MoB: A Mobile Bazaar for Wide Area Wireless Services

R.Chakravorty, S.Agarwal,
S.Banerjee and I.Pratt

mobicom 2005

What is MoB ?

- It is an infrastructure for collaborative wide-area wireless data services.
- Mobile users can trade services with each other in a wide area network setting.
- This allows a decoupling of services and infrastructure provisioning
  - I don't have to necessarily obtain everything from the Internet via my infrastructure provider.
  - I could rely on other mobile devices to help me.

Contributions in Brief

- The authors present a new open market architecture that allow mobile users to trade various services in a flexible manner.
- They address issues such as billing, security, etc.
- They do an implementation of the architecture (as a middleware).
- Augment the implementation with simulations to provide additional insight.

Roadmap

- Motivation for MoB and possibility of fine grained service provisioning.
- The Architecture
- Examples of operations.
- Some Implementation details
- Results
- Conclusions

The High Level Problem

- Cellular coverage is adequate at best.
- Public WLAN hotspots are also emerging --using these one can connect to the Internet -- however coverage is spotty.
- Typically, users connect to the networks provided by their infrastructure providers (such as T-Mobile).
  - These are long term relationships.
- Connectivity problems are location and provider specific -- these can result in poor performance of most Internet protocols (as we all know).
- Can users obtain services on a fine-grained basis ?

What does MoB give ?

- MoB -- standing for Mobile bazaars tries to provide an open market.
- It supports collaboration to improve data services.
- The key features are:
  - It decouples infrastructure providers from service provisioning and thus enable fine grained competition.
  - Allows service interactions on finer grained (they claim arbitrary) timescales
  - Promotes flexible composition of these fine-grained service interactions based on user needs.

Coffee Shop Study

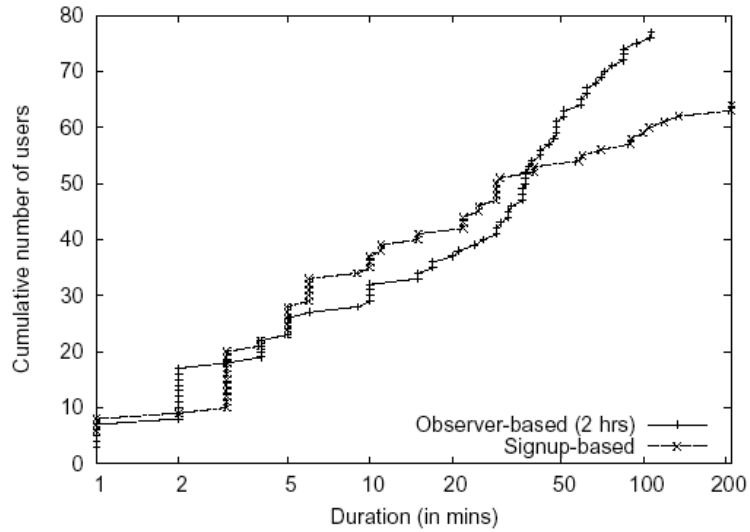
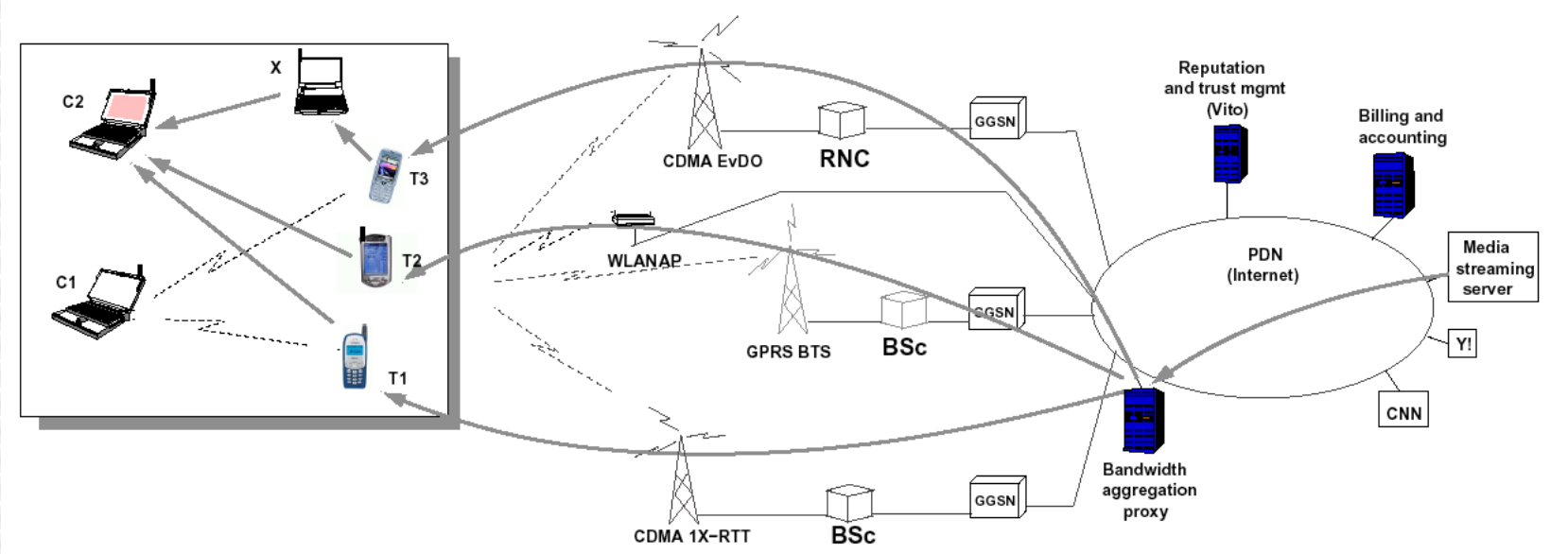


