

Meer begrip voor en probleemoplossing van SDLC voor LLC-netwerkmediaomzetting

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Conventies](#)

[SDLLC](#)

[SDLC-configuratie](#)

[SDLLC-configuratie](#)

[Debugging SDLLC](#)

[DLSw-mediaomzetting](#)

[Opdrachten tonen](#)

[SDLC-pakketten reinigen tijdens DLSw/SDLC voor PU2.1](#)

[DLSw-mediavertaalvoorbeeld](#)

[DLSw: uitvoeren van omgekeerde mediaconversie](#)

[Lokale DLSw-mediaomzetting](#)

[Gerelateerde informatie](#)

[Inleiding](#)

Dit document biedt informatie om een Synchronous Data Link Control (SDLC) met Logical Link Control (LLC)-netwerkmediaconversie te begrijpen en oplossen.

[Voorwaarden](#)

[Vereisten](#)

Er zijn geen specifieke vereisten van toepassing op dit document.

[Gebruikte componenten](#)

Dit document is niet beperkt tot specifieke software- en hardware-versies.

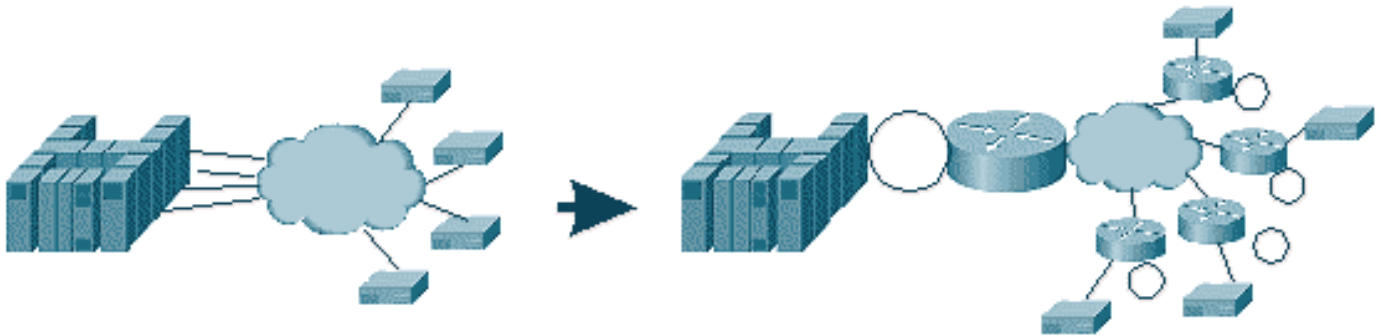
[Conventies](#)

Raadpleeg [Cisco Technical Tips Conventions \(Conventies voor technische tips van Cisco\)](#) voor meer informatie over documentconventies.

SDLLC

SDLC-to-LAN conversie (SDLLC) wordt gebruikt voor het converteren van een SDLC-sessie voor een Physical Unit 2 (PU2.0)-apparaat naar een Logical Link Control-sessie van type 2 (LLC2). Dit is erg nuttig als u een grote hoeveelheid afstandscontrollers hebt die in één Token Ring-poort op een front-end processor (FEP) worden ingevoerd.

De linkerkant van dit diagram toont een FEP met veel SDLC lijnen die naar verre plaatsen verhuizen. De rechterkant van dit diagram toont hetzelfde scenario met Cisco-routers.



Dankzij de routers kan de FEP alleen de Token Ring-interface hebben. Vanaf dat punt, zijn er meerdere verre plaatsen die SDLLC aan de gastheer uitvoeren, evenals het regelmatige bron-route bridge (SRB) verkeer.

Opmerking: Het gebruik van SDLLC voor LLC voor SDLC-conversie is alleen van toepassing op PU2.0-apparaten, niet voor Physical Unit type 2.1 (PU2.1). PU2.1 wordt ondersteund in Data-Link Switching (DLSw).

Om SDLLC te configureren hebt u een SRB in de router nodig. Raadpleeg het gedeelte [Begrip en Problemen oplossen Local Source-Route Bridging](#) voor informatie over hoe u een SRB kunt configureren.

