technical paper

Merkato Overview

A platform for real-time market-based network resource allocation

Abstract

This document presents a high-level overview of the Merkato platform for real-time market-based network resource allocation. Its highly scaleable nature allows Merkato to enable Internet bandwidth markets, in a variety of scenarios. Flexibility is a guiding principle, resulting in a system that meshes with the reality of heterogeneous service provider environments.

Merkato Application Topologies
Decentralized Marketplace of Agents

Merkato creates true markets in software. Unlike the situation in resource planning, management and analysis systems, in Merkato, intelligence is not centralized. A market consists of goods, a collection of buyers and sellers of those goods, each acting in its own interest (to maximize its utility), the rules which govern these interactions, and the means of interaction.

Thus, a fundamental concept in Merkato is that of agents—independent decision-making software entities. Agents act on behalf of the users, modeling their needs and buying or selling intelligently on their behalf. The users can trust their software agents to be private, rational, and faithful.

At the core of the market are rules, designed to be neutral and universal, in order to ensure the overall system's economic efficiency, given the fact of diverse and self-interested agents. The rules are realized through mechanisms that scale to very large numbers of participants, both from the point of view of computational complexity and messaging load.

Underlying the rules is a flexible model of resource allocation, allowing Merkato to be applied to a variety of resources. The shareable resource model makes it uniquely applicable to information resources, such as Internet service bandwidth.
This can be summed with the view shown in Figure 1. The participants, which can be very large in number, exist around a core of shared resources and market rules.

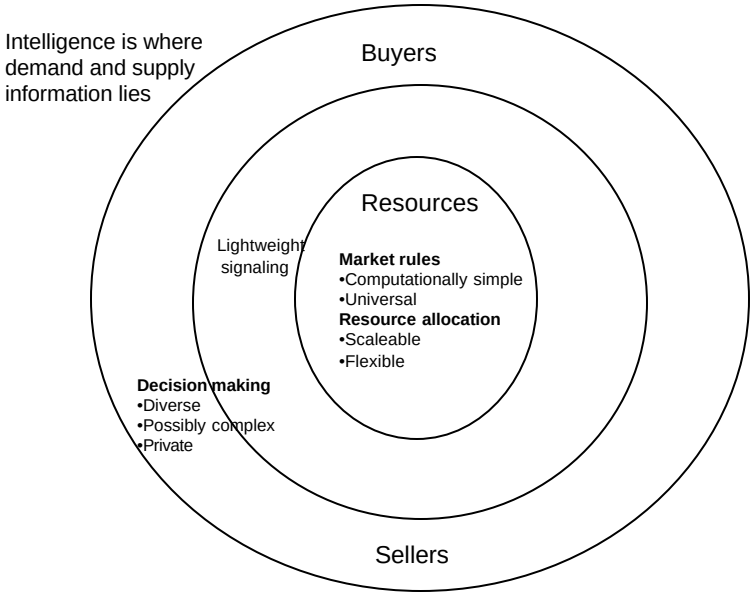

Figure 1 Decentralized Market

Distributed Micro-Markets

As shown in Figure 2, in Merkato, each resource has its own micro-market, with a resource agent running the market rules, and controlling allocation of the resource. Resource agents are purely local, and make decisions only based on the market rules, the availability of that resource and the actions of the seller and the buyers of that resource. The micro-market for that resource can exist alone or be integrated in a large distributed market of many resources. Primary resources can be bought and combined or aggregated into higher resources or services, which are sold in separate micro-markets, without requiring the markets to be directly tied together.