Figure 2: Distribution of user persistence in coffee-shop environments (x-axis on log-scale).

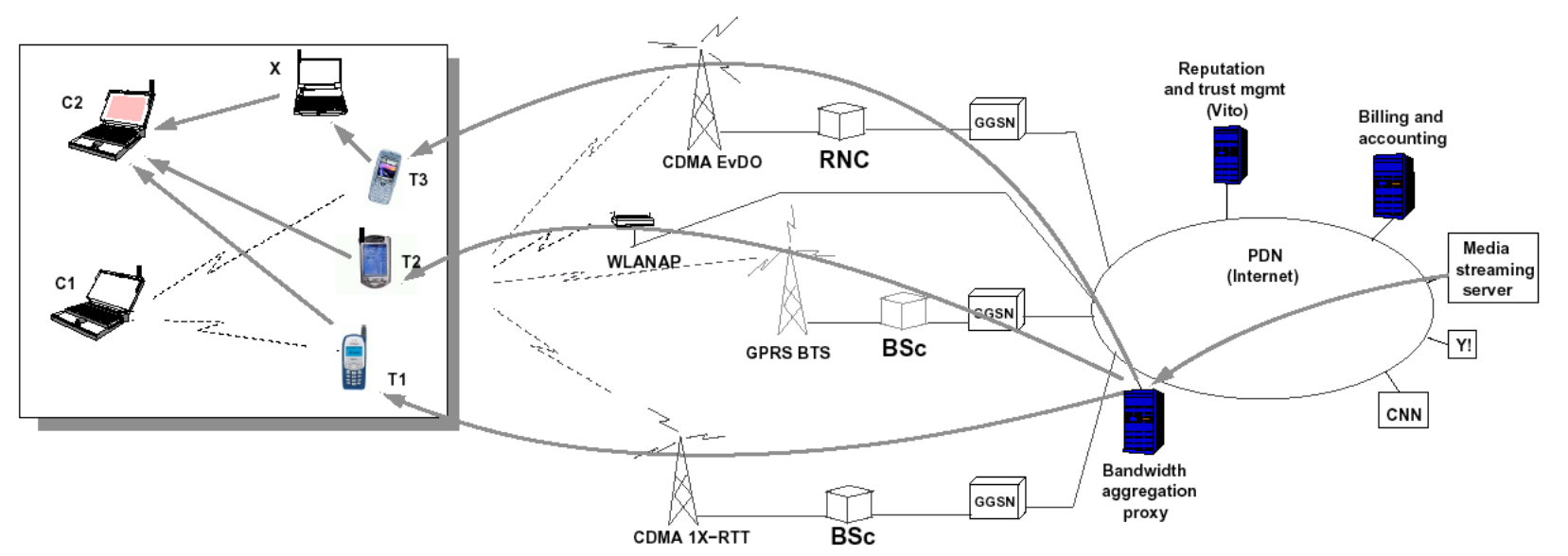
- There could be users who spend long durations in coffee shops! Significant opportunities for MoB interactions

