



uOttawa

L'Université canadienne  
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES  
ET POSTDOCTORALES



uOttawa

L'Université canadienne  
Canada's university

FACULTY OF GRADUATE AND  
POSTDOCTORAL STUDIES

Chen Hong

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.Sc. (Systems Science)

GRADE / DEGREE

*Faculty of Graduate and Postdoctoral Studies, System Science*

FACULTE, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Modeling Information Quality in Agent-Based E-commerce Systems

TITRE DE LA THÈSE / TITLE OF THESIS

T. Tran

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

C. Adams

S. Shirmohammadj

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

# **Modeling Information Quality in Agent-Based E-commerce Systems**

**Hong Chen**

Thesis submitted to the  
Faculty of Graduate and Post Doctoral Studies  
In partial fulfillment of the requirements  
For the Masters in System Science Degree

University of Ottawa

© Hong Chen, Ottawa, Canada, 2009



Library and Archives  
Canada

Published Heritage  
Branch

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

Bibliothèque et  
Archives Canada

Direction du  
Patrimoine de l'édition

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file* *Votre référence*  
ISBN: 978-0-494-58196-4  
*Our file* *Notre référence*  
ISBN: 978-0-494-58196-4

**NOTICE:**

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

**AVIS:**

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

■+■  
**Canada**

## **Abstract**

In agent-based e-commerce systems, most buying agents rely primarily on ranking algorithms to make purchase decisions. These algorithms are based on one or a combination of several features of goods information, such as price, seller's reputation, etc.; however, information quality is usually ignored. This leads to the problem that product information is retrieved without proper regard to their quality. Realizing the importance and necessity of information quality for buying agents, this thesis introduces an Information Quality (IQ)-based ranking approach. A model which combines information quality assessment with user expectation is proposed. An approach on representation and interpretation of information quality is also developed. Two sets of experiments are conducted to verify the applicability of the proposed model and explore the effectiveness of individual information quality dimensions. The experimental results show that information quality based ranking does lead to better user satisfaction.

**Keywords:** Information quality, agent system, e-commerce

## **Acknowledgement**

I would like to thank my thesis supervisor, Dr. Thomas Tran, for his constant guidance, motivation, and support throughout my thesis writing. I would also like to acknowledge constant encouragement and appreciation from my families through all stages of the program. Finally, I would like to extend my thanks and gratitude to all others who helped me to succeed.

# Table of Contents

<b>Abstract</b> .....	<b>ii</b>
<b>Acknowledgement</b> .....	<b>iii</b>
<b>Table of Contents</b> .....	<b>iv</b>
<b>List of Figures</b> .....	<b>vii</b>
<b>List of Tables</b> .....	<b>viii</b>
<b>List of Abbreviations</b> .....	<b>ix</b>
<b>Chapter 1 Introduction</b> .....	<b>1</b>
1.1 Introduction .....	1
1.2 Problem Description.....	2
1.3 Motivation .....	4
1.4 Thesis Objectives .....	5
1.5 Thesis Contributions .....	5
1.6 Thesis Organization.....	6
<b>Chapter 2 Related Work</b> .....	<b>8</b>
2.1 Information Quality.....	8
2.1.1 Information and Data .....	9
2.1.2 IQ Definition and IQ Dimensions .....	10
2.1.3 IQ Assessment.....	14
2.1.4 IQ Metadata.....	16
2.1.5 Models for IQ Assessment.....	17
2.1.6 IQ Research Frameworks .....	19
2.2 Software Agents in E-Commerce .....	20
2.3 Buying-Agent Systems.....	22
2.4 Summary .....	24
<b>Chapter 3 Our IQ Study in E-commerce Agent Systems</b> .....	<b>25</b>
3.1 Approach .....	25
3.1.1 Goal .....	25
3.1.2 Methodology .....	25
3.2 The Proposed IQ Dimensions .....	28
3.2.1 IQ Dimensions Recognition.....	28
3.2.2 IQ Dimensions Analysis .....	32
3.2.2.1 Relevancy .....	32
3.2.2.2 Reputation .....	33
3.2.2.3 Timeliness: .....	36
3.2.2.4 Amount of Data.....	39
3.3 The Information Metrics .....	40
3.4 The Proposed Model .....	41
3.4.1 The Proposed Architecture for Information Quality Assessment.....	41
3.4.1.1 Design Decisions Made in Our Architecture .....	42
3.4.1.2 The Proposed Architecture.....	43

3.4.2 The Proposed Model for IQ Representation and Assessment.....	48
3.5 Summary .....	51
<b>Chapter 4 System Implementation .....</b>	<b>52</b>
4.1 System Architecture .....	52
4.1.1 Introduction .....	52
4.1.2 Implementation Environment.....	52
4.1.3 Software Architecture .....	53
4.1.4 Architecture Considerations.....	54
4.2 System Implementation.....	55
4.2.1 Original Ranking:.....	57
4.2.2 Price-Based Ranking:.....	58
4.2.3 IQ-Based Ranking:.....	59
4.2.3.1 Implementation of IQ-Based Ranking .....	60
4.2.3.2 Database Design.....	62
4.3 Summary .....	63
<b>Chapter 5 Evaluation .....</b>	<b>64</b>
5.1 Experiment Setup .....	64
5.1.1 Hardware and Software Configuration .....	65
5.1.2 Experimental Data.....	65
5.1.3 Database design.....	66
5.1.4 Measurement Method.....	67
5.2 Simulation Evaluation .....	68
5.2.1 Experimental Approach.....	68
5.2.2 Experimental Goods.....	68
5.2.3 Evaluation 1: Information Quality Model.....	69
5.2.3.1 Experiment Process.....	69
5.2.3.2 Experimental Parameters.....	70
5.2.3.3 Results .....	71
5.2.4 Evaluation 2: Comparison of the Quality Dimensions .....	73
5.2.4.1 Experiment Parameters .....	74
5.2.4.2 Experiment Process.....	74
5.2.4.3 Results .....	74
5.3 User Evaluation .....	75
5.3.1 Experimental Design.....	76
5.3.2 Test User .....	77
5.3.3 Experimental Goods.....	77
5.3.3 Experimental Process .....	78
5.3.4 Evaluation 1: Information Quality Model.....	79
5.3.4.1 Experiment Parameters .....	79
5.3.4.2 Experiment Measurement and Results.....	80
5.3.5 Evaluation 2: Comparison of the Quality Dimensions .....	82
5.3.5.1 Experiment Process and Parameters .....	82
5.3.5.2 Experiment Results .....	83
5.4 Summary .....	84
<b>Chapter 6 Conclusions and Future Work.....</b>	<b>85</b>
6.1 Summary .....	85

6.2 Future Work .....	86
<b>References .....</b>	<b>88</b>
<b>Appendices .....</b>	<b>91</b>
Appendix A: Experiment Instructions for Test Users.....	91
Appendix B: Java Documentation.....	95

## List of Figures

Figure 1-1: Sample of item with less information (poor Information Quality) .....	3
Figure 1-2: Sample of item with more information (good Information Quality) .....	3
Figure 2-1: A data-driven model .....	18
Figure 2-2: A quality-driven model .....	18
Figure 2-3: Centralized agent systems sample: Amazon and Ebay .....	23
Figure 2-4: Distributed agent systems sample: Google.Shopping and Fetchnook.info ....	23
Figure 3-1: Research approach for the proposed model .....	26
Figure 3-2: Reputation component in Ebay.com .....	35
Figure 4-1: Topology of AQUAES .....	53
Figure 4-2: System Architecture of AQUAES .....	53
Figure 4-3: System view of AQUAES .....	56
Figure 4-4: AQUAES User Interface for buyers to input query .....	56
Figure 4-5: AQUAES User Interface for presenting result sets.....	57
Figure 4-6: AQUAES Goods Item Example.....	58
Figure 4-7: Example of AQUAES price-based ranking result.....	59
Figure 4-8: Example of AQUAES IQ-based ranking results.....	60
Figure 4-9: Entity Relationship Diagram of AQUAES .....	62
Figure 5-1: Goods Item table used in simulation .....	67
Figure 5-2 Experimental Results from IQ Model and Price-Based Model.....	73
Figure 5-3 Experimental Results on IQ Dimensions .....	75
Figure 5-4: Experimental Results from IQ Model and Price-Based Model .....	82
Figure 5-5: Experiment on IQ dimensions.....	84

## List of Tables

Table 2.1: Information Quality Dimensions .....	13
Table 2.2: Categories of Information Quality (IQ) .....	13
Table 2.3: Representative IQ Frameworks [1] .....	20
Table 3.1: Information Quality categories and dimensions .....	29
Table 4.1: Attributes required for Goods Items in AQUAES .....	61
Table 5.1 Hardware and Software Information of the Server .....	65
Table 5.2: Experimental Data ID .....	69
Table 5.3: Experimental Results from IQ-Based Approach and Price-Based Approach.....	72
Table 5.4: Experimental Results from Individual Information Quality Dimensions.....	75
Table 5.5: Experimental Goods and their ID .....	78
Table 5.6: Experimental Results from IQ Model and Price-Based Model .....	81
Table 5.7: Experiment on Information Quality Dimensions.....	83
Table A.1 Data collection form for first experiment.....	93
Table A.2: Data collection form for second experiment.....	94

## List of Abbreviations

<b>AQUAES</b>	Automatic (Information) Quality Assessment for E-Commerce Systems
<b>DQ</b>	Data Quality
<b>EJB</b>	Enterprise Java Beans
<b>IQ</b>	Information Quality
<b>J2EE</b>	Java 2 Enterprise Edition
<b>JSP</b>	Java Server Pages

## Chapter 1 Introduction

### 1.1 Introduction

Nowadays, many in-practice e-commerce systems, such as eBay, Amazon, etc., have been accepted by worldwide users. As a growing area of electronic commerce, these Consumer-to-Consumer applications have helped individuals to conduct their own e-business transactions. Within these e-commerce systems, software agents play a crucial role to help automate a variety of tasks, such as retrieving and filtering information, acting as a personal assistant, servicing as a domain expert, and helping people make decisions [6]. Undoubtedly, buying-agents, which aim to retrieve and filter suitable goods efficiently and economically for buyers, become an important part. In recent years, research on this area has been widely conducted [3, 27, 34].

Currently, the most prominent solutions discussed and implemented for buying agents are ranking algorithms which calculate and rank a list of goods items based on one or a combination of several simple features of these goods items and their sellers, such as goods price, seller's reputation [3, 6, 34]. Undoubtedly, people get some helps from these approaches for locating their interesting goods. However, the results are not satisfactory. These algorithms have their drawbacks: they are unsuitable for some situations, or have low efficiency in locating the required information. We will discuss these issues more detail in 1.2.

During an e-commerce transaction, information is a critical component. In general, buyers make decisions based on the information available to them, and misinformed people tend to make poor decisions [2]. Quality of information in terms of Completeness, Relevancy, Timeliness, and Reputation etc. plays a major role in supporting the decision making process of buyers. A buyer won't choose the product if its information seems incorrect, irrelevant, or un-timeliness.

In the next section, we will discuss the problem emerged in current researched ranking approaches. After outlining the potential problem, an Information Quality based approach is proposed to address the issues.

## 1.2 Problem Description

In practical e-commerce applications, given a selected product, a buying agent normally is able to find appropriate goods items for users by taking advantage of ranking algorithms, which calculate and rank goods item based on one or combination of several simple features of goods items and their sellers. The techniques widely used nowadays are price comparison and multi-attribute comparison. Nonetheless, these algorithms have their drawbacks. Two main issues exist when these ranking algorithms were implemented, which we will discuss as follows.

Firstly, these ranking algorithms are unsuitable for some circumstances. For instance, a ranking algorithm based only on reputation will be not fair for new sellers who just enter the market and therefore have low reputation, since their goods items will always get low rates whereas the items provided by a seller with high reputation will always receive high rates. Even if the ranking algorithm combines seller's reputation with goods price, it is still unfair for new sellers because they must quote very low price in order for their items to have equal rates with those items provided by sellers with higher reputation.

Secondly, these approaches have low efficiency in locating the required information. In a typical situation, after a buyer sending a request for a good, it is hard for him (or her) to find satisfactory items from the returned result sets easily and quickly. This may be because these goods items are less relevant to the buyer's requirements or because they don't have enough good information for the buyer to make decision. Let's look at Figure 1-1 and Figure 1-2 [7], which present two similar goods items: Apple iPhone.

## Chapter 1. Introduction

### Description *(revised)*

#### Item Specifics - Cell Phones & Smartphones

Carrier:	--	Technology:	--
Contract:	--	Camera:	1.2 Megapixels
Brand:	Apple iPhone	Features:	Unlocked, Bluetooth Enabled, Calendar, Color Screen, Email Access, Internet Browser, MP3 Player, SMS-Text Messaging, Speakerphone, Touch Screen
Model:	1st Generation 8GB	Condition:	Used
Type:	--		

Unlocked 1st generation iPhone available on any carrier (sim card pending). 8 Gigabytes of memory, perfect condition. Headphones, dock, charger, and connection cable included with original box.

Figure 1-1: Sample of item with less information (poor Information Quality)

### Description

#### Item Specifics - Cell Phones & Smartphones

Carrier:	T-Mobile, AT&T, Cingular, Rogers Wireless	Technology:	GSM
Contract:	Without Contract	Camera:	1.2 Megapixels
Brand:	Apple iPhone	Features:	Touch Screen, USB Interface, Speakerphone, SMS-Text Messaging, MP3 Player, Internet Browser, Email Access, Calendar, Bluetooth Enabled, GPS, Color Screen, Unlocked, Video Streaming, MMS Enabled
Model:	--	Condition:	Used
Type:	Smartphone (PDA Phone)		

Excellent Condition LOOKS BRAND NEW

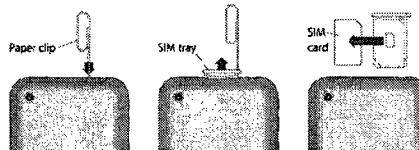
8GB Apple iPhone UNLOCKED NEW 2.0.2 Firmware  
(Compatible for any GSM Worldwide)

THEY ARE SOLD OUT EVERYWHERE!!  
NO RESERVE AND FREE SHIPPING!!

This is an ORIGINAL APPLE IPHONE  
This IS AN AUTHENTIC APPLE IPHONE 8GB

You are bidding on a used 8GB Apple iPhone with the new 2.0.2 Firmware Version. MINT CONDITION! LOOKS BRAND NEW! iPhone works with any GSM carrier worldwide including: AT&T, Cingular, T-Mobile, Rogers, Vodafone, Orange, T-Mobile, etc. No contract required, just insert your SIM Card and you are ready to go!!

UNLOCKED AND JAILBROKEN (WITH CYDIA APP INSTALLER)



I WILL SHIP SAME DAY THAT I RECEIVE PAYMENT! SATISFACTION GUARANTEED!

#### Shipping:

Free Shipping in United States! International available at the buyers expense

#### What is in the BOX:

- 8GB iPhone Device (Unlocked & Jailbroken)
- Stereo Headset
- Dock
- Dock USB Cable
- USB Power Adapter
- Cleaning/Polishing cloth
- iPhone information guide
- Firmware version 2.0.2
- Jailbroken, Unlocked and with Cydia App Installer

The phone has already been Unlocked!! All you have to do is put your SIM card in, and you are ready to make calls!! This Phone is compatible with All GSM Cellphone Carriers WorldWide! As long as you have a Sim Card for a GSM network This Phone Will Work For You! You shouldn't have to make any configurations at all. YouTube, SMS Text, WiFi and Safari works. (You will need WiFi connection or internet service with your provider for YouTube & Safari to work properly).

#### BONUS SOFTWARE: Application Installer

The phone has access via included Installer to over 200 additional FREE programs that regular iPhone users do not have access to including, Ring Tones, Games, eBooks, Applications, Images, Multimedia, Networking, Instant Messaging and other System Tools. And because this phone is unlocked, that means you can download and install even more games and applications easily and wirelessly! Something you can't do with a regular out of the box iPhone EVEN if they weren't sold out!!

#### SPECIFICATIONS

##### Capacity

8GB flash drive

##### Display

3.5-inch (diagonal) widescreen multi-touch display  
480-by-320-pixel resolution at 163 ppi  
Support for display of multiple languages and characters simultaneously

##### Operating system

iOS v

Figure 1-2: Sample of item with more information (good Information Quality)

The goods item in Figure 1-1 provides users very limited information: model, camera, some phone features, and phone condition. In contrast, if the users look at the goods item in Figure 1-2, they get much more information: contract info, shipping, capacity, etc. Suppose these two goods items have the same price and their sellers have the same reputation, the buyers will be more interested in the goods item presented in Figure 1-2. However, if we use some current algorithms, these two goods items will get the same rate due to their same price and sellers' reputation. Hence the algorithms can't distinguish them. In addition, if many goods items, as shown in Figure 1-1, are listed in the result list, it will take the buyers considerable amount of time to filter them out.

### 1.3 Motivation

Information Quality (IQ), also known as Data Quality (DQ), describes the quality of the information content of an information system in terms of its fitness for use in a particular application, such as business operations, decision making, or planning [30]. It is a pervasive concept and a key antecedent of information systems success [1]. As a recognized discipline, Information Quality has emerged over the past several years.

Actually, the two problems discussed in Section 1.2 can be analyzed and addressed using the lens of Information Quality. From the perspective of Information Quality, the goods items in Figure 1-1 and Figure 1-2 are different and will get different rates although they have similar price and seller reputation. Regarding to the first problem discussed in 1.2, since Information Quality is calculated on the whole information persisted in the goods items (which includes not only the goods price and seller's reputation, but also the amount of the information, the relevancy of the goods to the buyers etc.), new sellers will be able to have equal opportunity to get high rate if they present high quality information.

There is no question that in today's information-centric society the quality of information plays an important role. However, from our point of view, although the research on IQ has been conducted in different application areas, the research of Information Quality within agent-based e-commerce systems is still lacking.

## 1.4 Thesis Objectives

The main aim of this thesis is to provide an IQ-based approach:

- For buying agents in e-commerce systems. We present a new way to help buying agents in ranking goods items. Our approach analyzes and calculates the IQ of goods items, and therefore can be used by buying agents for ranking goods items based on their IQ value.
- To address the issues that exist in other ranking algorithms and therefore to facilitate the development of high efficiency buying agents to assist buyers in finding satisfactory goods. Buyers will be presented with as many high IQ goods items as possible.

In order to achieve these objectives, we need to:

- Study the characteristics of goods items in e-commerce system to find the IQ definition of the goods items.
- Construct a feasible framework to retrieve and assess Information Quality of goods items.
- Implement an IQ-based e-commerce system based on the proposed framework, which can simulate C2C e-commerce scenarios to evaluate the efficiency of our IQ-based approach.

## 1.5 Thesis Contributions

The main contributions of this thesis are:

- This thesis introduces the concept of Information Quality into the context of agent-based e-commerce systems and presents an IQ-based comparison approach for buying agents. We study the characteristics of Information Quality in agent systems, and recognize four Information Quality dimensions— Relevancy, Reputation, Amount of Data, and Timeliness, in an agent system for e-commerce. In addition, we show that

the proposed approach can enhance the efficiency of buying agents in finding suitable goods for buyers. This research, therefore, provides a different perspective of ranking algorithms for buying agents.

- A general framework for a query-answering model that integrates Information Quality assessment into agent-based e-commerce systems is constructed. Also, the thesis provides an implementation to evaluate the feasibility of the proposed approach.
- In reality, no serious information has data quality of 100%. We believe that the real concern with data quality is to ensure not that the data quality is perfect, but that the quality of the data in a result set is accurate enough, complete enough, and consistent enough for buyers to make reasonable decisions. Furthermore, it is not sufficient to evaluate data quality by considering only the source in which the data are contained. In addition, buyer's expectations vary along different variables and the same data can be perceived in different ways by different buyers. Therefore, in this thesis, we consider the buyers' requirements for Information Quality; that is, which IQ dimensions they care about, and to which extent they care about them. As a result, the framework proposed in this thesis allows "personalizing" the assessment procedure by combining the quality measurement together with buyer's requirements.
- An "index of data quality" is proposed to represent the data quality of items. It is a single aggregate measure of data quality combined with the buyer's expectation index. From our point of view, such an index could help to succinctly communicate the state of data quality to buyers and provide comparative assessments over time.
- We presents the design and implementation of a centralized buying agent system called AQUAES (Automatic (Information) Quality Assessment for e-Commerce Systems), which could helps buyers to look for goods items and provides comparison/ranking on the result set.
- Two types of evaluation—real user evaluation and simulation evaluation—were designed and conducted to explore the effectiveness of our proposed IQ model in different levels.

## 1.6 Thesis Organization

This thesis is organized as follows:

Chapter 1 provides a brief introduction on why Information Quality should be taken into consideration in e-commerce agent systems. The possible problems commonly faced by most ranking algorithms are also presented. We believe that the ranking approach based on Information Quality will be a good alternative for addressing the problems.

Chapter 2 discusses background knowledge and related work regarding Information Quality, software agents, and agent systems. All these issues are necessary for the investigation of the proposed framework. This chapter is intended to establish a firm understanding of Information Quality and to survey the current state-of-the-art technology in existing agent-related research.

Chapter 3 provides a detailed description of the proposed framework for the assessment and representation of Information Quality. We discuss the design methodology, recognize and analyze the IQ dimensions specific for goods items, analyze the information matrices, and illustrate the proposed model.

Chapter 4 presents a reference implementation of the framework, the AQUAES system. We discuss the basic concepts, functions, and features. Some detailed design issues are analyzed.

Chapter 5 provides an experimental evaluation of the proposed framework. We describe our process to conduct the experiments and present the evaluation results.

Chapter 6 discusses the benefits that our work brings in, discusses some future research directions, and concludes the thesis.

## **Chapter 2 Related Work**

The research on Information Quality in buying agent systems involves three areas: Information Quality, software agents, and buying-agent e-commerce systems. This chapter discusses the studies of these fields accordingly.

