Monitoração de Tráfego Par-a-Par em Tempo Real

Tiago Alves Macambira

Orientador: Dorgival Olavo Guedes Neto

Co-Orientador: Wagner Meira Jr.

Departamento de Ciência da Computação
Universidade Federal de Minas Gerais

Era uma vez...

- Pré-*Web*
  - Pesquisas sobre aspectos e baixo nível
  - 80% do tráfego mera devido a FTP e SMTP
- *Web*
  - mudança de paradigma
  - 1995 $\rightarrow$ 21%
  - 1997 $\rightarrow$ 60%–80%

Web no Divã

- Trabalhos de
  - Caracterização
  - Monitoração
  - redução
- Caches, proxies, hierarquia de caches, proxies transparentes

... e sejam feitas as redes P2P

- Nova Mudança de paradigma de tráfego
  - Em 2000:
    - 23% $\longrightarrow$ P2P
    - 20% $\longrightarrow$ *Web*
  - em 2003 $\longrightarrow$ 45%

Sistemas P2P

- Vários tipos de serviços
  - Até ICQ é considerado P2P!!!
- Características marcantes
  - troca de recursos entre nós similares
  - conectividade (ou disponibilidade) instável e variável

P2P no divã

- Mais trabalhos de caracterização
  - volume de tráfego
  - identificação de tráfego P2P
  - caracterização de buscas
  - distribuição e popularidade de recursos
  - disponibilidade

Monitoramento de P2P

- P2P ainda está em constante mudança
  - Novos sistemas
  - Novos estudos
- Monitoração
  - ativa
  - passiva
    - *Off-line* ou
    - em tempo real
      - Com recuperação de estado

Monitoração ativa

- Problema
  - grande consumo de recursos
  - interferência
  - protocolos fechados
  - *firewalls* e NAT
- Não queremos ser intrusivos

Monitoração passiva

- Formas de captura
  - por espelhamento
  - por interceptação
- Vantagens
  - não causa interferência
  - todo tráfego pode ser monitorado
  - pode ser usado com protocolos proprietários

Monitoração passiva

- Desvantagens
  - sobrecarga nos equipamentos de rede
  - velocidade de captura e análise
  - horizonte de monitoração limitado

"Sim mas... E dai?"

- O que queremos?
  - Ferramenta que viabilize a análise e a caracterização de tráfego de aplicações em tempo real, tendo como foco os aplicativos P2P de troca de arquivo.
- Como?
  - Sistema de monitoração passiva de tráfego com recuperação de estado em tempo real
  - Altas-velocidades?
  - Usando *software* de código aberto?
  - Sem gastar muito dinheiro?

Atividades Relacionadas

- Identificação de tráfego
  - assinaturas
- NIDS
- Roteamento, NAT e filtragem de pacotes

Captura de Pacotes

- Necessário para monitoração passiva

- Etapas na captura de um pacote:

  1. Recepção pela NIC
  2. Recuperação pelo *driver* da NIC
  3. Filtragem dos pacotes
  4. Cópia do *kernel-level* para *user-level*
  5. processamento do pacote pela aplicação