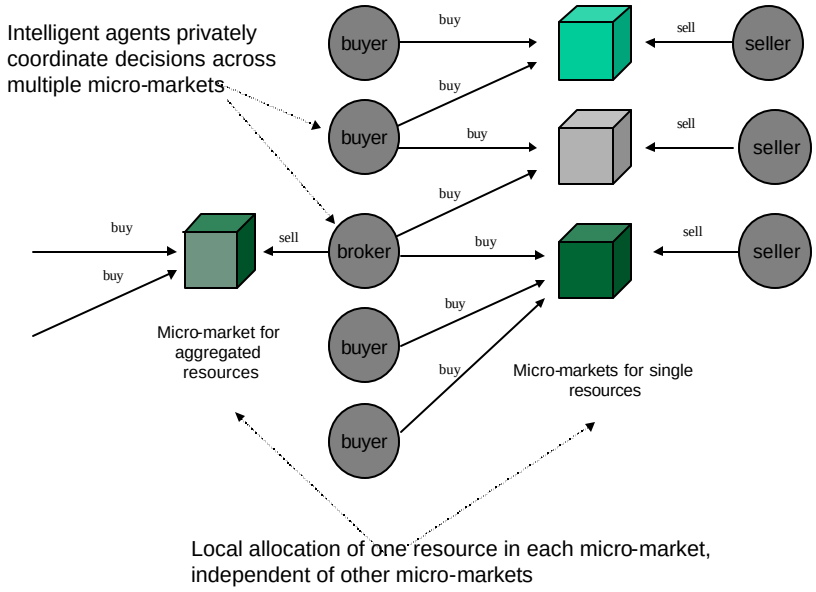

Figure 2 Distributed Micro-Markets

Since the key elements are either mobile (buyer or seller agents) or fully distributable (resource agents), the platform has a fully customizable application topology. The elements in Figure 2 can all run on a single computer or each on a different machine. Thus, Merkato can be deployed across resources that are physically dispersed, and operated by different entities. Furthermore, perhaps most importantly, Merkato can be deployed as a point solution, or as a very large-scale network-wide system, or anything in between.

Together, the decentralized and distributed aspects of Merkato can be summed up as follows: buyer and seller agents think globally (but selfishly), and resource agents act locally. On a high-level, this philosophy is akin to the “end-to-end principle” that underpins the design of the TCP/IP protocol suite. These aspects are key to making Merkato well suited to the reality of multi-provider interconnected communications networks.

Merkato-Network[1] Interaction Framework

Merkato is designed to co-exist with the widest possible range of network architectures, and operations procedures. In particular, it is fully compatible with the operations processes of Internet Service Providers.

Merkato integrates directly with the resource provisioning process. Users (buyers and sellers of resources) interact with their Merkato agents, and these agents negotiate through a micro-market run by a Merkato resource agent. The resource agent allocates resources and provisions the allocations in real-time. Unlike provisioning, routing and monitoring/measurement processes are orthogonal to Merkato. In other words, as shown in Figure 3, the Merkato core and these processes need not be directly aware of each other. However, these processes can optionally be linked with Merkato, interacting with the periphery of the market—in the decision-making of the buyer.

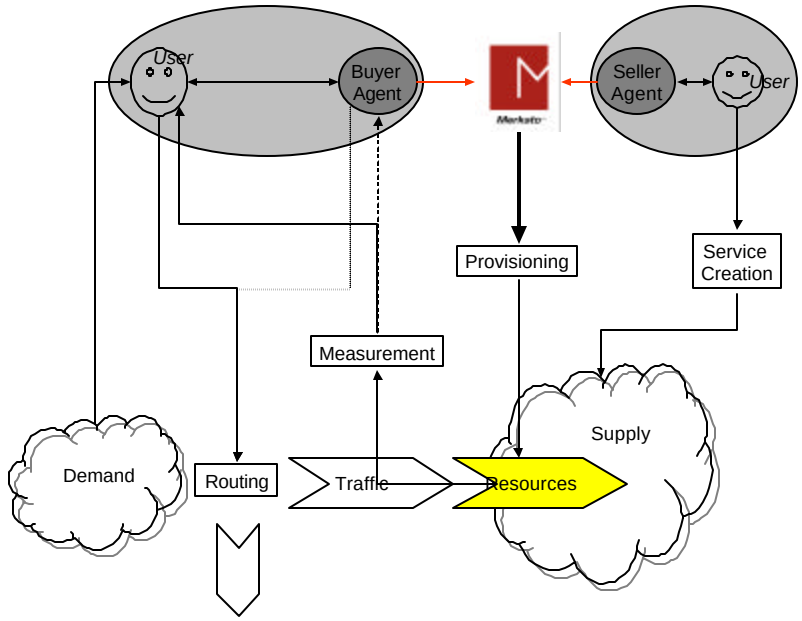

Figure 3 Merkato-Network Interacting Processes

The buyer, as a user of Merkato, configures a Merkato buyer agent, whose purpose is to get the resources within the desired service, at the right prices. The buyer measures traffic levels and service quality, and uses that information to determine the willingness to pay for the seller's services. Such feedback, coming from a wide variety of sources, can go into the Merkato buyer agent, either automatically or via the user. The user may be the human user, interacting with Merkato via its user interface, or an application into which the Merkato buyer agent has been integrated.

