Lire: Integrated Analysis of all your Internet and Intranet Services

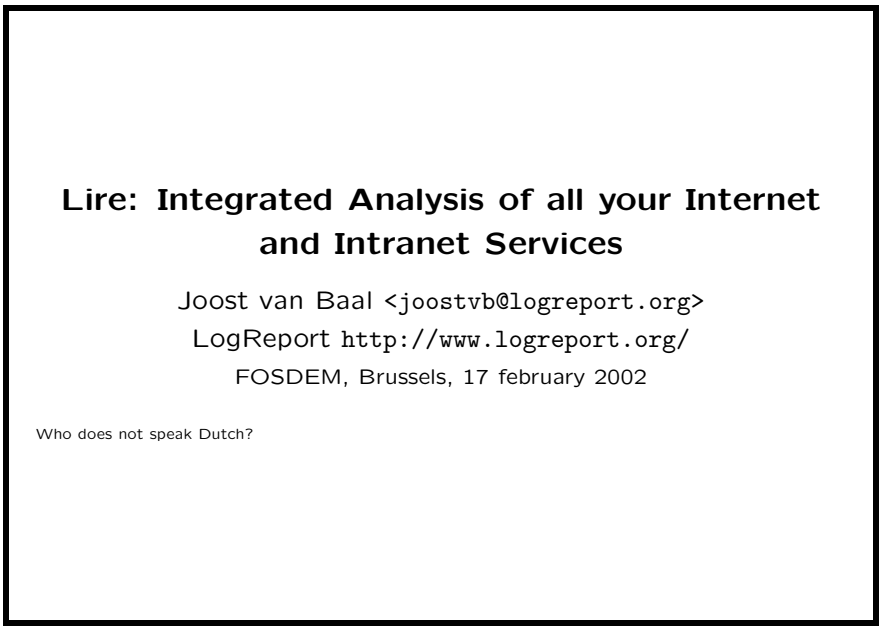
Joost van Baal <joostvb@logreport.org>

17 February 2002

Contents

Introduction

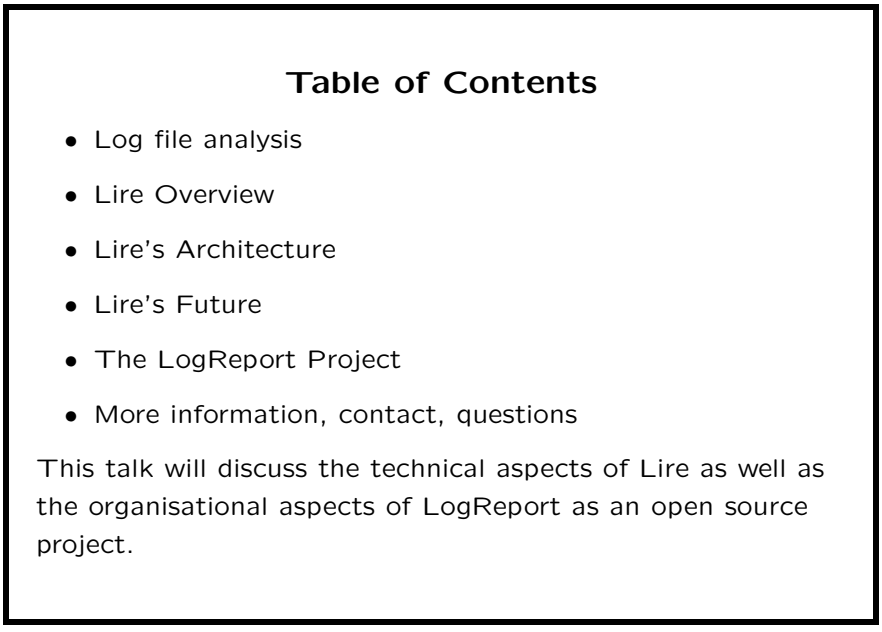
Slide 1



This paper gives an introduction to Lire, LogReport's tool for performing an integrated analysis of all ones Internet and Intranet Services. It accompanies the presentation Joost van Baal gave at FOSDEM, the Free and Open Source Software Developers Meeting which takes place Februari 16 and 17, 2002 at the Université Libre de Bruxelles, Brussels, Belgium.

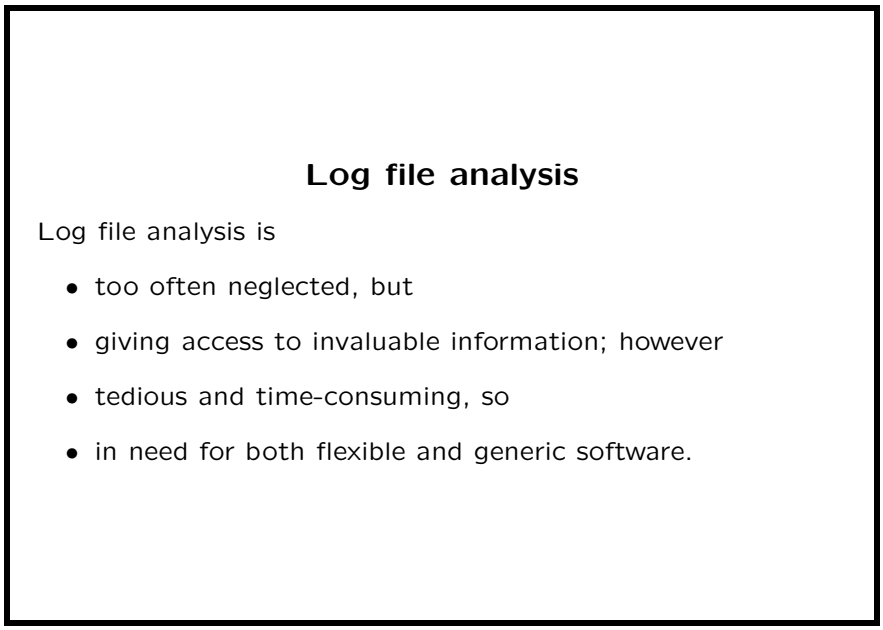
Joost van Baal is employed as a software developer by the Stichting LogReport Foundation; together with four other developers he works on maintaining and promoting Lire, LogReport's flagship product. Next to working for LogReport, Joost van Baal is doing volunteer work for the Debian project. His main interests are programming in Perl and deploying Internet services on Unix platforms.

