Link Layer Interface Specification

Link Layer Interface Specification

Version 1.1 Edition 7.20141001
Updated October 25, 2014
Distributed with Package openss7-1.1.7.20141001

Abstract:

This document is a Specification containing technical details concerning the implementation of the Link Layer Interface for OpenSS7. It contains recommendations on software architecture as well as platform and system applicability of the Link Layer Interface. It provides abstraction of the LAPB Protocol (ISO/IEC 7776) and LLC2 Protocol (ISO/IEC 8802) service interface to these components as well as providing a basis for link layer control for other protocols.

Brian Bidulock <bidulock@openss7.org> **for**
The OpenSS7 Project <http://www.openss7.org/>

Published by:

OpenSS7 Corporation
1469 Jefferys Crescent
Edmonton, Alberta T6L 6T1
Canada

Notice:

Short Contents

Table of Contents

List of Figures

List of Tables

Preface

Notice

Software in this document and related software is released under the AGPL (see [GNU Affero General Public License], page 57). Please note, however, that there are different licensing terms for some of the manual package and some of the documentation. Consult permission notices contained in the documentation of those components for more information.

This document is released under the FDL (see [GNU Free Documentation License], page 67) with no invariant sections, no front-cover texts and no back-cover texts.

Abstract

This document is a Specification containing technical details concerning the implementation of the Link Layer Interface for OpenSS7. It contains recommendations on software architecture as well as platform and system applicability of the Link Layer Interface.

This document specifies a Link Layer Interface Specification in support of the OpenSS7 Link Layer Control Protocol (LLC) protocol stacks. It provides abstraction of the Link Layer Control interface to these components as well as providing a basis for Link Layer Control control for other Link Layer Control protocols.

Purpose

The purpose of this document is to provide technical documentation of the Link Layer Interface. This document is intended to be included with the OpenSS7 STREAMS software package released by *OpenSS7 Corporation*. It is intended to assist software developers, maintainers and users of the Link Layer Interface with understanding the software architecture and technical interfaces that are made available in the software package.

Intent

It is the intent of this document that it act as the primary source of information concerning the Link Layer Interface. This document is intended to provide information for writers of OpenSS7 Link Layer Interface applications as well as writers of OpenSS7 Link Layer Interface Users.

Audience

The audience for this document is software developers, maintainers and users and integrators of the Link Layer Interface. The target audience is developers and users of the OpenSS7 SS7 stack.

Revision History

Take care that you are working with a current version of this documentation: you will not be notified of updates. To ensure that you are working with a current version, check the OpenSS7 Project website for a current version.

A current version of this specification is normally distributed with the *OpenSS7* package, `openss7-1.1.7.20141001`.[1]

[1] `http://www.openss7.org/repos/tarballs/openss7-1.1.7.20141001.tar.bz2`

Version Control

Although the author has attempted to ensure that the information in this document is complete and correct, neither the Author nor OpenSS7 Corporation will take any responsibility in it. *OpenSS7 Corporation* is making this documentation available as a reference point for the industry. While *OpenSS7 Corporation* believes that these interfaces are well defined in this release of the document, minor changes may be made prior to products conforming to the interfaces being made available. *OpenSS7 Corporation* reserves the right to revise this software and documentation for any reason, including but not limited to, conformity with standards promulgated by various agencies, utilization of advances in the state of the technical arts, or the reflection of changes in the design of any techniques, or procedures embodied, described, or referred to herein. *OpenSS7 Corporation* is under no obligation to provide any feature listed herein.

```
$Log: lli.texi,v $
Revision 1.1.2.2  2011-02-07 02:21:39  brian
- updated manuals

Revision 1.1.2.1  2009-06-21 10:53:45  brian
- added files to new distro
```

ISO 9000 Compliance

Only the TEX, texinfo, or roff source for this maual is controlled. An opaque (printed, postscript or portable document format) version of this manual is a **UNCONTROLLED VERSION**.

Disclaimer

OpenSS7 Corporation **disclaims all warranties with regard to this documentation including all implied warranties of merchantability, fitness for a particular purpose, non-infrincement, or title; that the contents of the manual are suitable for any purpose, or that the implementation of such contents will not infringe on any third party patents, copyrights, trademarks or other rights. In no event shall** *OpenSS7 Corporation* **be liable for any direct, indirect, special or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action or contract, negligence or other tortious action, arising out of or in connection with any use of this documentation or the performance or implementation of the contents thereof.**

U.S. Government Restricted Rights

If you are licensing this Software on behalf of the U.S. Government ("Government"), the following provisions apply to you. If the Software is supplied by the Department of Defense ("DoD"), it is classified as "Commercial Computer Software" under paragraph 252.227-7014 of the DoD Supplement to the Federal Aquisition Regulations ("DFARS") (or any successor regulations) and the Government is acquiring only the license rights granded herein (the license rights customarily provided to non-Government users). If the Software is supplied to any unit or agency of the Government other than DoD, it is classified as "Restricted Computer Software" and the Government's rights in the Software are defined in paragraph 52.227-19 of the Federal Acquisition Regulations ("FAR") (or any successor regulations) or, in the cases of NASA, in paragraph 18.52.227-86 of the NASA Supplerment to the FAR (or any successor regulations).

Acknowledgements

The OpenSS7 Project was funded in part by:

- Monavacon Limited
- OpenSS7 Corporation

Thanks to the subscribers to and sponsors of The OpenSS7 Project. Without their support, open software like this would not be possible.

As with most open source projects, this project would not have been possible without the valiant efforts and productive software of the Free Software Foundation, the Linux Kernel Community, and the open source software movement at large.

1 Introduction

UNIX System V Release 3.2 included a Logical Link Interface (LLI) that was later standardized as the Data Link Provider Interface (DLPI) by the *Open Group*.[1]

The Link Layer Interface (LLI) was developed by Spider Systems, Ltd., (now a division of Emerson Power) and is widely available on many platforms. For example, *AIX AIXlink/X.25*, *HP-UX X.25/9000*, *IRIX IRIS SX.25*, *PT X.25*, *RadiSys WAN*, *SBE X.25*, *Solaris SunLink X.25* and *Solaris Solstice X.25*, implement the Link Layer Interface (LLI).[2]

The Link Layer Interface (LLI) as designed to be used directly with standard STREAMS system calls and does not require the use of a cooperating user space shared library. Applications programs directly use the `getmsg(2s)`, `getpmsg(2s)`, `putmsg(2s)`, `putpmsg(2s)` and `ioctl(2s)` system calls.[3] Nevertheless, user shared object libraries can easily be constructed using this STREAMS service primitive interface.

The system header files that must be included when compiling user applications, or STREAMS drivers and modules that use the interface, are detailed in Chapter 6 [LLI Header Files], page 31.

A user library, `libsx25`, is provided, not for interfacing to the message primitive service interface, but for providing various helper functions when using the STREAMS service interface. This library is detailed in Chapter 7 [LLI Library], page 39.

[1] Evidence of this can still be found in the `dlpi.h` header file distributed by *The Open Group* in the DLPI standard. Reference is made in the header file to backward compatibility to the LLI interface.

[2] But, for the most part, it is not called the *Link Lyaer Interface*, but is instead sunk inside the Network Layer Interface (NLI).

[3] See `getmsg(2s)`, `getpmsg(2s)`, `putmsg(2s)`, `putpmsg(2s)` and `ioctl(2s)` manual pages.

2 Model of the Link Layer

3 LLI Services

3.1 LLI Commands

3.2 LLI Data Structures

4 LLI Message Primitives

Although it perhaps did not at first, the LLI uses the service primitives and message format of the *Data Link Provider Interface*, [DLPI], page 55.

5 LLI Input-Output Controls

5.1 Input-Output Control Data Structures

The `/usr/include/openss7/sys/snet/ll_control.h` header file (`<sys/snet/ll_control.h>` with proper compile flags) defines a number of structures, pointers to which are used as argumnets to input-output controls. These structures fall into four classes, identified by the value of the first byte of the structure, as follows:

`LI_PLAIN` A `ll_hdioc` structure that identifies the subnetwork (link)

`LI_SNID` A `ll_snioc` structure that identifies the subnetwork (link)

`LI_STATUS` A `ll_stnioc` structure that identifies the subnetwork (link)

`LI_STATS` A `ll_stioc` or `mlp_stioc` structure that identifies the subnetwork (link)

`LI_GSTATS` A `ll_gstioc` or `mlp_gstioc` structure that identifies the subnetwork (link)

`LI_LAPBTUNE` A `lapb_tnioc` structure that identifies the subnetwork (link)

`LI_LLC2TUNE` A `llc2_tnioc` structure that identifies the subnetwork (link)

`LI_MLPTUNE` A `mlp_tnioc` structure that identifies the subnetwork (link)

5.1.1 LI_PLAIN - Plain Data Structures

5.1.1.1 ll_hdioc Structure

The `ll_hdioc` structure is formatted as follows:

```
struct ll_hdioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lli_snid;
};
```

The `ll_hdioc` structure contains the following members:

lli_type Always `LLI_PLAIN`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

5.1.2 LI_SNID - Subnetwork Identifier Data Strucures

5.1.2.1 ll_snioc Structure

The `ll_snioc` structure is used in the `L_SETSNID`, `L_GETSNID`, `M_SETSNID` and `M_GETSNID` input-output controls.

