Merchant Differentiation through Integrative Negotiation in Agent-mediated Electronic Commerce

Robert H. Guttman

Bachelor of Science in Engineering, Computer Engineering
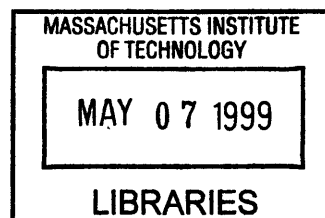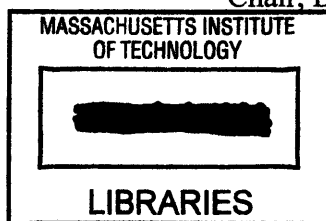University of Michigan at Ann Arbor
June 1992

Submitted to the Program in Media Arts and Sciences
School of Architecture and Planning
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Media Arts and Sciences
at the Massachusetts Institute of Technology
September 1998

Signature of Author...
Program in Media Arts and Sciences
August 24, 1998

Certified by...
Dr. Pattie Maes
Associate Professor of Media Arts and Sciences
Thesis Supervisor, MIT Media Laboratory

Accepted by...
Stephen A. Benton
Chair, Departmental Committee on Graduate Students
Program in Media Arts and Sciences

Merchant Differentiation through Integrative Negotiation in Agent-mediated Electronic Commerce

Robert H. Guttman

Abstract

This thesis proposes to fix online shopping by guiding it away from price comparisons and toward value comparisons. Though price comparisons may be adequate for simple products (e.g., books and music), they are inadequate for facilitating transactions of complex products (e.g., computers and automobiles). Consumers often must consider qualities other than price in their buying decisions and merchants usually prefer to differentiate themselves along alternative dimensions such as brand, customer service, delivery time, warranty, and other value-added services.

Tête-à-Tête is an agent-mediated comparison shopping system that allows consumers to consider dimensions other than price in their buying decisions for complex products. The system helps shoppers answer two questions: what to buy and who to buy from. Tête-à-Tête's integrative negotiation interaction model (based on bilateral argumentation), together with a decision support module (based on multi-attribute utility theory), create an improved online shopping environment for both consumers and merchants. Consumers gain increased satisfaction as their search costs for complex products are reduced and merchants potentially increase sales as a result of their enhanced differentiation in the marketplace.
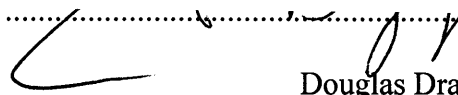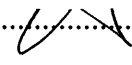
Merchant Differentiation through Integrative Negotiation in Agent-mediated Electronic Commerce

Robert H. Guttman

The following people served as readers for this thesis:

Thesis Reader ....

Dr. Erik Brynjolfsson
Douglas Drane Associate Professor of Management
MIT Sloan School of Management

Thesis Reader ....

Dr. Dave Cliff
Associate Professor of Computer Science and Engineering
MIT Artificial Intelligence Laboratory

Contents

list of figures

list of tables

list of equations

Acknowledgements

Progress arises from standing on the shoulders of those that came before us and from garnering inspiration from those around us. Although original work, this thesis could not have been accomplished without the wisdom and support of many people.

First and foremost, I owe a great deal of gratitude to Fernanda Viegas. Fernanda's fatal mistake was admitting that she was a professional graphics designer prior to joining the Media Lab in September '97. Recognizing early on that the success of Tête-à-Tête's shopping experience hinged on its shopping interface, I quickly began employing Fernanda's exceptional skills to no end. Invariably, this partnership turned into a friendship that I cherish deeply. I wish I could take full credit for the aesthetics of Tête-à-Tête's interface, but it is Fernanda who helped breathe visual beauty into my technical ideas. Any shortcoming of Tête-à-Tête's shopping interface, I can assure you, is not the fault of Fernanda, but is rather from my software team's valiant interpretations of her art.

Which brings me to my software team. There aren't words to express my appreciation for the work and dedication of these exceptional undergraduates. Tête-à-Tête has had a long design period with a burst of development in the last three months. An early member of the design team, Stanley Hu helped document Tête-à-Tête's software architecture that we had only partially sketched out on white boards numerous times before. An all-around exceptional designer, developer, and problem solver, Stanley helped transition our team from the design stage into the development stage thorough his insightful thoughts and comprehensive documentation. Another early contributor is Pedro Ripper. Among the admirable things about Pedro is his passion for understanding

and compassion to help others. Without pay, academic credit, or anything beyond the appreciation of his peers, Pedro helped spec out the technical options and associated risks for Tête-à-Tête's implementation. Pedro's good nature and drive created a healthy atmosphere for more rapid progress leading to Matthew Lee's early prototyping of Tête-à-Tête's shopping interface. Unfortunately, the end of the academic term also signaled the end to the efforts of these fine contributors.

Fortunately, Ruben Brown was able to stay onboard over the summer to continue with Tête-à-Tête's technical options and risk assessments. Blasting through the bugs and limitations of third-party products we were considering exploiting, Ruben helped with Tête-à-Tête's database connectivity and web server support in addition to his general help in architecting the system. Jane Huang joined the team next and tackled Tête-à-Tête's thorniest challenge of all – designing the ontologies for the notebook computers domain. Pouring over numerous web sites and even calling merchants to understand their policies, Jane almost single handedly architected all of the product and merchant features for notebook computers. In addition, Jane began development on Tête-à-Tête's SuperHelp system that may one day find its way into the service.

Perhaps the heroes of Tête-à-Tête's development, Jonathan Loflin and Iddo Gilon produced what can only be called "magic." With minimal Java experience and no prior user interface development experience, Jonathan became a Java applet developer wonder virtually overnight. Challenging design constraints were continuously being fed to Jonathan, and without fail, he delivered. But Jonathan contributed even more than UI development. Throughout Tête-à-Tête's development, Jonathan has been full of insights

and suggestions to help interpret and extend Fernanda's art and my ideas. I continue to be amazed at the outstanding contributions he made to the look and feel of Tête-à-Tête's shopping interface.

Then there's Iddo. I don't know how I found him, but the reason why I'm graduating now instead of three months from now is due largely in part to his exceptional technical skills and leadership abilities. I am confident that Iddo could have taken over the reins of the entire project if necessary. My right-hand assistant and inspiration to the rest of the team, Iddo has my unswerving admiration and appreciation for his diverse range of skills and flawless execution. From low-level communications and applet/back-end protocols to new Web display technologies and dynamic servlet-hosted Web pages, Iddo is one of the most well-rounded and finest programmers I have had the pleasure of working with. I can only hope that I will have the opportunity to continue working with Iddo in the future.

In the final hours of Tête-à-Tête's completion, Margaret Oh and Gaurav Tewari helped add the finishing touches. Margaret created several artistic buttons and helped design the layout of Tête-à-Tête's product offering details panel. Gaurav assisted in populating several merchant databases with each merchant's products and services in order to have meaningful screenshots for this thesis and provide for a good demo.

I have worked as a professional Software Engineer for many years on some outstanding technical teams, but I can safely say that the bright, young students above that helped me realize my vision of Tête-à-Tête will undoubtedly become of the finest in the industry.

Of course, there's more to the story of this thesis than Tête-à-Tête. I have been fortunate to have one of the most inspiring and helpful set of readers a thesis has ever seen. Dave Cliff's personal encouragement, friendship, and helpful comments on game theory were of particular value during the last days of writing. Perhaps the most challenging reader, Erik Brynjolfsson has made the last several days of writing a truly thought provoking experience. Helping me understand some of the deeper economic and game theoretic issues surrounding my negotiation protocol design has greatly strengthened this thesis, and more importantly, educated me further on the enormously exciting field of game theory. Furthermore, Erik's continued interest in my research throughout the last year has been a great source of encouragement for me. I am honored and grateful that Dave and Erik were both able to be official readers of my thesis.

Plagued by the anachronistic bureaucracies of academia, I unfortunately had to remove two other official readers of my thesis, Katia Sycara and Marty Tenenbaum. However, their contributions to this thesis cannot go unrecognized. Early in the process of determining my thesis, Katia helped guide me along the most viable technical paths. One of the most recognized researchers in the multi-agents systems community, Katia has been a source of inspiration and support for my thesis. In many ways, my thesis builds directly on her work in multi-agent argumentation and multi-attribute utility decision support. Katia's technical wisdom and kind words helped me push forward. I only wish I had the opportunity to interact with her and subsequently learn from her more.

The impetus of my thesis comes primarily from one person, Marty Tenenbaum, considered by many to be one of the founding fathers of electronic commerce. Taking

me under his wing and out to Silicon Valley and the home of CommerceNet, Marty

opened my eyes to the world of electronic commerce beyond the ivory towers. Working

with his phenomenal team at Veo Systems in the summer of '97, I formulated the seed

that has since bloomed into my thesis. This seed came directly from Marty.

Time and again, I would hear Marty elucidate the potential for the new economy in the

form of well-integrated business components and practices. His vision, called

component-based commerce and captured in a software design called eCo System,

dramatically reduces the transaction costs of all members of the system (among other

benefits). When challenged on his vision's incompatible incentives for suppliers, I found

one of Marty's main answers, although convincing, still wanting. Specifically, Marty

suggests that although the number one and two suppliers in an industry may not jump on

the eCo bandwagon for fear of losing market share, the number three, four, five, and so

forth will be eager to participate in order to leapfrog their competitors. (In fact, several

early electronic markets encompassing elements of Marty's vision played out exactly as

Marty predicted such as the BargainFinder example found in a later chapter.) Once the

system embodies sizable market power, Marty suggests, then the other suppliers will

have no choice but to follow suit since that is where the action is.

Although I agree with this assessment, this answer did not satisfy me. Following Marty's

vision to a logical conclusion, I envisioned our future marketplaces to be unhealthily

dominated by oligopolies providing far less selection and satisfaction to consumers. This

was not the future I wanted to live in. Many questions kept surfacing around the idea of

how to make this future environment – a truly inevitable commerce revolution – more

mutually beneficial for consumers and suppliers. Specifically, how could we simultaneously lower consumer search costs and create an incentive compatible marketplace for suppliers? Given that as Software Engineers, we have the opportunity to craft and direct our economic destiny on the Internet, I felt compelled to find a viable, healthy solution. Out of this self-challenge came this thesis and Tête-à-Tête.

If this thesis was all Marty had given me over this last year, that would have been more than enough. But Marty has given me so much more. Marty's constant encouragement and unwavering support of my research direction had given me the confidence to continue even when others suggested that my focus was too broad and overly ambitious. In addition, Marty has been a highly valued participant in several workshops that I helped host around the world. The insights and wisdom he has brought to each of these could not be overstated. Finally, on a deeply personal level, I am forever grateful of Marty's belief in me. If this thesis provides Marty and his team at Veo Systems with even the slightest bit of added support for their endeavors, then one of the main objectives of this thesis would be accomplished. I truly hope that I have the chance to work with Marty again in the future for his counsel and friendship.

Much more than an official reader, Pattie Maes has given me the guidance and support to realize a long-standing academic dream. Having read Stewart Brand's "The Media Lab: Inventing the Future at MIT" as an undergraduate at the University of Michigan, I had envisioned studying at what sounded like the most exciting technical playground in the world (which it is, I can now attest!). However, life decisions lead me to industry for several years where the academic itch only grew stronger. Not being satisfied with the

masters program I was half-way through while working full-time, I decided to finally apply to the Media Lab, and in particular, to Pattie's world-renowned Software Agents Group. I will be forever grateful to Pattie for recognizing my potential and advising me throughout these past two years. Pattie gave me just the right amount of guidance and encouragement to allow my academic ambitions to be fulfilled – from an intense introduction to demo-land with a major agent-mediated marketplace experiment I helped develop within the first month of joining the lab, to co-founding the lab's Agent-mediated Electronic Commerce (AmEC) Initiative and the joint papers we've written together over the years. Pattie's academic insights and warm friendship have made these past two years of the most memorable in my life. This opportunity at the Media Lab has truly changed who I am and my direction in life and I owe it all to Pattie.

There are many others at the Media Lab who have also enriched my experience here – surely too many to list. An early source of camaraderie, Daniel Dreilinger and I were both immediately thrust into developing the agent-mediated marketplace experiment (mentioned above) upon us entering the lab in September '96. From this project and numerous others that followed, one thing is perfectly clear: Daniel's technical prowess is unmatched. I continue to be amazed at his technical abilities and problem-solving skills. To describe Daniel as a source of technical inspiration for me would be a gross understatement. I can only hope to approach his technical abilities, regimented work ethic, and overall savvy style of being.

My daily interactions with the rest of Pattie's Software Agents Group have also been memorable. Of these colleagues, I've been influenced the most by Alexandros Moukas

(a.k.a. Moux). In many ways, Moux has been my "partner in crime" here at the lab. A fellow member of Pattie's Software Agents Group, Moux and I have teamed on several academic papers, have jointly participated in numerous conferences and workshops around the world, and have calmly "discussed" the most pressing philosophical issues of our time (despite that fact that I am forever a barbarian to his Greek education). A dear friend and respected colleague, Moux has made my stay at the Media Lab a constant source of intellectual and social enjoyment. I am greatly looking forward to our continued efforts together in the future.

In my research of viable retail electronic commerce, I inevitably must cross many academic disciplines. Of these, perhaps the most foreign (at the time) were business and marketing. From the germination of my thesis, I have viewed it from three main perspectives: multi-agent systems, human-computer interaction, and the business of retail electronic commerce. With Fernanda covering Tête-à-Tête's human-computer interaction component, and I covering the multi-agent systems component, I was still in search of a third academic to round out the triad. I could not have been more fortunate than to find Alexander F. Kleiner III to fill this role. And finding him was easy! Upon asking my friend's at MIT's Sloan School of Management who may be interested in exploring some deep issues in agent-mediated electronic commerce, all referred me to Alex. Soon after meeting him, it became evident why. An eclectic individual with interests ranging from non-Western History to computer science to business, Alex is one of the most honest, sincere, humorous, hard working, and intelligent people I know. Over a period of months, Alex and I hashed out various business implications and marketing aspects of viable, healthy retail electronic commerce. A constant contributor

to Tête-à-Tête's discussions, Alex provided a much needed business context and balance to our otherwise technical-heavy endeavor.

My admiration and respect for Alex and his family are without bound. Felicia, his wife, was immensely helpful in editing early drafts of this thesis. Their two-year old daughter, Haley, is too precious for words (and way too bright for her age!) in spite of the fact that she always pretends not to remember me each time she sees me (the squirt!). If it is at all in my power, I will stick with Alex on any venture he deems worthy. I couldn't have more faith or trust in an individual.

Last, and by far not the least, are the people that I love the most in this world and who have put me on this earth and given me the tools to explore it. My brothers, Evan and Marc, have each helped me to grow and taught me to constantly challenge my beliefs and the status quo. My appreciation and love for each of them transcends blood lines. Unique individuals, they inspire my pursuit for happiness, and in so doing, bring me a great deal of it themselves. My parents, Sam and Leslie, could not be more supportive or loving human beings. From their words of encouragement to knowing that they will always be there for me, my folks are my role models of the parent I hope someday to become. Words cannot express my deepest love, respect, and appreciation for all they are and all they've given me – unconditionally. If theses had dedications, there is no question to whom this one would be dedicated.

From everything I am and will become, thank you, Mom and Dad.

1 Introduction
Online Shopping

"We don't think that customers find price comparisons very useful. We would rather help people discover exactly what they're looking for online."
– Jeff Bezos, Founder and CEO, Amazon.com

Electronic commerce has come to be known as any activity of commerce over a digital medium. Although technologies such as Electronic Data Interchange (EDI), automated teller machines, and electronic credit card processing have been around for some time, the advent and popularity of the Internet has given new meaning to the term "electronic commerce." Today, electronic commerce over the Internet encompasses a broad range of issues including the economics of Internet pricing, marketing and advertising, payment mechanisms, security and privacy, trust and reputation, law and contracts, back-office management, supply chain management, and the buying and selling of goods and services.

This thesis concentrates on this latter category of electronic commerce, and, in particular, the buying and selling of goods and services in business-to-consumer (i.e., retail) markets, commonly known as "online shopping."

This thesis contemplates online shopping in the context of three general areas of research: multi-agent systems, human-computer interaction, and the business of retail electronic commerce, as depicted in Figure 1.
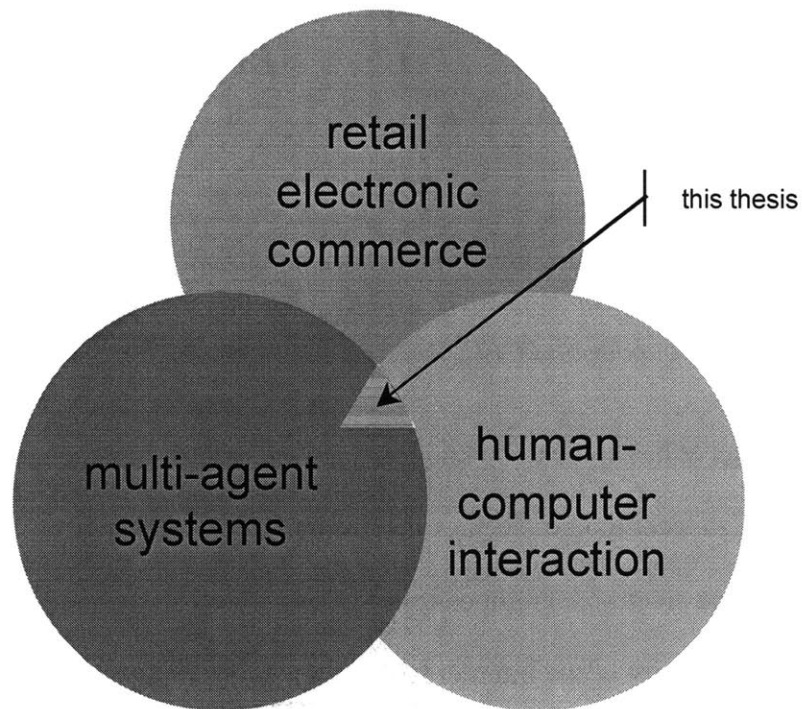
Figure 1: This thesis interesects three general areas of research

When building or evaluating an online shopping system that considers merchant health, consumer acceptance, and technical possibilities, aspects of all three of the research areas above must be brought to bear. However, of these three research areas, this thesis is positioned primarily from a multi-agent systems perspective.

Online Shopping: Opportunities and Challenges

Online shopping is both an opportunity and a threat to today's retail merchants. It is an opportunity because the Internet offers traditional merchants an additional channel to advertise and sell products to existing and new customers, thus potentially increasing

sales. Forrester Research estimates that online retail sales were at about $600 million USD in 1996, exceeded $2 billion in 1997, and will reach $17 billion by 2001 [1]. In addition, online marketplaces are more efficient than their physical-world counterparts which lowers transaction costs for both merchants and consumers.

Along with these opportunities, however, come great challenges. On the Internet, competitors' Web storefronts are only a "mouse-click away" – as opposed to a car-ride away in the physical world – making it relatively easy for consumers to compare merchants' offerings. At the same time, it is becoming increasingly difficult for consumers to find and consider product offerings as the number of online merchants and quantity of Web pages grow exponentially. In the face of this information glut, online merchants are struggling to differentiate themselves and attract shoppers to their sites [2]. As a potential remedy to this situation, software agent technologies have emerged to help consumers efficiently identify and compare product offerings from multiple merchants.

Agent-mediated Electronic Commerce

Software agents are programs to which one can delegate aspects of a task. They differ from "traditional" software in that they are personalized, continuously running and semi-autonomous. These qualities make agents useful for a wide variety of information and process management tasks [3], including those found in electronic commerce and, more specifically, in online shopping [4].

Several agent systems have been deployed over the Internet to help mediate transactions. These agent systems differ in the type and degree of assistance they provide consumers and merchants throughout the online shopping process.

Online Shopping Framework

Guttman, et al. present a framework for exploring the roles of agents as mediators in electronic commerce [5] as shown in Table 1. Based on traditional retail consumer buying behavior research, this framework provides a structure for discussing today's online shopping limitations and means for their mitigation. Tête-à-Tête (T@T), listed in the last column of Table 1, is the agent system introduced by this thesis.

	Persona Logic	Firefly	Bargain Finder	Jango	Kasbah	Auction Bot	T@T
1. need identification							
2. product brokering	√	√					√
3. merchant brokering			√	√			√
4. negotiation					√	√	√
5. payment & delivery							
6. service & evaluation							

Table 1: The online shopping framework with representative examples of agent mediation

The online shopping framework consists of six ordered stages:[1]

1. Need Identification
2. Product Brokering
3. Merchant Brokering
4. Negotiation
5. Payment & Delivery
6. Service & Evaluation

This thesis focuses on three of the six online shopping stages, namely Product Brokering, Merchant Brokering, and Negotiation. These three stages embody the core shopping activity and are buttressed by an identification of the need to make a purchase (Need Identification) and the purchase itself (Payment & Delivery). For a treatment of these other stages, see [5].

The Product Brokering stage comprises the retrieval of information to help determine *what to buy*. This encompasses the evaluation of product alternatives based on consumer-provided criteria. The result of this stage is the "consideration set" of products. The Merchant Brokering stage combines this "consideration set" with merchant-specific information to help determine *who to buy from*. This includes the evaluation of merchant alternatives based on consumer-provided criteria (e.g., price, warranty, availability, delivery time, and reputation). The Negotiation stage is about *how to settle on the terms of the transaction*. Negotiation varies in duration and complexity depending on the market. In traditional retail markets, price and other aspects of the transaction are often fixed, leaving no room for negotiation. In other markets (e.g., stock,

---

[1] Variations on this order exist. See [5] for a more complete discussion of this framework.

automobile, home electronics, and fine art), the negotiation of price or other aspects of the deal are integral to the shopping process.

As noted in [5], this online shopping framework represents an approximation and simplification of complex behaviors. Shopping stages often overlap, and migration from one to another is often non-linear and iterative. In fact, the recognition that consumers' buying behavior tends to be non-linear and iterative is a key insight into providing a comprehensive and superior online shopping experience than what is available today.

Tête-à-Tête

Tête-à-Tête is an agent-mediated online shopping system that facilitates the transaction of complex products. Assisted by a shopping agent, a consumer negotiates with several automated, merchant-owned sales agents concurrently to determine what to buy and who to buy from. Unlike other shopping systems which generally operate in only one stage of the online shopping process (see Table 1), Tête-à-Tête operates in the three core stages – namely the Product Brokering, Merchant Brokering, and Negotiation stages – to uniquely provide a cohesive and comprehensive online shopping experience that better facilitates transactions.

Tête-à-Tête relates to recommendation systems (e.g., PersonaLogic and Firefly) by offering an advanced decision support engine based on multi-attribute utility theory that meaningfully recommends complex products (not just simple products as in Firefly) and an interaction model that accommodates the non-linear and iterative process of shopping (unlike PersonaLogic). Tête-à-Tête relates to price comparison systems (e.g.,

BargainFinder and Jango) by extending the merchant brokering decision beyond just price to include merchants' policies and value-added services as dimensions for consumer consideration and merchant differentiation. Finally, Tête-à-Tête relates to online negotiation systems and auctions (e.g., Kasbah and AuctionBot) by proposing an alternative integrative negotiation protocol and interaction model based on bilateral argumentation that lies between the ad hoc haggling of Kasbah and the limited, unforgiving competitiveness of online auctions, with a suitable balance of formality, efficiency, and appropriateness for online retail shopping.