Slide 2



1 Log file analysis

Slide 3



Log files are often treated like a wasteful by-product of IT activity: they sit somewhere in a dark corner of a computer system and are only examined occasionally, usually in case of after-the-fact reactive problem solving. The infamous *rotate* is the only application dealing with them. This is unfortunate. Log files contain the traces of computer activity, and by intelligently analyzing these traces one can learn a lot about the behavior of a system and its users.

Log file analysis is both an essential and tedious part of system administration. It is essential because it's the best way of profiling the usage of the service installed on the network. It's tedious because programs generate a lot of data and tools to report on this data are unavailable or incomplete. When such tools exist, they are specific to one product, which means that you can't compare your qmail and Exim mail servers.

The Stichting LogReport Foundation detected this flaw in system administration and chose to serve a dual purpose: developing and maintaining Lire, our open source reporting and analysis software, and serving as a nexus of documentation, ideas, and thought on the topic of log files and their potential applications.

2 Lire Overview

2.1 Lire benefits

The LogReport project tries to tackle the problems as outlined above by developing Lire. Lire is a software package to generate useful reports from raw log files of various network programs.

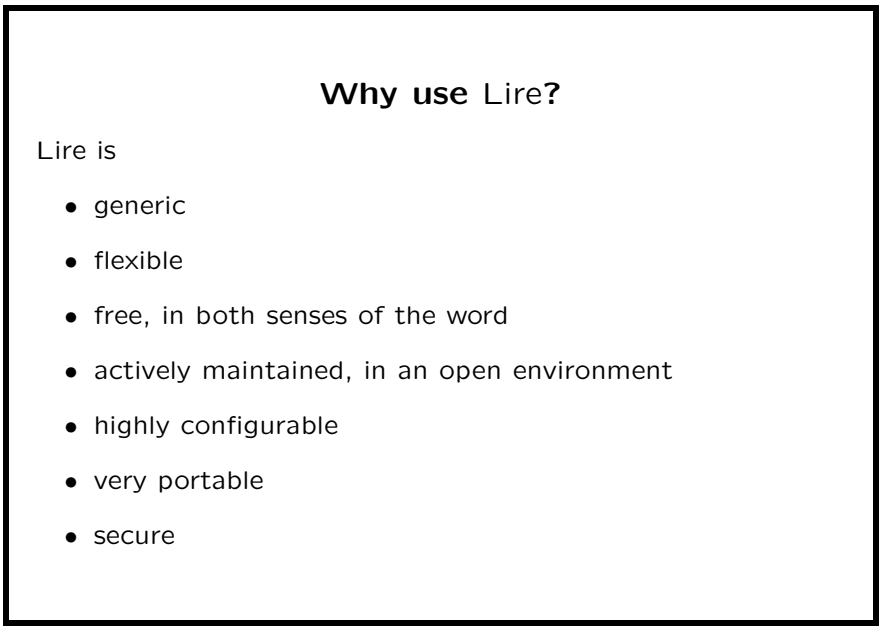
Lire is flexible. The tool can be accessed via a command line interface, but can also be run from cron, and can even get accessed via an email interface.

Lire is Free Software. When using Lire, you'll have all the benefits of Open Source software. Lire is available at no cost, from our website on `http://www.logreport.org/`.

Lire is actively being maintained by the LogReport team, which currently consists of five experienced software developers. The development can be followed live on our CVS on SourceForge. A new release gets shipped almost monthly.

Lire is highly configurable. All configuration files are in a very simple syntax. Of course, a userfriendly interface to write the configuration is shipped with Lire.

Slide 4

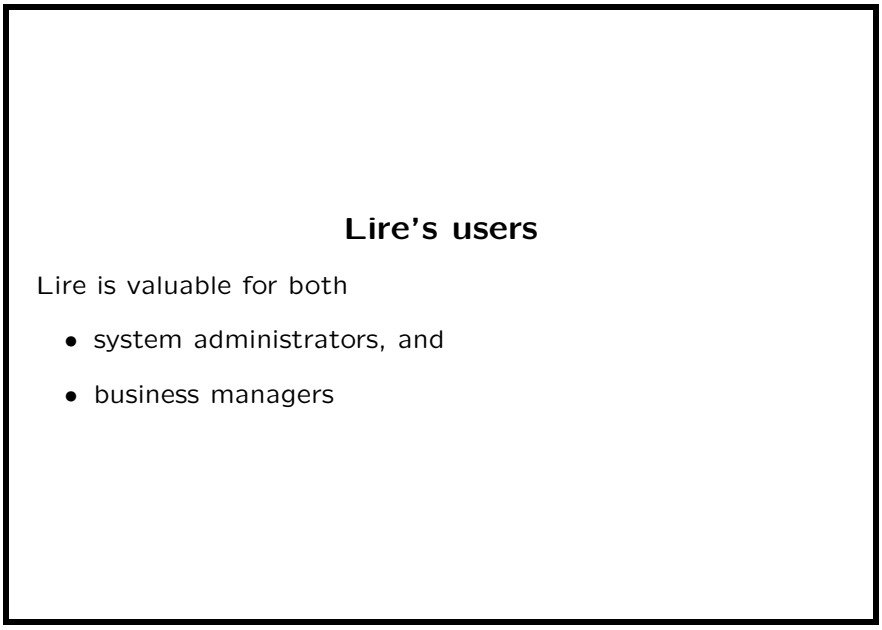


Lire is very portable. It runs on four different Unixen, GNU/Linux included. Since it's written in Perl, porting to different platforms is easy.

Lire is secure. It is run under a dedicated user account. No processes running as root are involved. Care is taken when installing Lire as an online responder. (Of course, this does not exempt the system administrator from defining and implementing her own security measures.)

2.2 Which problems does Lire solve?

Slide 5



It enables one to schedule hardware upgrades, detect anomalities in usage from services. It can be used as a tool in building a traffic-based accounting system for external customers. It gives insight in who's talking to who, which is valuable for marketing and business planners.