The `ll_snioc` structure is formatted as follows:

```
struct ll_snioc {
    uint8_t lli_type;
    uint8_t lli_class;
    uint8_t lli_spare[2];
    uint32_t lli_snid;
    uint32_t lli_index;
    uint32_t lli_slp_snid;
    uint16_t lli_slp_pri;
};
```

The `ll_snioc` structure contains the following members:

lli_type Always `LLI_SNID`.

lli_class Specifies the class of the link being registered. This field can be one of the following values:

`LC_LLC1`	LLC Type 1 link.
`LC_LLC2`	LLC TYpe 2 link.
`LC_LAPBDTE`	LAPB DTE.
`LC_LAPBXDTE`	LAPB DTE with extended addressing.
`LC_LAPBDCE`	LAPB DCE.
`LC_LAPBXDCE`	LAPB DCE with extended addressing.
`LC_LAPDTE`	LAP DTE.
`LC_LAPDCE`	LAP DCE.
`LC_HDLC`	HDLC.
`LC_HDLCX`	HDLC with extended addressing.
`LC_MLAPBDTE`	LAPB DTE multilink.
`LC_MLAPBXDTE`	LAPB DTE multilink with extended addressing.
`LC_MLAPBDCE`	LAPB DCE multilink.
`LC_MLAPBXDCE`	LAPB DCE multilink with extended addressing.
`LC_MLP`	MLP link.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

lli_index Specifies the lower multiplex identifier of the Stream linked under a multiplexing driver as returned from `I_LINK(7)`.[1]

lli_slp_snid Specifies the subnetwork (link) identifier for an SLP link within an MLP link.

lli_slp_pri Specifies the subnetwork (link) priority within an MLP link.

5.1.3 LI_STATUS - Status Data Strucures

5.1.3.1 ll_stnioc Structure

The `ll_stnioc` structure is formatted as follows:

```
struct ll_stnioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lli_snid;
};
```

[1] See the `I_LINK(7)` manual page.

The `ll_stnioc` structure contains the following members:

lli_type Always `LLI_STATUS`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

5.1.4 LI_STATS - Statistics Data Structures

5.1.4.1 lapb_stioc Structure

The `lapb_stioc` structure is formatted as follows:

```
struct lapb_stioc {
    uint8_t lli_type;
    uint8_t state;
    uint16_t lli_spare;
    uint32_t lli_snid;
    lapbstats_t lli_stats;
};
```

The `lapb_stioc` structure contains the following members:

lli_type Always `LLI_STATS`.

state

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

lli_stats Contains the `lapbstats_t` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.1.4.2 lapbstats_t Structure

The `lapbstats_t` structure is formatted as follows:

```
typedef struct lapb_stats {
    uint32_t lapbmonarray[49];
} lapbstats_t;
```

The `lapbstats_t` structure contains the following members:

lapbmonarray

Provides an array of 32-bit unsigned integers containing statistics. The indexes of the elements of the array are as follows:

0	`tx_ign`	-
1	`rx_badlen`	-
2	`rx_unknown`	-
3	`t1_exp`	-
4	`t4_exp`	-
5	`t4_n2_exp`	-
6	`RR_rx_cmd`	-
7	`RR_rx_rsp`	-
8	`RR_tx_cmd`	-
9	`RR_tx_rsp`	-

10	RR_tx_cmd_p	-
11	RNR_rx_cmd	-
12	RNR_rx_rsp	-
13	RNR_tx_cmd	-
14	RNR_tx_rsp	-
15	RNR_tx_cmd_p	-
16	REJ_rx_cmd	-
17	REJ_rx_rsp	-
18	REJ_tx_cmd	-
19	REJ_tx_rsp	-
20	REJ_tx_cmd_p	-
21	SABME_rx_cmd	-
22	SABME_tx_cmd	-
23	DISC_rx_cmd	-
24	DISC_tx_cmd	-
25	UA_rx_rsp	-
26	UA_tx_rsp	-
27	DM_rx_rsp	-
28	DM_tx_rsp	-
29	I_rx_cmd	-
30	I_tx_cmd	-
31	FRMR_rx_rsp	-
32	FRMR_tx_rsp	-
33	tx_rtr	-
34	rx_bad	-
35	rx_dud	-
36	rx_ign	-
37	XID_rx_cmd	-
38	XID_rx_rsp	-
39	XID_tx_cmd	-
40	XID_tx_rsp	-
41	TEST_rx_cmd	-
42	TEST_rx_rsp	-
43	TEST_tx_cmd	-
44	TEST_tx_rsp	-
45	SABM_rx_cmd	-
46	SABM_tx_cmd	-
47	SARM_rx_cmd	-
48	SARM_tx_cmd	-

5.1.4.3 llc2_stioc Structure

The `llc2_stioc` structure is formatted as follows:

```
struct llc2_stioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lli_snid;
    llc2stats_t lli_stats;
};
```

The `llc2_stioc` structure contains the following members:

lli_type Always `LLI_STATS`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

lli_stats Contains the `llc2stats_t` structure containing the statistics described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.1.4.4 llc2stats_t Structure

The `llc2stats_t` structure is formatted as follows:

```
typedef struct llc2_stats {
    uint32_t llc2monarray[49];
} llc2stats_t;
```

The `llc2stats_t` structure contains the following members:

llc2monarray

Provides an array of 32-bit unsigned integers containing the statistics. The indices of the individual elements are as follows:

0	`tx_ign`	-
1	`rx_badlen`	-
2	`rx_unknown`	-
3	`t1_exp`	-
4	`t4_exp`	-
5	`t4_n2_exp`	-
6	`RR_rx_cmd`	-
7	`RR_rx_rsp`	-
8	`RR_tx_cmd`	-
9	`RR_tx_rsp`	-
10	`RR_tx_cmd_p`	-
11	`RNR_rx_cmd`	-
12	`RNR_rx_rsp`	-
13	`RNR_tx_cmd`	-
14	`RNR_tx_rsp`	-
15	`RNR_tx_cmd_p`	-
16	`REJ_rx_cmd`	-
17	`REJ_rx_rsp`	-
18	`REJ_tx_cmd`	-
19	`REJ_tx_rsp`	-
20	`REJ_tx_cmd_p`	-
21	`SABME_rx_cmd`	-
22	`SABME_tx_cmd`	-
23	`DISC_rx_cmd`	-
24	`DISC_tx_cmd`	-
25	`UA_rx_rsp`	-
26	`UA_tx_rsp`	-
27	`DM_rx_rsp`	-
28	`DM_tx_rsp`	-
29	`I_rx_cmd`	-
30	`I_tx_cmd`	-
31	`FRMR_rx_rsp`	-

```
32    FRMR_tx_rsp          -
33    tx_rtr               -
34    rx_bad               -
35    rx_dud               -
36    rx_ign               -
37    XID_rx_cmd           -
38    XID_rx_rsp           -
39    XID_tx_cmd           -
40    XID_tx_rsp           -
41    TEST_rx_cmd          -
42    TEST_rx_rsp          -
43    TEST_tx_cmd          -
44    TEST_tx_rsp          -
45    I_rx_rsp             -
46    I_tx_rsp             -
47    UI_rx_cmd            -
48    UI_tx_cmd            -
```

5.1.4.5 mlp_stioc Structure

The `mlp_stioc` structure is formatted as follows:

```
struct mlp_stioc {
    uint8_t lli_type;
    uint8_t state;
    uint16_t lli_spare;
    uint32_t lli_snid;
    mlpstats_t lli_stats;
};
```

The `mlp_stioc` structure contains the following members:

lli_type Always `LLI_STATS`.

lli_state Provides the state of the link.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

lli_stats Contains the `mlpstats_t` structure containing the statistics described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.1.4.6 mlpstats_t Structure

