The Realization of Students Ranking Assessment Management Information System

Zhiqiang Yao
Dept. of Computer Science, North China Institute of Aerospace Engineering, Langfang, China
E-mail:yaonciae@126.com

Abhijit Sen
Dept. of Computer Science & Information System, Kwantlen Polytechnic University, Canada
Surrey, Canada V3W 2M8
E-mail: abhijit.sen@kwantlen.ca

Abstract—**Ranking is one of the effective techniques to measure and evaluate an individual student's performance relative to other students. The ranks are calculated based on data collected over several semesters. Ranks can provide feedback to students regarding their academic standing and may be used for nominating students for scholarships and other awards. This paper describes the design and implementation of computerized ranking assessment system which is highly flexible and can easily be integrated with any other learning assessment system. The system designed is web based and can be used to generate ranking trend chart for each student. This chart will highlight the student's performance over various semesters and will enable students to take remedial actions to improve their performances in future. Test results show that ranking assessment data can be easily generated and delivered through web using the proposed design of the ranking assessment system.**

Index Terms—**Ranking assessment system, Web, Stored procedure, User-defined function, Chart, Learning assessment system**

I. Introduction

A learning management system (LMS) [1] is a software application for the administration, documentation, tracking, and reporting of training programs, classroom and online events, e-learning programs, and training content. Learning assessment management information system (LAMIS) [2] is one important sub system of LMS. LAMIS provides on-line web-based assessment system to facilitate access to various learning performance related to indicators, such as students' appraisals [3], transcripts, and skills-gap analysis data [4]. Most LMSs are web-based, developed using a variety of platforms, like Java/J2EE, Microsoft .NET or PHP. They usually employ database such as MySQL, Microsoft SQL Server [5] or Oracle as back-end.

One of the main yardsticks [2] in assessing the students' learning in school is their rank in class. The trend of the rank in class in the latest semesters provides pointer as to the degree of effort a student puts to improve the performance to become the top student. Critical components of any good students learning assessment management information system are the design of database and its implementation.

The learning assessment system must provide simple mechanism for users to query the data base for extracting student ranking score and generate graphically the ranking trend among the latest several semesters. In this article, we have presented the design of the database using Microsoft SQL Server, to store and retrieve data [6] as requested by other software applications. Microsoft Visual Studio development environment (IDE) [7] is used to develop applications using C# programming language for the learning assessment system.

II. Database Design And Implementation

Microsoft SQL Server 2000 [5] is used to design the database named StudMISDB, which contains the following two tables:

- TB_SCORE - stores scores for each student in each course on each semester
- TB_SUMSCORE - stores cumulative score for students on each semester.

The designs of these two tables are shown in Table Ⅰ and Table Ⅱ.

A. Stored Procedure Design and SQL Server Job Scheduling

Project Number: No.2011011019 of Langfang Scientific Technology Bureau, Hebei province
Corresponding author: Zhiqiang Yao
Dept. of Computer Science, North China Institute of Aerospace Engineering, Langfang, China 065000 E-mail:yaonciae@126.com

doi:10.4304/jsw.8.3.633-638

TABLE I.
DESIGN OF THE DATA TABLE :TB_SCORE

ColumnName	Data Type	Remarks
scoreID	int	Automatic Increase by 1,ID for table data
studID	Varchar(30)	Students identity number
studName	Varchar(30)	Students' name
classID	Varchar(30)	Class Identity number
courseID	Varchar(30)	Courses' Identity number
courseName	Varchar(30)	Courses' name
Semester	Varchar(30)	Semester
Score	Float	Score of the course

TABLE II.
DESIGN OF THE DATA TABLE : TB_SUMSCORE

ColumnName	**Data Type**	**Remarks**
studID	Varchar(30)	Students identity number
classID	Varchar(30)	Class Identity number
Semester	Varchar(30)	Semester
SumScore	Float	Total Score of the student in one semester

