_k, LT_1, \dots, LT_l, TO_1, \dots, TO_m, OBJ_1, \dots, OBJ_n$

하나의 코스 안에서 학습과제들은 ‘학습소개(intro)’, ‘선행(preq)’, ‘후속(succ)’, ‘연습(prct)’, ‘예시(exam)’, ‘도전(chal)’ 관계로 연결되어 있다.

$intro(LT_i, LT_j)$  for certain  $LT_i \neq LT_j$ ,  
 $preq(LT_i, LT_j)$  for certain  $LT_i \neq LT_j$ ,  
 $succ(LT_i, LT_j)$  for certain  $LT_i \neq LT_j$ ,  
 $prct(LT_i, LT_j)$  for certain  $LT_i \neq LT_j$ ,  
 $exam(LT_i, LT_j)$  for certain  $LT_i \neq LT_j$ ,  
 $chal(LT_i, LT_j)$  for certain  $LT_i \neq LT_j$

각 평가객체는 학습목표 맵과 연결되어 있으며, 학습목표 맵에서는 학습목표 유형을 ‘지식’, ‘적용’, ‘문제해결’로 구분하여 학습자가 어떠한 부분의 학습과제에 관하여 어떠한 유형의 학습목표를 획득하였는지를 판단하게 된다.

$obj\_assignment(TO_i, OBJ_j(t))$  for certain  $TO_i$  and  $OBJ_j$

$t(type)=K(knowledge), A(apply)$   
 $,PS(problem\ solving)$

본 시스템은 ‘후속’ 관계로 연결되는 문서와 ‘선행’ 관계로 연결되는 문서에는 평가객체를 필수적으로 할당하도록 하여 주된 학습흐름에서의 학습목표 달성여부를 체크하도록 하였다. 평가객체인 TO의 인수는 학습목표 맵상에서 참조하는 학습목표(OBJ; Objective), ‘상/중/하’의 값을 갖는 평가객체 난이도(L; Level), 그리고 합격/불합격 판단의 기준이 되는 교사가 지정하는 기준점수(x)를 갖는다. 평가객체 난이도는 학습자의 수준에 따른 평가를 제공하기 위하여 사용되며, 학습자의 평가 수행점수가 교사가 명시한 기준값을 넘으면 그 평가객체에 연결되는 학습목표는

‘passed’로 체크되고 그렇지 않은 경우는 ‘failed’로 체크된다.

$\forall LT_j \forall TO_i$   
 $preq(LT_i, LT_j) \vee succ(LT_k, LT_j)$   
 $\Rightarrow test\_assignment(LT_j, TO_i(OBJ_m(t), L, x))$  for certain  $LT_j$  and  $TO_i$   
 $,L('H':high, 'M':medium, 'L':low)$   
 $,x: threshold\ score$

(2) 관찰(Observations)

ILS에서의 관찰은 크게 2개의 계층으로 구분하며, 이러한 관찰값은 학습자 모델에 수집된다. 첫 번째 계층은 어떠한 학습과제 페이지를 학습자가 방문했는지를 표시한다. 방문했다는 것은 학습목표의 달성여부와 상관없이 학습자가 학습과제를 학습하고 평가객체가 연결되어 있는 경우 평가까지 끝냈음을 나타낸다.

$obs(LT_j, U_i\ Visited)$  for certain  $LT_j, U_i$

두 번째 계층은 학습자가 후속이나 선행관계로 연결된 문서에 할당된 평가객체를 수행했는지를 표시한다. Worked\_testitem은 학습자가 평가객체에 접근했음을 표시하며, Solved\_testitem은 학습자가 평가를 수행하고 완료하여 y라는 점수를 획득하였음을 나타낸다.

$obs(TO_k, U_i, Worked\_testitem)$  for certain  $TO_k, U_i$ , and  
 $obs(TO_k, U_i, Solved\_testitem(y))$  for certain  $TO_k, U_i$   
 $, y: 학습자의\ 평가점수$

(3) 학습자모델(User Model)

ILS에서의 학습자모델은 크게 3가지 기능으로 나누어 생각할 수 있는데, 코스를 완료하였음을 표시하는 기능과 평가를 통한 학습목표 달성을 체크하는 기능, 그리고 학습자의 학습수준값을 처리하는 기능이다.

코스를 완료한다는 것은 ‘후속’ 관계로 연결된 주-학습과제를 방문하였음을 의미한다. 주-학습과제에 할당된 평가객체의 학습목표 달성여부는 관계없이 주-학습과제를 모두 방문하였으면 코스를 완료한 것으로 처리한다. ‘후속’ 관계로 연결된 학습과제 리스트를 생성하는 과정은 다음과 같다.