For a given external demand, the buyer decides to route some traffic into the resources offered by the seller (portions of the demand may be routed to different services from other sellers in other markets). Routing may be influenced, directly or indirectly, by the

[1] We use the terms "network", "service", "resource" as very general abstractions. Thus routing for example, is generically the decision making process that leads to demand for services being directed to service offerings, and provisioning is the process by which a portion of resources within a service are set aside to serve a particular buyer's traffic. For specificity, one may think for example, of the seller as an Internet Service Provider, and the resources as IP bandwidth.

price, availability, and measured quality of resources. Some network routing protocols have this kind of feedback built-in, and others do it through human input to set parameters such as weights, preferences, costs, etc. The Merkato buyer agent may optionally provide automated input to influence routing, if the routing protocol and algorithms have such capabilities.

A valuable prism through which to view the overall interaction between Merkato and the resources is that of the time scales of the processes. The whole consists of four “loops” operating on vastly different time scales, as illustrated in Figure 4.

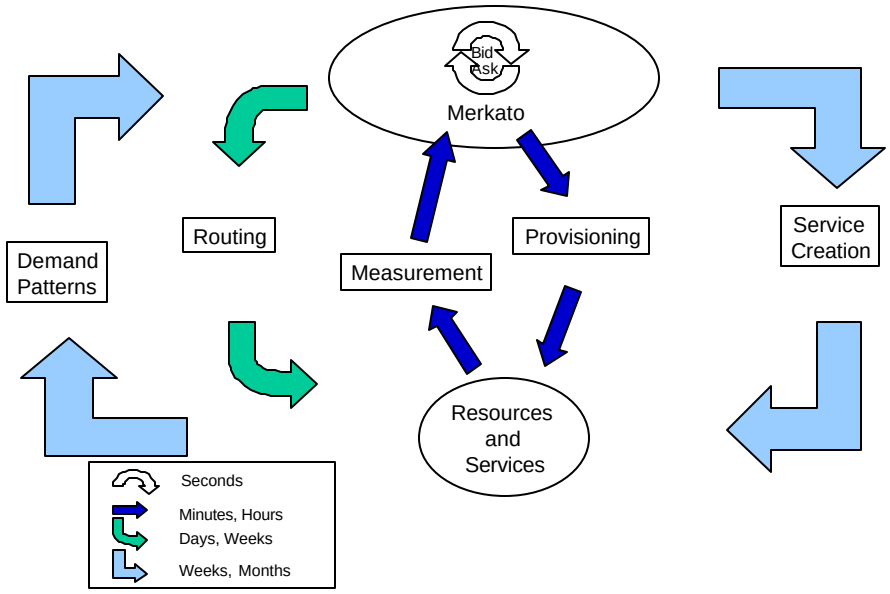

Figure 4 Merkato-Network Interaction Timescales

The innermost loop, operating on the timescale of seconds, consists of agents negotiating in real-time, making buying and selling decisions, reacting to market conditions (i.e. each other), through bids and asks.

On a slightly slower timescale, on the order of minutes, we have the market transactions, which are executed through real-time provisioning. This process includes short-term demand and supply changes that are measured automatically by the agents, or input by a user into Merkato agents.

On a still slower timescale (days or weeks), the buyer can adjust routing, to decide what proportion of it’s overall demand should be sent to the resources in this market, and how much to direct to other resources in other markets. This is a relatively “loose” or “soft” coupling, which can be integrated in the buying agents, or, left to a manual process for the human user. In particular, Internet routing is a delicate area that, currently, most service providers do not couple to other processes in an automated way – instead they prefer to change routing carefully, through human intervention. In such cases, Merkato will simply provide price and bandwidth information that the user may or may not use in configuring routing.