An Example



- C1 is the customer device. Others (called traders)-- T1 is a 3G enabled cell phone, T2 is a PDA that connects through a WLAN Infrastructure (Boingo Wireless) etc.
- The traders are idle. C1 chooses a subset of the traders and uses them as if they were its own interfaces.
- C1 achieves high bandwidth due to aggregation, T1 and T3, their revenue.

Example Continued



- Notice -- we need Vito -- a reputation and trust management system, a Billing and Accounting System and a proxy that allows the application layer packets to be spliced between three networks.
- Also note that these are connected to the Internet.
- Finally, note that different networks are possible.

Services with MoB

- Bandwidth Aggregation (as seen before).
- Location determination -- a mobile user carrying a PDA with street map software can provide this to other users.
- Time Synchronization -- important in wireless gaming-- each gaming device connects to a cellphones using a common provider and acquire timing from these.
- Web Proxy Caching -- locally stored web pages may be faster to get.

More Services with MoB

- Media streaming -- high bandwidth video applications may be striped over multiple interfaces to improve user performance.
- Peer to Peer data search: A downloader from Kazaa or Gnutella may suffer poor connectivity and thus suffer loss of performance and monetary costs over cellular links.
- Traffic Filtering: The MoB trader can also filter and remove malicious content targeted at the end user.

Some Key Issues

- Single hop or multihop
  - Multi-hop possible. However, since MoB is application layer, the connections need to be independent on “each” of the hops.
- How would you go about doing pricing ? How would you choose the MoB traders ?
  - A system that is fixed -- they call the reputation system Vito -- similar to eBay.
    - Each user has an ability to rate the MoB trader with which it had an interaction.
  - The billing system is also on the fixed Internet.

Pricing and Reputation

- Individual traders can name their own price (as in eBay) --however, intelligent traders will consider the competitive forces in the market.
- The authors propose Vito -- a system that is based on what is in eBay.
  - The idea is that eBay has been largely successful due to its reputation management system.