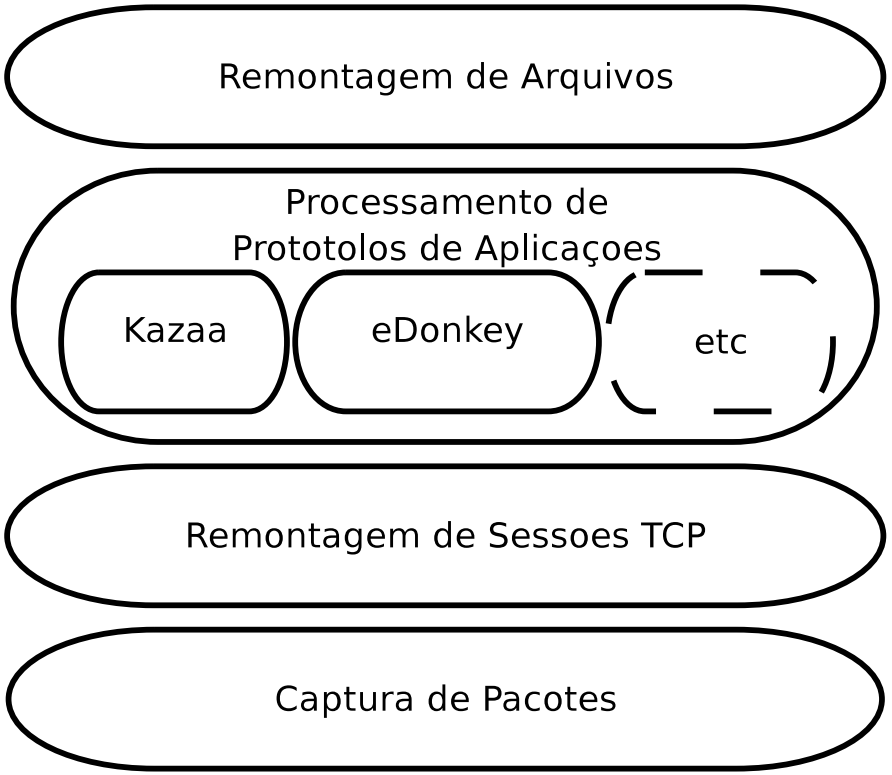
Captura em enlaces de alta-velocidade

- Problemas:
  - Processador
  - Barramento PCI
  - Memória
  - Discos rígidos
- Abordagens
  - Processar parte do todo ou o todo de um parte
  - *Software* genérico e *hardware* especializado
  - Dividir para conquistar