On the outermost loop, we have classical supply and demand dynamics. Long-term demand changes create market trends, to which sellers respond by creating or removing services, and which are taken into account in physical capacity planning.

Overall, this flexible framework in which Merkato interacts with, and improves, a variety of operations processes makes Merkato a true middleware platform that makes resource allocation more efficient, in a manner that is transparent to the end applications that use the services.

Merkato-powered Internet Bandwidth Transactions Exchange Topologies

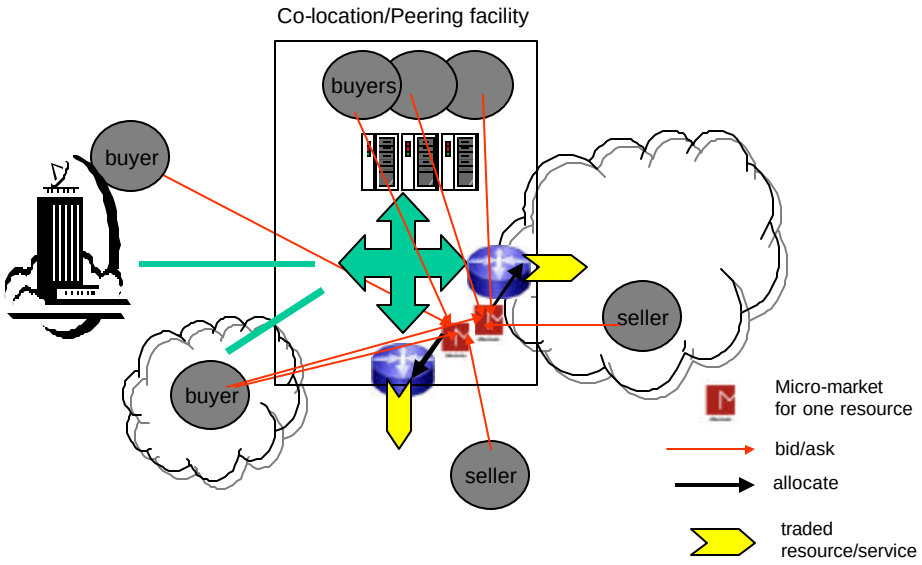

Figure 5 Single-Facility Bandwidth Exchange

The Merkato platform can enable Internet bandwidth markets in a variety of ways. In the simplest scenario, Merkato is deployed within a single bandwidth exchange facility (or "pooling point"). These co-location/peering facilities bring buyers and sellers together in a place where they can physically exchange traffic. Within the facilities, high bandwidth cross-connections are available for all participants, at a low fixed cost, through a shared VLAN or switch fabric.

The resources traded in this market are the outgoing services offered on the networks through that access point. For each resource (network service being sold), a micro-market is created, with a resource agent controlling the allocation of shares of that resource to any number of buyers. Buyers are enterprises bringing traffic to the facility over dedicated connections; content providers with servers co-located in the facility, or network service providers buying bandwidth from peer networks.

The micro-markets can run each on separate Merkato servers operated by the sellers, or, all on shared Merkato servers operated by the neutral exchange operator. Merkato servers are not required to be physically co-located, but for security and reliability reasons, it is preferable that they be close to the resource allocation point (which is the edge router on the seller's network).

This framework can be extended to a metropolitan area bandwidth exchange, which extends the low cost, high-bandwidth cross-connectivity to multiple co-location facilities through a high-speed metropolitan "ring", as shown in Figure 6. A buyer can buy from any seller in any of the facilities. This scenario creates greater liquidity in the market by enlarging to a larger number of participants.

