Network Working Group
Request for Comments: 1307

J. Young
A. Nicholson
Cray Research, Inc.

March 1992

# Dynamically Switched Link Control Protocol

## Status of this Memo

This memo defines an Experimental Protocol for the Internet community. Discussion and suggestions for improvement are requested. Please refer to the current edition of the "IAB Official Protocol Standards" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

## Abstract

This memo describes an experimental protocol developed by a project team at Cray Research, Inc., in implementing support for circuit-switched T3 services. The protocol is used for the control of network connections external to a host, but known to the host. It is documented here for the benefit of others who may wish to perform further research.

While working with circuit-switched T3 networks, developers at Cray Research, Inc., defined a model wherein a host would generate control messages for a network switch. This work is described in RFC 1306, "Experiences Supporting By-Request Circuit-Switched T3 Networks". In order to simplify the model it was decided that the inconsistencies of switch control should be hidden from the host generating the control messages. To that end, a protocol was defined and implemented. This RFC documents the Dynamically Switched Link Control Protocol (DSLCP), which is used for creation and control of downstream network links by a host.

## 1.0 INTRODUCTION

The Dynamically Switched Link Control Protocol (DSLCP) allows a host with knowledge of a special downstream network link to issue messages to control the status of that link.

This document describes the functions of the DSLCP to control external network connections.

## 1.1 Motivation

Circuit Switched Networks are becoming available to the Internet community. These networks are made available by requesting a connection through a switch. Normally circuit switched network links are disconnected, and their prohibitive cost suggests that it is very costly to leave them connected at all times.

Internet users and hosts wish to send data over a circuit switched networks, but only connect the network links when a transport connection is to be established. While it would be possible to use packet routers to identify the need for switching a connection on and off, only the transport provider can positively identify the beginning and end of a transport session. There must be a mechanism to activate and deactivate the link at the beginning and end of a transport session.

The DSLCP assumes that a transport provider has knowledge of a downstream link which must be setup before data transfer may take place. However, the details of link setup may vary by the type of link (circuit-switched or other), specific hardware, or administrative differences. The DSLCP hides these details from the transport provider by offering a simple request/release model of link preparation. The model assumes an entity in control of the link which handles the details of connection preparation while responding to the DSLCP commands of the transport provider. This entity is called the link controller.

The DSLCP allows internet hosts to dynamically change the fabric of the internet by sending messages through the internet in advance of data which is to travel across the newly created links.

## 1.2 Scope

DSLCP is intended to provide an interface between transport providers and arbitrary network links requiring creation, control, setup, or conditioning before data communications may take place.

## 1.3 Interfaces

There are no specific user level interfaces to DSLCP, although they are not precluded. Link control is a function of the network layer, initiated by requests from the transport provider.

A DSLCP transaction is defined as a transport provider communicating with a link controller for the duration of transport session. A network path between the host providing transport services and the link controller must exist in advance of the DSLCP transaction.

Either party to an DSLCP transaction may asynchronously generate messages.

### 1.4 Operation

The purpose of the DSLCP is to allow a transport provider to request the setup of a downstream network link so that data transfer may take place through that link. DSLCP messages are assumed to be communicated between the transport provider and the link controller through a transport service, such as UDP or TCP, or through a network service such as IP.

DSLCP provides messages for link setup and teardown. All the details of link management are left to the link controller. The transport provider is interested only whether the link is ready to carry data.

### 1.5 Transmission

DSLCP messages are carried through the network in datagrams using either IP or UDP. DSLCP is designed to not require a reliable transport protocol.

## 2.0 DSLCP Architecture

DSLCP is used in a host environment. Normally, transport users on the host will make requests of a transport provider to carry data to other hosts. Some of these requests may require the preparation of a downstream network link. The transport provider has knowledge of these special network links, and issues a request to DSLCP that the link be prepared to carry data. This happens transparently to the transport user.

When a transport user requests transport services, the transport provider will normally attempt to establish a connection. In the event the transport provider discovers that the connection requires special link control, the transport provider will call upon DSLCP to send a link setup message to the link controller. The transport provider does not attempt to use the connection until DSLCP informs the transport provider that the link is setup or that the setup attempt failed. If the setup failed, then the transport provider is free to attempt to find another way to create a connection.

When the transport user is finished using the services, then the transport provider will call DSLCP to release the link. The transport provider may now assume that the link is no longer available.

