

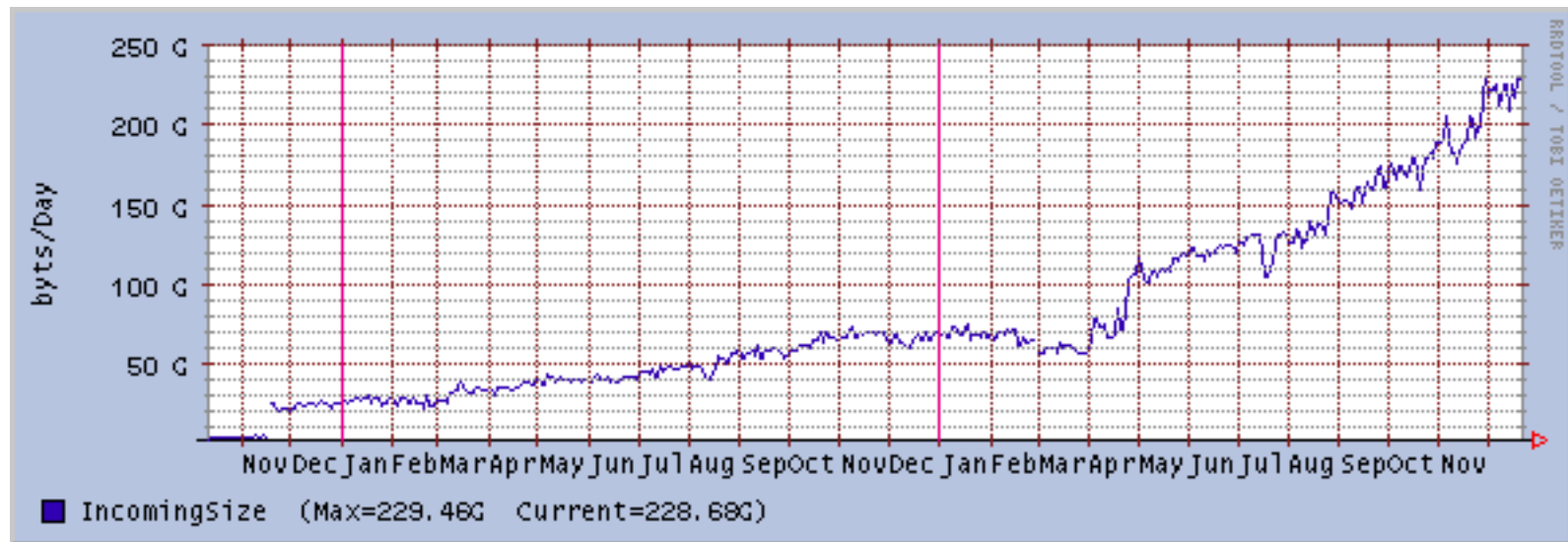
Net News の効率的な配送と Top 1000

kondou@isc.org

ターゲットサーバ

- ・ 配送経路上のトランジットとなる（≠末端、リーダ用）
- ・ INN 2. 3以降
 - diabloやCycloneなサーバは対象外
 - ▷ この際INNに乗り換えてみては？
 - ▷ INNで困ったときにはinn-workers@isc.orgをはじめとして
news.software.nntp、fj.news.system.inn等
経験 豊富なニュース管理者がきっと知恵を貸してくれるはず
 - ここで紹介することは経験的なものも含まれるし、サーバ環境にも
よるのでかならずしもすべてがあてはまるわけではない

Net Newsのトラフィック傾向 (1)