Database interaction is done using one T-SQL [5] based stored procedure that aggregates the students score and inserts the aggregation result into the table table TB_SUMSCORE. The stored procedure on the SQL server, is executed within one SQL job[5] (A SQL Server job is a collection of steps executed by SQL Server Agent and it has many different options that can be configured during its creation in order to perform the duties that are needed of it). This SQL job [8] is deployed and it is scheduled for execution automatically every three months on the first Sunday of that month. The score code of the stored procedure proc_aggregate is shown as follows.

```
CREATE proc proc_aggregate
as
begin
delete from tb_sumscore
insert into tb_sumscore
select studid,classid,semester,sum(score) from tb_score where convert(int,substring(semester,1,4))+2>=datepart(yy,getdate())
group by studid,classid,semester
end
```

As shown above, the old aggregated data in the table TB_SUMSCORE is emptied firstly and then the latest data in the table TB_SCORE is aggregated and inserted into the table tb_sumscore. Only the latest 2 years data is aggregated in order to do the assessment meaningfully.

Fig.1 shows the job in the database server. The scheduling of the job is shown in Fig. 2 and Fig. 3.

B. Design the User-Defined Function and Fetch Clause

The core stored procedure [9] is proc_getrankdata designed to query the student's rank result. In the key query clause, i.e. the fetch clause, we integrate one user-defined function[5] func_getrank, which accepts the 3 parameters, the student's ID, the student's class ID and the named semester, and returns the rank [10] result.

- function func_getrank - returns the value, which is the calculation of the total numbers of the classmates whose total score is greater than the named students, and the stored procedure
- proc_getrankdata - this stored procedure is called to display the rank data in his class in the latest four semesters

The source code for func_getrank is shown as follows.

```
CREATE function [dbo].[func_getrank] (@studID varchar(20),@classID varchar(20),@semester varchar(12)) returns int
as
begin
declare @cnt int
declare @sum float
select @sum=sumscore from tb_sumscore where studID=@studID and semester=@semester
if exists(select sumscore from tb_sumscore where semester=@semester and classID=@classID and sumscore>@sum)
begin
select @cnt=count(sumscore)+1 from tb_sumscore where semester=@semester and classid=@classID and sumscore>@sum
end
else
set @cnt=1
return @cnt
end
```

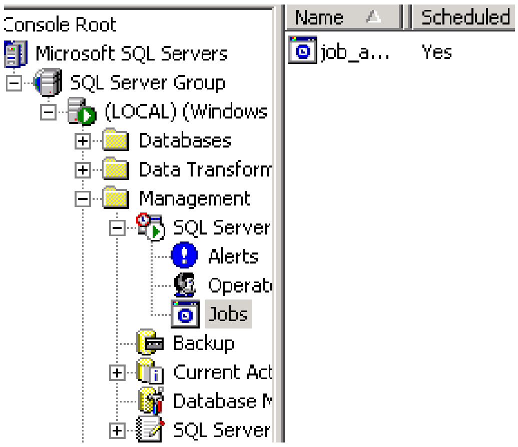
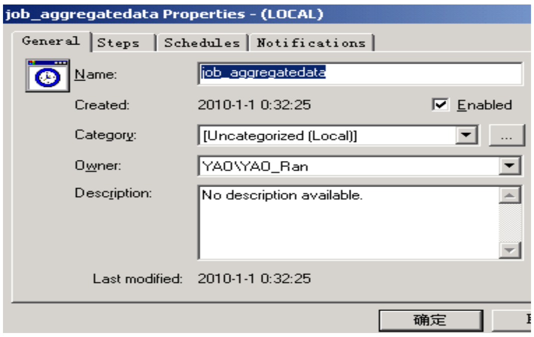


Figure 1. The job in the database server


job_aggregatedata Properties - (LOCAL)

Name: job_aggregatedata

Created: 2010-1-1 0:32:25 ☑ Enabled

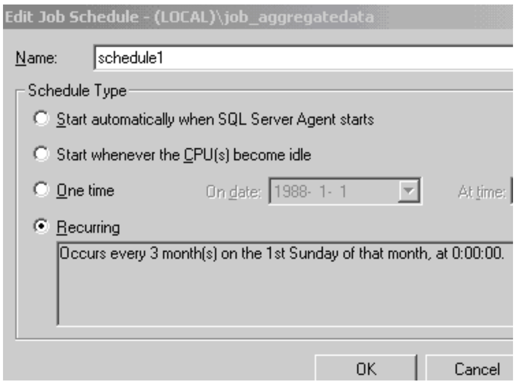
Category: [Uncategorized (Local)]

Owner: YAO\YAO_Ran

Description: No description available.