### **2.1 Information Quality**

As stated in [4], compared to the core database concepts such as database integrity and security, the notion of data quality (DQ) has only emerged during the past 10 years but has received a steadily increasing interest. The reason for this is the increased interconnectivity among data producers and data consumers, mainly spurred through the development of the Internet and various Web technologies. More than ever before, businesses, governments, and research organizations rely on the exchange and sharing of data. It is widely recognized that dealing with data quality problems is expensive and time consuming, leading to new IT technology branches that exclusively focus on the assessment of data quality in an organization and cleaning poor quality data.

Information Quality as a recognized discipline has been widely investigated in the literature. IQ research has been conducted in different application areas, such as information and database systems [15, 16], websites [10, 12, 17], and e-businesses [11]. Among these researches, Alexandra Martinez et al. provided an approach to count the quality of information in a database system [16]. An approach to measure the Information Quality of the website was provided in [10]. Chiara Francalanci et al. conducted the data quality assessment from a user's perspective [2]. Majki et al. presented a general framework for query answering in Data Quality-based Information Systems [15]. Kim et al. analyzed the characteristics of e-businesses and developed an E-Quality framework [11]. Lee, Younghwa

et al. investigated the effect of website quality on e-business success [13]. In the research of Xiaolan Zhu et al., IQ is used to retrieve documents in centralized and distributed internet search environments [39].

By means of a systematic review of the relevant literature, we identify that existing research has been investigated IQ definition/dimensions [2, 36, 37], modeling [17, 23, 25], assessment [4, 23, 25, 26, 39], and control [12, 26]. IQ definition /dimensions deal with the analysis of common IQ characteristics and the recognition of IQ dimensions in different fields. IQ modeling research focuses on the approach used to retrieve and provide the Information Quality to user. IQ assessment mostly involves the measurement of Information Quality. The objective of IQ control is to improve the quality of information in the application based on the IQ assessment result.

Next we will review some techniques which are involved in our study. Before we discuss the Information Quality research further, it is necessary to discuss the information and data first.

### **2.1.1 Information and Data**

In Al-Hakim's book "Information Quality Management: Theory and Application" [1], he provides the following commonly accepted view of the terms "information" and "data":

- **Data:** Items about things, events, and transactions are recorded, classified, and stored but are not organized to convey any specific meaning. Data items can be numeric, alphanumeric, figures, sounds, or images.
- **Information:** Data that have been organized in a manner that gives meaning for the recipient. They confirm something the recipient knows, or may have "surprise" value by revealing something not known.

These definitions clarify the relationship between data and information. They are consistent with the concept of information product in which information is a product of an information manufacturing system. The input for this information manufacturing is data. Similar to a product manufacturing system, an information manufacturing system is hierarchical in that information output from a certain stage can be considered data for the next stage of the information manufacturing system. From this perspective, the term information can be used to refer to both data and information.

Early literatures dealing with Information Quality as well as some recent publications use the terms “Information Quality (IQ)” and “data quality (DQ)” interchangeably. Data and information are often used synonymously. In practice, people differentiate information from data intuitively, and describe information as data that has been processed. Note that, unless specified otherwise, this thesis will use the term “data” interchangeably with “information”.

Information Quality aims at ensuring the quality of data. In the field of IQ, existing research investigates IQ definition/dimensions [2, 36, 37], modeling [17, 23, 25], assessment [4, 23, 25, 26, 39], and control [12, 26]. We will discuss these research areas below.

### **2.1.2 IQ Definition and IQ Dimensions**

According to Al-Hakim [1], customers view quality in differing criteria based on their individual roles in the production-marketing chain. Thus, it is important to understand the various perspectives from which Information Quality (IQ) is viewed. Wang [37] finds an analogy between the quality issues in product manufacturing and those in information, and further asserts that information manufacturing can be viewed as processing system acting on raw data to produce information products. Wang urges organizations to manage information as they manage product if they want to increase productivity.

Al-Hakim [1] also emphasizes that quality could be a confusing concept. He provides two main reasons for this assertion: (1) people view quality using different perspectives and

dimensions based on their individual roles, and (2) the meaning of quality continues to evolve as the quality profession grows and matures. Similar to product quality, IQ has no universal definition. To define IQ, it is important to comprehend both the perspective from which IQ is viewed and its dimensions. The Cambridge Dictionaries Online [38] defines perspective as “a particular way of considering something” and dimension as “a measurement of something”.

Individuals have different ways of considering the quality of information as they have different wants and needs. Hence, there are different quality standards which lead to a user-based quality perspective. Information users can view IQ from various perspectives, as “fitness for intended use”, “conformance to specifications”, or “meeting or exceeding customer expectations”. While these perspectives capture the essence of IQ, they are very broad and are difficult to use in the measurement of quality. There is a need to identify the dimension that can be used to measure IQ. [1]

Problems with data quality cannot be addressed effectively without an understanding of the data quality dimensions [32]. Among the studies of Information Quality, the IQ dimension is most widely studied. From the author’s perspective, there are two following reasons:

On the one hand, Information Quality, just like quality management of physical products, is a multidimensional concept, which is also confirmed by many literatures. This means that organizations must use multiple measures to evaluate the quality of their information or data. Several researchers have attempted to identify the IQ dimensions. According to the research of Wand and Wang [36], more than twenty data quality dimensions are often cited. Table 2.1 provides the definitions of the most common IQ dimensions used in the literature:

Dimensions	Definitions
Accessibility	The extent to which data is available, or easily and quickly retrievable
Appropriate	The extent to which the volume of the data

Amount of Data	is appropriate for the task at hand
Believable	The extent to which the data is regarded as true and credible
Completeness	The extent to which the data is not missing and is of sufficient breadth and depth for the task at hand
Concise Representation	The extent to which the data is compactly represented
Consistent Representation	The extent to which the data is presented in the same format
Ease of manipulation	The extent to which the data is easy to manipulate and apply to different tasks
Free-of -Error	The extent to which the data is correct and reliable
interpretability	The extent to which the data is in appropriate languages, symbols, and units, and the definitions are clear
Objectivity	The extent to which the data is unbiased, unprejudiced, and impartial
Relevancy	The extent to which the data is applicable and helpful for the task at hand
Reputation	The extent to which the data is highly regarded in terms of its source or content
Security	The extent to which the data is restricted appropriately to maintain its security
Timeliness	The extent to which the data is sufficiently up-to-date for the task at hand
Understandability	The extent to which the data is easily comprehended
Value-Added	The extent to which the data is beneficial

	and provides advantages from its use
--	--------------------------------------

Table 2.1: Information Quality Dimensions

Strong, Lee, and Wang [30] conducted an empirical two phases sorting study and provided the most comprehensive list of IQ attributes, which is reduced to 20 dimensions. These dimensions in turn are grouped into four categories as shown in Table 2.2:

<b>IQ Categories</b>	<b>IQ Dimensions</b>
Intrinsic IQ	Accuracy, Objectivity, Believability, Reputation
Accessibility IQ	Access, Security
Contextual IQ	Relevancy, Value-Added, Timeliness, Completeness, Amount of Data
Representational IQ	Interpretability, Ease of understanding, Concise representation, Consistent representation

Table 2.2: Categories of Information Quality (IQ)

It is clear that the focus of current IQ approaches has primarily involved the content of the information, the interface-related aspects of information presentation, and end-user delivery.

On the other hand, data quality problems are application specific, which is because (1) the definition of the dimension often depends on the context where the information is used, and (2) the dimension definition is also from specific user within a given context. The dimension definition is closely tied to the problem domain, the data itself, and how it is used. So, there is no general agreement on data quality dimensions, and it is necessary to define dimensions, or the aspect of dimensions for a specific application [26].

### 2.1.3 IQ Assessment

In [5], IQ assessment is defined as the process of assigning numerical or categorical values (quality scores) to quality criteria in a given data setting. In general, however, quality assessment is a difficult task due to the following reasons: First, many DQ criteria are subjective and therefore cannot be assessed automatically (e.g., trust or understandability). Second, many sources do not publish quality related metadata. Finally, for sources with large amounts of data, DQ assessment can be performed only on a sample of the data, thus resulting in a decreased precision of quality scores. Consequently, in reality IQ measurement is most implemented. IQ measurement can be considered as a special form of IQ assessment where well-defined DQ metrics are used and therefore result in that the DQ metadata, which are used, are more objective.

In Pipino, Lee and Wang's research [26], the authors study various types of metrics and the ways of combining the values of quality indicators. Objective data quality metrics are developed by using three functional forms, which are simple ratio, min or max operation, and weighted average.

**Simple ratio:** As stated in [26], the simple ratio measures the ratio of desired outcomes to total outcomes. However, since most people measure exceptions, a preferred form is the number of undesirable outcomes divided by total outcomes subtracted from 1. This simple ratio adheres to the convention that 1 represents the most desirable and 0 the least desirable score. Although a ratio illustrating undesirable outcomes gives the same information as one illustrating desirable outcomes, our experience suggests that managers prefer the ratio showing positive outcomes, since this form is useful for longitudinal comparisons illustrating trends of continuous improvement. Many traditional data quality metrics, such as free-of-error, completeness, and consistency take this form. For instance, a type of completeness is called population completeness. If a column should contain at least one occurrence of all 50 states but it only contains 43 states, then we have population incompleteness, which can be measured by taking the ratio of the number of incomplete items to the total number of items and subtracting from 1, that is  $1 - 7 / 50 = 86\%$ . Thus, the population completeness is 86%. In the

next chapter, the metrics of dimensions of Relevancy, Reputation, and Amount of Data make use of this functional form.

**Min or Max Operation:** This function form is applied to handle dimensions that require the aggregation of multiple data quality indicators (variables). One computes the minimum (or maximum) value from among the normalized values of the individual data quality indicators. The min operator is conservative in that it assigns to the dimension an aggregate value no higher than the value of its weakest data quality indicator (evaluated and normalized to between 0 and 1). The maximum operation is used if a liberal interpretation is warranted. The individual variables may be measured using a simple ratio. Two interesting examples of dimensions that can make use of the min operator are **believability** and **appropriate amount of data**. The max operator proves useful in more complex metrics applicable to the dimensions of **Timeliness** and **Accessibility**. We take believability as example. Believability is the extent to which data is regarded as true and credible. Among other factors, it may reflect an individual's assessment of the credibility of the data source, comparison to a commonly accepted standard, and previous experience. Each of these variables is rated on a scale from 0 to 1, and overall believability is then assigned as the minimum value of the three. Assume the believability of the data source is rated as 0.6; believability against a common standard is 0.8; and believability based on experience is 0.7. The overall believability rating is then 0.6 (the lowest number) [26]. In chapter 3, we will use this functional form for Timeliness dimension.

**Weighted Average:** For the multivariate case, an alternative to the min operator is a weighted average of variables. If a company has a good understanding of the importance of each variable to the overall evaluation of a dimension, for example, then a weighted average of the variables is appropriate. To insure the rating is normalized, each weighting factor should be between zero and one, and the weighting factors should add to one. [26]. A simple average can be used if a single variable must be evaluated for a set of data. Regarding the believability example mentioned earlier, if the company can specify the degree of importance of each of the variables to the overall believability measure, the weighted average may be an appropriate form to use. In our thesis, we proposed a concept of "index of data quality" which uses this form. We will discuss our proposed concept in Chapter 3.

According to [5], results of assessment can be represented in different forms. Numbers are used to represent dimension values for a certain quality criteria (e.g., a completeness of 80%). Such numbers can be aggregated to single scores, allowing a comparison of the quality of different sources or even a quality ranking. In contrast, assessment categories provide only a few values (e.g., accurate versus not accurate). They are easy to use and to interpret by the user, but are difficult to aggregate in a single score.

### 2.1.4 IQ Metadata

**Metadata (meta data, or sometimes meta information)** is "data about data", of any sort in any media. An item of metadata may describe an individual datum, or content item, or a collection of data including multiple content items and hierarchical levels, for example a database schema. In data processing, metadata is definitional data that provides information about or documentation of other data managed within an application or environment [38].

According to a report of the Dagstuhl Seminar "Data Quality on the Web" [5], metadata plays an important role in supporting quality assessment and interpretation. In the report, a basic model consisting of data sources, transformation components, and clients (users/applications) was assumed. Transformation components represent data processing units such as schema mapping and data integration services, which either influences the quality of the processed data, or where the quality of the result data depends on the quality of the input data. For both sources and transformation components, it was assumed that an assessment of data quality at a component is possible. The interpretation of a resulting quality measures is then done by the user or application that consumes the processed data. Based on the above model, a coarse classification of metadata has been developed. This classification about metadata includes the following components [5]:

- Sources, that is, information about the coverage and completeness of the source, source schemas (data model and granularity), individual objects (timeliness, lineage, accuracy etc.), and the intended usage or the domain of sources.
- Transformation services, in particular metadata that describes the signatures, parameters, and requirements of operators, compositions of transformation services, and metadata capturing trace information about objects processed in the model.
- Users, e.g., data expectation or requirements of a user, or simply uninterrupted query/answer pairs.

### **2.1.5 Models for IQ Assessment**

In an agent-based e-commerce system, when a buyer is in need of some goods, he/she will send a request to the system, which then triggers a buying agent to search suitable goods items within the system for the buyer. In this study, we will take Information Quality into consideration and applied IQ-based ranking in the buying agent. The IQ-based approach will, act as a “shadow” operator, compute the quality of goods items based on the quality values of the input data and available IQ metadata [5]. An important question arises: how buying agents provide buyers with quality information in addition to the query results. According to [5], two suggested models are given: the data-driven model and the quality-driven model.

In a data-driven model, a user queries the source as usual, while additional information about the quality of the query results is provided. That is, operators of the data quality algorithm compute the quality of each item in results while the query is processed. If we applied this model to a buying agent system, the structure would be as shown in Figure 2-1. In this structure, after receiving a request from a buyer, a buying agent will go to search the Data repository for the suitable goods items that meet the buyer’s requirement (see “Normal Query” in Figure2-1); at the same time, it make use of the data quality algorithm to calculate

the IQ of each goods items within the search result (see “IQ Assessment”). Finally, the agent sends both the search result and the IQ result to the buyer.

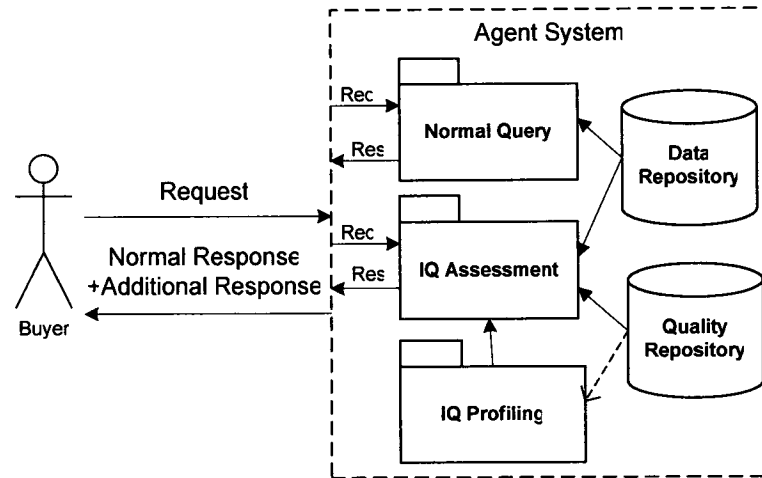


Figure 2-1: A data-driven model

In a quality-driven model, the user can specify the IQ requirements. During query processing, in order to meet the specified requirements, the system decides what sources and IQ metadata to use, calculates the IQ, and then sends the result of the IQ to the user. Consider Figure 2-2, which illustrate an agent system that applies this model. In this model, a buyer agent will calculate IQ for each item (see “IQ Assessment”) and then send the buyer with items that meet buyer’s IQ requirement. This approach is more like a query optimization problem by treating quality values as criteria.

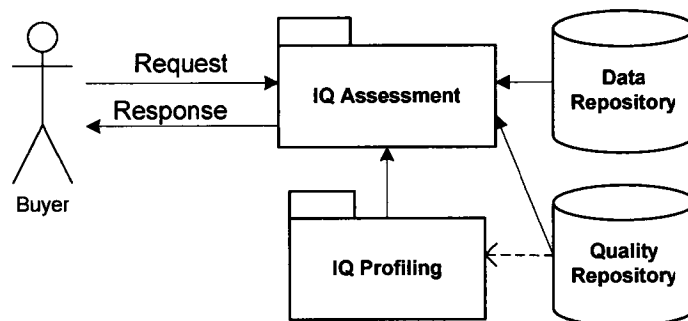


Figure 2-2: A quality-driven model

In both models, the system computes the Information Quality of the output based on the buyer's input data and the available quality-related metadata.

### 2.1.6 IQ Research Frameworks

In the book "Information Quality Management: Theory and Applications" [1], the author summarised the works conducted on IQ framework. He mentioned that personnel management and organisational factors, as well as effective technological mechanisms, affect the ability to maintain IQ. The author clarified this relationship by drawing an analogy between manufacturing and the production of data.

A number of IQ frameworks (see Table 2.3) have also been proposed to organise and structure important concepts in IQ and to provide a more complete perspective on IQ. The frameworks shown in Table 2.3 were presented in [1], which represent some existing IQ research efforts. Most of these frameworks focus on the process that produces high-quality data. Organisations can use these frameworks to assist them to better understand IQ, identify desirable IQ dimensions, and provide guideline on how to design IQ policies to improve IQ in their organisations.

Authors	Representative IQ frameworks
Shanks & Darke, 1998 Price & Shanks, 2004	A Framework of Understanding Data Quality Based in Semiotics
Giannoccaro, Shanks, & Darke, 1999	A Framework of Stakeholder Perceptions of Data Quality in Data Warehouse Environment
Kahn, Strong, & Wang 2002	The Product and Service Performance Information Quality Model
Eppler, 2001	An Interdisciplinary Evaluation Framework for Information Quality Frameworks
Caballero & Piattini,	A Data Quality Model for Data Quality

2003	Management Improvement
------	------------------------

Table 2.3: Representative IQ Frameworks [1]

Lately, Cappiello et al. [2] present a framework for data quality assessment from the user's perspective. The framework considers both user expectations and the selected services in the data quality assessment process, and consequently, selects a suitable evaluation function to obtain a correct interpretation of results. This framework is a supplementary to the above frameworks. It focuses on the process that produces high-quality data. Our research on the model for IQ assessment and IQ presentation is motivated by this work.

## 2.2 Software Agents in E-Commerce

As stated in Wikipedia [38], e-commerce consists of the buying and selling of products or services over electronic systems such as the Internet and other computer networks.

With the development of e-commerce, intelligent software agents (or "agents" for short) are widely researched. A multi-agent system (MAS) can be defined as a network of multiple agents that interact with one another. Within this system, an autonomous and intelligent agent is well suited for many areas; for example, managing the information overload, retrieving and filtering information, acting as a personal assistant, servicing as a domain expert, and helping people make decisions [6]. In the research of multi-agent systems for e-commerce, the technology applied in buying-agents, which aims to search and find suitable goods efficiently and economically for buyers, is a crucial part. This thesis also concentrates on the study of this technology.

As the tasks of agents that are involved in e-commerce, the authors of the paper "Agents in E-Commerce" [6], He and Leung, said: "We employ the Consumer Buying Behavior (CBB) model (Guttman et al., 1998) to capture consumer behaviour in order to analyze the tasks of agents involved in e-commerce. There are six stages in the CBB model: (1) need identification, (2) product brokering, (3) merchant brokering, (4) negotiation, (5) purchase

and delivery, and (6) product service and evaluation. From the CBB model perspective, agents can act as mediators in consumer behaviour in buying commodities in three stages: product brokering, merchant brokering, and negotiation. In the product brokering stage, on behalf of consumers, agents can determine what to buy according to different product information. In the merchant brokering stage, a shopping agent can find, on behalf of a customer, an appropriate merchant given a selected product. In the negotiation stage, two or more parties bargain with each other to determine the price or other terms of the transaction on the Internet.”

The study of this paper based on the premise that the user has known exactly what they want and then let the agent help them to find the appropriate goods. Consequently, the study will focus on the merchant brokering stage.

In the merchant brokering stage, the technique widely used in buying agents is price comparison. Price comparison is used to compare the prices of a product from different merchants, and select the cheapest one. BargainFinder is the first system using an agent for on-line price comparison. For example, if a customer wants to buy a music CD, BargainFinder will request its price from different merchants, and select the merchant with the lowest price for the customer [6].

However, price is not always everything. Considering price only causes other value-added services from merchants to be ignored, and thus it fails to offer enough chances for merchants to differentiate themselves in competitive markets. Moreover, a customer cannot have an option to choose between the price of a product and its other features (e.g. its delivery time and extended warranty). Thus, multi-attribute comparison is needed in shopping agents. With this kind of shopping agent, it is easier for customers to perform cross-merchant product comparisons [6]. Actually, more and more studies have focused on this research. Jango present the opportunity to balance the model and make of a product. Tete-a-Tete (T@T) allows buyers to have more control over the attributes of the product required. In fact, buyers can specify the importance of each attribute. The returned rating list

will then show how well a product fits the buyer's requirements [6]. Jonker et al. present a multi-attribute negotiation with efficient use of incomplete preference information [9].

More recently, attention was focus on the attribute of seller's reputation. According to Sabater [28], Reputation is "the opinion or view of one about something". In electronic marketplaces, the Reputation that a user has is the result of aggregating all the impressions of the other users that interacted with him/her in the past. Amazon Auctions<sup>1</sup>, eBay<sup>2</sup>, for instance, are online auction houses where users buy and sell goods. Each time a new transaction is finished, the buyer rates the seller. These ratings are used to build the Reputation of a seller. Buying agents are then able to use these Reputation values to compare on goods items, or combine Reputation values with other goods on sale attributes to compare goods items. For instance, Tran and Cohen study the effectiveness when combine seller reputation with goods quality [34].

Apparently, these approaches analyze agent behaviours and goods attribute; however, none of them addresses the notion of Information Quality, that is, they do not take into account the characteristics of information itself. Actually, Information Quality is realizing the strategic role in decision-making activities [18]. A buyer won't choose the product if its information seems incomplete, irrelevant, or un-timeliness. This builds the ground of this research, that is, we would like to take the Information Quality into the research of the merchant brokering stage of buying agent and apply it to help search and rate the goods, and then explore the feasibility of the idea.