The `mlpstats_t` structure is formatted as follows:

```
typedef struct mlp_stats {
    uint32_t mlpmonarray[12];
} mlpstats_t;
```

The `mlpstats_t` structure contains the following members:

mlpmonarray

Contains an array of 32-bit unsigned integers containing the statistics. The indices of the individual elements of the array are as follows:

```
0    MLP_frames_tx    -
```

1	MLP_frames_rx	-
2	MLP_reset_tx	-
3	MLP_reset_rx	-
4	MLP_confs_tx	-
5	MLP_confs_rx	-
6	MLP_slps	-
7	MLP_num_slps	-
8	MLP_mt1_exp	-
9	MLP_mt2_exp	-
10	MLP_mt3_exp	-
11	MLP_mn1_exp	-

5.1.5 LI_GSTATS - Global Statistics Data Structures

5.1.5.1 lapb_gstioc Structure

The `lapb_gstioc` structure is formatted as follows:

```
struct lapb_gstioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lapbgstats[6];
};
```

The `lapb_gstioc` structure contains the following members:

lli_type Always `LLI_GSTATS`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lapbgstats Contains an array of 32-bit unsigned integers containing the statistics. The indices of the individual elements of the array are as follows:

0	frames_tx	-
1	frames_rx	-
2	sabm_tx	-
3	sabm_rx	-
4	bytes_tx	-
5	bytes_rx	-

5.1.5.2 llc2_gstioc Structure

The `llc2_gstioc` structure is formatted as follows:

```
struct llc2_gstioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t llc2gstats[6];
};
```

The `llc2_gstioc` structure contains the following members:

lli_type Always `LLI_GSTATS`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

llc2gstats Contains an array of 32-bit unsigned integers containing the statistics. The indices of the individual elements of the array are as follows:

0	frames_tx	-
1	frames_rx	-
2	sabme_tx	-
3	sabme_rx	-
4	bytes_tx	-
5	bytes_rx	-

5.1.5.3 mlp_gstioc Structure

The `mlp_gstioc` structure is formatted as follows:

```
struct mlp_gstioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t mlpgstats[10];
};
```

The `mlp_gstioc` structure contains the following members:

lli_type Always `LLI_GSTATS`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

mlpgstats Contains an array of 32-bit unsigned integers containing the statistics. The indices of the individual elements of the array are as follows:

0	MLP_frames_tx	-
1	MLP_frames_rx	-
2	MLP_reset_tx	-
3	MLP_reset_rx	-
4	MLP_confs_tx	-
5	MLP_confs_rx	-
6	MLP_slps	-
7	MLP_num_slps	-
8	MLP_bytes_tx	-
9	MLP_bytes_rx	-

5.1.6 LI_LAPBTUNE - LAPB Tuning Data Structures

5.1.6.1 lapb_tnioc Structure

The `lapb_tnioc` structure is formatted as follows:

```
struct lapb_tnioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lli_snid;
    lapbtune_t lapb_tune;
};
```

The `lapb_tnioc` structure contains the following members:

lli_type Always `LLI_LAPBTUNE`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

lapb_tune Contains the `lapbtune_t` tuning structure containing tuning parameters described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.1.6.2 lapbtune_t Structure

The `labtune_t` structure is formatted as follows:

```
typedef struct lapbtune {
    uint16_t N2;
    uint16_t T1;
    uint16_t Tpf;
    uint16_t Trej;
    uint16_t Tbusy;
    uint16_t Tidle;
    uint16_t ack_delay;
    uint16_t notack_max;
    uint16_t tx_window;
    uint16_t tx_probe;
    uint16_t max_I_len;
    uint16_t llconform;
    uint16_t sabm_in_x32;
} lapbtune_t;
```

The `lapbtune_t` structure contains the following members:

N2

T1

Tpf

Trej

Tbusy

Tidle

ack_delay

notack_max
tx_window

tx_probe

max_I_len

llconform

sabm_in_x32

5.1.7 LI_LLC2TUNE - LLC2 Tuning Data Structures

5.1.7.1 llc2_tnioc Structure

The `llc2_tnioc` structure is formatted as follows:

```
struct llc2_tnioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lli_snid;
    llc2tune_t llc2_tune;
};
```

The `llc2_tnioc` structure contains the following members:

lli_type Always `LLI_LLC2TUNE`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

llc2_tune Contains the `llc2tune_t` tuning structure containing tuning parameters described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.1.7.2 llc2tune_t Structure

The `llc2tune_t` structure is formatted as follows:

```
typedef struct llc2tune {
    uint16_t N2;
    uint16_t T1;
    uint16_t Tpf;
    uint16_t Trej;
    uint16_t Tbusy;
    uint16_t Tidle;
    uint16_t ack_delay;
    uint16_t notack_max;
    uint16_t tx_window;
    uint16_t tx_probe;
    uint16_t max_I_len;
    uint16_t xid_window;
    uint16_t xid_Ndup;
    uint16_t xid_Tdup;
} llc2tune_t;
```

The `llc2tune_t` structure contains the following members:

N2

T1

Tpf

Trej

Tbusy

Tidle

ack_delay

notack_max
tx_window

tx_probe

max_I_len

xid_window
xid_Ndup

xid_Tdup

5.1.8 LI_MLPTUNE - MLP Tuning Data Structures

5.1.8.1 mlp_tnioc Structure

The `mlp_tnioc` structure is formatted as follows:

```
struct mlp_tnioc {
    uint8_t lli_type;
    uint8_t lli_spare[3];
    uint32_t lli_snid;
    mlptune_t mlp_tune;
};
```

The `mlp_tnioc` structure contains the following members:

lli_type Always `LLI_MLPTUNE`.

lli_spare Spare bytes for alignment: set to zero by issuer and ignored by responder.

lli_snid Specifies the subnetwork identifier (link identifier) for the operation.

mlp_tune Contains the `mlptune_t` tuning structure containing tuning parameters described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.1.8.2 mlptune_t Structure

The `mlptune_t` structure is formatted as follows:

```
typedef struct mlptune {
    uint16_t mw;
    uint16_t mx;
    uint16_t mt1;
    uint16_t mt2;
    uint16_t mt3;
    uint16_t mn1;
} mlptune_t;
```

The `mlptune_t` structure contains the following members:

mw The size of the MLP window.

mx The size of the MLP guard region.

mt1 The time interval to wait for MN(S) == MV(R) in deciseconds.

mt2 The time interval to wait for unblock in deciseconds.

mt3 The time interval to wait for reset confirmation.

mn1 The number of SLP transmission retries.

5.2 Input-Output Control Commands

`L_SETSNID`	Set subnetwork identifier. (Also `L_SETPPA`.)
`L_GETSNID`	Get subnetwork identifier. (Also `L_GETPPA`.)
`L_SETTUNE`	Set common tuning parameters.
`L_GETTUNE`	Get common tuning parameters.
`L_GETSTATS`	Get subnetwork statistics.
`L_ZEROSTATS`	Zero subnetwork statistics.
`L_TRACEON`	Turn message tracing on.
`L_TRACEOFF`	Turn message tracing off.
`L_GETGSTATS`	Get global statistics.

`L_ZEROGSTATS`	Zero global statistics.
`L_LINKDISABLE`	Disable link (subnetwork identifier).
`L_LINKENABLE`	Enable link (subnetwork identifier).
`L_PUTX32MAP`	Put X.32 table mapping.
`L_GETX32MAP`	Get X.32 table mapping.
`M_SETSNID`	Set subnetwork identifier.
`M_GETSNID`	Get subnetwork identifier.
`M_SETTUNE`	Set common tuning parameters.
`M_GETTUNE`	Get common tuning parameters.
`M_GETSTATS`	Get subnetwork statistics.
`M_ZEROSTATS`	Zero subnetwork statistics.
`M_TRACEON`	Turn message tracing on.
`M_TRACEOFF`	Turn message tracing off.
`M_GETGSTATS`	Get global statistics.
`M_ZEROGSTATS`	Zero global statistics.

