

# Development of Intelligent Tutoring System Framework: Using Guided Discovery Learning

Raja Shekhar  
10305034  
M.Tech-2

Under the guidance of  
*Prof. Sridhar Iyer*

June 25, 2012

# Outline

- Existing Systems
- Guided Discovery Learning
- ITS Support for Guided Discovery
- Architecture and Modules
- Integration of the System
- Conclusion and Future Work

# Existing Systems

- Developed in- geography, circuits, medical diagnosis, computer programming
- Some Example ITSs:
  - SQLT-Web: SQL Tutor
  - Autotutor: Physics Tutor. Supports voice interaction.
  - Activemath: Mathematics Tutor

# SQLT-Web Tutor

**Problem 58** For all directors who made more than 5 movies, list their number, names and the total number of movies. [Change database](#)

NEW PROBLEM Logout and save your student model

Select   
From   
Where   
Group by   
Having   
Order by

Run query (music and movies only) Click [HERE](#) to view your history

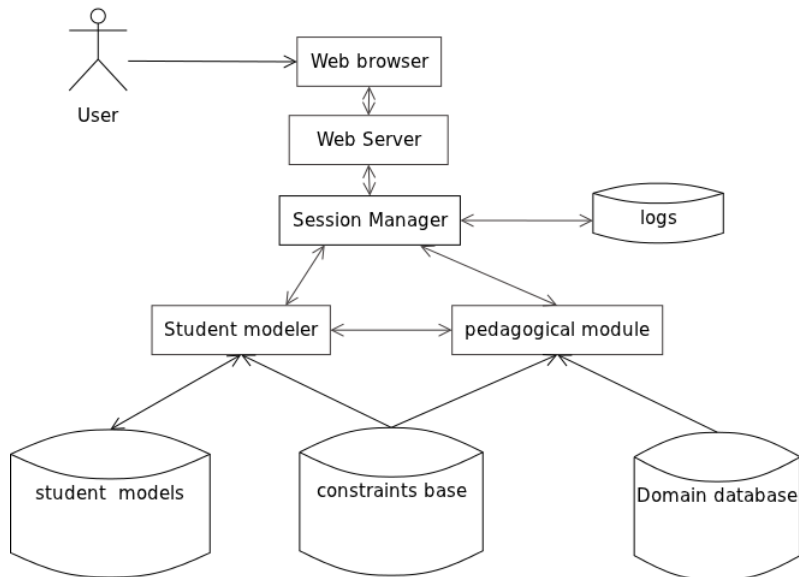
Almost there - you made 2 mistakes.  
Check the attributes you are using in FROM to join the tables!  
You can correct your query and press 'Submit' again, or try getting some more feedback.  
Would you like to have another go?

Schema for **movies** (click on database name for more information)

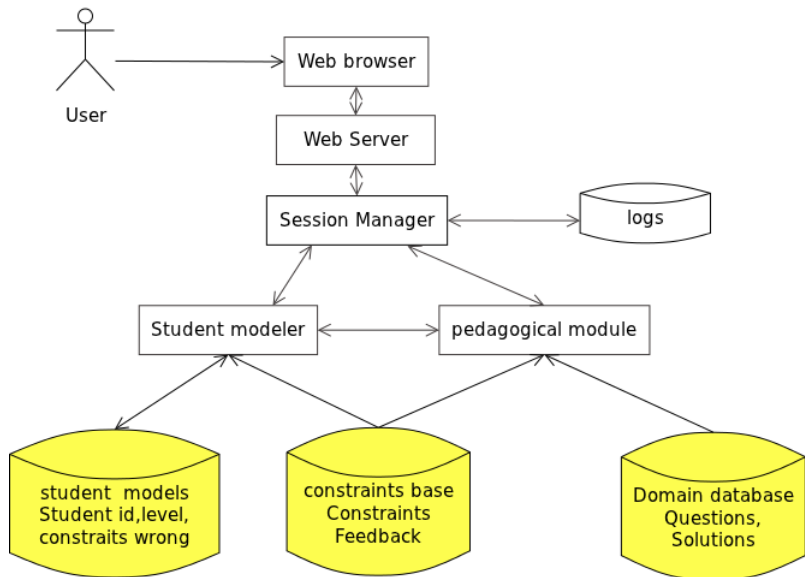
Table name	Attribute list (primary keys <u>underlined</u> and foreign keys in <i>italics</i> )
<a href="#">DIRECTOR</a>	<u>NUMBER</u> LNAME FNAME BORN DIED
<a href="#">MOVIE</a>	<u>NUMBER</u> TITLE TYPE AANOM AAWON YEAR CRITICS <i>DIRECTOR</i>
<a href="#">STAR</a>	LNAME FNAME <u>NUMBER</u> BORN DIED CITY