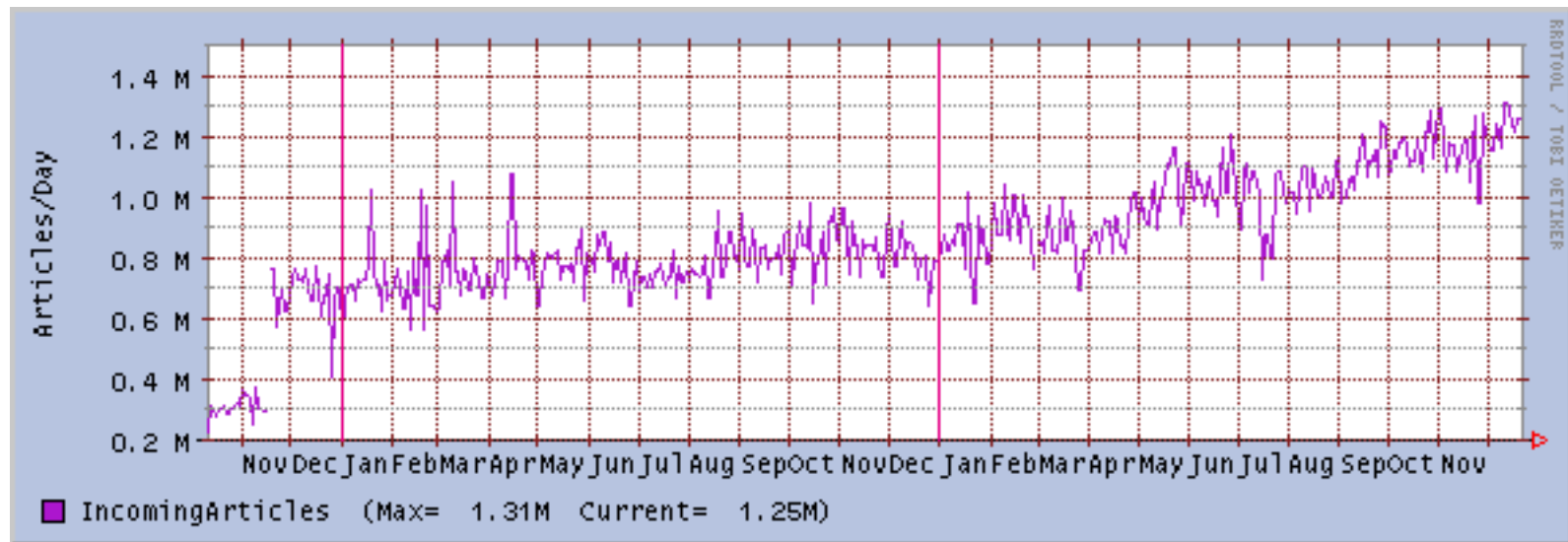


2000年12月中旬現在で220GB/日



約7.5ヵ月で倍のペース

Net Newsのトラフィック傾向 (2)



2000年12月中旬現在で125万記事/日



約26～27ヵ月で倍のペース

Net Newsのトラフィック傾向 (3)