5.2.1 SLP Input-Output Control Commands

5.2.1.1 L_SETSNID

This input-output control command is used to set the subnetwork (link) identifier associated with the Stream upon which the input-output control command is issued. *Solaris X.25* calls this input-output control command `L_SETPPA`. The argument of this input-output control command is a pointer to the `ll_snioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.2 L_GETSNID

This input-output control command is used to retrieve the subnetwork (link) identifier associated with the Stream upon which the input-output control command is issued. *Solaris X.25* calls this input-output control command `L_GETPPA`. The argument of this input-output control command is a pointer to the `ll_snioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.3 L_SETTUNE

This input-output control command is used to set the tunable parameters associated with the LAPB or LLC2 link associated with the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `lapb_tnioc`, `llc2_tnioc` or `mlp_tnioc` structure.[2]

5.2.1.4 L_GETTUNE

This input-output control command is used to retreive the tunable parameters associated with the LAPB or LLC2 link associated with the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `lapb_tnioc`, `llc2_tnioc` or `mlp_tnioc` structure.[3]

[2] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[3] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.5 L_GETSTATS

This input-output control command is used to retrieve the LAPB or LLC2 link-specific statistics associated with the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `lapb_stioc`, `llc2_stioc` or `mlp_stioc` structure.[4]

5.2.1.6 L_ZEROSTATS

This input-output control command is used to zero the LAPB or LLC2 link-specific statistics associated with the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `lapb_stioc`, `llc2_stioc` or `mlp_stioc` structure.[5]

5.2.1.7 L_TRACEON

This input-output control command is used to initiate tracing of messages on the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `ll_hdioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.8 L_TRACEOFF

This input-output control command is used to terminate tracing of messages on the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `ll_hdioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.9 L_GETGSTATS

This input-output control command is used to retrieve global statistics for the driver associated with the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `lapb_gstioc`, `llc2_gstioc` or `mlp_gstioc` structure.[6]

5.2.1.10 L_ZEROGSTATS

This input-output control command is used to zero the global statistics for the driver associated with the Stream upon which the input-output control command is issued. The argument of this input-output control command is a pointer to the `lapb_gstioc`, `llc2_gstioc` or `mlp_gstioc` structure.[7]

5.2.1.11 L_LINKDISABLE

The argument of this input-output control command is a pointer to the `ll_hdioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

---

[4] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[5] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[6] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[7] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.12 L_LINKENABLE

The argument of this input-output control command is a pointer to the `ll_hdioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.1.13 L_PUTX32MAP

5.2.1.14 L_GETX32MAP

5.2.2 MLP Input-Output Control Commands

5.2.2.1 M_SETSNID

The argument of this input-output control command is a pointer to the `ll_snioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.2.2 M_GETSNID

The argument of this input-output control command is a pointer to the `ll_snioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.2.3 M_SETTUNE

The argument of this input-output control command is a pointer to the `lapb_tnioc`, `llc2_tnioc` or `mlp_tnioc` structure.[8]

5.2.2.4 M_GETTUNE

The argument of this input-output control command is a pointer to the `lapb_tnioc`, `llc2_tnioc` or `mlp_tnioc` structure.[9]

5.2.2.5 M_GETSTATS

The argument of this input-output control command is a pointer to the `lapb_stioc`, `llc2_stioc` or `mlp_stioc` structure.[10]

5.2.2.6 M_ZEROSTATS

The argument of this input-output control command is a pointer to the `lapb_stioc`, `llc2_stioc` or `mlp_stioc` structure.[11]

5.2.2.7 M_TRACEON

The argument of this input-output control command is a pointer to the `ll_hdioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

---

[8] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[9] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[10] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[11] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.2.8 M_TRACEOFF

The argument of this input-output control command is a pointer to the `ll_hdioc` structure described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

5.2.2.9 M_GETGSTATS

The argument of this input-output control command is a pointer to the `lapb_gstioc`, `llc2_gstioc` or `mlp_gstioc` structure.[12]

5.2.2.10 M_ZEROGSTATS

The argument of this input-output control command is a pointer to the `lapb_gstioc`, `llc2_gstioc` or `mlp_gstioc` structure.[13]

[12] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

[13] These structures are described in ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩, and ⟨undefined⟩ [⟨undefined⟩], page ⟨undefined⟩.

6 LLI Header Files

6.1 ll_control.h Header File

```
#ifndef __SYS_SNET_LL_CONTROL_H__
#define __SYS_SNET_LL_CONTROL_H__

#include <sys/types.h>
#include <stdint.h>

#define LAPB_STID       201
#define LLC_STID        202

#define OFF              0
#define START            1
#define D_CONN           2
#define ADM              3
#define ARM              4
#define POLLING          5
#define PRESETUP         6
#define REGISTERING      7
#define SETUP            8
#define RESET            9
#define LL_ERROR        10
#define NORMAL          11
#define ADM_CONN_REQ    12

#define L_SETSNID       ('L'<<8 |  1)
#define L_GETSNID       ('L'<<8 |  2)
#define L_SETTUNE       ('L'<<8 |  3)
#define L_GETTUNE       ('L'<<8 |  4)
#define L_GETSTATS      ('L'<<8 |  5)
#define L_ZEROSTATS     ('L'<<8 |  6)
#define L_TRACEON       ('L'<<8 |  7)
#define L_TRACEOFF      ('L'<<8 |  8)
#define L_GETGSTATS     ('L'<<8 | 10)
#define L_ZEROGSTATS    ('L'<<8 | 11)
#define L_LINKDISABLE   ('L'<<8 | 12)
#define L_LINKENABLE    ('L'<<8 | 13)
#define L_PUTX32MAP     ('L'<<8 | 15)
#define L_GETX32MAP     ('L'<<8 | 16)

#define LI_PLAIN        0x01
#define LI_SNID         0x02
#define LI_STATS        0x04
#define LI_GSTATS       0x16
#define LI_LAPBTUNE     0x13
#define LI_LLC2TUNE     0x23

#ifdef _SUN_SOURCE
#define LI_SPPA         LI_SNID
#define lli_ppa         lli_snid
#endif                          /* _SUN_SOURCE */
```

```
struct ll_hdioc {
        uint8_t lli_type;
        uint8_t lli_spare[3];
        uint32_t lli_snid;
};

struct ll_sntioc {
        union {
                uint8_t lli_type;
                uint8_t lli_status;
        };
        uint8_t lli_spare[3];
        uint8_t lli_snid;

};

struct ll_snioc {
        uint8_t lli_type;
        uint8_t lli_class;
        uint8_t lli_spare[2];
        uint32_t lli_snid;
        uint32_t lli_index;
        uint32_t lli_slp_snid;
        uint32_t lli_slp_pri;
};

typedef struct lapbtune {
        uint16_t N2;                    /* max numb retransmissions */
        uint16_t T1;                    /* ack timer (deciseconds) */
        uint16_t Tpf;                   /* poll timer (deciseconds) */
        uint16_t Trej;                  /* reject timer (deciseconds) */
        uint16_t Tbusy;                 /* busy timer (deciseconds) */
        uint16_t Tidle;                 /* idle timer (deciseconds) */
        uint16_t ack_delay;             /* ack delay timer (deciseconds) */
        uint16_t notack_max;            /* max unacked I-frames */
        uint16_t tx_window;             /* transmit window (no XID) */
        uint16_t tx_probe;              /* window probe position */
        uint16_t max_I_len;             /* max I-frame size */
#define IGN_UA_ERROR            (1<<0)
#define FRMR_FRMR_ERROR         (1<<1)
#define FRMR_INVRSP_ERROR       (1<<2)
#define SFRAME_PBIT             (1<<3)
#define NO_DM_ADM               (1<<4)
#define IGN_DM_ERROR            (1<<5)
        uint16_t llconform;
        uint16_t sabm_in_x32;
} lapbtune_t;

struct lapb_tnioc {
        uint8_t lli_type;               /* LI_LAPBTUNE */
        uint8_t lli_spare[3];
        uint32_t lli_snid;
        struct lapbtune lapb_tune;
};

typedef struct llc2tune {
```

```
        uint16_t N2;                    /* max numb retransmissions */
        uint16_t T1;                    /* ack timer (deciseconds) */
        uint16_t Tpf;                   /* poll timer (deciseconds) */
        uint16_t Trej;                  /* reject timer (deciseconds) */
        uint16_t Tbusy;                 /* busy timer (deciseconds) */
        uint16_t Tidle;                 /* idle timer (deciseconds) */
        uint16_t ack_delay;             /* ack delay timer (deciseconds) */
        uint16_t notack_max;            /* max unacked I-frames */
        uint16_t tx_window;             /* transmit window (no XID) */
        uint16_t tx_probe;              /* window probe position */
        uint16_t max_I_len;             /* max I-frame size */
        uint16_t xid_window;            /* XID window size (receive window) */
        uint16_t xid_Ndup;              /* dup MAC XID count */
        uint16_t xid_Tdup;              /* dup MAC XID time (deciseconds) */
} llc2tune_t;

struct llc2_tnioc {
        uint8_t lli_type;               /* LI_LLC2TUNE */
        uint8_t lli_spare[3];
        uint32_t lli_snid;
        struct llc2tune llc2_tune;
};

#define tx_ign                   0
#define rx_badlen                1
#define rx_unknown               2
#define t1_exp                   3
#define t4_exp                   4
#define t4_n2_exp                5
#define RR_rx_cmd                6
#define RR_rx_rsp                7
#define RR_tx_cmd                8
#define RR_tx_rsp                9
#define RR_tx_cmd_p             10
#define RNR_rx_cmd              11
#define RNR_rx_rsp              12
#define RNR_tx_cmd              13
#define RNR_tx_rsp              14
#define RNR_tx_cmd_p            15
#define REJ_rx_cmd              16
#define REJ_rx_rsp              17
#define REJ_tx_cmd              18
#define REJ_tx_rsp              19
#define REJ_tx_cmd_p            20
#define SABME_rx_cmd            21
#define SABME_tx_cmd            22
#define DISC_rx_cmd             23
#define DISC_tx_cmd             24
#define UA_rx_rsp               25
#define UA_tx_rsp               26
#define DM_rx_rsp               27
#define DM_tx_rsp               28
#define I_rx_cmd                29
#define I_tx_cmd                30
#define FRMR_rx_rsp             31
#define FRMR_tx_rsp             32
```

```
#define tx_rtr                  33
#define rx_bad                  34
#define rx_dud                  35
#define rx_ign                  36
#define XID_rx_cmd              37
#define XID_rx_rsp              38
#define XID_tx_cmd              39
#define XID_tx_rsp              40
#define TEST_rx_cmd             41
#define TEST_rx_rsp             42
#define TEST_tx_cmd             43
#define TEST_tx_rsp             44

#define SABM_rx_cmd             45
#define SABM_tx_cmd             46
#define SARM_rx_cmd             47
#define SARM_tx_cmd             48
#define lapbstatmax             49

typedef struct lapb_stats {
        uint32_t lapbmonarray[lapbstatmax];
} lapbstats_t;

struct lapb_stioc {
        uint8_t lli_type;
        uint8_t state;
        uint16_t lli_spare;
        uint32_t lli_snid;
        lapbstats_t lli_stats;
};

#define I_rx_rsp                45
#define I_tx_rsp                46
#define UI_rx_cmd               47
#define UI_tx_cmd               48
#define llc2statmax             49

typedef struct llc2_stats {
        uint32_t llc2monarray[llc2statmax];
} llc2stats_t;

struct llc2_stioc {
        uint8_t lli_type;
        uint8_t lli_spare[3];
        uint32_t lli_snid;
        llc2stats_t lli_stats;
};

#define frames_tx                0
#define frames_rx                1
#define sabm_tx                  2
#define sabm_rx                  3
#define sabme_tx                 2
#define sabme_rx                 3
#define bytes_tx                 4
#define bytes_rx                 5
```

```
#define globstatmax              6

struct lapb_gstioc {
        uint8_t lli_type;
        uint8_t lli_spare[3];
        uint32_t lapbgstats[globstatmax];
};

struct llc2_gstioc {
        uint8_t lli_type;
        uint8_t lli_spare[3];
        uint32_t llc2gstats[globstatmax];
};

typedef union lli_union {
        struct ll_hdioc ll_hd;
        struct ll_snioc ll_sn;
        struct lapb_tnioc lapb_tn;
        struct lapb_stioc lapb_st;
        struct lapb_gstioc lapb_gst;
        struct llc2_tnioc llc2_tn;
        struct llc2_stioc llc2_st;
        struct llc2_gstioc llc2_gst;
} lliun_t;

#endif                          /* __SYS_SNET_LL_CONTROL_H__ */
```