Tête-à-Tête embodies a general purpose infrastructure that supports multiple manufacturers and merchants within multiple product domains. Currently, Tête-à-Tête is encoded with a "notebook computer" ontology and associated product and merchant data, as shown in Tête-à-Tête's shopping interface depicted in Figure 2.
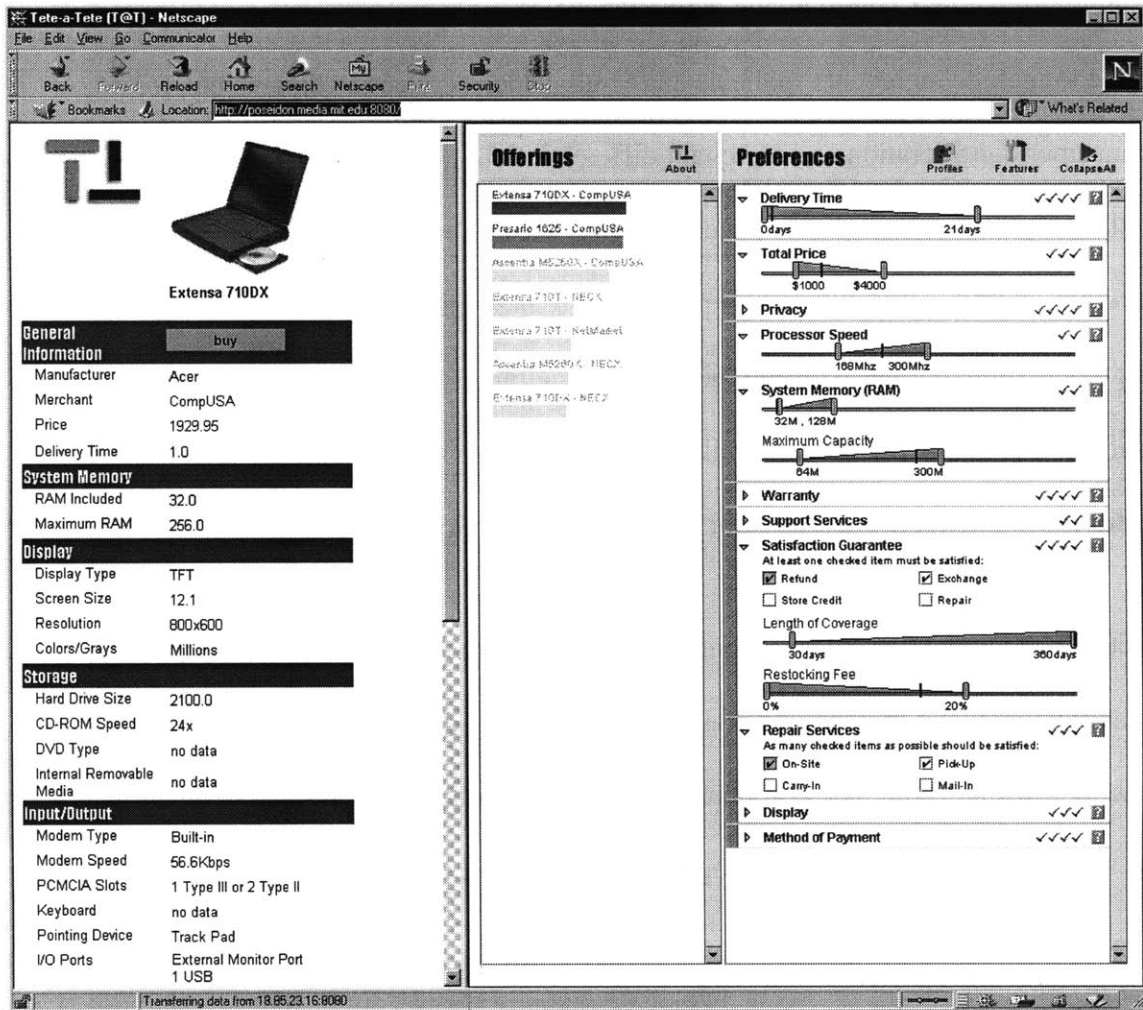
Figure 2: Tête-à-Tête's shopping interface

Merchant Differentiation

The problem of differentiating merchants online is complex. This thesis proposes a solution to this problem by focusing on how merchants are perceived in the marketplace within the context of a shopping experience. Invariably, a merchant's perceived differentiation rests with the shopper and will influence the shopper's merchant brokering decision (see Table 1). As such, this thesis concentrates on merchant differentiation predominantly from a shopper's perspective.

Furthermore, while each Tête-à-Tête shopping agent interacts with its owner to help the

shopper through the three core stages of the shopping process, each merchant's sales

agent is completely autonomous once it has captured its owner's utilities for profit

margin and has been granted access to its merchant's database of products and services.

Lacking a human interactive component, Tête-à-Tête's sales agents do not currently have

graphical user interfaces. Instead, sales agents are "programmed" through their

respective relational databases and internal software. An enhancement to Tête-à-Tête

would entail providing a more user-friendly interface for sales agents to help with their

administration.

Guide to This Document

Chapter 2, "Related Work: Online Shopping is Broken," presents related work

representing the state-of-the-art in online shopping technologies along with their

limitations in the context of the online shopping framework. Chapter 3, "Tête-à-Tête:

Fixing Online Shopping," introduces Tête-à-Tête as it pertains to the online shopping

framework and explains how Tête-à-Tête overcomes the limitations discussed in Chapter

2. Tête-à-Tête's integrative negotiation technologies and interaction model are detailed

in Chapter 4. Chapters 5 and 6 detail Tête-à-Tête's shopping agent and sales agent

decision support modules respectively, followed by a discussion of Tête-à-Tête's

implementation in Chapter 7. Chapter 8 concludes this thesis and contains evaluation

suggestions and future research directions.

2 *Related Work*
Online Shopping is Broken

Several agent systems have been deployed over the Internet to help mediate transactions. These systems differ in the type and degree of assistance they provide consumers and merchants throughout the online shopping process. As shown in Table 1, most agent systems play within only one shopping stage which is problematic for facilitating transactions.[2] Ideally, agent systems playing in complementary stages of the online shopping framework could be mixed and matched to provide a comprehensive and coherent online shopping experience. Unfortunately, today's agent-mediated electronic commerce systems are not designed to interoperate in this way and linking these disparate systems together would require a good deal of work.

Nevertheless, it is worth exploring these pioneering agent systems in detail to understand their limitations and to understand how these limitations can be improved upon. To structure this discussion, the agent systems are presented within the dominant shopping stage in which they operate.

---

[2] A few of the agent systems listed in Table 1 provide limited support for a second shopping stage, but not sufficient support for acknowledgement within that stage.

Product Brokering: What to Buy?

The Product Brokering stage of the online shopping framework is where consumers determine what to buy. This occurs after a need has been identified and is achieved through a critical evaluation of retrieved product information. Two representative agent systems that perform product brokering are PersonaLogic and Firefly (see Table 1).

PersonaLogic

PersonaLogic [6] helps consumers identify which products best meet their needs by guiding them through a large product feature space in the format of a "deep interview" as shown in Figure 3.

How important to you are the following characteristics?

Bicycle characteristics will be most important to people who want to ride aggressively. If you don't plan to ride fast or tackle trails, we suggest that you skip this page and continue to the next.

Frame durability is the bike frame's ability to resist denting, cracking, and bending. The type of material used in the frame determines its durability.

No Opinion | Somewhat | Very | Extremely

The weight of the bike includes the components and wheels, not just the frame.

No Opinion | Somewhat | Very | Extremely

(several more Web pages of "deep interview")

Figure 3: Shopping with PersonaLogic is by a "deep interview"

PersonaLogic filters out unwanted products within a given domain after a shopper specifies constraints on product features. A constraint satisfaction engine then returns a

list of products that satisfy all of the shopper's hard constraints ordered by how well they

satisfy the shopper's soft constraints, as depicted in Figure 4.

Figure 4: PersonaLogic orders product results by how well they satify the shopper's preferences

Although aesthetically pleasing and moderately helpful for identifying what to buy,

PersonaLogic does not assist shoppers in identifying who to buy from.[3] This obstructs

sales and reduces customer satisfaction. Also, it is unclear how the system's soft

constraints (e.g., the "How important..." answers of Figure 3) are evaluated and affect

---

[3] PersonaLogic recently added price comparisons to its deep interviews for computer hardware. See the discussions on BargainFinder and Jango later in this chapter for the problems with this practice.

the "scores" of each product as shown in Figure 4. The lack of visual cues to help the shopper understand how this "fuzzy" information is being applied can hurt the shopper's confidence in the system as a recommendation tool. Furthermore, PersonaLogic's "deep interview" interaction model engenders two significant problems: (1) it forces shoppers down a long path that is not easily undone and (2) it divorces the preference solicitation from the display of results, making it unclear how a shopper's preferences affect which products best meet the shopper's needs.

In a typical scenario, an initial result list of products generated by PersonaLogic (see Figure 4), while possibly useful, is rarely sufficient for making a buying decision. It is difficult to accurately express preferences for complex products, especially the first time a shopper is confronted with product features not considered before. In most cases, the feedback of results elicits a need to refine previously entered search criteria. In PersonaLogic, the iterative process of refining the shopper's preferences for features in order to change the results (i.e., the consideration set) is time-consuming and cumbersome due to its "deep interview" interaction model. Each time a shopper wishes to change preferences expressed earlier in the interview, the shopper must find and jump back to the Web page (or pages) that contain the preferences, change them, and then jump back to the results page to view the new list of products.

In addition to being time-consuming and cumbersome, the "deep interview" interaction model divorces the preference solicitation from the resulting list of products, making it difficult for shoppers to understand how their preferences influence the results. In the bicycle domain, for example, it is not clear how preferences for frame durability and a

bicycle's weight affect the overall value (or "score") of each available product. This gap in understanding can be confusing which lowers customer satisfaction and hinders sales.

Firefly

Like PersonaLogic, Firefly's services [7] help consumers find products to purchase. However, instead of filtering products based on their features, Firefly recommends products via a "word of mouth" recommendation mechanism called automated collaborative filtering (ACF) [8]. ACF first compares a shopper's product ratings with those of other shoppers. After identifying the shopper's "nearest neighbors" (i.e., users with similar tastes), ACF recommends products that they rated highly but which the shopper has not yet rated, potentially resulting in serendipitous finds. Up until being purchased by Microsoft in April 1998, Firefly was used to recommend simple products such as music, books, and movies, as depicted in Figure 5.
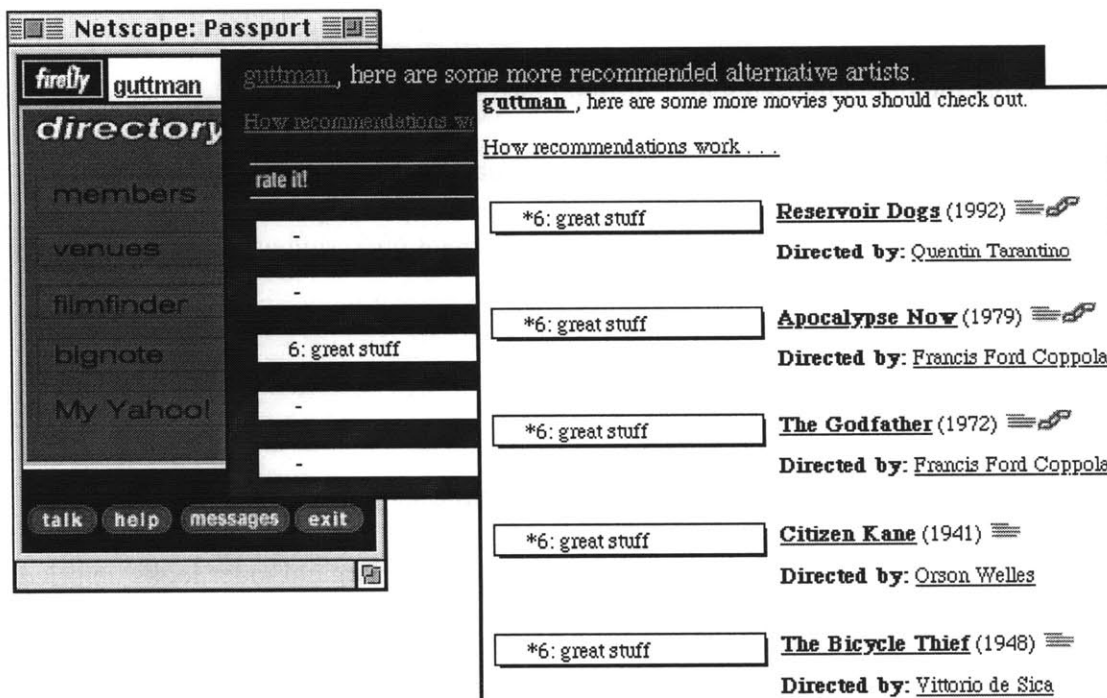
Figure 5: Firefly recommends simple products based on opinions of like-minded people

As with PersonaLogic, Firefly suffers from only assisting consumers in identifying what to buy, not who to buy from. Again, this break in the online shopping process hinders sales and lowers customer satisfaction. Moreover, although useful for recommending simple products, ACF is not useful for recommending complex products such as computers, insurance policies, and bicycles (as in the previous PersonaLogic example). This is because consumers often must ensure that complex products meet specific feature constraints and preferences – e.g., weight, size, warranty, price, etc. ACF does not consider any such features when making product recommendations.

Merchant Brokering: Who to Buy From?

Whereas the Product Brokering stage compares product alternatives, the Merchant Brokering stage compares merchant alternatives. Merchant brokering is achieved through a shopper's critical evaluation of merchants' product offerings. The importance of the merchant brokering decision, as with the product brokering decision, depends on many factors including the shopper's current goals, knowledge, preferences, constraints, influences, moods, and attitudes as well as the nature of the product to be purchased. Two representative agent systems that perform merchant brokering are BargainFinder and Jango (see Table 1).

BargainFinder

Andersen Consulting's BargainFinder is the first online shopping agent to perform price comparisons [9]. Given a specific music item description, BargainFinder looks up price information from approximately nine different merchant Web sites using the same

requests as from a Web browser. BargainFinder then displays the results in a list ready to be compared by the shopper, as shown in Figure 6.

Figure 6: BargainFinder is the original price comparison shopping agent

Although a limited proof-of-concept system, BargainFinder offers valuable insights into the issues involved in price comparisons in online shopping. For example, CDnow blocks all of BargainFinder's price requests. CDnow is able to do this because BargainFinder is a centralized service hosted by Andersen Consulting and its price requests can easily be identified and blocked. The *reason* CDnow blocks requests, however, is, presumably, because CDnow is the leader in online music sales and would rather not compete on price alone.[4] Value-added services that CDnow invested in to help differentiate its Web site – e.g., music reviews, extensive discographies, sound clips, etc. – are bypassed by BargainFinder and therefore not likely considered in a consumer's merchant brokering decision.

---

[4] This information was distilled from an email from Bruce Krulwich, the primary author of BargainFinder.

Jango

Jango [10] can be viewed as an advanced BargainFinder. The original Jango version "solved" the merchant blocking problem by having its price requests originate from each consumer's Web browser rather than from a central site as used in BargainFinder. This way, price requests to merchants from a Jango-augmented Web browser appeared as price requests from "real" customers. This made it nearly impossible for merchants to detect and block Jango agents allowing consumers to compare products from multiple online catalogs whether merchants wanted this or not.

Jango and other virtual database technologies (such as offered by Junglee [11]) use specialized "wrappers" to access each merchant's catalog since each merchant's catalog is published differently in HTML (hypertext markup language). Although learning techniques exist to semi-automatically compose these "wrappers" [12], much of this work is still largely done by hand and is extremely tedious. In the near future, XML (extensible markup language) [13] may make price comparison agents more robust, extensible, and easier to implement.

After being purchased by Excite in June 1997, Jango's functionality diminished upon becoming a centralized service akin to BargainFinder.[5] A shopper interacts with Excite's Jango by entering a price constraint and a few product feature preferences into a single Web page.[6] Figure 7 shows a shopper's preferences for a notebook computer.

---

[5] The original Jango, when owned by NetBot, required a separate Internet download that sported a semi-dynamic interaction with the user. Excite prioritized ubiquity over functionality, however, and converted Jango into an HTML-based service that doesn't require a separate download.

[6] Jango's preference solicitation, relative to PersonaLogic's, for example, is too limited to be considered supportive for making product brokering decisions.

Figure 7: Jango captures shoppers' preferences for price and a limited set of product features

The shopper then submits these to Jango's search engine which responds with a list of notebook computer product offerings from multiple merchants based on price, as shown in Figure 8.

Figure 8: Jango returns a list of product offerings differentiated by price

Although perhaps useful for buying simple products such as books and music, Jango's price comparisons are not adequate for facilitating transactions of complex products such as skis, bicycles, tents, automobiles, insurance policies, camcorders, and notebook computers (contrary to the example above). Consumers often consider dimensions other than price when making merchant brokering decisions for these types of products because they tend to be more expensive, longer lasting, more personal, and generally more important than simple products.

For example, when buying a notebook computer (see Figure 8), other critical factors in the shopper's merchant brokering decision may include extended warranties, forgiving return policies, extensive service contracts, special gift services, high product availability, superior customer service and support, diverse payment, loan and leasing options, fast delivery times with low costs, promotions and coupons, cross-manufacturer product configurations, community involvement and social awareness, privacy policies, and more.

These factors constitute a merchant's policies and value-added services that may help differentiate the merchant from its competitors (in addition to price). The value-added services listed above are more important than on-site services such as chat rooms, media clips, and localized recommendation services (such as those found at CDnow), because they are integral to the buying decision. They add value to the base manufactured product during and after its purchase and delivery. Without the ability for consumers to consider and express preferences for these value-added services, online marketplaces, such as those created by Jango, appear more homogenous than they actually are. None of

today's merchant brokering services allow consumers to consider these other dimensions in their buying decisions and this is a disservice to consumers and merchants alike.

Negotiation: How to Settle on the Terms of the Transaction?

The Negotiation stage is where the price or other terms of the transaction are settled. Perhaps less intuitive than product and merchant brokering, negotiation is a form of decision-making where two or more parties come together to jointly search a space of possible solutions with the goal of reaching a consensus [14, 15]. Economics and game theory describe such interactions in terms of protocols and strategies.

The *protocols* (or *opportunity sets*) of a negotiation comprise the rules (i.e., the valid actions) of the game. An example of a simple negotiation protocol is the Dutch auction where the only legal bidding action is an open outcry of "mine!" as an auctioneer decrements the price of the good. For a given protocol, a rational bidder uses a *strategy* (i.e., a plan of action) to maximize the bidder's utility. Decision analysis tools can help identify optimal strategies given a bidder's preferences and knowledge (e.g., motivation, valuation, risk, and information asymmetry) [14].

Negotiation protocols partially define an interaction model[7] and vary in their formality, efficiency, and appropriateness for a given market. When buying a car, for example, it is customary in many countries to haggle over price and other factors, such as warranty coverage, delivery time, and servicing as well as the bundling of other products such as floor mats and new tires. Although possibly effective, entertaining, and culturally

---

[7] An "interaction model" embodies the general manner and style of human-computer interaction.

appropriate, ad hoc negotiation protocols are usually less efficient than other protocols such as second-price, sealed-bid (a.k.a. Vickrey) auction protocols [16]. Designers of online marketplaces have the opportunity to weigh these tradeoffs to find the optimal balance of formality, efficiency, and appropriateness for buyers and sellers in each market. Two representative agent systems that employ differing negotiation protocols are Kasbah and AuctionBot (see Table 1).

Kasbah

One of the first online agent systems for negotiating consumer products, Kasbah [17], is still in use today. Kasbah assists MIT students in their transaction of books and music with one another. An MIT student wanting to buy or sell a good creates an agent, gives it some strategic direction, and sends it off into a centralized agent marketplace. Kasbah agents pro-actively seek out potential buyers or sellers and negotiate with them on their owner's behalf. Each agent's goal is to complete an acceptable deal, subject to a set of user-specified constraints such as a desired price, a highest (or lowest) acceptable price, and a date by which to complete the transaction, as shown in Figure 9.

Figure 9: Kasbah is one of the first online agent systems for negotiating consumer products

Negotiation in Kasbah is straightforward. After buying agents and selling agents are matched, the only valid action in the negotiation protocol is for buying agents to offer a bid to selling agents with no restrictions on time or price. Selling agents respond with either a binding "yes" or "no." Given this protocol, Kasbah provides buyers with one of three negotiation "strategies": anxious, cool-headed, and frugal. These strategies respectively correspond to a linear, quadratic, or exponential function for increasing a buying agent's bid for a product over time (or decreasing a bid over time in the case of a selling agent).

Kasbah does not concern itself with optimal efficiency, strategies, or convergence properties. Rather, Kasbah provides more descriptive heuristics that model typical haggling behavior found in classified ad markets. The simplicity of Kasbah's negotiation protocols and strategies, though less efficient than others, makes it intuitive for users to understand their agents' behaviors in the Kasbah marketplace. This instilled trust in the system which proved crucial to the success of an earlier experiment [18]. As such, Kasbah's simple protocol and strategies seem appropriate for the markets it serves, namely used books and music on a single college campus. However, for other markets, alternative negotiation protocols and strategies may be more appropriate.

AuctionBot

AuctionBot [19, 20] is a general purpose Internet auction server. Auctions are price discovery mechanisms that relieve sellers from needing to determine the value of a good a priori. Rather, this burden is pushed into the marketplace. A resulting benefit of this is that limited resources are allocated fairly – i.e., to those buyers who value them most.

AuctionBot users create new auctions to sell products by choosing from a selection of auction types and then specifying its parameters, such as clearing times, method for resolving bidding ties, the number of sellers permitted, etc., as shown in Figure 10. Buyers and sellers can then bid according to the multi-lateral negotiation protocols of the created auction. In a typical scenario, a seller would bid a reservation price after creating the auction and let AuctionBot manage and enforce buyer bidding according to the auction protocol and parameters. A public programmatic API allows AuctionBot buyers and sellers to construct their own bidding agents to automate their bidding strategies.

Figure 10: AuctionBot offers many auction protocol permutations

AuctionBot is useful for building prescriptive theories of coordination among heterogeneous agents with (partially) predictable system-wide dynamics. Economics and game theory research have much to say about the equilibrium states, efficiencies, and optimal strategies of various auction protocols [21]. Auctions are actively used in numerous business transactions such as the bidding of contracts, the redistribution of property, the allocation of electricity, and the global management of public stocks.

Perhaps not surprisingly, auctions have become pervasive on the Internet in diverse markets, including classified ad, B-goods (i.e., refurbished or older models), and retail.

In retail markets, auctions are used by retailers as an additional means to sell their products. In some cases, the retailer acts as both the seller and auctioneer (e.g., Internet Shopping Network's FirstAuction [22] and Cendant's NetMarket [23]), and in other cases, auctions are run by an intermediary (e.g., OnSale [24]). Although only used for entertainment purposes in some cases, a critical look at applying auction protocols to the retail of complex products uncovers many problems and limitations. These are summarized below.[8]

> *"Availability is more important than price."*
> *– Gerry Heller, CEO of FastParts, an online auction for semiconductors*

1. ONLINE AUCTIONS SUFFER FROM THE SAME PROBLEMS AS PRICE COMPARISONS

At the heart of each auction protocol lies a focus on price. Online auctions, therefore, suffer the same core problems as do price comparisons. For complex products, consumers often must consider qualities other than price in their buying decisions and merchants usually prefer to differentiate themselves along alternative dimensions such as brand, customer service, delivery time, warranty, and other value-added services. In fact, consumers are not inherently price-sensitive during any given shopping session [25]. As with price comparisons, auctions have the ability to make consumers more price-sensitive than they may otherwise have been and thus distract them from the added value merchants may offer.

---

[8] For a greater treatment of these problems, see [4].

2. SELF-HOSTED AUCTIONS JEOPARDIZE CONSUMER TRUST

A key characteristic of auctions is that buyers and sellers come together in an unbiased marketplace – e.g., NASDAQ. Having an unbiased marketplace ensures fairness before, during, and after the negotiation process. When a retailer also acts as the auctioneer, the retailer controls the marketplace and has exclusive access to who is bidding, how much they are bidding, as well as the dissemination of this knowledge to the bidders. This control makes it easy for the retailer to unfairly manipulate the outcome of the auctions – either through the withholding of information, the propagation of misinformation, etc. Such exclusive control by the retailer jeopardizes trust and fairness which, if recognized by consumers, impedes sales.

3. OPTIMAL BIDDING STRATEGIES ARE NON-INTUITIVE

The protocols for the two most prevalent types of online retail auctions, first-price open-cry English and Yankee [16], are simple to understand and to bid on, but determining the optimal bidding strategy for either is non-trivial. Factors to be considered include information asymmetry, risk aversion, motivation, valuation, and probabilities of how other bidders will behave. This requisite knowledge and speculations are beyond the abilities of the average online shopper. As in all auctions, bidding sub-optimally can be financially adverse. If the tedium of negotiating an optimal strategy is abstracted away from the shopper (e.g., by an intelligent bidding agent), then there remains the critical issues of which shopping interaction model to employ and the underlying technologies to support it.

4. AUCTIONED PRODUCTS ARE NON-RETURNABLE

Two common auction rules that compound the sub-optimal bidding problem are: (1) bids

are non-retractable, and (2) products are non-returnable. This policy will invariably lead

to some unsatisfied customers. If, however, merchants choose to honor full returns, then

all of the risk in negotiating the price of products is removed. This questions the value of

using auctions to sell retail products if not simply for entertainment purposes.

5. ONLINE AUCTIONS ARE TIME-CONSUMING AND INEFFICIENT

Most online retail auctions endure substantial delays for sometimes as much as several

days between the start of negotiations and the actual purchase of the product.[9] This

ensures a critical mass of bidders or deals with communication latencies over the Internet.

Since bids are non-retractable, the consumer cannot consider other product offerings

during this delay. In cases where the auction closes and a bidder does not win the

auction, the bidder must wait until the product is auctioned again to restart the negotiation

process. These delays do not cater to impatient or time-constrained consumers.

6. PRICE DISCOVERY IS LESS REALIZABLE FOR PRODUCTION GOODS

Although auctions can relieve merchants of the burden of establishing prices for limited

resources (e.g., fine art and stocks), this benefit is less realizable for production goods as

in retail markets. Unlike fine art, for example, it is relatively easy to determine the

marginal costs of production goods. If auctioning these goods, however, it is non-trivial

for the merchant to determine the optimal size of the auctioned lots and the frequency of

their auction [26]. Such a determination requires an understanding of the demand for the

---

[9] Although sealed-bid auction protocols (e.g., as used by Priceline [39]) and others are more efficient than
the typical first-price open-cry auction protocols, these are rarely used for online retail auctions.

good since it directly affects inventory management and indirectly affects production schedule. Therefore, retailers are still burdened with determining the value of their goods a priori. This nullifies the primary benefit of using an auction as a price discovery mechanism.