## 2.3 Buying-Agent Systems

From our perspective, current in-practice developed buying-agent systems can be classified into two broad categories based on their architectures: 1) centralized; and 2) distributed. Centralized agent systems typically require that all the goods information reside locally at a

---

<sup>1</sup> Amazon Auctions: <http://www.amazon.com>

<sup>2</sup> eBay.: <http://www.ebay.com>

single site and all queries are also handled by that site. Ebay.com [20] and Amazon.com [19] (shown in Figure 2-3) typically belong to this type. All the transaction will also happen in the local system.

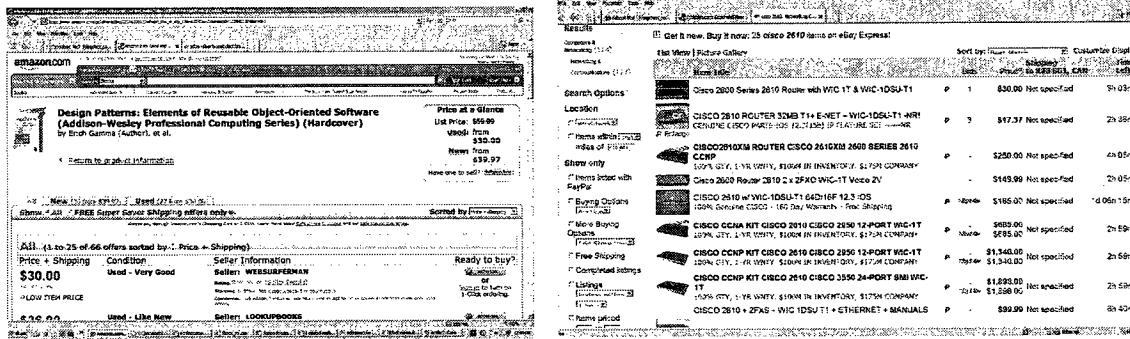


Figure 2-3: Centralized agent systems sample: Amazon and Ebay

In contrast, distributed agent systems allow agents to simultaneously access goods information distributed across multiple remote sites, such as Google.Shopping [22] and fetchbook.info [21] (as illustrated in Figure 2-4). In the local site, there are only some basic information about the goods, such as seller/other store's name, price, and the link to the original site that the goods information is persisted. Users also have to go to other sites to get their transactions completed.

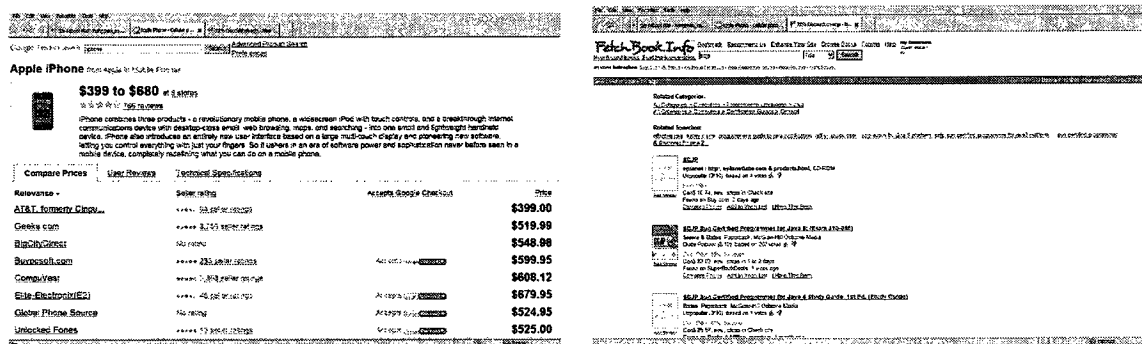


Figure 2-4: Distributed agent systems sample: Google.Shopping and Fetchnook.info

Based on our analysis, no quality research on Information Quality has been conducted on both categories of systems. The other conclusion we have drawn is that the measurement of Information Quality in centralized buying-agent systems will be easier than that in distributed

buying-agent systems. This is because in centralized systems, all data of goods items are persisted as the same data schema in one database, which make it easy for buying agents to retrieve data and calculate Information Quality of individual goods item; whereas in distributed systems, goods data are persisted in different subsystems and with different data schemas, requiring an extra step from the buying agents before they could calculate the Information Quality for each goods item. That is, the buying agents need to have knowledge of separated databases to identify which data to collect and where to retrieve them. Therefore, as the first step of research, this thesis deals with the study of Information Quality in centralized e-commerce agent systems.

## **2.4 Summary**

In this chapter, current issues in Information Quality, agent technology, and buying-agent systems were presented. We discussed in detail the different concepts and aspects involved in these issues. In addition, challenging issues were pointed out, as well as the directions of current research. The next chapter is dedicated to describe the design of our proposed model.

## **Chapter 3 Our IQ Study in E-commerce Agent Systems**

This chapter presents the design of a framework of Information Quality assessment and representation. First, the steps that apply the Information Quality into an agent-based e-commerce system are identified. Secondly, the specific IQ dimensions are recognized and analyzed, followed by the discussion of, the information matrices which are used in our research. Finally, we describe and illustrate our proposed model.

### **3.1 Approach**

Based on the thesis objectives outlined in Section 1.4, we further discuss the goal and the methodology of designing the framework in this section.

#### **3.1.1 Goal**

Our goal of this chapter is aimed at designing an IQ assessment and representation framework, based on which we can implement an IQ-based e-commerce system (discussed in chapter 4), which, in turn, can be used to evaluate the efficiency of this proposed framework and our approach. This chapter mainly delivers a conceptual architecture that applies Information Quality to facilitate e-commerce in getting better user satisfaction with search result. Users' requirement for data quality was taken into consideration in the architecture design.

#### **3.1.2 Methodology**

Information Quality is a systemic technology which has many facets. In order to apply it to a centralized buying-agent e-commerce system, that is, in order to measure and assess data quality, our model is intended to address the following key issues of e-commerce data quality:

- Defining and measuring e-commerce data quality
- Quantitative and qualitative metrics of e-commerce data quality
- Acceptable levels of quality
- Methodology for assessing e-commerce data quality.

Let's begin from the methodology. Normally, there are several things that our research needs to go through orderly, which are illustrated in the Figure 3-1:

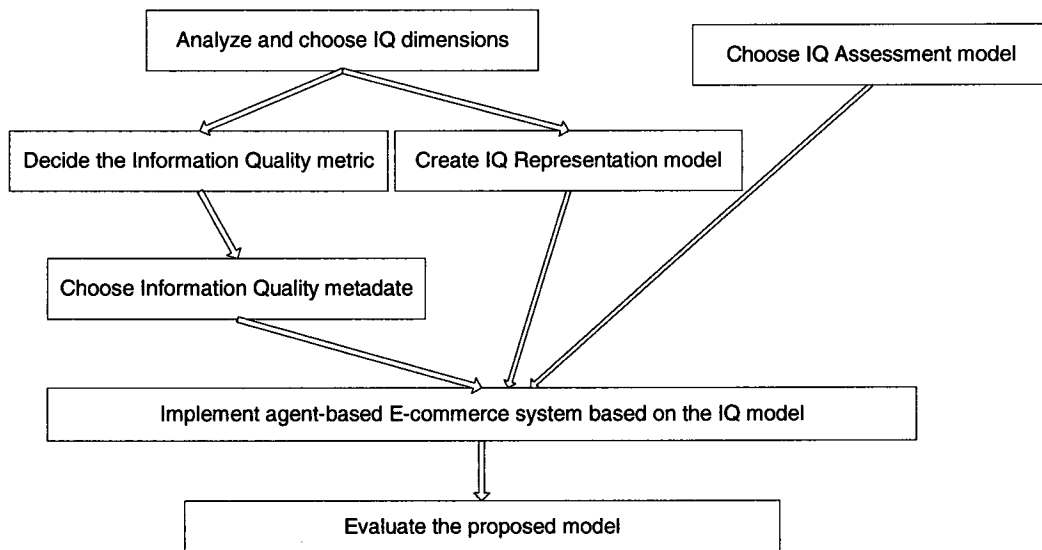


Figure 3-1: Research approach for the proposed model

First and perhaps the most important thing is to analyze and choose IQ dimensions according to the characteristics of e-commerce systems. As discussed in 2.1.2, since Information Quality is a multi-dimensions concept and, at the same time, data quality problems are application specific, there is no set of IQ dimensions which can be applied to all fields. The researcher must define dimensions, or the aspect of dimensions for the specific application. So, with respect to e-commerce, there is a set of suitable Information Quality dimensions

need to be explored. Theoretically speaking, this is a critical step to some extent, because it will significantly affect all the following steps on what dimensions and how many dimensions will be researched further. The dimensions should be chosen from several options (as illustrated in Table 2.1), which indicates that this task is challenging.

After determining IQ dimensions, it is needed to decide how to assess and measure each dimension. The development of data quality metrics in this study make use of three functional forms which Pipino and Lee [26] proposed (as described in 2.1.2 IQ Assessment): simple ratio, min or max operation, and weighted average. For each of the dimensions, the suitable functional form will be chosen.

Once the quality metrics are decided, quality metadata can be analyzed and identified. The objective of the step is to identify and classify metadata related to data quality from a data processing point of view.

After IQ dimensions are determined, the other thing to be done is creating an IQ representation model. On the basis of our analysis, Information Quality is a theoretical concept and the value of Information Quality is very abstract. Normally IQ value is presented as a list of dimensional data, (such as Completeness: 0.7, Reputation: 0.98, and Relevancy: 0.87), which is hard for end users/buyers to understand and to make decision on. Hence, it is necessary to present an IQ representation model to make the IQ result more understandable and communicable. In our study, an “index of data quality” is created to represent data quality of goods items which could help to succinctly communicate the state of data quality to buyers and provide comparative assessments over time.

At the same time, choosing a suitable Information Quality model should be conducted, that is, the researcher need to decide how to integrate the IQ algorithm into the normal query process. As described in 2.1.4, there are two possible models for consideration, namely the data-driven model and the quality-driven one. The model we build in this thesis is an enhancement of the data-driven model.

The next step will be the implementation of a buying-agent e-commerce system based on the Information Quality model. The system will be built on all the results of the above steps as well as the consideration of some special features; for instance, it allows users to define their preferences for IQ and then based on this to calculate the IQ of each goods items, and even filter the search results based on users' preferences..

Finally, the data can be imported into the system and some experiments can be conducted to evaluate the model. Further modifications for the model may be conducted based on the experimental results.

The research discussed in this chapter mainly focuses on how we choose IQ dimensions, IQ metrics, IQ metadata, and build IQ models in the context of centralized buying-agent e-commerce systems. We will discuss system implementation and evaluation in the next two chapters, respectively.

## **3.2 The Proposed IQ Dimensions**

In this section we will concentrate on the IQ dimensions which we chose for the given context. The reasons why the four IQ dimensions are chosen are discussed first; then each dimension is analyzed in detail.

### **3.2.1 IQ Dimensions Recognition**

As discussed above, there is no general agreement on data quality dimensions, and it is necessary to define the dimensions or the aspects of the dimensions according to a specific application. So, it is no exception to analyze and choose IQ dimensions firstly when we study the IQ of e-commerce systems.

According to Wang, et al. [37], IQ has been viewed as fitness for use by information consumers, with four IQ categories and fifteen dimensions identified, which are shown in Table 3.1. The intrinsic IQ captures the fact that information has quality in its own right. Accuracy is merely one of the four dimensions underlying this category. Contextual IQ highlights the requirement that Information Quality must be considered within the context of the task at hand. Representation and accessibility IQ emphasize the importance of the role of information systems [37].

<b>IQ Categories</b>	<b>IQ Dimensions</b>
Intrinsic IQ	Accuracy, Objectivity, Believability, Reputation
Accessibility IQ	Access, Security
Contextual IQ	Relevancy, Value-Added, Timeliness, Completeness, Amount of Data
Representational IQ	Interpretability, Ease of understanding, Concise representation, Consistent representation

Table 3.1: Information Quality categories and dimensions

The choices of the IQ dimensions are application related. It is closely tied to the problem domain, the data itself, and how it is used. It is necessary to define dimensions or the aspect of dimensions for a specific application. Precisely defining a dimension or the aspect of a dimension that relates to a specific application is often the most difficult task.

Based on Table 3.1 and combined with the analysis of the characteristic of E-commerce agent systems (described in the following paragraphs), we identify four potential data quality dimensions: Relevancy, Reputation, Timeliness, and Amount of Data. Comparing with Table 3.1, we choose Reputation dimension to assess with respect to the intrinsic IQ. The Relevancy, Timeliness, and Amount of Data dimensions are chosen to assess the Contextual IQ. Many other data quality characteristics are closely tied to these dimensions. For example, lack of relevancy of data leads to inaccuracy of the data available to users. Similarly,

inconsistency results from irrelevancy or incompleteness in data sources. Incompleteness to some extent means inappropriate amount of data.

The choice of these dimensions is primarily based on our understanding of the characteristics of E-commerce agent systems, industrial experience [11], and literature review [23]. Our concerns are listed as follows:

- The first and most important reason is that these four dimensions are tightly related with our research domain of e-commerce systems as they reflect the characteristics of the information in this area. Without Reputation and Timeliness, information on an e-commerce system will be useless and untrustable. The Relevancy of the goods information received by buying-agents will give buyers ground to make further decisions. The Amount of Data is another factor to make sure that the information of the goods which the buying agents retrieved is sufficient for the buyers to make decisions.
- The second reason is based on the availability of metadata. Actually, our first intuitive choice of dimension on Intrinsic IQ was accuracy, instead of Reputation. In the literature, information accuracy essentially means the degree to which information represents a real-world state. In the context of e-commerce, information accuracy implies the extent to which the contents of the product information provided by a selling agent represent the real product, which will be a good indicator for buyers to determine the degree to which the goods information is reliable. We thought that it could express the Information Quality more precisely and therefore would be an ideal dimension candidate for buyers. Unfortunately, after some careful analysis, we realized that the metadata that is needed for assess the accuracy is not sufficient in an agent based e-commerce system, since we cannot access the real goods of sellers. This would make it impossible for buying-agents to measure the accuracy dimension. Based on this consideration, we decided to use the Reputation dimension to give some indicator of accuracy.

- Another concern on our choice is the practicality of the metrics of the dimension in the context of an e-commerce system. This is applied to the completeness dimension, which is maybe the most frequently mentioned data quality dimension in the literature, and which we think will also be applicable to e-commerce systems.

Information completeness defines the degree to which all data relevant to an application domain have been recorded in an information system. In a centralized buying-agent e-commerce system, information completeness means that the information contents that a buying agent retrieved from the source are available when needed for users to make decisions in an effective manner. With regard to measurement, completeness is measured by the quotient of the number of non-null values in a source and the size of the universal relation [1], or by the ratio of the number of incomplete items to the total number of items and subtracting from 1 [26].

This value seems quite obvious and easy to retrieve. However, when studying the metadata in practical e-commerce systems, we realized that the value of this dimension is unnecessary to measure in the context since almost every value of the completeness dimension of goods items is 1. From our point of view, this is understandable: In order to get more successful transactions and make the system easy to use, system's designers would make use of some technologies (such as providing default values) to make sure that sellers fill out all the information required. This way, the information of items for sale will have no null values and the completeness values will be 1 in these e-commerce systems. So, for this research, we didn't choose this dimension. But we would like to point out that, for the research on distributed buying-agent e-commerce systems, this dimension may still be a good option since each subsystem have different fields designed and the required fields may be different too. Consequently, most of the values of the completeness dimension of goods items retrieved from these systems may not be 1, and the research on it will be interesting.

- The last reason is that these four dimensions are among the most wildly researched dimensions in the field [1, 2, 10, 11, 26]; they are also the most fully-fledged dimensions [1, 2, 26]. Also, their metrics are available so that we could apply them in our research with some modification on metadata or parameters. This will allow us to spend more time on other related fields of studies, such as, choosing metadata.

### **3.2.2 IQ Dimensions Analysis**

This section will give a close view of each dimension that we chose on its definition and measurement.

#### **3.2.2.1 Relevancy**

Relevancy is a dimension that is among the most discussed ones. Normally, information relevancy assures that the information content is pertinent to users' requirement. In the context of e-business systems, information relevancy indicates the extent to which the information of the goods presented by sellers meets buyers' need. From the perspective of buyers, it will be better if all the goods in the result set retrieved by buying agents meet their needs. However, in a real implemented e-commerce system, it is not always so satisfactory. For instance, in eBay if you input "Olympus sp-510" and try to get all the available items of this type of camera, you will get not only some items of this type of cameras, but also some other items as well, such as hand strap, tripod, macro lens etc., which all are accessories for cameras, and not what you want. Obviously, further process on the result set is needed. So, can we sort the result set and make camera items placed on the top of the list? Is there a way that we can measure the degree of the items matching the query keywords? This is the motivation that makes us decide to include the Relevancy dimension in this study. We look through all available quality dimensions, and decide that Relevancy dimension meets this requirement. We believe Relevancy could be a very important factor in retrieving goods items, and actually, our experimental result (discussed in Chapter 5) supports this hypothesis.

✓ **Definition**

In [30], Relevancy is defined as the extent to which the data is applicable and helpful for the task at hand. In the context of e-commerce systems, Relevancy is the extent to which the goods items, retrieved by buying agents and sent to buyers, are pertinent to the buyers' needs and interests.

✓ **Measurement**

There is no clear statement about the information Relevancy measurement in the literature. Based on the definition of Relevancy stated above, we take advantage of the frequency of query keywords in the description information of goods items to measure Relevancy. Query keywords express buyers' needs and interested; description information is the data in the goods items for sale. Hereby, the frequency of the query keywords in goods items can express the extent to which the goods items are pertinent to buyers' needs. Imagine, when a seller wants to sell an item, he/she would present the item information and mention the item name in its description; and at the same time, the item name is normally the query keyword used by buyers to retrieve the item. Therefore, the higher the frequency of the search keyword is found in the item information, the more closely the item is what the buyer wants to buy, and the higher Relevancy the item has. This consideration in turn motivates the following measurement for Relevancy:

$$\text{Relevancy} = \frac{\text{number of query keywords in description field}}{\text{total number of words in the goods item}} \quad (3.1)$$

With this definition, the value of Relevancy ranges between 0 and 1.

### 3.2.2.2 Reputation

As stated in [10], given the great importance that has in social and commercial relations, the study of Reputation has attracted the interest of researchers from different fields: sociology,

economics, and psychology. Computer science, particularly in the areas of multi-agent systems and online communities, is not an exception.

In an e-commerce agent system, we believe that the Reputation dimension deals with information believability issues geared toward giving users confidence to continue their actions. Reputation is an important dimension of Information Quality in the context of e-commerce systems because it will be a premise for buyers to make further decisions; that is, the buyer must have the information provided before he/she makes a decision. This is certainly understandable in the context of e-commerce systems in which buyers have transactions with sellers through the internet and their trust on the sellers as well as on the goods is a major concern for buyers and controls the success of the transactions. Low seller reputation to some extent means low priority of being chosen. Therefore, it is reasonable to take into consideration the Reputation dimension.

The other interesting point we notice is that, among most current research works on multi-attribute comparison, Reputation receives a lot of attention and is used as a very important attribute in rating sellers and goods. Hence, our method and the approach of multi-attribute comparison happen to have the same view on this point. This evidence gives us confidence in our choice.

✓ **Definition:**

Reputation, as described in [28], is the “opinion or view of one about something”. This opinion is formed and updated through direct interactions with an entity (e.g., a seller) or through the information provided by other members of the society about the experience they have had with that entity in the past. As a result of these interactions, individuals (e.g., buyers) record impressions that reflect how they value the experience. These individual impressions are the bricks that, if properly combined, can be used to build the Reputation of other entities.

✓ **Measurement:**

In electronic marketplaces, the reputation that a seller has is the result of aggregating all the impressions of the users who have interacted with him/her in the past. Amazon and eBay, for instance, are online auction houses where users buy and sell goods. Each time a new transaction is finished, the buyer rates the seller. These ratings are used to build the reputation of the seller. For centralized agent systems, Reputation is dealt with in a similar way. Normally, these ratings are displayed as the combination of two data parts: the percentage of the positive feedback and the total number of feedbacks (see Figure 3-2).

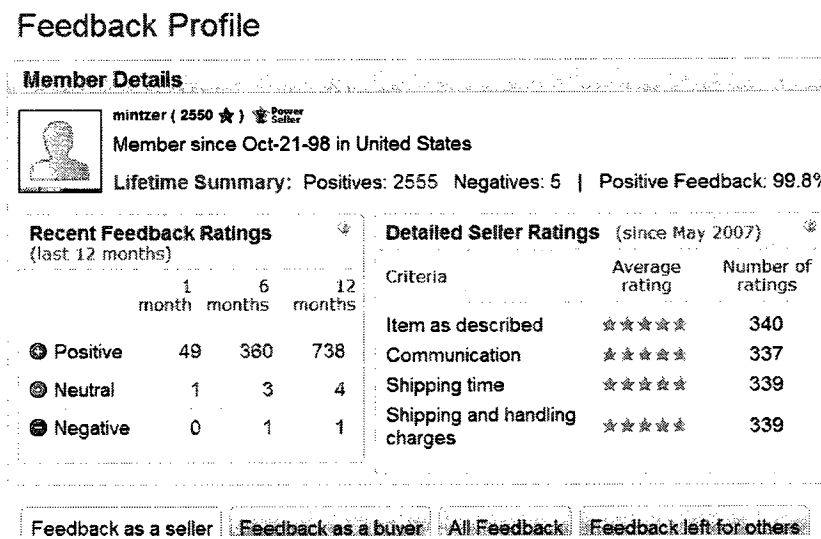


Figure 3-2: Reputation component in Ebay.com

The total number of feedback gives the buyer a sense of how many transactions the seller has conducted, and the percentage of positive feedback indicates to which extent the buyer is satisfied with the seller. Therefore in our research, the measurement on the value of Reputation dimension makes use of the product of these two rating values, which is the number of positive feedbacks for the seller.