6.2 ll_proto.h Header File

```
#ifndef __SYS_SNET_LL_PROTO_H__
#define __SYS_SNET_LL_PROTO_H__

#define LC_LLC1         15
#define LC_LLC2         16
#define LC_LAPBDTE      17
#define LC_LAPBXDTE     18
#define LC_LAPBDCE      19
#define LC_LAPBXDCE     20
#define LC_LAPDTE       21
#define LC_LAPDCE       22
#define LC_HDLC         27
#define LC_HDLCX        28

#define LS_FAILED        0
#define LS_SUCCESS       1
#define LS_EXHAUSTED     2
#define LS_RESETDONE     3
#define LS_CONFLICT      4
#define LS_DISCONNECT    5
#define LS_RST_FAILED    6
#define LS_RST_REFUSED   7

#endif                          /* __SYS_SNET_LL_PROTO_H__ */
```

6.3 mlp_control.h Header File

```
#ifndef __SYS_SNET_MLP_CONTROL_H__
#define __SYS_SNET_MLP_CONTROL_H__

#include <sys/snet/ll_control.h>

#define M_SETSNID       ('M'<<8| 1)
#define M_GETSNID       ('M'<<8| 2)
#define M_SETTUNE       ('M'<<8| 3)
#define M_GETTUNE       ('M'<<8| 4)
#define M_GETSTATS      ('M'<<8| 5)
#define M_ZEROSTATS     ('M'<<8| 6)
#define M_TRACEON       ('M'<<8| 7)
#define M_TRACEOFF      ('M'<<8| 8)
#define M_GETGSTATS     ('M'<<8|10)
#define M_ZEROGSTATS    ('M'<<8|11)

/* MLP TUNING */

#define LI_MLPTUNE      0x33

typedef struct mlptune {
        uint16_t mw;                    /* MLP window */
        uint16_t mx;                    /* MLP window in guard region */
        uint16_t mt1;                   /* lost frame timer */
        uint16_t mt2;                   /* group busy timer */
        uint16_t mt3;                   /* MLP reset confirmation timer */
        uint16_t mn1;                   /* */
} mlptune_t;

struct mlp_tnioc {
        uint8_t lli_type;               /* always LI_MLPTUNE */
        uint8_t lli_spare[3];           /* for alignment only */
        uint8_t lli_snid;               /* subnetwork (link) ID */
        mlptune_t mlp_tune;             /* MLP tuning parameters */
};

/* MLP STATISTICS */

/* shared global and subnetwork ID stats */
#define MLP_frames_tx           0
#define MLP_frames_rx           1
#define MLP_reset_tx            2
#define MLP_reset_rx            3
#define MLP_confs_tx            4
#define MLP_confs_rx            5
#define MLP_slps                6
#define MLP_num_slps            7

/* MLP GLOBAL STATISTICS */

/* global stats */
#define MLP_bytes_tx            8
#define MLP_bytes_rx            9
```

```
#define MLP_gstatmax            10

struct mlp_gstioc {
        uint8_t lli_type;               /* always LI_STATS */
        uint8_t lli_spare[3];           /* for alignment only */
        uint32_t mlpgstats[MLP_gstatmax];       /* global MLP stats */
};

/* MLP SNID STATISTICS */

/* subnetwork ID specific stats */
#define MLP_mt1_exp             8
#define MLP_mt2_exp             9
#define MLP_mt3_exp             10
#define MLP_mn1_exp             11
#define MLP_statmax             12

typedef struct mlp_stats {
        uint32_t mlpmonarray[MLPSTATMAX];       /* MLP subnetwork statistics */
} mlpstats_t;

struct mlp_stioc {
        uint8_t lli_type;               /* always LI_GSTATS */
        uint8_t state;
        uint16_t lli_spare;             /* for alignment only */
        uint32_t lli_snid;              /* subnetwork (link) ID */
        mlpstats_t lli_stats;           /* subnetwork (link) MLP stats */
};

/* UNION OF MLP STRUCTURES */

typedef union mlp_union {
        struct ll_hdioc ll_hd;          /* link ID structure */
        struct ll_snioc ll_sn;          /* subnetwork ID structure */
        struct mlp_tnioc mlp_tn;        /* MLP tuning structure */
        struct mlp_stioc mlp_st;        /* MLP statistics structure */
        struct mlp_gstioc mlp_gst;      /* MLP global statistics structure */
} mlpiun_t;

#endif                          /* __SYS_SNET_MLP_CONTROL_H__ */
```

6.4 mlp_proto.h Header File

```
#ifndef __SYS_SNET_MLP_PROTO_H__
#define __SYS_SNET_MLP_PROTO_H__

#define LC_MLAPBDTE     23
#define LC_MLAPBXDTE    24
#define LC_MLAPBDCE     25
#define LC_MLAPBXDCE    26
#define LC_MLP          27

#endif                          /* __SYS_SNET_MLP_PROTO_H__ */
```

7 LLI Library

8 LLI Drivers and Modules

8.1 hdlc Driver

The `hdlc` driver is a pseudo-device STREAMS driver that provides raw HDLC framing as specified in *ISO/IEC 3309* described in reference [ISO3309], page 55.