SDLC-configuratie

Omdat SDLLC vanuit een SDLC-interface converteert, hebt u SDLC eerst correct ingesteld. Volg deze stappen om SDLC te configureren:

1. Geef de opdracht **insluitingssdcl uit** om de seriële insluiting in SDLC te wijzigen.
2. Geef de opdracht de **primaire rol van sdcl uit** om de rol van de router in primaire positie in de lijn SDLC te veranderen. **Opmerking:** In Serial Tunnelling (STUN) omgevingen zijn er primaire en secundaire rollen. Raadpleeg het gedeelte [Seriële tunneling \(STUN\) configureren en oplossen](#) voor meer informatie.
3. Geef het **sdcl adres xx** opdracht uit om SDLC stemadres te configureren.

SDLLC-configuratie

Om SDLLC te configureren heeft de eerste opdracht een **overtrek**. Deze opdracht definieert waarnaar SDLC converteert in de LLC2-omgeving. Volg deze stappen om SDLLC te configureren:

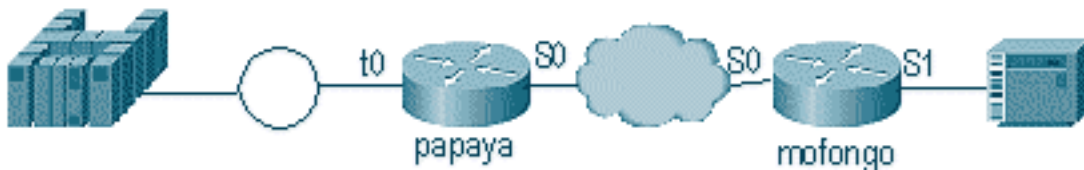
1. Geef de opdracht **sdllc-overstap xxxx.xx.xx00 lr bn tr uit** om SDLLC-mediaconversie op

een seriële interface mogelijk te maken. Deze opdracht vertelt de router het virtuele MAC-adres van het SDLC-station. Vervolgens specificeert de opdracht het lokale ring nummer (**lr**), het brugnummer (**bn**) en het doelring nummer (**tr**). De **lr** moet uniek zijn in het netwerk. De **bn** kan een waarde van 1 tot 15 zijn. De **trn** moet de virtuele ring in de router zijn. Als u lokale SDLLC configureren kunt u dit punt maken naar een virtuele ring of naar een interface (fysieke ring die aangesloten is op de Token Ring-interface) in de router. **Opmerking:** de laatste twee cijfers van het MAC-adres in deze opdracht zijn **00**. U kunt de laatste twee cijfers van de **overstap niet instellen** omdat de router deze cijfers gebruikt om het SDLC-adres van deze regel in te voegen. Als u de laatste twee cijfers specificeert, vervangt de router hen met het SDLC-adres. Dan antwoordt de host niet voor dat MAC-adres.

Bijvoorbeeld, als overstap MAC is geconfigureerd als 4000.1234.5678 en SDLC adres 0x01, gebruikt de router de MAC van 4000.1234.5601 om het SDLC-apparaat in het LLC-domein te vertegenwoordigen. Bovendien is de transdr MAC in niet-kanonisch formaat, dat het zelfde formaat zoals Token Ring frame is.

- Geef de opdracht **sdllc xid-adres xxxxxx** uit om de waarde voor de ruilidentificatie (XID) te specificeren die geschikt is voor het SDLC-station om VTAM-waarden (Virtual Telecommunications Access Methode) aan te passen. Dit wordt bepaald aan de hand van IDBLK en IDNUM in het hoofdknooppunt van de switch in VTAM. Als dit niet overeenkomt, mislukt de XID-uitwisseling.
- Geef de opdracht van de **partner sdllc-adres sdllc-adres uit om verbindingen voor SDLLC mogelijk te maken**. Dit specificeert het MAC-adres van de partner, wat gewoonlijk de host is.

Er wordt een eenvoudige SDLLC-voorbeeldconfiguratie weergegeven. De SDLC aangesloten controller verschijnt als een lokaal Token Ring-apparaat dat op de FEP is aangesloten.