Figure: SQLT GUI

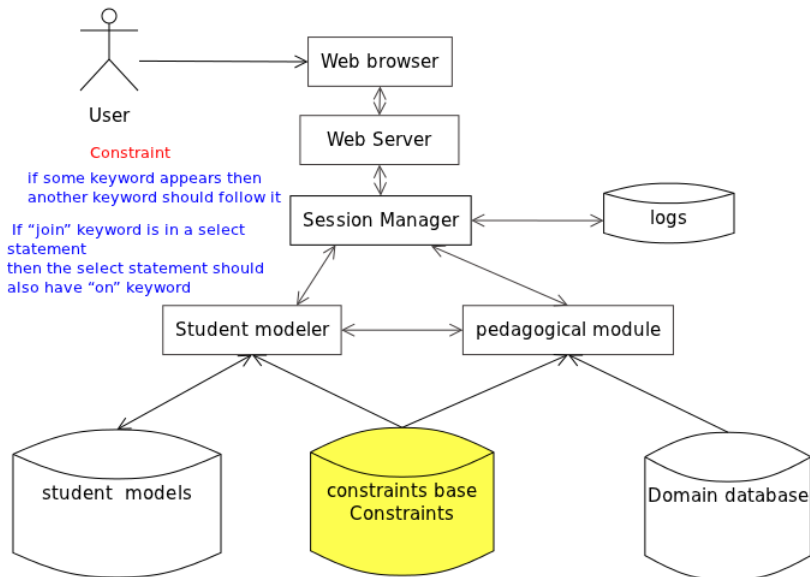
# SQLT-Web Tutor- Architecture



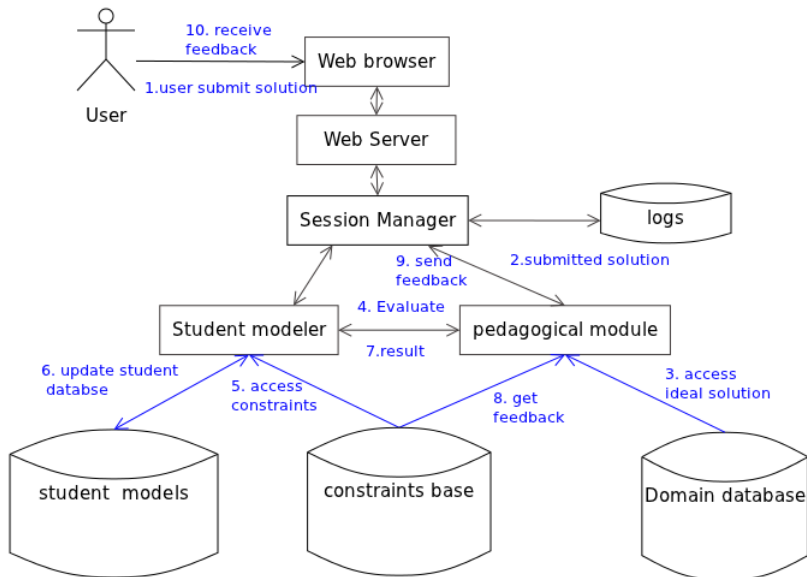
# SQLT-Web Tutor- Architecture



# SQLT-Web Tutor- Architecture



# SQLT-Web Tutor- Architecture





# Assessment- CBM

- No data

Find top 5 students of the semester who have taken atleast 4 courses

# Assessment- CBM

- No data
- Uses Constraint Based Modeling(CBM) for assessment
  - Syntax Verification
  - Equivalent constructs checking - Constraints
- Feedback - Associated to constraints

# Our Framework

- Problems with existing systems
  - Single teaching style
  - Subject specific
    - Due to assessment process

# Our Framework

- Problems with existing systems
  - Single teaching style
  - Subject specific
    - Due to assessment process
- Our ITS Framework
  - 4 teaching styles
  - Is not subject specific
    - Using MCQ for our ITS

- Existing Systems
- Guided Discovery Learning
- ITS Support for Guided Discovery
- Architecture and Modules
- Integration of the System
- Conclusion
- Future Work