Last modified: 2010-1-1 0:32:25

Figure 2. The scheduling of the job (to be continued)



Edit Job Schedule - (LOCAL)\job_aggregatedata

Name: schedule1

Schedule Type

○ Start automatically when SQL Server Agent starts

○ Start whenever the CPU(s) become idle

○ One time On date: 1988- 1- 1 At time:

● Recurring

Occurs every 3 month(s) on the 1st Sunday of that month, at 0:00:00.

Figure 3. The scheduling of the job

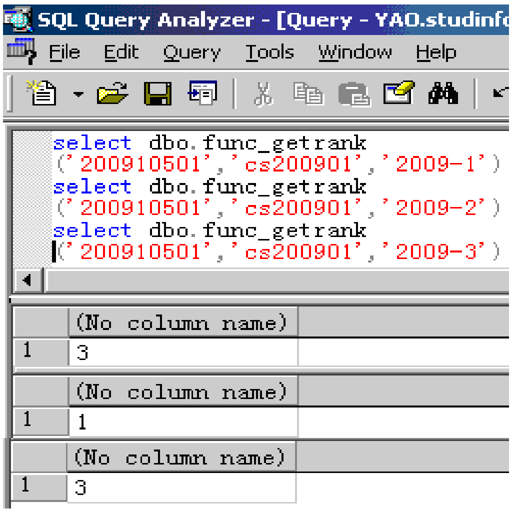
With the completion of the user defined function [11], we can call this function in the stored procedure, which provides the fetch service [12] from the database, and the parameters of this function are sent by the table's fields, which are studID, ClassID and semester, and the return value are correalted to each row of the table. The following is the source code for proc_getrankdata.

```
CREATE proc proc_getrankdata
@studID varchar(30)
As
select top 4 semester,dbo.func_getrank(studID,classID,semester) as
'rank' from tb_sumscore where studID=@studID order by semester desc
```

C. Testing the Stored Procedure and User-Defined Function

The system [13] is tested by populating table TB_SCORE with the sample student data [14] shown in the appendix. Both the user defined function [11] and the stored procedure are tested using the Query Analyzer and the test result is shown in Fig.4 and Fig.5.

As can be seen from Fig.5, the student whose student ID is 200910501 ranks third in the third semester of 2009, ranks first in the second semester of 2009 and ranks third in the first semester of 2009; and then we can analyze that this student's rank in his class is neither in the up mode nor in the down mode. It is in an unsteady wave mode [15].



SQL Query Analyzer - [Query - YAO.studinfo

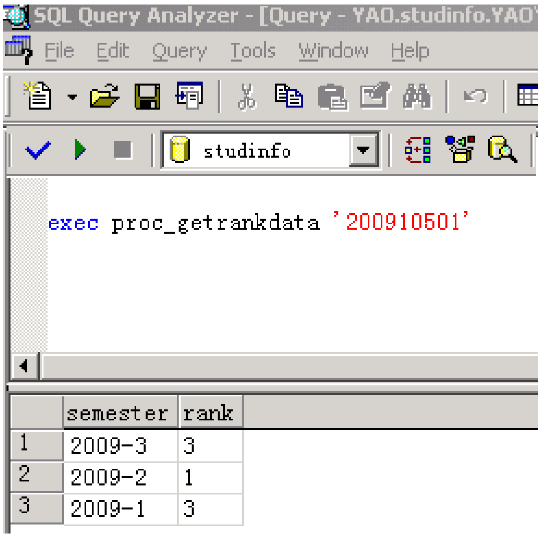
```
select dbo.func_getrank
('200910501','cs200901','2009-1')
select dbo.func_getrank
('200910501','cs200901','2009-2')
select dbo.func_getrank
('200910501','cs200901','2009-3')
```

```
  (No column name)
1 3

  (No column name)
1 1

  (No column name)
1 3
```