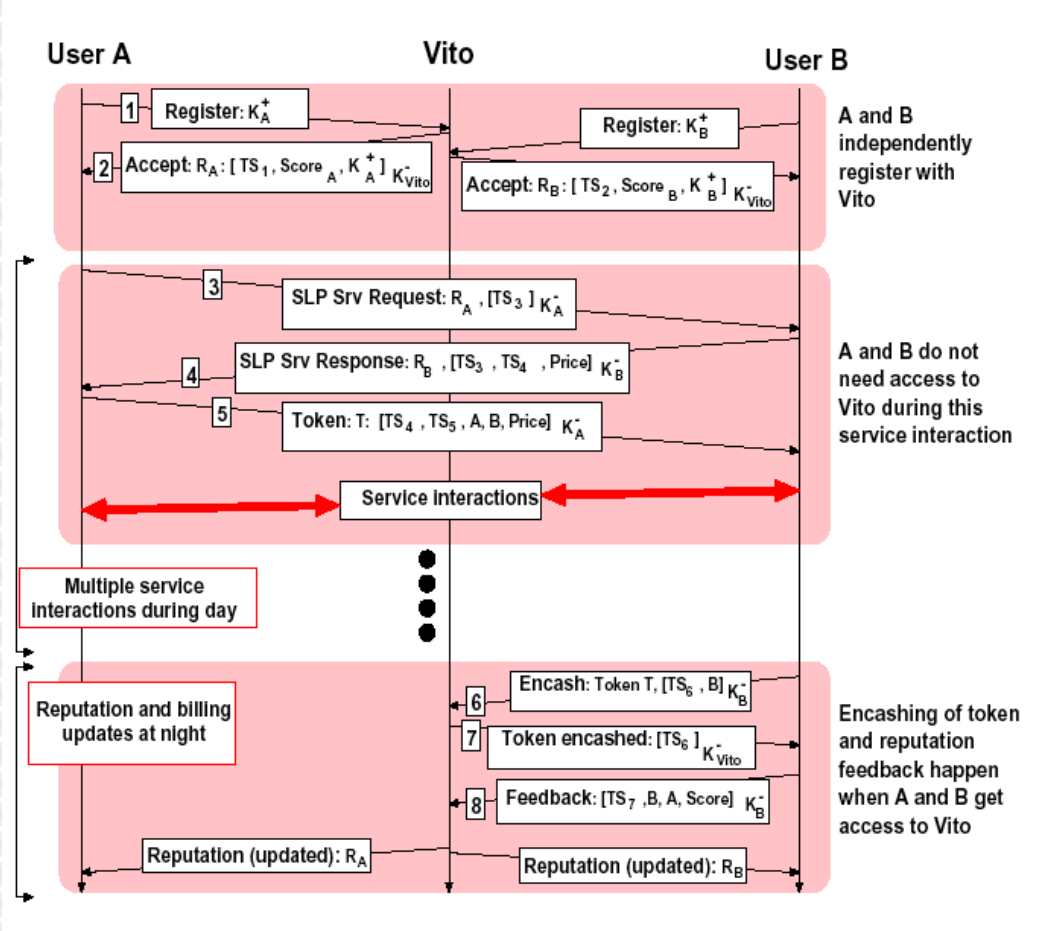
In summary

- MoB has the following features:
  - Open market architecture -- flexibility, fine grained service provisioning.
  - Better performance through Wireless diversity.
  - Incentive-based collaboration.
  - Customization and support for diverse applications.

The MoB architecture

- It has three basic components:
  - An infrastructure that allows wireless devices to connect to the Internet -- cellular, WLAN, or any combination thereof.
  - Mobile devices with the ability to communicate with “each other” and with the infrastructure. (Key!)
  - third party services for accounting and billing as well as for reputation and trust management.

Sequence of Operations



Reputation and Vito

- Each user registers with Vito and obtains a reputation certificate (has a timestamp).
- This certificate has to be provided to customers.
- Negotiated price can depend on reputation.
- Note that the reputations are then refreshed at a later time to Vito.
- Vito will periodically distribute reputation certificates.