Recuperação de estado

- O quê?
  - Fluxos TCP
- Para quê mesmo?
- Recuperação de estado em tempo real

O *Palantír*



Organização em camadas do *Palantír*

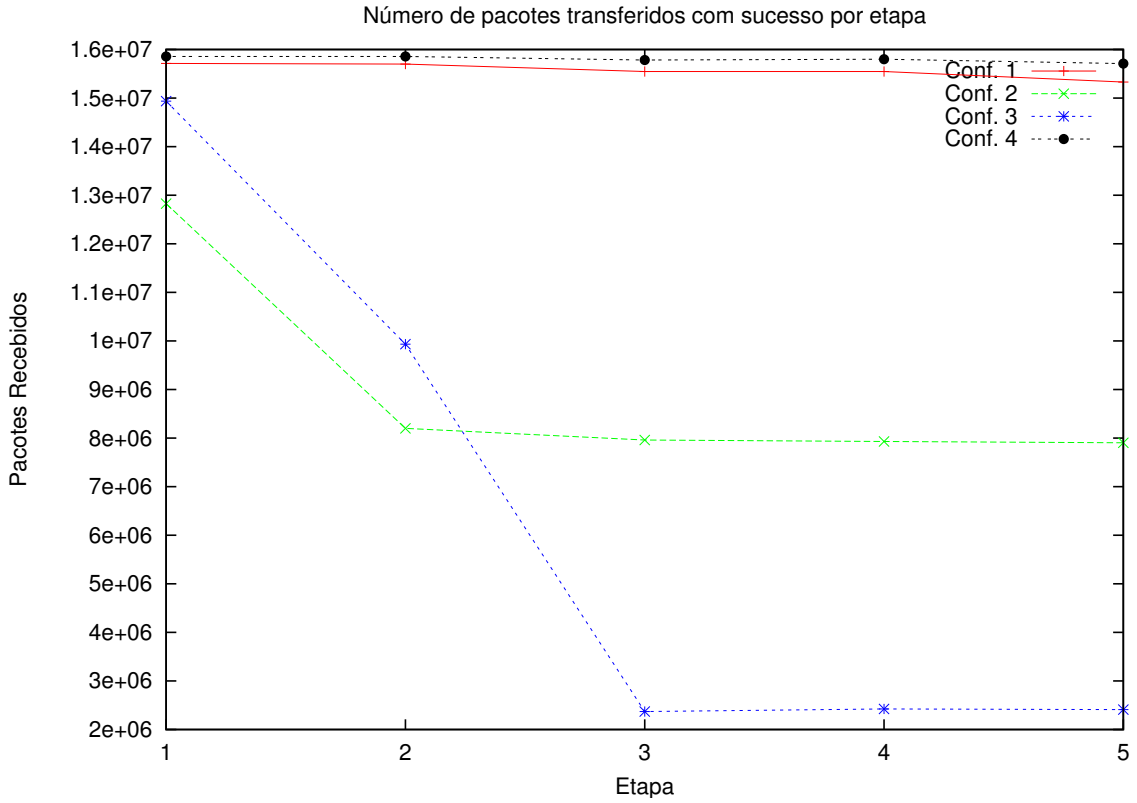
O *Palantír*

- Abordagens utilizadas
  - Captura: PF_RING
  - remontagem de sessões: libNIDS
  - Processamento do protocolo de aplicações:
    - Kazaa
    - eDonkey

Avaliação de desempenho

- Perda de pacotes
- 3 máquinas
- 500Mbps
- 5 etapas
- 4 configurações
  - PF_RING
  - Hyper-Threading
  - *Packet pooling* no *driver* da Intel Pro/1000 (NAPI)

Avaliação de desempenho



Quantidade de pacotes capturados

Avaliação de desempenho

- PF_RING realmente tem efeito:
  - diferença entre a melhor configuração com PF_RING e a pior sem PF_RING:
    - de 19,1% na primeira etapa
    - de 83,8% na última etapa
  - PF_RING no pior caso: perda de 3,32% dos pacotes
  - "Degradação" da captura não é abrupta
- Peculiaridades do *Hyper-Threading*

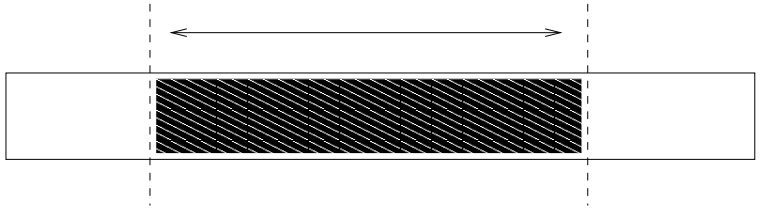
Sistemas P2P de troca de arquivo

- Redes monitoradas
  - Kazaa
  - eDonkey
- Aspectos importantes (ou que influenciam no trabalho de monitoração)
  - Identificação de recurso
  - Modelos de transferência de arquivos
  - Organização e localização de recursos na rede

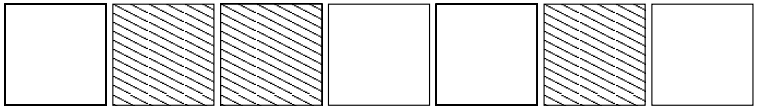
Identificação de recursos

- Identificação *vs.* Localização
- URLs
- Identificadores desvinculados de localização
- Hashes