2.3 How does Lire do this?

Lire converts stuff like

```
1.example.com - - [03/Feb/2002:06:25:27 +0100] "GET /contact/lists/commit/msg01057.php HTTP/1.0" 200 11193 "-" "Googlebot/2.1
  (+http://www.googlebot.com/bot.html)"
2.example.com - - [03/Feb/2002:06:25:46 +0100] "GET /robots.txt HTTP/1.0" 404 5126 "-" "htdig/3.1.5 (webmaster@logreport.org)"
2.example.com - - [03/Feb/2002:06:25:46 +0100] "GET / HTTP/1.0" 200 12745 "-" "htdig/3.1.5 (webmaster@logreport.org)"
1157.example.com - - [10/Feb/2002:05:23:38 +0100] "GET /css.php HTTP/1.1" 200 3682 "http://logreport.org/doc/gen/dns/" "Mozilla/4.0 (compatible;
  MSIE 6.0; Windows NT 5.0; Q312461)"
1158.example.com - - [10/Feb/2002:06:22:44 +0100] "GET /lire/ex/plain.php HTTP/1.1" 200 14253
  "http://www.google.com/search?hl=en\&q=ascii+text+pics\&spell=1" "Mozilla/4.0 (compatible; MSIE 6.0; Windows 98; Win 9x 4.90)"
```

to a graph like the one below.

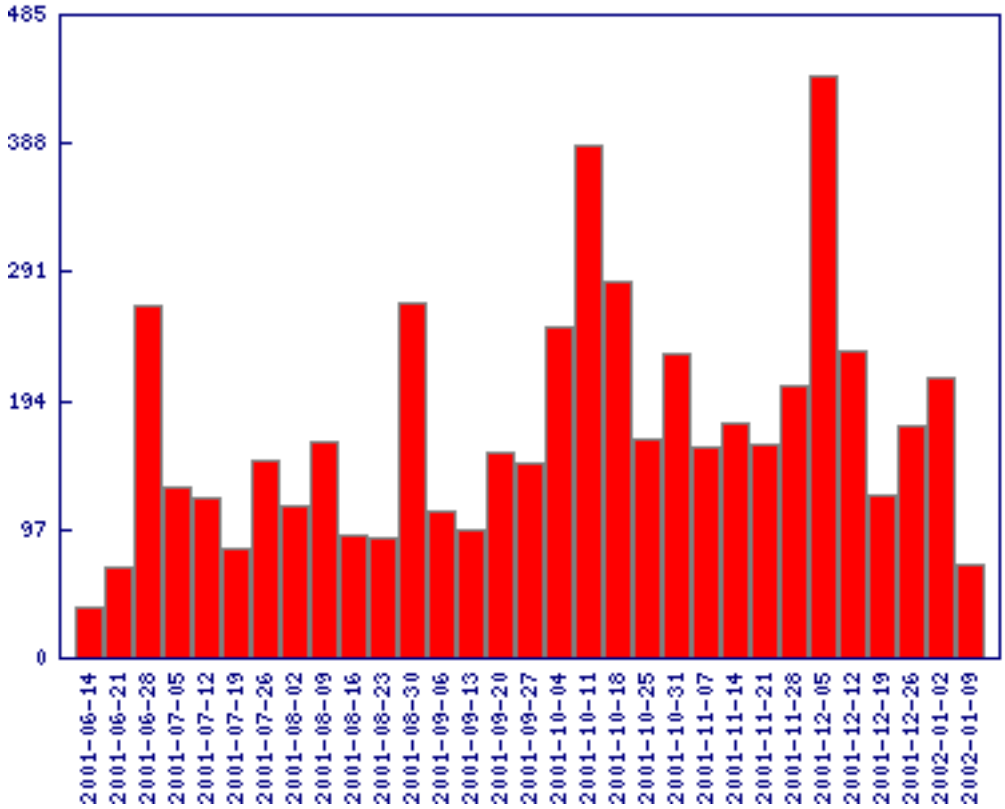


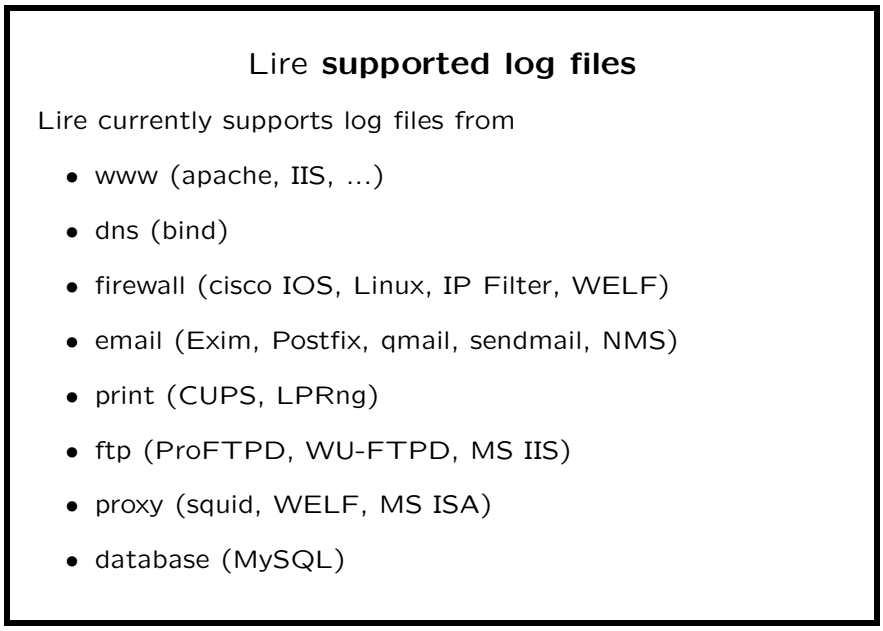
Figure 1: Lire tarball downloads from LogReport webserver, per week

2.4 Lire supported log files and output formats

Multiple programs are supported for various types of network services:

- *www* log files in various formats from various webservers (Apache, IIS, Boa);
- Bind version 8 and version 9 *dns* querylogs;
- logs from *firewalls* such as IOS CISCO router, Linux ipchains, ipfilter and iptables, BSD IP Filter, as well as logs in the WELF format as used by a lot of commercial firewall products;
- *email* logfiles from Exim, Postfix, qmail, sendmail and Netscape Messaging Server;
- *print* logfiles (CUPS and LPRng);
- log files from *ftp* servers in the xferlog format, as used by e.g. ProFTPD and WU-FTPD as well as logs from the MS IIS ftpserver;