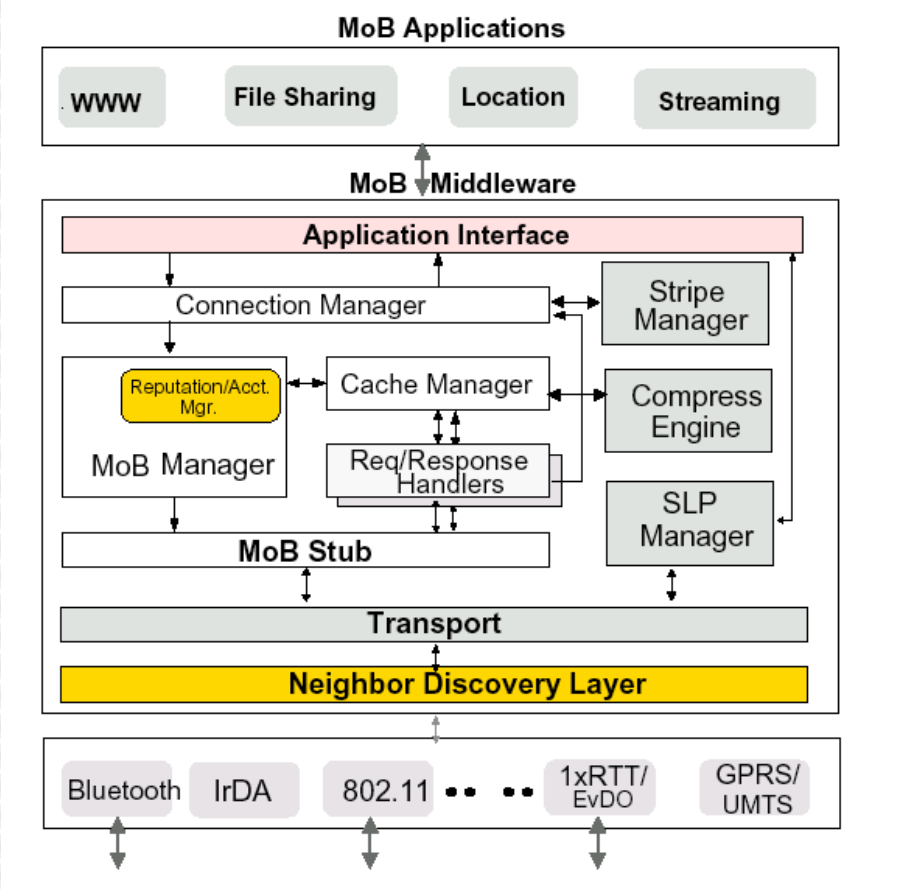
Service Location Protocol

- The authors suggest that the service location protocol be used for transactions.
- To request a service SLP Agent sends a service request to the SLP multicast address 239.255.255.253 and port 427 with a TTL of 1.
- A response includes the trader's certificate, a price quote and the service description.
- The customer can choose the right service based on the responses received.
- Straightforward -- I won't go into it.

Feedback

- Again straightforward.
- Payment in the form of a token prior to service (else, it is difficult to determine if the customer will pay).
  - If service is not delivered, can report to the billing and accounting system.
- Each party has an opportunity to rate the other on Vito.
- Vito charges a transaction fee -- can discourage fake reputation uploads etc.

Implementation



- Implemented as a middleware.
- Connection manager -- provides the interface with applications.
- Cache manager -- check for locally stored objects.
- Stripe manager -- stripes between different networks.
- Transport layer typically TCP

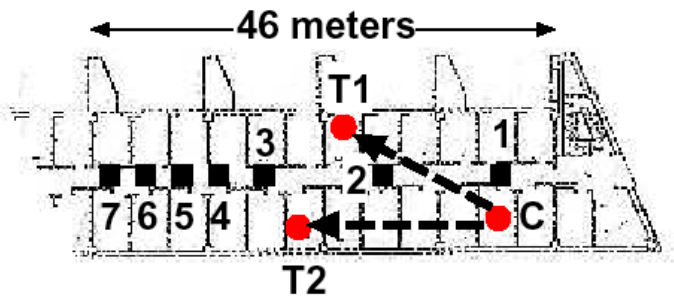
Neighbor Discovery

- Via link-specific mechanisms as provided by various wireless interfaces.
- In 802.11, the authors set aside one channel for neighbor discovery.
  - In tranquil state, MoB device monitors traffic on this channel --announcements are sent as well periodically -- SSID set to MoB and the mode to ad hoc.
- With Bluetooth (the authors perform experiments with this) -- the device initiates a scan procedure to detect other MoB devices at large.
  - I will skip the few details that are provided in the paper.

Evaluations

- A full implementation.
- Bluetooth, 802.11a, 802.11b and 3G EVDO and 3G 1xRTT data services were also used.
- They look at File transfer, Web browsing and data aggregation via media streaming.
- They also do simulations for other things (such as location determination).