We noticed that this value is an absolute value. When considering the functional forms of objective data quality metrics described in Section 2.1.2, we decide to modify the

equation to make the result value conform with [0, 1]. The new method is shown as followed:

$$\text{Reputation} = \frac{\text{seller's percentage of positive feedback} \times \text{number of feedbacks}}{\max(\text{seller's percentage of positive feedback} \times \text{number of feedbacks})} \quad (3.2)$$

Here the value in numerator (top of quotient) is the numer of this seller's positive feedback; the value in denominator (bottom of quotient) is the maximum value in numerator within the result set. Since Information Quality value will be used to sort goods items in the result set, so it makes sense to make use of the maximum value in the result set. Now the value of Reputation ranges between 0 and 1.

We also would like to point out that, in distributed e-business systems, since some subsystems may not have these two rating values, they can not always be retrieved directly, which means that some further calculations need to be done, for instance, we may need to retrieve other related data first in order to calculate these two rating values. Detailed analysis about this would be a topic for future works.

### **3.2.2.3 Timeliness:**

Time is a very important dimension of Information Quality in the context of e-commerce systems because buyers must continually have a sense of the time when the information was provided in order to make good purchase decisions. In our research, we recognize the Timeliness as the dimension to provide buyers with the needed time measurement. Timeliness is among the most frequently mentioned data quality dimensions in the literature. Before we discuss it further, it is necessary to introduce a related dimension: Freshness.

#### **✓ Related Dimension: Freshness**

Verónika and Mokrane [35] have done a lot of work on the research of freshness dimension. According to their paper “A Framework for Analysis of Data Freshness”, the concept of data freshness intuitively introduces the idea of how old the data is, e.g., whether it is fresh enough with respect to the user expectations, whether the given data source has the more recent data, If the extracted data is stale, and when data was produced etc.

Data freshness comprises a family of quality factors, each one representing some freshness aspect and having its own metrics. For that reason freshness is commonly mentioned as a quality dimension. From a user point of view, we distinguish two sub-dimensions of this quality dimension:

- Currency factor: It captures the gap between the extraction of data from the sources and its delivery to the users. For example, currency indicates how stale is the account balance presented to the user with respect to the real balance at the bank.
- Timeliness factor: It captures how often data changes or how often new data is created in a source. For example, Timeliness indicates how often the product prices change in a store or how often new books are added to a library.

In [35], the authors present some interesting points about Currency and Timeliness: On the one hand, when data changes very frequently, there is no interest in measuring timelines, which captures the stable data behavior. On the other hand, there is no sense to evaluate the currency of long-term and stable data because they are almost always current as they do not change very often. In the latter case, the system can assure currency even without explicit evaluation.

According to the discussion above, combined with the characteristics of the information in e-commerce systems where the information of goods items may change after the seller put it in the system, we think that the Timeliness dimension is an appropriate one for the research. Next, we will look at this dimension more closely.

**✓ Timeliness Definition**

Timeliness has been defined in terms of “whether the data is out of date and the availability of output on time” [26]. A closely related concept is currency, which is interpreted as the time a data item was stored [35]. In an e-commerce agent system, we believe that Timeliness dimension deals with information delivery issues geared toward giving users better control over the temporal aspects of their actions. Also, because information is frequently time-sensitive, this dimension also deals with providing temporal accuracy and current information to buyers.

**✓ Measurement**

According to the research of Pipino et al. [26], Timeliness is suggested as a function of the ratio of currency and validity; that is, the currency is good or bad depending on the validity. A large value of currency is unimportant if the useful-life is infinite. On the other hand, a small value of currency can be deleterious if the useful-life is very short. They also suggested that Timeliness is measured as the maximum of one of the two following terms: 0 and 1 minus the ratio of currency to volatility, which can be expressed as:

$$\text{Timeliness} = \max \left[ 0; \left( 1 - \frac{\text{Currency}}{\text{Volatility}} \right) \right]^s \quad (3.3)$$

Where Currency is defined as the age plus the delivery time minus the input time; Volatility refers to the length of time the data remains valid; delivery time refers to when data is delivered to the user; input time refers to when data is received by the system; and age refers to the age of the data when first received by the system.

In the research of Cinzia Cappiello et al. [2], Currency is defined as the time interval between the time instant in which data are updated and the time instant in which data is

used. Volatility refers to the length of time that data remains valid. This definition seems quite suitable to e-commerce systems. In our model, Currency indicates how recently the information of the product has been updated.

With this definition, the value of Timeliness ranges between 0 and 1. Conventionally, 1 represents the most desirable score and 0 represents the least desirable. That is, the more recently the information gets updated, the smaller Currency value will be, and the closer the value of Timeliness is to 1. Hence, higher Timeliness value reflects more fresh information.

The exponent in equation (3.3) is a sensitivity factor necessary to control the sensitivity of Timeliness to the currency-volatility ratio. According to [26], the value of the exponent  $s$  is task-dependent and reflects the analyst's judgment. For example, suppose the Timeliness rating without using the sensitivity factor (equivalent to a sensitivity factor of 1) is 0.81. Using a sensitivity factor of 2 would then yield a Timeliness rating of 0.66 (higher sensitivity factor reflects the fact that the data becomes less timely in a faster rate) and 0.9 when sensitivity factor is 0.5 (lower sensitivity factor reflects the fact that the data loses its Timeliness at a lower rate).

In our current research, for the sake of simplicity, the sensitivity factor  $s$  is set to 1. Further research to explore the impact of this exponent on the Timeliness dimension value could be a research topic for future work.

#### **3.2.2.4 Amount of Data**

By common sense any decision is made on an Amount of Data. There is no exception in e-commerce systems. From a buyer's perspective, it is insufficient if goods items provide very limited information (price, location, payment method etc.). If more information is provided, e.g., purchase time, working condition etc., then buyers could have a more comprehensive understanding of the situation of the goods, in order to make decisions. So, the assessment of

Information Quality should consider the Amount of Data, and accordingly we take it into consideration as one of the IQ dimensions. In the context of e-commerce, this dimension refers to the amount of the information on the goods items.

✓ **Definition**

According to [5], the dimension of Amount of Data indicates the extent to which the volume of the data is appropriate for the task at hand.

✓ **Measurement**

The calculation of this dimension is obvious, i.e., measuring the amount of the data. In our model, this is to measure the number of the words in the goods information provided by the seller:

$$\text{Amount of Data} = \text{number of words in goods information} \quad (3.4)$$

It is obvious that this is an absolute value and does not fall into the interval [0, 1]. Same as the approach used in the Reputation dimension, we change this metrics to:

$$\text{Amount of Data} = \frac{\text{number of words in goods information}}{\max(\text{number of words in goods information})} \quad (3.5)$$

where the maximum number of words of the goods items in the whole result set is used.

With this definition, the value now ranges between 0 and 1.

### 3.3 The Information Metrics

In the previous section, we analyze four dimensions and give a measurement method for each of them. Now let's look at the measurement method from the perspective of information

metrics. According to 2.1.2, simple ratio and min/max operation function are used to assess dimensions in this research. Both approaches are simply described as follows:

Simple ratio: In the model, the Relevancy, Reputation, and Amount of Data dimension take this form. Conventionally, 1 represents the most desirable score, and 0 represents the least desirable one.

Min or max operation: The minimum or maximum value is computed among the normalized values of the individual data quality indicators. These individual variables are measured by a simple ratio. The min operator assigns to the dimension an aggregate value no higher than the value of its weakest data quality indicator, while the maximum operator assigns a maximal value. As indicated above, the Timeliness dimension makes use of the max operator.

### **3.4 The Proposed Model**

This section presents our architecture for Information Quality assessment. A model for representation and assessment of Information Quality is also proposed.

#### **3.4.1 The Proposed Architecture for Information Quality Assessment**

This thesis is dedicated to apply IQ assessment in agent systems. Considering the characteristics of both Information Quality and agent systems in e-commerce (see 3.4.1.1 below for a detailed analysis), we proposed an integration architecture. Basically, our proposed model is a data-driven oriented model, which deals with buyers' queries using normal query technologies while additional information about the quality of the query result is processed. The model is composed of two main parts: a normal query-answering part and an IQ sector. The proposed architecture for IQ assessment in a generic agent system is depicted in Figure 3-3.

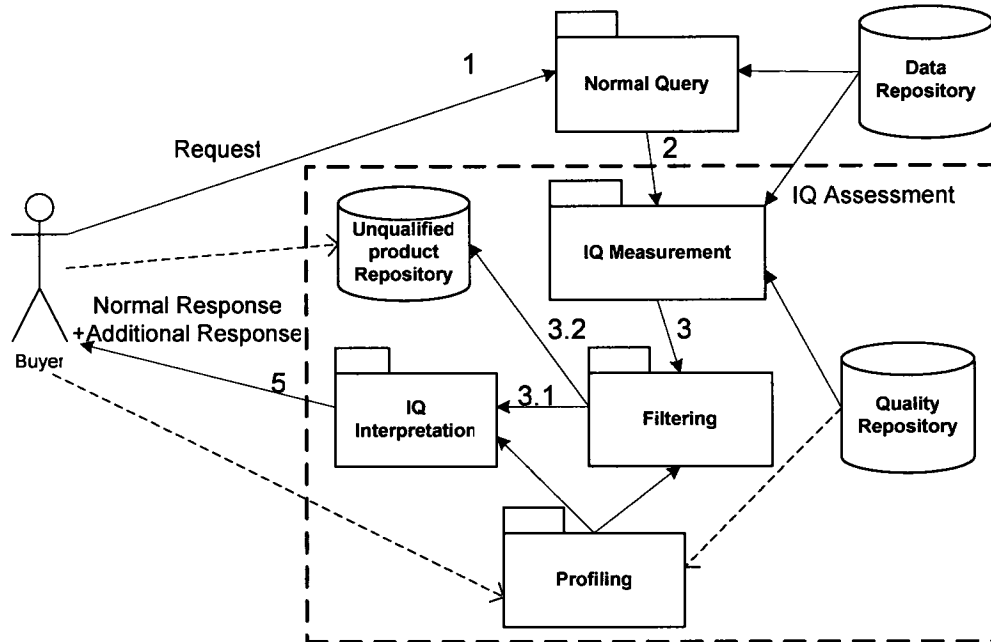


Figure 3-3: Architecture for IQ assessment in buyer agent

### 3.4.1.1 Design Decisions Made in Our Architecture

The reason why we choose to build the new architecture on the basis of the data-driven model is that we want to keep the proposed architecture apart from the other parts of an agent-based e-commerce system. We just want to provide the agent system with an alternative method for retrieving goods items and don't want to interfere with the other original operations of the agent system. Next, we will explain why we added two modules as our modification to the traditional data-driven model.

One added module is the filtering module. The explosive growth of data on the Web makes information management and the discovery of knowledge increasingly difficult. In e-commerce, enormous products are provided and exist in the data repository for retrieval, and it is an onerous burden for a buyer to analyze every product in the result sets and choose the suitable one, when facing the overwhelming response information after sending only one request. As far as our model is concerned, it would be extremely helpful for a buyer to make a decision if the agent system can search and limit the number of response results according

to some criteria, such as buyers' IQ requirement. Based on this consideration, the filtering module is implemented in the model.

Normally, the same data can be judged as being satisfactory to one user or poor to another user. It is hard for users to make decision based on the results received from the IQ measurement, which produces raw and unweighted quality metadata. For instance, is the quality of information with 0.5 Relevancy, 0.8 Reputation, and 0.6 Timeliness better than the one with 0.6 Relevancy, 0.9 Reputation, and 0.4 Timeliness? This is the reason why we added and implemented the IQ interpretation module.

As stated in [5], IQ interpretation uses the measurement output in order to perform reasoning on IQ using additional information such as users' input and IQ requirements. Quality interpretation is performed after assessment. As showed in Figure 3-3, the quality scores obtained by quality measurement steps can be used as input for quality interpretation. In the interpretation module, by taking users' requirements into consideration, the system makes use of an "index of data quality" to calculate the quality value. The "index of data quality" is first mentioned by Pipino et al. in their research on IQ dimensions as a single aggregate measure of data quality [26]. In our model it is combined with the buyer's expectation values, which are set by buyers for each IQ dimension, and stored in the profiling module. From our point of view, such an index could help to succinctly communicate the state of data quality to buyers and provide comparative assessment over time. The IQ interpretation module is used to implement the functionality of calculating the "index of data quality" for each goods item in search result sets. We shall revisit the "index of data quality" and present an equation by which our agent system calculates this index in Section 3.4.2 below. Let us now describe the component modules in our proposed architecture first.

### **3.4.1.2 The Proposed Architecture**

As shown in Figure 3-3, our proposed architecture consists of five main modules: the normal query module, the quality measurement module, the filtering module, the IQ interpretation

module, and the profiling module. The last four modules act as the IQ sector of the system, which is structured as a sequence of steps in calculating the data quality measures and assessing the data quality. This sector works on the result set of the normal query module, calculates the IQ of good items, and then sends the result (including goods data and their IQ values) to buyers. The first step in assessing the quality of data is to determine the Information Quality dimensions of the application. The other steps include methods for calculating each data quality measure to allow quantitative assessment of quality, establishing acceptable quality targets, and reporting the data quality.

Next we will discuss these five modules, respectively.

### **Normal query module**

Buyers' request is first sent to this module. The module analyzes buyers' requests and then queries the data repository. At the end of the execution of this module, the query result is sent to the quality assessment module. These query result will be a list of goods items which satisfied the query criteria.

To understand the framework better, let us consider a typical scenario taken from daily life, where a buyer, named Frank, try to buy a Konica camera from the system. In this phase, he input the "Konica camera" as search keywords; the module will query the database and retrieve all the goods items which match this criteria. Suppose the system shows that there is 264 items found for "Konica camera". All these items as well as their detail information are put into the result set and then sent to the next module.

### **Quality measurement module**

This module is the core module of the whole architecture, which is responsible for the retrieval of the requested data and the necessary metadata, and then performs the evaluation process based on these data.

Within the module, there are several steps that are conducted in order:

- Firstly, the module retrieves the quality metadata from the Quality Repository for the queried goods.

In the proposed architecture there are two repositories: data repository and quality repository. Data repository is the place where all information of goods items are persisted, while quality repository only stores Information Quality related metadata (see Section 2.1.4). Currently, it include: Quality dimensions and their relation with the data schema of goods items in the data repository. For example, the Timeliness dimension is related to the fields of created time and modified time of goods items; the Reputation dimension is related to the fields that store sellers' reputation information.

- Secondly, the module measures four data quality dimensions one by one, namely Relevancy, Reputation, Timeliness, and Amount of Data. For each of dimension, the module retrieves the necessary data and quality metadata, and evaluates it. The evaluation function is defined by establishing the metric, which make use of the functional forms described in Section 2.1.2. These functions will be further discussed in Chapter 4.
- Finally, at the end of the execution of the measurement procedure, the module sends the requested data, along with the results of the quality evaluation, which is a set of values for the Information Quality dimensions, to the filtering module.

Let's continue with the example above. At this stage, every dimension of each product items in the result set is measured. That is, for each item, the module has calculated the four dimension values (i.e., Relevancy, Reputation, Timeliness, and Amount of Data). Accompanied by these quality values, the result set will be sent to the filtering module for further processing.

**Profiling module:**

Before introducing the filtering and IQ interpretation module, we would like to discuss the profiling module, since this module specifies user's requirements for Information Quality, which are used by the next two modules.

Regarding to the quality of the information system and the information it provides, the perspective of an information provider may be different from that of an information consumer. However, ultimately it is the information consumers who will judge whether or not the information is suitable for their usage. Therefore, the quality of the information cannot be assessed independent of the data consumers who use the information [30].

From an information management perspective, human factors represent the characteristics of buyer requirements [30]. The same product information can be perceived in different ways and be judged as satisfying by a buyer or poor by another buyer. The model proposed here can support buyers in this assessment.

In our proposed model, buyers' requirements for Information Quality are set in the profiling module. This module contains the specification of buyers' Subjective Quality Level that they chose for each data quality dimension, namely Relevancy, Reputation, Timeliness, and Amount of Data, denoted by  $W_r$ ,  $W_p$ ,  $W_t$ , and  $W_a$ , respectively. These values are obtained based on the understanding and requirements of buyers for the IQ dimensions. In the filtering module these values serve as the thresholds to filter the product items in the result set. In the IQ Interpretation module they are used as Information Quality weight to calculate the "index of data quality" for each product item. If the data quality dimensions are well understood by buyers and are assigned values accordingly, from our point of view, then the filtering result and the later result set of quality index could help buyers to interpret the data quality status and make good purchase decision.

Let's continue with the "Konica camera" example above. Before sending the request for the camera, Frank is prompted to specify his Subjective Quality Level for IQ. Suppose he specify  $\{0.6, 0.5, 0.4, 0.5\}$  as his personal requirements on limits of acceptability for each quality

dimension. That is, his requirement for product Relevancy is 0.6 (60%), for seller's Reputation is 0.5, for information Timeliness is 0.4, and for Amount of Data of product information is 0.5. These values will be used in the next two modules for different roles.

### **Filtering module:**

As discussed above, the objective of this module is to limit the size of the result set. There are three steps involved in the process of filtering:

- Firstly, the module retrieves the Subjective Quality Level values of the buyer from the profiling module.
- Secondly, the module filters the product items according to their IQ values and the buyer's Subjective Quality Level values.
- Thirdly, the module sends the filtered results to two different places: the selected qualified product information is sent to the IQ Interpretation module (step 3.1 in Figure 3-3), while the information with poor quality is labelled with an alert sign and is put in the Unqualified Product Repository (step 3.2 in Figure 3-3). This unqualified information will not be dealt with further, although the buyer can still access them if they would like to. Here, the quality is considered poor when the buyer's requirements are not satisfied; that is, when the data quality values are below the Subjective Quality Level values.

Let's demonstrate this using the example mentioned before. In this module, each item in the result set will be filtered using the Subjective Quality Level values of {0.6, 0.5, 0.4, 0.5}. That is, those items, whose information Relevancy is greater than or equal to 0.6, whose seller's Reputation is greater than or equal to 0.5, whose information Timeliness is greater than or equal to 0.4, and whose Amount of Data is greater than or equal to 0.5, will be treated as qualified items and sent to the IQ Interpretation module. The other items, which do not meet the buyer's requirements, will be stored in the Unqualified Product Repository. So, after this step, the size of the result set may be changed to 168 items from the original 264 items.

### **IQ Interpretation module:**

In this module, the “index of data quality” is developed to deal with the information of qualified goods items and presented to the buyer. As discussed in detail in Section 3.4.2 below, an index of data quality is a single aggregate measure of Information Quality, calculated using the average function of dimension values (see equation (3.6) in Subsection 3.4.2).

Within the module, two things are conducted in turn:

- Firstly, the module retrieves the Subjective Quality Level values of the buyer from the profiling module.
- Secondly, making use of the equation (3.6) and taking the buyer’s Subjective Quality Level values as quality weights, the module calculates the index of data quality for each item in the result set. These values will give the buyer some indication on their Information Quality level.
- At the end of the execution, the module sends the goods items along with the values of their quality index to the buyer. The buyer then can make purchase decision based on the information provided.

Now, let’s finish our example. After getting 168 items from the filtering module, the module use the equation (3.6) and the buyer’s Subjective Quality Level values to calculate the quality index for each item. For example, if an item has Relevancy dimension value of 0.8, Reputation value of 0.95, Timeliness value of 0.86, and Amount of Data value of 0.7, then the index of data quality for this item is 0.8325. This quality index as well as the item’s product information will be sent and presented to user Frank.

### **3.4.2 The Proposed Model for IQ Representation and Assessment**

At the research of Cappillo et al. [2], they present a generic model that ties the assessment phase to users’ requirements, and also combines user class (class of users with similar characteristics) and services (services are the functionality that is offered) into the assessment

procedure. Our proposed model is built based on this model, where users are internet buyers and services are merchant broking service provided by buying agents.

In this section we present our proposed model by which Information Quality is assessed and represented.

Consider the scenario where a buyer  $B_m$  announces its request for a product  $P_i$ . Let  $B$  be the set of buyers,  $P$  be the set of products. After the normal query for  $P_i$ , a result set of goods,  $G$ , is generated. In this result set, every goods item  $G_k$  is product  $P_i$ .  $B$ ,  $P$ , and  $G$  are finite sets. In the system level, the association among buyers, products, and goods can formally be represented as follows:

$$B = \{ B_1, B_2, \dots, B_m, \dots, B_n \}$$

$$P = \{ P_1, P_2, \dots, P_i, \dots, P_s \}$$

$$G = \{ G_1, G_2, \dots, G_t \} \quad \forall G_k \in G : G_k \in P_i$$

In the result set  $G$ , each of goods item  $G_k$  has four Information Quality dimensions, which are calculated in the quality measurement module. Let  $Q_k$  be the Information Quality of goods item  $G_k$ , and  $Q_{rk}$ ,  $Q_{pk}$ ,  $Q_{tk}$ ,  $Q_{ak}$  be the data quality dimensions value of Relevancy, Reputation, Timeliness, and Amount of Data, respectively. Then, the Information Quality dimensions of goods item  $G_k$  can be presented as follows:

$$Q_k = \{ Q_{rk}, Q_{pk}, Q_{tk}, Q_{ak} \} \quad \text{with } Q_{rk}, Q_{pk}, Q_{tk}, Q_{ak} \in [0,1]$$

$$\text{Where: } Q_{rk} = f_{rk} (G_k)$$

$$Q_{pk} = f_{pk} (G_k)$$

$$Q_{tk} = f_{tk} (G_k)$$

$$Q_{ak} = f_{ak} (G_k)$$

Where:

- $G_k$  is the data source from which information is extracted

- $f_{rk}$  ,  $f_{pk}$  ,  $f_{tk}$  , and  $f_{ak}$  are the measurement functions for Relevancy, Reputation , Timeliness , and Amount of Data dimension, which have been discussed in Section 3.2.2 and the detailed implementation of these formulas will be discussed in Chapter 4.