Slide 6

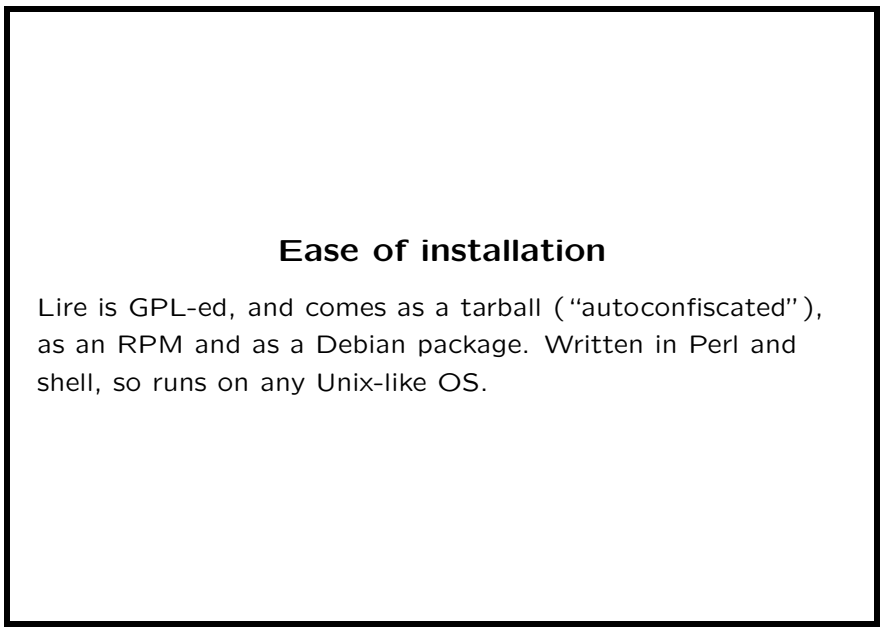


- *proxy* log files from squid and from WELF proxies, as well as from MS ISA servers;
- *database* transaction log files from MySQL servers.

Lire also supports various output formats for the generated reports: HTML, XHTML, XML, PDF and plain ascii. Some of these formats support graphical representation of the data.

2.5 Lire installation

Slide 7

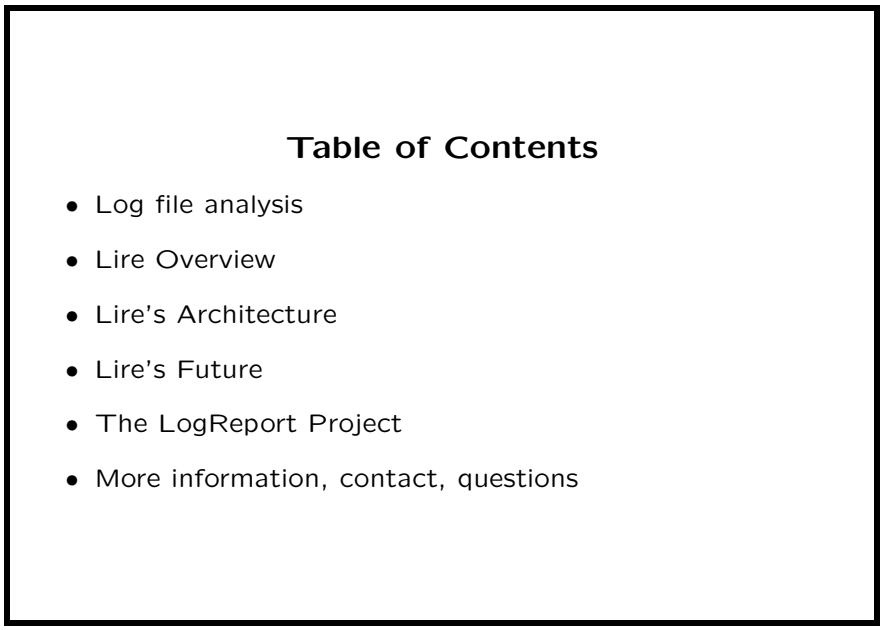


Lire is released under the GNU GPL. A tarball is available for download from the LogReport website at `http://www.logreport.org/pub/`. A binary package for Debian GNU/Linux as well as an RPM is also available.

Lire is written in Perl and shell code. Supported platforms are GNU/Linux, the BSD's and Sun Solaris, but since it's written in Perl, it very likely runs fine on a lot of other platforms too.