$main\_learning\_tasks(C_j) = []$   
 $\forall LT_k \forall U_i$

$$\begin{aligned} & \forall LT_k \text{succ}(LT_j, LT_k) \\ & \Rightarrow \text{main\_learning\_tasks}(C_j) = [\text{main\_learning\_tasks}(C_j), \\ & LT_k] \end{aligned}$$

그리고나서, 이러한 관찰이 다음과 같은 방법으로 처리된다. 모든 주-학습과제 리스트를 학습자가 방문하였다면 코스를 완료한 것으로 처리한다.

$$\begin{aligned} & \exists C_j \forall LT_k \forall U_i \\ & \forall LT_k (\text{main\_learning\_tasks}(LT_1, \dots, LT_k, \dots, LT_n) \\ & \quad \wedge \text{obs}(LT_k, U_i, \text{Visited})) \\ & \Rightarrow p\_obs(C_j, U_i, \text{Completed}) \end{aligned}$$

학습자모델에서 다루어지는 두 번째 기능은 평가를 통해 학습목표 달성을 체크하는 기능이다. 학습자의 평가점수가 교사가 명시한 기준 점수를 넘었을 때, 다음과 같이 평가객체를 참조하는 학습목표를 달성한 것으로 처리한다.

$$\begin{aligned} & \forall LT_j \forall U_i \\ & \forall LT_j (\text{test\_assignment}(LT_j, TO_k(\text{Obj}_i(t), L, x)) \\ & \quad \wedge \text{solved\_testitem}(U_i, LT_j, y)) \\ & \wedge (y \geq x) \\ & \Rightarrow p\_obs(LT_j, U_i, \text{passed}) \end{aligned}$$

학습자의 수준에 맞는 평가객체를 제공하여 학습자로 하여금 학습에의 도전의지를 갖도록 하기 위하여 본 시스템에서는 학습자모델 안에 학습자별 학습수준값을 처리하여 보관하도록 하였다. 이러한 학습수준값을 통하여 학습자는 '상/중/하'로 구분된 평가객체 중 자신의 수준에 맞는 평가객체를 제공 받게 되며, 또한 학습수준값이 '상'에 해당하는 학습자에게는 '도전' 관계로 연결된 학습과제를 추천하도록 한다. 학습자의 학습수준값의 범위는 0.0에서 5.0까지 실수값으로 표현되며, [0.0, ..., 2.0] 범위는 '하', [2.0, ..., 4.0] 범위는 '중', [4.0, ..., 5.0] 범위는 '상'으로 학습자 수준을 처리하도록 하였다. 이러한 학습수준값은 학습자가 평가객체를 수행 시 평가점수 결과에 따라 동적으로 변동된다.

$$\begin{aligned} & \forall U_i \text{learning\_value}(U_i, z), z = [0.0, \dots, 5.0] \\ & \exists z \text{learning\_value}(U_i, z) \in [0.0, \dots, 2.0] \\ & \Rightarrow \text{learner\_type}(U_i, 'E') \quad , E; \text{Elementary} \\ & \exists z \text{learning\_value}(U_i, z) \in [2.0, \dots, 4.0] \\ & \Rightarrow \text{learner\_type}(U_i, 'I') \quad , I; \text{Intermediate} \end{aligned}$$

$$\begin{aligned} & \exists z \text{learning\_value}(U_i, z) \in [4.0, \dots, 5.0] \\ & \Rightarrow \text{learner\_type}(U_i, 'A') \quad , A; \text{Advanced} \end{aligned}$$

#### (4) 적응 규칙(Adaptation Component)

ILS에서는 선행학습이 제대로 이루어지지 않은 경우 선행학습을 제공하고, 학습수준값이 높은 학습자에게 도전과제를 추천하며, 학습자의 수준에 따라 난이도가 다른 평가객체를 제시하는 3가지 적응 규칙을 설정하였다.

첫번째 적응규칙은 선행학습과 관련된 내용이다. 학습자가 '후속' 관계로 연결된 주-학습흐름 상에 있는 어떤 학습과제에 진입하고자 할 때, 시스템은 우선적으로 그 학습과제에 연결된 '선행' 학습과제의 학습여부를 살펴본다. '선행' 관계로 연결된 학습과제가 모두 'passed' 되었다면 그 학습과제가 제공된다. 선행학습 중 어느 하나가 'failed'일지라도 그 선행학습 과제에 접근한 학습 시도 횟수가 2번을 넘는다면 선행학습이 모두 'passed'되지 않았을지라도 '후속' 관계로 연결된 학습과제를 제시하게 된다.

$$\begin{aligned} & \forall LT_j \forall U_i \\ & \forall LT_k (\text{succ}(LT_k, LT_j) \wedge \text{obs}(LT_k, U_i, \text{Visited})) \\ & \wedge \forall LT_l (\text{preq}(LT_j, LT_l) \wedge p\_obs(LT_l, U_i, \text{passed})) \\ & \vee \exists LT_l (\text{preq}(LT_j, LT_l) \wedge \neg p\_obs(LT_l, U_i, \text{passed})) \\ & \Rightarrow \text{learning\_trial}(LT_l, U_i) \geq 2 \\ & \Rightarrow \text{next\_page}(LT_j, U_i) \end{aligned}$$