Modelos de Transferência



Segmentado



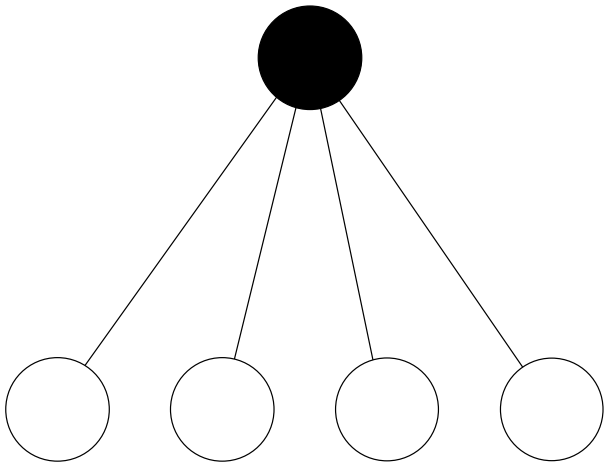
Fragmentado

Localização de recursos

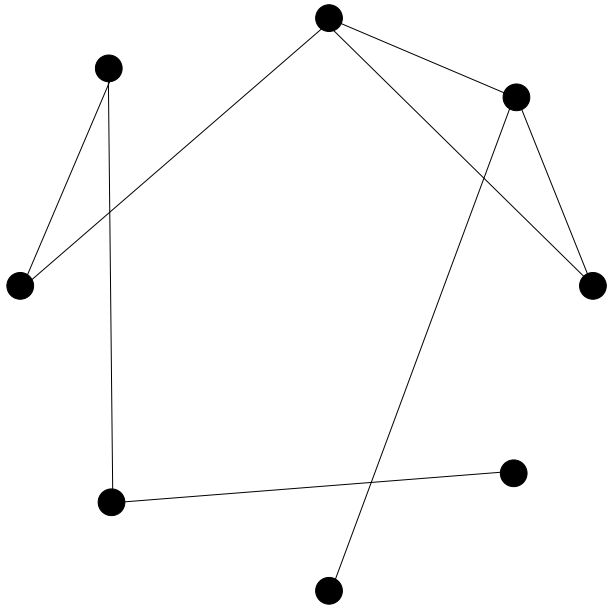
- Etapas

  1. Descoberta de recursos (busca)
  2. Busca por fontes

Organização da rede



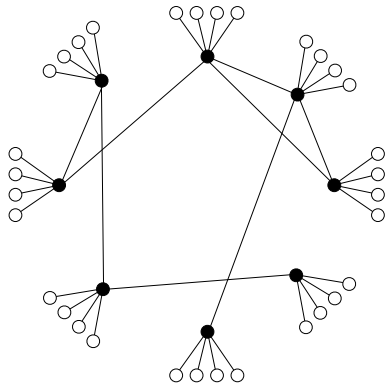
Rede Centralizada



Rede Descentralizada desestruturada

Organização da rede

- Descentralizada semi-estruturada



- Descentralizada estruturada

KaZaa

- Rede FastTrack
- Rede semi-estruturada - *SuperNodes*
- Protocolo proprietário
  - Buscas: cifradas
  - Downloads: HTTP
- Portos variável

KaZaa

- Transferência segmentada
- ContentHash, UUHash e UserNames
- Poluição (50%)

eDonkey

- Rede Centralizada
  - redes descentralizadas estruturadas auxiliares
- Protocolo proprietário
  - binário, mesmo para buscas e para transferências
- Portos bem conhecidos
- Transferências Fragmentadas
- FileIDs, MD4 e UserIDs
- Localização de recursos em 5 etapas