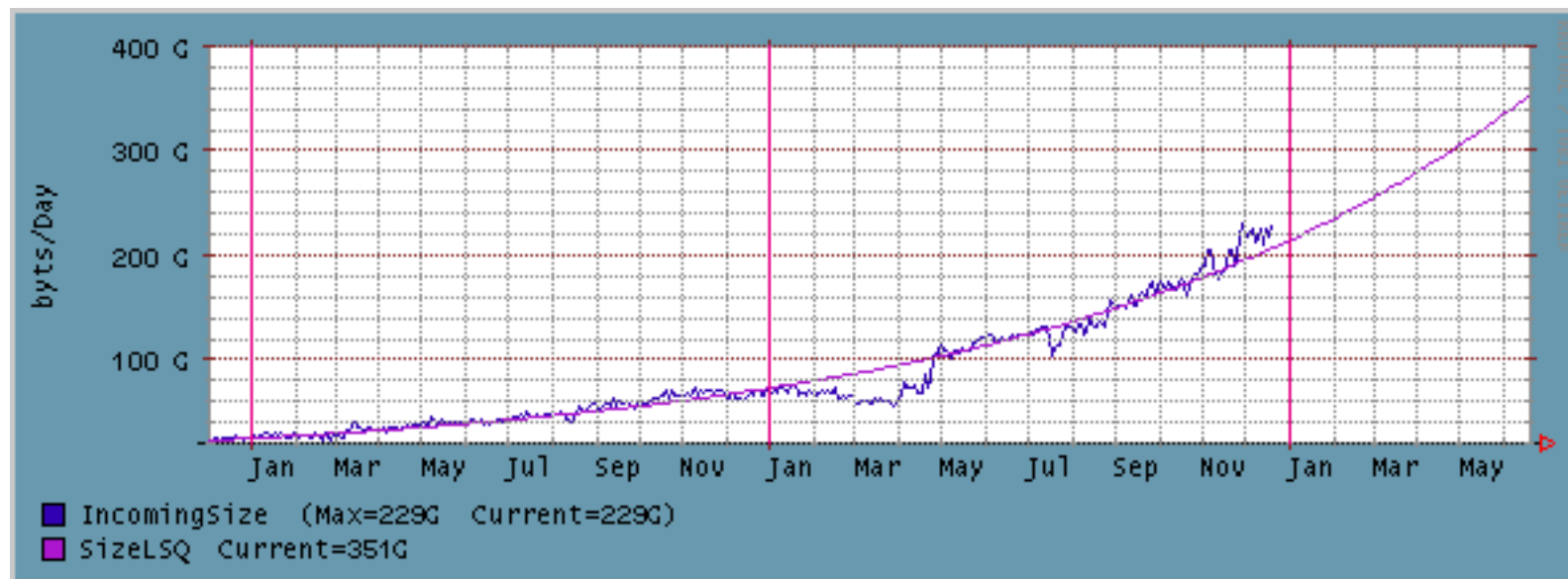
- ・ 記事量の95% (200GB/日) はalt
- ・ 記事数の60% (72万記事/日) はalt
- ・ キャンセルは記事数で5~12% (~15万記事/日) 程度



キャンセルはサーバの挙動に影響を
るほどの流量でもない

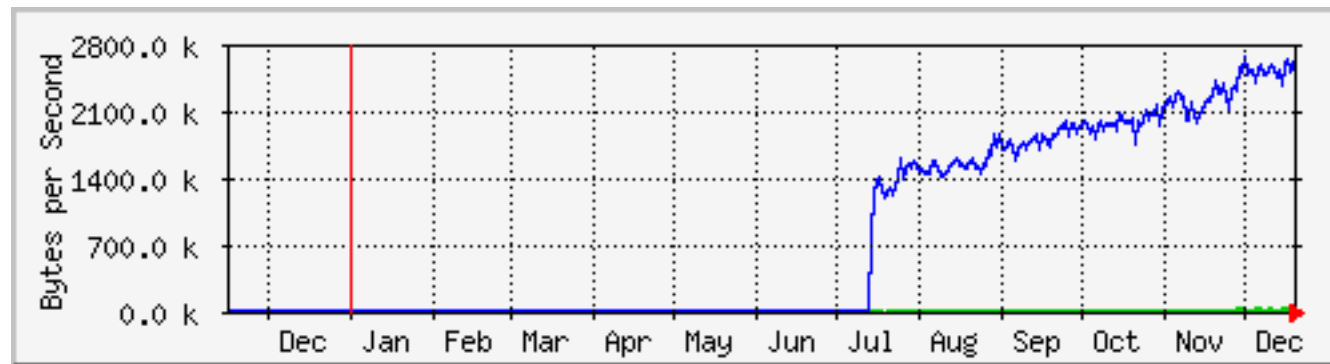
与え

Net Newsのトラフィック傾向 (予測1)



- ・ 2001年5月ぐらいには300GB/日

Net Newsのトラフィック傾向 (予測2)



leafサイトへのfull feedの場合
2001年4月ぐらいには
30Mbps程度の帯域が必要

効率的な配送とは？

- ・ 配送量のわりにサーバの負担が軽い
- ・ 帯域をムダに浪費しない

サーバの負担を軽くする (1)

