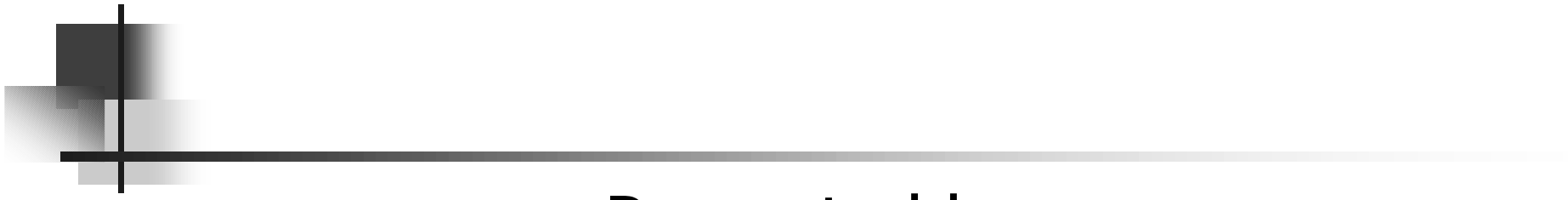
Mojo Nation



Presented by
Dongning Tian
Hyun Chul Lee

2001 P2P Seminar

Outline

- Introduction
- Service/Application
- Architecture
- Protocol
- Comparison
- Conclusion

Introduction

- Mojo Nation is a new P2P content distribution technology.
- Mojo Nation = Protocol + Application.
- It was created by Jim McCoy who founded Evil Geniuses.
- Beta Version 0.994.7 was released in August 16, 2001 with 2000 users.

Introduction (cont'd)

- One philosophy of Mojo Nation is

  "Lazy cyber parasites won't have a chance in in Mojo Nation"

- To achieve this goal*, MOJO* is introduced.
- Mojo Nation is decentralized.
- Mojo Nation is ***P2P***.

Application/Service

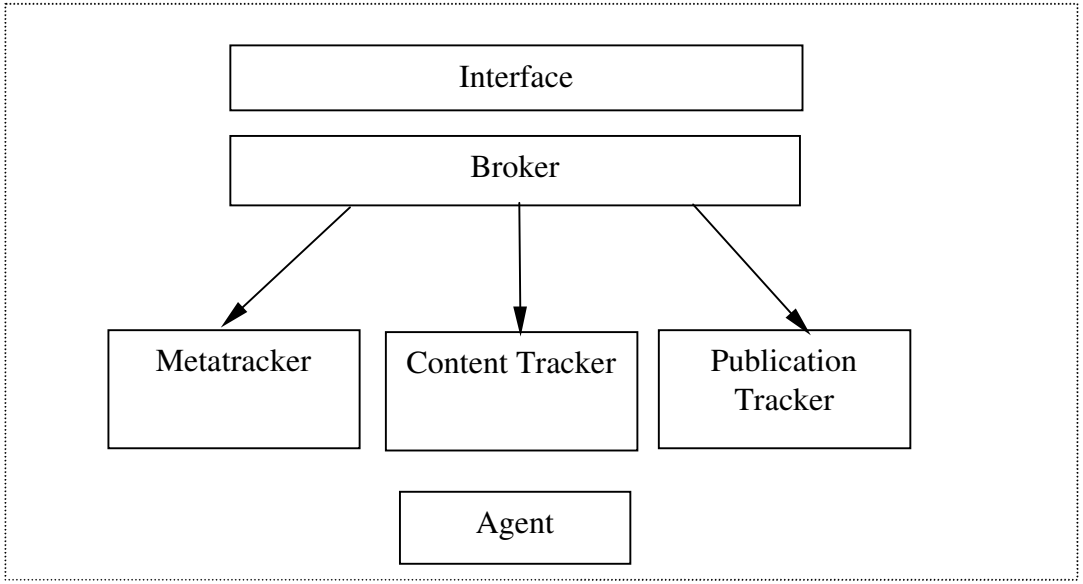
- With Mojo Nation you can exchange any kind of files. (images, audio, webpages, text files, adult material)
- In Mojo Nation, every transaction has some cost.
- Mojo is Mojo Nation's digital currency.
  - If you provide some service then you earn mojo.
  - If you get service then you lose mojo.