만일 선행학습 과제 중 학습자가 'passed'하지 못한 과제가 있고, 선행학습 접근 횟수가 2번 미만이라면 그 선행학습을 제공하게 된다.

$$\begin{aligned} & \forall LT_i \forall U_i \\ & \forall LT_k (\text{succ}(LT_k, LT_j) \wedge \text{obs}(LT_k, U_i, \text{Visited})) \\ & \wedge \exists LT_l (\text{preq}(LT_j, LT_l) \wedge \neg p\_obs(LT_l, U_i, \text{passed})) \\ & \Rightarrow \text{learning\_trial}(LT_l, U_i) < 2 \\ & \Rightarrow \text{next\_page}(LT_i, U_i) \end{aligned}$$

두번째 적응 규칙은 '도전' 관계로 연결된 학습과제 제시에 관한 내용이다. 도전 학습과제는 학습수준이 높은 학습자에게만 제시하여 학습에서의 도전력을 키워 성취감을 느끼게 하도록 하기 위함이며, 학습수준이 낮은 학습자에게는 도전과제를 제함으로써 과제수행의 부담을 줄이고자 하였다. 학습수준값을 살펴 '상'에 해당하는 학습자에게는 도전과제를 추천하도록 처리한다.

$$\begin{aligned} & \forall LT_j \forall U_i \\ & \forall U_i \text{learner\_type}(U_i, 'A') \\ & \quad \wedge \exists LT_j \text{chal}(LT_k, LT_j) \\ & \Rightarrow \text{recommend\_docs}(LT_j, U_i) \end{aligned}$$

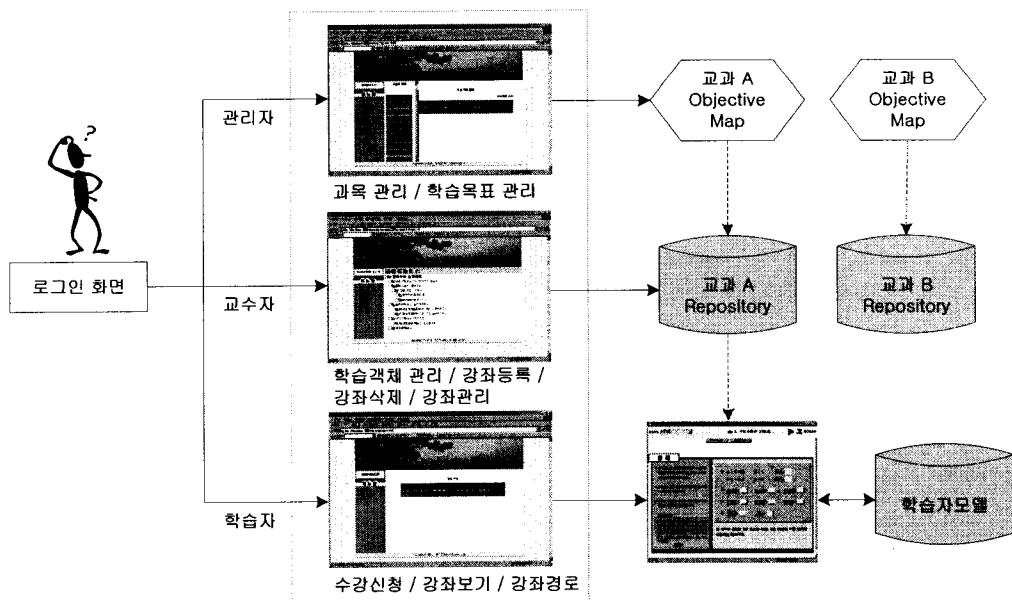
본 시스템에서 설정한 마지막 적용 규칙은 학습자의 수준에 따라 난이도가 다른 평가객체를 제공하고자 하는 것이다. 모든 학습자에게 동일한 학습과제나 평가객체를 제공하는 것은 학습수준이 높은 학습자에게는 학습을 지루하게 만들 수 있으며 학습수준이 낮은 학습자에게는 학습에의 부담을 주게 될 수도 있다. 자신의 수준에 적절한 평가를 제공했을 때 학습자는 그에 따른 도전의지와 성취감을 느낄 수 있으리라 생각한다.