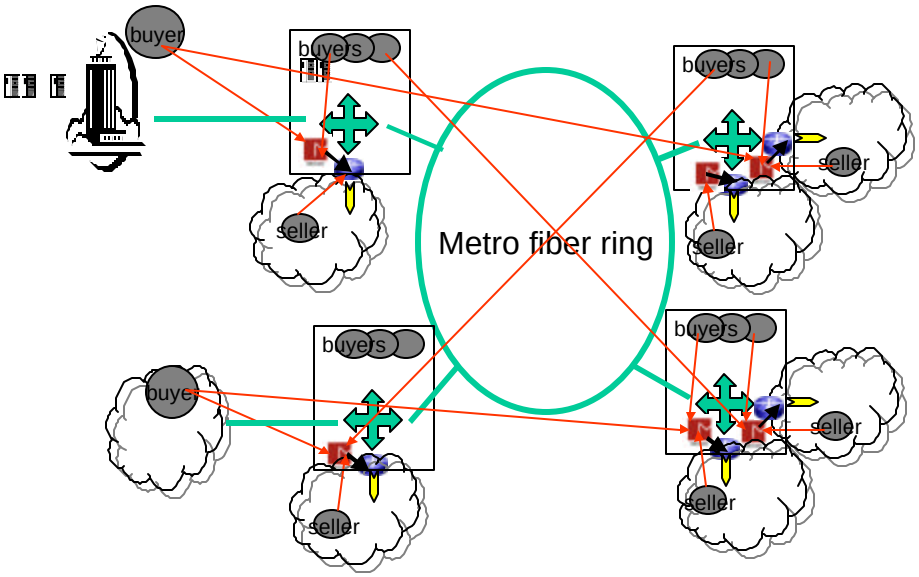

Figure 6 Metropolitan Area Bandwidth Exchange

A wide variety of other market scenarios are possible, including trading of point-to-point "circuits" along pre-determined paths, or a hop-by-hop resource allocation model where buyers construct their own paths or virtual networks by buying from a large number of micro-markets, one for each link in a network. In essence, the flexible topology of the Merkato platform allows it to match both the physical and the business topology of the marketplace.

Benefits of Merkato

The general economic benefits of dynamic bandwidth markets are well known. For buyers they are: price transparency, and on-demand bandwidth (as opposed to prohibitive long-term contracts). For sellers, the benefits are: extracting more revenue from fixed resources whose costs are sunk, reduced cost of selling, and long-term risk management. Merkato markets provide all of the above benefits, but we focus here on the unique benefits of the decentralized, and distributed real-time nature of the Merkato platform.

The decentralized nature of Merkato allows sellers to always offer an optimal mix of services, even where there is little prior knowledge of demand. Indeed market research is notoriously difficult for Internet traffic, since new applications and traffic demands appear in an explosive manner. Witness web browsing, streaming media, Internet telephony, and distributed music file sharing: for each of these traffic types that emerged over the past decade, demand has been completely unforeseen, or greatly under- or over-estimated by service providers.

In the example of Figure 7, a service provider is capable of offering two service types, e.g. 1) simple guaranteed throughput of IP data, and 2) guaranteed throughput with relatively low latency[2]. However, the amount of traffic accepted in each type must be controlled, otherwise the service quality will degrade. Moreover, the traffic of both types must be jointly controlled, or else both service types may be degraded.

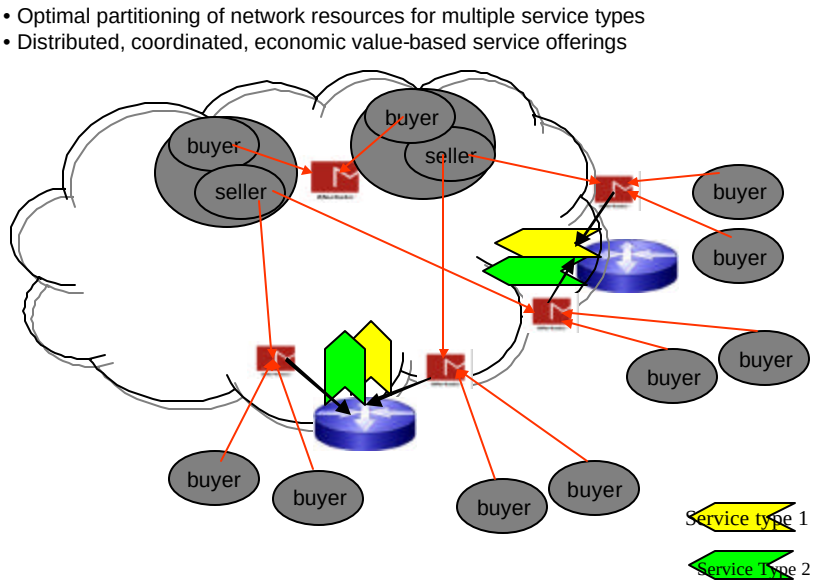

Figure 7 Market-Based Multi-Service Offerings

With Merkato, the provider creates a micro-market for raw-bandwidth, which represents the total resources available. The raw bandwidth market is an abstraction, which does not control any physical resource elements.