Application/Service(cont'd)

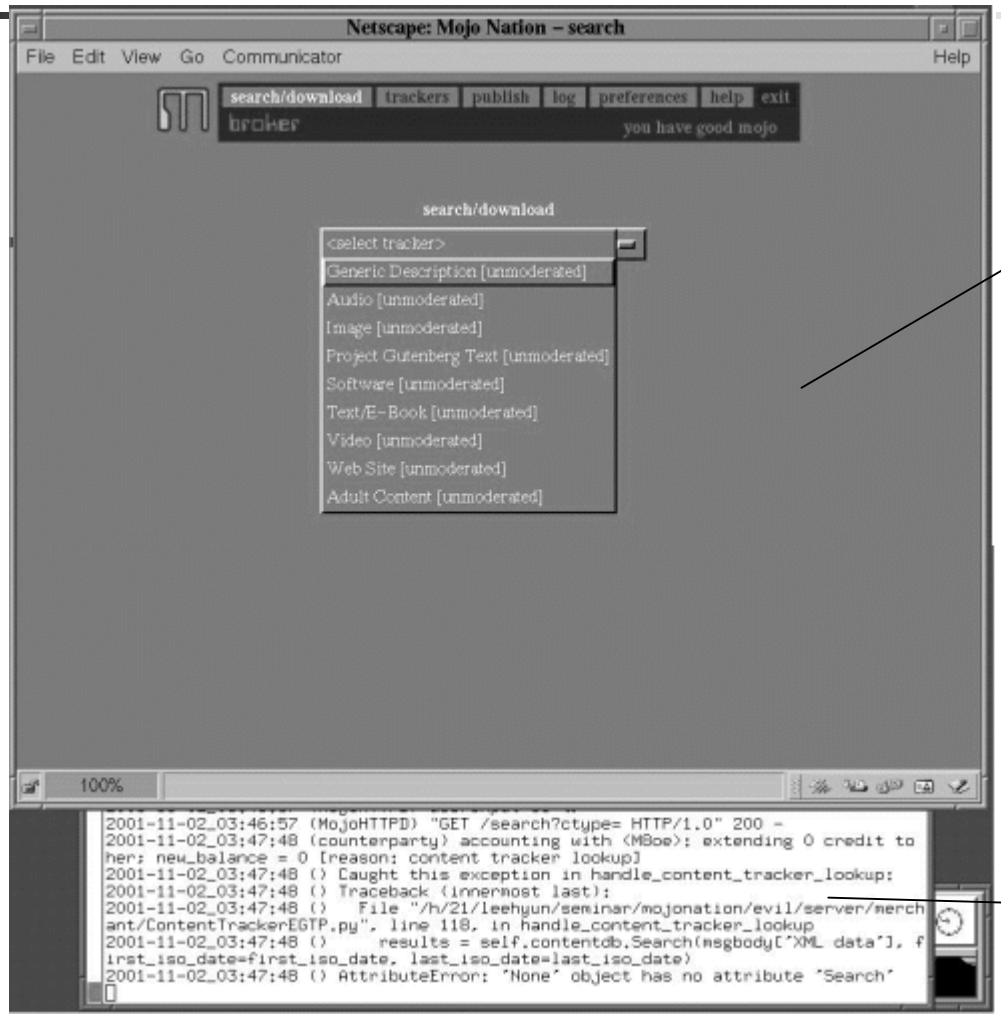
- Each agent(host) supports 3 types of services.
  - Download service
  - Upload service
  - Sharing of its resources for other agents(hosts)
    - Disk Space (e.g. block server)
    - Bandwidth (e.g. distribution of data)
    - Processing cycles. (e.g. relay server)

Architecture

- Architecture of Mojo Nation Software



Mojo Nation software



Interface

Broker

Core technology of Mojo Nat.

- Swarm Distribution
- Micropayment and distributed Load Balancing
- Relay service
- Reputation system

Swarm distribution

- File is fragmented into several pieces for storage and retrieval.
- The fragmented pieces are distributed over the network.
- When an agent requests a file, many agents participate in the collaborative effort of data delivery.