$$\begin{aligned} & \forall LT_j \forall TI_k \forall U_i \\ & \forall TI_k \text{test\_assignment}(LT_j, TI_k(OBJ_i(t), L, x)) \\ & \quad \wedge U_i \text{learner\_type}(U_i, 'A') \\ & \Rightarrow \text{next\_page}(TI_k(OBJ_i(t), 'H, x), U_i) \\ & \forall TI_k \text{test\_assignment}(LT_j, TI_k(OBJ_i(t), L, x)) \\ & \quad \wedge U_i \text{learner\_type}(U_i, 'I') \\ & \Rightarrow \text{next\_page}(TI_k(OBJ_i(t), 'M, x), U_i) \\ & \forall TI_k \text{test\_assignment}(LT_j, TI_k(OBJ_i(t), L, x)) \\ & \quad \wedge U_i \text{learner\_type}(U_i, 'E') \\ & \Rightarrow \text{next\_page}(TI_k(OBJ_i(t), 'L, x), U_i) \end{aligned}$$

### 5. 시스템 설계 및 구현

본 연구에서 제안하는 학습객체 기반 개별화 학습 시스템 인터페이스의 구성도는 (그림 1)과 같이 크게 관리자 모드와 교수자 모드, 그리고 학습자 모드로 구분된다. 시스템의 저작모듈은 Java Application, Java Servlet, JSP로 구현되었으며 사용자 환경은 Java Applet과 JSP로 구현되었고, 시스템 기반 환경은 윈도우즈 2000, Apache Tomcat Server, 데이터베이스는 MS Access 2000이다.

관리자 모드에서는 시스템 활용을 위해 기본적으로 요구되는 과목 등록과 각 과목에서의 학습 목표를 등록하며, 이렇게 입력된 자료는 교과별 학습목표 맵을 생성하게 된다. 교수자 모드에서는 학습객체와 코스에 관련된 작업을 수행하는데, 우선 학습객체 관리 부분은 학습객체를 등록하는 부분이다. 이렇게 등록된 학습객체는 각 교과별 Repository에 보관이 되고, 이러한 학습객체를 조합하여 코스를 구성하게 된다. 본 시스템에서는 저작자의 동의를 얻어 2001년과 2002년 전국 교육용 소프트웨어 공모전에 출품된 불대수 관련 웹 코스웨어[1, 3, 4, 5, 8]를 학습객체로 구성하였으며, 이러한 학습객체를 조합하여 코스를 구성해보았다. 학습자 모드에서는 학습자가 수강



(그림 1) 시스템 인터페이스 구성도

신청을 하면 강좌가 뷰어창을 통하여 보여진다. 학습자모델을 통하여 코스에서는 학습자의 인지 상태에 따른 적절한 분기가 이루어지며, 이렇게 개별화된 학습경로는 강좌경로 메뉴에서 확인할 수 있다.

### 5.1 관리자 모드

관리자 모드에서는 크게 ‘과목 관리’와 ‘학습목표 관리’가 이루어진다. 교수자가 생성하는 학습객체나 코스는 여기에서 등록된 과목별 리파지토리에 저장이 되며, 코스 생성 시 다른 교사가 다른 과목을 위해 생성한 학습객체를 검색하여 코스에 적용할 수 있다. 과목코드는 대분류와 소분류로 구분하여 교육과정상의 기본 목차를 구성하도록 하였다.

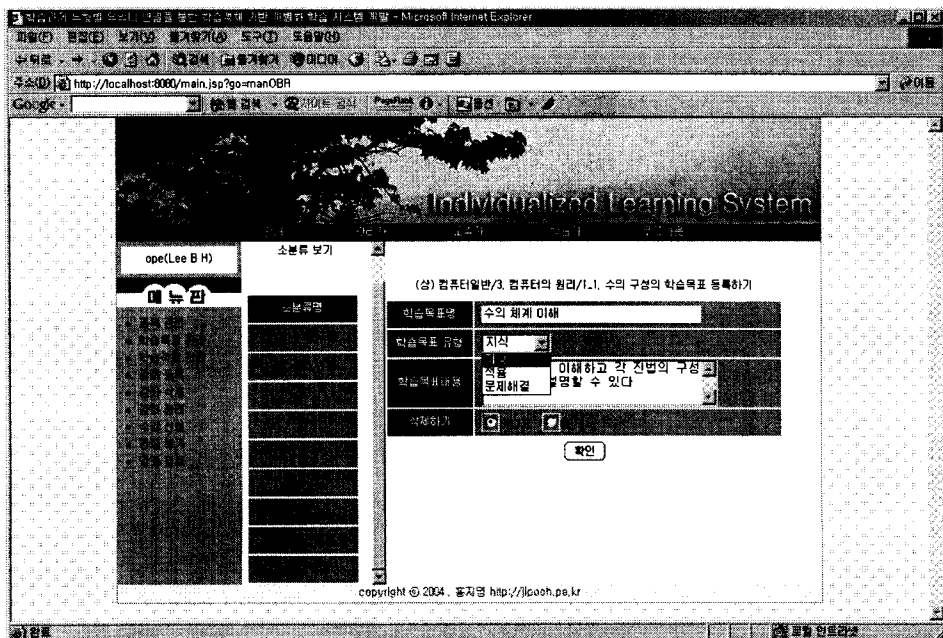
(그림 2)는 관리자가 학습목표 관리 메뉴를 통하여 학습목표 맵을 생성하는 화면이다. 각 교과의 목차에 해당되는 학습목표를 등록하는데, 각 학습목표는 ‘지식’, ‘적용’, ‘문제해결’이라고 하는 유형을 구분하도록 하여 학습자가 학습과제 수행 후 어떠한 유형의 학습목표를 달성했는지를 파악하는 기준이 되도록 했다.