Figure 4. The test of the user defined function



SQL Query Analyzer - [Query - YAO.studinfo.YAO

```
exec proc_getrankdata '200910501'
```

```
  semester rank
1 2009-3   3
2 2009-2   1
3 2009-1   3
```

Figure 5. The test of the stored procedure

III. Software Architecture Of Students Ranking Assessment System

Dynamic applications are applications that build content "on the fly" in response to request made. They can reflect one named student's latest trend of rank in class from the database.

Internet Information Services (IIS) [16] – formerly called Internet Information Server – is a web server application [17] and set of feature extension modules created by Microsoft for use with Microsoft Windows. IIS 7.5 supports HTTP, HTTPS, FTP, FTPS, SMTP and NNTP. It is an integral part of Windows Server family of products, as well as certain editions of Windows XP [18], Windows Vista and Windows 7. IIS [16] is not turned on by default when Windows is installed. The configuration of the IIS is shown in Fig.6, which shows one virtual directory--studinfo, whose web files include queryRank.aspx, queryRank.aspx.cs and web.configure and it is started when the operation system is running.

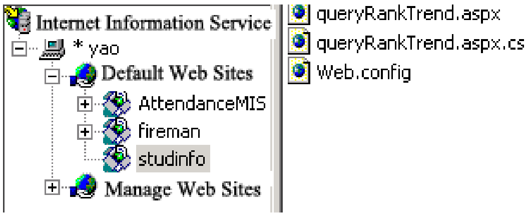


Figure 6. The configuration of IIS

A. Conceptual Software Architecture

The conceptual software architecture [19] is shown in Fig.7 and the architecture contains the following components:

- Database server, which serves the data storage [20], accepts the requests from the web server, sends the responses to the web server, and starts the SQL Server engine [21], whose function is to maintain the deployment of the SQL job.

- Web server, which is realized by the configuration of the Internet Information Server (IIS), gets data directly from database server. ASP.NET framework [22] is used to develop web application and services which sends requests to the database server and accepts the responses from the database server and delivers data to the client(PC, Phone, etc.). The students ranking assessment system [23] is deployed on the web server.