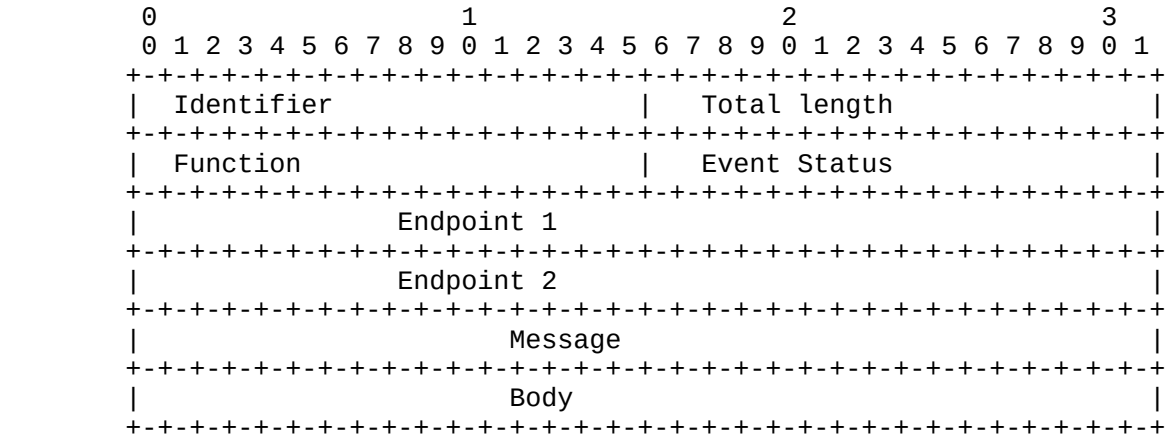
In general, DSLCP maintains and hides the status of link control

transactions from the transport provider. This way the transport provider does not need to keep track of multiple DSLCP transactions. For example, if the transport provider requests a link be setup for a new transport user while another transport user has the link active, the DSLCP may inform the transport provider that the link is ready without delay, provided that the link can support multiple transport connections.

## 3.0 FUNCTIONAL SPECIFICATION

This document specifies both a message format and a state machine for DSLCP protocol transactions.

### 3.1 Control Message Format




Identifier: 16 bits

The identifier is a value assigned by the DSLCP used to uniquely identify link setup transactions. It is intended to be used with the endpoint addresses by a link controller to identify a transaction.

Total length: 16 bits

The total length, in octets, including the header of this DSLCP control message.

Function: 16 bits

The operation to be processed or being responded to.

Functions currently defined are:

Bring up value 0
Bring down value 1

Event Status: 16 bits

The state of the controlled link, relative to the last function request.

The possible event states are:

| | |
|---|---|
| Setup request succeeded | value 2 |
| Setup request failed | value 3 |
| Teardown request succeeded | value 4 |
| Teardown request failed | value 5 |
| Asynchronous network down | value 7 |

Endpoint addresses: 32 bits each

The internet addresses of the two communicating parties for which the link is being prepared.

Message body: arbitrary length up to 65499 octets

An ascii string which is meaningful the link controller. When the requesting host is configured, the system administrator sets the control strings for each network link that may be accessed by the requesting host.

## 3.2 State Machine

The transport provider is aware of only 2 possible states for the controlled link: up or down. Furthermore, transport users may request or release transport services from the transport provider at any time. Thus, there must be a state machine employed by DSLCP when communicating between the transport provider and the controlled link. This state machine hides the details of link control transactions from the transport provider. The state machine has 6 possible states.

Down: There is no active transport connection and the controlled link is not setup.

Coming Up: A transport user has requested a connection for which the transport provider has given a setup request to the DSLCP. The DSLCP has sent a setup request to the link controller and is awaiting a response.