7. ONLINE AUCTIONS ARE SUSCEPTIBLE TO COLLUSION

Exclusive control of the marketplace is not required to manipulate online auctions. Sellers may introduce "shills" into the auction who unfairly bid up the price of the auctioned good on behalf of the seller. This is done to entice buyers to bid higher. Likewise, buyers may unfairly form "coalitions" (or "rings") so as to not outbid one another. After winning a discriminatory (i.e., multi-good) auction, the coalition can distribute the spoils amongst themselves (e.g., evenly or by holding a second private auction), oftentimes winning more and paying less than if they hadn't formed a coalition. Both shills and coalitions are illegal yet hard to detect, especially online where bidders are not co-located. In fact, technologies from multi-agent systems research have been developed that can efficiently form coalitions among previously unknown parties [27], thus increasing the threat to merchants selling through online auctions.

8. ONLINE AUCTIONS ESTABLISH A COMPETITIVE RELATIONSHIP BETWEEN A MERCHANT AND ITS CUSTOMERS

Retail merchants generally prefer to have long-term relationships with their customers in the hopes that they will make repeat purchases and refer new customers to them. This helps secure long-term profitability. In fact, prioritizing long-term profitability over short-term profits is a dominant theme on the Internet today as retailers operate at great losses (e.g., Amazon.com [28]) in order to acquire consumer mindshare. Online auctions,

however, focus on short-term profits rather than long-term profitability since each auction is a competition over price. Furthermore, merchants who interact with their customers through online auctions are at risk for deterring them due to the antagonistic nature of the auction experience which has been known to disappoint, confuse, frustrate, anger, and even victimize their participants for all of the reasons described earlier.

Summary

This chapter explored the state-of-the art in online shopping within the online shopping framework. Common limitations of the systems described above include a focus on price rather than value, an incomplete shopping experience, and an ineffective interaction model for helping shoppers identify either what to buy or who to buy from. The next chapter introduces an agent-mediated online shopping system, Tête-à-Tête, which embodies an alternative and comprehensive interaction model with supporting technologies to overcome these limitations.

3   Tête-à-Tête
Fixing Online Shopping

The goals of consumers and merchants in online shopping are interdependent. On the one hand, consumers want reduced search costs, low prices, high value, wide selections, better education, more entertainment, personalized service, and a comprehensive shopping experience. On the other hand, merchants want increased sales, reduced transaction costs, a positive reputation, long-term profitability, and differentiation in the marketplace. These goals, while contentious, are not mutually exclusive. For example, reducing consumer search costs for complex products does not preclude merchants from being able to differentiate their product offerings in the marketplace. With this in mind, this thesis introduces a cooperative and comprehensive online shopping system called Tête-à-Tête that concurrently maximizes the interdependent goals of consumers and merchants.

At the core of Tête-à-Tête is an integrative negotiation interaction model that differs significantly from the distributive negotiation interaction model exemplified by today's online auctions. This chapter provides an overview of Tête-à-Tête and its encompassing negotiation protocol and decision support technologies.

Distributive vs. Integrative Negotiation

Business negotiation literature defines two types of negotiation: distributive and integrative [29]. *Distributive negotiation* is the decision-making process of resolving a conflict involving two or more parties over a single, mutually exclusive goal. Economics literature describes this more specifically as the effects on market price of a limited resource given its supply and demand among self-interested parties [21]. Game theory literature describes this situation as a zero-sum game where, as the value along a single dimension shifts in either direction, one side is better off and the other is worse off [14].

Fundamentally, all auctions are distributive negotiation protocols since they formulate competitions along a single dimension – price. As price shifts in either direction, it either favors the buyer or the seller, but never both. In the case of online retail auctions, when the merchant benefits, the customer suffers and vice versa.

Integrative negotiation, on the other hand, is the decision-making process of resolving a conflict that involves two or more parties with multiple interdependent, but non-mutually exclusive, goals [29, 30]. In economics, multi-attribute utility theory studies how to analyze the multi-objective decisions found in integrative negotiations [31]. In game theory, integrative negotiation is a non-zero-sum game that can simultaneously benefit all parties as the values along multiple dimensions shift in different directions [14].

Comprehensive Online Shopping

The interdependent goals of online shopping for complex products characterize an integrative negotiation between consumers and merchants. As discussed in Chapter 2,

consumers often must consider qualities other than price in their buying decisions and merchants usually prefer to differentiate themselves along multiple dimensions. Unlike price comparisons and online auctions, integrative negotiation protocols afford consumers and merchants the opportunity to cooperatively search the space of product offerings across their full range of value with the purpose of maximizing each party's interdependent goals.

More specifically, Tête-à-Tête's integrative negotiation protocol embodies a framework to seamlessly integrate the product brokering, merchant brokering, and negotiation stages of the online shopping process (see Table 1), thus providing a comprehensive online shopping experience. Unlike the interaction model of online retail auctions which require a completion of the product and merchant brokering stages, Tête-à-Tête's interaction model allows consumers to evaluate the full value of multiple product offerings (including price) prior to committing to any specific product or merchant.

Bilateral Argumentation

Tête-à-Tête employs a specific integrative negotiation protocol that is based on bilateral argumentation [32].[10] As detailed in Chapter 4, the performatives of argumentation include proposals, critiques, and counter-proposals which form the core of Tête-à-Tête's negotiation protocol.[11] The exchange of these performatives is mediated by consumer-owned shopping agents and merchant-owned sales agents, as depicted in Figure 11.

---

[10] Unfortunately, the negative connotations of the term "argumentation" misrepresents the cooperative nature of this protocol relative to auction protocols.

[11] The term "performative" comes from speech act theory and denotes an utterance that itself constitutes an act such as requesting or proposing [33]. More recent work in inter-agent speech acts include KQML (Knowledge Query and Manipulation Language) [34].
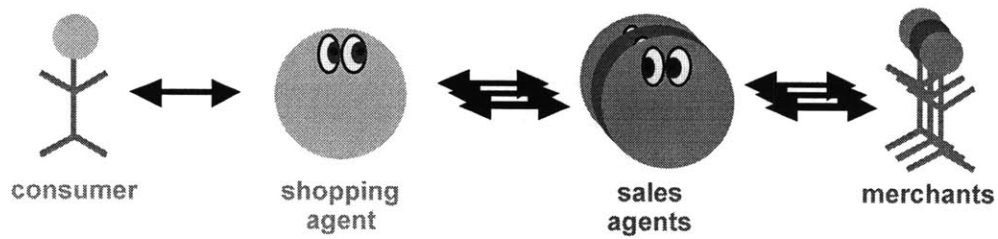
Figure 11: Tête-à-Tête employs consumer-owned shopping agents and merchant-owned sales agents

Each Tête-à-Tête consumer is provided a single shopping agent. For a given domain, a shopping agent captures its owner's preferences for product offerings which comprise *product features* and *merchant features*. For example, in the notebook computers domain, a shopping agent will capture preferences for product features such as system memory, modem speed, and display size, as well as merchant features such as delivery time, warranty length, and total price. These preferences are broadcast as an initial *criteria* performative to every sales agent whose owner sells notebook computers.

Each Tête-à-Tête merchant is provided a single sales agent.[12] Upon receiving a criteria performative from a shopping agent, the sales agent uses this criteria to evaluate its catalog of products and value-added services as well as its owner's preference for profit. This evaluation helps the sales agent customize the most appropriate product offerings for the shopper which it then sends as a response in the form of *proposal* performatives to the shopping agent.

The shopping agent, in turn, evaluates each of the product offerings it receives from all sales agents and displays them to the shopper ordered by how well they match the

---

[12] In practice, a different sales agent manages each shopping agent negotiation session as discussed in Chapter 7. However, for purposes of this discussion, it is convenient and logically equivalent to view each merchant as having only one sales agent since each sales agent maintains the same merchant preferences.

shopper's expressed preferences. Negotiation proceeds as a series of preference changes the consumer makes based on the continuous feedback from sales agents. These preference updates are sent as *critique* performatives to each sales agent. Each sales agent then responds with an updated list of product offerings packaged as *proposal*, *counter-proposal*, and *withdraw-proposal* performatives corresponding respectively to new, updated, or withdrawn product offerings. These updated product offerings are re-ordered and presented to the shopper for consideration. Negotiations continue until the shopper withdraws from negotiations or accepts a product offering (which leads to the payment and delivery stage, as shown in Table 1).

Product and Merchant Brokering Decision Support

Recommendation systems such as Firefly's automated collaborative filtering (ACF) are useful for recommending simple products such as books and music. However, to meaningfully recommend complex products such as computers and automobiles, alternative technologies are required (as discussed in Chapter 2). These technologies must also complement the system's underlying protocols and interaction model.[13]

Tête-à-Tête includes a recommendation engine for complex products based on multi-attribute utility theory (under certainty) and a complementary bootstrapping mechanism called "stereotyping" (discussed in Chapter 5). Each Tête-à-Tête shopping agent contains a decision support module that assists shoppers in negotiating with sales agents and in making optimal product and merchant brokering decisions given the shopper's expressed preferences. Preferences are expressed for product and merchant features. Each feature

---

[13] An "interaction model" embodies the general manner and style of human-computer interaction.

has a name (e.g., "Display") and is comprised of one or more *attributes* (e.g., "Display Size" and "Display Resolution"). Each expressed preference for an attribute represents a *criterion* for evaluating a product offering. When a product offering is proposed by a sales agent, the shopping agent compares the *offer value* of each of its attributes with its respective criterion.[14]

For example, a shopper may have expressed a preference for a delivery time between 1 and 14 days and favors shorter delivery times. If a product offering includes overnight (i.e., 1 day) delivery, then this would satisfy the criterion perfectly and the "Delivery Time" attribute would be assigned a perfect score. If, however, the product offering implicates a delivery time of 14 days, then this would only minimally satisfy the criterion and the attribute would be assigned a minimal score.

After assessing each attribute by evaluating its offer value with its criterion, the shopping agent aggregates their scores and assigns an overall score to the product offering.[15] This overall score represents the value that the product offering brings to the shopper based on the shopper's expressed preferences. Once a product offering has been evaluated, it is ready to be compared with other product offerings and presented to the shopper.

This integration of product and merchant brokering through integrative negotiations has the unique benefit in that constraints on product features can affect the decision of who to buy from. For example, only a certain merchant may be able to support a particular

---

[14] The term "offer value" signifies the value of a product offering's attribute – e.g., 2 days as the value for delivery time. This should not be confused with the economic meaning of value as in "value comparisons."
[15] As discussed in Chapter 5, Tête-à-Tête's value assessment is more involved than described here. For example, attributes are categorized by feature and each feature may be weighted relative to other features.

product configuration. Likewise, constraints on merchant features can affect the decision of what to buy. For example, if no merchant can accommodate the overnight delivery of a specific product, an alternate product which can be delivered overnight may be determined to have a better overall value. Related systems discussed in Chapter 2 are not able to consider product and merchant features concurrently.

The Tête-à-Tête Shopping Experience

Stepping through a shopping experience will help illustrate Tête-à-Tête's interaction model. However, it should be noted that this section only presents a brief overview of the components discussed in more detail in subsequent chapters. Furthermore, as discussed in Chapter 1, Tête-à-Tête approaches the problem of merchant differentiation by focusing on how merchants are perceived in the marketplace within the context of a shopping experience. A merchant's perceived differentiation rests with the shopper and will influence the shopper's merchant brokering decision (see Table 1). Therefore, the following illustration of Tête-à-Tête's interaction model is viewed predominantly from a consumer's perspective.

To shop online with Tête-à-Tête, a shopper accesses Tête-à-Tête's Web site through a standard Web browser. Upon selecting a specific product domain[16] (e.g., "notebook computers"), the shopper is asked one question, "Which of the following profiles best characterizes the person you are shopping for?"[17] This is followed with a list of *stereotypes* (discussed in Chapter 5) pertinent to the selected product domain. In the

---

[16] Currently, only anonymous shopping is supported. Furthermore, although Tête-à-Tête can handle multiple domains simultaneously, it is currently only populated with a notebook computer ontology and associated product and merchant data.

notebook computers example, these profiles may be Power User, Budget Conscious, Average User, and Road Warrior as shown in Figure 12.
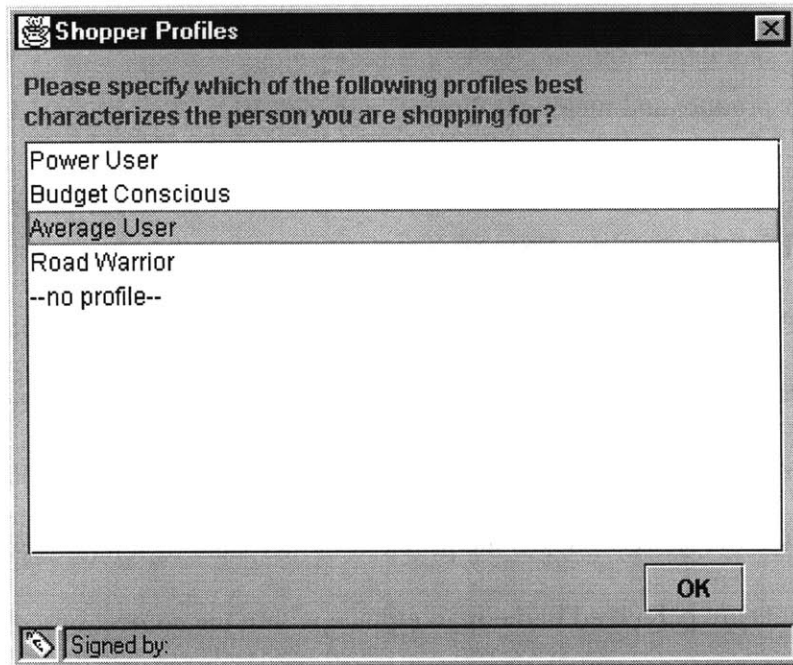
Figure 12: Tête-à-Tête's uses stereotypes (a.k.a. profiles) to bootstrap the shopping experience

Upon selecting a profile (e.g., Average User), several things happen "behind the scenes" in quick succession. The default preferences for the selected profile are sent as a criteria performative to all of the sales agents that sell notebook computers, which, in turn, respond with product offerings as proposal performatives.

From the shopper's perspective, however, selecting a profile immediately launches the shopper into a Tête-à-Tête shopping session with a presentation of product offerings that best satisfy the selected profile's default preferences as shown in Figure 13.

---

[17] Tête-à-Tête does not present the word "stereotype" to the shopper due to its negative connotations.
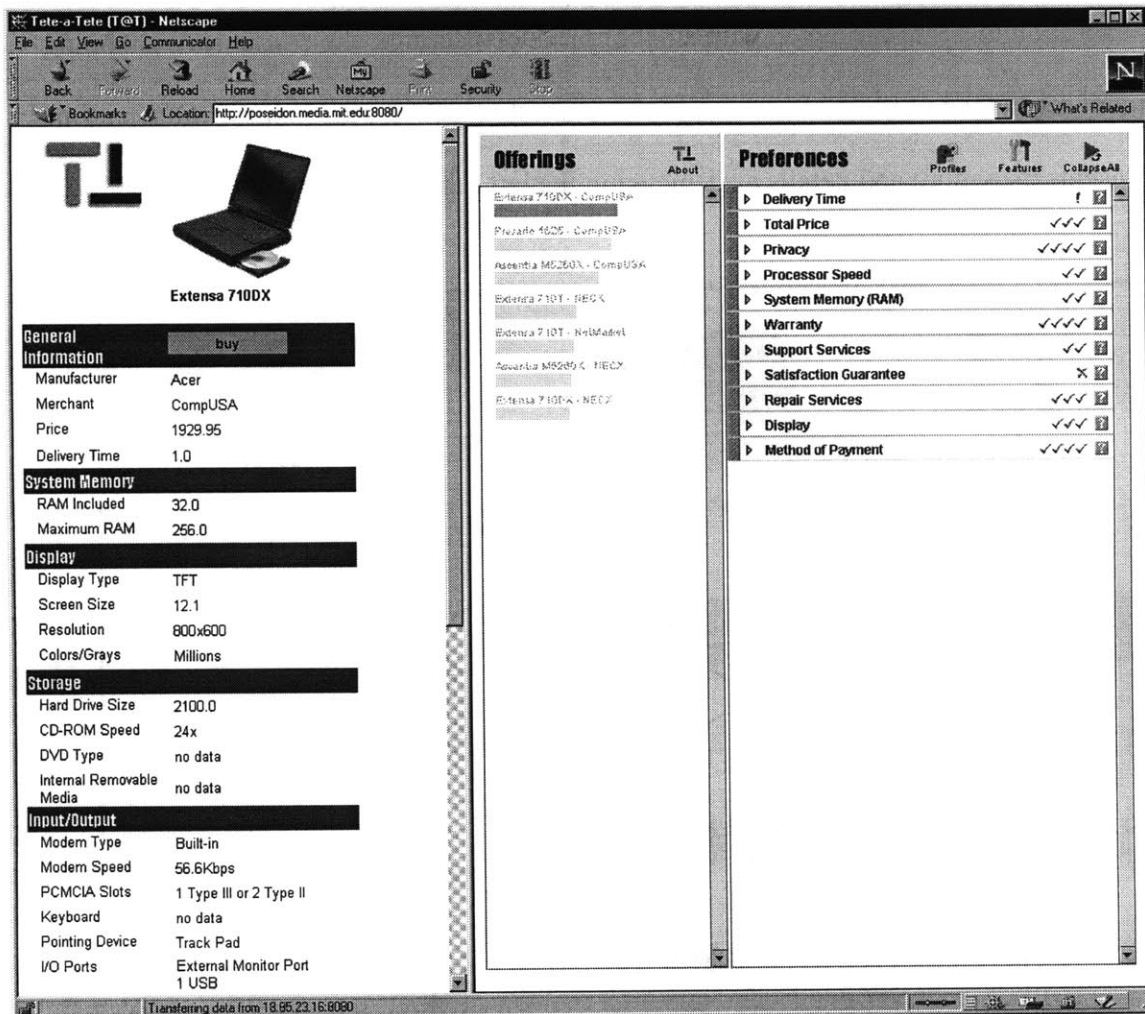
Figure 13: Tête-à-Tête has a three-panel interface for (from right to left) capturing preferences, listing and selecting product offerings, and displaying and comparing product offering details

The proposed product offerings are listed in the middle panel of Tête-à-Tête's three-panel interface as *value bars*. The absolute length of a product offering's value bar represents the value that that product offering brings to the shopper based on how well it satisfies the shopper's preferences as they are currently defined. The relative length of each value bar indicates the relative value each product offering brings to the shopper. The size, color, and shape of the vertically oriented value bars provide the shopper with a very quick read of the available product offerings including how constrained the shopper's preferences are.

In addition to being presented with a list of product offerings, the consumer's shopping agent selects the product offering with the greatest value (i.e., the longest value bar) and presents its picture and details as depicted in the left panel of Figure 14.
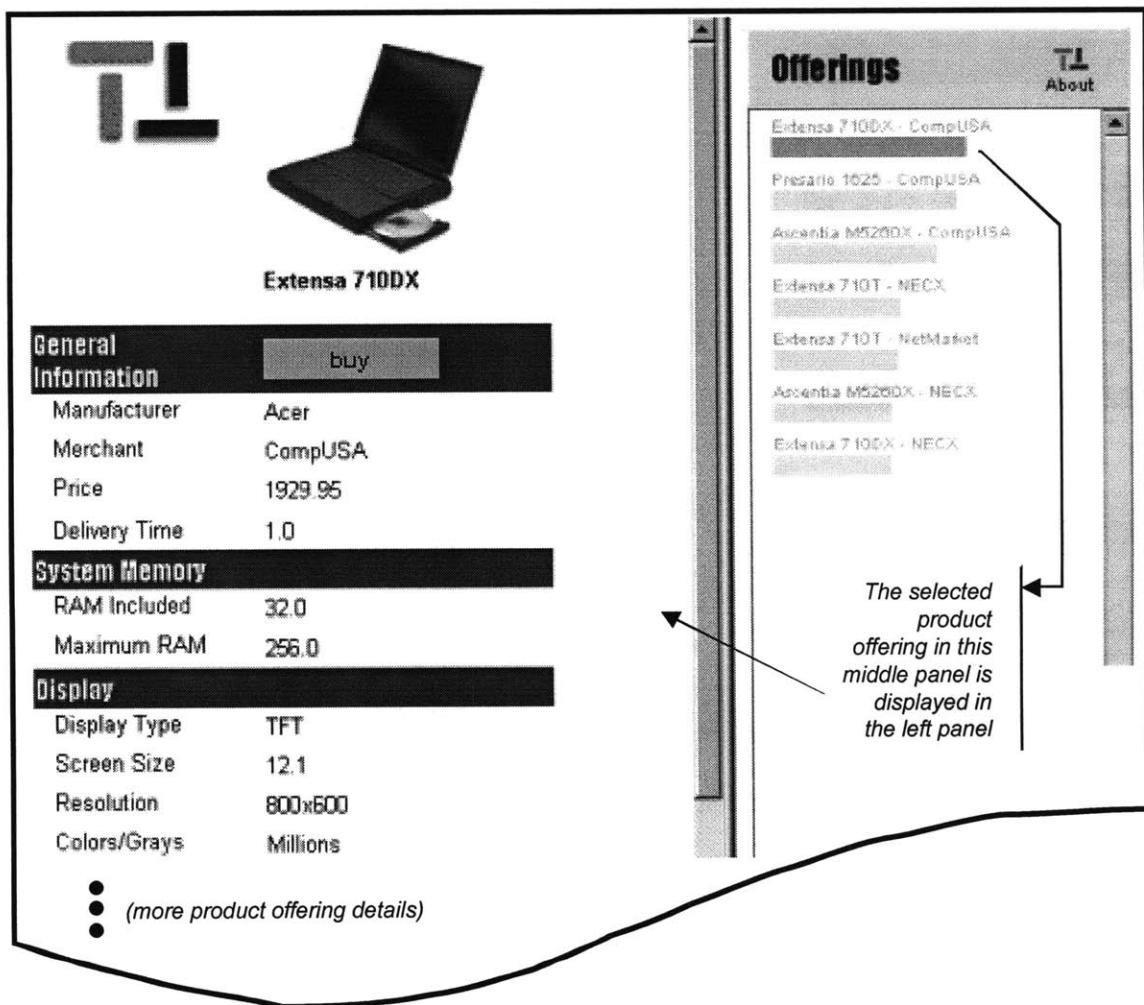
Figure 14: The left panel displays details of the selected product offering(s)

At this point, the shopper is able to make a purchase by clicking on the "buy" button below the product offering's picture in the left panel. Alternatively, the shopper can compare several product offerings side-by-side by selecting multiple value bars in the middle panel as shown in Figure 15.

Figure 15: Product offering details and side-by-side comparisons are displayed in the left panel

However, the shopper may not be satisfied with the default product offerings. In this case, the shopper can refine her preferences using the controls in the right panel. For example, if the shopper needs the notebook computer to be delivered quickly, she can change the "Delivery Time" feature by expanding its *module* and adjusting the range of days that is acceptable for delivery. This is done by sliding the lower and upper bounds of the *numeric range control* to their appropriate settings. Figure 16(a) shows an expanded delivery time module with its numeric range control set to the default preference for delivery time (between 3 and 21 days). Figure 16(b) shows the same

delivery time module after the shopper expresses a preference for a delivery time between 1 and 5 days. The triangular shape of the numeric range control with the taller end at the "1 day" setting, indicates that the shopper favors fast delivery times (fewer days) over slow delivery times (more days).
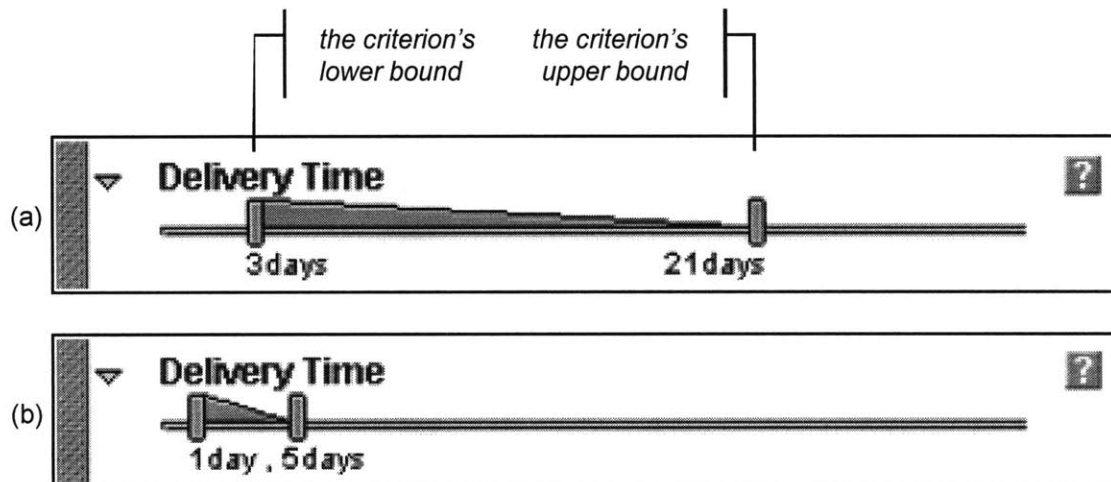
Figure 16: The "Delivery Time" feature's preference module

As the shopper adjusts her preference for delivery time, the change in criterion is immediately broadcast as critique performatives to all appropriate sales agents. The sales agents then respond with updates to their last set of product offerings in the form of proposal, counter-proposal, and withdraw-proposal performatives.[18] The shopping agent adjusts the product offerings and the lengths of their respective value bars accordingly.