Then the Information Quality of the whole result set G can be represented as:

$$Q(P_i) = \begin{bmatrix} Q_{r1} & Q_{p1} & Q_{t1} & Q_{a1} \\ Q_{r2} & Q_{p2} & Q_{t2} & Q_{a2} \\ \dots & \dots & \dots & \dots \\ Q_{rt} & Q_{pt} & Q_{tt} & Q_{at} \end{bmatrix}$$

It is obvious that this set of Information Quality is difficult for buyers to understand and compare regarding to IQ of goods items. As we discussed in Section 3.4.1.1, we introduce the concept of “index of data quality” in interpretation model to address the issue, that is, we use an index to represent the data quality of goods items. Followed is how buying agents calculate the index.

Before sending a request, in the profiling module, the buyer  $B_m$  assigns his/her Subjective Quality Level values for the product  $P_i$ , we name it  $W$ , which has four values:  $W_{mir}$  ,  $W_{mip}$  ,  $W_{mit}$  , and  $W_{mia}$  for the four IQ dimensions, separately. So the buyer’s constraint for quality dimension can be represented as:

$$W = \{ W_{mir} , W_{mip} , W_{mit} , W_{mia} \} \quad \text{where } W_{mir} , W_{mip} , W_{mit} , W_{mia} \in [0,1]$$

As mentioned before, these values are used as weights to calculate the quality index. Let  $IQ_{mik}$  be the quality index of the goods item  $G_k$  (of product  $P_i$ ) when considering buyer  $B_m$  requirements. Then, the quality index is calculated as follows:

$$IQ_{mik} = \frac{Q_{rk} \times W_{mir} + Q_{pk} \times W_{mip} + Q_{tk} \times W_{mit} + Q_{ak} \times W_{mia}}{W_{mir} + W_{mia} + W_{mip} + W_{mit}} \quad (3.6)$$

Where

- $Q_{rk}$  ,  $Q_{pk}$  ,  $Q_{tk}$  ,  $Q_{ak}$  is the Information Quality values for the Relevancy, Reputation, Timeliness, and Amount of Data dimensions of the goods item  $G_k$  ;

$$- Q_{rk}, Q_{pk}, Q_{tk}, Q_{ak} \in [0,1], \quad W_{mir}, W_{mip}, W_{mit}, W_{mia} \in [0,1], \quad IQ_{mik} \in [0,1].$$

### 3.5 Summary

This chapter presents the process through which we conducted our IQ study in e-commerce agent system. At the beginning, we discussed the study path which we need to go through when conducting the research. Then, abiding by this study path, we implemented our research, which include reorganizing IQ dimension, analyzing information matrices, proposing IQ assessment architecture, and creating IQ representation model. Now, based on these researches, we are able to apply Information Quality in buyer agent for goods searching and ranking. Next chapter, we will present an implementation of the framework that we proposed here.

## **Chapter 4 System Implementation**

As stated in Section 1.4 – Thesis Objectives, one of our goals is to implement an e-commerce system, which can perform Information Quality assessment on goods items, in order to evaluate our proposed approach. In Chapter 3 we have proposed a conceptual architecture. This chapter presents the design and implementation of a centralized buying agent system called AQUAES (Automatic (Information) Quality Assessment for e-Commerce Systems), which is built on the basis of the proposed Information Quality architecture discussed in Chapter 3 (and depicted in Figure 3-3).

### **4.1 System Architecture**

#### **4.1.1 Introduction**

Fundamentally, AQUAES is a simplified buying-agent system that helps buyers to look for goods items and provides comparison/ranking on the result set. According to the CBB model [6], this system implements the functionality of buying-agents in the merchant brokering stage. It is a centralized-buying agent system since all the data is persisted in local databases.

#### **4.1.2 Implementation Environment**

Figure 4-1 depicts the topology of the elements that comprise the implemented system. Two servers are deployed in the local area networks (LAN, IEEE 802.3 Ethernet). Server 1 runs an Apache Tomcat Servlet Engine and serves as mediator server. Server 2 installs a PostgreSQL database and serves as database server. Client PCs use web browser to communicate with Server 1 through the internet over HTTP protocol. So, the development is

divided into two parts, the client side and server side applications. The client side provides user interface support for the web-based users and the server side offers services.

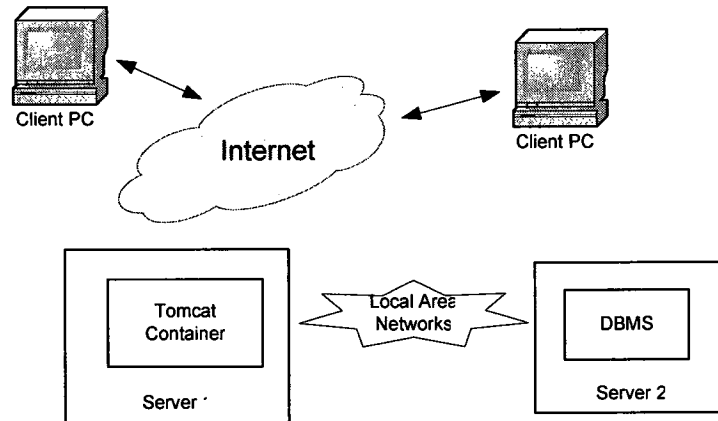


Figure 4-1: Topology of AQUAES

### 4.1.3 Software Architecture

The object-oriented programming language Java was used for the programming and its latest version, Java 2 standard Edition Development kit 1.5, was adopted. The advantage of using Java for the development is that it has great support for web application development.

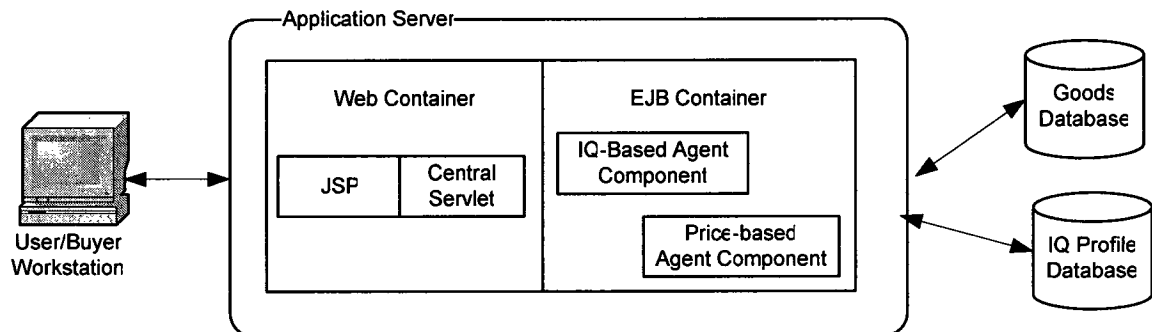


Figure 4-2: System Architecture of AQUAES

As shown in Figure 4-2, AQUAES is implemented in Java EE<sup>3</sup> platform. The JSP<sup>4</sup> pages serve as user interface where users could type their queries and review a list of search results of goods items. A central JAVA Servlet<sup>5</sup> is called to deal with the interface requirements. In the EJB<sup>6</sup> container, the IQ-based Agent component and the Price-based component work as core parts to analyze the goods information using IQ-based comparison or price-based comparison approaches, respectively.

The Tomcat 5.0 Servlet container was used as the servlet container in the official Reference Implementation for the Java Servlets and JavaServer Pages technologies. A servlet container is a runtime shell that manages and invokes servlets on behalf of users.

#### 4.1.4 Architecture Considerations

Since one main purpose of building AQUAES is to evaluate our proposed IQ-based ranking approach, a reference substance approach is needed. Therefore, a price-based ranking approach is chosen and implemented in AQUAES to serve as baseline.

There are three reasons that make us choose the price-based ranking:

(1) The first reason for choosing price comparison is in that it's an important approach in buying agent research. As discussed in Section 2.2, this technique is widely used in buying agents. It is the earliest method used by researchers and also has long history of being implemented in e-commerce systems. Nowadays, almost every practical e-commerce system provides the price ranking for buyers, e.g., eBay, Amazon, etc.

---

<sup>3</sup> Java EE stands for Java Platform Enterprise Edition. Java EE is a platform-independent, Java-centric environment from Sun for developing, building and deploying Web-based enterprise applications online.(<http://www.webopedia.com/TERM/J/J2EE.html>)

<sup>4</sup> JavaServer Pages (JSP) is a Java technology that allows software developers to dynamically generate HTML, XML or other types of documents in response to a Web client request. The technology allows Java code and certain pre-defined actions to be embedded into static content.( [http://en.wikipedia.org/wiki/JavaServer\\_Pages](http://en.wikipedia.org/wiki/JavaServer_Pages))

<sup>5</sup> The Java Servlet API allows a software developer to add dynamic content to a Web server using the Java platform ([http://en.wikipedia.org/wiki/Java\\_servlet](http://en.wikipedia.org/wiki/Java_servlet))

<sup>6</sup> EJB is short for Enterprise JavaBeans™, which is a managed, server-side component architecture for modular construction of enterprise applications (<http://en.wikipedia.org/wiki/EJB>)

(2) The second reason for choosing price comparison is in that it is widely accepted by users of e-commerce. From the buyer's perspective, price is a very important factor for making purchase decision. So, this approach has been widely accepted and understood by buyers.

(3) The last reason is that it is easy to implement, which will allow us to spend more time and to concentrate in our research on the Information Quality based ranking approach.

Therefore, the following three basic strategies were implemented in our AQUAES system:

- **Original ranking:** Original ranking approach means that the result set is received from normal query, and does not get further processed, but is sent and presented to the buyer at once. Normally, the order of goods items in this ranking is the record of the order in which those goods items are persisted in the database.
- **Price only ranking:** Once a buyer chooses this approach, AQUAES will compare the prices of goods items in the result set, and rank the goods item according to their prices. There are two ways to implement this: lowest price first or highest price first.
- **IQ only ranking:** If a buyer chooses this approach, AQUAES will calculate the Information Quality values for each goods item in the result set and then sort and present the goods items to buyers according to their IQ values. In this thesis, a greater value of the Index of Information Quality (calculated by Formula (3.6)) indicates a higher Information Quality of the goods item. So AQUAES will list goods items in a descending order of their Information Quality values.

## 4.2 System Implementation

As stated in 4.1.4 AQUAES is composed of three ways of ranking. Figure 4-3 shows the AQUAES system from the perspective of the user.

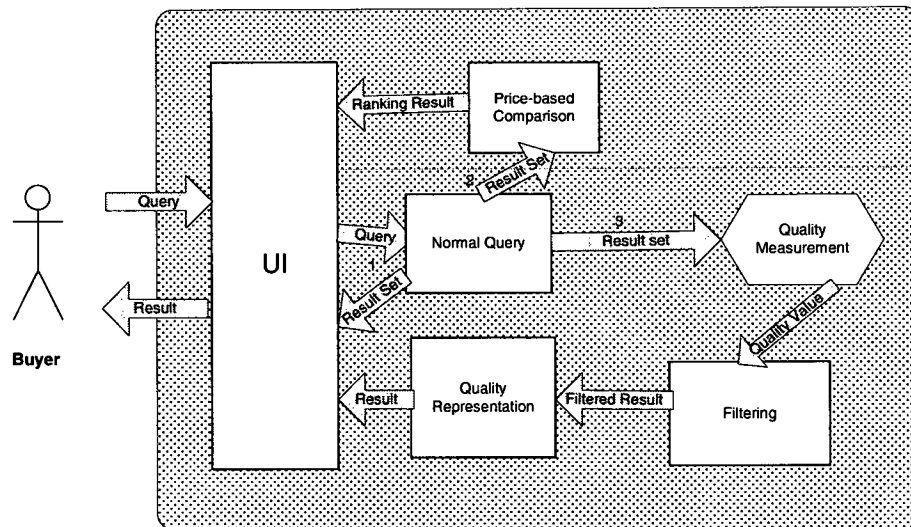


Figure 4-3: System view of AQUAES

The User Interface (UI) module deals with buyers' requests and presents result sets to buyers. Figure 4-4 shows the user interface of the AQUAES system where buyers can input query words and express their requirements.

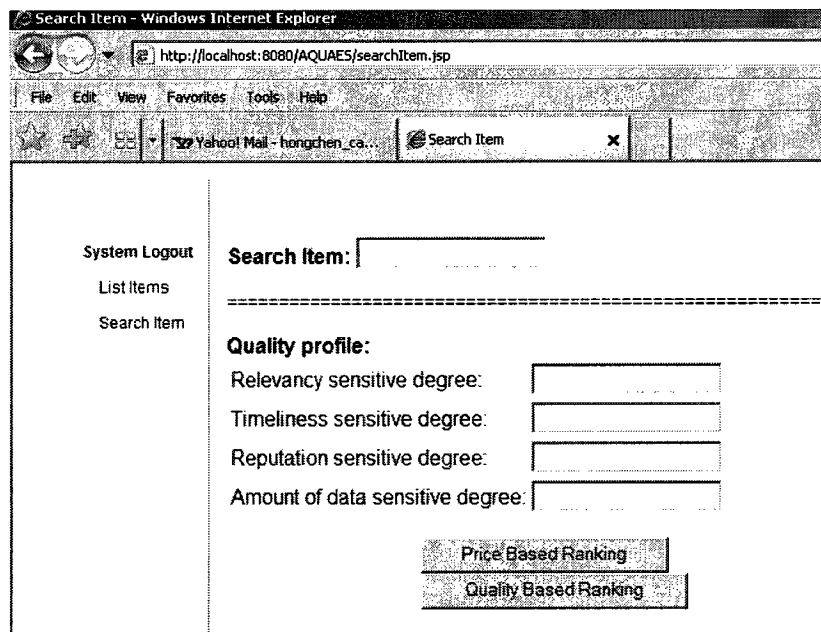


Figure 4-4: AQUAES User Interface for buyers to input query



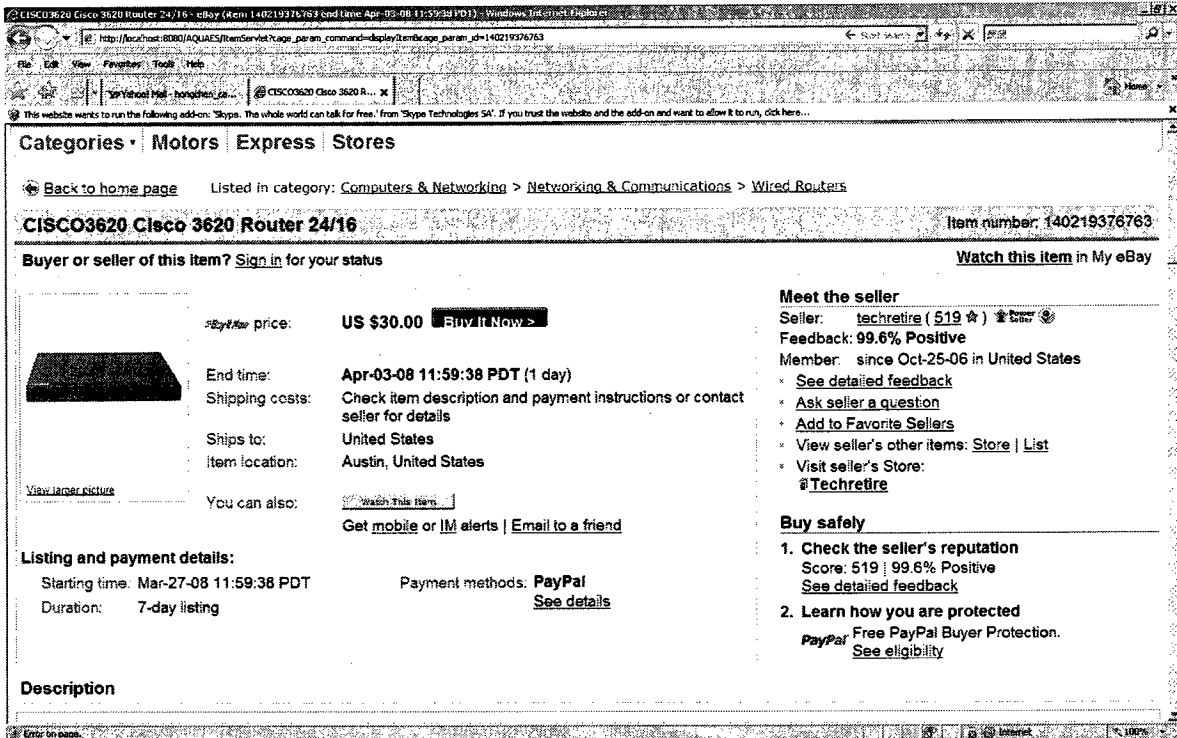


Figure 4-6: AQUAES Goods Item Example

## 4.2.2 Price-Based Ranking:

The second way of processing the result set is to transport it into the Price-based Comparison module (indicated by arrow 2 in Figure 4-3). The goods items in the set will be ordered by their prices and then presented to the buyers as shown in Figure 4-7

Name	SellerId	PostDate	Location	Price
1. Cisco post item for 3509 E001 3509-3140 3520 11.1119-3A	lgheup06	Apr-05-06 22:09:40 PDT	Lexington, Kentucky, United States	0.99
2. 32 MB memory kit for Cisco 3500 3520 router MEM3500-19D	lgheup06	Apr-05-06 06:45:47 PDT	Lexington, Kentucky, United States	0.99
3. 32 MB memory kit for Cisco 3500 3520 router MEM3500-32D	lgheup06	Apr-05-06 06:21:03 PDT	Lexington, Kentucky, United States	0.99
4. Cisco router 3520 with 2E2W A 1E2W	all_cisco	Apr-05-06 09:25:31 PDT	San Diego, California, United States	0.99
5. CISCO 3520 Router 3500 Series (Serial Condition)	sanitcan	Apr-05-06 18:07:50 PDT	Norcross, GA, United States	0.99
6. CISCO 3520 Router 3500 Series (Serial Condition)	canitcan	Apr-03-06 16:07:13 PDT	Norcross, GA, United States	0.99
7. CISCO 3520 Router 3500 Series (Serial Condition)	sanitcan	Apr-04-06 07:15:10 PDT	Norcross, GA, United States	0.99
8. Cisco 3520 Series Integrated Services Router - Used	sgprod_inc	Apr-02-06 13:45:21 PDT	Melbourne, FL, United States	0.99
9. Cisco 3520 Replacement Front Panel Light Bussing	cablesandits	Apr-03-06 17:31:02 PDT	Atlanta, GA, United States	3.99
10. Cisco 3520 Replacement Front Panel Chassis Light	cablesandits	Apr-03-06 17:30:57 PDT	Atlanta, GA, United States	9.95
11. Cisco 3520 Series Router 3520	shopmic	Apr-03-06 14:23:04 PDT	Columbus, United States	9.99
12. DC Terminal Block For Cisco 3520 3540 3560 Power Supply	cablesandits	Apr-03-06 15:00:47 PDT	Atlanta, GA, United States	11.99
13. Cisco 3520 Router Replacement Chassis Fan	cablesandits	Apr-04-06 15:20:22 PDT	Atlanta, GA, United States	14.99
14. Cisco NIA-1E2W Module for 3500 3520 3540 Series Routers	candelanet	Apr-03-06 09:20:22 PDT	Brooklyn, New York, United States	14.99
15. Cisco 3520 AC power supply PWR-3520-AC	gpmcayun01	Apr-03-06 17:19:11 PDT	Lancaster, California, United States	17.0
16. Cisco 3520 19" Rack Mount Kit	cablesandits	Apr-04-06 15:40:21 PDT	Atlanta, GA, United States	19.99
17. Cisco 35-0889-01 PWR 3520-AC AC Power Supply for 3520	psd_etc	Apr-05-06 10:02:32 PDT	Phoenix, AZ, United States	19.99
18. Cisco NIA-1E1R2W Network Module for 3520 3540 3560	mtiadsupply	Apr-03-06 15:15:00 PDT	New York, NY, United States	20.0
19. Cisco 3520 AC Power Supply PWR-3520-AC	cablesandits	Apr-03-06 14:59:24 PDT	Atlanta, GA, United States	24.99
20. Cisco 3520 CNA Lab Router 3520 w/5A RAM	shopmic	Apr-03-06 14:39:17 PDT	Columbus, United States	24.99
21. Cisco 3520 Series 2 Slot 4000 Router 3520	evcsfsw	Apr-03-06 18:38:36 PDT	San Jose, CA, United States	24.99
22. MEM3520-16D 1.41-B DRAM Kit for Cisco 3500 3520 ASB-QTY	metusa	Apr-03-06 14:00:00 PDT	Brooklyn, New York, United States	24.99
23. CISCO 3520 LAB ROUTER CNA, COMP CODE NON DCA 3540 2990	mlntbr	Apr-03-06 17:51:31 PDT	Rochester, NY, United States	24.99
24. CISCO 3520 LAB ROUTER CNA, COMP CODE NON DCA 3540 2990	mlntbr	Apr-03-06 18:28:54 PDT	Rochester, NY, United States	24.99
25. Cisco 3520 43" 24" Rack Mount Kit	cablesandits	Apr-03-06 13:26:55 PDT	Atlanta, GA, United States	29.99
26. 18MB Flash MEM3500-16B for Cisco 3500 3520 3540 ASB-QTY	metusa	Apr-03-06 14:00:00 PDT	Brooklyn, New York, United States	29.99
27. CISCO 3520 Cisco 3520 Router 2410	teckrate	Apr-03-06 11:59:36 PDT	Austin, United States	30.0
28. CISCO 1M-2E2W 2E2W 3520 3540 - WARRANTY GG	mindstechnologies	Apr-05-06 11:20:31 PDT	Oklahoma City, OK, United States	32.0
29. CISCO 1M-2E2W 2E2W 3520 3540 - WARRANTY GG	mindstechnologies	Apr-05-06 11:21:26 PDT	Oklahoma City, OK, United States	32.0
30. Cisco NIA-1E2W Module for 3500 3520 3540 routers Genuine	router-certs	Apr-02-06 21:35:42 PDT	NJNY, United States	39.99
31. Cisco 3520 AC Power Supply New PWR 3520-AC	cablesandits	Apr-03-06 13:29:54 PDT	Atlanta, GA, United States	44.99
32. Cisco NIA-4A1S Module for 2600 2900VNI 3820 3140 3651 3860	candelanet	Apr-04-06 06:20:19 PDT	Brooklyn, New York, United States	47.0
33. Cisco 3520 Router 3520	tarlis	Apr-03-06 05:54:12 PDT	Taunton, Massachusetts, United States	49.99

Figure 4-7: Example of AQUAES price-based ranking result

### 4.2.3 IQ-Based Ranking:

The third way of processing the result set is to send it to the Quality Measurement module (indicated by arrow 3 in Figure 4-3) where Information Quality assessment is conducted on the goods items. The Quality Measurement module calculates the quality values for each of the IQ dimensions. The items then go through the Filtering and the IQ Interpretation module and are presented to the buyer in the order of the values of their Index of Information Quality (see Figure 4-8). At this point, AQUAES contains an IQ ranking of the result set of the goods items.