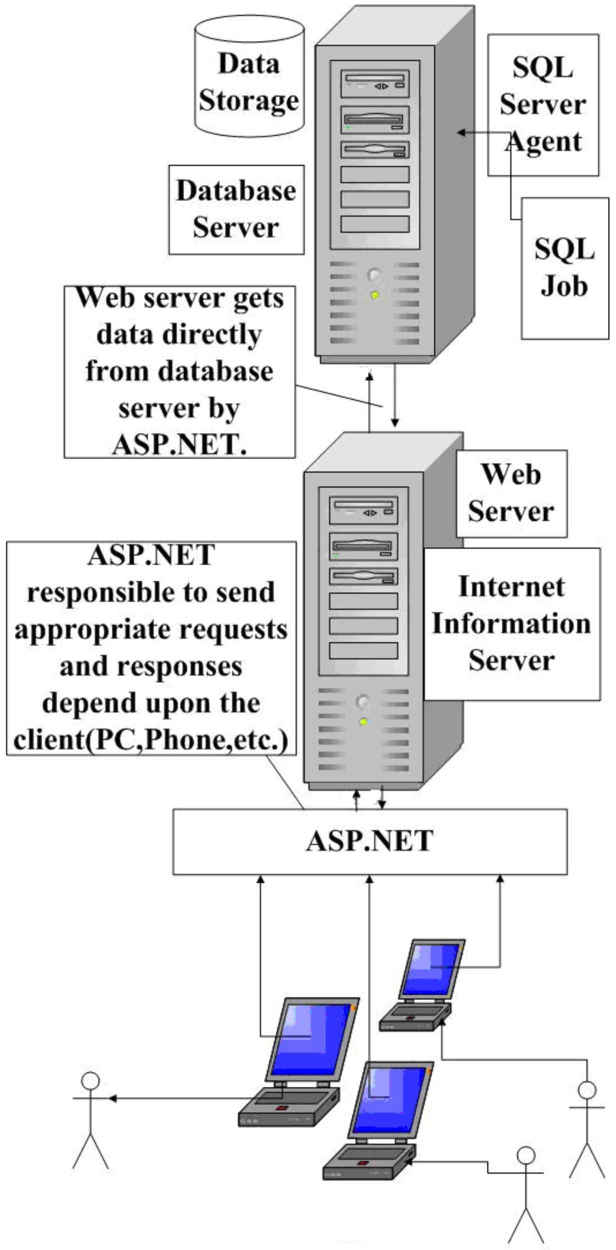
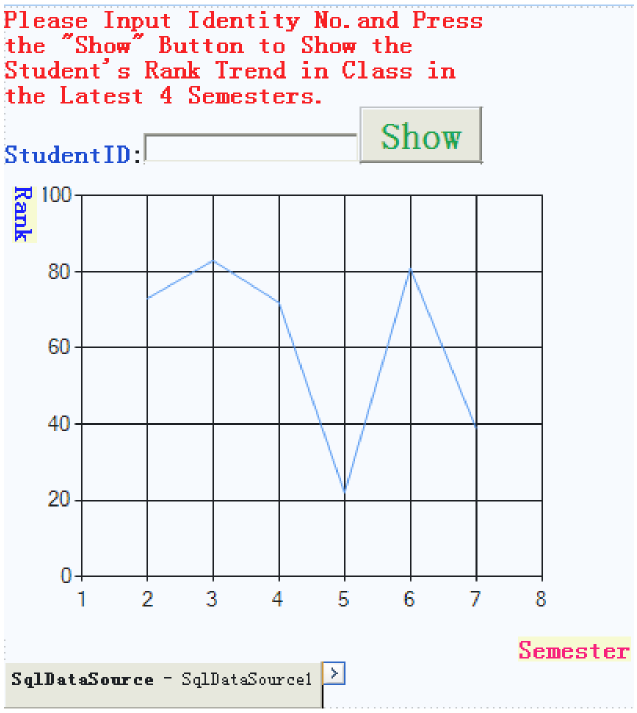

Data Storage
Database Server
SQL Server Agent
SQL Job
Web server gets data directly from database server by ASP.NET.
Web Server
Internet Information Server
ASP.NET responsible to send appropriate requests and responses depend upon the client(PC,Phone,etc.)
ASP.NET

Figure 7. The conceptual architecture of students ranking assessment system

B. Design of User Interface

The user interface [22] is shown in Fig.8. It contains the following controls:

- One input textbox for inputting Student ID
- One data source control [22] SqlDataSource1 for connecting to database and execute one command of the stored procedure, which accepts one parameter of student ID from the textbox control
- One chart control [22] for displaying rank information graphically with x-axis data is set to semester and y-axis data is set to rank in class.
- One button control [22] SHOW that executes the related stored procedure [9] proc_getrankdata to display the results graphically.


Please Input Identity No. and Press the "Show" Button to Show the Student's Rank Trend in Class in the Latest 4 Semesters.