# Guided Discovery Learning

- Hands-on activities

# Guided Discovery Learning

- Hands-on activities

## Example:

Goal: *array memory allocation* concept

What is the output of the following snippet

```
main(){
int a[]={1,2,3,4};
float b[5]={3.2,8.7,8,9.8};
printf("%u %u %u %u %u",sizeof(int),&a[0],&a[1],&a[2],&a[3]);
printf("%u %u %u %u %u",sizeof(float),&b[0],&b[1],&b[2],&b[3]);
}
```

# Guided Discovery Learning-Steps

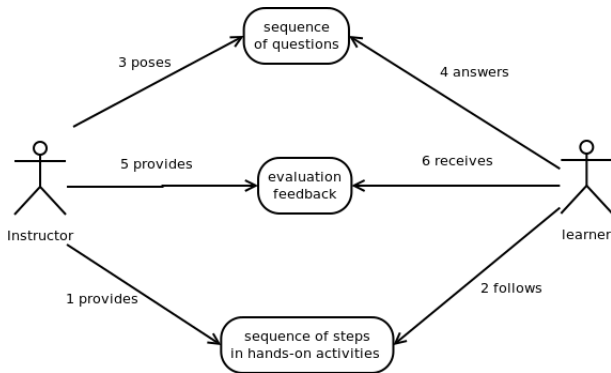


Figure: Steps in guided discovery learning



- Existing Systems
- Guided Discovery Learning
- ITS Support for Guided Discovery
- Architecture and Modules
- Integration of the System
- Conclusion
- Future Work

# ITS Support for Guided Discovery

## Course Structure

- Course - C Language
- Topic - Arrays
- Subtopic - 1D Arrays
- Quiz

# ITS Support for Guided Discovery

## Course Structure

- Course - C Language
- Topic - Arrays
- Subtopic - 1D Arrays
- Quiz

## Order of teaching/Pre-requisite relation

- 1 Topic dependency
- 2 Subtopic dependency

## Quiz-

- Multiple choice questions
- 2 types of questions
  - 1 Guiding questions : Hands-on activities  
Interactive pop-up window  
Not used for assessment
  - 2 Testing questions : Used for assessment

# Steps to be followed by instructor

- 1 Select/create course
- 2 Create topic
- 3 Enter Topic Dependency

# Steps to be followed by instructor

- 1 Select/create course
- 2 Create topic
- 3 Enter Topic Dependency

Total number of topics are: 5

Please enter the dependencies for the topic created

## TOPIC DEPENDENCY TABLE

	Arrays	Data types	Linked Lists	Pointers	Trees
Arrays	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Linked Lists	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Pointers	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Trees	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

# Steps to be followed by instructor

- 1 Select/create course
- 2 Create topic
- 3 Enter Topic Dependency
- 4 Create subtopic
- 5 Enter Subtopic Dependency

# Steps to be followed by instructor

- 1 Select/create course
- 2 Create topic
- 3 Enter Topic Dependency
- 4 Create subtopic
- 5 Enter Subtopic Dependency

Please enter the dependencies for the subtopic created

## SUBTOPIC DEPENDENCY TABLE

	1-D array	2-D arrays	MultiDimensional arrays
1-D array	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2-D arrays	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MultiDimensional arrays	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Submit



# Steps to be followed by instructor

- 1 Select/create course
- 2 Create topic
- 3 Enter Topic Dependency
- 4 Create subtopic
- 5 Enter Subtopic Dependency
- 6 Enter questions
- 7 Enter threshold value

# Steps to be followed by Learner

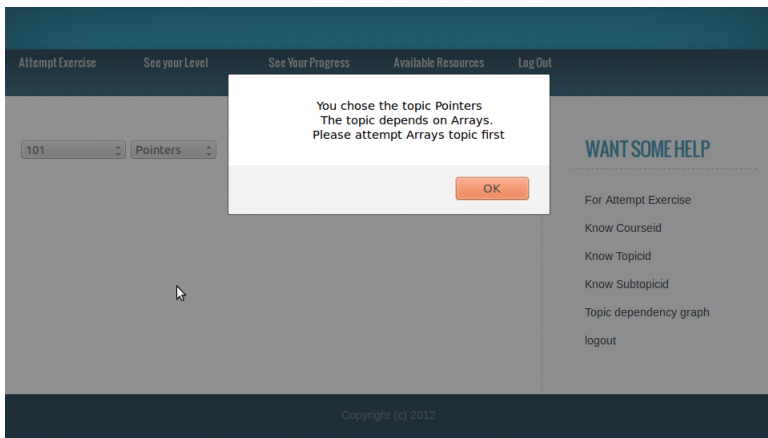
- 1 Select course
- 2 Select topic
- 3 Select subtopic
- 4 Use pop-up window
- 5 Submit answers
- 6 Reattempt or attempt remaining questions