### 5.2 교수자 모드

관리자가 생성한 학습목표 맵은 교수자가 학습객체를 생성하고 코스를 생성하는 과정에서 각각의 학습객체와 코스가 어떠한 학습목표와 관련되는지를 나타내게 된다. 교사는 학습객체 생성을 위한 ‘학습객체 관리’ 메뉴를 포함하는데, 기존에 만들어져 있는 것을 시스템으로 가져오는 기능을 구현하였다. 코스 생성을 위한 메뉴로는 강좌명이 무엇이고 관련된 학습목표가 무엇인가 연결짓는 ‘강좌등록’, 기존 강좌를 삭제하는 ‘강좌 삭제’, 그리고 구체적으로 강좌를 다양한 관련성으로 학습객체를 엮어나가는 기능을 ‘강좌 관리’에서 제공하고 있다.

#### (1) 학습객체 관리

학습객체 관리 부분에서는 기존에 이미 생성된 학습객체를 등록하고 관련된 학습목표와 연결시키는 기능을 포함한다. 각각의 학습객체와 평가객체는 SCORM의 SCO 형태로 구성을 하였으며, 하나의 학습객체를 구성하는 모든 파일들을 압축한 ZIP 파일을 업로드 한다. ‘학습목표명’에서 연



(그림 2) 학습목표 관리 메뉴



결되는 학습목표는 '후속'이나 '선행' 링크로 연결되는 학습과제의 학습목표 달성여부를 체크하는 기준이 되며, 또한 교수자가 코스를 구성하는 과정에서 학습객체를 검색할 때 메타데이터로 이용될 수 있다.

(2) 코스 생성

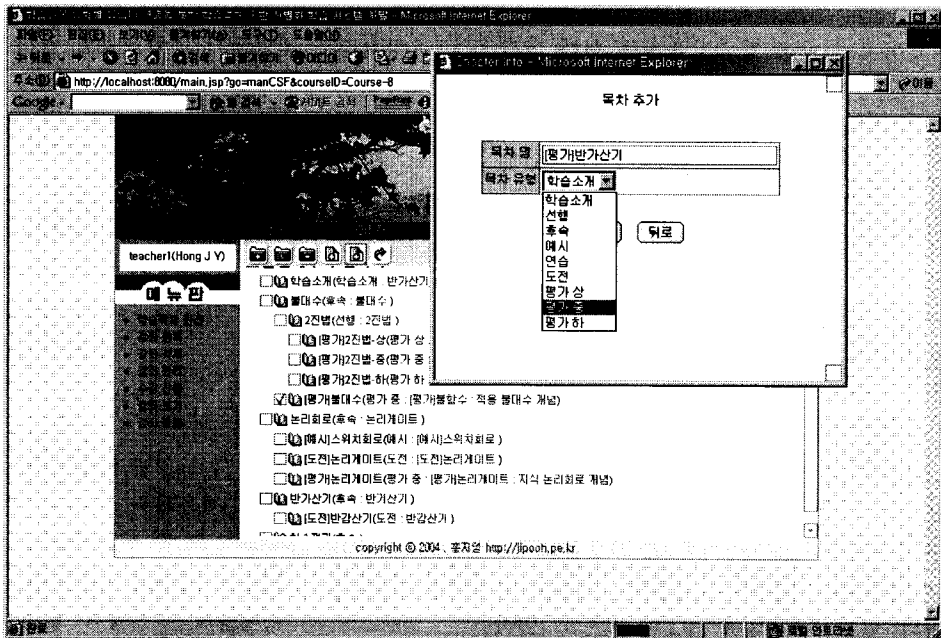
(그림 3)은 교수자가 Repository에 등록되어 있는 학습객체를 조합하여 코스를 생성하는 화면이다. 교수자는 우선 코스를 목차의 형태로 구성하게 되는데, 이 때 각각의 학습과제를 연결하는 과정에서 7가지의 관련성 요소를 적용하여 학습객체를 유의미하게 연결하도록 하였다. 또한 평가부분을 '평가-상', '평가-중', '평가-하' 링크로 구분하여 학습자의 수준에 따른 평가객체가 제공되도록 구성하였다. 목차가 구성되면 각각의 학습과제 노드에 해당되는 학습객체를 등록하고 그 학습객체에서 참조하는 학습목표를 등록한다.