StudentID:

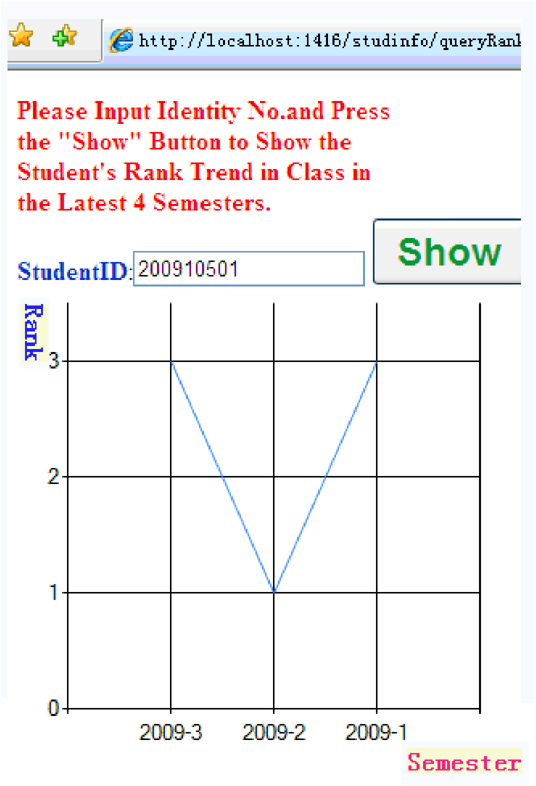
Figure 8. The design of the webpage

IV. Testing Of The Programs

In order to test programs, web server [24] is configured and test results depicting the individual ranking trend charts are shown for two students in Fig.9 and Fig.10. The chart displays the semester on x-axis and student rank on y-axis.

V. Conclusion

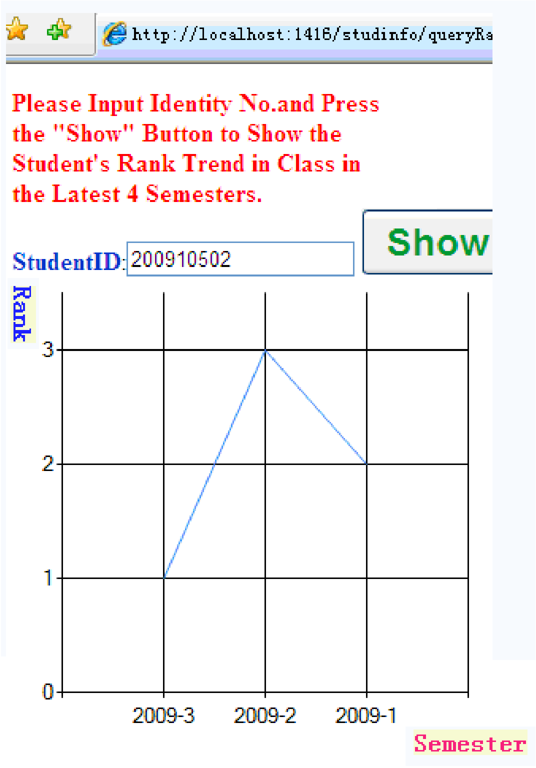
In this paper, we have discussed the design and implementation of the student ranking assessment [25] management information system. We have proposed a web based solution interacting with Microsoft SQL server database to produce charts depicting students' rank in their class every semester. A benefit to our approach is that with the help of the chart control, the trend of the students learning results can be shown clearly, and it provides a good feedback to the students, who will take measures to improve their ranks in future. This conceptual design framework can be easily integrated in any other teaching or learning management information systems [26].



Please Input Identity No.and Press the "Show" Button to Show the Student's Rank Trend in Class in the Latest 4 Semesters.

StudentID: 200910501

Figure 9. Test of the program (to be continued)



Please Input Identity No.and Press the "Show" Button to Show the Student's Rank Trend in Class in the Latest 4 Semesters.

StudentID: 200910502

Figure 10. Test of the program

Appendix A SAMPLE DATA FOR THE PROGRAM TESTING

Sample data of the table TB_SCORE is shown in the following tableIII.