- ・ 配送用とリーダ用でサーバを分割

- `overview` 生成は配送用サーバにとっては負担
- `overview` は配送用サーバに不要

- ・ `storage method` は `CNFS` を使う

- `open ()` / `close ()` のオーバヘッドがサーバを重たくする

- ・ `$pathlog`、`$pathdb` および `spool` 配下はそれぞれ 別スピンドルとする

サーバの負担を軽くする (2)

- ・ `innod` の起動オプションに ' -u ' は付けない

▷ `innod` 自身が出力するログはバッファして出力させること

- ・ `newsfeeds` の各エントリ名はラベルとして使用し、
できるだけ短い名前にすること (by Joe Greco)

▷ `path` ホスト名の平均文字数が 20 で配送先が 50 ある場合には `$pathlog/news` には 1 エントリあたり 1000 バイト程度になり `innod` 出力時および `news.daily` 時の
ログの処理の負荷が上る

▷ 同様に `innfeed` に渡すデータ量を減らすことができ、`innfeed` への標準入出力
バッファの浪費を抑えることができる

サーバの負担を軽くする (3)

- ・ historyはstripeされたディスク上とする

```
# sar -d | egrep 'sd47,a|^00:00:00'
```

00:00:00	device	%busy	avque	r+w/s	blks/s	avwait	avserv
	sd47,a	12	10.1	31	520	11.6	314.4
	sd47,a	13	10.9	32	534	22.3	323.4
	sd47,a	13	12.3	34	567	25.2	340.7
	sd47,a	15	16.1	38	636	50.8	375.8
	sd47,a	14	13.9	36	604	31.1	356.2
	sd47,a	15	15.2	37	630	36.8	368.1
	sd47,a	15	14.6	37	618	34.6	361.8
	sd47,a	14	13.3	35	587	31.1	350.2

サーバの負担を軽くする (4)

- ・ `history cache` はある程度大きめにしておくだけ

- ・ `history (dbz)` の検索をしなくてもいいようにする

- ▷ `history cache` は `innodb` が確保したメモリを使用するため大きくしすぎるとメモリ不足におちいるおそれがある

- ▷ メモリに余裕がない場合には `news.dailyreport` にある `history cache` の統計情報を確認しながら落ち着く値になるまで調整する

サーバの負担を軽くする（5）

- ・ `cleanfeed` は使用しない
 - ▷ 使用するとしないでは `fullfeed` の場合フィルタによるCPU使用量が20%程度違う
 - ▷ 配送専用サーバは配送に徹する

サーバの負担を軽くする（6）

- ・ 1 プロセスの `innfeed` のみでの配送は避ける
 - ▷ 100 記事／秒あるいは 4 MB／秒程度ぐらいまでならひとつでも可
 - ▷ 遅いサーバのバックログの処理が増えてしまい、配送に追い付けなくなる
- ・ 配送先毎に個別に `innfeed` を立ち上げるのはシステムに優しくない
 - ▷ 同じ記事の読み込みが個別の `innfeed` で発生する

サーバの負担を軽くする (7)

- ・ 複数の `innfeed` を使用する場合は記事の種類で使い分ける

- ▷ 記事の大きさに使用する `innfeed` を分けることによって同じ記事の読み込みを避ける
- ▷ ニュースグループで分けても記事がクロスポストされていれば意味なし

- ・ バックログが恒常的にたまりがちあるいは頻繁に 4 3 1 を返される `peer` へは `innfeed.conf` で `no-backlog` を `true` とする

- ▷ おそらく後で配送しようとしてもすでに受け取っているであろうから配送不要とみなす

サーバの負担を軽くする (8)

- ・ `innxmit` や `ihave/sendme` はやめる

- ▷ 即時配送しないと記事がカーネルキャッシュに載っている可能性がかなり低く、
`innfeed` に比べて記事の読み込みに関してディスク I/O が激しくなる