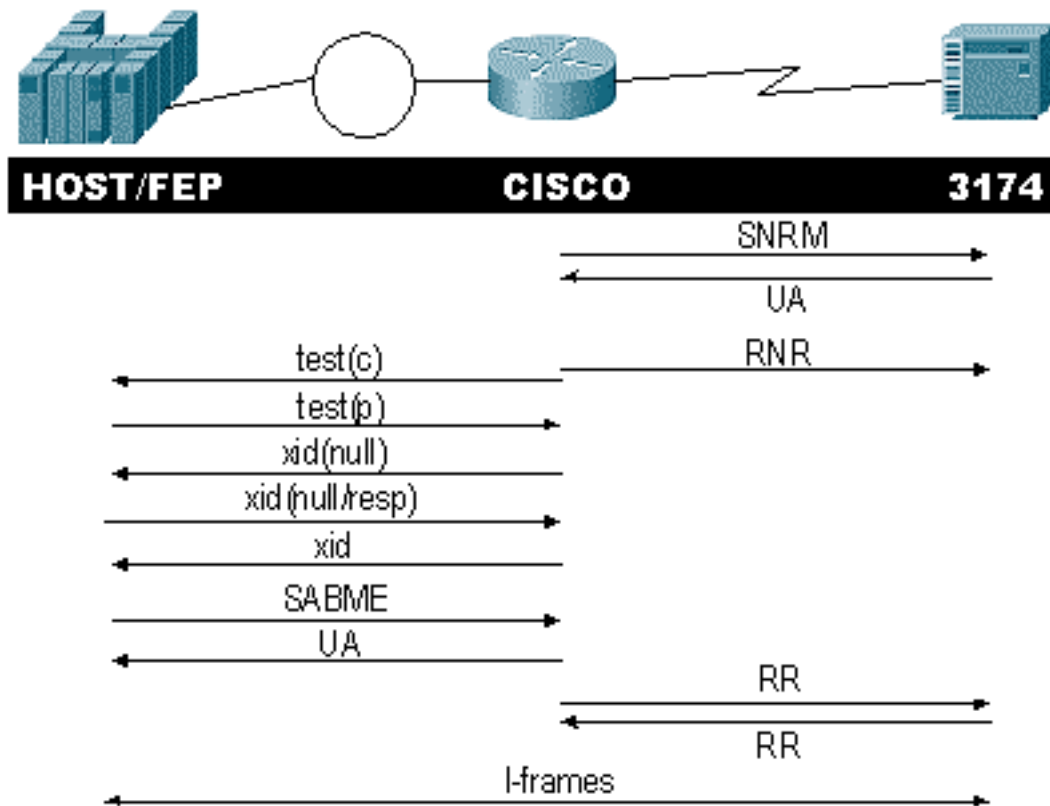


Papaya	Mofongo
<pre>source-bridge ring-group 100 source-bridge remote- peer 100 tcp 1.1.1.1 source-bridge remote- peer 100 tcp 1.1.2.1 local-ack interface tokenring 0 ip address 1.1.3.1 255.255.255.0 source-bridge 33 2 100 source-bridge spanning interface loopback 0 ip address 1.1.1.1 255.255.255.0</pre>	<pre>source-bridge ring group 100 source-bridge remote-peer 100 tcp 1.1.2.1 source-bridge remote-peer 100 tcp 1.1.1.1 local-ack source-bridge sdllc local-ack interface serial 0 encapsulation sdllc-primary sdllc address c6 sdllc traddr 4000.3174.1100 333 3 100 sdllc partner 4000.1111.1111 c1 sdllc xid c1 17200c6 interface loopback 0 ip address 1.1.2.1 255.255.255.0</pre>

[Debugging SDLLC](#)

Een SDLLC-probleem vereist dat u twee verschillende omgevingen probleemoplossing: de SDLC-wereld en de Logical Link Control, type 2 (LLC2)-wereld naar de plaats waar u de frames vertaalt. Omdat u slechts één type controller kunt hebben, is het debuggen van SDLLC makkelijker te begrijpen dan data-link switching (DLSw)/SDLC.