The `hdlc` driver provides a *Data Link Provider Interface* as its upper multiplex service interface, as described in [DLPI], page 55.

The `hdlc` driver is documented in the `hdlc(4)` manual page.

8.2 lapb Driver

The `lapb` driver is a pseudo-device STREAMS driver that provides the X.25 compatible LAPB procedures as specified in *ISO/IEC 7776* described in reference [ISO7776], page 55. This driver, in combination with the `x25` multiplexing driver, provides X.25 over LAPB as described in reference [ISO8208], page 55.

The `lapb` driver provides a *Data Link Provider Interface* as its upper multiplex service interface, as described in [DLPI], page 55.

The `lapb` driver is documented in the `lapb(4)` manual page.

8.3 llc1 Driver

The `llc1` driver is a pseudo-device STREAMS driver that provides the IEEE 802.2 LLC Type 1 (LLC1) procedures as specified in *ISO/IEC 8802-2* described in reference [ISO8802-2], page 55. This driver, in combination with the `x25` multiplexing driver, provides X.25 over LLC1 as described in refeference [ISO8881], page 56.

The `llc1` driver provides a *Data Link Provider Interface* as its upper multiplex service interface, as described in [DLPI], page 55.

The `llc1` driver is documented in the `llc1(4)` manual page.

8.4 llc2 Driver

The `llc1` driver is a pseudo-device STREAMS driver that provides the IEEE 802.2 LLC Type 2 (LLC2) procedures as specified in *ISO/IEC 8802-2* described in reference [ISO8802-2], page 55. This driver, in combination with the `x25` multiplexing driver, provides X.25 over LLC2 as described in refeference [ISO8881], page 56.

The `llc2` driver provides a *Data Link Provider Interface* as its upper multiplex service interface, as described in [DLPI], page 55.

The `llc2` driver is documented in the `llc2(4)` manual page.

8.5 s_llic Module

The `s_llic` module is a pushable STREAMS module that converts between the LLI input-output controls described in this manual and the DLPI input-output controls used by the OpenSS7 Project. This particular module does not convert primitives, as both the LLI and the DLPI use the primitives of the *Data Link Provider Interface, Revision 2.0.0* described in reference [DLPI], page 55.

The `s_llic` module provides a *Data Link Provider Interface* as its upper and lower service interface, as described in [DLPI], page 55.

The `s_llic` module is documented in the `s_llic(4)` manual page.

8.6 s_lli Module

The `s_lli` module is a pushable STREAMS module that converts between the LLI input-output controls described in this manual and the DLPI input-output controls used by the OpenSS7 Project. This particular module converts between the LLI primitives described in this manual and the primitives of the *Data Link Provider Interface*, described in reference [DLPI], page 55.

The `s_lli` module provides a *Data Link Provider Interface* as its upper and lower service interface, as described in [DLPI], page 55.

The `s_lli` module is documented in the `s_lli(4)` manual page.

9 LLI Utilities

9.1 lltune Utility

The `lltune` utility is documented in the `lltune(8)` manual page.

9.2 mlptune Utility

The `mlptune` utility is documented in the `mlptune(8)` manual page.

9.3 linkadd Utility

The `linkadd` utility is documented in the `linkadd(8)` manual page.

9.4 linkdel Utility

The `linkdel` utility is documented in the `linkdel(8)` manual page.

9.5 linklist Utility

The `linklist` utility is documented in the `linklist(8)` manual page.

9.6 linkreset Utility

The `linkreset` utility is documented in the `linkreset(8)` manual page.

9.7 linkstart Utility

The `linkstart` utility is documented in the `linkstart(8)` manual page.

9.8 linkstate Utility

The `linkstate` utility is documented in the `linkstate(8)` manual page.

9.9 linkstop Utility

The `linkstop` utility is documented in the `linkstop(8)` manual page.

10 LLI File Formats

10.1 lapbtemplate File Format

The `lapbtemplate` file format is documented in the `lapbtemplate(5)` manual page.

Name

`lapbtemplate` — Link Access Protocol (Balanced) File Format

Description

The `lapbtemplate` describes the file format for input to the `lltune(8)` command for LAPB class subnetworks. The file format consists of a number of parameter values, one per line, formatted as described below. Each parameter value is described using its line number in the file, a parameter name, and a description of the format of the value. Only the value appears in the file, each value on a line by itself, one value per line.

Each of the LAPB configuration parameters corresponds to the member and values of the `lapb_tune` structure, that is carried in a `lapb_tnioc` structure by the `L_LAPBTUNE` input-output control command.

These protocol parameters, and the default values that exist when tuning has not been applied to a newly created LAPB subnetwork, correspond directly to the protocol parameters and defaults in *ISO/IEC 7776*, *ITU-T Rec. X.25* and *X.75*.

Format

The LAPB template consists of 16 to 18 lines containing the following configuration information:

1. `N2_VAL` is the maximum number of times that a protocol data unit (PDU) is set following the expiry of the acknowlegement timer, the P-bit timer, or the reject timer. It also limits the number of times an RR with the P-bit set is sent when remote busy is true and the busy timer expires.
2. `T1_VAL` is the time during which the LAPB expects to receive an acknowledgement to an outstanding I-PDU or an expected response to a sent UI-PDU. The value is in units of 0.1 seconds (deciseconds).
3. `TPF_VAL` is the time during which the LAPB expectes to receive a PDU with the F-bit set to 1 in response to a command with the P-bit set to 1. The value should be less than the acknowledgement timer. The value is in units of 0.1 seconds (deciseconds).
4. `TREJ_VAL` is the time interval during which the LAPB expects to receive a reply to a sent REJ DPU. The value is in units of 0.1 seconds (deciseconds).
5. `TBUSY_VAL` is the time interval during which the LAPB waits for an indication of the clearance of a busy condition at the other LAPB. The value is in units of 0.1 seconds (deciseconds).
6. `IDLE_VAL` is the time interval during which the LAPB expects to receive a PDU from the other LAPB. If it expires then the P/F cycle is intiated which may result in link disconnection. The value is in units of 0.1 seconds (decicseconds).
7. `ACK_DELAY` is the maximum delay in 0.1 second units before transmitting a delayed RR. This must be considerably less that the acknowledgement timer value, `T1_VAL`.
8. `NOTACK_MAX` is the maximuum number of unacknowledged receive I PDUs before the RR acknowledging them all must be sent.

9. `LOC_WIND` is the number of unacknowledged I PDUs that may be sent.
10. `LOC_PROBE` is the position before the window is closed at which an I PDU is sent with the P-bit set to solicit an acknowledgement from the receiver.
11. `MAX_I_LEN` is the maximum isze of a LAPB I-frame. LAPB requires all incoming I-frames above a certain size to be rejected by a FRMR. This parameter specifies the maximum size. It is constructured as the sum of the maximum X.25 data size, the X.25 protocol length and the LAPB protocol length.
12. `IGN_UA_ERROR` define whether or not to ignore any UA frams received, when the connection is in ERROR state. The value is '`1`' for *true* and '`0`' for *false*. The defalt value is *false*.
13. `FRMR_FRMR_ERROR` defines whether or not to re-transmit a frame reject if a frame reject is received, when the connection is in `ERROR` state. The value is '`1`' for *true* and '`0`' for *false*. The default value is *false*.
14. `FRMR_INVRSP_ERROR` defines whether or not to transmit a frame reject if an invalid frame response is received, when the connection is in `ERROR` state. The value is '`1`' for *true* and '`0`' for *false*. The default value is *false*.
15. `SFRAME_PBIT` defines whether or not to send a frame reject if an S-frame is received without the P-bit set. The value is '`1`' for *true*a nd '`0`' for *false*. The default value is *false*.
16. `NO_DM_ADM` defines whether or not to send a `DM` on entry to `ADM` state after an N2 count expiry. The value is '`1`' for *true* and '`0`' for *false*. The default value is *false*.

The following two fields are optional extensions:

17. `IGN_DM_ERROR` defines whether or not to ignore DM frames received, when the connection is in `ERROR` state. The value is '`1`' for *true* and '`0`' for *false*. The default value is false.
18. `SABM_IN_X32` defines the action to take when a SABM is received in X.32 setup. The value is '`1`' for *true* and '`0`' for *false*. The default value is *false*.