From the shopper's perspective, the product offerings and their respective value bars in the middle panel get updated immediately upon each expressed change in preference. This provides the shopper with instant feedback of how her preferences affect which

---

[18] As discussed in Chapters 4 and 6, updating a list of previously sent proposals may entail a series of performatives to update and/or replace them.

product offerings best meet her needs. Also, by having the search criteria and its results in the same visual context, the shopper is better able to understand how each expressed preference determines which product offerings get proposed and how well each product offering satisfies the shopper's preferences.

As the shopper changes her preferences, if only one of the value bars is selected, then the shopping agent indicates in the right panel how well the respective product offering stacks up against each feature's criterion. Figure 17 shows the same delivery time preference settings as in Figure 16, but now with only one product offering selected in the middle panel.
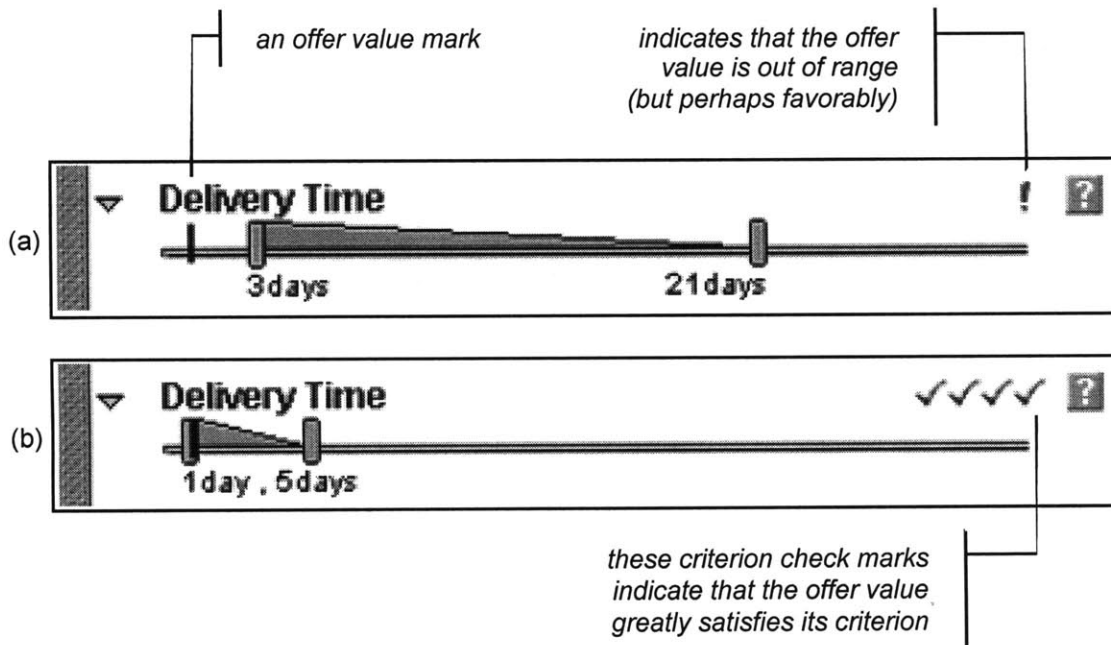
Figure 17: The "Delivery Time" feature's preference module (with feedback)

The selected product offering has an offer value of "1 day" as shown by the *offer value mark* in Figure 17(a). This offer value does not satisfy the delivery time criterion defined in Figure 17(a) as indicated by the red '!' mark. This mark alerts the shopper that the

offer value is out of range, but beyond the favorable end of the range. In this case, the product offering can be delivered even faster than the preference defines. The criterion defined in Figure 17(b), however, includes the offer value's "1 day" delivery time in its preference. This new criterion greatly satisfies the offer value indicated by the award of four green *criterion check marks*. The number of criterion check marks indicates how well the offer value satisfies its associated criterion. This feedback helps shoppers evaluate specific features of product offerings independently and helps them understand how well each individual attribute's offer value stacks up against its criterion. Unique to Tête-à-Tête, this feedback also assists the shopper in understanding how the feature evaluations affect the product offering's overall value assessment.

The shopper may still not be satisfied with the product offerings and may wish to continue refining her preferences. If the shopper is concerned with privacy for example, she may express several preferences regarding how merchants deal with privacy concerns and how they intend to use her personal information, as shown in Figure 18.
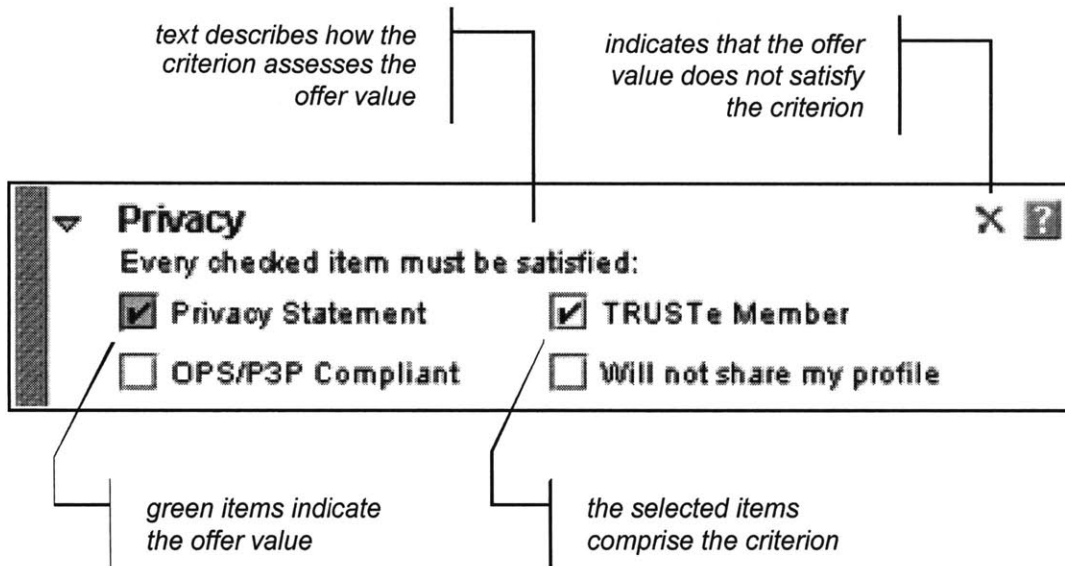
Figure 18: The "Privacy" feature's preference module

In this example, the shopper expresses a preference for each merchant to have a "Privacy Statement" and be a "TRUSTe Member" [35], indicated by their respective selections in the privacy module's *discrete set control*. The merchant that is offering the selected product has a privacy statement (indicated by the green *offer value mark*), but is not a TRUSTe member. The selected offering therefore does not meet the expressed criterion for the "Privacy" feature. This is indicated by the red "X" in place of criterion check marks.

If privacy is particularly important to the shopper, for example, she may also drag the "Privacy" module and drop it at the top of the list of features. The vertical orientation of the list of features (see Figure 29) represents their relative importance with the more important features towards the top of the list. Any feature may be dragged and dropped above or below any other feature to express their relative importance. These preferences are used by the shopping agent to more accurately assess product offerings.

Similarly to expressing preferences for merchant features (e.g., delivery time and privacy), the shopper may also express preferences for product features. Figure 19 shows the shopper having expressed preferences for a notebook computer's "Display" characteristics. The "Display" feature module incorporates several display attributes including whether the display may be color or grayscale, the type of display, as well as the display's size, resolution, and number of colors or grays.

Figure 19: The "Display" feature's preference module

The shopper may also express a preference for "Total Price," as shown in Figure 20.
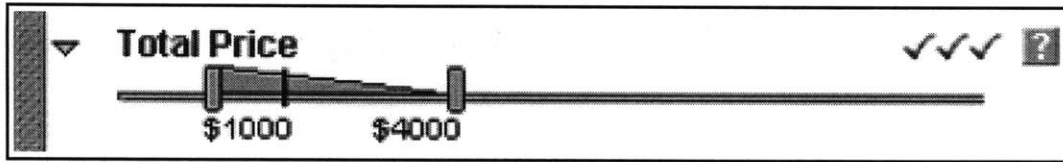
Figure 20: The "Total Price" feature's preference module

The total price attribute is evaluated similarly to other attributes. Total price includes all costs associated with the full transaction of the product including the price of the core product, the cost of additional options and upgrades, delivery costs, extended warranties, taxes, etc. By expressing price inclusively, the shopper is protected from costs that may be presented only after a product offering was selected for purchase (e.g., exaggerated delivery costs and service charges), and merchants are better able to differentiate on "free" services (e.g., automatic two-year extended warranty) as well as the efficiency of their order fulfillment. For example, local merchants or merchants with nearby warehouses may be able to offer added value to their customers if their delivery costs are lower than remote merchants'.

If the shopper doesn't understand what a specific feature means, she may press the module's help button to reveal some descriptive text. If the shopper doesn't have a preference for a specific feature, she can remove it from the list of features in the right panel. This is done by first pressing the "Features" button in the top of the preferences title bar. This displays the dialog box shown in Figure 21.
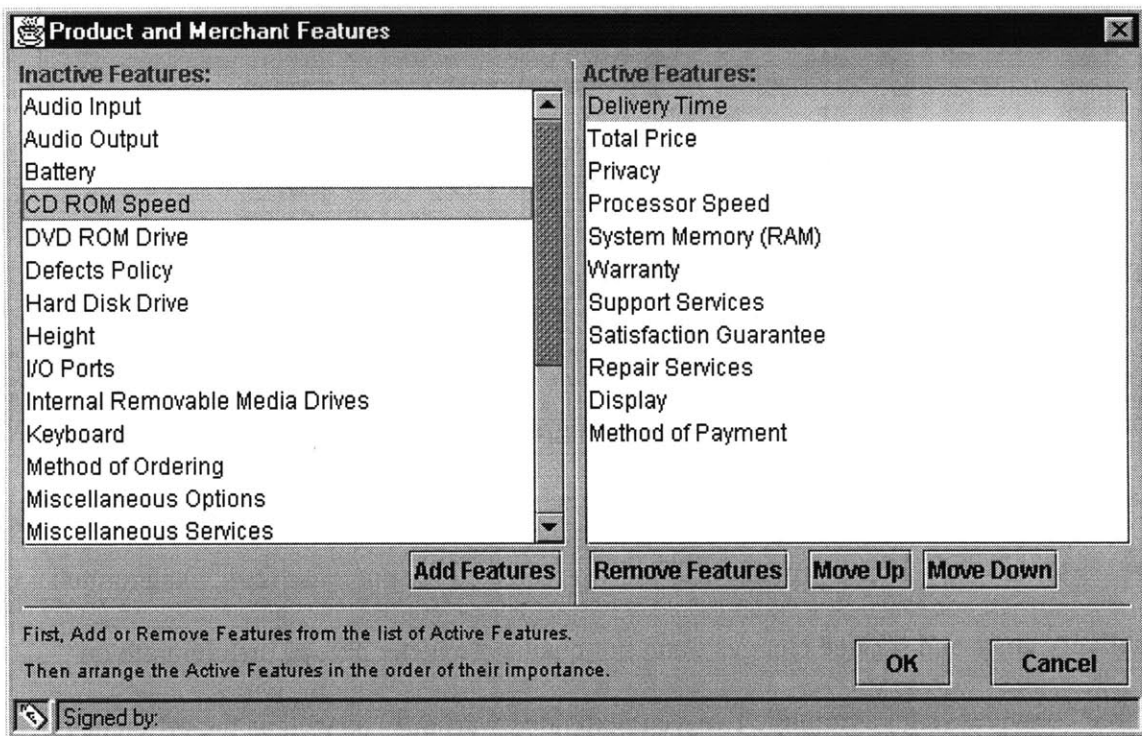
Figure 21: The "Features" dialog box

The Features dialog box allows a shopper to adjust the list of features in the right panel.[19] The right column of the dialog box lists the currently *active* features as listed in the right panel. Only active features are used to assess product offerings. To remove an active feature from consideration, the shopper selects it and presses "Remove Features." This moves it to the left column which lists the *inactive* features. To add a currently inactive feature to the right panel, the shopper selects it and presses "Add Features," moving it to the right column of the dialog box. Pressing the "OK" button hides the dialog box and adjusts the list of features in the right panel of the main shopping interface accordingly.

The shopping session continues until the shopper decides to withdraw from the session or purchase a product.

---

[19] See the "Shared Ontologies" section of Chapter 4 for a discussion on how these lists get defined. Thoughts on how these lists could be evolved and maintained over time are presented in Chapter 8.

Policies and Value-Added Services

Tête-à-Tête allows merchants to differentiate themselves during the shopping experience along dimensions other than just price. These other dimensions include a merchant's *policies* and *value-added services* such as extended warranties, forgiving return policies, extensive service contracts, special gift services, high product availability, superior customer service and support, diverse payment, loan and leasing options, fast delivery times with low costs, promotions and coupons, cross-manufacturer product configurations, community involvement and social awareness, privacy policies, and more. A merchant's policies and value-added services constitute a set of features that help define the merchant. These *merchant features* help differentiate the merchant in Tête-à-Tête by being presented along with product features in the right panel of the shopping interface, as shown in Figure 22.
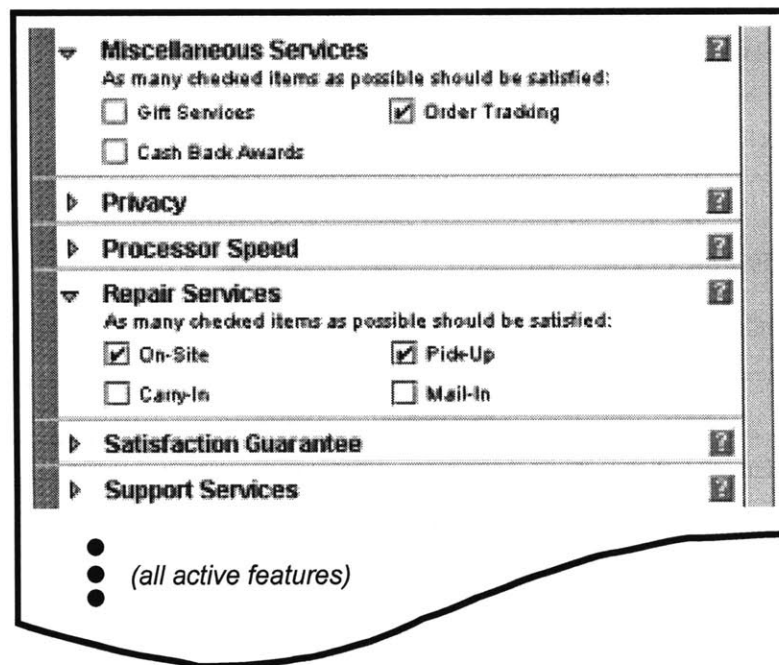
Figure 22: Tête-à-Tête allows shoppers to consider merchant features along with product features

Most importantly, shoppers are able to express preferences for these merchant features which are used to assess each product offering. Additionally, a shopper's expressed preferences for merchant features (as well as product features) influence which product offerings are proposed by each sales agent.

Product Domains

The example shopping experience above is for the complex product domain of "notebook computers." However, Tête-à-Tête is a general purpose online shopping system conducive to shopping in numerous complex product domains (not simple product domains such as books and music CDs). For example, once populated with the product and merchant features and data for "mountain bikes," Tête-à-Tête can assist shoppers in determining what mountain bike to buy and who to buy it from, as shown in Figure 23.
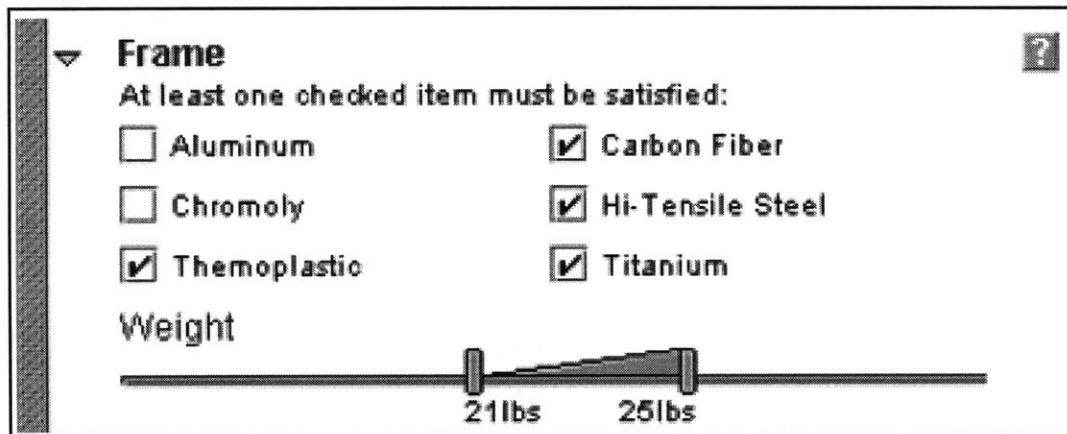
Figure 23: Tête-à-Tête supports numerous complex product domains

Other complex product domains that lend themselves to being shopped with Tête-à-Tête include skis, tents, automobiles, insurance policies, camcorders, VCRs, snow mobiles, telescopes, mutual funds, backpacks, stereos, refrigerators, boats, jet skis, and more.

Summary

This chapter introduced and illustrated Tête-à-Tête, an agent system that helps shoppers determine what to buy and who to buy from in a manner that overcomes the limitations of today's price comparison systems and online auctions. To accomplish this, Tête-à-Tête adopts an integrative negotiation protocol (based on bilateral argumentation) and decision support tools (based on multi-attribute utility theory) to assess product offerings. The following chapters elucidate each of these technologies.

4 Integrative Negotiation
Based on Bilateral Argumentation

The performatives of typical online auction protocols (English and Yankee) are binding price bids. These price bids fully capture a bidder's value of a given product offered by a specific online merchant at a certain point in time. Price bids are centrally managed through an auctioneering service which enforce the rules of the specified auction. Typically, merchants of an online retail auction require that each bidder provide financial account information (e.g., a credit card number) which gets automatically charged or debited each time the holder of the account wins an auction.

Looking at price comparisons (as for example used in BargainFinder [9] and Jango [10]) as a negotiation protocol, their performatives would be price requests and price quotes. Price requests are sent from consumers to merchants with or without the aid of an intermediary. Merchants respond to price requests with price quotes, also with or without the aid of an intermediary. Both price comparison performatives are non-binding allowing either party to withdraw from negotiations or accept the latest offer at any time. Ordinarily, price requests may be sent anonymously without needing to reveal identities or personal financial account information.

In practice, there are much fewer "rules of the game" involved in today's price comparisons than in online auctions. For example, the language of today's price comparison performatives are based on HTML (hypertext markup language) and are ambiguous and not standardized. Also, the "protocol" does not require that merchants respond to price requests at all (as in the BargainFinder case discussed in Chapter 2). This flexibility, expressiveness, and lack of enforcement accrue both benefits and costs.

Many of the benefits of using a flexible negotiation protocol, are the complement of what makes using online auction protocols problematic (see Chapter 2). Unlike the typical online auction protocols, more flexible protocols such as price comparisons allow for immediate purchases without delays, are simple to understand including the closing of deals, allow for consumers to consider alternatives throughout negotiations, permit more forgiving post-negotiation return policies, allow for consumers to (optionally) protect their identities and personal account information during negotiations, may allow for more expressive information to be sent (e.g., valuable consumer preferences), may permit merchants to differentiate themselves along multiple dimensions other than just price, and may afford consumers the opportunity to evaluate the full value of each product offering (including price) prior to committing to any specific product or merchant. In short, flexible negotiation protocols are appropriate for today's retail markets.

However, a flexible negotiation protocol such as price comparisons also accrue costs. Most notably, the fact that requests and responses (e.g., price quotes) are retractable and non-binding allows for the dissemination of misinformation without consequences. For example, a merchant may execute a "bait and switch" – i.e., offer a product that it may

not even have in stock (or pretend not to have in stock) and then trick the consumer into buying an inferior or alternate product with a greater profit margin.[20] Additionally, flexible negotiation protocols demand far more performatives to be sent, each containing more information than a single price bid (e.g., product name, model, manufacturer, price, etc.) which is problematic in low-bandwidth environments. Furthermore, if overly expressive and non-standardized, a flexible negotiation protocol leaves more room for misunderstandings among negotiating parties – a lack of availability, a misinterpretation of currencies, an incorrect identification of products, etc. This could lead to more post-negotiation reconciliations and thus more unsatisfied customers and greater overall transaction costs. Finally, the nature of the protocol may suffer other problems such as those found in price comparisons (discussed in Chapter 2).

Therefore, an open question is: For complex retail products, is there a way to reap the benefits of more flexible protocols than online auction protocols without their costs? Tête-à-Tête proposes one answer to this question with an integrative negotiation protocol based on bilateral argumentation. Tête-à-Tête's negotiation protocol defines a more formalized exchange of performatives between consumers and merchants than today's price comparisons, but is less restrictive than today's online auctions. Tête-à-Tête's negotiation protocol attempts to overcome many of the limitations of these other protocols by finding the right balance of formality, expressiveness, and appropriateness for helping consumers and merchants migrate towards a pareto optimal deal.

---

[20] This occurs today in physical-world retail. Although there are "truth in advertising" laws to help minimize "bait and switch," it is technically more robust for the negotiation protocol itself to prevent (or at least minimize) lying and cheating rather than rely on governmental regulations.

Tête-à-Tête's Integrative Negotiation Protocol

Automated argumentation has been studied and applied to a number of problem domains. Sycara's PERSUADER system, for example, acts as a mediator and uses argumentation to resolve conflicts in labor management disputes [36]. Built on this earlier work, Parsons, et al have formalized the argumentation process as a symmetric exchange of proposal, critique, counter-proposal, withdraw, and accept illocutions [32].

Tête-à-Tête adopts a *bilateral argumentation* style of negotiation with each shopping agent negotiating in parallel with multiple sales agents.[21] Requests from shopping agents are broadcast to all sales agents. Sales agents' responses to requests are not broadcast to other sales agents, however. Rather, they are sent only to the requesting shopping agent (i.e., a sealed bid). Tête-à-Tête also extends the current set of argumentation performatives with two new performatives: criteria and withdraw-proposal. The *criteria* performative is used by shopping agents to initiate negotiations. The *withdraw-proposal* performative is used by sales agents to manage their set of product offerings. This thesis introduces the concept of managing a limited number of product offerings which, once proposed, are henceforth updated and retracted using counter-proposal and withdraw-proposal performatives, respectively.

Furthermore, Tête-à-Tête employs argumentation performatives asymmetrically to more accurately model today's retail environment. Criteria, critique, withdraw, and accept performatives are sent only from shopping agents to sales agents and proposal, counter-

---

[21] Unlike other negotiation systems such as PERSUADER [36], Negotiation Assistant [37], OptiMark Technologies [38], and Priceline [39], Tête-à-Tête does not perform or require any type of mediation or arbitrage independent of its shopping agents and sales agents. Mediation is achieved through the full automation of sales agents and a shopping agent's solicitation of its owner's preferences.

proposal, and withdraw-proposal performatives are sent only from sales agents to

shopping agents, as depicted in Figure 24.
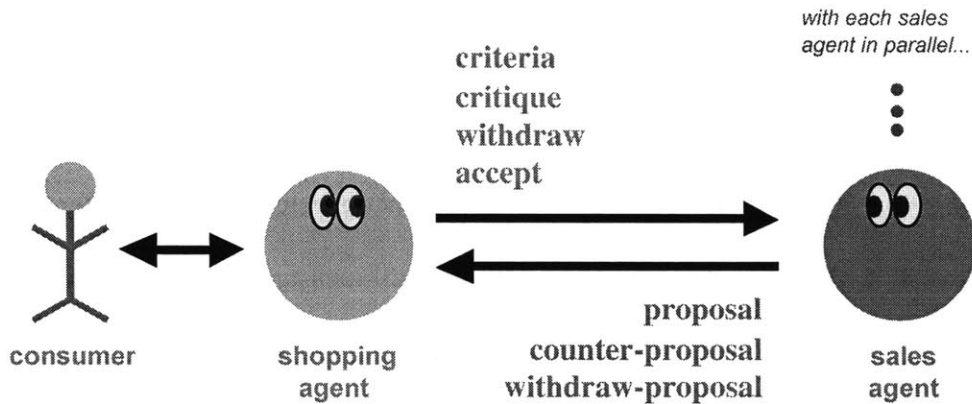
Figure 24: Tête-à-Tête's integrative negotiation protocol performatives

Tête-à-Tête's performatives are mostly composed of information on product and

merchant features. The negotiation protocol defines these features identically. Each

feature has a name (e.g., "Display"), a relative *position* (i.e., a vertical orientation in the

list of features), and a flag indicating whether it is currently *active* or not. Also, each

feature is comprised of one or more *attributes* which also have names (e.g., "Display

Size" and "Display Resolution"). Each attribute has both an *offer value* (which are

defined by a product offering) and a *criterion* (which are defined through a shopper's

expression of preferences) for evaluating the offer value.

Attribute and Criterion Types

Tête-à-Tête defines three types of attributes: numeric range, discrete range, and discrete

set. The type of attribute determines how an attribute's offer value and criterion are

represented within negotiation performatives.