# Steps followed by ITS

- 1 Topic dependency check

# Steps followed by ITS

## 1 Topic dependency check



The screenshot displays a user interface for an Intelligent Tutoring System (ITS). At the top, there is a navigation bar with links: "Attempt Exercise", "See your Level", "See Your Progress", "Available Resources", and "Log Out". Below this, a dialog box is centered on the screen with the following text: "You chose the topic Pointers. The topic depends on Arrays. Please attempt Arrays topic first". An "OK" button is located at the bottom right of the dialog box. In the background, a dropdown menu shows "101" and "Pointers". To the right of the dialog box, there is a "WANT SOME HELP" section with a list of options: "For Attempt Exercise", "Know Courseid", "Know Topicid", "Know Subtopicid", "Topic dependency graph", and "logout". At the bottom of the page, there is a copyright notice: "Copyright (c) 2012".

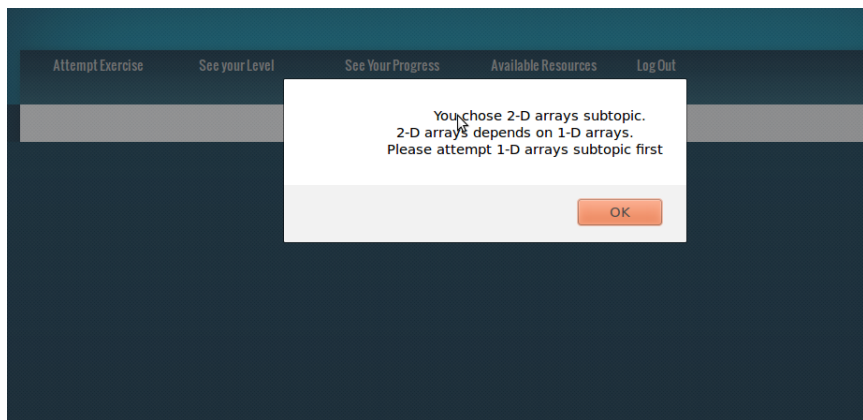
Figure: Topic dependency check

# Steps followed by ITS

- 1 Topic dependency check
- 2 Subtopic dependency check

# Steps followed by ITS

- 1 Topic dependency check
- 2 Subtopic dependency check



# Steps followed by ITS

- 1 Topic dependency check
- 2 Subtopic dependency check
- 3 Use adaptation logic

# Steps followed by ITS

- 1 Topic dependency check
- 2 Subtopic dependency check
- 3 Use adaptation logic

- 1) Present guiding question
- 2) Evaluate
- 3) Update learner knowledge
- 4) Repeat steps 1 to 3 until all guiding questions finish
- 5) if (#correct ans > threshold)  
    reattempt/attempt remaining option  
    else  
        display the testing question
- 6) Evaluate
- 7) Update learner knowledge
- 8) Repeat steps 5 to 7 until all testing questions finish



# Adaptation levels

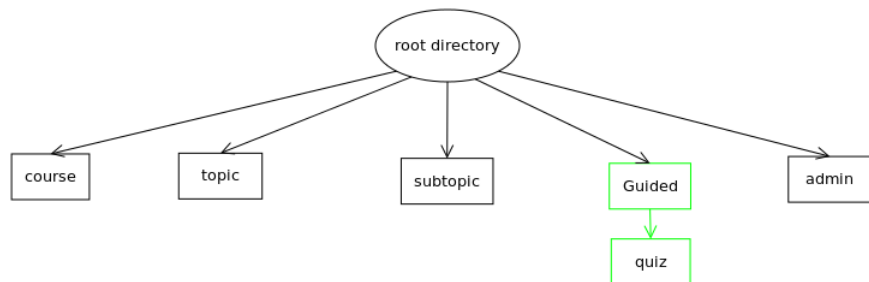
- Where is the adaptation applied?