Two broker agents are created, each selling one of the service types. Each broker agent (which is an agent containing coordinated buyer and seller sub-agents) buys from the raw-bandwidth market, and based on the obtained amount, offers certain quantities of that particular service in micro-markets at multiple ingress points (which may be exchange points as in Figure 5).

[2] On today's Internet, this kind of differentiation may be achieved by scheduling and buffering classes of packets differently (using Diff-Serv) or traffic engineering (using MPLS), or combinations thereof.

Buyers choose to bid for bandwidth in a particular service micro-market. A single buyer can buy some bandwidth in both service types (for simplicity, the above figure shows each buyer buying only one type of service).

On their “buy-side”, the two broker agents compete against each other in the raw bandwidth micro-market, and their willingness to pay is derived from the real-time demand they see on their “sell-side” from the buyers. Conversely, on their sell-side, the quantity they offer is constrained by the quantity and price they obtain in the raw bandwidth market.

Thus, the service provider achieves an overall result that is not possible without a decentralized and distributed market. The quality of each service is maintained, and the trade-off between the two services is dynamically adjusted by the markets to optimize the service provider’s overall profit. For example, if at some time, there is very low demand for one of the services and high demand for the other, the offered quantity of the second service will increase, and that of the first service will decrease.

Similarly, Merkato micro-markets can enable multiple providers to offer particular services across multiple domains, as show in Figure 8. The broker agent for a service in a given provider’s network ensures that sufficient resources for that type of service are purchased from other providers (“peers”), to support the traffic of that type that will flow into the peer domain. Conversely, if the peer provider also sells access to that service through Merkato (as shown with the provider on the left in the figure), then the broker ensures that resources are purchased from the peer only to the extent justified by the demand from buyers.

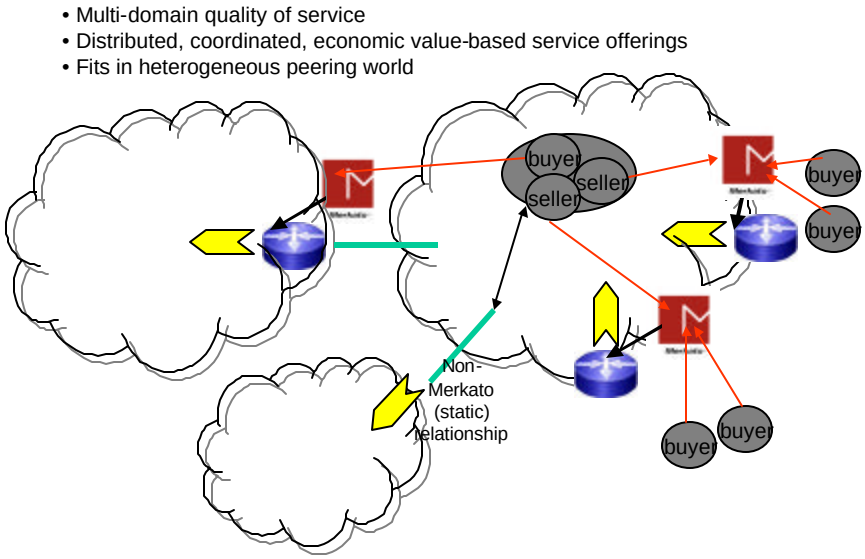

Figure 8 Market-Based Multi-Domain Services

Both of these wide area results (economic optimization of the mix of different services and multi-provider quality) are achieved without requiring centralized control, or prior knowledge of demand. Through particular Merkato topologies, simple and scalable

distributed markets obviate the need for non-scalable central planning, and impossible market forecasts. Furthermore, since the resource allocation model is flexible enough to take into account statically provisioned resources, the Merkato platform can be deployed progressively alongside existing practices, providing clear and increasing benefits at each step.