TABLE III.
SAMPLE DATA OF THE TABLE TB_SCORE

studID	studName	classID	Semester	Course	Score
200910501	Zhang	cs200901	2009-1	Database	89
200910501	Zhang	cs200901	2009-2	C++	85
200910501	Zhang	cs200901	2009-1	Maths	78
200910501	Zhang	cs200901	2009-1	English	93
200910501	Zhang	cs200901	2009-2	ASP.Net	77
200910501	Zhang	cs200901	2009-2	Algebra	99
200910501	Zhang	cs200901	2009-3	Computer	80
200910502	Li	cs200901	2009-1	Database	82
200910502	Li	cs200901	2009-2	C++	89
200910502	Li	cs200901	2009-1	Maths	97
200910502	Li	cs200901	2009-1	English	90
200910502	Li	cs200901	2009-2	ASP.Net	85
200910502	Li	cs200901	2009-2	Algebra	70
200910502	Li	cs200901	2009-3	Compile	78
200910502	Li	cs200901	2009-3	computer	99
200910503	Zhao	cs200901	2009-3	Computer	90
200910504	Wang	cs200901	2009-1	Database	99
200910504	Wang	cs200901	2009-2	C++	69
200910504	Wang	cs200901	2009-1	Math	87
200910504	Wang	cs200901	2009-1	English	95
200910504	Wang	cs200901	2009-2	ASP.Net	79
200910504	Wang	cs200901	2009-2	Algebra	97
200910504	Wang	cs200901	2009-3	Computer	90

ACKNOWLEDGMENT

The author, Zhiqiang Yao wishes to thank all the reviewers for their valuable comments and suggestions that improved the quality of this paper; what's more, he wishes to thank Abhijit Sen for his good advice on updating this paper and for his necessary support in teaching and scientific research. Thanks a lot for his great cooperation in this paper.

This research is supported by the project (No.2011011019) of Langfang Scientific Technology Bureau, Hebei province.

REFERENCES

[1] Ellis, Ryann K. (2009), *Field Guide to Learning Management Systems*

[2] Haggai Kupermintz, Richard E. Snow (1997), "Enhancing the Validity and Usefulness of Large-Scale Educational Assessments: III. NELS: 88," Mathematics Achievement to 12th Grade, A*merican Educational Research Journal,* March 20, 1997; vol. 34, 1: pp. 124-150.

[3] Erin M. Snyder, Shane A. Snyder,etc(2000), "A scoring and ranking system for persistent, bio-accumulative, and toxic substances for the North American Great Lakes," *Environmental Science and Pollution Research* Volume 7, Number 1, pp.52-61

[4] David E. Wiley, Bokhee Yoon (1995), *Teacher Reports on Opportunity to Learn: Analyses of the 1993 California Learning Assessment System (CLAS) Educational Evaluation and Policy Analysis*, September 21, 1995; vol. 17, 3: pp. 355-370.

[5] Tobias Thernstrom, Ann Weber, Mike Hotek, *MCTS Self-Paced Training Kit (Exam 70-433): Microsoft® SQL Server® 2008—Database Development*, Microsoft Press, February 2009, pp:306-453

[6] Tao Tan,Hongjun Chen," A Personalization Recommendation Method Based on Deep Web Data Query", *Journal of Computers,* Vol 7, No 7 (2012), 1599-1606

[7] Scott Guthrie. "VS 2008 Web Designer and CSS Support". Retrieved 2007-12-31.

[8] DAI Wan-rong, XIONG Qian-xing, "The Study and Application of SQL SERVER Data Synchronization Mechanism", *COMPUTER KNOWLEDGE AND TECHNOLOGY*, 2009, 5(19)

[9] CAO Yaohui, "Application with SQL Server Procedure in System Development", *Modern Electronics Technique,* 2005 28(20)

[10] Zehui Zhan, Fuyin Xu, Hu Mei, "Learner Requirement Analysis based on the Classification Framework of Online Learning Activities: A Mixed Method Research", *Journal of Computers,* Vol 6, No 12 (2011), 2623-2630

[11] V. Linnemann, K. Kflspert, P.Dadam, P.Plstor(1988), "Design and Implementation of an Extensible Database Management System Supporting User Defined Data Types and Functions," *Proceedings of the 14th VLDB Conference Los Angeles, California 1988* pp294-305

[12] H. Pirahesh and B. Reinwald, "SQL table function open architecture and data access middleware." In SIGMOD, 1998.