# Adaptation levels

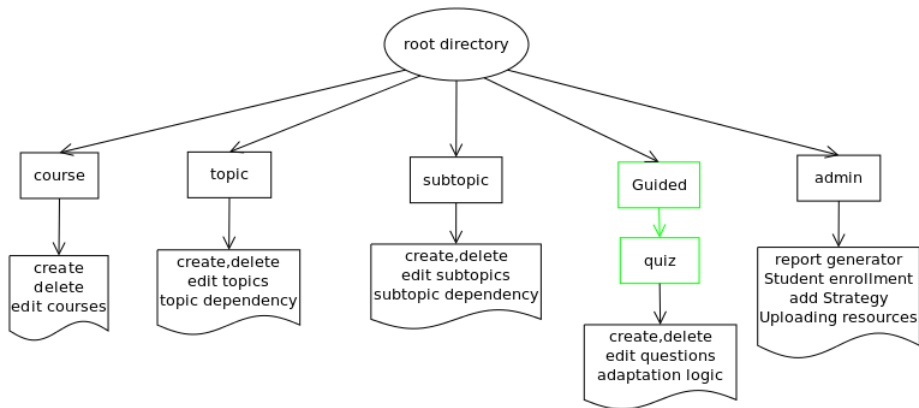
- Where is the adaptation applied?
  - Strategy Switching for learner
  - Topic level: Topic Dependency
  - Subtopic level: Subtopic Dependency
  - Promoting to next subtopic

- Existing Systems
- Guided Discovery Learning
- ITS Support for Guided Discovery
- Architecture and Modules
- Integration of the System
- Conclusion and Future Work