Let eerst op de stromen voor deze specifieke sessie opstarten:



Controleer de reactie op de standaard-responsmodus (SNRM) van de controller. De router start het LLC-gedeelte niet totdat het SDLC-deel in bedrijf is.

Geef deze opdrachten uit om de SNRM-respons te controleren:

- **sdlc_staat**
- **sdlc_staat**

In dit voorbeeld wordt SNRM naar de controller verzonden, die de status van de lijn in SNRMSSENT wijzigt. Als de router in deze staat blijft, heeft hij de ongenummerde ontvangstbevestiging (UA) van de controller niet ontvangen. Dit kan betekenen dat er iets mis is met de SDLC-lijn. Als dit voorkomt, wordt het debug weergegeven in de vorm:

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed state to up
s4f#
SDLLC_STATE: Serial1 C6 DISCONNECT
-> SDLC PRI WAIT
SDLC_STATE: (5234984) Serial1 C6 DISCONNECT
-> SNRMSSENT
%SYS-5-CONFIG_I: Configured from console by console
%LINK-3-UPDOWN: Interface Serial1, changed state to up
Serial1 SDLC output      C693
Serial1 SDLC input       C673
SDLC_STATE: (5235700) Serial1 C6 SNRMSSENT
-> CONNECT
SDLLC_STATE: Serial1 C6 SDLC PRI WAIT
```

```
-> NET UP WAIT
SDLC_STATE: (5235700) Serial1 C6 CONNECT
-> USBUSY
```

Als de router de UA ontvangt, zal **sdlc_state** van SNRM_SENT naar CONNECT bewegen. Daarna beweegt de SDLLC staat van SDLC_PRI_WAIT naar NET_UP_WAIT. Wanneer dit voorkomt, kan de router beginnen de LLC kant van de verbinding op te halen. De laatste actie is het begin van het verzenden van een niet-gereed(e) bericht(en) naar de SDLC-lijn. Hierdoor kan de controller geen informatie verzenden totdat de LLC-zijde in gebruik is.

Daarna, verstuurt de router een ontdekkingsreiziger om de plaats van zijn partner te vinden.

```
SDLLC: O TEST, dst 4000.1111.1111 src 4000.3174.11c6 dsap 0 ssap 0
To0: out: MAC: acfc: 0x8040 Dst: 4000.1111.1111 Src: c000.3174.11c6 bf: 0x82 0x304A210
To0: out: RIF: 8800.14D3.0642.0210
To0: out: LLC: 0000F300 00800000 000C3BF0 7D000000 00800000 000C3BF0 ln: 25
SDLLC: NET UP WAIT rcv FORWARD TEST P/F(F3) 4000.3174.11c6 c000.1111.1111 00 01 -> Serial1
C6
caching rif
```

De bovenstaande uitvoer geeft de testenquête weer die wordt verzonden en ontvangen. Omdat dit voorbeeld een lokaal aangesloten controller en Token Ring heeft, laat de testenquête de router op zoek naar het partneradres. Nadat de router het testkader ontvangt, begint het de XID uitwisseling. De router caches het Routing Information Field (RIF) voor deze sessie, die u kunt verifiëren met de opdracht **Show Rf**. Omdat dit een PU2.0 is, verstuurt de router een Format 0 Type 2 XID naar de host na de reactie op de XID-waarde.

```
SDLLC: O xid(null), 4000.1111.1111 4000.3174.11c6 4 4 [1000.14D3.0641.0051.12C2.0194.01F1.02C0]
SDLLC: NET UP WAIT rcv FORWARD XID P/F(BF) 4000.3174.11c6 c000.1111.1111 04 05
-> Serial1 C6
SDLLC: O xid(0T2), 4000.1111.1111 4000.3174.11c6 4 4 [1000.14D3.0641.0051.12C2.0194.01F1.02C0]
SDLLC: NET UP WAIT rcv FORWARD SABME P/F(7F) 4000.3174.11c6 c000.1111.1111 04 04
-> Serial1 C6
SDLLC: SABME for Serial1 C6 in NET UP WAIT
%SDLLC-5-ACT_LINK: SDLLC: Serial1 LINK address C6 ACTIVATED: Net connect
SDLLC_STATE: Serial1 C6 NET UP WAIT -> CONNECT
```

Na de XID-uitwisseling ontvangt de router de Set Asynchronous Balanced Mode Extended (SABME) van de host. Dit voltooit de opstartprocedure en de router reageert met een UA op de host. Nu, de staat van de lijn SDLC verandert van USBUSY aan CONNECT, en de I-frames kunnen door de router gaan.