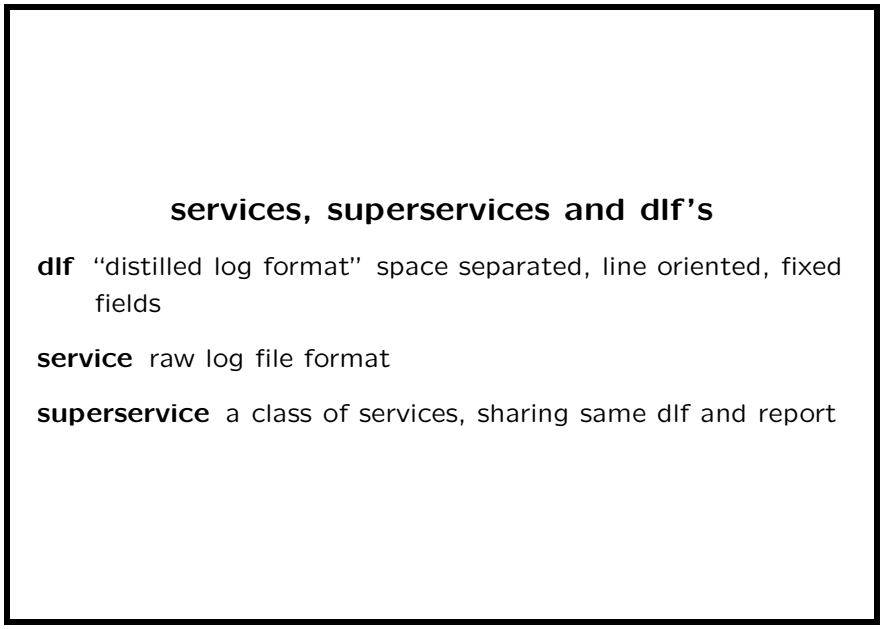
3 Lire's Architecture

Slide 8



3.1 Overview

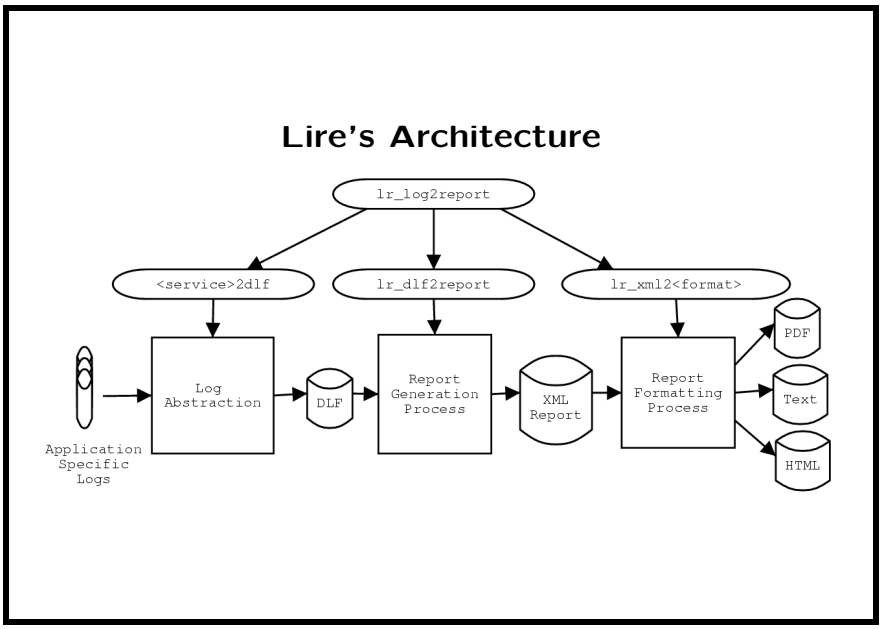
Slide 9



Internally, Lire represents the log file in a DLF file (for Distilled Log Format). This is a simple space-separated line-oriented ascii file. Each logged event is represented by one fixed-fields line.

A service coincides with one well-defined log file format. So, a service generally coincides with one application: the `sendmail` service handles sendmail log files. However, a lot of webservers use W3C defined formats, and a lot of commercial firewalls use the WELF format. Therefore, `w3c_extended` and `welf` are services. Each service has its "2dlf"-convertor. We ship e.g. `sendmail2dlf` and `w3c_extended2dlf`.

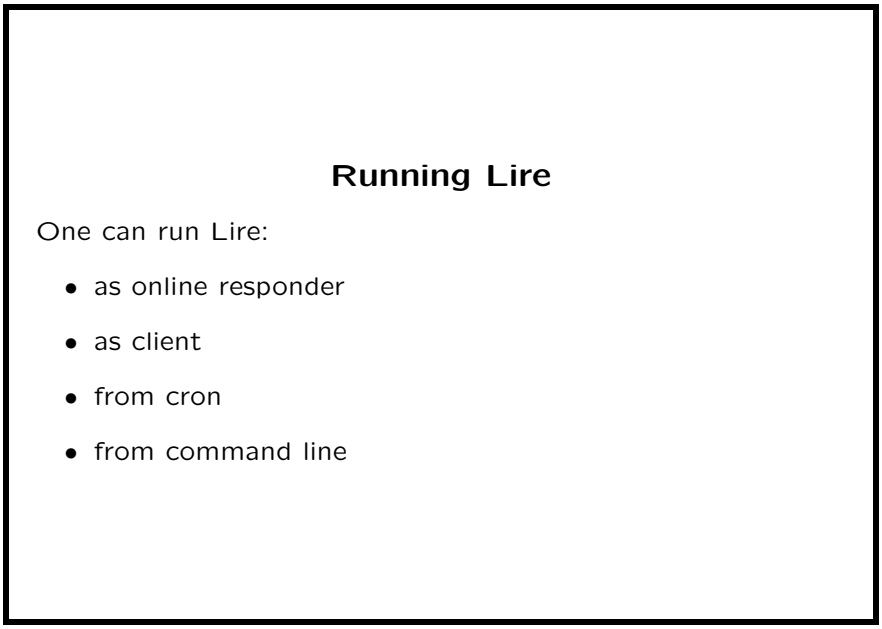
Slide 10



A superservice is a class of applications which share the same DLF format, and which will give the same reports.

3.2 Receiving the log file

Slide 11

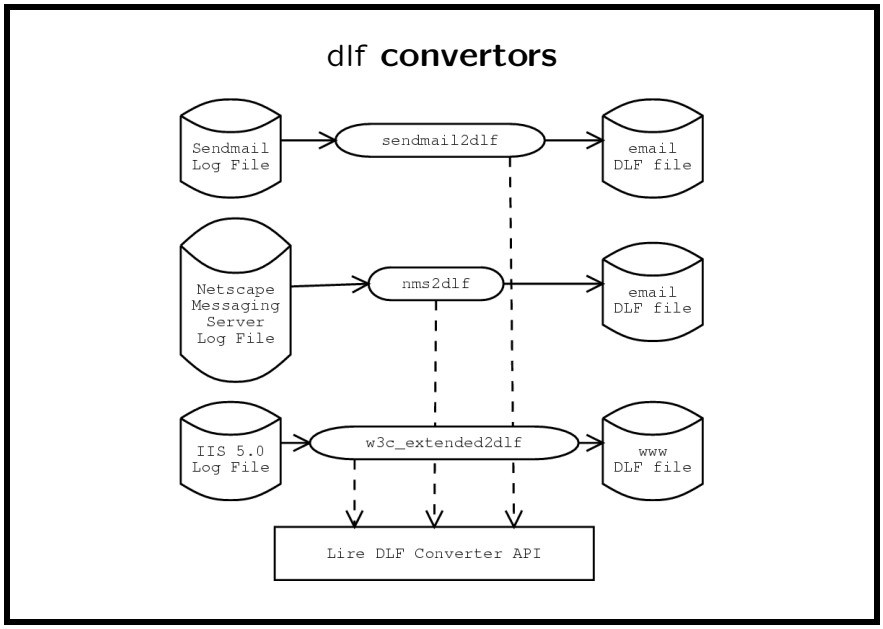


Lire can run in an online responder setup, as a client, as a cron driven system, or as a command line driven system. In an online responder setup, the Lire system receives emails containing log files from other hosts, and sends generated reports back by email. Currently

we're implementing an HTTP file upload interface for feeding log files to a Lire Online Responder system. In a client setup, the system sends log files by email to another Lire system, and receives reports back. Optionally, the logs can be anonimized before being sent. A cron driven setup reads and processes log files after they're rotated, on the local host. A userfriendly script (`lr_config`) is supplied which sets up the cronjob to your taste. In a command line driven system, users run the Lire scripts on an ad-hoc basis. One can use e.g. the `lr_log2report` script.