The set up

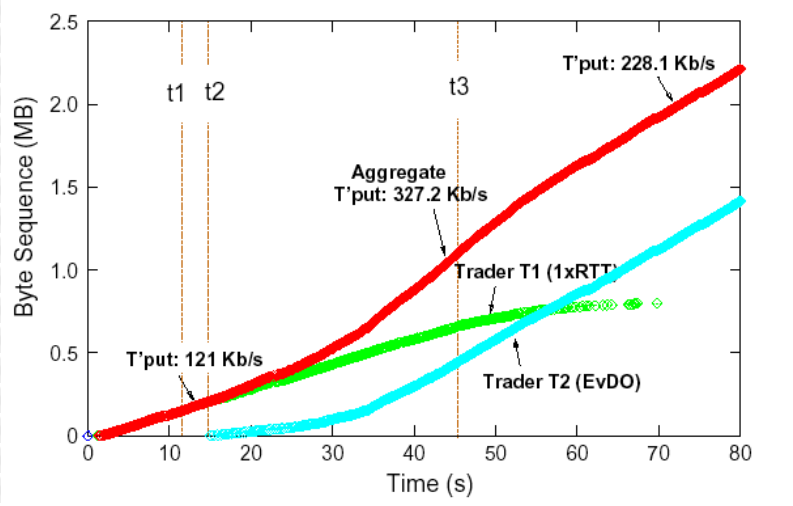


Location	Latency (ms)	Loss(%)
1	43.4	0
2	49.5	0
3	56.4	0
4	67.6	1
5	78.0	2
6	81.1	2
7	-	-

Figure 5: Floor-plan of building for MoB experiment scenarios. C is customer, T_1, T_2 are traders (positions changed in different experiments as described in this section). The table indicates the latency and loss characteristics obtained in this environment using a source at C and destinations varying from $1-7$ using Bluetooth.

Example Scenario 1

- Single customer C, using a Bluetooth wireless interface with two mobile traders T1 and T2. T1 has a CDMA 1xRTT interface (max rate 144 Kbps and uplink data rate of 64 Kbps) and T2 has a CDMA -EVDO (maximum downlink data rate of 2.4 Mbps and uplink data rate of 153 Kbps).
- Initially, only T1 is in range of C
- Later T2 moves into range.
- Finally, T1 moves out of range.

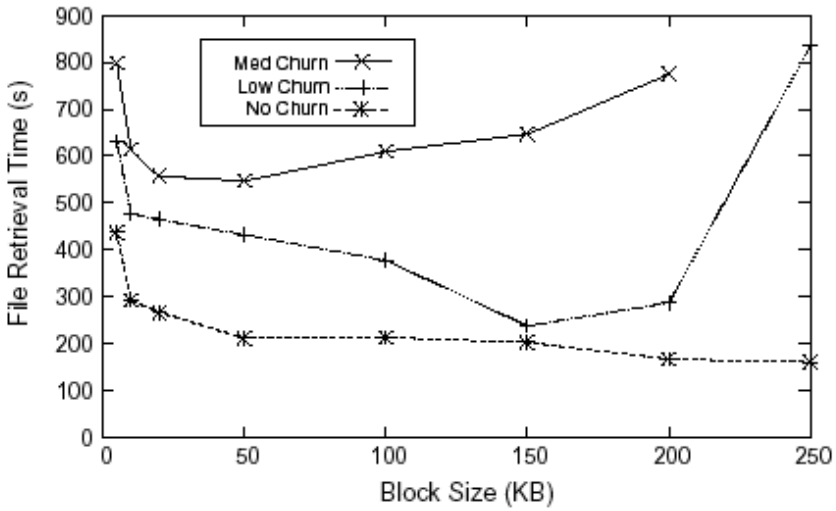


Example Scenario 2

- The idea is to show the impact of mobility (churn) on Gnutella and Kazaa applications.
- High churn -- potential trader stays in customer's range for 10-20 seconds.
- Medium churn -- trader stays in range for 40-60 seconds.
- Low churn -- trader stays in range for 60-120 seconds.
- No churn.
- Customer locates a trader device that has the queried data object and starts block-based data download.
- If trader moves out, then, a search occurs for a new (alternate) trader.
- In their study they use a 5 MB (audio MP3) file -- block sizes are varied.

Example Scenario 2 (cont)

- Block size important.
- If it is large, then customer pays all at once and if there is a problem, then liability becomes an issue.
- Second, smaller block sizes can lead to more parallel downloads.
- Note -- optimal block size depends on churn. If no churn, a smaller block size -- higher overhead -- more time taken.
- For higher churn, there is an optimal since if block size is large, trader may move out.



The impact of user churn and block size variations on peer-to-peer object download (Scenario 2: only one trader allowed per customer). Interaction using single Bluetooth interface.

Other experiments

- I won't go over other experiments.
- However, they are similar in spirit.
- The authors also look at how web downloads can be done more efficiently using collaborative caching (nodes have cached pages).
- They do simulations with regards to location accuracy -- how can they obtain location info from GPS equipped devices -- for this simulation, it seems that they simulate the Manhattan street network -- roads and vehicles in mid-town Manhattan.