# Directory Structure



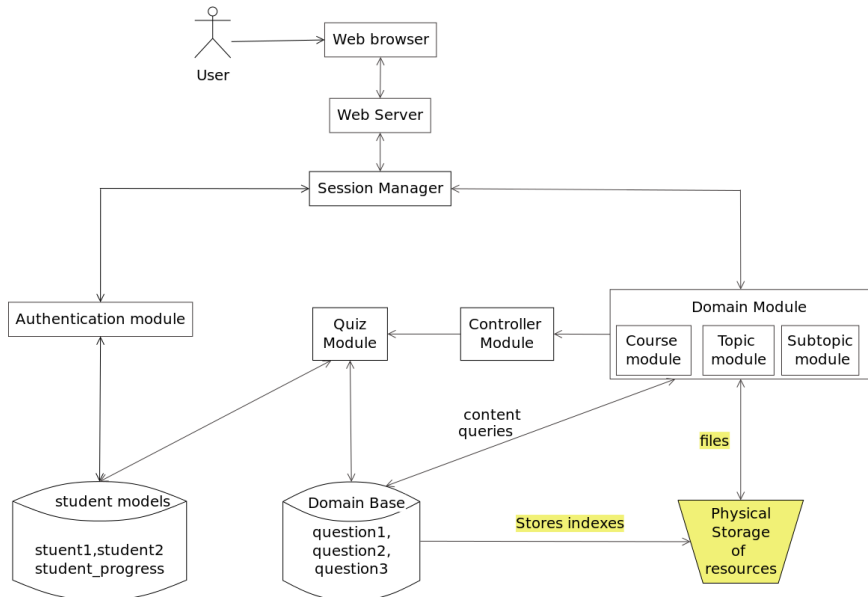
# Directory Structure







# Architecture of ITS



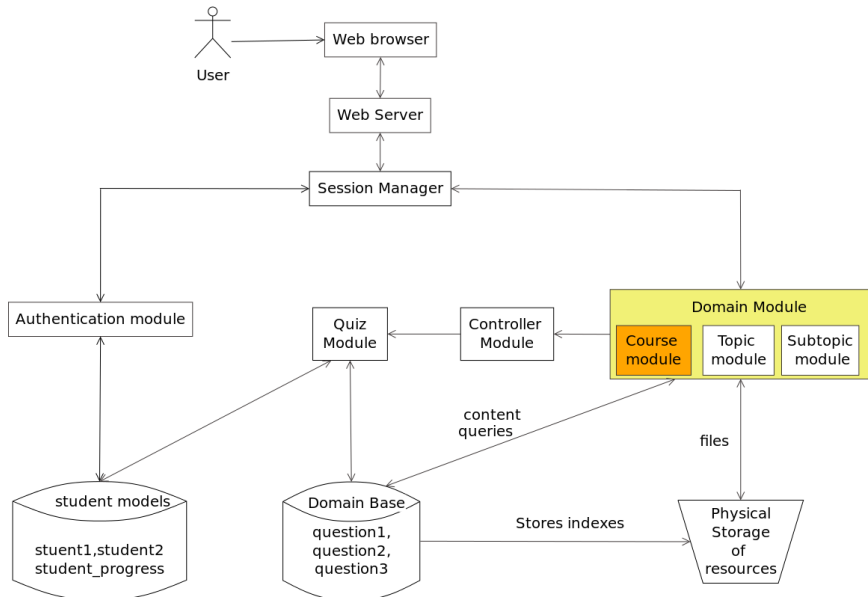




# Session Manager

- Authentication
- Access Control
  - Php Session Variables

# Domain module





# Topic Module

- Content creation
- Topic dependency

# Topic Module

- Content creation
- Topic dependency

Total number of topics are: 5

Please enter the dependencies for the topic created

## TOPIC DEPENDENCY TABLE

	Arrays	Data types	Linked Lists	Pointers	Trees
Arrays	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Linked Lists	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Pointers	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Trees	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Submit

# Topic Module-Loop Detection

- Content creation
- Topic dependency

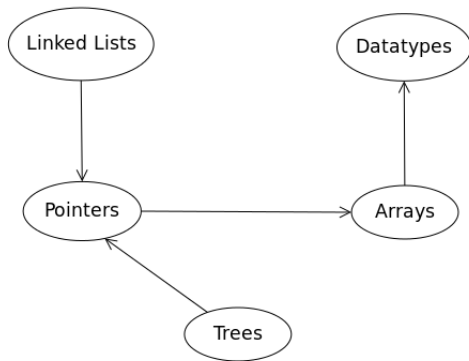


Figure: Topic Dependency Graph

# Topic Module-Loop Detection

- Content creation
- Topic dependency

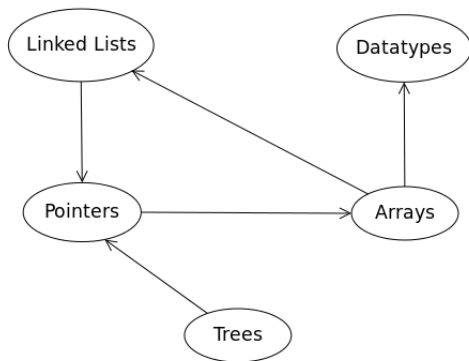


Figure: Loop formation



# Topic Module-Loop Detection

## Modified DFS Algorithm

---

- 1) All nodes are colored white
- 2) When a node is visited it is turned into red
- 3) Move on to descendants using DFS algorithm
- 4) When a node is visited completely it is turned into green
- 5) If we ever visit a red node during traversal then we have a cycle

# Topic Module-Loop Detection

## INTELLIGENT TUTORING SYSTEM

Add Course

Add Topic

Add SubTopic

Add Question

Add Strategy

Entered topic dependencies lead to a cycle.  
Please resolve them

OK

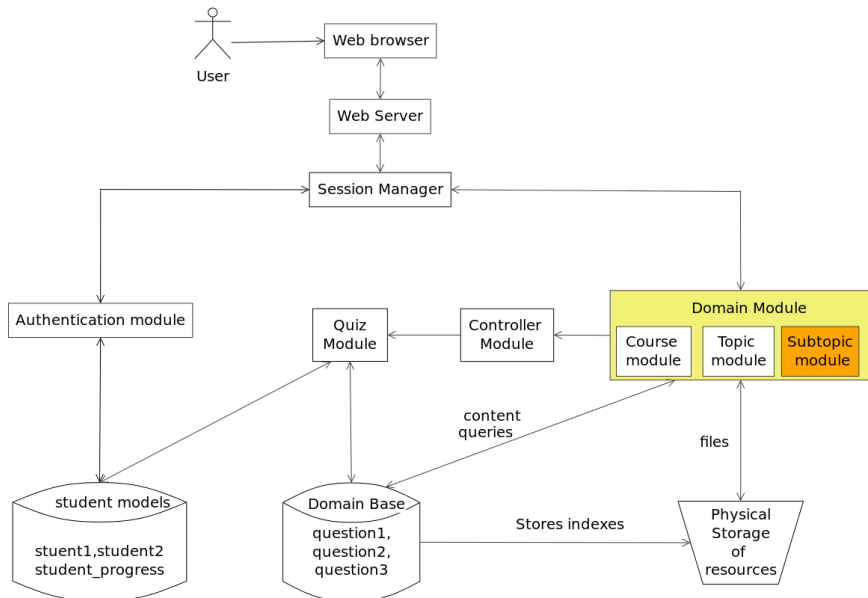
# Topic Module-Ensuring Dependency

- Learner can attempt an independent topic

# Topic Module-Ensuring Dependency

- Learner can attempt an independent topic
- Independent topic: If topic is independent of all topics
- Topic-A is independent of Topic-B iff
  - No edge from Topic-A to Topic-B or
  - All subtopics in topic-B are completed