The last two fields ('`17`' and '`18`') are enhancements.

Files

Files following this format are normally kept in the `/etc/sysconfig/openss7/template/` directory.[1]

See Also

- `lltune(8)`
- `lapb(4)`
- `x25netd(8)`

Compatibility

The `lapbtemplate` file format is compatible with *Spider X.25*, and implementations based on *Spider X.25*, such as *AIXlink/X.25*, *HP-UX*, *IRIS SX.25*, *Solstice X.25*, *PT X.25*, *SBE X.25*, with the following compatibility considerations:

— Most implementations only define the first 16 lines. This implementation defines 18 lines, where the first 16 lines are compatible with other implementations and the last additional two lines are optional.

[1] Note that this directory varies depending on whether the build was on a `dpkg(1)`-based or `rpm(1)`-based system.

— *PT X.25* documents the `SABM_IN_X32` LAPB template field but not the `IGN_DM_ERROR` LAPB template field. *Solstice X.25* and *IRIS SX.25* do not document either the `IGN_DM_ERROR` nor `SABM_IN_X32` LAPB template fields.

For additional compatiblity infromation see, `lapb(4)`, and `STREAMS(9)`.

Conformance

AIXlink/X.25, *HP-UX*, *IRIS SX.25*, *Solstice X.25*, *PT X.25*, *SBE X.25*, documentation.

History

The `lapbtemplate` file format first appeared in *Spider X.25*.

10.2 llc2template File Format

The `llc2template` file format is documented in the `llc2template(5)` manual page.

Name

`llc2template` — Logical Link Control Type 2 File Format

Description

The `llc2template` describes the file format for input to the `lltune(8)` command for LLC2 class subnetworks. The file format consists of a number of parameter values, one per line, formatted as described below. Each parameter value is described using its line number in the file, a parameter name, and a description of the format of the value. Only the value appears in the file, each value on a line by itself, one value per line.

Each of the LLC2 configuration parameters corresponds to the member and values of the `llc2_tune` structure, that is carried in a `llc2_tnioc` structure by the `L_LLC2TUNE` input-output control command.

These protocol parameters, and the default values that exist when tuning has not been applied to a newly created LLC2 subnetwork, correspond directly to the protocol parameters and defaults in *ISO/IEC 8802-2:1998*.

Format

The LLC2 template consists of 14 lines containing the following configuration information.

1. `N2_VAL` is the maximum number of times that a Protocol Data Unit (PDU) is sent following the expiry of the acknowledgement timer, the P-bit timer, or the reject timer. This parameter also limits the number of times an RR is sent with the P-bit set when remote busy is true and the busy timer expires.
2. `T1_VAL` is the time interval during which the LLC2 expects to receive an acknowledgement to an outstanding I-PDU or an expected response to a sent UI-PDU. The value is in units of 0.1 seconds.
3. `TPF_VAL` is the time during which the LLC2 expects to receive a PDU with the F-bit set to 1 in response to a command with the P-bit set to 1. The value should be less than that specified for the akcnowledgement timer. The value is in units of 0.1 seconds.
4. `TREJ_VAL` is the time interval during which the LLC2 expects to receive a reply to a sent REJ PDU. The value is in units of 0.1 seconds.
5. `TBUSY_VAL` is the timer interval during which the LLC2 waits for an indication of the clearance of busy condition at the other LLC2. The value is in units of 0.1 seconds.
6. `TIDLE_VAL` is the time interval during which the LLC2 expects to receive a PDU from the other LLC2. The value is in units of 0.1 seconds.
7. `ACK_DELAY` is the RR delay time. This is the time interval for which the LLC2 will withhold acknowledgements of unacknowledged received I-PDUs. The value is in units of 0.1 seconds.
8. `NOTACK_MAX` is the maximum number of unacknowledged received I-frames.
9. `TX_WINDOW` is the transmit window (if no XID received).
10. `TX_PROBE` is the position before the window is closed at which an I-PDU is sent with the P-bit set to solicit an acknowledgement from the receiver.
11. `MAX_I_LEN` is the maximum size of an LLC2 I-frame. LLC2 requires all incoming I-frames above a certain size to be rejected by a FRMR. This parameter specifies the maximum size of data that may bereceived starint from the LLC2 protocol header.

In an X.25 network, it is constructed as (maximum X.25 data length + X.25 protool header length + LLC2 protocol header length). In an SNA network, it is constructed as (maximum SNA data length + SNA request header length + SNA transmission header length + LLC2 protocol header length).

12. `XID_WINDOW` is the XID window size (receive window), when the remote window size is unknown or zero.
13. `XID_NDUP` is the duplicate MAC XID count (0 means no test).
14. `XID_TDUP` is the duplicate MAC XID time. The value is in units of 0.1 seconds.

Files

Files following this format are normally kept in the `/etc/sysconfig/openss7/template/` directory.[1]

See Also

- `lltune(8)`
- `llc2(4)`
- `x25netd(8)`

Compatibility

The `llc2template` file format is compatible with *Spider X.25*, and implementations based on *Spider X.25*, such as *AIXlink/X.25*, *HP-UX*, *IRIS SX.25*, *Solstice X.25*, *PT X.25*, *SBE X.25*, with the following compatibility considerations:

— *PT X.25* does not support LLC2. *OpenSS7* supports LLC2 in support of XOL and porting applications from *AIXlink/X.25*, *Solstice X.25*, *HP-UX*, *IRIS SX.25*, *VxWorks*, *pSOS*, *SpiderX*, and many other implementations based on *SpiderX.25* support LLC2. Portable X.25 and XOL applications will use *OpenSS7* instead of *PT X.25*.

For additional compatibiltiy information see, `llc2(4)`, and `STREAMS(9)`.

Conformance

AIXlink/X.25, *HP-UX*, *IRIS SX.25*, *Solstice X.25*, *PT X.25*, *SBE X.25*, documentation.

History

The `llc2template` file format first appeared in *Spider X.25*.

[1] Note that this directory varies depending on whether the build was on a `dpkg(1)`-based or `rpm(1)`-based system.

10.3 mlptemplate File Format

The `mlptemplate` file format is documented in the `mlptemplate(5)` manual page.

Appendix A LLI Compatibility and Porting

A.1 Compatibility with AIXlink/X.25

A.2 Compatibility with HP X.25/9000

A.3 Compatibility with IRIS SX.25

A.4 Compatibility with PT X.25

A.5 Compatibility with RadiSys WAN

A.6 Compatibility with SBE X.25

A.7 Compatibility with Solstice X.25

Glossary of LLI Terms and Acronyms

ANSI	American National Standards Institute
CCITT	Old name for ITU-T
CONS	Connection-Oriented Network Service
ENSDU	Expedited Network Service Data Unit
ETSI	European Telecommunications Standards Institute
IEEE	Institute of Electrical and Electronics Engineers
ITU	International Telecommunications Union
ITU-T	ITU Telecom Sector
LCI	Logical Channel Identifier
LLC1	Logical Link Control Type 1
LLC2	Logical Link Control Type 2
LLC	Logical Link Control
MAC	Media Access Control
NLI	Network Layer Interface
NPDU	Network Protocol Data Unit
NSAP	Network Service Access Point
NSDU	Network Service Data Unit
NSP	Network Service Provider
NS	Network Service
NSU	Network Service User
PDU	Protocol Data Unit
PVC	Permanent Virtual Circuit
SAP	Service Access Point
SDU	Service Data Unit
VC	Virtual Circuit
X.121	ITU-T Recommendation X.121
X.25	ITU-T Recommendation X.25
X.29	ITU-T Recommendation X.29

References

[AIXlink/X.25] *AIXlink/X.25 Version 2.1 for AIX: Guide and Reference*, No: SC23-2520-07, Eighth Edition, September 2006, (Bolder, CO), International Business Machine Corp., IBM. IBM Documentation Library.

[ARTIC WAN] *ARTIC STREAMS Support WAN Driver Interface Reference*, Release 1.7, June 2004, (Hillsboro, OR), RadiSys Corporation, RadiSys. [Doc No: 007-01232-0003], RadiSys Support Documentation.

[CDI] *OpenSS7 CAE Specification: Communications Device Interface (CDI) Specification*, Revision 0.9.2, Draft 2, July 15, 2007, (Edmonton, Canada), B. Bidulock, OpenSS7 Corporation. Distributed with package `strxns-0.9.2` and `openss7-0.9.2`. OpenSS7 Documents.