The quality values calculated using these formulas are denoted as  $Q_{rk}$ ,  $Q_{pk}$ ,  $Q_{tk}$ ,  $Q_{ak} \in [0,1]$ , as described in Section 3.4.2, where we presented our proposed model for Information Quality representation and assessment.

In order to be able to calculate IQ dimensions of goods items using the four equations stated above, we define 12 fields required for each of goods items in our dataset, as shown in Table 4.1.

<b>Attribute</b>	<b>Description</b>
Title	Indicates the name of the goods on sale
Category	Indicates the category of the goods on sale, which often has several levels
Postdate	Indicates the date that seller post the goods item in the system
Price	Indicator the goods price that the seller predefined
Description	Describes information of goods on sale
Seller Id	Indicates a seller's Id
Feedback Number	Indicates number of feedback a seller got
Percentage	Describes the positive feedback percentage of a seller got
Location	Indicates the location of goods
Payment method	Specifies some payment methods that a seller would like to accepted
Shipping destination	Indicates some pre-defined destinations that a seller prefer to send goods to
Shipping cost	Indicates the shipping cost that is paid by buyers

Table 4.1: Attributes required for Goods Items in AQUAES

Among these fields Title and Category are used for search and display purpose; Postdate is used for measuring Timeliness; Price is used for price-based ranking; Description is used for

measuring Relevancy; Seller Id, Feedback Number, and Percentage are used for measuring Reputation; all the information in these 12 fields are used for measuring Amount of Data.

After receiving the IQ value for each dimension, AQUAES then uses equation (3.6) to calculate the Index of Information Quality for each goods item and ranking them accordingly (see Figure 4-8).

#### 4.2.3.2 Database Design

Currently, the database is designed to have two tables and a view (see Figure 4-9) in order to meet the requirement of the system for storing and displaying data.

The table “Goods Item” stores all the goods information that is posted by sellers and will be searched and retrieved by buying agents. The table “Seller Info” stores all the sellers’ information registered with the system. The fields needed to implement the IQ-based ranking (described in Table 4.1) are all included in these two tables. The table “Goods Item” can meet the requirement of price-based ranking. Therefore, these two tables are sufficient for the current requirements of the system.

The view “Goods\_IQ\_View” is designed to temporarily store the result of the IQ-based ranking, which will finally be presented to buyers.

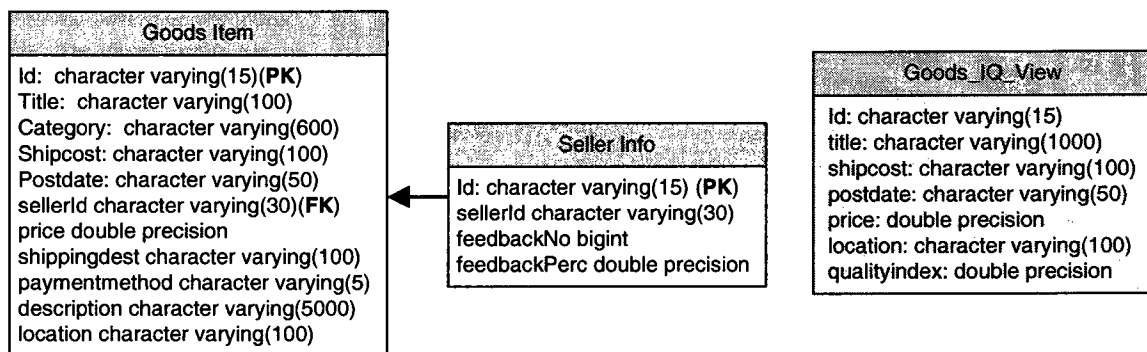


Figure 4-9: Entity Relationship Diagram of AQUAES

In the Figure 4-9, “PK” stands for “primary key”; “FK” stands for “foreign key”.

### **4.3 Summary**

This chapter describes the implementation of AQUAES system and three basic ranking strategies that are implemented in the system. We used the standard web and Java technologies to accomplish the implementation. The chapter first describes the topology and software architecture of the system. It then discusses some architectural considerations during the implementation of the system. Finally, it presents in detail the implementation of the three basic ranking strategies. The next chapter will discuss the evaluation of our proposed approach.

## Chapter 5 Evaluation

This chapter presents the extensive experimentation we have performed to measure the value of our model on both macroscopic and microscopic levels.

- On the macro level, our main objective is to test the applicability of our proposed Information Quality approach; that is, to verify the effectiveness of the approach that makes use of Information Quality to rank goods items. Here, the result of our IQ-based ranking approach is compared with that of a price-based ranking approach, which is used as the benchmark method.
- On the micro level, we investigate the relative importance of the Information Quality dimensions that we recognized for goods information in E-commerce systems. Currently, we propose four dimensions: Relevancy, Reputation, Timeliness, and Amount of Data. We would like to explore which information dimension has more impact on the ranking result. This can be accomplished by measuring only one value of Information Quality dimension, calculating the Information Quality index based on this single value, and then ranking the goods items based on the corresponding Information Quality index.

Two types of experiments, simulation evaluation and user evaluation, were conducted to evaluate the proposed model.

### 5.1 Experiment Setup

All experiments described in this chapter were conducted using our AQUAES system, which is presented in detail in Chapter 4. Two sets of experiments have been performed to evaluate the effectiveness of the Information Quality model that we propose, and to compare the four

Information Quality dimensions, respectively. Measurements of the evaluation results were taken during both experiments.

### 5.1.1 Hardware and Software Configuration

A HP personal computer was used to simulate Server 1 and Server 2 as illustrated in the system topology in Figure 4-1. The HP computer had both the AQUAES system and the PostgreSQL database installed. The computer was connected with the Internet through a 6.7 Mbit/s cable modem. The characteristics of the computer are described in Table 5.1. In our experiments, any computers that had a web browser could access the service through the Internet.

Item	Description
Operation System	Microsoft Windows XP Home Edition Version 2002 Service pack 2
Processor	Intel Pentium 4 CPU 2.93 GHz
Total Memory	1.49 GB of RAM
Network Adapter	Realtek RTL8139/810X Family Fast Ethernet NIC
Database	PostgreSQL 8.2
Java	Sun JDK 1.6

Table 5.1 Hardware and Software Information of the Server

### 5.1.2 Experimental Data

The experimental data were derived from the actual data in Ebay.com. There are two reasons that influenced our choice of Ebay.com as a data resource:

- We needed to use large data sets in our experiments. It would be time-consuming and difficult for us to create the data by ourselves. In addition, there are enormous data sets that are available in practical e-commerce systems and that meet our system

requirements (as specified in Table 4.1). Therefore, we decided to import the data from Ebay.com instead of creating the data by ourselves.

- From our perspective, actual online e-commerce systems such as eBay.com could provide more goods information compared to other systems, which would benefit our measurement of Information Quality. Moreover, the data of goods items from eBay.com has more than 18 fields including all aspects of goods information (e.g., price, shipping, payment, description, seller reputation etc.), which fully meet the AQUAES system requirements for data; whereas in other systems such as Amazon.com, only a small amount of goods information is available. Therefore, we decided to make use of the data from eBay.

### 5.1.3 Database design

We have discussed the database design of our AQUAES system in the Chapter 4. During the evaluation, we made some change in the database schema for the consideration of the automation simulations.

In the design of the database described in 4.2.3, the AQUAES system has two tables, which store information of goods and seller separately. During the evaluation, we joined these two tables into one (see Figure 5-1) for two reasons:

Firstly, since all the experimental data were imported from eBay.com, we received all these data as a whole and it would be time-consuming for us to separate them into two tables and maintain the data integrity. For the sake of simplicity, it would be easy for us to increase the fields in the “Goods Item” table to include those fields in the “Seller Info” table.

Secondly, the objective of the simulation is to test the applicability of our proposed Information Quality model, which has no relation with the database design except for the

data in the database. Therefore, the change of the database schema will not affect the result of our experiments.

During our simulations, around 2,000 goods items were imported into AQUAES from eBay.com with each item containing over 80k characters of goods information. These items were all imported and persisted in the “Goods item” table (see Figure 5-1).

In addition, a new field – “isUserSatisfied (see Figure 5-1) is also added into the table “Goods Item” for the purpose of our evaluation, for which we will discuss the reason in Section 5.2.1.

Goods Item
Id: character varying(15)(PK)
Title: character varying(100)
Category: character varying(600)
Shipcost: character varying(100)
Postdate: character varying(50)
sellerId: character varying(30)
feedbackNo: bigint
feedbackPerc: double precision
Price: double precision
Shippingdes: character varying(100)
Paymentmethod: character varying(5)
Description: character varying(5000)
Location: character varying(100)
isUserSatisfied: boolean

Figure 5-1: Goods Item table used in simulation

### 5.1.4 Measurement Method

Normally, users tend to open and review the goods items presented at the top positions in a result list. The more satisfactory (or higher quality) goods item found in the top of the list, the more satisfied the users are with the system. So, the number of the satisfactory goods in the top n results will be an appropriate measurement for the effectiveness of the approach. We made use of this measurement method in our evaluation.

In Thomas Mandl's research on a quality-based search engine [33], they use a similar idea. Their research proposes an approach for the implementation of a quality based Web search engine. They evaluate their proposed approach by comparing the result of quality ranking with that of original ranking and random ranking. Their measurement method calculates the number of high quality (grade) documents in the top 3, 5, and 10 documents listed in results list.

As to our experiments, the results were measured as the precision of the top 10, 20, and 30 search results. That is, we measured the number of the satisfactory goods items in the top 10, 20, and 30 results.

## **5.2 Simulation Evaluation**

In this section, we present our evaluation of the proposed IQ-based ranking approach via means of simulation.

### **5.2.1 Experimental Approach**

Our methodology is described as follows: In the experiments, we simulated two types of agents, the agents who only use the price-based ranking approach, and the agents who only use the IQ-based ranking approach. We also classified the goods items into two groups: one that does not satisfy the users, and one that does satisfy the users. For every sub-experiment, the system chose the same amount of goods items and combined them into a new goods list. Two types of agents then ran their ranking program on the goods list, and the result lists will be recorded by the system. Using the measurement method discussed in Section 5.1.4 above, the system could measure the results and provided the final data for the evaluation.

### **5.2.2 Experimental Goods**

Before starting the experiments, we had to decide which goods we were going to use for our experiments within AQUAES system, which allows different kinds of goods to be searched. As far as our concerns, we would like to choose goods that we are familiar with. Therefore, we sampled and experimented on two goods, Apple iPhone and Cisco 3620 Router, which we have knowledge and experiences with.

In the following experiments, we identified each goods item with a goods ID as shown in Table 5.2:

Goods Id	G1	G 2
Goods Name	Apple iPhone	Cisco3620 Router

Table5.2: Experimental Data ID

### 5.2.3 Evaluation 1: Information Quality Model

The effectiveness of our proposed Information Quality model was examined first. We intended to experimentally compare the effectiveness of the agent using the proposed quality model with that of the agent using only price as its query criteria.

#### 5.2.3.1 Experiment Process

If the user would like to buy a good, he/she will send a request to a buying agent which will then search the database and retrieve a result list. The simulation program randomly takes 50 goods items to form a list. An agent in group  $B_I$ , which use price-based ranking, and an agent in group  $B_{II}$ , which use IQ-based ranking, will rank the list respectively, and then present two separate result lists. The goods in these result lists were divided into 2 groups:

- Group A consists of goods  $g_0, g_1, \dots, \text{and } g_m$ , which do not satisfy the buyer's need. Basically, these goods items have low Information Quality, i.e., low relevance (e.g.,

the items do not match the search criteria well), low seller reputation, poor timeliness (e.g., the items have been posted on the system a long time ago), and little information.

- Group B consists of goods  $g_{m+1}$ ,  $g_{m+2}$ , ..., and  $g_{49}$  that meet the buyer's requirement for Information Quality; that is, the user would like to make further consideration after reviewing the information presented in the result list.

As discussed in 5.1.4, the simulation program calculates the number of satisfactory (or high quality) goods items, i.e., the number of goods items in Group B, from the top 10, 20, and 30 items in the results lists, which will be used to compare the effectiveness of the two approaches.

The program recognizes whether or not a goods item is user-satisfied (high quality) based on the value of the field "isUserSatisfied" (in Table "Goods Item"). Before the experiments, each of the goods items was reviewed and decided to be user-satisfied or not from our perspective. We assign the "isUserSatisfied" field as "1" if the goods item is user-satisfied (high quality), or "0" if the goods item is not user-satisfied (low quality). So, the simulation program need only to count how many goods items, whose "isUserSatisfied" value is "1", are in the top n items.

### 5.2.3.2 Experimental Parameters

We simulated an e-commerce system populated with 120 buying agents, which were divided into two groups:

- Group I consists of buying agents  $b_0$ ,  $b_1$ , ..., and  $b_{59}$ . These agents use the price-based approach alone to rank the goods items in a list formed by the simulation program. Recall that the price-based approach is one that ranks goods items according to their prices, normally in increasing order. This set of buying agents is described as:

$$B_I = \{ b_0, b_1, \dots, b_{59} \}$$

- Group II consists of buying agents  $b_{60}$ ,  $b_{61}$ , ..., and  $b_{119}$ . These agents use our proposed IQ-based approach to rank the goods items in the list. This group is described as:

$$B_{II} = \{ b_{60}, b_{61}, \dots, b_{119} \}$$

The parameters in equation (3.6) used to calculate the quality index of the goods item are set as follows:

- For the IQ-based approach, we set the weights of four quality dimensions in equation (3.6),  $W_{mir}$ ,  $W_{mip}$ ,  $W_{mit}$ ,  $W_{mia}$  as equal values; that is,  $W_{mir} = W_{mip} = W_{mit} = W_{mia} = 1$ . The reason to set their values equal is because our goal for this experiment is to test the effectiveness of the proposed Information Quality model, which combines all quality dimensions. Therefore, we don't want to differentiate the quality dimensions in weights, which will be the objective of the next experiment. Also, we set these weights to 1 in order to simplify the calculation in equation (3.6).

### 5.2.3.3 Results

For each evaluation, the system randomly chose 25 goods items from each of the two goods groups A and B separately, and combined them into a new goods list. Two agents (from the sets  $B_I$  and  $B_{II}$  respectively) then ran their ranking programs on this goods list, and their result lists were recorded by the system.

This evaluation process was run for 60 times. All the recorded results were added together and summarized in Table 5.3 below.

In this table, column “ $B_I$ ” indicates the number of satisfactory goods items in the result lists handled by agents in group  $B_I$  (which ranks goods items based on their prices). Column “ $B_{II}$ ” indicates the number of satisfactory goods items in the result lists handled by agents in group  $B_{II}$  (which use the proposed IQ-based approach to rank goods items). Each “First n

Results” row shows the number of satisfactory goods in the top n results. Rows G1 and G2 indicate that the results were received from the experiments conducted on product G1 and G2, respectively. The data in column “Improvement” show the difference between the data in column  $B_I$  and the data in column  $B_{II}$ , calculated from the same row. When the difference is a positive number, it demonstrates that group  $B_{II}$  of agents obtains better results than group  $B_I$ . This further suggests that the approach used by agents in group  $B_{II}$  is more effective than that used by agents in group  $B_I$ , and vice versa.

ResultSet used	Goods Id	$B_I$	$B_{II}$	Improvement
First 10 Results	G1	117	134	17
	G2	98	313	215
First 20 Results	G1	219	356	137
	G2	218	475	257
First 30 Results	G1	234	398	164
	G2	356	579	225

Table 5.3: Experimental Results from IQ-Based Approach and Price-Based Approach

It can be observed that each value in “Improvement” column is a positive number. These numbers suggests two things: first, the IQ model we propose can improve the effectiveness of agent in retrieving goods item; second, the proposed Information Quality model is applicability in the context of agent systems.

The results in Table 5.3 also reveal that the improvement is increasing as the size of the result set increases. This increasing is significant between the first 10 and first 20 results, while not so dramatic between the first 20 and 30 results.

In order to present the results found and shown in the Table 5.3 intuitively, we expressed them in the Figure 5-2 below. In this figure, the purple line connected all the data received

from the quality-based approach, while the blue line connected all the data received from the price-based approach. We can see that the purple line is always above the blue line.

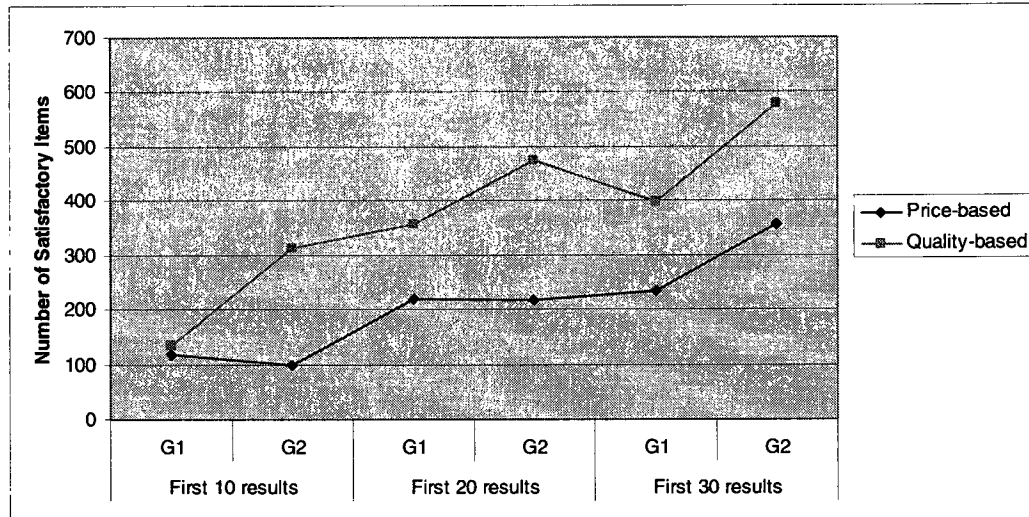


Figure 5-2 Experimental Results from IQ Model and Price-Based Model

### 5.2.4 Evaluation 2: Comparison of the Quality Dimensions

In the previous section, we have evaluated and showed that our proposed IQ-based approach is more effective in ranking goods items. Now, we would like to take a further step in examining our approach. We know that, the IQ-based approach ranks the goods items based on their quality index, which is calculated using equation (3.6). In this equation, we presented four information dimensions. We now would like to explore whether any one dimension would have more impact on the ranking result than the others.

In this section, we are interested in examining the relative importance of the four recognized individual Information Quality dimensions. We therefore intend to experimentally compare the ranking results that uses only one quality dimension each time. Consequently, four set of experiments need be conducted separately, and each time only one quality dimension incorporated.

### 5.2.4.1 Experiment Parameters

In order to consider only one dimension at a time, we set the weight of the queried quality dimension to 1 and the weights of other quality dimensions to zero. For instance, the parameters are set as follows:

- If we would like to examine the Relevancy dimension, we set the weights of the four quality dimensions (as described in equation (3.6))  $W_{mir}$ ,  $W_{mip}$ ,  $W_{mit}$ ,  $W_{mia}$  as  $W_{mir}=1$ ,  $W_{mip} = W_{mit} = W_{mia} = 0$ . Therefore, each time the Information Quality value is calculated based only on the value of the Relevancy dimension.

In addition, since we only conduct experiments on the IQ – based ranking approach, there will be only one type of buying agents populated in the system, i.e., those agents that use the IQ-based approach.

### 5.2.4.2 Experiment Process

Similar to the first experiment, we experimented on two goods, G1 and G2. During the experiment, simulation program retrieves 50 goods items respectively from the G1 and G2 goods sets, calculates the quality index of each items using one dimension, and then ranks these items according to their values of quality index. For the other three dimensions, we follow suite and get three more result lists. The simulation program counts the number of satisfactory goods items in the first 10 and 20 top results and recorded them, and then we can compare the dimensions using these data.

### 5.2.4.3 Results

The experimental results are summarized in Table 5.4. The table shows the results of the experiment on individual IQ dimension, namely Relevancy, Reputation, Timeliness, and

Amount of Data. The data in each column is the sum of the satisfactory goods items after 60 runs in the corresponding dimension. The results are observed in top 10 and top 20 results.

ResultSet used	Goods Id	Relevancy	Timeliness	Reputation	Amount of Data
First 10 Results	G1	117	103	98	84
	G 2	209	186	193	105
First 20 Results	G 1	128	122	96	109
	G 2	473	395	308	438

Table 5.4: Experimental Results from Individual Information Quality Dimensions

This table suggests that the number of satisfactory goods items obtained by using Relevancy dimension is a slightly higher those that are obtained by using the other three dimensions. Furthermore, the other three dimensions have almost equal effect on the ranking result. These results are illustrated as Figure 5-3 below.