The *numeric range attribute* type defines an attribute with a real number range between established minimum and maximum bounds. An offer value for a numeric range attribute is a real number within these bounds. The *numeric range criterion* type is used to define the criterion for numeric range attribute types. It defines a real number range between an established *lower bound* and *upper bound* that each fall within its attribute's minimum and maximum bounds. The real numbers defining an attribute's range, an offer value, and a criterion's range are each tagged with its units of measurement (which may be dissimilar). The "Delivery Time" attribute shown in Figure 16 is an example of a numeric range attribute with a numeric range criterion.

The *discrete range attribute* type defines an attribute with an ordered list of tokens. An offer value for a discrete range attribute is one of the attribute's list of tokens. The *discrete range criterion* type is used to define the criterion for discrete range attribute types. It defines a sub-range on the attribute's ordered list of tokens between an established *lower bound* and *upper bound*. The "Display Resolution" attribute shown in Figure 19 is an example of a discrete range attribute with a discrete range criterion.

The *discrete set attribute* type defines an attribute with an unordered set of tokens. An offer value for a discrete set attribute is an unordered subset of the attribute's set of tokens. The *discrete set criterion* type is used to define the criterion for discrete set attribute types. Like the offer value, these define an unordered set of tokens that are a subset of its attribute's set of tokens. The "Privacy" attribute shown in Figure 18 is an example of a discrete set attribute with a discrete set criterion.

Criteria and Critique Performatives

Tête-à-Tête's integrative negotiation protocol defines an initialization step consisting of a *criteria* performative being sent from a consumer's shopping agent to each sales agent as a request for proposals. This initial criteria performative comprises the shopper's default preferences as defined by the initially selected stereotype (see Chapter 5) as well as meta-data to further define the interaction. The criteria performative is defined as shown in Table 2.

Field	Value
userid	the ID of the shopper or "anonymous"
requestid	the ID of the request during this shopping session
domain	the product domain being shopped
maxproposals	the max. # of proposals permitted in response
ufn	the function defining the features' relative weightings
feature*	zero or more features

Table 2: The criteria performative

The *userid* field in the criteria performative uniquely identifies the shopper to the sales agent. For example, the shopper may have an account with the merchant and the userid would map to this unique account. Alternately, the shopper may shop anonymously.[22] The *maxproposals* field in the criteria performative indicates the maximum number of proposals each sales agent is permitted to send in response. The shopping agent enforces this constraint for purposes of scalability as discussed in Chapter 6. The *ufn* field refers to a utility function that defines the relative weightings of the features. For example, a

---

[22] As discussed in Chapter 7, a sales agent is created to manage negotiations with each shopping agent. These negotiations occur over a socket connection which uniquely identifies the shopper to the sales agent throughout a shopping session. This unique identification is necessary in order for Tête-à-Tête to support anonymous shopping.

value of "LinearPreferLowerUFN" indicates a linear utility function with a preference towards features that have lower criterion positions. These are discussed further in Chapter 5. The features listed in the criteria performative contain each attribute's default criterion and are defined as shown in Table 3.

Field	Value
name	the name of the feature
position	the position (i.e., vertical orientation) of the feature
active	'true' if the feature is active; 'false' otherwise
attribute+	one or more attributes with criterion

Table 3: The feature definition for the criteria performative

Each of the three types of attributes and criterion are defined somewhat differently. The criterion for the numeric range, discrete range, and discrete set attributes are defined respectively in Table 4, Table 5, and Table 6.

Field	Value
name	the name of the attribute
units	the criterion's units of measurement
ufn	the function defining the criterion's weightings
lowerbound	the criterion's lower bound (a real number)
upperbound	the criterion's upper bound (a real number)

Table 4: The numeric range attribute's criterion definition for the criteria performative

Field	Value
name	the name of the attribute
ufn	the function defining the criterion's weightings
lowerbound	the criterion's lower bound (a token)
upperbound	the criterion's upper bound (a token)
item+	an ordered list of one or more tokens

Table 5: The discrete range attribute's criterion definition for the criteria performative

Field	Value
name	the name of the attribute
ufn	the function defining the criterion's weightings
additem*	an unordered set of zero or more tokens

Table 6: The discrete set attribute's criterion definition for the criteria performative

The *ufn* fields in the criterion definitions above each refer to a utility function that defines the criterion's weightings of the feature's offer value. For example, a value of "LinearPreferUpperUFN" indicates a linear utility function that prefers offer values that are towards the upper bound (for numeric and discrete range criterion). These are discussed in more detail in Chapter 5. The *item* field in the discrete range criterion definition is the complete ordered list of tokens for the attribute. For example, for the "Display Resolution" attribute shown in Figure 19, all of the possible display resolutions in the discrete range control would be listed in this item field. Finally, the *additem* field in the discrete set criterion definition lists the attribute's default preferences. For example, for the "Privacy" attribute shown in Figure 18, the "Privacy Statement" and "TRUSTe member" items in the discrete set control would be listed in this additem field (assuming they were the default preferences).

Critique performatives are defined similarly to criteria performatives except that only changes to criterion are sent. All of the other fields are optional. This requires that each sales agent maintain the shopping agent's criteria state. The benefit of this approach (over sending the entire criteria each time) is that the size of critique performatives are much smaller than criteria performatives. Table 7 shows an example critique performative showing the change in criterions between Figure 16(a) and (b).

Field	Value
userid	"debbie"
requestid	14
feature 1	
name	"Delivery Time"
attribute 1	
name	"Delivery Time"
lowerbound	1
upperbound	5

Table 7: An example critique performative for the "Delivery Time" feature

Proposal, Counter-Proposal, and Withdraw-Proposal Performatives

Upon receiving an initial criteria performative from a shopping agent, a sales agent may respond with a set of *proposal* performatives each defining a product offering. However, the sales agent is constrained to only send up to the number of proposals defined by the criteria's maxproposals field. How the optimal proposals are determined is the subject of Chapter 6. Once they are determined, however, they are sent to the shopping agent as a response that includes proposal performatives as defined in Table 8.

Field	Value
merchantid	the ID of the merchant
requestid	the ID of the request being responded to (from criteria)
note?	a note to the shopper (optional)
proposal*	zero or more product offerings

Table 8: The response to an initial criteria request includes zero or more proposal performatives

The value of the proposal field is zero or more product offerings. Each product offering is packaged as a proposal performative as shown in Table 9.

Field	Value
proposalid	the ID of the proposal
manufacturerid	the ID of the manufacturer of the base product
productid	the ID of the base product
productname	the product's name and model
imageurl	the URL of an image of the product
feature*	zero or more features

Table 9: The proposal performative

The ID of the proposal is unique for a given merchant. It allows the merchant's sales agent to manage the proposal throughout negotiations. This includes altering one or more of its offer values or withdrawing the proposal so that a new one could take its place. The manufacturer ID and product ID uniquely define a specific base product – for example, a manufacturer ID of "Compaq" and a product ID of "Presario1625." Proposals may have a one-to-one mapping with specific base products, but this is not necessary.[23] If deemed optimal, a sales agent may send multiple proposals for the same base product

---

[23] Tête-à-Tête's sales agents currently use this approach as discussed in Chapter 6.

(but likely configured in different ways). The features listed in the proposal performative contain each attribute's offer value and are defined as shown in Table 10.

Field	Value
name	the name of the feature
attribute+	one or more attributes with offer values

Table 10: The feature definition for the proposal performative

The types of attributes in product offerings must exactly match those in the initial criteria request. However, instead of containing criterion, each proposal performative contains the offer values of its associated product offering as defined in Table 11, Table 12, and Table 13.

Field	Value
name	the name of the attribute
units	the offer value's units of measurement
value	the offer value (a real number)

Table 11: The numeric range attribute's offer value definition for the proposal performative

Field	Value
name	the name of the attribute
value	the offer value (a token)

Table 12: The discrete range attribute's offer value definition for the proposal performative

Field	Value
name	the name of the attribute
additem*	an unordered set of zero or more tokens

Table 13: The discrete set attribute's offer value definition for the proposal performative

After the sales agent responds to the initial criteria performative, the shopping agent may send critique performatives which capture the shopper's change in preference. The sales agent can now either add, update, or remove either of its previously sent proposals. Updating the configuration of a product offering is accomplished by sending a *counter-proposal* performative. Counter-proposal performatives are like proposal performatives except that only changes to offer values are sent and that the base product cannot be changed (only its configuration). All of the other fields are optional. This requires that each shopping agent maintain each sales agent's set of proposals and each proposal's state. As with critique performatives, the benefit of this approach (over sending the entire proposal each time) is that the size of counter-proposal performatives are much smaller than proposal performatives. Table 7 shows an example counter-proposal performative showing a change in offer value for the "Delivery Time" and "Total Price" attributes.

Field	Value
merchantid	"CompUSA"
requestid	14
note?	"Hello, Debbie!"
proposal 1	
proposalid	4
feature 1	
name	"Delivery Time"
attribute 1	
name	"Delivery Time"
value	1
feature 2	
name	"Total Price"

attribute 1	
name	"Total Price"
value	1999.00

Table 14: An example counter-proposal performative

Sales agents may also add and replace product offerings. Adding a new product offering is equivalent to sending another proposal performative. However, sales agents are not permitted to exceed their maximum number of proposals (as defined by the maxproposals field in criteria and critique performatives). Therefore, sales agents may also need to send *withdraw-proposal* performatives to remain within their allotted number of proposals. A withdraw-proposal performative contains the proposalid of the proposal to withdraw as shown in Table 15.

Field	Value
proposalid	the ID of the proposal

Table 15: The withdraw-proposal performative

Withdraw and Accept Performatives

At any point during negotiations, the shopper may either withdraw from the shopping session or accept a proposal.[24] *Withdraw* performatives are simple messages indicating a withdrawal from the shopping session. *Accept* performatives are only sent to the sales agent offering the accepted proposal and not broadcast to all sales agents, unlike the other performatives. Accept performatives are also simple messages and lead the shopper to the payment and delivery stage (see Table 1).

---

[24] Tête-à-Tête currently assumes that merchants will never have an incentive to withdraw from negotiations after having agreed to participate in them.

Qualitative Analysis

Tête-à-Tête's three types of attributes – numeric range, discrete range, and discrete set –

allow product offerings to be defined with an expressive and extensible set of features.

This affords merchants numerous dimensions to differentiate their product offerings. At

the same time, consumers are able to express preferences for features with a fair amount

of detail and accuracy to help them identify which product offering best meets their

overall needs. Tête-à-Tête's integrative negotiation protocol even permits features to be

introduced dynamically. The potential benefit of this is discussed in the "Distributed

Ontology Evolution" section of Chapter 8.

As do price comparisons, Tête-à-Tête's negotiation protocol overcomes most of the

limitations of online auction protocols (discussed earlier in this chapter). Furthermore,

Tête-à-Tête also overcomes many of the limitations of price comparisons. For example,

Tête-à-Tête helps minimize the relatively large bandwidth requirements for exchanging

expressive performatives by only sending changes in preferences and proposals after they

have been initially sent. Also, Tête-à-Tête leaves less room for misinterpretations by

allowing more terms of the transaction to be explicitly defined in a formalized manner

including unique product IDs and units of measurement for numeric range features.[25]

Additionally, Tête-à-Tête can be extended to include a reputation facility which helps

indirectly solve the problem of merchants supplying misinformation (e.g., inaccurate

delivery times). This reputation facility can be incorporated as one or more features for

consumers to consider in their merchant brokering decision, as suggested in Chapter 8.

---

[25] Tête-à-Tête's protocol formalism is an XML application as revealed in Chapter 7.

Tête-à-Tête's integrative negotiation protocol resembles a non-binding, iterative, first-price, sealed-bid auction (which is not a typical online auction). Shopping agents broadcast a request for proposals which is responded to in kind. Competition is among sellers for buyer patronage (similar to Priceline [39], but non-binding). The non-binding, iterative nature of the protocol is uncommon for auctions, but it may share properties (e.g., optimal strategies, equilibrium points, and efficiencies) with other iterative games. Such an analysis is proposed in Chapter 8.

From a game theoretic perspective, a player's strategy is usually held private; revealing a strategy is tantamount to losing the game.[26] Online auctions assume as much by keeping bidding strategies private from sellers. In cases where the bidder voluntarily reveals his strategy to an automated bidding agent, it is assumed that the market maker is to be trusted and will not divulge this information to sellers. Tête-à-Tête, on the other hand, exposes all of a consumer's expressed preferences for product offerings including total price and utility weightings on features.[27]

A fair question is: Isn't it possible and likely for Tête-à-Tête sales agents to price discriminate since they know each consumers' product offering valuations? For example, if a shopper expresses a preference for a total price between $2000 and $4000 and another shopper expresses a preference for a total price between $2000 and $5000, what would prevent a merchant's sales agent from price gouging each customer – e.g., offering

---

[26] However, there are games where it is, in fact, advantageous to publicly and irrevocably commit to a strategy (e.g., games of chicken, defending market share, and nuclear deterrence).

[27] An alternative protocol suggested in Chapter 8 is one where all preferences except price are revealed to merchants.

a given product to the first shopper for $4000 and to the second shopper for $5000? There are three answers to this question.

First, there is nothing technically stopping merchants from price discriminating in Tête-à-Tête. In fact, in some markets (e.g., airline tickets), price discrimination may be seen as an expected and accepted business practice.

Second, if, however, there is an expectation for fixed and equal pricing and a retail merchant is recognized for attempting to price discriminate, then the reputation of the merchant could be sullied which hurts long-term profitability. The point is not that it is inherently wrong to price discriminate, but rather that the merchant is not satisfying its customers' expectations. Online reputation brokering services such as BizRate [40] are already allowing customer's to globally publish complaints (and compliments) about online merchants. More sophisticated reputation brokering services are currently being developed in academic settings [41]. In general, the fact that shopping can be anonymous (even in the physical world) and that the Internet allows information (including prices and reputations) to be shared freely around the world among even unknown parties makes it difficult for online merchants to price discriminate without it being known and potentially used against them.

Third, retail markets are a competition among merchants (and not consumers, as online auctions would suggest). Tête-à-Tête unabashedly presents this competition through direct comparisons of competitors' product offerings. If a merchant attempts to price discriminate in this environment (all else being equal), then the likely result is that the merchant's product offerings will not be judged highly relative to a competitors' product

offerings. This characteristic of Tête-à-Tête offsets a merchant's ability to price discriminate. Although it is technically feasible for merchants to collude in order to jointly price discriminate (similar to coalitions in auctions), the fact that requests for proposals are non-binding, that shoppers can be anonymous, and that consumers can incrementally decrease the upper bound preference for total price makes this practice tenuous. In addition, there are antitrust laws that legally prevent such practices. Nevertheless, this important topic warrants further study as suggested in Chapter 8.

It is important to note that price discrimination is not the same as pricing with market power. Price discrimination entails offering the same product to different customers at different prices (not withstanding external factors such as differences in delivery costs). Pricing with market power is the ability to price at a premium due to a greater perceived value in the marketplace. While somewhat hindering the former, Tête-à-Tête promotes the latter through its mechanisms for differentiating merchants' product offerings.

Today, there is an expectation that prices in most retail markets are fixed. However, it has also been shown that prices in certain online markets fluctuate more readily than their physical-world counterparts [42]. This begins to blur the lines of fixed versus dynamic pricing (which is related to, but not the same as, price discrimination). It is not difficult to imagine an online shopping environment in the near future where prices fluctuate as quickly as they do in stock markets with agents continuously analyzing and influencing market dynamics. However, simulations have shown that online markets where price is the only merchant differentiator are unstable with "endless price wars" [43]. Fortunately, similar simulations where merchants have alternate dimensions for differentiation have

been shown to be stable and achieve "price equilibrium" [44]. This suggests that even if retail markets embrace dynamic pricing, there is still a critical need for merchant differentiation mechanisms such as those provided by Tête-à-Tête.

Shared Ontologies

Tête-à-Tête's integrative negotiation protocol is expressive and extensible. Most of the strengths of these characteristics are described above and include a rich mechanism for differentiating product offerings and expressing preferences as well as permitting new features to be introduced dynamically. However, in an environment with heterogeneous sales agents (which is the environment for which Tête-à-Tête is designed), the definition of features need to be commonly shared and understood. Otherwise, meanings are ambiguous, comparisons are senseless, and negotiations become futile.

Tête-à-Tête partially solves this problem by maintaining open and centralized Registry and Dictionary databases comprising, respectively, product domain hierarchies (including which merchants sell products within each domain) and feature definitions (for both product and merchant features). These are detailed in Chapter 7. Currently, Tête-à-Tête's ontologies are centrally managed through a human editorial board similar to how the categories within Yahoo! are managed [45]. This "first to market with a powerful idea" approach sometimes leads to de facto standards.

An alternative approach is to adopt standards from industry associations and other standards bodies. For example, the RosettaNet consortium aims to define the business interfaces for Information Technology (IT) supply chains [46]. The problem with this

single "standards body" approach, however, is that it often takes too long for the partners to make progress and reach a consensus. This is particularly problematic for the potential dizzying pace of electronic commerce.

The current environment is one where there are multiple competing product ontologies, virtually one for every manufacturer, requiring laborious translations from one definition to the next. Furthermore, most proposed solutions to this problem still require a central human editorial board (e.g., Consumer Reports [47]). Ideally, shared ontologies are created and evolved quickly, effectively, and without a centralized human editorial board. Chapter 8 proposes such a mechanism as an enhancement to Tête-à-Tête.

Summary

This chapter discussed Tête-à-Tête's integrative negotiation protocol. This protocol extends the current set of argumentation performatives with two new performatives (criteria and withdraw-proposal) and includes the concept of managing a limited number of proposals. The performatives of the protocol are largely based on the preferences and offer values for attributes of which there are three types – numeric range, discrete range, and discrete set. These allow for expressive and extensible product offerings and preferences to be defined which help differentiate merchants and help consumers identify which product best meets their needs. This need assessment is the subject of the next chapter.

5   Decision Support
Based on Multi-Attribute Utility Theory

To be useful to the decision maker, an attribute should be both comprehensive and measurable. An attribute is comprehensive *if, by knowing the level of an attribute in a particular situation, the decision maker has a clear understanding of the extent that the associated objective is achieved. An attribute is* measurable *if it is reasonable [...] to assess the decision maker's preferences for different possible levels of the attribute, for example, in terms of a utility function or, in some circumstances, a rank ordering. Furthermore, we would like to accomplish both [of] these tasks without taking an inordinate amount of time, cost, or effort.*

- R. Keeney and H. Raiffa, Decisions with Multiple Objectives: Preferences and Value Tradeoffs, *first published 1976*

Given a negotiation protocol, there remains the issue of determining an optimal strategy in order to maximize a decision maker's objectives. More specifically, given Tête-à-Tête's integrative negotiation protocol, the shopper is still challenged with the decisions of what to buy and who to buy from among the proposals from sales agents.[28] For complex products, making this decision can be a daunting task since the goal of identifying the most suitable product offering involves multiple interdependent objectives. For example, a shopper may want the notebook computer with the best performance, the biggest display, the lightest weight, the quickest delivery time, the longest warranty, and the cheapest price. In most cases, the shopper is not able to

---

[28] The decision support for sales agents to determine these proposals is the subject of the next chapter.

maximize each of these objectives independently, but instead, is forced to make value tradeoffs to determine which product offering best meets all of these objectives when they are considered concurrently. The purpose of Tête-à-Tête's decision support module, is to help shoppers make these value tradeoffs.

Multi-Attribute Utility Theory

The study of how to analyze multi-objective decisions comes from economics research and is called *multi-attribute utility theory* (MAUT). This theory has been successfully applied to making decisions in numerous domains such as electrical power vs. air quality, airport location, heroin addiction treatment, medical diagnostic and treatment, labor relations [36], business decisions, and political decisions [31]. General purpose software tools based on MAUT, such as Logical Decisions for Windows (LDW) [48], help decision makers formulate and evaluate multi-objective decisions. However, general purpose tools do not fit all tasks. For the specific task of online shopping, MAUT packages like LDW are too involved and time-consuming for the average online shopper to use.

The decision support module for comprehensive online shopping must complement the system's underlying protocols and interaction model. For example, in Tête-à-Tête, product offerings are only revealed during negotiations, not a priori as in PersonaLogic [6]. Also, Tête-à-Tête's proposed integrative negotiation interaction model allows shoppers to iteratively refine their preferences which are used by the decision support module to evaluate proposals as well as to elicit counter-proposals from sales agents. Furthermore, burdening the shopper with MAUT's requisite numeric details would deter

even the most staunch shopper. User-friendly mechanisms are needed to capture and refine shoppers' preferences. In addition, to help consumers understand their needs and how they relate to the available product offerings, new techniques for visualizing results are essential. These characteristics demand a holistic design and a specialized decision support module.

Value Tradeoffs

Tête-à-Tête formulates the goal of identifying the "best" proposal from sales agents as a multi-objective decision. This encapsulates the value tradeoffs necessary to determine which complex product to buy and who to buy it from. As noted in the opening quote of this chapter, it is important that the attributes comprising a multi-objective decision are comprehensive and measurable [31]. Tête-à-Tête's attributes are *measurable* by each having an offer value and a criterion to evaluate the offer value, as discussed in Chapter 4. The measurement is complete in that every product and merchant attribute of the product offering is evaluated. Tête-à-Tête's attributes are also *comprehensive* by helping shoppers have a "clear understanding of the extent that the associated objective is achieved" [31]. This requirement is satisfied through novel interactions and visual feedback mechanisms associated with the each attribute's evaluation and the evaluation of the product offering as a whole. This is discussed in more detail in the "Value-based Online Shopping" section of this chapter.

In Tête-à-Tête, proposals from sales agents each represent a specific product offering. As discussed in Chapter 4, each product offering is comprised of product features and merchant features. Each feature has a relative *position* (i.e., a vertical orientation in the

list of features) and a flag indicating whether it is currently *active* or not. Also, each

feature is comprised of one or more *attributes*. Each attribute has both an *offer value*

(which are defined by a product offering) and a *criterion* (which are defined through a

shopper's expression of preferences) for evaluating the offer value. Given this data

organization, Tête-à-Tête assesses the value of each product offering as follows:

$$v = \sum_{j=1}^{m} w_j \left(\frac{\sum\limits_{i=1}^{nj} f_i(x_i)}{nj} \right)$$

Equation 1: Tête-à-Tête's raw value assessment equation for proposals

where x_i is the normalized utility of attribute i, $f_i(x_i)$ is the weighted utility of attribute i, n_j

is the number of attributes in feature j, w_j is the relative weighting for feature j, and m is

the number of active features. The calculation proceeds as follows: for every j active

feature, each i attribute is first normalized resulting in x_i, and then weighted using $f_i(x_i)$.

These weighted attribute utilities are averaged for each feature resulting in a normalized

utility for the feature. Each feature j is then weighted relative to other features using

relative weighting w_j. These weighted feature utilities are summed resulting in raw value

assessment, v.

Utility Functions

As discussed in Chapter 4, there are three types of attributes each with an associated

criterion: numeric range, discrete range, and discrete set. Each attribute type is evaluated

somewhat differently; however, they each use a similar set of utility functions to weight

the attribute's utility based on how well the offer value satisfies the attribute's criterion.

Utility functions are also used to relatively weight the attributes. Tête-à-Tête's utility functions are listed in Table 16.

Name	Graph	f(x) =
LinearPreferLowerUFN		$1 - x$
LinearPreferMiddleUFN		$x < 0.5 ? 2x ; 2(1 - x)$
LinearPreferUpperUFN		x
SqaurePreferLowerUFN		$(1 - x)^2$
SquarePreferMiddleUFN		$x < 0.5 ? 4x^2 ; 4(1 - x)^2$
SquarePreferUpperUFN		x^2
UniversalUFN		$x = 1 ? 1 ; 0$
ExistentialUFN		$x = 0 ? 0 ; 1$

Table 16: Tête-à-Tête's utility functions where $0 < x < 1$ (inclusive)

Given a *normalized utility* of x between 0 and 1 inclusive, the utility functions above translate x into a *weighted utility* also between 0 and 1 inclusive. For the piece-wise, universal and existential quantification utility functions above, the specific translation is conditional upon x.

Currently, Tête-à-Tête defaults which utility function is used for each attribute. For example, the "Delivery Time" attribute is weighted with a "LinearPreferLowerUFN" utility function indicating that (all else being equal) the shopper prefers faster delivery times than slower delivery times and that this preference is weighted linearly. To help shoppers understand how the attribute is assessed, Tête-à-Tête visually reveals this weighting to the shopper through an appropriately shaped range control, as shown in Figure 16. A future enhancement to the system is to allow shoppers to choose their own

utility weightings by selecting one of the corresponding graphs shown in Table 16. User studies can help determine whether shoppers would benefit from this.