[13] Scott Elliot (2011),*Computer-Assisted Scoring of Performance Tasks for the CLA*

[14] Jaques, P. A., & Viccari, R. M. (2006). "Considering students' emotions in computer-mediated learning environments. In Z. Ma (Ed.)," *Web-based intelligent e-learning systems: Technologies and applications* (pp. 122-138). Hershey, PA: Information Science Publishing.

[15] O'Brien, J (1999). *Management Information Systems – Managing Information Technology in the Internetworked Enterprise*. Boston: Irwin McGraw-Hill.

[16] Ken Schaefer,Jeff Cochran,etc. (2011), *Professional IIS 7,* Wiley Publishing, Inc. pp40-75

[17] M. Birenbaum, K. Breuer; E. Cascallar,etc(2006), "A Learning Integrated Assessment System", *Educational Research Review* pp.61–67

[18] LIU Lian, YANG Guokai , *Windows 2000/XP NetWork Construction Technology Application (in Chinese)*, Beijing Hope Electronics Press , Dec. 2002 241-276

[19] Hua Wang "Web-Service Based English Test Score Query System", *2012 International Conference on Electrical Energy and Networks, Chengdu, China. January 7-8, 2012*

[20] IBM Corporation, *DB~ Universal Database Application programming guide*, Version 5, 1997.

[21] Jim Melton , Alan R. Simon, *Understanding the new SQL: a complete guide,* Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993

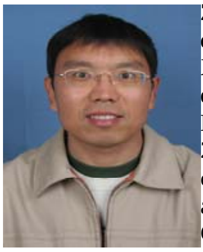
[22] Geoffrey T Leblond,Michael A Kittel, *ASP.Net 2.0 Cookbook*, O'Reilly Media , 2005-12-01

[23] Cascallar, E. C. (2004), "A new advanced integrated assessment system, " *Paper presented at the Avignon international invitational conference on assessment, Avignon, France* (September 6–8).

[24] Zhou Jianyun, Zhong Keding, Zheng Binbin2010, "Networked Learning Assessment Based on Wiki Technology, " *DISTANCE EDUCATION IN CHINA*

[25] Peter Elbow(1993), "Ranking, Evaluating, and Liking:Sorting out Three Forms of Judgment", *College English Vol. 55,* No. 2 (Feb., 1993), pp. 187-206

[26] Effy Oz , *Management Information System Six Editon,* Course Technology, January 2008, pp.208-254



Zhiqiang Yao received the 4-year bachelor degree of Mechanical Engineering from Xi'an JiaoTong University in 1994 and earned the 3-year master of Hydraulic Engineering from Tsinghua University in 2000. His major field of study is on computer programming and computer application.He was an MCSD (Microsoft Certified Solution Developer).

He has 4 years experiences of working in IT companies and during that time, he was responsible for developing the e-learning systems. In 2003, he worked at the Research Institute of the Armed Police Force Academy and did several scientific research projects. In 2009, he began to work at Computer Science Department of North China Institute of Aerospace Engineering, and he taught the First Course of Database, ASP.Net Programming, and XML and so on.

Associate Professor Yao published more than 10 papers, 4 of which were indexed by EI.

Abhijit Sen holds a PhD. from McMaster University, Hamilton, Ontario, Canada and Master of Science degree from University of California, Berkeley, USA, B.Tech in Electrical Engineering from Indian Institute of Technology, Kharagpur, India. He has over 25 years of academic and administrative experience at Kwantlen Polytechnic University. He served as a chair of the department for over 14 years. He is currently the Professor of the Computer Science and Information System at Kwantlen Polytechnic University, BC, Canada. Abhijit also was a visiting professor at Waikato University, Hamilton, New Zealand, and Munich University of Applied Sciences, Germany, Centre for Development of Advanced Computing, India, and North China Institute of Aerospace Engineering, China.

His current research interests are in the areas of Wireless Security, Radio Frequency Identification (RFID), Computing Education and Teaching Methodologies, Distributed Systems and Databases.