News Admins are Users Too[1]

Christopher Lueg
Department of Information Systems
University of Technology Sydney
PO Box 123, Broadway NSW 2007, Australia
lueg@it.uts.edu.au

ABSTRACT

In this paper we discuss some of the tasks administrators of Usenet news servers are dealing with. Although significant progress has been made in regard to server technology (both hardware and software) most news server administration tools still are unixoid, text–based tools. Traffic visualizations are commonly used but graphical interfaces as they might be envisioned by HCI people hardly exist. We speculate about the "mental image" news administrators might have and if a high–level language / graphical programming environment like LabView were useful to the realm of Usenet servers.

INTRODUCTION

In these days commercial and academic Usenet servers are continuously connected to the Internet through high–speed links, exchanging news articles with their peers in realtime. Propagating news articles over highspeed NNTP connections is a matter of a few milliseconds. This means making mistakes when changing feed configurations may impact news servers around the world within seconds.

EXAMPLE: A NEWS SERVER OPERATED IN AN ACADEMIC ENVIRONMENT

As an example server we are looking at a Usenet News server which is located at Department of Information Technology at the University of Zurich, Switzerland (the author's previous employer). The server is fairly small compared to commercial news servers but being used in a professional environment administration of the server can be used to illustrate a number of aspects relevant to the topic of this workshop. The author has been administrating the server since early 1995 (since 2000 remotely via ssh).

The news server (the actual software) is a recent version (2.2.1) of the INN (InterNetNews) server software running on a SparcStation5 / Solaris 6 box with 128MB memory and a few gigs of disk space. It's is connected to the Internet through the University's 155MBit highspeed network and exchanges news over realtime NNTP connections with peer servers located in Switzerland and Germany. The server carries some 3000 local, national, and international newsgroups (out of the around 20000 actively used newsgroups that are available world–wide). Incoming news traffic is about 150MB per day, evenly distributed over 24h. In a global list of news servers (Top1000) the server lives in the 2000–2500 area (out of almost 18000 listed servers).

Administrating a news server involves managing a number of interdependent tasks. Among other things (the following list is not intended as a complete description of a news admin's tasks), the admin has to make sure the server

- serves its users, i.e. users can connect to the server using their favorite news reading tools;
- receives news articles from peers;
- stores news for a reasonable amount of time;
- propagates news articles (locally produced and received from peers) to peers;
- creates new newsgroups / removes bogus newsgroups (if appropriate)
- honors cancel requests (if appropriate)

In addition to these news server specific tasks, the admin needs to make sure that the server runs smoothly: enough performance for handling traffic peaks (e.g., "hidden" binary groups doubling the traffic load within hours), enough disk space for news articles; enough memory/swap space for processing peaks; enough resources for usage peaks (INN spawns a new NNRPD process for every client reading news), etc.

In this department the news admin focusses on the news server and its users whereas low–level system maintenance (hardware, operating system) is being done by one of the department's full time system administrator. Thus the situation is different from environments in which news administration is handled by system administrators.

The news server described in this paper was used as a testbed for a number of information filtering projects (Lueg 1998; 2002; 2002; 2003).

INFORMATION RESOURCES

Broadly speaking we can distinguish two types of information sources providing aggregated, summary–type information and real–time information.

Aggregated information

INN offers a number of summaries providing information about a certain period of time, often aggregated to reports such as the "Daily Usenet report". This report delivers information about a number of different aspects relevant to maintaining the server:

1. how the server copes with the data processing;
2. how Usenet evolves;
3. how local and external users use the server.

---

[1] Submitted to the Workshop "System Administrators are Users, Too: Designing Workspaces for Managing Internet–Scale Systems" at ACM SIGCHI, April 2002, Fort Lauderdale, Florida, USA.

Information about the server load includes INN's own server status information, disk usage, log file sizes (for disk space reasons; unusual size may indicate problems), server connections, log entries, control commands passed to INND (INN demon), incoming/ outgoing traffic, history expire (in particular size, time).

Information about how Usenet evolves includes control commands (e.g., newgroup, rmgroup control messages requesting creation/removal of groups) and articles for newsgroups that do not exist on the local server (newsgroups may be unwanted, formally non–existent, not yet created, etc.).

Information about local and external users includes NNRPD readership statistics (reader clients connecting via NNTP), local newsgroup requests, etc.