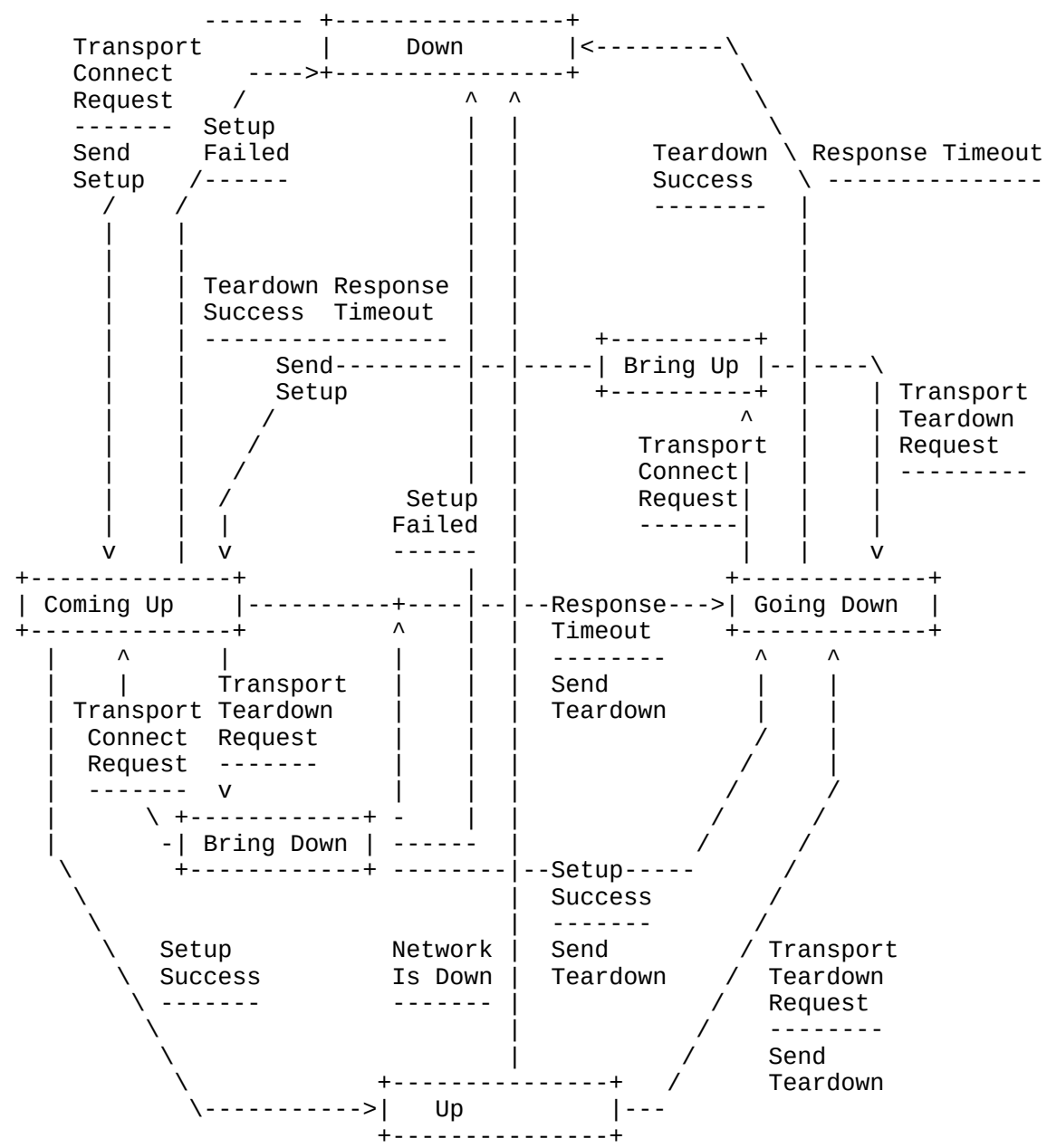
Up: At least one transport connection is active and the controlled link is setup.

Going Down: All transport connections have been terminated and the transport provider has sent an equivalent number of up requests and down requests to the DSLCP. The DSLCP has sent a teardown request to the link controller and is awaiting a response.

Bring Down: While DSLCP is in the Coming Up state, the transport provider requested link teardown. As soon as a response is received from the link controller, the DSLCP will send a teardown request if the link setup was successful.

Bring Up: While in the Going Down state, the transport provider requested connection setup. As soon as a response is received from the link controller, the DSLCP will send a setup request.

DSLCP state diagram:



Events and State Transitions

The DSLCP will process three type of events:

A link control request from the transport provider
An DSLCP message from the link controller
DSLCP message timeout

The transport provider will make link setup and and teardown requests to the DSLCP when transport users request and release services requiring link control operations. The transport provider should not keep track of the status of a particular link, as this is a function of the DSLCP. The transport provider may be unaware of redirection or other processing of link setup requests performed by DSLCP, so this is a function best left to DSLCP. The DSLCP will inform the transport provider as to the success or failure of a particular setup request, and transport providers may assume the success of teardown requests (the DSLCP will always return a success response to a teardown request).

The DSLCP will engage in link control transactions with link controllers. This will include accepting messages from link controllers in response to requests as well as unexpected messages from the link controller. Unexpected messages may include redundant responses to redundant requests sent as a result of timeouts.