# Domain module-Subtopic Module



# Subtopic Module

- Content creation
- Subtopic dependency- Loop detection

# Subtopic Module

- Content creation
- Subtopic dependency- Loop detection
- Learner can attempt an independent subtopic
- Independent subtopic: If subtopic is independent of all subtopics
- Subtopic-A is independent of Subtopic-B iff
  - No edge from Subtopic-A to Subtopic-B or
  - Subtopic-B is completed





# Quiz Module

- Content creation
- Evaluation
- Update student knowledge
- Adaptation logic

# Quiz Module

- Pop-up window for hands-on activities

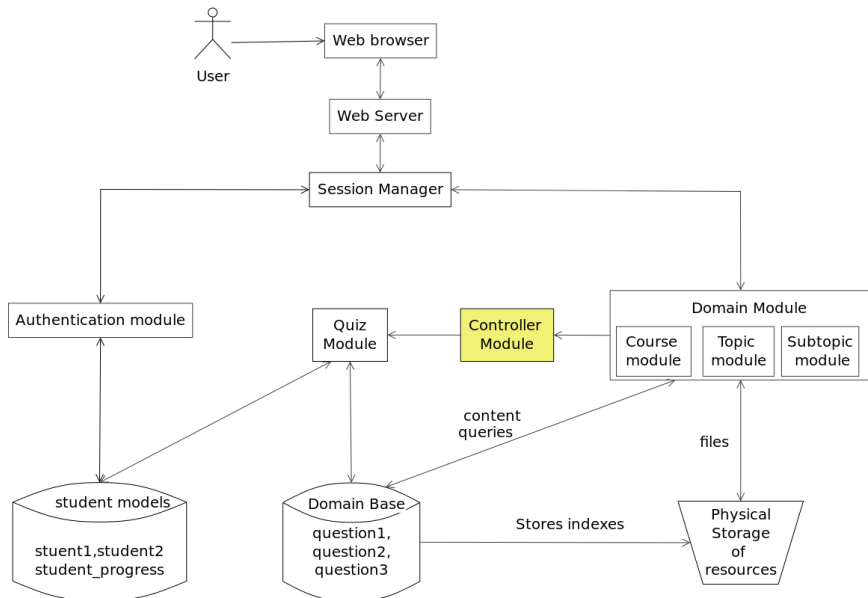
The screenshot displays the 'INTELLIGENT TUTORING SYSTEM' interface. At the top, there are navigation links: 'Attempt Exercise', 'See your Level', and 'See Your Progress'. The main content area contains a 'Guiding question. Use the commands button for typing c command:' and a 'Question:' section. The question asks for the output of a C program. The program code is: 

```
int a[4]={2,3,7},b[5]{1,5};
printf("%d %d %d",a[0],a[1],a[2]);
printf("%d %d %d
",a[1],1[a],b[1]);
```

 Below the code are four radio button options: '2 3 7 3 5', '0 1 2 1 1 1', '2 3 7 3 garbage 5', and '2 3 7 3 0 5'. There are 'submit' and 'commands' buttons. A 'SHOW RESULT' button is also present. A 'Mozilla Firefox' pop-up window is overlaid on the right, showing the same C code in a text area. Below the code in the pop-up, the output '2 3 done' is displayed.