In its default setting the report does not distinguish between the aforementioned types of information (the output format can be customized). Separation is not straightforward as information, such as control messages processed by the server, may be used as indicators for different aspects: control messages are important with regards to server load and they are also an indicator how Usenet evolves (creation/removal of groups; server attacks).

Real–time information
This kind of information includes information about the system itself (in particular the log files in `/var/adm/messages` and unixoid commands such as *top, ps* for showing load and process details, respectively, and *netstat* for showing open NNTP connections) and a variety of log files about the status of the news server. Log files *news.notice* (peers and users connecting to the server; execution of control commands), *news.err* (things going wrong), and *news.crit* (critical problems) are created through through the system's *syslog* demon. The server creates logs such as *control.log* listing control messages as they arrive at the server; *news* listing processing details of incoming news articles (timestamp, received from, message–ID, how processed locally). At `/var/spool/news/innfeed` the state of the various innfeed queues (typically one per peer) can be inspected.

A nice tool for checking the basic operational status of a news server is a news reader client. Reading news on a regular basis provides information that at least the basics work: successful connects indicate the server "reacts" and spawning new processes works, smoothly going news reading sessions indicate a moderate load, new articles indicate that peering works (or worked) to some extent, etc.

GRAPHICS
There are a number of packages available for visualizing specific aspects of a server's operational status. Typically these packages visualize aggregated information. INFLOW for example can be used to visualize incoming traffic. Normally, the program is started periodically through the *cron* demon (e.g., every full hour). The two pictures show INFLOW having calculated incoming traffic per hierarchy and a visualization of the traffic by the hour. The red line is the total of the pink, blue and green lines representing different feeds.


Netnews hierarchies, sorted by volume – Top 20

calculated at: Mon Jan 20 12:00:05 MET DST 19103

Hierarchy	articles count	%	volume Mbytes	%	kbps	avg_size kB	avg_del hours
soc	3840	19.58	12.542	22.53	2.32	3.266	1.00
comp	4324	22.05	10.363	18.61	1.92	2.397	1.40
de	4221	21.52	8.689	15.61	1.61	2.059	1.15
news	1007	5.14	5.539	9.95	1.03	5.501	0.94
talk	1624	8.28	5.453	9.79	1.01	3.358	0.94
sci	1942	9.90	5.169	9.28	0.96	2.662	2.25
misc	1588	8.10	3.773	6.78	0.70	2.376	1.11
junk	506	2.58	2.628	4.72	0.49	5.194	2.35
rec	99	0.50	0.570	1.02	0.11	5.758	1.86
at	282	1.44	0.506	0.91	0.09	1.796	0.92
humanities	68	0.35	0.256	0.46	0.05	3.760	0.85
gnu	53	0.27	0.112	0.20	0.02	2.119	0.90
ch	48	0.24	0.067	0.12	0.01	1.401	0.86
alt	6	0.03	0.009	0.02	0.00	1.423	0.88
control	1	0.01	0.002	0.00	0.00	2.234	0.97
ethz	1	0.01	0.001	0.00	0.00	1.169	8.22

Created by inflow–stat V2.2.0. inflow is a project of the RIPE netnews-wg.

Articles per hour for different sites
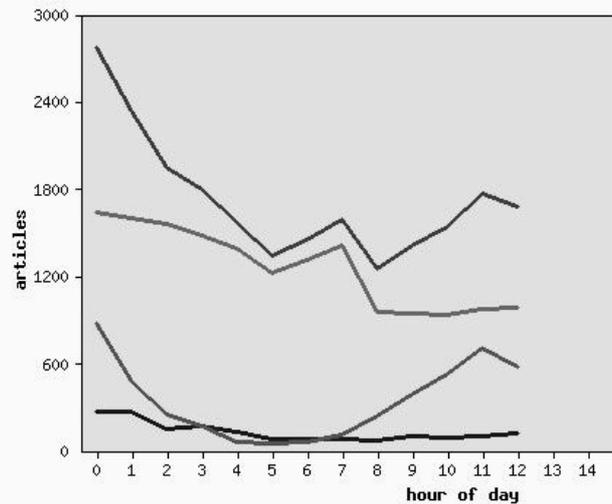
KNOWLEDGE REQUIRED
The author's (and other admins') experiences suggest that maintaining a news server is more than installing and then "using" a software. Often, detecting and fixing problems requires detailed knowledge of the server's internal processing and/or Usenet in general

Internal knowledge
Over the past few years news servers all over the world were attacked by flooding them with control messages. It was vital to understand how the server (and thus the attack) worked in order to resolve the problem: the INN server used to spawn a new process for every newgroup/rmgroup request; Usenet terrorists exploited this weakness by sending thousands of these control messages in order to overload unprotected news servers. The exploit has been resolved by installing a channel–fed control message handler acting like a pipeline for control requests; processing of control messages is serialized rather than processed in parallel.