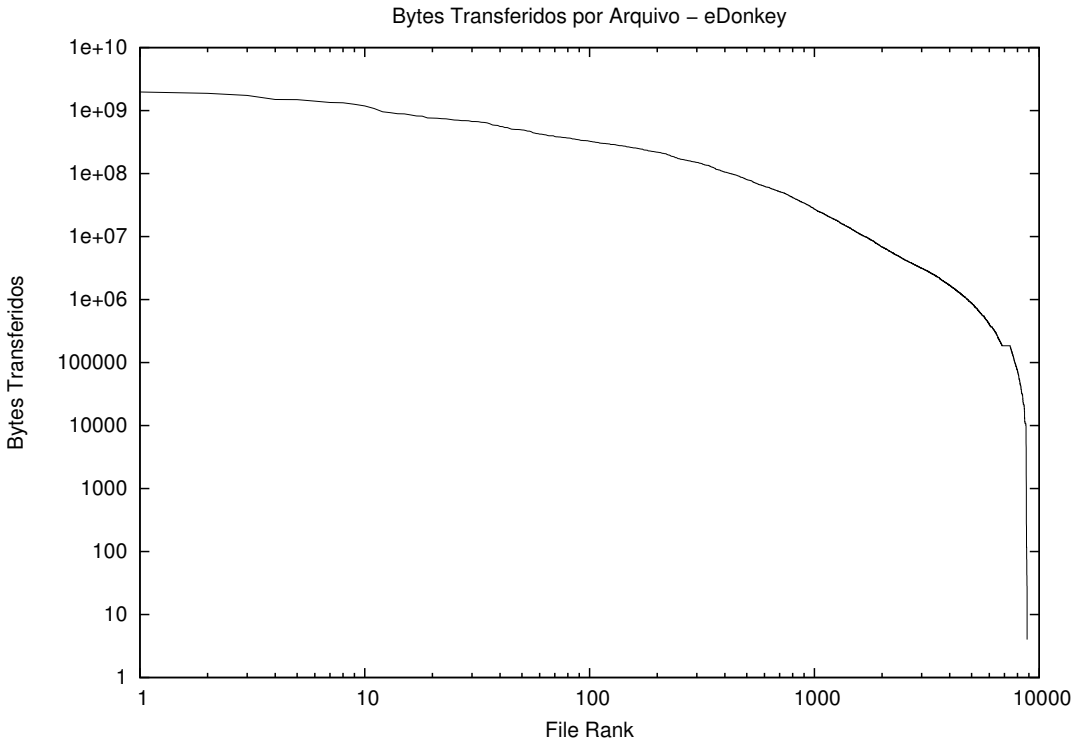
Caracterização de tráfego

- 20 dias

Rede P2P	Kazaa	Edonkey
Intervalo	2004/10/18-28	2004/10/18-28
Bytes transferidos	1.644.589.908	175.419.189.687
Requisições	8.490	59.324
Recursos únicos	3.042	8.835
Sessões	5.512	53.020
Usuários únicos	4.388	48.206