Figure: Pop-up window

# Controller Module



# Controller Module

- Redirects to corresponding strategy's quiz
- Uses strategy switching logic for student

- Existing Systems
- Guided Discovery Learning
- ITS Support for Guided Discovery
- Architecture and Modules
- Integration of the System
- Conclusion and Future Work

# Integration of the System

- Common database
- Developed individual systems
- Controller module
- Strategy switching logic

# Integration of the System-Adding new strategy

## Adding new strategy

- Implement quiz module
- Edit controller module
- Edit strategy switching logic

# Sequence Diagram for student

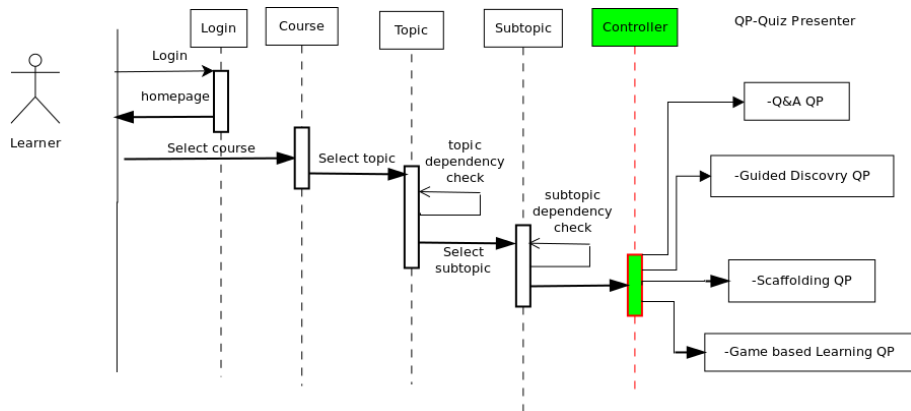


Figure: Sequence Diagram for student



# Challenges

- Interdisciplinary area
- Non-existing features
- Choosing teaching-learning strategy
- Common database
- Mapping teaching-learning steps to software system

- Existing Systems
- Guided Discovery Learning
- ITS Support for Guided Discovery
- Architecture and Modules
- Integration of the System
- Conclusion and Future Work

# Conclusion

- Developed ITS framework using guided discovery
- Integrated 3 strategies

## Limitations

- MCQs only
- No collaborative learning
- Response time not considered

# Future work


- Improved strategy switching algorithm
- Add more strategies
- Subjective questions- Latent semantic analysis
- Introducing Artificial Intelligence- SmartTutor
- Collaborative learning
- Response time

Thank You


# Key References

 Artificial intelligence. <http://www.aaai.org/AITopics/html/tutor.html>.

 Wikipedia. [http://en.wikipedia.org/wiki/Intelligent\\_tutoring\\_system](http://en.wikipedia.org/wiki/Intelligent_tutoring_system).

 Patrick Chipman, Andrew Olney, and Arthur C. Graesser. In The Autotutor3 Architecture, University of Memphis, USA.

 Albert T. Corbett, Kenneth R. Koedinger, and John R. Anderson. Handbook of Human-Computer Interaction, chapter 37. USA, 1997.

 Farhad Soleimanian Gharehchopogh and Zeynab Abbasi Khalifelu. Using intelligent tutoring systems in instruction and education. In 2nd International Conference on Education and Management Technology, volume 13, Singapore, 2011.

 Jong Suk Kim. The Effects of a Constructivist Teaching Approach on Student Academic Achievement, Self- concept, and Learning Strategy. Asia Pacific Education Review by

# Key References



Brown Ann L. and Campione Joseph C. Classroom lessons: Integrating cognitive theory and classroom practice, chapter 9, pages 229241. Cambridge, MIT Press, 1994.



Tom Lord, Holly Travis, Brandi Magill, and Lori King. Comparing Student-Centered and Teacher-Centered Instruction in College Biology Labs. Indiana University of Pennsylvania, Indiana, 2005.



Richard E. Mayer. Should There Be a Three-Strikes Rule Against Pure Discovery Learning? California, Santa Barbara. University of



Erica Melis and J rg Siekmann. In ActiveMath: An Intelligent Tutoring System for Mathematics, German Research Institute for Artificial Intelligence, Germany.



Antonija Mitrovic. An intelligent sql tutor on the web. University of Canterbury, New Zealand, 2003.

# Key References



Muska Mosston and Sara Ashworth. Teaching Physical Education. 2008.



Tom Murray. Authoring intelligent tutoring systems: An analysis of the state of the art. In International Journal of Artificial Intelligence in Education, Computer Science Dept., University of Massachusetts, 1999.



National Extension Water Outreach Education. Explanation of Teaching Continuum.



L Jean Piaget. To Understand Is To Invent. The Future of Education. Grossman publishers., NEW YORK, 1976.



Michael J. Prince and Richard M. Felder. Inductive teaching and learning methods: Definitions, comparisons and research bases.



Catherine J. Rezak. In Improving Corporate Training Results with Discovery Learning Methodology.