- ▷ 複数配送している環境下で `ihave/sendme` はほとんど意味をなさない

- ・ 配送が遅れがちな `peer` への対処

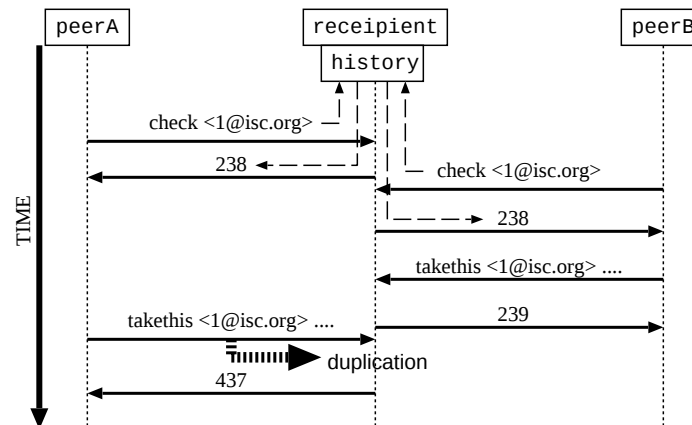
- ▷ `peer` のサーバの性質あるいはネットワーク的な影響により配送が遅れがちな `peer` は
`innfeed` 内で `queue` を慢性的にあふれさせてしまい、バックログが多くなる

- ▷ 配送する記事を制限する

ネットワーク負荷軽減（１）

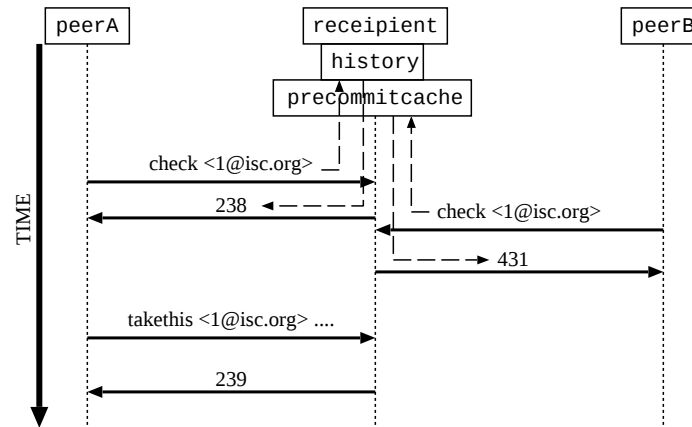
- 記事の重複防止

- 発生条件



ネットワーク負荷軽減（2）

○防止方法



ネットワーク負荷軽減（3）

- ・記事の `reject` の防止

- 発生条件

- ▷ `active` にない

- ▷ 記事が大きすぎる

- ▷ 記事の `filtering`

ネットワーク負荷軽減 (重複によるもの 1)

- ・ 配送側

- 相手が `d i a b l o` な場合

- ▷ まともな `p r e c o m m i t c a c h e` を持つバージョン (2. 0 以降) にしてもらう

- 相手が `I N N` な場合

- ▷ `p r e c o m m i t c a c h e` が組み込まれている `I N N` 2. 2. 3 以降にってもらう

- 相手が `C y c l o n e` な場合

- ▷ 不明

ネットワーク負荷軽減（重複によるもの 2）

- p e e r のバージョンアップが望めない場合
 - ▷ 同時接続数を減らし、s t r e a m i n g をやめる
 - ▷ 他に同じように配送する p e e r がいるであろうから支障はないはず

ネットワーク負荷軽減 (重複によるもの 3)

- ・ 受け取り側

- NNTP の 4 3 1 (`try sending it again later`) を解 釈することが必要

- 相手が `diablo` な場合

- ▷ `diablo 2.0` より前なら 4 3 1 を無視してくれるので `incoming.conf` の `noresendid` を `true` にする