Evaluating Numeric Range Attributes

The numeric range attribute type is defined in Chapter 4. Briefly, the offer value (from the product offering) is a real number and the criterion (from the shopper's preferences) is a real number range between a lower bound and an upper bound as shown in Figure 25.
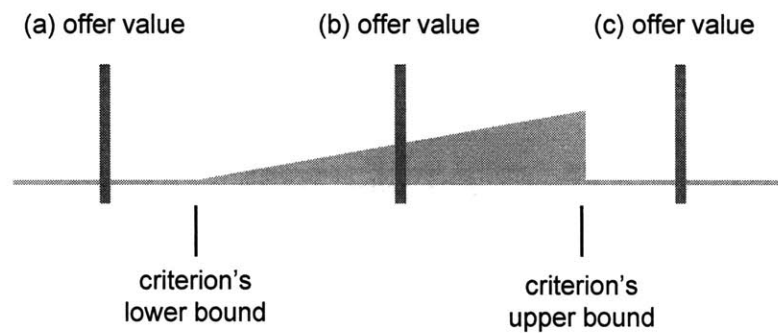
Figure 25: Evaluating a numeric range attribute

If the offer value is less than the criterion's lower bound as shown in Figure 25(a), then the attribute is tagged as being *out of range* and the attribute is assigned a weighted utility of 0. Likewise, if the offer value is greater than the criterion's upper bound as shown in Figure 25(c), then the attribute is tagged as being out of range and the attribute is assigned a weighted utility of 0. Otherwise, the offer value is somewhere between the criterion's lower and upper bounds inclusively, as shown in Figure 25(b). In this case, the next step for evaluating a numeric range attribute's utility is to normalize the offer value between 0 and 1 as follows:

$$x = \frac{\textit{offer value} - \textit{lower bound}}{\textit{upper bound} - \textit{lower bound}}$$

Equation 2: Normalizing a range attribute

This produces a linearly normalized utility, *x*, between 0 and 1 inclusive. Finally, translating *x* using the attribute's utility function (one of those found in Table 16) yields the attribute's weighted utility, *f(x)*, also between 0 and 1 inclusive.

Evaluating Discrete Range Attributes

The discrete range attribute type's offer value (from the product offering) is a token and the criterion (from the shopper's preferences) is an ordered range of tokens containing all valid offer values with a lower bound and an upper bound set to specific tokens. The first step for evaluating a discrete range attribute is to determine the *index* of the lower bound, offer value, and upper bound by looking up their respective token's index in the criterion's ordered range of tokens, as shown in Figure 26.
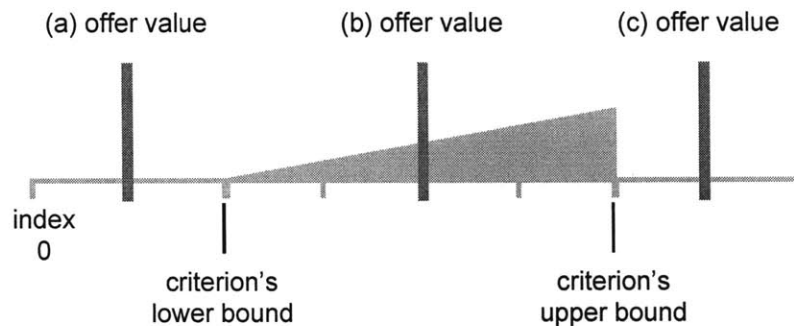
Figure 26: Evaluating a discrete range attribute

For example, a criterion may be comprised of the following ordered list of tokens: a, b, c, d, e, f, g, h, i, j, k. If the lower bound is set to 'c', the offer value to 'e', and the upper bound to 'k', then their respective (0-based) indexes are 2, 4, and 10. With these indexes,

evaluating a discrete range attribute becomes equivalent to evaluating a numeric range attribute continuing with Equation 2. The normalized utility for the example above would be:

$$0.25 = \frac{4-2}{10-2}$$

If a "LinearPreferLowerUFN" utility function is then applied to this normalized utility, the attribute's weighted utility would be:

$$0.75 = 1 - 0.25$$

Evaluating Discrete Set Attributes

The discrete set attribute type is defined in Chapter 4. Briefly, the offer value (from the product offering) and the criterion (from the shopper's preferences) are both an unordered set of tokens, as shown in Figure 27. The offer value tokens indicated by Figure 27(a) intersect with the criterion's tokens whereas those in Figure 27(b) do not.
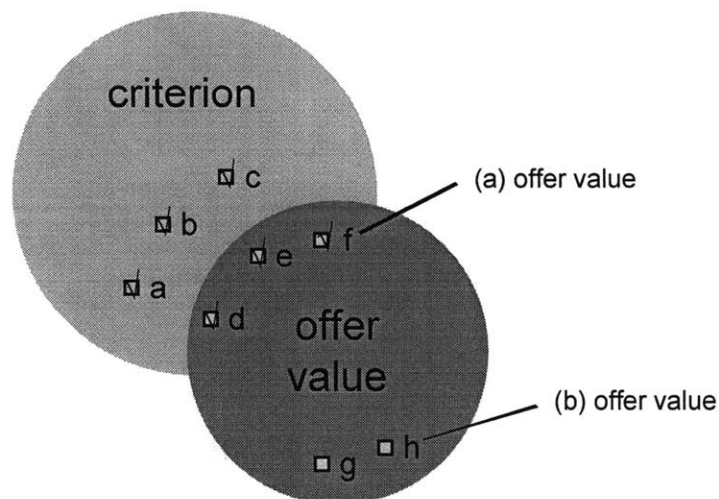
Figure 27: Evaluating a discrete set attribute

The discrete set attribute's normalized utility is the percentage of the criterion's set of tokens that are intersected with the offer value's set of tokens. For example, if the criterion's unordered set of tokens is a, b, c, d, e, f and the offer value's unordered set of tokens is d, e, f, g, h, then the attribute's normalized utility would be 0.50 since half of the criterion is satisfied by the offer value. Any utility function in Table 16 can weight this normalized utility accordingly. Of the these utility functions, however, "UniversalUFN", "ExistentialUFN", and "LinearPreferUpperUFN" are the most commonly used for discrete sets in Tête-à-Tête. Respectively, these help define an attribute where "Every checked item must be satisfied" (see Figure 18), "At least one checked item must be satisfied" (see Figure 19), or where "As many checked items as possible should be satisfied" (see Figure 22).

Evaluating Product Offerings

Once each attribute has been evaluated using the methods described above, then the product offering's active features can be evaluated. (Inactive features are not considered in the value assessment.) Each feature's normalized utility is determined by averaging the weighted utilities of its attributes. However, features are also weighted relative to one another. A feature's normalized utility is relatively weighted by using its (1-based) *position* (i.e., vertical orientation) in the list of preferences, as shown in the right panel of Figure 13. Shoppers can drag and drop feature modules above or below one another to express their relative importance. This, in turn, alters features' positions. Each feature's relative weight w_j is determined as follows:

$$w_j = f\left(\frac{position_j}{m+1}\right)$$

Equation 3: Evaluating the relative weighting w_j of feature j

where m is the number of active features and $f(.)$ is one of the utility functions listed in

Table 16. Tête-à-Tête currently defaults $f(.)$ to "LinearPreferLowerUFN". 1 is added to

m to always ensure that $w_j > 0$. A feature in the last position of 9 active features, for

example, would have a relative weighting w_j of:

$$0.1 = 1 - \frac{9}{9+1}$$

The normalized utility of feature j is multiplied by relative weighting w_j to determine its

weighted utility. Finally, after each feature's weighted utility is determined, these are

summed to determine the product offering's *raw* value assessment. This raw value

assessment is compared with the product offering's *ideal* value assessment (determined

by assuming a perfect weighted utility of 1 for each attribute) to determine the product

offering's *absolute* value assessment (as a percentage of the ideal). This absolute value

assessment is visually represented as a *value bar*, as shown in Figure 15. This evaluation

process is repeated for each new proposal.

Evaluating Counter-Proposals Efficiently

Computing the utilities of each proposal takes linear time based on the number of

attributes. A counter-proposal, however, represents only the changes of a previously

assessed product offering. Tête-à-Tête shopping agents take advantage of these previous

assessments by maintaining their partial results. Specifically, shopping agents remember

each product offering's raw value assessment v and weighted utilities $f_i(x_i)$ of each of its

attributes. A counter-proposal updates the normalized utilities x_I' of one or more

attributes. For each attribute change, the product offering's value is reassessed as

follows:

$$v' = v - \frac{wj}{nj} \sum_{i=1}^{nj} \left(fi(xi) - fi(xi') \right)$$

Equation 4: Evaluating a counter-proposal

Compared with the value assessment for proposals (see Equation 1), assessing counter-

proposals are relatively more efficient.

Value-based Online Shopping

Helping online shoppers make important value tradeoffs is more than about assigning a

score to a product. Shopping by value instead of by price allows shoppers to consider

and understand of how well product offerings satisfy the shopper's overall needs. Tête-à-

Tête's decision support module provides visual feedback at four levels of granularity to

help shoppers comprehensively understand the value that each product offering brings to

them.

Level 1: Understanding the Value of Attributes

When a single proposal is selected in Tête-à-Tête's middle panel, each of its offer values

is overlaid onto its respective criterion control in the right panel, as shown in Figure 28.
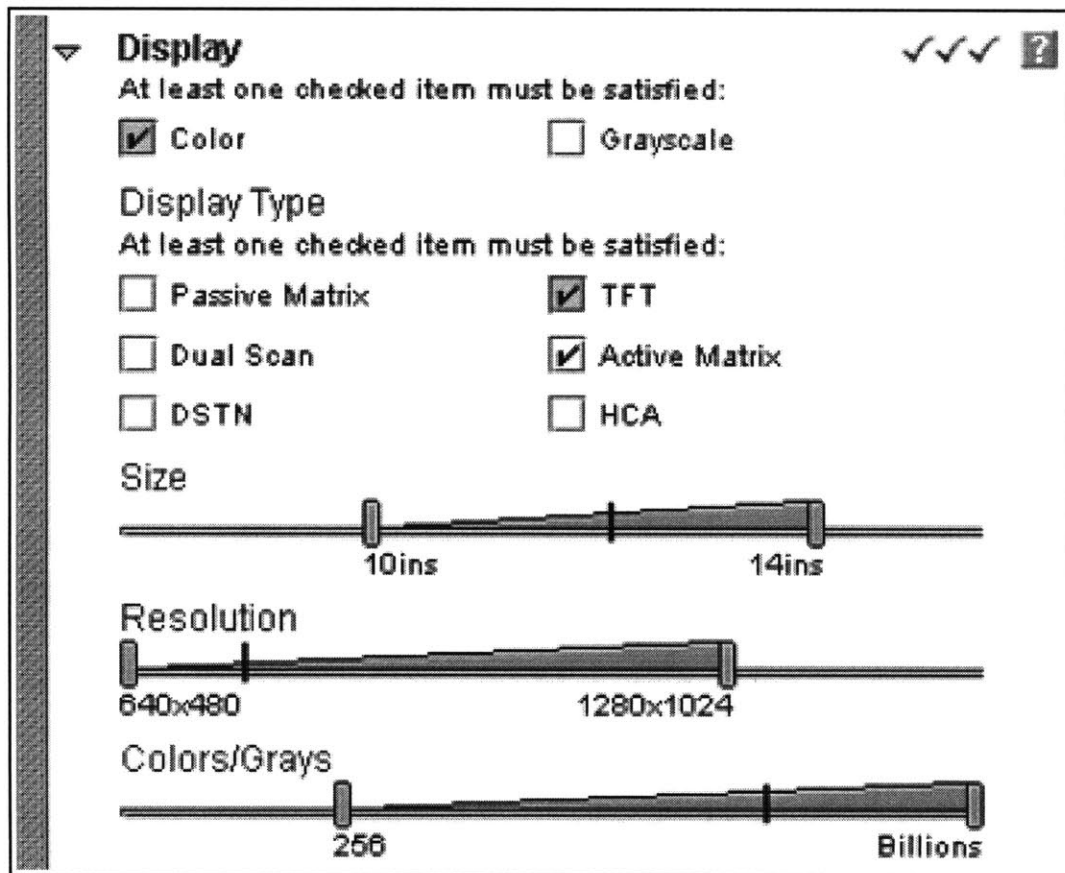
Figure 28: Visual feedback of offer values

Numeric range and discrete range offer values are overlaid as *offer value marks* directly on the range control. Discrete set offer values are indicated by the green coloring of the appropriate discrete set control's checkboxes. These visual feedback mechanisms make it clear which specific attributes are and are not being satisfied. This allows shoppers to assess product offerings at a fine level of detail (if they desire).

Level 2: Understanding the Value of Features
Tête-à-Tête also helps shoppers assess product offerings at the feature level. As shown in Figure 29, *criterion check marks* give shoppers a quick read on how well each feature stacks up with its respective preferences.
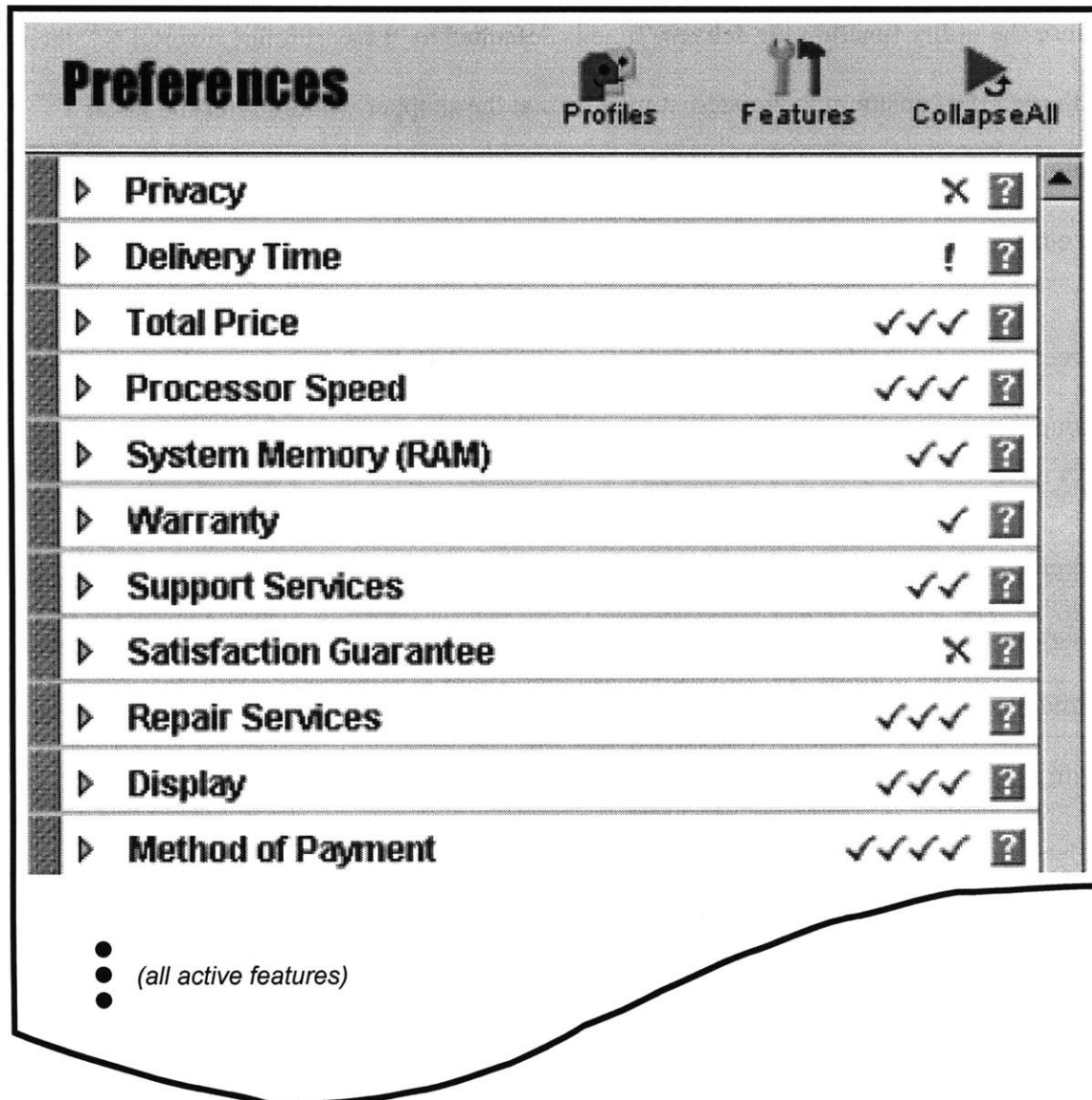
Figure 29: Visual feedback of feature assessments

The criterion marks for features perform several functions. If the mark is a red '!' or 'X', it means that at least one of the its attribute's offer values is *out of range* (i.e., it violates the attribute's criterion) and the attribute is therefore assigned a weighted utility of 0. However, if the mark is a red '!', an attribute's offer value may be out of range in a potentially beneficial way. For example, a "Delivery Time" attribute's lower bound may be set at 3 days, but the selected product offering may have a delivery time of 1 day.

Since the utility function for delivery time is defaulted to "LinearPreferLowerUFN", then although the constraint is violated, it may be that the shopper would prefer the faster delivery time.[29] The '!' mark alerts the shopper to this possibility in case she may want to change her preference accordingly.

Green criterion check marks help shoppers assess how well a feature's within-range offer values stack up against the feature's criterion. One green check mark indicates that, on average, the feature's offer values only minimally satisfy their respective criterions. Four green check marks, on the other hand, indicates that, on average, the feature's offer values greatly satisfy their respective criterions. This visual feedback, situated within the same regions as where the preferences for the features are solicited, provides the shopper with a convenient and clear understanding of how well the selected product offering stacks up against each feature.

Level 3: Understanding the Value of Product Offerings

A product offering encompasses the full value of a product. However, the assessment of this value is a personalized decision based on individual needs. For complex products, this value assessment embodies tradeoffs among multiple objectives. To help shoppers understand these tradeoffs, Tête-à-Tête visually represents the assessed values of product offerings as *value bars* in the middle panel of the user interface, as shown in Figure 30.

---

[29] Of course, the shopper may have expressed a preference for a minimum three day delivery time because she will be unable to receive the product before then.
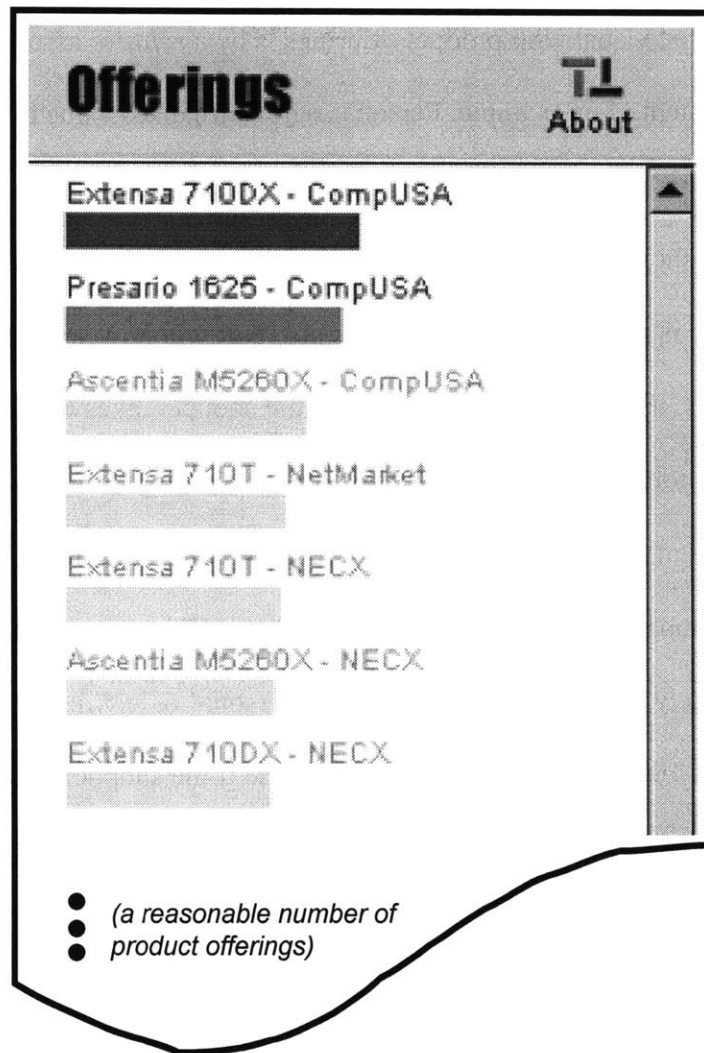
Figure 30: Visual feedback of product offering assessments

The length of these value bars represent how well the product offerings meet the expressed preferences of the shopper. In other words, Tête-à-Tête's value bars represent *absolute* assessments of a product offering's value. A long value bar indicates that the product offering greatly satisfies the expressed preferences, and a short value bar indicates that the product offering only minimally satisfies the expressed preferences. Value bars are ordered in the middle panel based on these absolute value assessments.

An alternative way of visualizing product offerings is by a *relative* assessment rather than an absolute assessment. For example, PersonaLogic normalizes value assessments relative to the one with the greatest value. This gives a score of 100% to the greatest valued product, as shown in Figure 4. Although perhaps initially inspiring to see a score of 100%, this score is misleading in that it does not represent how well the product meets the shopper's needs. For instance, if a PersonaLogic shopper takes the time to change her answers to one or more "deep interview" questions, she may not see any change in PersonaLogic's list of results since they all are relatively scored. More importantly, it is possible that the shopper expressed highly constrained preferences resulting in products that do not adequately meet her needs. However, PersonaLogic will still score poor product matches as 100%. This is misleading because if the shopper purchases such a product, she will likely be disappointed.

In Tête-à-Tête, the value bars of product offerings that have at least one attribute *out of range* are colored gray (as opposed to their usual purple color). This helps the shopper quickly differentiate product offerings that fully satisfy their expressed needs to those that do not. In many scenarios, proposed product offerings satisfy all of a shopper's preferences. However, in other scenarios, although only a few attributes may be out of range, the overall assessment of the product offering may be very favorable and worth considering. A typical database query would miss these favorable product offerings due to the infamous "near miss" problem. For example, if a shopper expresses a price range between $2000 and $4000 using any of today's price comparison systems (e.g., Jango and Junglee [11]), these systems will filter out products with prices of $1999 and $4001,

either of which may be worth considering. Tête-à-Tête allows shoppers to consider these potentially favorable outliers.

Level 4: Understanding the Emergent Patterns from Negotiations

Tête-à-Tête's interaction model and feedback mechanisms allow shoppers to visualize and understand the space of product offerings in a way not possible before. For example, the vertical layout of Tête-à-Tête's value bars reveals a pattern that emerges during negotiations. The absolute and relative sizes and colors of the value bars tell a story as preferences are changed and product offerings proposed. A plummeting of the value bars' sizes indicates overly constrained preferences. A burst of color from gray portrays a successful compromise in negotiations. These emergent properties, unique to Tête-à-Tête, can subsequently help shoppers better understand their needs and their opportunities.

Stereotyping System

When a shopper begins shopping with Tête-à-Tête, it may be overwhelming and discouraging to enter all of her preferences "from scratch." For example, in the laptop computer domain, there may be 40 product and merchant features which the shopper may feel obliged to consider if immediately presented with all of them. To help bootstrap the shopping experience, Tête-à-Tête's decision support module is complemented with a *stereotyping system*. As discussed in Chapter 3, upon selecting a specific product domain (e.g., "notebook computers"), the shopper is asked one question, "Which of the following profiles best characterizes the person you are shopping for?" This is proceeded with a list of *stereotypes* pertinent to the selected product domain. For example, in the notebook

computers domain, these profiles may be Power User, Budget Conscious, Average User, and Road Warrior as shown in Figure 12.

Each presented stereotype is associated with information that characterizes the stereotype. This information includes the most important features that someone of that stereotype would care about, the preferences for each of these features, and the relative importance of these features. This information is used to default the shopper's preferences by setting which features are active, setting the criterion for each feature's attributes, and setting the position of each feature in the right panel, respectively. The shopper's shopping agent broadcasts these default preferences to sales agents which, in turn, respond with relevant proposals, as discussed in Chapter 4. As the shopper refines the default preferences to best meet her personal shopping needs, she is immediately rewarded with better tailored proposals.

Currently, Tête-à-Tête uses a static database of several "notebook computer" stereotypes determined through a market segmentation exercise. The plan for the stereotyping system, however, is to have the stereotypes be dynamically defined based on how shoppers of a self-selected stereotype actually define their preferences while shopping with Tête-à-Tête. A more advanced version of this stereotyping system is one that identifies significant clusters of shoppers and automatically generates the list of stereotypes (with a human only needing to assign a label to each cluster). These ideas are discussed in Chapter 8.

Qualitative Analysis

An important characteristic of MAUT is that it assumes a rational decision maker with limitless resources. Unfortunately, these assumptions rarely hold when applying this "perfectly rational" economic perspective to real-world tasks. For example, it may not be possible to determine the probability distributions of the outcomes of future uncertain events (e.g., the demand for a product offering), there may be limited computational resources (e.g., time and memory), there may be a mismatch between a human's decision making processes with the need in MAUT to express preferences in a relatively rigid, "rational" manner, and there is often the general difficulty of formulating a complex problem into comprehensive and measurable attributes. Tête-à-Tête addresses these limitations of applying multi-attribute utility theory to real-world tasks to varying degrees for both consumers and merchants.

1. VALUE TRADEOFFS UNDER CERTAINTY

Tête-à-Tête does not consider probabilities directly in its value assessments. Rather each product offering is measured against a shopper's preference under certainty that the information provided is accurate (but not necessarily complete) and that merchants will flawlessly execute their responsibilities of the transaction as defined by their proposals. However, to help shoppers combat uncertainties surrounding what to buy (e.g., uncertainties of reliability, specification accuracy, and interoperability) and which merchant to buy from (e.g., uncertainties of privacy, availability, and delivery time), Tête-à-Tête can be augmented with a reputation brokering service to add reputation features to product offerings for their enhanced evaluations (discussed in Chapter 8).