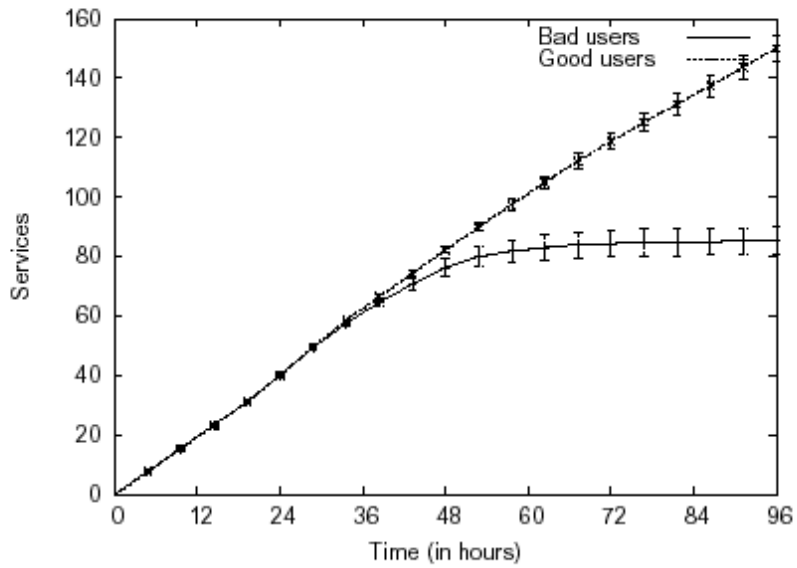
Evaluating Vito

- Based on simulations.
- Different policies considered for choosing traders:
  - Reputation
  - Price
  - Price with reputation threshold
  - Ratio of Reputation to Price.

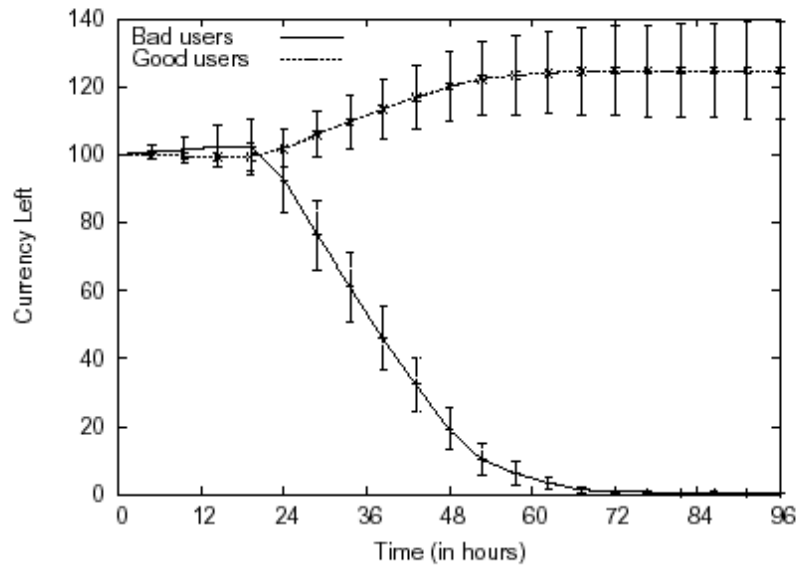
Price Changes

- A price reduction factor γ and a price increment factor λ.
- When a user does not get chosen, he lowers the price p by the factor γp for the next service offering.
- Similarly, if a user gets chosen, he increases the price by λp for the next offering.
- Note -- the values of γ and λ are arbitrary.

Sample Results



A time evolution of services obtained by good and bad users in MoB based on Vito's design policies.



A time evolution of currency left at good and bad users in MoB based on Vito's design policies.

What did I learn ?

- A complete system design taking into account the intricacies of system interactions (even if some details are missing) is of interest.
- Application driven -- could be something that we want to think about.
- What are the implementation challenges?
  - TCP
  - Impact of mobility on striping

Comments ?