- ▷ 後で送ってもらうよりむしろ送ってもらわないにすることが賢明 (多分他の `peer` から受け取っているから)

ネットワーク負荷軽減 (重複によるもの 4)

○相手がINNな場合

- ▷ `innfeed` なら相当古いバージョンでも 431 を問題なく扱える
- ▷ `innxmit` は 1.4unoff3 以降なら 431 を問題なく扱える
- ▷ `nntp link` は 431 を扱えないので `innfeed` にしてもらう
- ▷ `innfeed` への変更が無理なら古い `diablo` と同じ対応

○相手がCycloneな場合

- ▷ 不明

ネットワーク負荷軽減 (active がない 1)

- 配送側

- active になくとも INN (inn.conf) での want trash が true 相当になっていれば気にすることはない

- ▷ ただし peer に問い合わせなければわからない

- ▷ full feed な peer で want trash が false 相当なところは少ないと思われる

ネットワーク負荷軽減 (activeにない 2)

- ・ 受け取り側

- fullfeedならばactiveになくともwanttrashはtrueにする
- fullfeedでなければpeerとactiveを合わせて不要なニュース グループを配送しないようにしてもらう

ネットワーク負荷軽減 (記事が大きすぎる 1)

- ・ 配送側

- peer が INN の場合

- ▷ peer の `innreport` や `inn_status` を Web 等で確認
 - ▷ `innreport` が確認できれば、`Miscellaneous innd statistics [Top 10:]` の `Huge articles` に自分のサーバが入っていないかどうか確認する
 - ▷ `inn_status` の `Largest Article:` に合わせて配送する記事の大きさを限する
 - ▷ peer に問い合わせる

ネットワーク負荷軽減（記事が大きすぎる 2）

- p e e r が d i a b l o の場合

- ▷ 記事の大きさ制限は現在のところしていない（ようだ）

- p e e r が C y c l o n e の場合

- ▷ ドキュメントによれば o u t g o i n g に対しての制限しかない

- ▷ p e e r に問い合わせるしかない

ネットワーク負荷軽減（記事が大きすぎる 3）

- ・ 受け取り側

- p e e r に連絡して大きさを配送の制限をしてもらう

ネットワーク負荷軽減 (フィルタ 1)

- ・ 配送側

- `peer` が `fullfeed` と銘打ってるのであればフィルタを外してもらう

- ▷ INNなら `ctl innd perl n` あるいは2.4ならば `innd` 起動時に `-N` を付けてもらう

- ▷ `diablo` ならば `-S0` として `spam` フィルタを `disable` してもらう

- ▷ `Cyclone` でフィルタをおこなうとCPU利用率が上って使いものにならないとの報告があるので `fullfeed` で利用してはいないはず

ネットワーク負荷軽減 (フィルタ 2)

○ p e e r でフィルタを外せない (o r 外してくれない) 場合

▷ 配送する記事を制限する (しかない)

▷ 簡単なのは記事の大きさで制限

▷ これが結構効く (フィルタに引っかかる記事の多くは巨大なものであるため)

▷ フィルタに引っかからない大きな記事は他の p e e r から配送してくれるハズ :)

ネットワーク負荷軽減 (フィルタ 3)

- ・ 受け取り側

- `peer` でフィルタしてくれることを願う

- ▷ トランジットなところは来るもの拒まずの心構え

- ▷ 細い回線 (* エコノミーとか FLEET' S) なサイトは `newsfeed@yynet.or.jp` に入っている。なのでローカル側での対処法 (`cleanfeed`) は知ってるハズ

Top 1000とは？（1）

- ・ Net News 記事が伝搬する際の各サーバの重みづけを
サーバで受け取った記事の Path ヘッダを元に集
計ラン キングしたもの

Path: newsfeed.mesh.ad.jp!newshub2.rdc1.sfba.home.com!
news.home.com!feeder.via.net!xfer10.netnews.com!netnews.com!
newspeer1.nac.net!newsfeed.wirehub.nl!telocity-west!TELOCITY!
newsrump.sjc.telocity.net!not-for-mail