5.3 학습자 모드

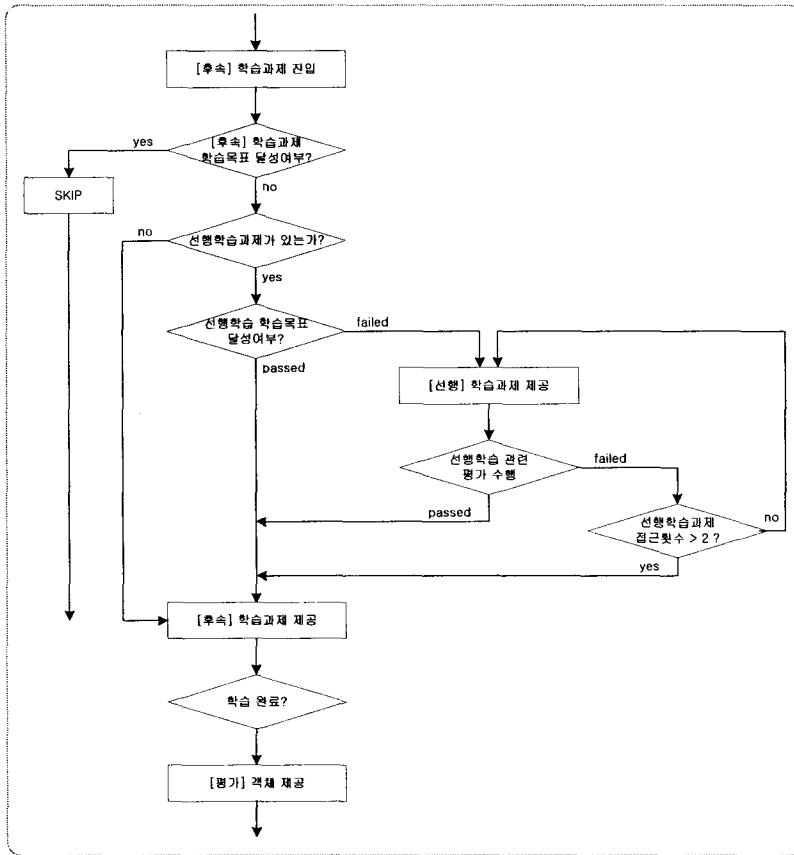
학습자는 학습강좌 수강과 관련하여 등록되어 있는 과목의 수강 신청을 하고 강좌 보기를 할

수 있다. 학습자가 강좌를 신청하면 코스 리파지토리 안에 들어있는 코스를 학습하게 되는데, 여기에서 학습자별로 개별화된 코스의 경로를 제시해 주기 위하여 학습자모델의 정보를 참고하게 된다. 학습자모델에서는 학습자별 학습목표 맵을 이용하여 학습목표 달성여부를 판단하게 되고, 인지상태를 평가하여 수준별 학습과 평가객체를 제공하며, 학습번호도 판단을 통하여 학습자에게 적절한 학습과제를 선택할 수 있다. 본 시스템에서는 학습목표 맵과 제한된 인지상태 판단 부분만을 구현하였다. 학습자의 전반적인 인지상태를 판단하기 위한 연구와 학습자의 선호도를 통하여 학습자의 성향에 맞는 학습과제를 제시하는 기능은 계속적으로 연구되어야 할 과제이다.

학습자에게 개별화된 경로를 제시해 줄 때 적용되는 기본 분기의 흐름은 (그림 4)와 같다. 학습자가 '후속' 링크로 연결된 학습과제에 진입하면 우선 그 학습과제의 학습목표 달성여부를 살핀다. 진입하고자 하는 학습과제의 학습목표가 달성된 상태라면 그 부분을 스킵하고 다른 학습과제로 진행하도록 한다. 만일 'failed' 상태라면 이번에는 그 학습과제에 '선행' 관계로 연결되어 있는 학습과제의 학습목표를 체크하여 선행학습이