3.3 Converting to DLF

Slide 12



By invoking the right DLF convertor (e.g. `sendmail2dlf`, `cups_pagelog2dlf` or `squid2dlf`), the log file is converted to the DLF format, suited for the superservice involved. The DLF convertor coincides with a service.

Slide 13

Services and Superservices

A service:

- one raw log file format
- generally one service is one application
- belongs to one superservice
- example: the postfix service

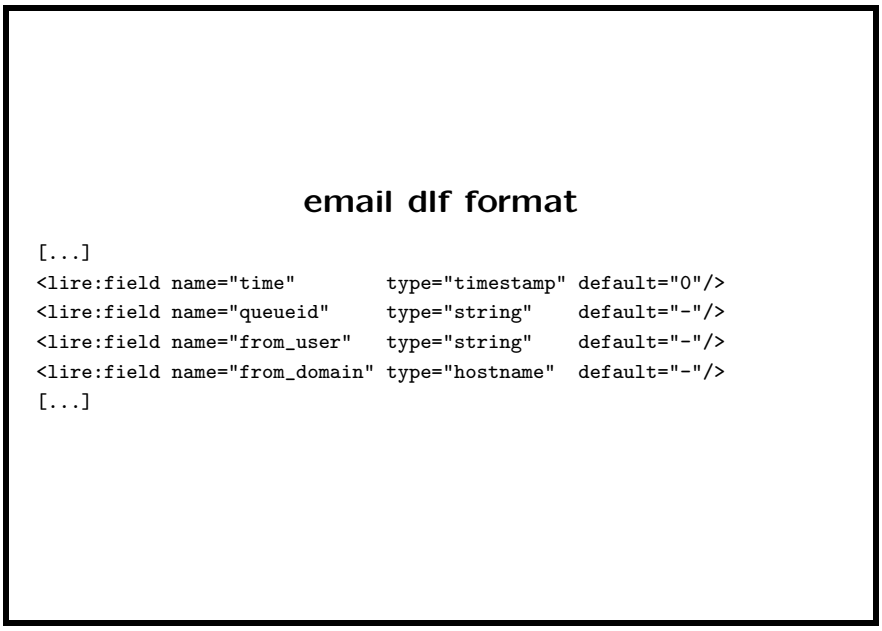
A superservice:

- one dlf format
- a set of supported subreports
- a set of services
- example: the email superservice.

The DLF format for a superservice is defined in a Lire defined XML format. E.g., for the email superservice, this features:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE lire:dlf-schema PUBLIC
  "-//LogReport.ORG//DTD Lire DLF Schema Markup Language V1.0//EN"
  "http://www.logreport.org/LDSML/1.0/ldsml.dtd">
<lire:dlf-schema superservice="email" timestamp="time"
 xmlns:lire="http://www.logreport.org/LDSML/">

<!-- snip -->

 <lire:field name="time"        type="timestamp"    default="0"/>
 <lire:field name="logrelay"    type="string"       default="-"/>
 <lire:field name="queueid"     type="string"       default="-"/>
 <lire:field name="msgid"       type="string"       default="-"/>
 <lire:field name="from_user"   type="string"       default="-"/>
 <lire:field name="from_domain" type="hostname"     default="-"/>
 <lire:field name="from_relay_host" type="hostname" default="-"/>
 <lire:field name="from_relay_ip"   type="ip"       default="-"/>
 <lire:field name="size"        type="bytes"        default="0"/>
 <lire:field name="delay"       type="duration"     default="0"/>

<!-- snip -->

</lire:dlf-schema>
```

Slide 14



Please note that Lire defines its own datatypes. These are used later in the report generating mechanisms.

3.4 Generating an XML Report

A Lire report consists of several subreports, which can be displayed in graphical form, or as a table. A lot of subreports (142, as of february 2002) come with Lire predefined, but of

course one can define ones own reports. A report definition is written in the Lire Report Specification Markup Language; it looks like e.g.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE lire:report-spec PUBLIC
  "-//LogReport.ORG//DTD Lire Report Specification Markup Language V1.0//EN"
  "http://www.logreport.org/LRSML/1.0/lrsml.dtd">
<lire:report-spec xmlns:lire="http://www.logreport.org/LRSML/"
 superservice="email" id="top-volume-to-domain" charttype="bars">

 <lire:title>Largest Volume Sent To Domain Email Report</lire:title>
 <lire:description>
  <para>This report lists the domains to which the
   largest volume of mail was sent.</para>
 </lire:description>

 <lire:param-spec>
  <lire:param name="domain_to_show" type="int" default="10">
  <!--- snip -->
  </lire:param>
 </lire:param-spec>

 <lire:display-spec>
  <lire:title>Largest Volume Sent To Domain, Top $domain_to_show</lire:title>
  <lire:description>
   <para>Volume is in bytes</para>
  </lire:description>
 </lire:display-spec>

 <lire:filter-spec>
  <lire:eq arg1="$stat" arg2="sent"/>
 </lire:filter-spec>

 <lire:report-calc-spec>
  <lire:group sort="-mail_volume" limit="$domain_to_show">
   <lire:field name="to_domain"/>
   <lire:sum name="mail_volume" field="size"/>
  </lire:group>
 </lire:report-calc-spec>

</lire:report-spec>
```

We reuse the `stat`, `to_domain`, and `size` fields from the email DLF specification. The `mail_volume` field is internal to this report; it's used only within this report calculation. Operators like `lire:group`, `lire:timegroup`, `lire:rangegroup`, `lire:timeslot`, `lire:field`, `lire:sum`, `lire:avg`, `lire:min`, `lire:max`, `lire:count` can be used in the `report-calc-spec`.

The `domain_to_show` variable is offered as a hook for user configuration. Users can set this variable in a report configuration file.