- ・ しかしすべてのサーバからの情報が集まっているわけでは
なく、やり方によっては（虚偽のデータを含めて）い
くらでも 上位に食い込むこともできるため正しいデータ
とはなっていない

Top 1000とは？（2）

- ・ 一方でいつのまにかサーバの「凄さ」を示す指標となつてしまった
- ・ 特に海外のNet Newsサーバ管理者間では上位ランキングされていることがpeerする際の条件としていあるいは検討材料となっているようである
- ・ 一部Net Newsサーバ管理者ではランキングすることにより自己満足の世界に浸っている面もある ：)

Top 1000とは？ (3)

- ・ 正式なWebは `http://www.top1000.org/`
- ・ 管理者が多忙のため9月以降メンテされていない
- ・ `alternate` ということで
`http://news.anthologeeek.net/`
が10月頃立ち上がった

(多分) ただしい T o p 1 0 0 0 への潜 入術 (1)

- ・ 流量よりも配送記事数を増やす
 - * . b i n a r i e s . * を除けば一日 1 0 G B 未満の世界
- ・ p e e r を増やす
 - 実は自分がトランジットとなる方が速くなるかもしれない
 - s t u b からの脱却 (トランジットサーバ化)
 - 似たような p e e r だと効果は薄い

(多分) ただしい T o p 1 0 0 0 への潜 入術 (2)

- ・ 記事は分散して受ける
 - 記事の流入が一つの p e e r に集中すると結果的にその p e e r が w e i g h t を占めることになる

(多分) ただしいTop 1000への潜入術 (3)

- ・ `innfeed` で配送する
 - `dynamic-method` は 0 が効果的
 - `min-queue-connection` も同時に `true` に
 - `max-connections` もできれば大きく
 - ▷ ただし `peer` も含めてリソース（ファイルディスクリプタ数）には要注意
 - ▷ 記事数が多くなるようにし、できるだけトラフィックに影響させない

(多分) よくない T o p 1 0 0 0 への潜入術 (1)

- s t u b なサーバは. . .
 - 配送に貢献していないサーバからデータが多く集まると不正確になる
 - これを利用すればいくらでも上位にいける
- ゴミ (含 s p a m) 記事を大量に投稿しない

Top 10000への潜入術 (Tips 編1)

• <http://tf.feed.maxwell.syr.edu/usepaths.html>

○どのサーバをどれぐらいの記事が経由してきているかの指標

・となりの芝は青いかも？

- T o p 1 0 0 0 上位のサイトに多く記事を流し込んでいるところと p e e r するのも一理あり

○実際には思ったほどでないことも

▶ネットワーク的に配送経路がサイト毎に違うので青かったり茶色だったりする

Top 1000への潜入術 (Tips 編2)

- ・ 大器晩成

- まずは peer を増やす

- ▷ 将来その peer が他のサイトと peer することによって上位にのぼる要因となるかも

- ▷ <http://www.usenet-se.net/peering/>

- いつも大器になるわけじゃないのであたりはずれはある

- ・ 洋物

- TTL が 200 ~ 300 ms でも以外といける

Top 1000への潜入術 (Tips 編3)

- ・ 真似ッコ

- サーバの統計情報をWebで見せているところから設定情報やpeerを調べ上げる

- ・ お客様は神様

- 投稿が多いサイトとpeerする

- ▷ spamな投稿が多くてハズレをひかないとも限らない

最後に

- ・メンテを忘れずに
 - `innfeed`や`news.daily`の統計から`peer`の状態の把握
 - 定常トラフィックを感覚的に把握しておくとい

参考

○ <http://www.isc.org/products/INN/>

○ <http://newsfeed.belnet.be/links2.html>

○ <http://www.yynet.org/top100jp/>

○ <http://newsfeed.mesh.ad.jp/>