Because of the possibility of unavailable links and link controllers, DSLCP should not wait indefinitely for message responses from link controllers to which it has sent messages. Since DSLCP does not require the use of a reliable transport protocol to carry DSLCP messages, DSLCP must have a timeout and retransmission mechanism. Since we have used DSLCP in a local network context with switch controllers which offer a quick turnaround (on the order of 1 second), we use a 5 second timeout with a 3 retransmit limit. These figures would require adaptation to different network and link controller configurations, and a self-adapting algorithm would be most appropriate for a general solution.

The specific events of interest to the DSLCP are:

Transport provider link setup request
Transport provider link teardown request

Link setup request failed
Link setup request succeeded
Link teardown request succeeded
Link teardown request failed
Network link is down

Timeout waiting for DSLCP response from link controller

The necessary processing for each event while in each state is as follows:

Transport provider link setup request

Down:
Send setup request to link controller.
Enter Coming Up state.
Notify transport provider to wait until link is up.

Coming Up:
Bring Up:
Notify transport provider to wait until link is up.

Up:
Notify transport provider that link is up.

Bring Down:
Enter Coming Up state.
Notify transport provider to wait until link is up.

Going Down:
Enter Bring Up state.
Notify transport provider to wait until link is up.

Discussion:

If the controlled link is not capable to support multiple transport connections, then the DSLCP must return appropriate errors when it detects multiple transport setup requests for that link.

Transport provider link teardown request.

Down:
Bring Down:
Going Down:
Notify transport provider that link is down.

Coming Up:
Enter Bring Down state.
Notify transport provider that link is down.

Bring Down:
Notify transport provider that link is down.

```
            Up:
                Send teardown request.
                Enter Going Down state.
                Notify transport provider that link is down.

        Link setup request failed

            Down:
            Going Down:
            Bring Up:
                Unexpected message, possibly due to duplicate requests -
                ignore it.

            Up:
                Unexpected message, link controller may be refusing
                multiple setup requests sent because of timeout - ignore
                it.

            Coming Up:
            Bring Down:
                Enter down state.

        Link setup request succeeded

            Down:
                Unexpected message, possibly due to duplicate requests
                and reordering of request packets by network.
                Send teardown request.

            Going Down:
            Bring Up:
            Up:
                Unexpected message, possibly due to duplicate requests -
                ignore it.

            Coming Up:
                Enter Up state.
                Notify transport provider(s) waiting for link that it is
                available.

            Bring Down:
                Send teardown request.
                Enter Going Down state.

        Link teardown request succeeded

            Down:
            Coming Up:
```

Bring Down:
Unexpected message, possibly due to duplicate requests - ignore it.

Up:
Unexpected message, possibly due to duplicate requests and reordering of request packets by network.
Send teardown request.
Enter Going Down state.
Notify transport providers that link has gone down.

Bring Up:
Send setup request
Enter Coming Up state

Going Down:
Enter Down state

Discussion:

If a teardown request succeeded message arrives when the DSLCP is in the UP state, then some error has occurred, and the conservative approach is to bring down the connection and resynchronize. However, it may be satisfactory to ignore the message without ill effect.

Link teardown request failed

Down:
Coming up:
Bring Down:
Bring Up:
Going Down:
Up:
DSLCP sent a teardown request message for an invalid transaction. The link controller has no identifier/endpoints transaction record for the request. Continue as if request had succeeded.

Network link is down

Down:
Ignore message.

Bring Down:
Going Down:
Enter Down state.

```
            Coming up:
            Bring Up:
            Up:
                Enter down state.
                Notify transport provider that link is down.


        Timeout waiting for DSLCP response from controller

            Down:
            Up:
                DSLCP protocol error - fix bug, don't set timer when
                there are no outstanding requests.

            Coming Up:
            Bring Down:
                Send teardown request.
                Enter Going down state.

            Going Down:
                Enter Down state.

            Bring Up:
                Send setup request.
                Enter Coming Up state.
```

## References

[1] Nicholson, et. al., "High Speed Networking at Cray Research", Computer Communications Review, January, 1991.

[2] Nicholson, A., and J. Young, "Experiences Supporting By-Request Circuit-Switched T3 Networks", RFC 1306, Cray Research, Inc., March 1992.

## Security Considerations

Security issues are not discussed in this memo.

## Authors' Addresses

Jeff Young
Cray Research, Inc.
655F Lone Oak Drive
Eagan, MN 55123

Phone: (612) 452-6650
EMail: jsy@cray.com

Andy Nicholson
Cray Research, Inc.
655F Lone Oak Drive
Eagan, MN 55123

Phone: (612) 452-6650
EMail: droid@cray.com