[DLPI] *Open Group CAE Specification: Data Link Provider Interface (DLPI) Specification*, Revision 2.0.0, Draft 2, August 20, 1992, (Parsippany, New Jersey), UNIX International, Inc., UNIX International Press. The Open Group, The OpenSS7 Project.

[IRIS SX.25] *IRIS SX.25 NLI Programmer's Guide*, 1995, (Mountainview, CA), Silicon Graphics, Inc., SGI Technical Publications. [No: 007-2268-002]. SGI Technical Publications.

[ISO3309] **ISO/IEC 3309:1988**, *Information Technology – Data Communications – High-Level Data Link Control procedures – Frame Structure*, 1988, ISO/IEC, International Orgianization for Standardization. International Organization for Standardization.

[ISO7776] **ISO/IEC 7776:1995**, *Information technology — Telecommunications information exchange between systems — High-level data link control procedures — Description of the X.25 LAPB-compatible DTE data link procedures*, Second Edition, July 1, 1995, International Organization for Standardization. International Organization for Standardization.

[ISO8208] **ISO/IEC 8208:2000**, *Information Technology — Data Communications — X.25 Packet Layer Protocol for Data Terminal Equipment*, Fourth Edition, November 1, 2000, (Geneva), ISO/IEC, International Organization for Standardization. International Organization for Standardization.

[ISO8802-2] **ANSI/IEEE Standard 802.2-1998 [ISO/IEC 8802-2:1998]**, *IEEE Standard for Information Technology — Telecommunications and Information Exchange Between Systems — Local and Metropolitan Area Networks — Specific Requirements — Part 2: Logical Link Control*, May 7, 1998, (New York), ANSI/IEEE, IEEE Computer Society. [ISBN 1-55937-959-6]. Institute of Electrical and Electronics Engineers.

[ISO8881] **ISO/IEC 8881:1989**, *Information Processing Systems — Data Communications — User of the X.25 Packet Level Protocol in Local Area Networks*, 1989, ISO/IEC, International Organization for Standardization. International Organization for Standardization.

[NLI] *OpenSS7 CAE Specification: Network Layer Interface (NLI) Specification*, Revision 0.9.2, Draft 1, June 2008, (Edmonton, AB), B. Bidulock, OpenSS7 Corporation. Distributed with package `strx25-0.9.2` and `openss7-0.9.2`. The OpenSS7 Project.

[NPI] *Open Group CAE Specification: Network Provider Interface (NPI) Specification*, Revision 2.0.0, Draft 2, August 17, 1992, (Parisppany, New Jersey), UNIX International, Inc., UNIX International Press. The OpenSS7 Project.

[Solstice X.25] *Solstice X.25 9.2 Administration Guide*, October 1999, (Palo Alto, CA), Sun Microsystems, Inc., Sun. [Part No: 806-1234-10], Solaris Documentation.

[TPI] *Open Group CAE Specification: Transport Provider Interface (TPI) Specification*, Revision 2.0.0, Draft 2, 1999, (Berkshire, UK), Open Group, Open Group Publication. The Open Group, The OpenSS7 Project.

[V.25 bis] **ITU-T Recommendation V.25 bis (10/96)**, *Synchronous and asyncrhonous automatic dialing procedrues on switched networks*, October 1996, (Geneva), ITU, ITU-T Telecommunication Standardization Sector of ITU, (Previously "CCITT Recommendation"), `http://www.itu.int/rec/T-REC-V.25bis/en/T-REC-V.25bis`.

[X.21] **ITU-T Recommendation X.21 (09/92)**, *Interface between Data Terminal Equipment (DTE) and Data Circuit-terminating Equiment (DCE) for synchronous operation on Public Data Networks*, September 1992, (Geneva), ITU, ITU-T Telecommunication Standardization Sector of ITU. (Previously "CCITT Recommendation"), T-REC-X.21.

[X.21 bis] **ITU-T Recommendation X.21 bis (03/88)**, *Use on Public Data Networks of Data Terminal Equipment (DTE) which is designed for interfacing to synchronous V-series modems*, March 1988, (Geneva), ITU, ITU-T Telecommunication Standardization Sector of ITU. (Previously "CCITT Recommendation"), T-REC-X.21bis.

[X.25] **ITU-T Recommendation X.25**. T-REC-X.25.

[X.75] **ITU-T Recommendation X.75**. T-REC-X.75.

[X.29] **ITU-T Recommendation X.29**. T-REC-X.29.

[XX25] *X/Open CAE Specification: X.25 Programming Interface using XTI (XX25)*, No. c411, November 1995, (Berkshire, UK), X/Open, Open Group Publication. [ISBN: 1-85912-136-5]. The Open Group.

Licenses

All code presented in this manual is licensed under the [GNU Affero General Public License], page 57. The text of this manual is licensed under the [GNU Free Documentation License], page 67, with no invariant sections, no front-cover texts and no back-cover texts. Please note, however, that it is just plain wrong to modify statements of, or attribute statements to, the Author or *OpenSS7 Corporation*.

GNU Affero General Public License

The GNU Affero General Public License.
Version 3, 19 November 2007

Preamble

The GNU Affero General Public License is a free, copyleft license for software and other kinds of works, specifically designed to ensure cooperation with the community in the case of network server software.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, our General Public Licenses are intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

Developers that use our General Public Licenses protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License which gives you legal permission to copy, distribute and/or modify the software.

A secondary benefit of defending all users' freedom is that improvements made in alternate versions of the program, if they receive widespread use, become available for other developers to incorporate. Many developers of free software are heartened and encouraged by the resulting cooperation. However, in the case of software used on network servers, this result may fail to come about. The GNU General Public License permits making a modified version and letting the public access it on a server without ever releasing its source code to the public.

The GNU Affero General Public License is designed specifically to ensure that, in such cases, the modified source code becomes available to the community. It requires the operator of a network server to provide the source code of the modified version running there to the users of that server. Therefore, public use of a modified version, on a publicly accessible server, gives the public access to the source code of the modified version.

An older license, called the Affero General Public License and published by Affero, was designed to accomplish similar goals. This is a different license, not a version of the Affero GPL, but Affero has released a new version of the Affero GPL which permits relicensing under this license.

The precise terms and conditions for copying, distribution and modification follow.

Terms and Conditions

0. Definitions.

   "This License" refers to version 3 of the GNU Affero General Public License.

   "Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

   "The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

   To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

   A "covered work" means either the unmodified Program or a work based on the Program.

   To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

   To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

   An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

   The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

   A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

   The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

   The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the

source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

   a. The work must carry prominent notices stating that you modified it, and giving a relevant date.

b. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e. Convey the object code using peer-to-peer transmission, provided you inform other peers

where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d. Limiting the use for publicity purposes of names of licensors or authors of the material; or

e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

   You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

    Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

    An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

    You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

    A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

    A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

    Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

    In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

    If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to

downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others’ Freedom.

    If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Remote Network Interaction; Use with the GNU General Public License.

    Notwithstanding any other provision of this License, if you modify the Program, your modified version must prominently offer all users interacting with it remotely through a network (if your version supports such interaction) an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge, through some standard or customary means of facilitating copying of software. This Corresponding Source shall include the Corresponding Source for any work covered by version 3 of the GNU General Public License that is incorporated pursuant to the following paragraph.

    Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the work with which it is combined will remain governed by version 3 of the GNU General Public License.

14. Revised Versions of this License.

    The Free Software Foundation may publish revised and/or new versions of the GNU Affero General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU Affero General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU Affero General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU Affero General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
one line to give the program's name and a brief idea of what it does.
Copyright (C) year name of author

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU Affero General Public License as published by
the Free Software Foundation, either version 3 of the License, or (at
your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
Affero General Public License for more details.

You should have received a copy of the GNU Affero General Public License
along with this program.  If not, see http://www.gnu.org/licenses/.
```

Also add information on how to contact you by electronic and paper mail.

If your software can interact with users remotely through a network, you should also make sure that it provides a way for users to get its source. For example, if your program is a web application, its interface could display a "Source" link that leads users to an archive of the code. There are many ways you could offer source, and different solutions will be better for different programs; see section 13 for the specific requirements.

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU AGPL, see http://www.gnu.org/licenses/.

GNU Free Documentation License

GNU FREE DOCUMENTATION LICENSE
Version 1.3, 3 November 2008

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

   If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

   If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

   If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

   It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

   You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

   A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

   B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

   C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

   D. Preserve all the copyright notices of the Document.

   E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

   F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

   You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

   You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with. . . Texts." line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index

M

N

P

R

S

T

U

X