Per superservice, there is one configuration file, for all subreports. Such a file looks like e.g.

```
# Report configuration for the email superservice

deliveries-by-period                    period=1d
volume-by-period                        period=1d
top-volume-to-domain                    domain_to_show=10
top-to-email-by-domain                  domain_to_show=30 user_to_show=5
```

Slide 15

Report Specification

```
[...]
<lire:report-calc-spec>
 <lire:group sort="-mail_volume" limit="$domain_to_show">
  <lire:field name="to_domain"/>
  <lire:sum name="mail_volume" field="size"/>
 </lire:group>
</lire:report-calc-spec>
[...]
```

```
deliveries-by-size                  size=1k
deliveries-by-delay                 delay-size=1s
tracked-recipients                  tracked_email_re="root@example\.com"
[...]
```

The configuration file defines which subreports we want to show up in our report, and defines the settings of the variables for these subreports. For the proxy superservice, the report configuration file looks like

Slide 16

Report configuration file

```
# Report configuration for the proxy superservice

=section General
requests-summary

=section Denied Sites Reports
|select-cache_result result=TCP_DENIED
top-destinations           dsts_to_show=50
top-users-by-destinations  users_to_show=30 dsts_to_show=50
[...]
```

Note the | line: this defines a filter, shared among the reports below.

The `lr_config` script, which comes with Lire, gives a userfriendly interface to set these variables.

3.5 Typesetting and publishing the Report

All graph generation is done by the GD::Graph perl module from CPAN. We show some more examples of graphs, from the `www` and `dns` supervices. Similar graphs are generated out-of-the box for the six other superservices.

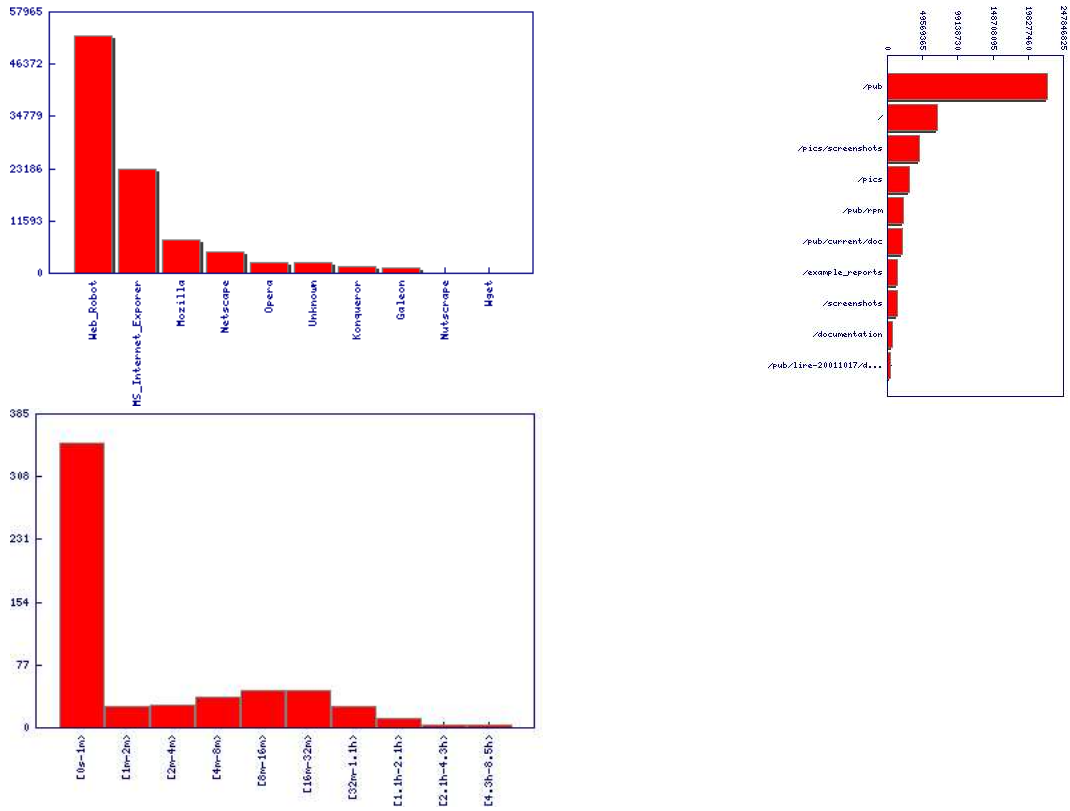


Figure 2: Reports from the www superservice: requests by browser, size by directory, user session visit times

3.6 Implementation

For small and medium-size logs Lire is fast. Although some heavyweight external programs are used to process XML files (jade, for typesetting pdf and rtf), performance is above expectations. When very high speed is needed, plain ascii reports can get produced. Due to its modular design, it's fairly easy to reimplement performance bottlenecks in e.g. C, to fulfill extreme performance demands. We are currently working on improving performance when handling big log files. However, we'd need access to industry-size environments for this.

We do have a knob (`LR_MAX_MEMORY`) to tweak the amount of memory one is willing to dedicate to Lire. This enables one to exchange disk i/o for memoryaccess.

4 Lire's future

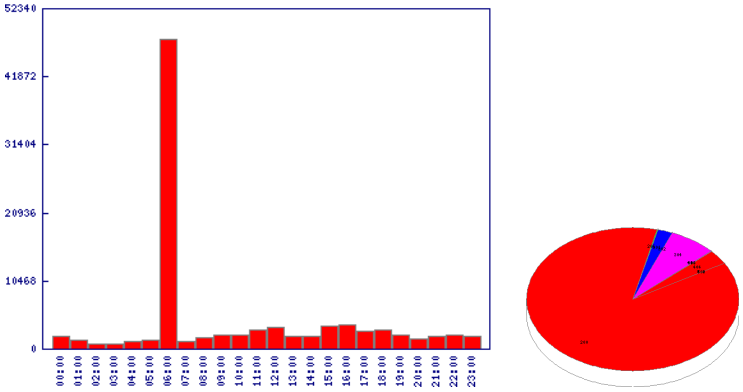
4.1 lire-20020214.tar.gz

On thursday, february 14 2002, a new Lire package was released. Latest improvements are: lots of new superservices, services and reports (NMS, WELF, IIS FTP, MySQL ...), various improvements in the reporting stage, a lot of developers' documentation got added.

