• Debian, Ubuntu, lots of users
• Distributed
• Users fetch the latest
• ... usually at the same time
• Saturation
• ... send less bits
• Compression
• ... already **done** (gzip, bzip2)
HTTP raw download

HTTP pre-compressed

Server CPU usage
Network data transfer
Client CPU usage

Net
CPU

Net
CPU
• Delta-encoding
• ... known versions only (deltarpm)
• ... or dynamic (rsync)
• Edit script
  – ADD("...") literals
  – COPY(length, position)
  – EOS()
rsync client

file alpha

MD4  MD4  MD4  MD4

rsync server

file beta

MD4
MD4
MD4
MD4

checksum match?
• Where are you?
  – I'll send you directions.

• Here's a map...
  – Find your own way!
• Reverse rsync
  – Precalculated digest ("cached")
  – No server side
  – Unknown local version v0.9, v1.0, v.1.0special
  – See rdiff, zsync
• Restrictions

  – Existing dumb mirror network (HTTP/1.1)
    • no server side
  
  – Any version to latest (avoid n^2 patches)
    • reuse literal data
  
  – Scalable: CPU entirely on client
needed ranges

compression 'sectors'
• Not bump disk space usage
  • academic mirrors

• .Bit-for-bit reconstruction
  • GPG signatures

• .deb is not a 'normal' file
  • offsets of real data
.deb file

raw offset
• Reconstruction
  • Deterministic

• Any decision/choice is not deterministic
  • record the choice

• Big list of decisions
  • reduce, by diffing against a model (eg. zlib -9)
• DEFLATE (rolling)
  – gzip, pkzip, png, pdf...
  – 32kB LZ string match, **Huffman**
  – *rolling*

• Bzip2
  – 900kB BWT
  – RLE, **BWT, MTF**, RLE/Huffman
  – *block*
Paul Sladen
Nineteen Inch
Questions?