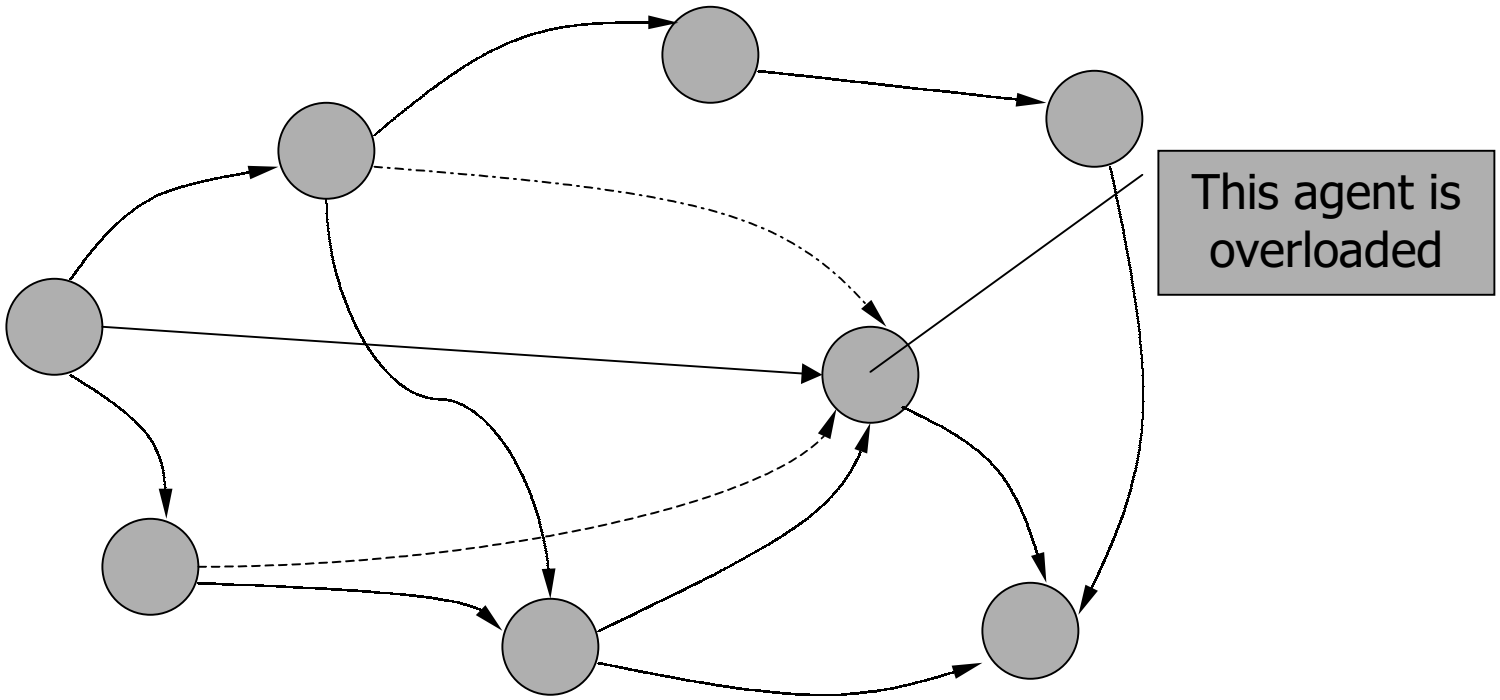
Micropayment and Distributed Load Balancing

- Each interaction between agents involves some offer of Mojo
- The system is resistant to “denial service” vandalism
- When demand for a certain resource of an agent is high, the agent may bog down or grind entirely to a stop.

Micropayment and Dist. Load Balancing (cont'd)

- By keeping score of what resources are contributed by agents within network, Mojo Nation locks-in customer resources
- Agents who have contributed resources to the system and got "score" will a choice of moving to the head of the line for that resource, and the clients automatically migrate to less-occupied servers

Example of Distributed Load Balancing



Relay Service

- Many potential Mojo Nation users cannot link to the system because of the firewall or other reasons
- A relay server enables these customers to use the Mojo Nation by holding their system messages outside the firewall until the user's Broker makes an outgoing connection to retrieve them

Reputation system

- Each agent maintains its own local database of reputation for other agents, including a list of others with which it has done business and information about these transactions
- For instance, if one agent tries to cheat another by not delivering information, the requesting party will lessen the local reputation of the other, and complete the desired transaction with a different agent

Protocol

- Transport layer
  - TCP/IP, if it is available
  - Other options when it is not.
- Encryption and authentication layer
  - Provide security communication
  - Using RSA algorithm to encrypt messages
  - Guarantee authenticity by digital signature

Protocol

- Conversation layer
  - Two types of messages, initiating and responding
  - Initiator sends initiating message only once
  - More than one responding message

Protocol

- Mojo transaction layer
  - Every conversation involves an offer of Mojo
  - Initiator send initiating message and Mojo offer
  - Respondent checks service price and initiator's credit limit

Comparison

- No quantity report
- Scale better
  - Swarm distribution aggregates the bandwidth of hundreds or thousands of low –bandwidth agents.
- Using economic incentives

Conclusion

- Mojo Nation combines the flexibility of demand-driven marketplace with a secure “swarm distribution” mechanism that goes far beyond any current file-sharing system
- Efficient, massively scalable and secure toolkit.