```
SDLC_STATE: (5235944) Serial1 C6 USBUSY
-> CONNECT
Serial1 SDLC output C611
Serial1 SDLC input C611
s4f#
```

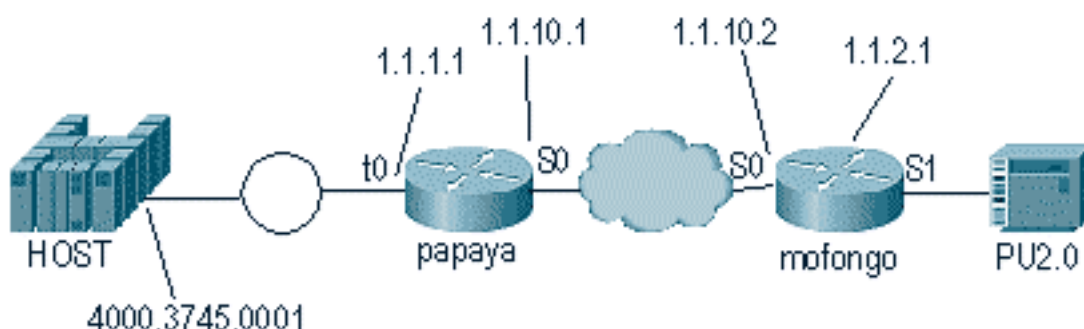
[DLSw-mediaomzetting](#)

DLSw biedt een belangrijke verbetering van mediavertaling omdat deze PU2.1 ondersteunt. Hierdoor kan SDLLC worden geconverteerd naar LLC2 voor controllers, zoals de 5494 en de 5394 (met upgradeoptie naar PU2.1 - IBM RPQ 8Q0775) naar AS/400. Dit verwijdert de noodzaak voor STUN en slechte AS/400 multipoint lijnen.

De configuratieparameters voor DLSw-mediavertaling zijn een beetje anders dan de SDLLC-parameters. Er is één DLSw-opdracht die wordt toegevoegd, de rest zijn SDLC-opdrachten. Voltooi deze stappen om DLSw-mediavertaling te configureren:

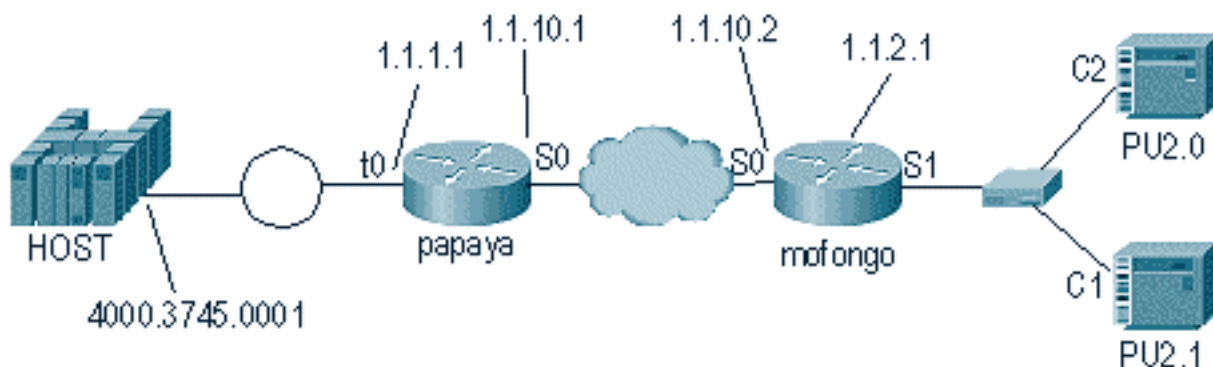
1. Geef de opdracht **insluitingssdlc uit** om de seriële insluiting in SDLC te wijzigen. Omdat u de lijn SDLC in de router wilt beëindigen, moet de router als primair voor opiniepeilingsdoeleinden handelen. Dit is anders dan STUN omdat de hoofdpersoon de HOST of de AS/400 zal zijn.
2. Geef de opdracht de **primaire rol van sdlc uit** om de rol van de router in primaire positie in de lijn SDLC te veranderen.
3. Geef het **sdlc adres xx** opdracht uit om het SDLC stemadres te configureren. Dit is waar DLSw verschilt van SDLLC. In SDLLC, specificeert u opdrachten met het **sdlc-trefwoord**. In DLSw, specificeer opdrachten met het **sdlc-trefwoord**.
4. Geef de opdracht **sdlc vmac xxxx.xx.xx00 uit** om het virtuele MAC-adres voor de SDLC-controller te configureren. Deze parameter vertelt de router het virtuele MAC-adres voor deze SDLC-controller in de LLC2-omgeving. Vergeet niet de laatste byte van **00** te laten, want het stemadres is erbij gevoegd (**sdlc-adres**).
5. Geef de opdracht **sdlc xid nn xxxxxx uit** om de XID voor deze PU 2.0 te configureren. In deze opdracht is **nn het stemadres van de controller** en **xxxxxxx** is de XID voor deze PU2.0 (het IDBLOCK en IDNUM dat is gecodeerd in het hoofdknooppunt van de switch in VTAM). **Opmerking:** als je een PU2.1 hebt, is er onderhandeling over XID. Hierdoor verandert de opdracht.