(그림 3) 강좌 관리 메뉴



(그림 4) 개별화된 경로 제공을 위한 기본 분기흐름

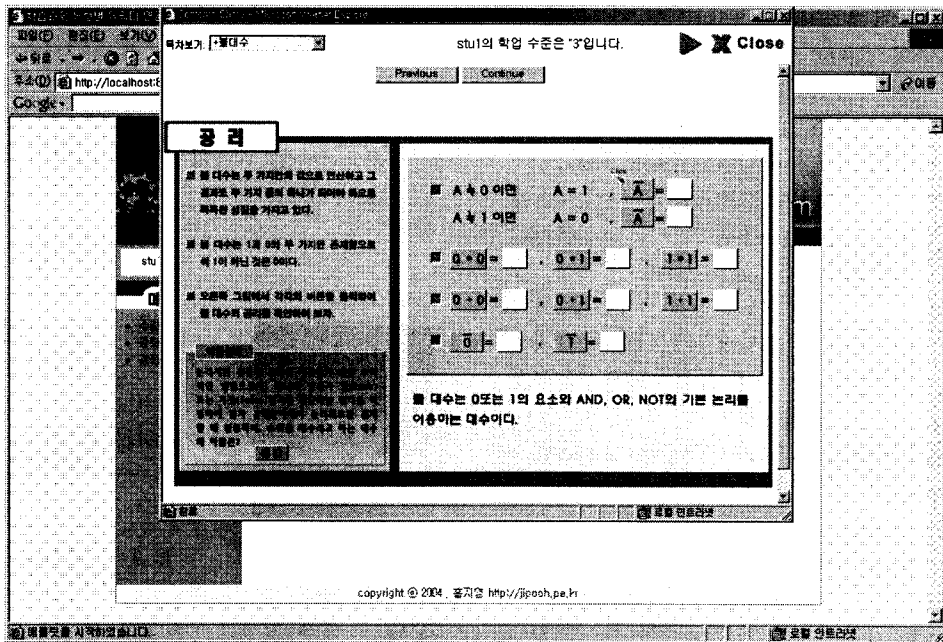
제대로 이루어지지 않은 학습자에게는 선행 학습과제를, 선행학습이 이미 만족된 상태는 선행 학습과제를 무시하고 '후속'으로 연결된 학습과제를 바로 제시한다. 선행학습이 2번이나 제공되었는데도 불구하고 학습목표를 통과하지 못하면 그 부분의 학습목표를 'failed'로 남겨두고 다음 학습을 진행한다. 주흐름상의 학습과제를 완료하면 학습자에게는 수준에 따른 평가객체가 제공되어 학습자 개인별로 저장되어 있는 학습수준값을 동적으로 변화시킨다.

학습자가 수강신청을 하면 강좌보기 메뉴를 통하여 (그림 5)와 같이 코스 뷰어 화면을 통한 학습이 진행된다. 코스 뷰어 화면의 상단에는 학습수준값을 나타내도록 하여 학습자의 학습동기를 부여하고자 하였다. 이러한 학습수준값은 평가를 거칠 때마다 최고 0.1, 최저 -0.1의 값으로 변화하며 학습수준값의 범위는 0.0에서 5.0 사이이다.

강좌경로 메뉴에서는 학습자의 강좌경로 및 학습목표 달성여부를 보여준다. 콘텐츠 설계 시 '후속' 링크로 연결된 주된 학습과제는 학습자 수준에 관계없이 모든 학습자가 필수적으로 거치는 학습과제이며, 학습자가 갖고 있는 학습목표 맵의 상태와 동적으로 변화하는 학습자의 학습수준값을 기초로 학습자별로 차별화되어 적용된 학습경로를 확인할 수 있다.

## 6. 결론

본 연구는 학습객체 기반에서의 코스 설계에서 좀 더 개별화되고 적응화된 코스를 제공할 수 있도록 하기 위하여 크게 두가지의 문제제기에서 출발하였다. 현재 SCORM 기반에서의 시퀀싱 유형은 '장-절-항' 형태인 선형적 흐름의 학습활동 트리(activity tree) 중심 설계로 인하여 다양한



(그림 5) 학습자 강좌보기 메뉴

분기 제공이 어렵다는 것과, 융통적으로 활용하기에는 제한적이라는 것이었다. 이러한 문제를 해결하기 위하여는 SCORM 기반 코스 설계에서의 선형적인 형태가 아닌 관련성 요소로 엮어진 비선형적 프레임워크로 설계되어야 하며, 학습자 모델의 다양화된 적용규칙을 위해서는 학습자의 인지상태의 판단에 관한 연구가 필요함을 제안하였고, 이를 위해 ‘관련성 요소 추출에 관한 연구’, ‘학습목표와의 연결방법에 관한 연구’, ‘학습자의 인지상태 판단에 관한 연구’ 문제를 선정하여 연구를 수행하였다.

본 연구에서는 ‘학습객체’의 개념에서 개별화 학습의 가능성을 발견하고, 학습객체의 ‘재사용’이라고 하는 경제적 잇점을 넘어 개별화되고 적응적인 학습을 위한 설계의 기본 요소로서 접근해보고자 하였다. 이는 학습객체 기반의 코스 설계가 단지 무의미한 객체들의 집합이라는 비판이 대두되는 시점에서 e-Learning 안에 유의미한 학습과 진정한 교육을 담고자 하는 시도가 될 것이며, 또한 교사로 하여금 가르치고자 하는 내용에 관한 지식의 구조를 잘 생성하여 학습자에게 효율적으로 전달해주는 교사 본연의 임무를 찾을 수 있도록 할 것이다. 학습객체는 교사로 하여금

개별 과정에 있어 시간과 비용 면에서 상당한 잇점을 제공한다. 이러한 관점에서 볼 때 학습자의 ‘생각을 키우는’ 학습 설계가 가능할 것이라고 기대한다.