Functional Overview of Merkato

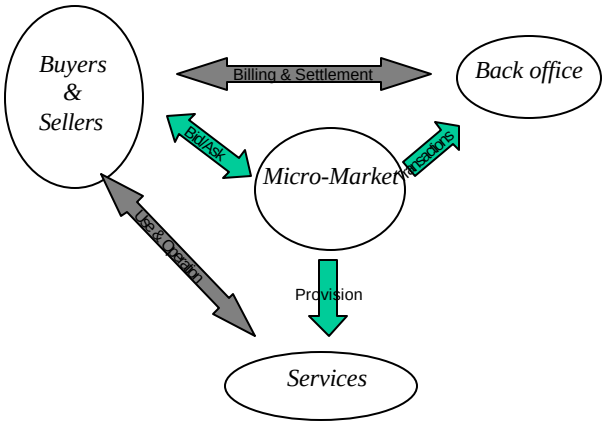

Figure 9 Business Roles

Merkato functions within a system that includes the resources and services allocated, but also the business and financial infrastructure of the market.

This is reflected in the software functionality. The micro-markets, running on standard application servers, contain agents, resource control, and transactions systems. Users interact with the market through mobile agents. A number of Merkato servers within an operating domain connect to a common Merkato back-office, which contains and manages transactions and participant information. The Merkato application servers are distributed throughout the network of resources, close to the allocation control points.

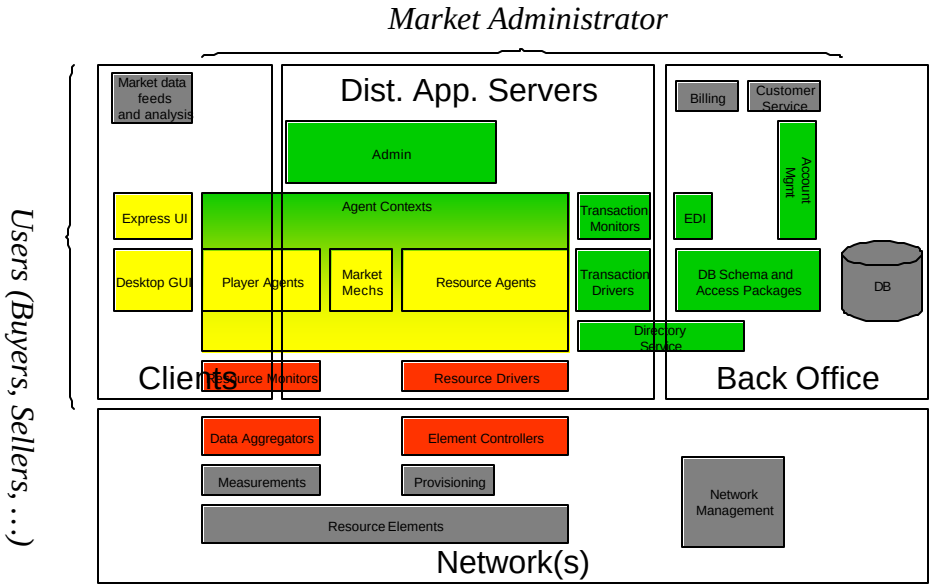

Figure 10 Functional Areas

As shown in Figure 11, aside from external changes input by users into agent configurations, the market, i.e. real-time buying, selling provisioning and transactions processing, operates in a fully automated manner.

In the real-time market, users interact with their mobile agents either directly on the application server, via a simplified browser-based “Express” user interface, or by downloading the agents and running them on a client-side platform, within a powerful Merkato “Desktop” graphical user interface. The Merkato desktop itself can run within a browser, or as a stand-alone installed application on the client operating system.

Agents live within agent contexts, either on the client side or on distributed application servers. The contexts provide the agents their run-time environment, including file systems, communications, and mobility. For resource agents, the context provides the real-time resource allocation capabilities, via an abstract resource driver interface, which allows a wide variety of underlying resource elements and services to be provisioned. The resource agent also interacts with the back office through transaction drivers. For player agents (i.e. buyer, seller and broker agents), the context provides real time monitoring of market data and service quality, as well as directory services to locate resource agents.