External knowledge

An example for a different kind of problem that occurred is when the daily traffic doubled within a few weeks to more than 300MB from 150MB. A good guess was that a "hidden binary group" had emerged within one of the hierarchies the server was receiving. "Hidden binary group" means that a group of people started using a new (often text–only) group for trading binaries (pictures, software, MPEGs, etc.), thus causing enormous traffic. Binary traders are invading groups because a) their activities are more difficult to trace if they hide in unrelated newsgroups, and b) non–binary newsgroups are more widely distributed because of the lower traffic.

Some knowledge of Usenet's social evolution is also required when making decisions about creating / removing newsgroups; coping with cancel wars, etc.

MENTAL IMAGES – DATA OR PROCESSES?

When thinking about novel tools in the news administration context an issue to start with would be whether a data–oriented or a process–oriented view of the news administration task were more appropriate. The author is not aware of any literature discussing mental models of news administrators or network operators in general. However, there might be some parallels to operators in supervisory process control (e.g., control of chemical processes):

Brehmer (1987) explains that in supervisory process control operators have to "build up" a mental image of processes based on fairly abstract representations of those processes: displays, charts, etc. because operators cannot really "see" what's going on (e.g., state of ongoing chemical reactions). Rasmussen (1983) maintains that one of the cognitive challenges in supervisory process control is that there are rarely precise definitions of "unwanted" process states; states of processes can only be assesses as "unwanted" in relation to "wanted" process states.

A striking similarity is that in the news admin domain it is also difficult to describe the "wanted" state whereas it is easy to figure out "unwanted" states. A "satisfycing" state does not only depend on the system's "normal" state. The current state has to be considered in relation to Usenet's "state". For example, before control message processing was serialized a server facing a high load –but still performing– was doing fine if the server was subject to a control message–based attack.

A major difference is that the "stuff" admins are dealing with is basically text and the tools they use are typically text–based. A central part of the admin's tasks is managing the flow of articles: setting up new feeds/ peerings; modifying feed configurations; checking feed performance; etc. This means that to some extent the stuff to be manage and the tools that are used are located within the same conceptual space (*this section needs to be re–written*).

WHAT GRAPHICAL USER INTERFACES?

It would be interesting to explore if a high–level language such as LabView could be used for administrating Usenet servers. "Wires" could be used to represent feeds; different feed configurations could be represented by devices. In a large research project involving field work in a paper mill (see Müller and Pfeifer 1997 for a description) we used LabView for modeling data flows and and visualizing aggregated information.

REFERENCES

Brehmer, B. (1987). Development of Mental Models for Decision in Technological Systems. In: Jens J. Rasmussen, K. Duncan, and J. Leplat (Eds.) *New Technology and Human Error.* John Wiley & Sons, pp. 111–120.

LabView www.ni.com/labview/

Lueg, C. (1998). Supporting Situated Actions in High Volume Conversational Data Situations. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Los Angeles, USA, April 1998, pp. 471–479.

Lueg, C. (2000). Supporting Social Navigation in Usenet Newsgroups *Proceedings of the Workshop Social navigation: a design approach?* at the ACM SIGCHI Conference on Human Factors in Computing Systems. April 2000, The Hague, The Netherlands.

Lueg, C. (2002). Enabling Dissemination of Meta Information in the Usenet Framework. *Journal of Digital Information*, Volume 3, Number 1.

Lueg, C. (2003). Exploring Interaction and Participation to Support Information Seeking in a Social Information Space In C. Lueg and D. Fisher (eds.) *From Usenet to CoWebs: Interacting with Social Information Spaces.* Springer, London. ISBN 1–85233–532–7. Pages 232–252.

Müller, M. and Pfeifer, R. (1997). Situated Design in a Large Paper Mill: Developing effective computer systems supporting knowledge intensive work. In: Khosrowpour, M. and Liebowitz, J. (eds.): Cases on Information Technology Management in Modern Organizations. Idea Group Publishing.

Rasmussen. J. (1983). Skills, Rules, and Knowledge: Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models. IEEE Transactions on Systems, Man, and Cybernetics (SMC), 13/3, May/June, pp. 257–266.

Top1000 http://www.top1000.org