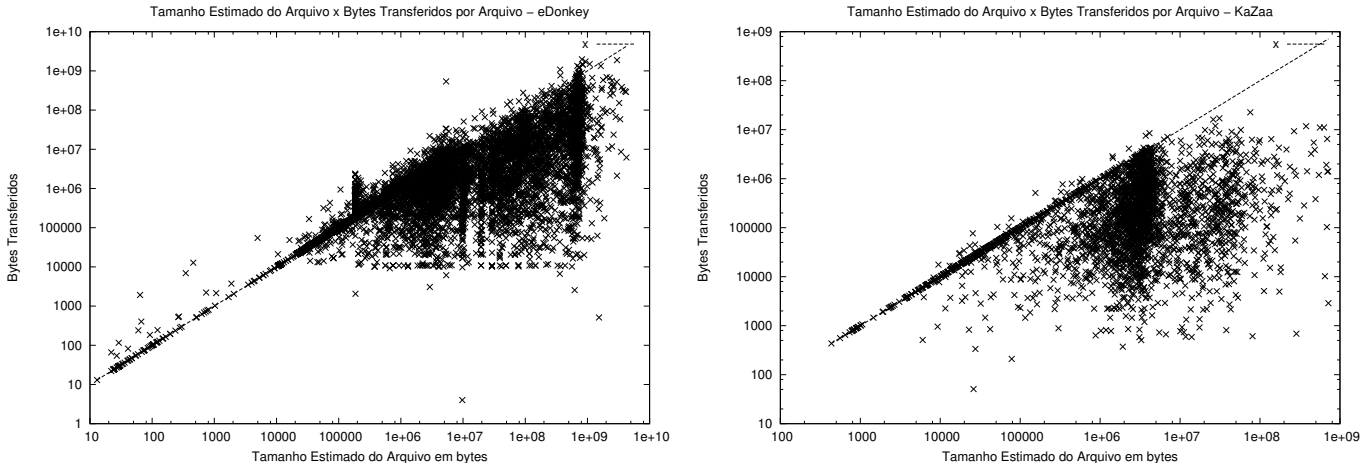
Estatísticas gerais observadas

Distribuição de *bytes* transferidos por recursos



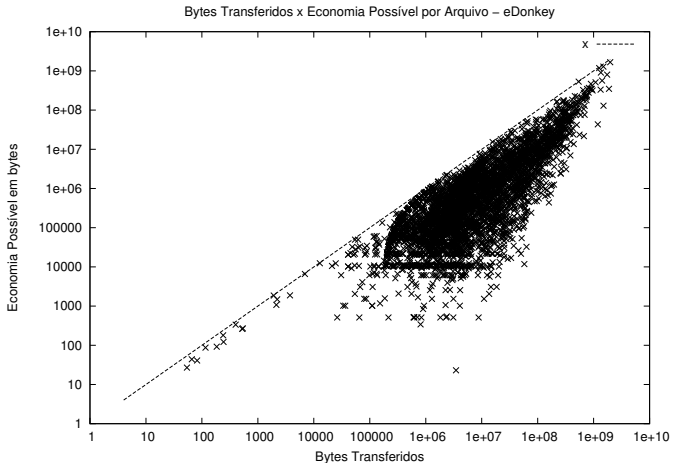
Distribuição de *bytes* transferidos por recurso.

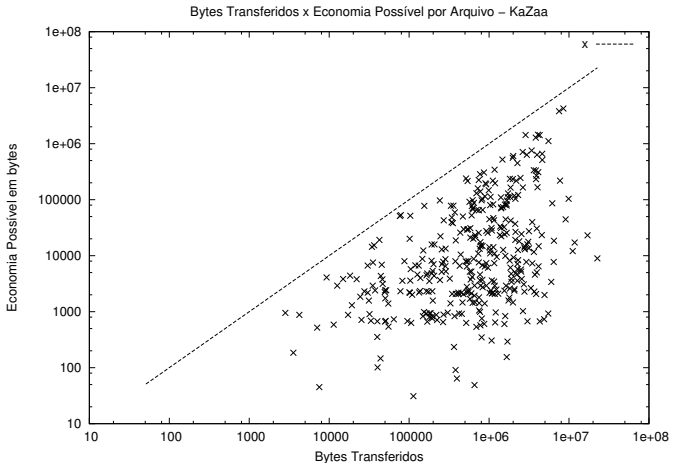
Bytes transferidos *vs.* tamanho dos arquivos