### 참 고 문 헌

- [1] 고선규, 박순자 (2002). 컴퓨터의 구조와 원리[웹 코스웨어]. [Online] Available: <http://cont111.edunet4u.net/2002/gosu93/>.
- [2] 권혁일 (2000). 적응적 웹 기반 수업의 학습 효과성 고찰. 교육공학연구, 16(4), 23-50.
- [3] 김교운, 변종건 (2002). 클릭! 논리회로[웹 코스웨어]. [Online] Available: <http://cont3.edunet4u.net/~magic1/>.
- [4] 김수영, 원선자 (2002). 웹으로 배우는 논리회로 구성원리. [웹 코스웨어]. [Online] Available: <http://cont111.edunet4u.net/2002/ksy/>.
- [5] 박근태, 배홍식 (2002). 정보의 표현과 불대수 [웹 코스웨어]. [Online] Available: <http://cont111.edunet4u.net/2002/ss306/>.
- [6] 박성익 (1997). 교수·학습방법의 이론과 실

제(II). 교육과학사.

- [7] 양영선 역 (1999). 교사를 위한 체제적 수업 설계. 교육과학사. [원전: Reiser, R. A. & Dick, W. (1996). Instructional Planning - A Guide for Teachers. Allyn and Bacon.]
- [8] 임중혁 (2001). 디지털 논리회로 [웹 코스웨어]. [Online] Available: <http://cont1.edunet4u.net/digital/>.
- [9] 홍지영, 이범진, 송기상 (2004). 적응형 하이퍼미디어 시스템 분석에 기초한 학습객체 기반 개별화 코스 설계 모형. 한국교육공학회 2004 춘계학술발표회.
- [10] ADL Technical Team (2004a). Sharable Content Object Reference Model(SCORM) 2004 2nd Edition.
- [11] ADL Technical Team (2004b). SCORM 2004 Photoshop Examples Version 1.1.
- [12] Baroque, L. B. & Melo, R. N. (2003). Learning Theory and Instructional Design using Learning Object. Learning Objects 2003 Symposium.
- [13] Carnegie Mellon Learning Systems Architectures Lab. (2002). SCORM Best Practices Guide for Content Developers, v1.8. Carnegie Mellon University.
- [14] Cisco Systems, Inc. (2003). Reusable Learning Object Strategy: Designing and Developing Learning Objects for Multiple Learning Approaches.
- [15] Department for Education (1999). The National Curriculum. London: HMSO. [Online] Available: <http://www.nc.uk.net/>
- [16] Douglas, I. (2001). Instructional Design based on Reusable Learning Objects: Applying Lessons of Object-Oriented Software Engineering to Learning System Design. 31st ASEE/IEEE Frontiers in Education Conference.
- [17] Henze, N. (2003). From Web-based Educational Systems to Education on the Web: On the road to the Adaptive Web. 7th International Conference on Knowledge-based Intelligence Information & Engineering Systems, September 3-5,

2003, Oxford, UK.

- [18] Henze, N. & Nedjdl, W. (2003). Logically Characterizing Adaptive Educational Hypermedia Systems. In Proc. of AH2003 Workshop, World Wide Web Conference, May, 2003.

## 홍 지 영

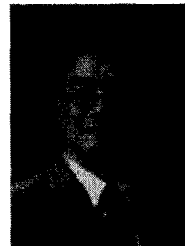


1994 충북대학교 컴퓨터공학과 졸업(공학사)  
 2000 한국교원대학교 컴퓨터교육과 졸업(교육학석사)  
 2004 한국교원대학교 컴퓨터교육과 졸업(교육학박사)

1994 ~ 2004 속초상고, 속초여고 교사  
 2004 ~ 현재 한국교육개발원 교육통계정보센터 연구원

관심분야 : 컴퓨터 교육, 학습객체, 개별화 학습  
 E-Mail: jipooh@kedi.re.kr

## 송 기 상



1983 아주대학교 전자공학과 졸업(학사)  
 1985 한국과학기술원 전기 및 전자공학과 졸업(석사)  
 1985 ~ 1990 한국원자력연구소 연구원

1994 University of Washington 전기공학과 졸업 (Ph. D.)

1994 ~ 1995 한국전자통신연구소 선임연구원  
 1995 ~ 현재 한국교원대학교 컴퓨터교육과 교수  
 관심분야: 지능형 교수 시스템(ITS), e-Learning, 분산 멀티미디어, CIM, 원격교육 및 컴퓨터를 이용한 교육

E-Mail: kssong@cc.knue.ac.kr