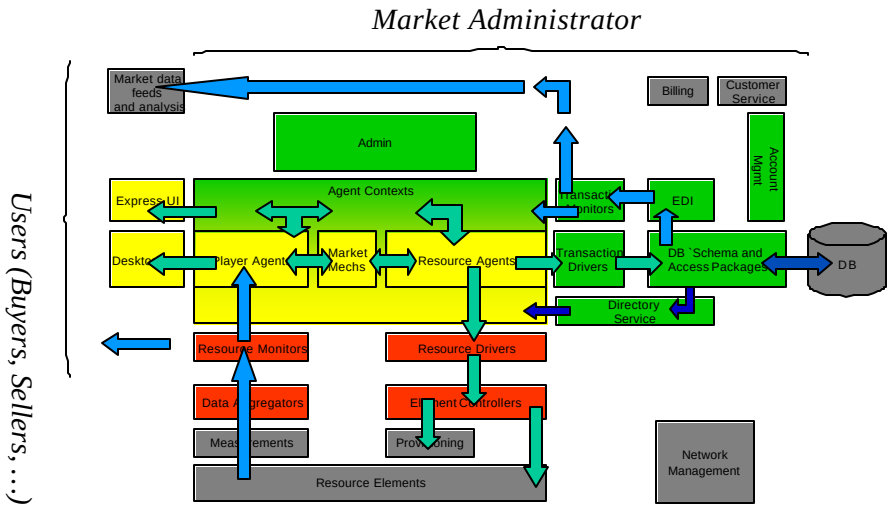

Figure 11 Functional View: Real-Time Market

Merkato servers can be fully administered through a web-based interface for creating and configuring all agents. The distributed servers also integrate with billing, and customer service in the back office, and co-exist with standard network management processes, as shown in Figure 12. In addition to a web-based administrator interface, the administrative functions can also be invoked via HTTP. Thus player (buyer or seller) agent creation can be integrated with the account management back-end. Similarly, resource agent creation can be connected to inventory management.

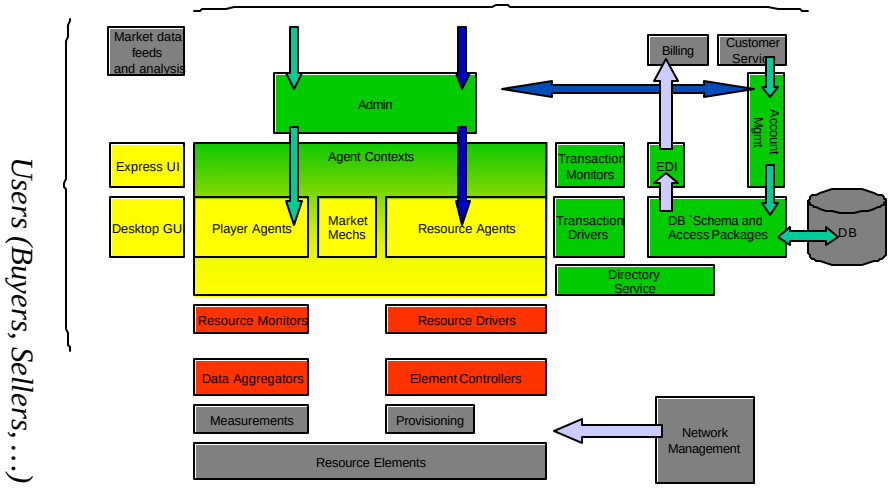

Figure 12 Functional View: Administration

Conclusion

In historical terms, the first generation of commodity markets, which has existed for centuries, consists of human negotiation and trading of physical goods. Current commodity markets for oil, electricity, pork bellies, etc., represent the second generation, which has electronic markets for physical goods, but the market mechanisms are necessarily separate from the physical delivery of the goods. The emerging commodities—information resources such as bandwidth, storage, and computation—can be delivered (provisioned) in real-time, and if not used, simply expire in real-time. Thus, they call for a new kind of market, one in which software agents model the infinitely diverse ways of valuing the resources, of sharing them, and negotiate in real-time. With a transaction model consisting of a distributed software platform like Merkato, lightly and strategically laid right in the network of goods themselves, these emerging "resource markets" can achieve greater economic efficiencies than any other markets in history.