Correlação entre *bytes* transferidos e tamanho dos arquivos

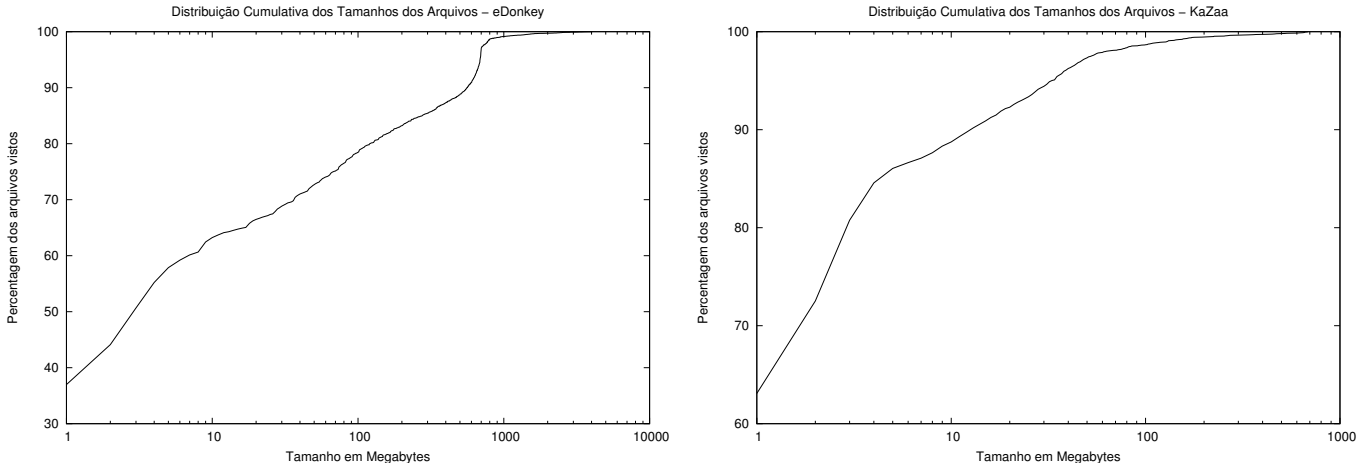
Badwidth Savings





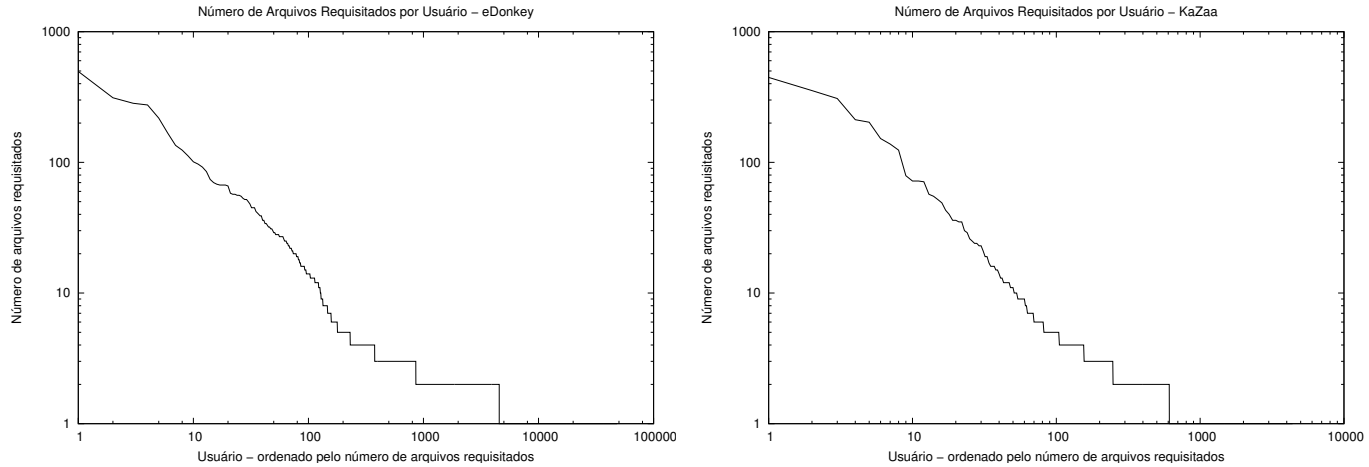
Correlação entre bytes transferidos e os ganhos potenciais de economia em largura de banda

Tamanho dos arquivos



Distribuição cumulativa dos tamanhos dos arquivos

Número de recursos solicitados por usuário



Distribuição do número de recursos solicitados por usuário

Localidade de referência inter-protocolo

P2P	Recursos	Sig1	Sig2	Sig3	Recursos Comuns	Bytes Comuns
Kazaa	3042	1481	288	54	8.20%	12.63%
Edonkey	8835	6211	2118	829	5.25%	7.04%

Localidade de referência entre o KaZaa e o eDonkey

Conclusões e trabalhos futuros

- Sim! É possível fazer o que queríamos sem *hardware* especializado
- Trabalhos Futuros
  - Melhorias de desempenho
  - Outras redes
  - Análise do tráfego de sinalização
  - Construir um *cache* oportunístico
  - Controle ou vigilância de tráfego
  - Dinâmica dos fragmentos

Fim

Perguntas?