2. INCREMENTAL AND DISTRIBUTED VALUE ASSESSMENTS

Tête-à-Tête manages constraints of limited computational resources by having facilities to reassess product offerings incrementally by only recomputing changes in product offerings or preferences. This yields a linear-time complexity when evaluating merchants' counter-proposals. Tête-à-Tête also embodies a scalable architecture for assessing numerous product offerings by distributing the burden of value assessments among all self-interested agents as discussed in Chapter 6.

3. WELL-BALANCED USER INTERFACE FOR EXTRACTING PREFERENCES

Expressing the requisite preferences for performing accurate MUAT value assessments is often difficult and tedious [31]. Tête-à-Tête assists shoppers in accurately, quickly, and effortlessly expressing their preferences in two ways. First, Tête-à-Tête incorporates a stereotyping system to default the shopper's preferences. Second, Tête-à-Tête's shopping interface provides simple controls to refine these preferences incrementally. These include two-sided numeric range and discrete range controls as well as discrete set checkboxes. Additionally, to express relative importance among features, a shopper simply drags and drops feature modules above or below other features indicating the shopper's preference through the features' relative vertical orientation. Although MAUT preferences may be captured using more expressive mathematical functions, Tête-à-Tête's interaction model and supporting decision support module propose a balance among expressiveness, completeness, accuracy, and ease of use.

Furthermore, alternative mechanisms can be employed to extract preferences from shoppers less explicitly. Chapter 8 discuss two such mechanisms: SuperHelp and Conjoint Shopping.

4. COMPREHENSIVE AND MEASURABLE ATTRIBUTES

Finally, Tête-à-Tête formalizes the problem of assessing the value of complex product offerings into comprehensive and measurable attributes by mapping product and merchant attributes one-to-one with shopper's preferences. With this formalism in place, high-order objectives may be constructed that map to these fundamental attributes in other ways. Tête-à-Tête's stereotyping system is one such example. Two other examples, SuperHelp and Conjoint Shopping, are discussed in Chapter 8.

Summary

This chapter detailed how Tête-à-Tête's decision support module helps online shoppers make value tradeoffs among the proposals from sales agents. Combined with several levels of visual feedback, this decision support module helps shoppers understand their needs and how these needs are used to evaluate and influence the proposed product offerings. However, having shopping agents assess proposals is only half of the story. On the other side of the negotiation table are multiple sales agents that are each burdened with determining which product offerings to propose. This challenge is the subject of the next chapter.

6   Product Customization
Based on Distributed Constraint Satisfaction

Shopping agents are burdened with *assessing* the value of product offerings from sales

agents. Imagine that shopping agents were also burdened with *configuring* each product

offering to best meet the needs of their owners. For example, for a shopping agent to

assess a product offering that can be configured using one of three different delivery

options (e.g., overnight delivery for $20, three day delivery for $10, and 14 day delivery

for $2), the shopping agent could enumerate each of the three configurations and assess

them individually. For the example above, assessing the product offering would only

require three times as much work than if the product offering was already configured

with one of the delivery options.

However, if the product offering was highly configurable, the number of configurations

would grow exponentially with each configuration option. In the case of notebook

computers, for example, if delivery time, warranty length, system memory, hard drive

size, display size, modem speed, network card, and processor speed were each

configurable with roughly 3 different options for a specific base product (e.g., Dell's

Latitude CP [49]), then the number of configurations the shopping agent would need to

consider for this one product offering would be $3^8 = 6,561$! Furthermore, if each sales

agent was permitted to send proposals for every notebook computer its merchant sold, the shopping agent would soon be overwhelmed with the computational burden of assessing the value of thousands upon thousands of product configurations.

Distributed Constraint Satisfaction

Our experiments [...] illustrate that solving constraint satisfaction problems in a loosely-coupled, distributed network is worthwhile if the local constraint satisfaction problems of each agent are large grained and nearly independent.

– *M. Yokoo, E. Durfee, T. Ishida, and K. Kuwabara [50]*

There are a number of approaches for combating the configuration problem described above. One approach is to provide the shopping agent with more sophisticated configuration algorithms for faster performance. Constraint satisfaction [50] and linear programming [51] techniques are useful for efficiently solving configuration problems such as this one. However, this approach is still not scalable. An alternate approach, and the one adopted by Tête-à-Tête, is to distribute the burden of configuring each product offering from the shopping agent to the sales agents. In particular, Tête-à-Tête requires that each sales agent configure their own product offerings and only send n of these to the shopping agent (where n can be dynamically set at run-time). This reduces the shopper's consideration set to a reasonable number of only the best product offerings. In this sense, Tête-à-Tête can be viewed as a *distributed constraint satisfaction* system. As supported by research in distributed constraint satisfaction (see the quote above), Tête-à-Tête's distributed, loosely-coupled design makes it a highly scalable system.

Distributed constraint satisfaction finds its roots in distributed artificial intelligence (DAI) research and is often perceived as requiring homogeneous agents that cooperate without self-interest [50]. However, this need not be the case. If the heterogeneous, self-interested agents have incentives to cooperate, then distributed constraint satisfaction techniques may apply. In Tête-à-Tête, self-interested sales agents are indeed incented to cooperate and share the computational burden of configuring the optimal product offerings for their customers.

Assuming Tête-à-Tête shoppers represent sizable market power, merchants' sales agents are incented to cooperate with shopping agents for the opportunity of their patronage. Perhaps more importantly, accepting the configuration burden allows merchants to control which of their product offerings their customers should consider. Instead of a shopping agent deciding which products to extract from the merchant's online catalog, each sales agent is now able to control this access to its owner's benefit. For example, merchants may want consumers to consider their current promotions even though they may not stack up as well as other product offerings that a shopping agent may have otherwise suggested to its owner. Similarly, merchants may want their inventory levels and profit margin to influence which product offerings get offered to the shopper. In short, the configuration of products offerings, although a computational burden, is an opportunity for *access control* and *customization*.

Tête-à-Tête's Product Offering Customizer

Tête-à-Tête's sales agents currently use a computationally *inefficient* product offering customizer.[30] Nevertheless, this customizer demonstrates the concept and potential benefit of distributing the computational burden of customizing product offerings among multiple sales agents. Specifically, Tête-à-Tête's product offering customizer is an extension of the same decision support module used by shopping agents to assess the value of fully configured product offerings. This allows sales agents to accurately determine which of their product offerings best meet the needs of each shopper. However, merchants have their own needs such as profit margin. Although Tête-à-Tête is designed to support heterogeneous sales agents (as discussed in Chapter 7), each Tête-à-Tête sales agent currently uses the same general heuristics for configuring proposals.

1. Enumerating each Configuration

First, each sales agent maintains a local list of potential product offerings, one per base product. For example, if a merchant sells ten different notebook computers, then the sales agent would maintain a local list of ten potential product offerings (henceforth PPOs).[31] In general, Tête-à-Tête's product offering customizer allows any numeric range or discrete range attribute to have one or more configurable options. In the notebook computer example, the configurable attributes include delivery time, warranty length, system memory, hard drive size, display size, modem speed, network card, and processor

---

[30] The reasons for this include a focus on the value assessment task over the configuration task as well as a lack of time to develop a more efficient configurator. Furthermore, Tête-à-Tête has been designed as a heterogeneous agent system with each merchant owning and operating its own sales agent configurator, such as Trilogy's Selling Chain™ [53].

[31] This heuristic was chosen arbitrarily. It is possible, for example, that a certain base product may have more than one configuration that are each assessed as having a better value than an alternative base product. Tête-à-Tête supports this latter approach as well. However, for purposes of demonstration, this former approach was adopted.

speed. The number of options that each of these attributes have depends upon the product and the merchant. For example, Dell's "Latitude CP" is a *base product* that can be configured along each of the product attributes above.[32] Compaq's "Presario 1625", however, is an already fully configured base product. However, all merchants tend to support a variety of delivery and extended warranty options. This means that all PPOs are at least minimally configurable. Upon initialization, each sales agent enumerates all of the possible configurations for each PPO.[33]

2. Determining the Optimal Configurations

When a sales agent receives the initial *criteria* performative from a shopping agent, the sales agent optimizes each PPO. A PPO is optimized by first assessing each of its enumerated configurations based on the shopper's preferences, the merchant's preference for profit, and the product's inventory level. In order to assess a configuration, the options of that configuration must propagate its costs to the total price attribute. For example, an option for overnight delivery will accrue added costs that must be included in the configuration's total price. Tête-à-Tête's customizer also keeps track of each option's actual cost and offer price along with the base product's actual cost (to the merchant) and its offer price. For example, an extra 32 MB of RAM may cost a merchant $50 but the merchant may prefer to charge $100 for the service of adding the memory to a base product due to the merchant's extra labor costs and profit needs.

The configuration's profit is equal to the summation of its offer prices minus the summation of its actual costs. This profit is included in the configuration's value

---

[32] A "base product" is a specific marketed product model that may or may not be configurable.
[33] As explained earlier in this chapter, this algorithm is inefficient for highly configurable products.

assessment as a numeric range attribute whose feature is positioned by the merchant relative to the other features. This approach allows merchants to assign arbitrary importance to their profit needs when simultaneously attempting to satisfy the needs of its customers. A further attribute interaction is that if a product is out of stock (i.e., has a 0 inventory level), then the configuration's delivery time attribute must include when the merchant expects the product to be in stock and its ultimate delivery time.

3. Proposing the Optimal Configurations

After all of a PPO's configurations have been assessed, each PPO is set to the configuration with the greatest assessed value. Once all PPOs are optimized in this way, the sales agent identifies the best *n* PPOs (where *n* is the value of the maxproposals field in the criteria performative), and sends these product offerings as a response to the shopping agent as *proposal* performatives. Note that these proposal performatives define all of the product and merchant features of the product offering, not just the configurable options.

4. Reconfiguring each Product Offering

After the initial proposals are sent, the sales agent may receive *critique* performatives from the shopping agent. The sales agent updates its local representation of the shopper's preferences and reevaluates its list of PPOs. If a PPO that was sent earlier was reconfigured due to the critique and the PPO is still one of the best *n* PPOs, then the changes in the product offering's configuration are sent as a *counter-proposal* performative to the shopping agent. If a PPO is valued greater than a previously sent product offering, then the sales agent sends a *withdraw-proposal* performative to remove the outdated product offering from consideration and, in addition, sends a new *proposal*

performative defining the product offering that has a greater value than the one it replaced. As discussed in Chapter 4, integrative negotiations occur in this fashion until the shopper withdraws or accepts a sales agent's proposal.

Qualitative Analysis

Tête-à-Tête does *not* include a viable customizer for highly configurable products. The search space is too large to use its naive "generate and test" approach. However, Tête-à-Tête *does* propose a distributed constraint satisfaction approach to online shopping with sales agents accepting the burden and opportunity for customizing their own product offerings. Tête-à-Tête's customizer helps demonstrate this concept. Also, for minimally configurable product offerings, Tête-à-Tête's customizer is sufficient.

Tête-à-Tête sales agents must consider several "special" attributes when configuring each product offering. For instance, there may be several delivery options which will affect the proposed delivery time. However, if the product is out of stock, then the merchant must adjust the delivery time accordingly, perhaps to when the next delivery of the product is expected. These uncertainties may require that the sales agent assess product offerings with probability distributions for uncertain outcomes.

Perhaps the most interesting attributes, however, are profit and total price. Each of these are dependent on how other attributes are configured (unlike all other attributes). For example, configuring the system memory with more RAM will require an adjustment in the total price (and perhaps profit). This characteristic of the sales agent's configuration task makes it a special type of constraint satisfaction problem. As suggested in Chapter

8, Tête-à-Tête's product offering customizer could be significantly improved through linear programming techniques such as Lagrangian relaxation [51].

Tête-à-Tête uses the above profit and inventory heuristics to demonstrate the incorporation of each merchant's need for profit into the decision of which product offerings to propose. Of course, other heuristics (or no heuristics!) can be employed to help facilitate the transaction. For example, BroadVision [52] allows merchants to express rules for recommending products to customers. Trilogy's Selling Chain [53] is a product configurator that could also be used in place of Tête-à-Tête's configurator. As with Tête-à-Tête's shopping agent, these sales systems may never be able to fully capture the preferences of their owners. Nevertheless, they may help facilitate more mutually beneficial transactions than a shopping system with only self-interested shopping agents.

Summary

Tête-à-Tête embodies an approach to distribute the burden of customizing product offerings among the sales agents incented to do so. This customization is an opportunity for merchants to tailor their offerings to the needs of their customers while also considering their own needs (such as for profit). Tête-à-Tête's product offering customizer demonstrates this concept. It is based on Tête-à-Tête's decision support module and supports Tête-à-Tête's integrative negotiation interaction model. The next chapter discusses the implementation details of each of Tête-à-Tête's primary software components.

7 Implementation
Java Agents with XML-based Negotiation

The design of Tête-à-Tête started with a set of requirements including the need to provide

a comprehensive online shopping experience that benefits both consumers and

merchants. This analysis was followed by a series of graphic design mockups including

specifications for the interaction model – i.e., the general manner and style of human-

computer interaction. What emerged from these specifications was the formulation of

Tête-à-Tête's current integrative negotiation protocol and decision support modules.

Tête-à-Tête's overall architecture is shown in Figure 31.
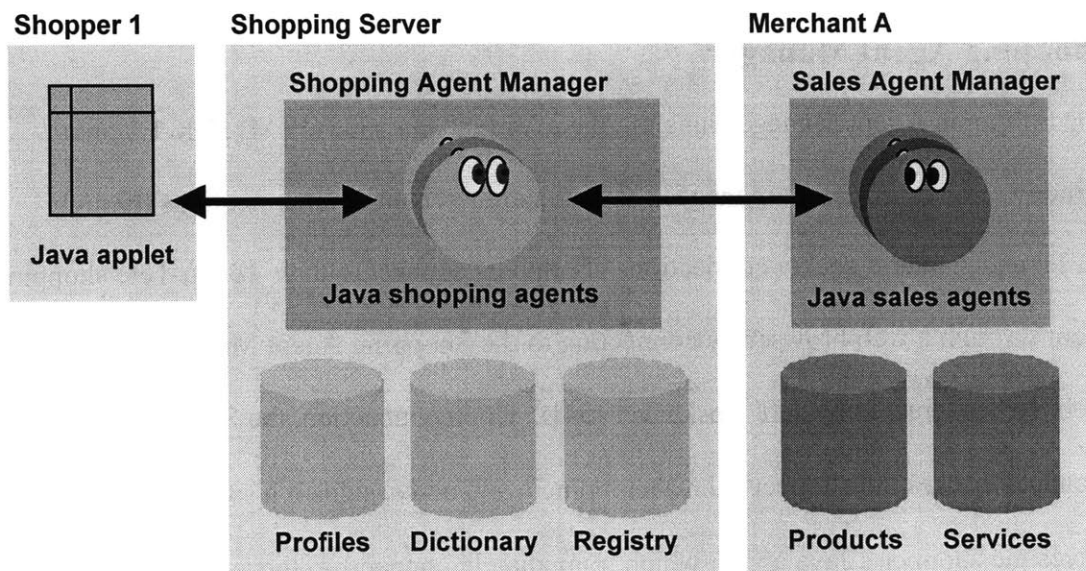
Figure 31: Tete-a-Tete's architecture supports multiple shoppers and merchants (not shown)

The main components of the Tête-à-Tête system are the Shopping Agent Manager, the Shopping Interface, the Sales Agent Manager, the Communication Manager (indicated by the arrows in Figure 31), and several databases: Registry, Dictionary, Profiles, Products, and Services.

Tête-à-Tête was designed for a heterogeneous agent environment. Although Tête-à-Tête's sales agents are currently all homogenous, they each run in separate Java Virtual Machines and access their own unique database of products and services. This means that sales agents can reside anywhere and be a part of the Tête-à-Tête shopping system as long as they can communicate via sockets and "speak" the XML-based performatives of Tête-à-Tête's bilateral argumentation (see Chapter 4). Other then these constraints, the implementations of Tête-à-Tête's sales agents can vary dramatically. Once entered into Tête-à-Tête's Registry database under one or more product domains, a merchant's sales agent becomes part of Tête-à-Tête's shopping system.

Shopping Agent Manager

The Shopping Agent Manager runs in a Java Virtual Machine (JVM) JDK 1.1 on a Windows NT machine. It manages shopper and merchant communications through HTTP and standard socket connections. Shoppers connect to their Tête-à-Tête shopping agent through a web browser by connecting to the Shopping Agent Manager's Acme Java Web Server (created by Jeff Poskanzer [54]). Upon connection, the Shopping Agent Manager creates the shopper's Java shopping agent and assigns it a socket port, then serves the shopper a Java applet which immediately connects to the shopper's shopping

agent via a socket connection on the assigned port. (The Java applet can be viewed as the shopping agent's user interface.)

Next, the shopping agent retrieves a list of merchants that sell notebook computers from the Registry database and sends a "request to connect" message to each of their respective Sales Agent Managers. Similar to the Shopping Agent Manager, upon connection requests, each Sales Agent Manager creates a Java sales agent and assigns it a socket port, then sends the port number back to the shopping agent which, in turn, disconnects from the Sales Agent Manager and reconnects to the newly created sales agent on the assigned port. At this point, communication from the Java applet through the shopping agent to each sales agent has been established.

Since Tête-à-Tête currently only contains a notebook computer ontology and associated data, the shopping agent's first interaction with the shopper is by asking her to choose from a list of profiles (or stereotypes). This list of stereotypes is retrieved from the Profiles database. After the shopper selects a stereotype, the shopping agent retrieves all of the features for the notebook computer domain from the Dictionary database and sends these to the Java applet over the socket connection using a proprietary protocol. Next, the shopping agent accesses the default preferences for the selected stereotype from the Profiles database and initializes the shopper's Java applet accordingly. Simultaneously, the shopping agent packages these default preferences as a criteria performative and broadcasts it to each sales agent thus commencing negotiations.

Shopping Interface

As shown in Figure 2, Tête-à-Tête's shopping interface uses a Web browser with two frames. The left frame comprises the system's "left panel" and is used to display HTML product offering specifications and comparisons. The right frame contains Tête-à-Tête's Java applet that itself is split into two panels, the "right panel" for soliciting shopper preferences (and giving special feedback), and the "middle panel" for listing value bars each corresponding to a product offering from a sales agent. The Java applet is based on JDK 1.1 and Sun Microsystems' Swing interface elements plus a new range control interface element created for Tête-à-Tête. Because today's Web browsers do not support Swing classes natively, Tête-à-Tête currently requires a separate, one-time download of the Swing classes to be placed in the appropriate folder.

When shoppers adjust the Java controls in the right panel to express preferences, these changes are sent the shopper's shopping agent which repackage the changes as critique performatives and broadcast these to the sales agents. Responses from sales agents are performatives that the shopping agent uses to maintain a list of product offerings. Product offerings are reassessed upon each response from a sales agent and their assessed values are sent to the Java applet which displays them as value bars in the middle panel. If one value bar is selected, the Java applet requests its associated product offering's offer values and weighted utilities from the shopping agent for display as feedback in the right panel. Also, the Java applet instructs the left panel to display the product offerings specifications. The left panel then requests a specific Web page from the Shopping Agent Manager's Acme Web Server. This calls a Java servlet which dynamically

generates the product offerings specifications by filling in the fields of an HTX (HTML extension) template.

Sales Agent Manager

As discussed earlier in this chapter, Tête-à-Tête is designed for heterogeneous sales agents to participate in the Tête-à-Tête marketplace. However, Tête-à-Tête does currently host its own sales agents for demonstration purposes. Each of these sales agents partially represents a real-world merchant and are designed as follows.

The Sales Agent Manager creates a Java sales agent to manage each connection to a shopping agent (as described above). The socket connection uniquely identifies the shopper throughout a shopping session. This unique identification is necessary in order for Tête-à-Tête to support anonymous shopping. Upon creation, a sales agent connects to its Products and Services databases and generates a complete list of product offerings, one per product. It then disconnects from the databases. (A future version may access the databases more regularly to capture any changes.) When the sales agent receives criteria and critique performatives, it maintains a list of these preferences and uses them to configure and assess its list of product offerings. The best product offerings are sent as proposal or counter-proposal performatives. If exceeding the permitted limit of proposals, the sales agent will send withdraw-proposal performatives to withdraw less promising product offerings from the shopper's consideration.

Communication Manager

Communication between the Java applet shopping interface and the shopping agent uses the same socket protocols provided by the Communication Manager tools as the communication between the shopping agent and each sales agent. However, the protocol layered on top of the socket connection differs. The protocol for coordinating the Java applet and the shopping agent uses proprietary but efficient messages. The purpose of these messages is mostly to keep the shopping agent informed of changes in the shopper's preferences and for displaying results to the shopper.

The protocol for negotiations between the shopping agent and sales agents is based on XML (extensible markup language), a meta-language based on SGML (structured general markup language) [13]. XML was chosen as the medium for Tête-à-Tête's inter-agent negotiation protocol because it is sufficiently expressive, yet simple to understand. Furthermore, XML is gaining support as a potential HTML replacement and the meta-language for the exchange of business-to-business documents [55]. This excitement has yielded numerous software tools for manipulating XML messages. For example, Tête-à-Tête shopping agents and sales agents use Microsoft's free XML parser to encode and decode their XML performatives. The DTD (data type definition) for the criteria and critique request performatives defined in Chapter 4 are defined as follows:

```
<!-- This DTD is part of the open protocol for MIT Media Lab's        -->
<!-- Tete-a-Tete (T@T) Project.  More information can be found at:     -->
<!-- http://ecommerce.media.mit.edu/tete-a-tete/                       -->
<!--                                                                   -->
<!-- This request DTD defines the message sent from a T@T shopping     -->
<!-- agent to a sales agent.  A request can either be criteria or      -->
<!-- a critique of an earlier proposal.                                -->
<!--                                                                   -->
<!-- @version  1.2, 31 July 1998                                       -->
<!-- @author   Robert H Guttman                                        -->
<!-- @author   Iddo Gilon                                              -->

<!ELEMENT request (userid,requestid,domain?,maxproposals?,(criteria|critique))>

<!ELEMENT userid (#PCDATA)>
<!ELEMENT requestid (#PCDATA)>
<!ELEMENT domain (#PCDATA)>
<!ELEMENT maxproposals (#PCDATA)>

<!ELEMENT criteria (ufn,category*)>
<!ELEMENT critique (ufn?,category*)>

<!ELEMENT category (position?,(numeric_range|discrete_range|discrete_set)*)>
<!ATTLIST category name CDATA #REQUIRED>
<!ATTLIST category active (true|false) "true">

<!ELEMENT numeric_range (ufn?,lower_bound?,upper_bound?)>
<!ATTLIST numeric_range name CDATA #REQUIRED>
<!ATTLIST numeric_range units CDATA #REQUIRED>
<!ELEMENT discrete_range (ufn?,lower_bound?,upper_bound?,item*)>
<!ATTLIST discrete_range name CDATA #REQUIRED>
<!ELEMENT discrete_set (ufn?,(additem|removeitem)*)>
<!ATTLIST discrete_set name CDATA #REQUIRED>

<!ELEMENT ufn (#PCDATA)>
<!ELEMENT position (#PCDATA)>
<!ELEMENT lower_bound (#PCDATA)>
<!ELEMENT upper_bound (#PCDATA)>
<!ELEMENT item (#PCDATA)>
<!ELEMENT additem (#PCDATA)>
<!ELEMENT removeitem (#PCDATA)>
```

Figure 32: Tête-à-Tête's XML-based criteria and critique performatives (request.dtd)

Similarly, proposal, counter-proposal, and withdraw-proposal performatives are encoded as XML messages. The DTD for these response messages are defined as follows:

```
<!-- This DTD is part of the open protocol for MIT Media Lab's      -->
<!-- Tete-a-Tete (T@T) Project.  More information can be found at:   -->
<!-- http://ecommerce.media.mit.edu/tete-a-tete/                     -->
<!--                                                                 -->
<!-- This response DTD defines the message sent from a T@T sales     -->
<!-- agent to a shopping agent.  The response can either be one or   -->
<!-- more proposals, counter-proposals, or proposal withdrawals.     -->
<!--                                                                 -->
<!-- @version  1.2, 23 July 1998                                     -->
<!-- @author   Robert H Guttman                                      -->
<!-- @author   Iddo Gilon                                            -->

<!ELEMENT response
(merchantid,requestid,note?,(proposal|counter_proposal|withdraw_proposal)*)>

<!ELEMENT merchantid (#PCDATA)>
<!ELEMENT requestid (#PCDATA)>
<!ELEMENT note (#PCDATA)>

<!ELEMENT proposal
(proposalid,manufacturerid,productid,productname,imageurl,category+)>
<!ELEMENT counter_proposal (proposalid,category+)>
<!ELEMENT withdraw_proposal (proposalid+)>

<!ELEMENT proposalid (#PCDATA)>
<!ELEMENT manufacturerid (#PCDATA)>
<!ELEMENT productid (#PCDATA)>
<!ELEMENT productname (#PCDATA)>
<!ELEMENT imageurl (#PCDATA)>

<!ELEMENT category (numeric_range|discrete_range|discrete_set)+>
<!ATTLIST category name CDATA #REQUIRED>

<!ELEMENT numeric_range (value)>
<!ATTLIST numeric_range name CDATA #REQUIRED>
<!ATTLIST numeric_range units CDATA #REQUIRED>
<!ELEMENT discrete_range (value)>
<!ATTLIST discrete_range name CDATA #REQUIRED>
<!ELEMENT discrete_set (additem|removeitem)+>
<!ATTLIST discrete_set name CDATA #REQUIRED>

<!ELEMENT value (#PCDATA)>
<!ELEMENT additem (#PCDATA)>
<!ELEMENT removeitem (#PCDATA)>
```

Figure 33: Tête-à-Tête's XML-based proposal, counter-proposal, and withdraw-proposal performatives (response.dtd)

The intent is to publish the above XML DTDs to create an open system. A possibility is to also open up the Registry database to allow merchants to register their own hand-crafted sales agents and become a part of the Tête-à-Tête shopping system.