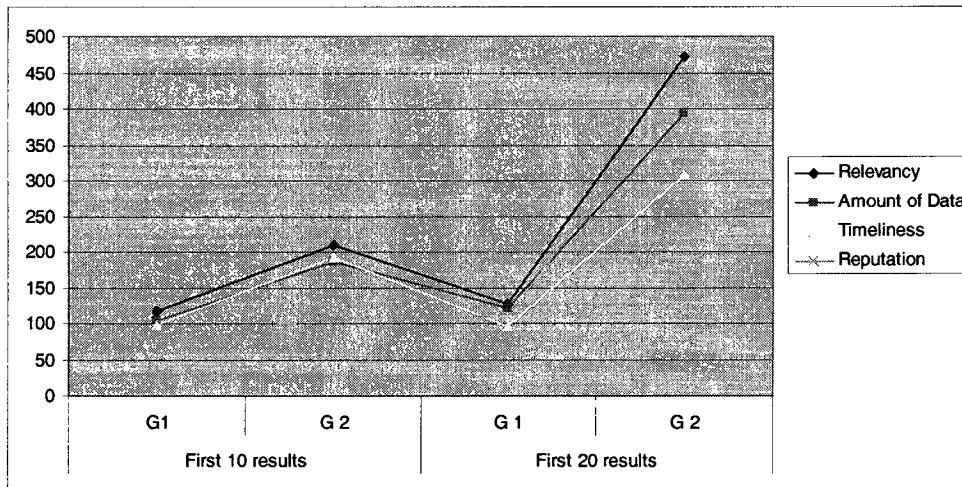


Figure 5-3 Experimental Results on IQ Dimensions

### 5.3 User Evaluation

Besides the simulation evaluation, a user evaluation was also carried out to allow real users to test on different ranking approaches (price-based and quality-based) and collect experiments data. We are therefore able to experimentally compare the satisfaction of users who use the proposed Information Quality based approach with the satisfaction of users who only use the normal price-based approach. The purpose of this evaluation is:

- Testing the effectiveness of the proposed approach with real users. Due to the fact that the quality assessment and interpretation of the proposed model are based on users' requirements, it will be ideal that the quality of the results be assessed by users in the context of their search.
- Exploring whether the results from user evaluation would be consistent with those from simulation evaluation (described in Section 5.2).

Similar to the simulation evaluation, AQUAES system (described in Section 4.1) was also used for the user evaluation. Two ways of ranking goods items were presented to the users. The ranking was either based on Information Quality values or based on the goods' prices.

Two sets of experiments were conducted to evaluate our Information Quality model as a whole, and to compare the Information Quality dimensions individually.

### 5.3.1 Experimental Design

The experimental design for user evaluation is similar to simulation evaluation in terms of experimental goods, experimental parameters, and experimental process. The only difference is that it uses real people to send requests and count result sets instead of using a simulation program.

During user evaluation, test users were told to search for specified goods and count the number of goods items in the result set which satisfied their requirements. Here comes a question: How does a user decide whether a goods item is satisfactory or not? We define the satisfactory goods item as the one that the user would like to make further consideration after

reviewing some information on the item. So after the user opens a goods item and looks through the description in the items, the user would know if he/she is satisfied with the item or not. Experiment instructions had been provided to the test users before they began the evaluation process (see Appendix A). These instructions include the instruction on the experimental process, the suggested experimental parameters, as well as the experimental data collection approach.

It is important to mention that test users were only asked to search for satisfactory goods and to record their results. Consequently, they did not have any clue on what our objective was and what we were trying to do with their result data. By this way, we could avoid the situation where testers were bias and reported the results dishonestly.

### 5.3.2 Test User

We have talked with 20 people and a group of 10 people was chosen to participate in experiments. In order to decide who are suitable to be a tester, we set two criteria:

- Firstly, a tester should have knowledge of e-commerce systems, and
- Secondly, he/she should have experience with online shopping.

We believe that if a tester meets these criteria, it will be easy for him/her to understand our experimental environment and be able to count the satisfactory goods based on their experiences.

All ten people that we chose meet these criteria: they all have knowledge of e-commerce systems and have purchased goods online successfully in some practical e-commerce systems, such as eBay, Taobao.com<sup>7</sup> etc.

### 5.3.3 Experimental Goods

---

<sup>7</sup> <http://www.taobao.com>

Similar to simulation evaluation, during user evaluation we need to choose some goods to perform the test. In our experiment, we sampled three goods, Apple iPhone, Cisco2610 Router, and Cisco 3620 Router. There are three reasons why we chose these goods:

- We would like to keep things consistent and be able to compare the results with those in our simulation evaluation (where we chose Apple iPhone, and Cisco 3620 Router for our experiments)
- We took into consideration the knowledge and experience of our test users with respect to the chosen goods. We found that all test users possess cell phones, seven of them have strong knowledge of Cisco products, and four of them had purchased Cisco routers online. Also, they all are quite familiar with the chosen goods.
- We sampled two similar goods, Cisco 2610 and Cisco 3620, in order to explore if the ranking approaches have different effectiveness on different goods type.

For each goods, we provide an ID as shown in Table5.5:

Goods Id	G1	G 2	G 3
Goods Name	Apple iPhone	Cisco2610 Router	Cisco3620 Router

Table 5.5: Experimental Goods and their ID

### 5.3.3 Experimental Process

During the evaluation, each test user carried out two types of experiments: Evaluation of the Information Quality Model and Comparison of the Quality Dimensions. For each experiment type, three experiments were conducted on the three goods listed in Table5-5.

For each experiment, a test user input the goods name and the predefined parameters of preference<sup>8</sup>. Our AQUAES system thus could retrieve two result lists corresponding to the price-based ranking approach and our IQ-based ranking approach, respectively. The test user then counted the number of satisfactory goods items in each result list. Within user evaluation, we applied the measurement method discussed in Section 5.1.4. the user was requested to count the satisfactory number of goods items in the first 10, 20, and 30 result sets.

At the end, the mean value of the numbers recorded by all test users are calculated and analyzed (see Section 5.3.4).

### 5.3.4 Evaluation 1: Information Quality Model

The effectiveness of our proposed Information Quality model was examined first. Similar to the simulation evaluation, in this evaluation we intended to experimentally compare the effectiveness of the agent using the proposed quality model with the effectiveness of the agent using only price as its query criteria.

#### 5.3.4.1 Experiment Parameters

For the experiment using the proposed Information Quality model, the Information Quality is calculated as the combination of all the dimensions, and we asked the test users to set the weights of all four quality dimensions to equal value (the reason is the same as what we discussed in 5.2.3.2); that is, to set the weights of the four quality dimensions to one:  $W_{mir} =$

$$W_{mip} = W_{mit} = W_{mia} = 1.$$

---

<sup>8</sup> These are the same as parameters as we described in 5.2.3.2 and 5.2.4.1. We presented them to the users before the experiments. Wee Appendix A - Experiment instructions for test users for more detail.

### 5.3.4.2 Experiment Measurement and Results

We use the same measurement method as discussed in 5.1.4. In this experiment, the result was measured as the number of the goods items in the top 10, 20, and 30 results that the test users feels meet their requirement and would like to consider it further. This way we can explore the effect of the quality ranking in different set levels.

During the experiment, according to the experiment instructions that we presented (see Appendix A), each test user typed their query for 3 goods respectively. AQUAES system then presented the result list that is ranked on quality and price; the test user counted the number of the satisfactory goods items in the first 10, 20, 30 items respectively, and write them down. In the end, we added these numbers and calculated the average value.

The results are summarized in Table 5.6. In this table, the “Price-Based Ranking” column indicates the average numbers of satisfactory goods items in the result sets that are retrieved based only on goods prices. The “Quality-based Ranking” column indicates the average numbers of satisfactory goods items in the result sets that are retrieved based on the Information Quality of goods items. Each “First n results” row indicates the number of satisfactory goods items in the top n results. The G1, G2 and G3 rows indicate the results conducted on product G1, G2, and G3. The “Improvement” column shows the difference between the data in the “Price-Based Ranking” column and the data in the “Quality-based Ranking” column, calculated from the same row.

ResultSet Used	Goods Id	Price-based Ranking	Quality-based Ranking	Improvement
First 10 results	G1	2	2.1	0.1
	G2	6.8	7.8	1
	G3	1	7.2	6.2
	Avg	3.3	5.7	2.4
First 20 results	G1	6.9	8.1	1.2
	G2	14.3	15.6	1.3
	G3	3.7	11.1	7.4
	Avg	8.3	11.6	3.3
First 30 Results	G1	6.7	8.4	1.7
	G2	22.3	23.6	1.3
	G3	7.3	14.7	7.4
	Avg	12.1	15.57	3.47

Table 5.6: Experimental Results from IQ Model and Price-Based Model

It can be observed, that each value in “Improvement” column is greater than zero, which suggests that the proposed IQ model can improve the effectiveness of buying agents in retrieving goods items. This also implies the applicability of the proposed Information Quality model in e-commerce agent systems. The results in the table also reveal that the improvement is increasing as the result set increases. This increasing is large between the first 10 and 20 results, while not so dramatic between first 20 and 30 results. These results are quite consistent with those found from the simulation evaluation (Section 5.2).

From Table 5.6, we also notice that although G2 and G3 belong to same product category, the Information Quality model has different effect on their ranking results. It seems that the result of the Information Quality ranking approach has no direct relation with the goods categories.

The results in Table 5.6 were expressed graphically in Figure 5-4. In the figure, the purple line connected all the data received from the quality-based approach, while the blue line connected all the data received from the price-based approach. Note that the purple line is always above the blue line, which means that the quality-based ranking has better effectiveness than the price-based ranking.

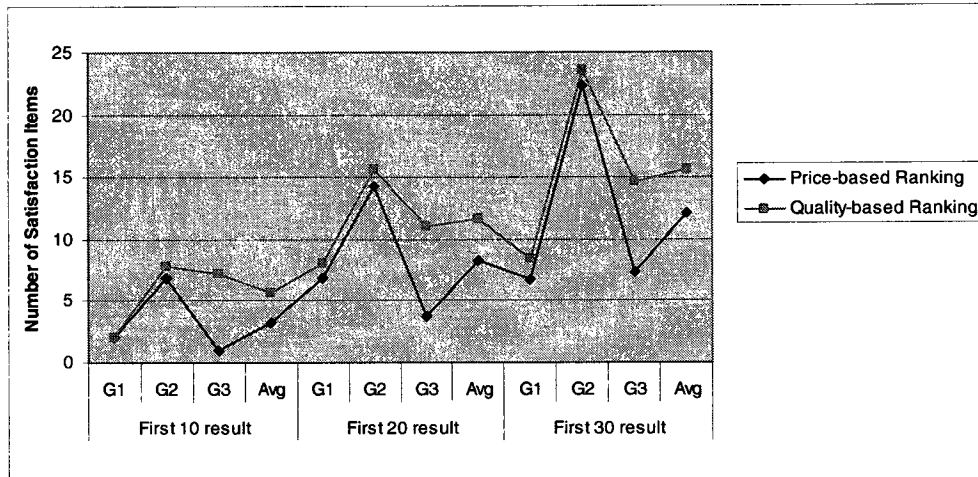


Figure 5-4: Experimental Results from IQ Model and Price-Based Model

### 5.3.5 Evaluation 2: Comparison of the Quality Dimensions

This evaluation is the counterpart of the one discussed in 5.2.4. In this section, we are interested in examining the relative importance of the four recognized individual Information Quality dimensions, i.e., whether or not one dimension has more impact on the others.

#### 5.3.5.1 Experiment Process and Parameters

In order to compare the four quality dimensions, test users queried every goods four times, and each time only one quality dimension was involved. Following the experiment instructions (Appendix A), each time test users set one weight of the queried quality dimension to 1 and the weights of others to zero.

Similar to the previous experiment, test users typed their query, counted the numbers of satisfactory goods items in the first 10 and 20 items of the result set, and wrote them down. In the end, they added these numbers and calculated the average value.

### 5.3.5.2 Experiment Results

The results of this experiment are summarized in Table 5.7. The table shows the results on individual IQ dimensions, Relevancy, Timeliness, Reputation, and Amount of Data, respectively. Each column indicates an individual dimension. The results are observed in the top 10 and top 20 results.

ResultSet Used	Goods Id	Relevancy	Timeliness	Reputation	Amount of Data
First 10 results	G1	2.3	2.1	2	1.8
	G 2	10.1	2.9	6.8	5.9
	G 3	5.4	5.3	5.4	3.2
First 20 results	G 1	3.3	3.7	2	3.2
	G 2	17.2	10.2	13.4	11.7
	G 3	9.4	8.9	7.3	9.6

Table 5.7: Experiment on Information Quality Dimensions

The following Figure5-5 graphically depicts the results in Table5.7. The figure suggested that the number of satisfactory goods items obtained by using Relevancy is higher than using the other three dimensions. Furthermore, the other three dimensions have almost equal effect on the ranking results.

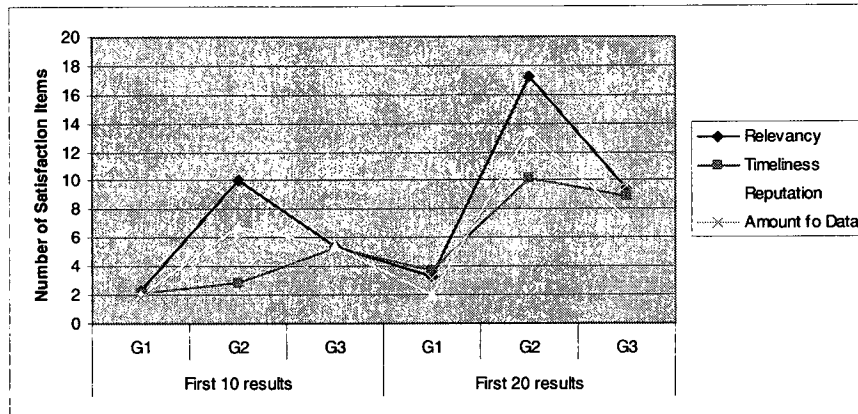


Figure 5-5: Experiment on IQ dimensions

When comparing Table 5.6 with table 5.7, we notice that the search effectiveness can be improved when ranking is conducted on the combination of all the quality dimensions rather than one only one quality dimension.

## 5.4 Summary

This chapter presents the evaluation of our Information Quality approach. Two types of evaluation, simulation evaluation and real user evaluation were conducted. For each type of evaluation, two experiments were carried out in order to explore the effectiveness of the proposed IQ model in different levels. The experiment environment, data, measurement method, and experiment process are discussed. The experiment results are illustrated. As confirmed by the experimental results, both simulation evaluation and user evaluation suggest that our proposed IQ model is more effective in ranking goods items. In addition, we found that among the four Information Quality dimensions, the Relevancy dimension has slightly greater impact than the other three dimensions.

## **Chapter 6 Conclusions and Future Work**

### **6.1 Summary**

In summary, the IQ-based approach proposed in this thesis provides a feasible and effective way in sorting goods items of e-commerce systems. This was successfully demonstrated through the implementation of a prototype system and the evaluation of the proposed approach.

Realizing the issues existing in current agent-based systems, this research, from the goods information point of view, ranks goods items based on the Information Quality dimensions. According to our best knowledge, this is the first time Information Quality was considered in buying agent e-commerce systems. In our study, information is analyzed as a whole, which considers not only goods price or seller reputation but also all the content fields included in the goods items. This approach presents a new solution to deal with goods items and fits well with the decision making behaviour of buyers, since buyers usually make decision based on the whole information that they have and not on the price or reputation alone.

As an initial study, our research begins from identifying IQ dimensions, and then conducts all other aspects of IQ technology research, including analyzing IQ metrics, designing IQ assessment and interpretation model. In this thesis we also propose a framework that implements information retrieving, IQ assessment, and IQ ranking. Although the framework is originally designed to use for buying agents in C2C e-commerce applications, we think that it is also appropriate for B2C and B2B applications, because in B2C and B2B environments, buying agents would also benefit from using this Information Quality based approach to rank goods items with suitable quality dimensions and metrics in their own applications.

We built an agent-based e-commerce system, AQUAES, which implements our proposed Information Quality model. Two set of experiments were conducted to evaluate the proposed model. This work is driven by buying agents' needs for better and effective solutions in searching and retrieving satisfactory goods. The results of our evaluation show that the proposed Information Quality model does improve the effectiveness of goods items retrieval by buying agents.

## 6.2 Future Work

As stated above, this is the first time Information Quality is applied in the area of goods items ranking. We studied several aspects of the Information Quality research in order to propose an IQ-based ranking approach. We realize that our research is only an initial work and many further studies could be conducted. Using our work as a starting point, some future research directions are outlined as follows:

- In the current study, we recognize four quality dimensions in the agent systems, and then apply them in our research. As discussed in Section 2.1.2, there are more than twenty different dimensions recognized for different applications. We believe that, there might be other dimensions which are also very useful or perhaps critical for agent systems. Completeness could be one of such dimensions. As discussed in Section 3.2.1, we think that this dimension is appropriate for this research; however, we just could not get suitable dataset for testing. Thus, further research on the recognition of Information Quality dimensions could be taken into consideration as future work.
- Furthermore, the relation among the four dimensions in this domain needs to be explored. Currently we treat every dimension as a single factor independent of each other. But it is possible that some relation exists within these dimensions. For the future research, through the approach of combining any two dimensions and changing their weights, we might be able to explore this issue further.

- During the evaluation experiments, we only investigated the model efficiency under two situations: one where the four quality dimensions have equal weight and the other where only one dimension is considered. Nevertheless, we have not yet explored the performance of the proposed Information Quality model in other situations, such as what if we incorporate all the dimensions, but give a different weight for each dimension. We, therefore, think that further evaluation methods could be useful.
- Price plays a very important role on buyers' decision making. However, in the Information Quality domain, it doesn't exist as a dimension, and in our study, it doesn't get involved. We assert that price should be taken into account when dealing with e-commerce transactions. We expect a solution to be proposed to integrate the price attribute with the Information Quality.
- As mentioned in Section 2.3, in this thesis we focus our research on the Information Quality in centralized buying-agent systems. Therefore the proposed quality model and the dimensions developed so far are only suitable for measuring the quality of goods items in these systems. In-practice, however, there exist several distributed buying-agent systems. We expect that our approach could be used for these systems as well. Thus, further study of Information Quality in distributed buying-agent systems should be investigated. Based on our experience, we suggest that this research could begin from analyzing the information format in different systems and recognizing Information Quality dimensions.
- It has been showed that more and more experimental systems are being developed on Information Quality study [1, 2, 4, 14, 15, 35, and 37]. Among these systems, from our point of view, the one that based on machine learning are the most promising one [4]. In the current stage of our research, the assessment on each information dimension is based on a simple mathematical function. We think that machine learning could be a promising direction, for example, on the research of the Relevancy dimension, machine learning may help for the assessment process.

## References

- [1] Al-Hakim, L., et al. (2007). *Information Quality Management: Theory and Applications* T58.64. I5287 2007 Idea Group Publishing ISBN 1-59904-024-7
- [2] Cappiello, C., Francalanci, C., & Pernici, B. (2004). *Data quality assessment from the user's perspective* (Jun 2004) Proceedings of the 2004 international workshop on Information quality in information systems Publisher: ACM Press
- [3] Dasgupta, P., & Hashimoto, Y. (2004). *Multi-attribute Dynamic Pricing for Online Markets Using Intelligent Agents*. In Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS-04), 2004.
- [4] Dasu, T., Vesonder, G., Wright, J. (2003). *Data quality through knowledge engineering* (Aug 2003) Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining Publisher: ACM Press
- [5] Gertz, M., Özsu, M., Saake, G., & Sattler, K. (2004). *Report on the Dagstuhl Seminar: "data quality on the Web"* (Mar 2004) ACM SIGMOD Record, Volume 33 Issue 1 Publisher: ACM Press
- [6] He, M. & Leung, H. (2002). *Agents in E-Commerce: State of the Art*. In Knowledge and Information Systems, Vol. 4, pp. 257-282, 2002.
- [7] Items in Ebay.com, Retrieved March 21, 2008, from <http://www.ebay.com>
- [8] Jespersen, S., Pedersen, B., & Thorhauge, J. (2003). *Evaluating the markov assumption for web usage mining* (Nov 2003) Proceedings of the 5th ACM international workshop on Web information and data management Publisher: ACM Press
- [9] Jonker, C. & Robu, V. (2004). *Automated Multi-Attribute Negotiation with Efficient Use of Incomplete Preference Information*. In Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS-04), 2004.
- [10] Katerattanakul, P. & Siau, K. (1999). *Measuring information quality of web sites: development of an instrument* (Jan 1999) Proceeding of the 20th international conference on Information Systems Publisher: Association for Information Systems
- [11] Kim, Y., Kishore, R., & Sanders G. (2005). *From DQ to EQ: understanding data quality in the context of e-business systems* (Oct 2005) Communications of the ACM, Volume 48 Issue 10 Publisher: ACM Press
- [12] Lee, W., Bowen, P., Funk, J., Jarke, M., Madnick, S., Wand, Y. (2000). *Data quality in internet time, space, and communities* (panel session) (Dec 2000) Proceedings of the twenty first international conference on Information systems Publisher: Association for Information Systems

## References

---

- [13] Lee, Y. & Kozar, K. (2005). *Investigating the effect of website quality on e-business success: An analytic hierarchy process (AHP) approach* Decision Support Systems (2005) 11204; 19-38
- [14] Liu, Y., Wang, C., Zhang, M., & Ma, S. (2005) *Web data cleansing for information retrieval using key resource page selection* (May 2005) Special interest tracks and posters of the 14th international conference on World Wide Web Publisher: ACM Press
- [15] Majki', Z. (2004). *A general framework for query answering in data quality-based Cooperative Information Systems* Proceedings of the 2004 international workshop on Information quality in information systems June 2004 Publisher: ACM Press
- [16] Martinez, A. & Hammer, J. (2005). *Making quality count in biological data sources* (Jun 2005) Proceedings of the 2nd international workshop on Information quality in information systems IQIS '05 Publisher: ACM Press
- [17] Na, K., Baik, D., & Kim, P. (2001). *Posters and Short Papers: A practical approach for modeling the quality of multimedia data* (Oct 2001) Proceedings of the ninth ACM international conference on Multimedia Publisher: ACM Press
- [18] Orr, K. (1998). *Data quality and systems theory* (Feb 1998) Communications of the ACM, Volume 41 Issue 2 Publisher: ACM Press
- [19] Pages in Amazon.com, Retrieved at March April 3, 2008, from [http://www.amazon.com/gp/offer-listing/0201633612/ref=dp\\_olp\\_new?ie=UTF8&condition=new](http://www.amazon.com/gp/offer-listing/0201633612/ref=dp_olp_new?ie=UTF8&condition=new)
- [20] Pages in Ebay.com, Retrieved at March April 3, 2008, from [http://search.ebay.com/search/search.dll?from=R40&\\_trksid=m37.11313&satitle=cisco+3620&category0=](http://search.ebay.com/search/search.dll?from=R40&_trksid=m37.11313&satitle=cisco+3620&category0=)
- [21] Pages in fetchbook.info, Retrieved at March April 3, 2008, from <http://www.Fetchbook.info/search.do.jsessionid=43D3F331FD57A41AA1D2CF810D7A3A1D?search=scjp&searchBy=Title>
- [22] Pages in Google.Shopping, Retrieved at March April 3, 2008, from <http://www.google.com/products?q=apple+iphone>
- [23] Parssian, A., Sarkar, S., & Jacob, V. (1999). *Assessing data quality for information products* (Jan. 1999). Proceeding of the 20th international conference on Information systems Publisher: Association for Information Systems
- [24] Perspective in Cambridge Dictionaries Online, Retrieved at June 30, 2007, from <http://dictionary.cambridge.org/define.asp?key=59094&dict=CALD>
- [25] Pierce, E. (2004). *Assessing data quality with control matrices*. Communications of the ACM /Vol. 47, No. 2 (February 2004), 82-86
- [26] Pipino, L., Lee, Y., & Wang, R. (2002). *Data quality assessment* (Apr 2002) Communications of the ACM, Volume 45 Issue 4 Publisher: ACM Press