Slide 17

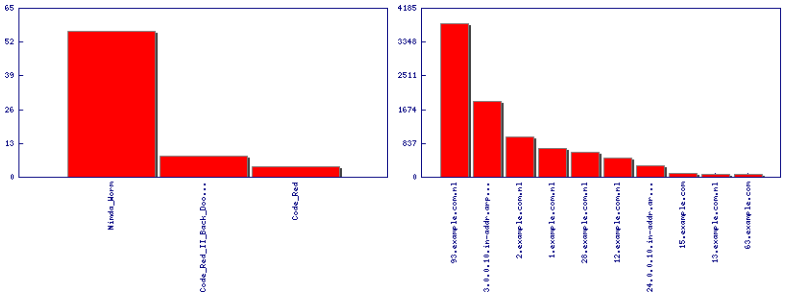
Example Graphical Reports

Some reports from the www superservice



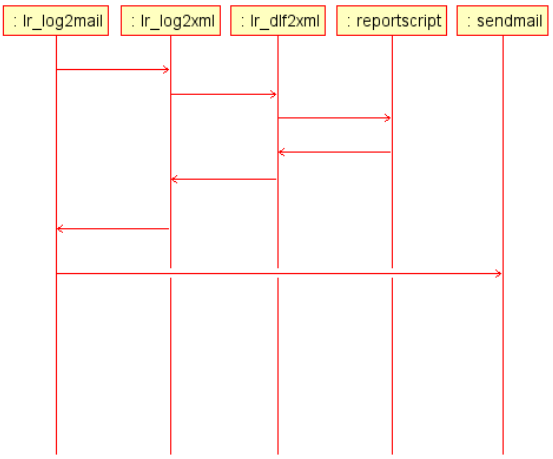
Slide 18

Some reports from the www and dns superservices



Slide 19

Lire's Scripts



Slide 20

Table of Contents

- Log file analysis
- Lire Overview
- Lire's Architecture
- Lire's Future
- The LogReport Project
- More information, contact, questions

Slide 21

Release, Roadmap

lire-20020214.tar.gz is realised!

But we have more plans:

- merging and splitting of reports and log files
- display
- performance: jade
- more services
- online responder
- configuration interface

4.2 Roadmap

We have an ambitious roadmap.

merging and splitting of reports and log files We would like Lire to be able to merge and split reports and log files. Currently, one log file gets converted to one report. It should be possible to combine different reports after the fact, e.g. for generating reports about longer timeframes, or to combine reports from different servers. It should be easy to generate handcrafted reports after the fact from already processed log files.

display User configuration hooks should be added to tweak various display settings, the overall sexyness of the output should be improved.

jade All jade dependencies should be removed, including the depency on jade to produce PDF output.

even more services We plan to create a new messagestore superservice, for POP and IMAP servers. Furthermore, LDAP log file support, and support for extra email services (iPlanet and Netscape Messaging Server) is planned.

online responder The online responder we offer on our website should be able to produce HTML output. Addresses like `<log-html@combined.logreport.org>` should get advertised. The HTTP upload interface should be completed. Email attachment handling should get improved. Installation of a responder from tarball should be better documented and should be easier.

configuration interface There should be a better configuration interface than the `lr_config` script we offer now. The CGI interface should get completed. A GUI interface should get added.

5 The LogReport Project

Slide 22

LogReport people

LogReport developers

- Joost van Baal
- Wessel Dankers
- Josh Koenig
- Francis Lacoste
- Egon Willighagen

LogReport board

- Teus Hagen (chairman)
- Wytze van der Raay (treasurer)
- Jakob Schripsema (secretary)

5.1 People working for LogReport

Five people are working for LogReport:

- Joost van Baal
- Wessel Dankers
- Josh Koenig
- Francis Lacoste
- Egon Willighagen

These people live and work from The Netherlands, Canada and the USA, all part time. Communication is done using IRC (`#logreport` on OPN) and email.

Next to a website, the project offers two mailinglists: `<questions@logreport.org>` and `<development@logreport.org>`. Both Lire and our website are maintained via CVS, hosted on SourceForge. We have our own server which hosts our website as well as the lists.

Furthermore this server hosts the LogReport Online Responder. One can sent (compressed) logfiles in email messages to dedicated addresses, like e.g. `log@qmail.logreport.org`, `log@bind9.logreport.org`, and get a report back as a response. Optionally, one can anonimize the log before submitting it, using a simple script which comes with Lire.

5.2 The LogReport Foundation

Stichting LogReport Foundation is a non-profit organization; it got a legal status as a foundation, and funding by the NLnet Foundation (`http://www.nlnet.nl/`), in August 2000.

- Teus Hagen (chairman)
- Wytze van der Raay (treasurer)
- Jakob Schripsema (secretary)

Slide 23

How to help

- Use our Online Responder
- Sent (anonimized) log files
- Download Lire, and use it
- Give feedback on our mailinglists: feature requests, bug reports, help other people
- Even better: send patches and add support for other services
- Promote Lire: via webpages and mailinglists
- Fund us.

5.3 How to help

We need log files to test our code, and to be able to add support for more services. We especially lack log files from expensive commercial services, like the WELF and Microsoft ones. (We don't run these ourselves...)

If you use the Debian package, you can use the Debian Bug Tracking System to report bugs.

If you intent to write code, be sure to use our current CVS, of course. Our CVS, hosted on SourceForge, is readable for anyone. Code contributions of non-trivial size are accepted if - of course - they meet our quality standards, and they're offered under a GPL compatible license, like the GPL itself, or e.g. the modified BSD license. People who are willing to contribute to the Lire project during a longer time, can get write access to our CVS tree. Contact us for more information.

Promote Lire: link to us from your webpage, suggest using Lire on mailinglists. Join the Lire community: help other users on our lists.

Fund us: Funding from Stichting NLnet which currently enables us to spend a lot of time on Lire will run out in the near future. Financial contributions to the LogReport foundation are tax-deductable under Dutch law, because LogReport is recognized as a charitable goal. However, other ways to support Lire's continued development are possible. Contact us if you're interested.

6 More information, contact info

6.1 More information

More information is on our website on `http://www.logreport.org/`. We have several mailing lists, which are archived. Egon Willighagen wrote a series of articles for LinuxFocus (`http://www.linuxfocus.org/`), called "Analyzing your internet applications' log files", available in eight languages. A lot of documentation and manpages come with Lire.

We sent newsitems via our `announcement@logreport.org` list. Subscribe to it if you wanna be kept informed.

Slide 24

6.2 Contact

Contact us via our lists `questions@logreport.org` and `development@logreport.org` (see our website for subscribtion info), or privately on `logreport@logreport.org`. To have an informal chat with the LogReport developers, join the `#logreport` IRC channel on the Open Projects Network.