Databases

Tête-à-Tête databases are either Microsoft SQL Server databases or flat files. The Dictionary database defines the ontologies for each product domain. These ontologies are defined as a set of product and merchant features using the numeric range, discrete range, and discrete set attributes discussed in Chapter 4. The Profiles database uses these ontologies to define the criterion for each stereotype. For each product domain, the Registry database lists the IP address and port of each Sales Agent Manager whose merchant sells products within that domain. Each sales agent has access to its owners database of Products and Services. These are used to construct product offerings.

Java Packages

Tête-à-Tête is comprised of several Java packages:

Package	Description
tete.dbi	a convenient SQL database abstraction
tete.comm	the socket communication package
tete.data	the core data structures including attributes and proposals
tete.maut	the decision support module (based on *data* package)
tete.neg	translates *maut* structures into XML-based performatives
tete.config	support to configure proposals (based on *neg* package)
tete.agent.shop	defines shopping agent manager and shopping agents
tete.agent.sale	defines sales agent manager and sale agents

Table 17: Tête-à-Tête's Java packages

Summary

Tête-à-Tête has a distributed architecture for supporting heterogeneous agents. Tête-à-Tête shopping agents are Java-based and execute within Tête-à-Tête's shopping system. They communicate with sales agents using XML-based performatives over standard socket connections. Sales agents may reside anywhere, be written in any language, and execute on any platform as long as they can communicate over sockets and "speak" Tête-à-Tête's open, XML-based performatives (defined in Chapter 4).

8 Conclusion
Evaluation Suggestions and Future Directions

This thesis explores the problems of today's online shopping environment, and in particular, the contention between lowering consumer search costs and differentiating merchants' product offerings. The approach taken to resolve this contention has been predominantly from the perspective of a consumer in the context of a shopping experience. As discussed in Chapter 1, taking a consumer's perspective is appropriate because the problem of a merchant attempting to differentiate itself in the marketplace rests in how the merchant is perceived by consumers. Invariably, this perception will critically influence the shopper's merchant brokering decision (see Table 1).

Tête-à-Tête helps merchants influence consumers' merchant brokering decisions by providing them a means to differentiate their product offerings along dimension other than just price, in particular, policies and value-added services. Through integrative negotiations of highly expressive proposals, Tête-à-Tête allows merchants to customize their product offerings in a manner that best meets the needs of their customers and satisfies their own needs (such as for profit). Furthermore, by tightly integrating the product brokering and merchant brokering stages of the online shopping process into a single decision support module, Tête-à-Tête allows merchants to influence consumers'

merchant brokering decisions earlier in the buying process through a merchant's wider selection or better customization of products.

Fixing the problems with today's online shopping crosses the boundaries of many disciplines. As such, this thesis challenges today's technology trends and business practices in order to create a better environment for both consumers and merchants. Tête-à-Tête holistically embodies this opportunity. However, to assess the viability of these ideas, Tête-à-Tête must be more formally evaluated. There are numerous ways that Tête-à-Tête can be evaluated and even more ways that it can be enhanced.

Evaluation Suggestions

Operating in multiple disciplines, Tête-à-Tête can be evaluated in an equally diverse manner. Below are several suggestions.

1. ECONOMIC & GAME THEORETIC

A game theoretic analysis of Tête-à-Tête's integrative negotiation protocol could reveal optimal strategies for both shopping agents and sales agents as well as uncovering whether the protocol lends itself to reaching equilibrium states. Simulations of agent economies suggest that it does [44], but an analysis of Tête-à-Tête's specific protocol may prove otherwise. A possible first approach to this analysis could be in that Tête-à-Tête's integrative negotiation protocol resembles a non-binding, iterative, first-price, sealed-bid auction protocol. A game theoretic analysis along these lines as well as a comparative analysis of this first-price bidding to a Vickrey (second-price, sealed bid)

auction is in order. An alternate approach to a theoretic analysis is to run high-speed

simulations of Tête-à-Tête agent economies such as was done by Kephart, et al [44].

2. USER INTERFACE STUDIES

Tête-à-Tête proposes a novel interaction model based on integrative negotiations for

shopping online. This interaction model involves a three-panel shopping interface design

and supporting user interface controls. To learn the effectiveness of the Tête-à-Tête

shopping experience, however, extensive user studies need to be performed. The systems

presented in Chapter 2 could be used for comparisons. Although many questions

regarding Tête-à-Tête's ease of use and navigation should be answered, the most

important question is whether merchants are significantly differentiated using Tête-à-Tête

versus other systems. Another important question is how similar are the online shopping

process and behaviors relative to physical-world shopping. For example, how relevant is

the online shopping framework presented in Table 1?

3. SOFTWARE ENGINEERING

Tête-à-Tête concerns itself with performance and scalability issues. However, further

analysis needs to performed in order to determine the system's performance bottlenecks

and scalability risks. For example, what type of fault tolerance, redundancy, and load

balancing is needed to maintain Tête-à-Tête's Registry, Dictionary, and Profile databases

and Shopping Agent Manager?

4. MARKETING RESEARCH

Although not an evaluation of Tête-à-Tête, per se, it would be interesting to analyze the

shopping preferences that the system captures. For example, the answers to questions

such as "Which value-added services do people care about?" would be valuable to merchants to help differentiate themselves and "Which product features are most important to people?" would be valuable to both manufacturers and merchants. Manufacturers could use the information to help direct product design and marketing materials and merchants could use the information to optimize their inventory portfolio. Tête-à-Tête can also be enhanced by providing this preference analysis in real-time upon demand from participating merchants and manufacturers.

5. BUSINESS PRACTICES

Although merchants are setting up their electronic shingles in droves, there is still a general lack of understanding of the pitfalls and opportunities that online retailing brings. In an effort to attract and retain customers, online retailers are resorting to money losing practices such as online auctions (e.g., Cendant's NetMarket [23]) as well as operations at planned losses for several years out (e.g., Amazon.com [28]). An interesting evaluation of Tête-à-Tête from a business perspective would be to survey current online retailers as well as ambivalent offline retailers to see if Tête-à-Tête's value proposition is sufficiently attractive to entice their participation. In fact, the most comprehensive evaluation of Tête-à-Tête's concepts would come from unleashing it into a commercial setting.

Future Directions

Tête-à-Tête is a prototype system that can either act as a platform for testing new online shopping ideas, technologies, and negotiation strategies or be run as an online service that supports an extensive number of shoppers, merchants, and product domains. Below is a list of suggestions to enhance Tête-à-Tête's online shopping experience.

1. ALTERNATIVE PROTOCOL (NO PRICE)

Tête-à-Tête shopping agents currently reveal all of its owner's preferences to sales agents. This may result in price discrimination (despite the arguments to the contrary in Chapter 4). An alternative protocol is one where shopping agents reveal all of its owner's preferences except total price. The implications of this, however, is that it would no longer be possible for sales agents to accurately configure product offerings since the key constraint would be missing. This means that a sales agent would need to send all of its configurable product offerings to the shopping agent for the shopping agent to configure them itself. As explained in Chapter 6, this is non-scalable for highly configurable product domains. Nevertheless, this approach would work for minimally configurable product domains while being incentive compatible.

2. ENHANCED CUSTOMIZER

As discussed in Chapter 6, Tête-à-Tête's product offering customizer was built only to demonstrate the concept of distributed product configurations and value assessments. More sophisticated configuration algorithms exist that could replace Tête-à-Tête's current customizer. In particular, the characteristics of Tête-à-Tête's price-dependent customization lends itself to a linear programming solution. Including total price (the complicating constraint) in the objective function of a Lagrangian relaxation algorithm could result in much better performance for even highly configurable products [51]. This approach would be worth pursuing. Also, the customizer could include other types of analysis to assess profit or even dynamically adjust price.

3. DYNAMIC STEREOTYPING

Currently, Tête-à-Tête uses a static database of several "notebook computer" stereotypes determined through a market segmentation exercise. A better stereotyping system would involve the dynamic modification of stereotype definitions (i.e., their list of active features and their preference settings) based on how shoppers who self-select themselves into a stereotype actually shop with Tête-à-Tête. A possible approach to do this would be to track a shopper's preference selection during a shopping session and when the shopper finishes the shopping session, have the shopper's preferences be averaged into the stereotype definition that the shopper self-selected. Only a certain number of these preferences for each stereotype should be maintained with older ones decaying out of the definition first. The averaging mechanism could use a mean-square difference or equally simple statistical algorithm to determine the stereotypical settings for each criterion.

A more dynamic version of the stereotyping system is one that identifies significant clusters of shoppers and automatically generates the list of stereotypes. This eliminates most of the manual marketing research except that a human would still (likely) need to assign intelligible labels to each cluster.

4. DISTRIBUTED ONTOLOGY EVOLUTION

As discussed in Chapter 4, manufacturers and merchants seldom share the same ontologies of products and services. In order for consumers to make an educated buying decision, the differences in these ontologies must first be reconciled. Third party services such as Consumer Reports [47] spend a lot of time and energy translating ontologies in order to permit apples-to-apples comparisons, thus lower consumer search costs. However, their labor-intensive process for constructing common ontologies does not

necessarily represent the product dimensions that most consumers care about. Also, these services usually only focus on product features, and neglect merchant features. Furthermore, some of the missed dimensions may include important points of differentiation among product manufacturers and merchants. Ideally, shared ontologies are created and evolved quickly, effectively, and without the delays and biases of a centralized human editorial board.

An enhancement to Tête-à-Tête is an ontology management mechanism that allows for the maintenance of product and merchant features within each product domain to be proposed by consumers, merchants, and manufacturers. For example, a "Create Feature" and "Edit Feature" buttons could be added to the Features dialog box shown in Figure 21. The "Create Feature" button would allow consumers to propose new features that aren't yet known to the system. Along with the new feature's name, the consumer would enter text describing the meaning of the feature along with each of the feature's attributes including which of the three attribute types best represent the nature of each attribute.

This new feature would be added to Tête-à-Tête's Dictionary database as an *inactive* feature within each stereotype in the product domain making the feature immediately accessible to all shoppers. Any shopper may then manually add this new feature to her *active* list of features and have them sent as critique performatives along with her other preferences. Of course, merchants may not understand these feature requests initially and may choose to ignore them. However, the merchants may have their sales agents monitor the occurrences of new features and, if many customers find a new feature to be an important dimension of consideration, then sales agents can alert their merchants of this

who may be compelled to address their customers' needs. Merchants may also wish introduce their own features to further differentiate themselves in the marketplace. Similarly, manufacturer's would benefit from knowing which product features shoppers clamor for and would likely want to learn about and introduce new product features.

If many shoppers within a stereotype find the new feature useful, the stereotype could evolve to include the new feature in the *active* list of preferences. At the same time, features that were once important, but are no longer considered while shopping, would get "demoted" to the inactive list. Eventually, particularly unimportant features drop out of a stereotype's definition of the product offering (and perhaps the Dictionary database) entirely. There would need to be a chronological decay function to help determine when such features get demoted or dropped. Also, the system would probably want to enforce a limited number of feature requests from each consumer and, more importantly, each manufacturer and merchant to prevent abuses of the system (such as introducing "marketing-speak" as features rather than introducing features with measurable value).

A harder problem is the modification of existing preferences. An "Edit Feature" button added to the Features dialog box shown in Figure 21 could begin this process. The consumer or merchant would then be able to modify the existing feature's description and attributes. Once proposed, the feature could be added as if it were a new feature, i.e., added to the inactive list of features for each stereotype. However, the features with multiple versions could be tagged as such. A simple user interface element could alert shoppers of new definitions for features which they may opt to use instead. This will allow shoppers to decide whether they prefer shopping with the new feature or not. Of

course, it will still be up to merchants to support them. In many cases, the changes may be minor such as adding a new display type to the appropriate discrete set attribute in a notebook computer domain. New attributes would impose similar issues as adding new features (discussed above). Changes in core attribute types, however, would be problematic, and possibly disallowed.

A dynamic, distributed, and evolvable shared ontology mechanism is important for reducing consumer search costs. Basing the evolution on what product and merchant features are important to stereotypical shoppers while they shop may be a viable approach to take as outlined above.

5. SUPERHELP

Tête-à-Tête maps attributes to specific, low-level characteristics of the product offering. This provides for a comprehensive evaluation of the product offering. However, in some instances, shoppers may prefer a coarser level of granularity when expressing preferences for product offerings. Tête-à-Tête's stereotyping system (see Chapter 5 and above) is one example of mapping high-level preferences (e.g., an Average User stereotype), to low-level specifics. Another such mechanism, termed SuperHelp, acts more like a common "wizard" in desktop applications in order to automate the expression of certain preferences.

For example, Tête-à-Tête's notebook computer domain may support SuperHelp for something called "Performance" that helps the shopper establish preferences for several performance-related features such as Processor Speed, System Memory, and Total Cache. A shopper would notice that a certain feature had SuperHelp by having its '?' help button

changed to a '?' icon with a lightening bolt through it (for example). Clicking on this SuperHelp button would reveal descriptive help text about the feature (as with normal help) but include the option to by guided through the setting of one or more features. Similar to PersonaLogic [6], SuperHelp would then interview the shopper about her needs in high-level terms (e.g., "Do you need fast access to the Internet?" and "Do you intend to play games on your computer?"). Hand-crafted heuristics would then translate the answers to these questions into specific preferences for features. Most important and similar to the dynamism of the rest of Tête-à-Tête (and unlike PersonaLogic), the answers to the interview questions would immediately register in a co-located results panel to help the shopper understand how their high-order preferences map to specific feature preferences. This understanding builds confidence in the system and helps facilitate transactions.

6. CONJOINT SHOPPING

Another mechanism for mapping high-order preferences to product offering specifics is through a conjoint analysis of product offering alternatives. Rather than express feature preferences for products, shoppers may simply express a preference of one product over another (without needing to explain why). The system could then extract or refine specific preferences and utilities based on analyzing the differences in feature sets. This mode of shopping could be implemented in Tête-à-Tête by having shoppers express their conjoint preferences through the act of dragging and dropping product offering value bars above or below one another. As before, the vertical orientation of the value bars represents the relative value that each product offering brings to the shopper. When conjoint shopping, the shopper manually instructs the system of these relative values.

These, in turn, help extract preferences which get fed back into the system allowing new product offerings to be proposed. Essentially, Tête-à-Tête would allow shoppers to conjoint shop, feature shop, or even mix the two styles of shopping. This would allow shoppers to express specific constraints on features while leaving the rest of the preferences to be determined via conjoint analysis.[34]

7. USER PROFILES

Tête-à-Tête currently only supports anonymous shopping. Acting as a trusted shopping advisor, however, each shopper's shopping agent should keep a profile of its owner to provide a better shopping experience. For example, preferences could be saved for future consideration, account information would only need to be entered once, and email alerts could be sent of interesting marketplace events such as the assessed value of a product reaching a shopper-specified threshold. This last suggestion would, of course, require that the shopping agent proactively shop for the shopper even when the shopper is not actively using the Tête-à-Tête system.

8. SHOPPING INTERFACE ENHANCEMENTS

Tête-à-Tête's shopping interface can be enhanced in numerous ways. For example, the details of product offerings in the left panel could be personalized to include only those features in the right panel and in the order that the shopper established. Also, although Java works well with the system and allowed Tête-à-Tête to be developed in a very short period of time, alternative user interface technologies may be more appropriate such as DHTML (dynamic HTML). Additionally, a "Find" button could help shoppers more

---

[34] This merger of conjoint analysis and feature-based shopping was inspired by [56].

quickly identify known products and a "Products" button could allow the shopper to select from multiple product domains. Furthermore, profiles should be maintained including certain shopping preferences such as the maximum number of proposals each shopper is willing to consider form each merchant. Finally, special features could be added to the system to further help facilitate transactions such as reviews and reputation facilities (described next).

9. REPUTATION FACILITY

Reputation facilities can be included as one or more features within Tête-à-Tête as additional dimensions for shoppers to consider. For example, a reputation feature for notebook computers may be one where the shopper can express a preference for the manufacturer. The reputation attribute could be a new type of attribute where manufactures can be relatively ranked similar to features in the right panel. The relative ordering of manufacturers could either be established manually or via a reputation service. For example, Consumer Reports [47] or BizRate [40] may provide an ordered list of manufacturers based on various criteria. The shopper could select the third-party reputation service that most closely shares the shopper's values.

10. SALES AGENT INTERFACE

Currently, Tête-à-Tête does not support a graphical user interface for sales agents. Although completely autonomous when proposing product offerings, sales agents could benefit from easier administration tools. For example, merchants may want to alter their profit preference or specify which product catalog to access (and how). For integration with heterogeneous systems, the interoperability issues could get rather involved.

11. CROSS-SALE ENGINE

A further enhancement of Tête-à-Tête that would benefit merchants is the ability of merchants to offer cross-sales through the shopping interface. For example, part of the left panel could be dedicated to entice shoppers to consider products related to the one currently being shopped. In the notebook computer domain, for example, these products may include carrying cases or docking stations. In general, Tête-à-Tête could open up new channels of interaction with one or more merchants. The *note* field in the sales agent's response performatives (see Table 14) is a small step into providing such a channel.

12. HYBRID NEGOTIATIONS AND SPIDERING

Tête-à-Tête shopping agents currently assume that a sales agent will be on the other end of the negotiating table. However, there are currently no such sales agents in the market today. An interim solution is one where advanced spidering technologies access product offering information from multiple manufacturers and merchants and cache these in a local database. The shopping agent would then request information from the database rather than from a sales agent directly. As discussed in Chapter 6, this is sub-optimal from a scalability perspective for highly configurable product domains, but possibly optimal in terms of launching a service that's of immediate value to shoppers and merchants alike.

References

[1] Forrester Research Report. *On-Line Internet Spending.* 1997.

[2] Forrester Research Report. "Affordable Intimacy Strengthens Online Stores." September, 1997.

[3] P. Maes. "Agents that Reduce Work and Information Overload." *Communications of the ACM,* vol. 37, no.7, pp. 31-40, 146, ACM Press, July 1994.

[4] R. Guttman and P. Maes. "Agent-mediated Integrative Negotiation for Retail Electronic Commerce." *Proceedings of the Workshop on Agent Mediated Electronic Trading (AMET'98),* Minneapolis, Minnesota, April 9, 1998.

[5] R. Guttman, A. Moukas, and P. Maes. "Agent-mediated Electronic Commerce: A Survey." *Knowledge Engineering Review,* Vol. 13:3, June 1998.

[6] PersonaLogic URL: <http://www.personalogic.com>

[7] Firefly Network URL: <http://www.firefly.com>

[8] U. Shardanand and P. Maes. "Social Information Filtering: Algorithms for Automating 'Word of Mouth'." *Proceedings of the Computer-Human Interaction Conference (CHI'95),* Denver, Colorado, May 1995.

[9] BargainFinder URL: <http://bf.cstar.ac.com/bf>

[10] Jango URL: <http://www.jango.com>

[11] Junglee URL: <http://www.junglee.com>

[12] R. Doorenbos, O. Etzioni, and D. Weld. "A Scalable Comparison-Shopping Agent for the World Wide Web." *Proceedings of the First International Conference on Autonomous Agents (Agents'97).* Marina del Rey, CA, February 1997.

[13] J. Bosak. "XML, Java, and the Future of the Web." March 1997.

[14] R. Gibbons. *Game Theory for Applied Economists.* 1992.

[15] J. Rosenschein and G. Zlotkin. *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers.* MIT Press, 1994.

[16] P. Milgrom. "Auctions and Bidding: A Primer." *Journal of Economic Perspectives,* pp. 3-22. Summer 1989.

[17] Kasbah URL: <http://kasbah.media.mit.edu>

[18] A. Chavez, D. Dreilinger, R. Guttman, and P. Maes. "A Real-Life Experiment in Creating an Agent Marketplace." *Proceedings of the Second International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM'97).* London, UK, April 1997.

[19] AuctionBot URL: <http://auction.eecs.umich.edu>

[20] P. Wurman, M. Wellman, and W. Walsh. "The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents." *Second International Conference on Autonomous Agents (Agents'98),* May 1998.

[21] P. Samuelson and W. Nordhaus. Economics, 16th ed. 1997.

[22] First Auction URL: <http://www.firstauction.com>

[23] NetMarket URL: <http://www.netmarket.com>

[24] OnSale URL: <http://www.onsale.com>

[25] J. Lynch, Jr., and D. Ariely. "Interactive Home Shopping: Effects of Search Cost for Price and Quality Information on Consumer Price Sensitivity, Satisfaction with Merchandise, and Retention." Technical Report, Fuqua School of Business, Duke University.

[26] C. Beam, A. Segev, and J. G. Shanthikumar. "Electronic Negotiation through Internet-based Auctions." CITM Working Paper 96-WP-1019, December 1996.

[27] T. Sandholm and V. Lesser. "Coalition Formation among Bounded Rational Agents." *14th International Joint Conference on Artificial Intelligence (IJCAI'95),* Montreal, Canada, 1995.

[28] Amazon.com URL: <http://www.amazon.com>

[29] R. Fisher and W. Ury. *Getting To Yes: Negotiating Agreement Without Giving in*, 2nd ed. 1991.

[30] R. Lewicki, D. Saunders, and J. Minton. *Essentials of Negotiation*. Irwin, 1997.

[31] R. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley & Sons, 1976.

[32] S. Parsons, C. Sierra, and N. Jennings. "Agents that Reason and Negotiate by Arguing." *Journal of Logic and Computation*.

[33] J. Austin. *How to do Things with Words*. Cambridge, MA: Harvard University Press, 1962.

[34] T. Finin, Y. Labrou, and J. Mayfield. "KQML as an Agent Communication Language." Invited chapter in Jeff Bradshaw (ed.), *Software Agents*. MIT Press, Cambridge, 1997.

[35] TRUSTe URL: <http://www.truste.org>

[36] K. Sycara. "The PERSUADER." In *The Encyclopedia of Artificial Intelligence*. D. Shapiro (ed.), John Wiley and Sons, January, 1992.

[37] A. Rangaswamy and G. Shell. "Using Computers to Realize Joint Gains in Negotiations: Toward an 'Electronic Bargaining Table'." *Management Science*, vol. 43, no. 8, August 1998.

[38] OptiMark Technologies URL: <http://www.optimark-tech.com/home.html>

[39] Priceline URL: <http://www.priceline.com>

[40] BizRate URL: <http://www.bizrate.com>

[41] G. Zacharia and P. Maes. "Collaborative Reputation Mechanisms in Electronic Marketplaces." Working paper, available online at: <http://www.media.mit.edu/~lysi/reputation.html>

[42] J. Bailey, E. Brynjolfsson, and M. Smith. "In Search of 'Friction-Free' Markets: An Exploratory Analysis of Prices for Books, CDs and Software Sold on the Internet." MIT working paper, 1998.

[43] J. Kephart, J. Hanson, D. Levine, B. Grosof, J. Sairamesh, R. Segal, and S. White. "Dynamics of an Information-Filtering Economy." *Proceedings of the Workshop on Cooperative Information Agents (CUA '98)*, Paris, France, 1998.

[44] J. Sairamesh and J. Kephart. "Price Dynamics of Vertically Differentiated Information Markets." To appear, *Proceedings of the First International Conference on Information and Computational Economies*, Charlestown, S.C., October 1998.

[45] Yahoo! URL: <http://www.yahoo.com>

[46] RosettaNet URL: <http://www.rosettanet.org>

[47] Consumer Reports URL: <http://www.consumerreports.com>

[48] Logical Decisions URL: <http://www.logicaldecisions.com>

[49] Dell Latitude CP URL:
<http://commerce.us.dell.com/dellstore/config.asp?order_code=890404>

[50] M. Yokoo, E. Durfee, T. Ishida, and K. Kuwabara. "Distributed Constraint Satisfaction for Formalizing Distributed Problem Solving." *Proceedings of the 12th IEEE International Conference on Distributed Computing Systems*, 1992.

[51] M. Fisher. "An Application Oriented Guide to Lagrangian Relaxation." *Interfaces 15: 2*, March-April 1985.

[52] BroadVision URL: <http://www.broadvision.com>

[53] Trilogy's Selling Chain™ URL:
<http://www.trilogy.com/prodserv/products/main.html>

[54] Acme URL: <http://www.acme.com>

[55] R. Glushko, J. Tenenbaum, and B. Meltzer. "The XML Revolution: Making the Web Accessible to Commerce Agents." To appear, *Communications of the ACM, special issue on Agents and Electronic Commerce*, 1998.

[56] A. Kleiner. "The Merchant Health Question of Electronic Markets: Why Retail Electronic Commerce is Broken and How it Can be Fixed." MBA Thesis, Massachusetts Institute of Technology, Sloan School of Management, May 1998.