- [27] Regan, K., Tran, T., & Cohen, R. (2005). *Sharing Models of Sellers amongst Buying Agents in Electronic Marketplaces*. In Proceedings of the 10th International Conference on User Modeling (UM'2005) Workshop on Decentralized, Agent Based and Social Approaches to User Modeling
- [28] Sabater, J. & Sierra, C. (2001). *REGRET: A Reputation Model for Gregarious Societies*. In Proceedings of the 5th International Conference on Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies, 2001.
- [29] Scannapieco, M. & Berti-Équille, L. (2006). *Report from the First and Second International Workshops on Information Quality in Information Systems: IQIS 2004 and IQIS 2005 in conjunction with ACM SIGMOD/PODS Conferences* (Jun 2006) ACM SIGMOD Record, Volume 35 Issue 2 Publisher: ACM Press
- [30] Strong, D.M., Lee, Y.W., & Wang R.Y. (1997). *Data quality in context*. Communications.ACM 40, 5 (May 1997), 103–110.
- [31] Talburt, J.R., et.al.. *Creating a new program with industry and research community support*
- [32] Tayi, G.K. & Ballou D.P.(1998). *Examining data quality* (Feb 1998) Communications of the ACM, Volume 41 Issue 2 Publisher: ACM Press
- [33] Thomas, M. (2006). *Implementation and evaluation of a quality-based search engine*. (August 2006). ACM 1-59593-417-0/06/0008, 73-84.
- [34] Tran, T. & Cohen, R. (2004). *Improving User Satisfaction in Agent-Based Electronic Marketplaces by Reputation Modeling and Adjustable Product Quality*. In Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'2004), Volume 2, pp. 828-835.
- [35] Veronika, P. & Mokrane, B. (2004). *A framework for analysis of data freshness*. IQIS 2004 Maison de la Chimie, Paris, France. ACM 1-58113-902-0/04/0006, 59–67.
- [36] Wand, Y. & Wang, R.Y. (1996). *Anchoring data quality dimensions in ontological foundations*. Commun. ACM 39, 11 (Nov. 1996), 86–95.
- [37] Wang, R.Y. (1998). *A product perspective on total data quality management*. Commun. ACM 41, 2 (Feb. 1998), 58–65.
- [38] Wikipedia E-commerce definition, retrieved at March 23, 2008, from <http://en.wikipedia.org/wiki/E-commerce>
- [39] Zhu, X. & Gauch, S. (2000). *Incorporating Quality Metrics in Centralized/Distributed Information Retrieval on the World Wide Web*. In Proc. ACM Conference on Research and Development in Information Retrieval (SIGIR 2000), Athens.288 -295

## Appendices

### Appendix A: Experiment Instructions for Test Users

In chapter 5, we discussed two types of evaluation, one of which is user evaluation, which allows real users to evaluate our prototype system, AQUAES, and collect experimental results for us. This appendix displays the experiment instructions that we have presented to test users before their experiments.

The instructions include experimental system, experimental process, the suggested experimental parameters, and the experimental data collection approach.

#### 1. Experimental system introduction:

After inputting <http://localhost:8080/AQUAES/> in your IE or Firefox, you will see a web page as Figure A-1. You can input the goods name that you want to search for, and specify some detail requirement in the “Quality profile” area. Once you click “Price Based Ranking” or “Quality Based Ranking”, the system will present you search result as shown in Figure A-2.

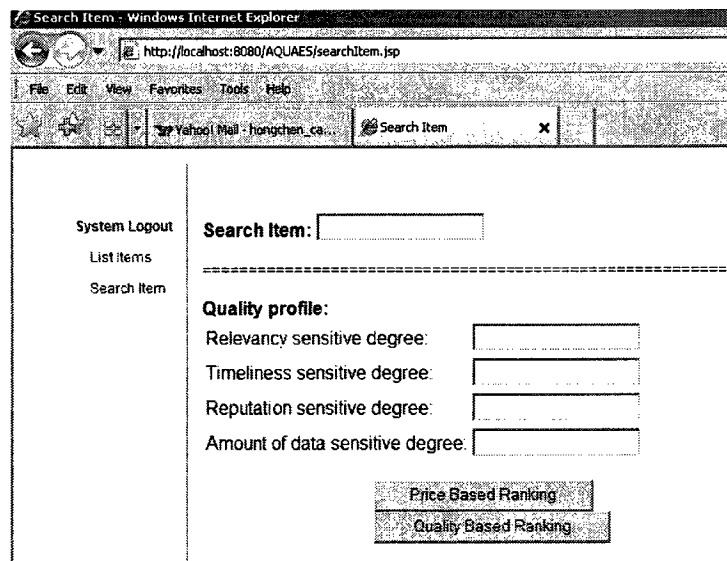


Figure A-1: AQUAES first pages

Name	SellerId	PostDate	Location
1 Cisco 3620 Series Integrated Services Router - Used	seprat_inc	Apr-02-08 19:48:21 PDT	Maitland, FL, United States
2 CISCO 3620 Cisco 3620 Router 24-16	techtire	Apr-03-08 11:59:38 PDT	Austin, United States
3 Cisco 3620 Router w/ 18F84D	razroc	Apr-05-08 16:55:38 PDT	Orange, CA, United States
4 CISCO 3620 Router: 3600 Series (Great Condition!!	centricst	Apr-04-08 07:15:10 PDT	Norcross, GA, United States
5 CISCO 3620 Router: 3600 Series (Great Condition!!	centricst	Apr-03-08 16:07:13 PDT	Norcross, GA, United States
6 CISCO 3620 Router: 3600 Series (Great Condition!!	centricst	Apr-05-08 16:07:50 PDT	Norcross, GA, United States
7 Cisco NM-4AS Module for 2600 2900M 3620 3640 3621 3660	candelanet	Apr-04-08 06:26:18 PDT	Brooklyn, New York, United States
8 NM-2FE2T Cisco 3600 3620 3640 Network Module V3 2 FE1	horizontals.com	Apr-02-08 12:06:12 PDT	Columbus, Ohio, United States
9 Cisco 3620 2 Slot Modular Router 7 Ethernet Ports	horizontals.com	Apr-03-08 12:04:35 PDT	Columbus, OH, United States
10 Cisco NM-1E2W Module for 3600 3620 3640 routers. Genuine	router-certs	Apr-02-08 21:35:42 PDT	NJNY, United States
11 Cisco 3620 64-16 12.3.22 IPFW/IDS/DES CCNA COMP 3640	router-certs	Apr-03-08 08:29:00 PDT	Baltimore, New Jersey, United States
12 Cisco 3620 AC power supply PWR-3620-AC	primarunit	Apr-03-08 17:19:11 PDT	Lancaster, California, United States
13 Cisco NM-1E-FC Module for 3620 3640 1-Yr WARRANTY	caudelanet	Apr-04-08 12:00:12 PDT	Brooklyn, New York, United States
14 CISCO 3620 w/32mb 5 Smb memory NM-1E2W	nuseriesale	Apr-03-08 22:16:59 PDT	San Antonio, United States
15 Cisco 3600 Series Router 3620	ehaptic	Apr-03-08 14:39:04 PDT	Columbus, United States
16 Cisco 3600 Series 2 Slot Modular Router 3620	svcsare	Apr-03-08 18:36:36 PDT	San Jose, CA, United States
17 Cisco 3620 Router 3600 Series CISCO 3620 CCNA WARRANTY	andovercp	Apr-04-08 17:16:43 PDT	San Francisco, CA, United States
18 CISCO 3620 NM-2E2W-16S CCNA COMP WARRANTY GG	maasitechnologies	Apr-03-08 16:26:31 PDT	Oklahoma City, OK, United States
19 CISCO NM-2E2W-16S 3620 3640 - WARRANTY GG	maasitechnologies	Apr-05-08 11:20:34 PDT	Oklahoma City, OK, United States
20 Cisco router 3620 with 2E2W & 1E2W	all_disc	Apr-05-08 06:26:34 PDT	San Diego, California, United States
21 91119 - CISCO 3620 CCNA 3616 MAX MEM - WARRANTY GG	maasitechnologies	Apr-03-08 10:22:45 PDT	Oklahoma City, OK, United States
22 CISCO 3620 LAB ROUTER CCNA COMP COE HON DOA 3640 2600	mintcar	Apr-03-08 16:28:54 PDT	Rochester, NY, United States
23 Cisco 3640998-01 PWR-3620-AC AC Power Supply for 3620	scs_ene	Apr-05-08 10:03:32 PDT	Phoenix, AZ, United States
24 CISCO 3620 LAB ROUTER CCNA COMP COE HON DOA 3640 2600	mintcar	Apr-03-08 17:53:31 PDT	Rochester, NY, United States
25 32MB Flash 2x MEM3600-16S for Cisco 3600 3620 3640 DTY	inelus	Apr-03-08 14:00:00 PDT	Brooklyn, New York, United States
26 4x MEM3620-16D 64MB DRAM Kit for Cisco 3600 3620 4SKCTY	inelus	Apr-03-08 14:00:00 PDT	Brooklyn, New York, United States
27 USED CISCO 3620 ROUTER CISCO 3600 SERIES 3620 1-YR WNTY	networkdivaresale	Apr-07-08 15:24:06 PDT	Santa Barbara, CA, United States

Figure A-2: AQUAES search result

In Figure A-2, by clicking on any goods name, you will be able to look at the detailed information about that goods item.

## 2. Experimental process and data collection:

**First Experiment:** Please input “Apple iPhone” in “search item” of Figure A-1, then click on “Price Based Ranking” button, a result set will appear. In the result set, review each of goods items in the first 10 results, count the number of goods items that you are satisfied with, and then write the number in “A11” of Table A.1. Similarly, count the number of satisfactory goods items in first 20 and first 30, and write the number in “A21” and “A31” of Table A.1.

Back to the first web page, input “Apple iPhone” in “search item”, and at the same time, input “1” in the four fields shown in the “Quality profiles” area, then click “Quality Based Ranking” button. In the result set that system presents to you, review each item, count the number of satisfactory goods items in the first 10, 20 and 30 results, and write the number in “A12”, “A22”, and “A32” of Table A.1.

Repeat the previous steps with “Cisco2610 Router” and “Cisco3620 Router”, record the numbers in TableA.1.

Result	Goods Id	Price-based Ranking	Quality-based Racking
First 10 results	Apple iPhone	A11	A12
	Cisco2610 Router		
	Cisco3620 Router		
First 20 results	Apple iPhone	A21	A22
	Cisco2610 Router		
	Cisco3620 Router		
First 30 results	Apple iPhone	A31	A32
	Cisco2610 Router		
	Cisco3620 Router		

Table A.1 Data collection form for first experiment

**Second Experiment:** in the first web page, input “Apple iPhone” in “search item” and input “1” in “Relevancy sensitive degree” fields, then click “Quality Based Ranking” button. In the result system presented to you, review each item, count the number of satisfactory goods items in first 10 and 20 results, and write the number in “B11” and “B21” of Table A.2. Then back to first web page again, input “1” in “Timeliness sensitive degree”, click “Quality Based Ranking” button and count the number of satisfactory goods items in the first 10 and 20 result, then record the number in “B12” and “B22”. Similarly, input “1” in “Reputation sensitive degree” or “Amount of Data sensitive degree”, then record number of satisfactory goods item.

Repeat the previous steps with “Cisco2610 Router” and “Cisco3620 Router”, record the numbers in TableA.2.

ResultSet	Goods Id	Relevancy	Timeliness	Reputation	Amount
-----------	----------	-----------	------------	------------	--------

used					of Data
First 10 results	Apple iPhone	B11	B12	B13	B14
	Cisco2610 Router				
	Cisco3620 Router				
First 20 results	Apple iPhone	B21	B22	B23	B24
	Cisco2610 Router				
	Cisco3620 Router				

Table A.2: Data collection form for second experiment

## Appendix B: Java Documentation

This appendix presents a summary of the Java documentation for all the classes of the system. Since the prototype of our system was fully implemented in Java, the Java documentation helps us to analyze the structure of our application. This Java documentation was generated using the Javadoc generation tool provided by Eclipse. In our project, there is one primary package, `esystem.agent`. The package is the top-level element in the documentation structure. The sub-level element, a Class Summary, lists all classes in the entire package. Every class is demonstrated by respective sub-level elements, i.e., Method summary that presents all methods provided by the class.

### Package `esystem.agent`

Class Summary	
<u>AgentUtil</u>	
<u>Item</u>	
<u>ItemServlet</u>	
<u>PersistenceItemManager</u>	
<u>PersistentManager</u>	
<u>QualityDimension</u>	
<u>Seller</u>	

`esystem.agent`

### Class `AgentUtil`

`java.lang.Object`

└ `esystem.agent.AgentUtil`

---

```
public class AgentUtil
extends java.lang.Object
```

---

## Constructor Summary

<code>AgentUtil()</code>
--------------------------

## Method Summary

<code>long</code>	<code>getEarliestUpdate</code> (java.util.List<Item> listItems)
<code>java.lang.String</code>	<code>getFileContent</code> (java.lang.String fName)
<code>long</code>	<code>getMaxLength</code> (java.util.List<Item> listItems) Use for "Amount of data"
<code>double</code>	<code>getMaxReputation</code> (java.util.List<Item> listItems) Use for reputation
<code>long</code>	<code>getMaxWordCount</code> (java.util.List<Item> listItems, java.lang.String words)
<code>void</code>	<code>getQualityIndex</code> (java.util.List<Item> listItems, java.lang.String searchstr, java.util.Map<java.lang.String, java.lang.String> qualityMap)
<code>static void</code>	<code>main</code> (java.lang.String[] args)

**Methods inherited from class java.lang.Object**  
 equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructor Detail

### AgentUtil

```
public AgentUtil()
```

## Method Detail

### getMaxWordCount

```
public long getMaxWordCount(java.util.List<Item> listItems,
                             java.lang.String words)
```

### getMaxLength

```
public long getMaxLength(java.util.List<Item> listItems)
```

Use for "Amount of data"

### getMaxReputation

```
public double getMaxReputation(java.util.List<Item> listItems)
```

Use for reputation

**esystem.agent**

## Class Item

java.lang.Object

└ **esystem.agent.Item**

public class **Item**

extends java.lang.Object

### Constructor Summary

**Item()**

### Method Summary

java.lang.String	<b><u>getCategory()</u></b>
java.lang.String	<b><u>getDescription()</u></b>
long	<b><u>getFeedbackNo()</u></b>
java.lang.String	<b><u>getId()</u></b>
java.lang.String	<b><u>getLocation()</u></b>
java.lang.String	<b><u>getPaymentmethod()</u></b>
double	<b><u>getPercentage()</u></b>

java.lang.String	<b><u>getPostdate()</u></b>
double	<b><u>getPrice()</u></b>
double	<b><u>getQualityIndex()</u></b>
java.lang.String	<b><u>getSellerId()</u></b>
java.lang.String	<b><u>getShippingcost()</u></b>
java.lang.String	<b><u>getShippingdest()</u></b>
java.lang.String	<b><u>getTitle()</u></b>

void	<b><u>setCategory</u></b> (java.lang.String category)
void	<b><u>setDescription</u></b> (java.lang.String description)
void	<b><u>setFeedbackNo</u></b> (long feedbackNo)
void	<b><u>setId</u></b> (java.lang.String id)
void	<b><u>setLocation</u></b> (java.lang.String location)
void	<b><u>setPaymentmethod</u></b> (java.lang.String paymentmethod)
void	<b><u>setPercentage</u></b> (double percentage)
void	<b><u>setPostdate</u></b> (java.lang.String postdate)
void	<b><u>setPrice</u></b> (double price)
void	<b><u>setQualityIndex</u></b> (double qualityIndex)
void	<b><u>setSellerId</u></b> (java.lang.String sellerId)
void	<b><u>setShippingcost</u></b> (java.lang.String shippingcost)
void	<b><u>setShippingdest</u></b> (java.lang.String shippingdest)
void	<b><u>setTitle</u></b> (java.lang.String title)

**Methods inherited from class java.lang.Object**

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**esystem.agent****Class ItemServlet**

```

java.lang.Object
├─ javax.servlet.GenericServlet
│   └─ javax.servlet.http.HttpServlet
│       └─ esystem.agent.ItemServlet

```

**All Implemented Interfaces:**

java.io.Serializable, javax.servlet.Servlet, javax.servlet.ServletConfig

```

public class ItemServlet
extends javax.servlet.http.HttpServlet

```

**See Also:**

[Serialized Form](#)

**Constructor Summary**

**[ItemServlet\(\)](#)**

**Method Summary**

void **[doGet](#)**(javax.servlet.http.HttpServletRequest req,  
javax.servlet.http.HttpServletResponse res)

void **[doPost](#)**(javax.servlet.http.HttpServletRequest req,  
javax.servlet.http.HttpServletResponse res)

void **[init](#)**()

**Methods inherited from class javax.servlet.http.HttpServlet**

service

**Methods inherited from class javax.servlet.GenericServlet**

destroy, getInitParameter, getInitParameterNames, getServletConfig, getServletContext,  
getServletInfo, getServletName, init, log, log

**Methods inherited from class java.lang.Object**

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

`esystem.agent`

## Class PersistenceItemManager

`java.lang.Object`

↳ `esystem.agent.PersistentManager`

↳ `esystem.agent.PersistenceItemManager`

public class `PersistenceItemManager`  
 extends `PersistentManager`

### Constructor Summary

`PersistenceItemManager()`

### Method Summary

<code>void</code>	<code><b>addItem()</b></code>
<code>void</code>	<code><b>clearItemView()</b></code>
<code>java.util.List</code>	<code><b>getAllCategories()</b></code> get the category lsit
<code>java.util.List</code>	<code><b>getAllItems()</b></code>
<code>Item</code>	<code><b>getItem()</b></code>
<code>Item</code>	<code><b>getItemById(long id)</b></code> get the item by Id
<code>java.util.List</code>	<code><b>getItems(java.lang.String name)</b></code> get the searching result of item collections
<code>java.util.List</code>	<code><b>getItemsByPrice(java.lang.String name)</b></code> get the searching result of item collections
<code>java.util.List</code>	<code><b>getItemsFromView()</b></code> get the searching result from item view
<code>Seller</code>	<code><b>getSellerById(long id)</b></code>
<code>static void</code>	<code><b>main(java.lang.String[] args)</b></code>
<code>void</code>	<code><b>putItemInView(Item item, double index)</b></code>

**Methods inherited from class `esystem.agent.PersistentManager`**

`checkObjectName`, `close`, `commitTransaction`, `convertString`, `execute`, `executeQuery`, `executeUpdate`, `generateTimestamp`, `getTableName`, `inTransaction`, `reset`, `rollbackTransaction`, `setBinaryStream`, `setDate`, `setDouble`, `setInt`, `setLong`, `setSql`, `setString`, `setTableName`, `setTimestamp`, `startTransaction`

**Methods inherited from class `java.lang.Object`**

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

`esystem.agent`

## Class `PersistentManager`

`java.lang.Object`

└ `esystem.agent.PersistentManager`

**Direct Known Subclasses:**

`PersistenceItemManager`

```
public class PersistentManager
    extends java.lang.Object
```

### Constructor Summary

`PersistentManager()`

### Method Summary

boolean	<code><b>checkObjectName</b>(java.lang.String name)</code>
void	<code><b>close</b>()</code>
void	<code><b>commitTransaction</b>()</code>
java.sql.Timestamp	<code><b>convertString</b>(java.lang.String sTime)</code>
boolean	<code><b>execute</b>()</code>
java.sql.ResultSet	<code><b>executeQuery</b>()</code>

`esystem.agent`

## Class `QualityDimension`

`java.lang.Object`

└ `esystem.agent.QualityDimension`

---

```
public class QualityDimension
extends java.lang.Object
```

---

### Constructor Summary

<code><b>QualityDimension</b>()</code>
--

### Method Summary

<code>double</code>	<code><b>getAmountofData</b>()</code>
---------------------	---------------------------------------

<code>double</code>	<code><b>getCompleteness</b>()</code>
---------------------	---------------------------------------

<code>double</code>	<code><b>getCurrency</b>()</code>
---------------------	-----------------------------------

<code>long</code>	<code><b>getDescLength</b>()</code>
-------------------	-------------------------------------

<code>long</code>	<code><b>getDescWordCount</b>()</code>
-------------------	--

<code>java.lang.String</code>	<code><b>getItemId</b>()</code>
-------------------------------	---------------------------------

<code>double</code>	<code><b>getRelevancy</b>()</code>
---------------------	------------------------------------

double	<b><u>getReputation</u></b> ()
void	<b><u>setAmountofData</u></b> (double amountofData)
void	<b><u>setCompleteness</u></b> (double completeness)
void	<b><u>setCurrency</u></b> (double currency)
void	<b><u>setDescLength</u></b> (long descLength)
void	<b><u>setDescWordCount</u></b> (long descWordCount)
void	<b><u>setItemId</u></b> (java.lang.String itemId)
void	<b><u>setRelevancy</u></b> (double relevancy)
void	<b><u>setReputation</u></b> (double reputation)

<b>Methods inherited from class java.lang.Object</b>	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	