6. Geef de opdracht **sdlc xid nn xid-poll uit** om de XID voor deze PU 2.1 te configureren. In deze opdracht is **nn het stemadres van het station**.
7. Geef de opdracht **van de partner xxxx.xxxx.xxxx van sdlc uit** om het MAC-adres van de routerpartner te configureren. In deze opdracht is **nn het stemadres van de betrokken controleur**. Het is belangrijk het adres van de controller te specificeren, omdat er in meerdere lijnen een controller kan worden geplaatst die leidt naar de ene host en een andere controller die leidt naar een andere host.
8. Geef de **opdracht sdlc dlsw nn uit** om DLSw te configureren voor de specifieke controller. In deze opdracht is **nn het stemadres van de controller of controllers in de multidrop**. Met deze opdracht kunt u meerdere stemadressen in één opdracht instellen. **Opmerking:** Pas op voor bug #CSCdi75481. Raadpleeg voor meer informatie de [Bug Toolkit](#) (alleen [geregistreerde](#) klanten). Als de opdracht **sdlc dlsw nn** niet is verwijderd voordat u het SDLC-adres van de router wijzigt, kan de CLS-code niet correct DLSw met de SDLC-interface communiceren. Dit defect is vastgesteld in Cisco IOS® software releases 11.1(8.1)11.1(8.1)AA01(01.03) 11.1(8.1)AA01(01.02) en later.

Er wordt een voorbeeldconfiguratie weergegeven voor een DLSw SDLC PU2.0-controller.



Papaya	Mofongo
<pre> source-bridge ring-group 100 dlsw local-peer peer-id 1.1.1.1 dlsw remote-peer 0 tcp 1.1.2.1 ! interface serial 0 ip address 1.1.10.1 255.255.255.0 ! interface tokenring 0 ip address 1.1.1.1 255.255.255.0 ring-speed 16 source-bridge 1 1 100 source-bridge spanning </pre>	<pre> dlsw local-peer peer-id 1.1.2.1 dlsw remote-peer 0 tcp 1.1.1.1 ! interface loopback 0 ip address 1.1.2.1 ! interface serial 0 ip address 1.1.10.2 255.255.255.0 ! interface serial 1 no ip address encapsulation sdhc sdhc role primary sdhc vmac 4000.3174.0000 sdhc address c1 sdhc xid c1 01767890 sdhc partner 4000.3745.0001 c1 sdhc dlsw c1 </pre>

Denk er bij het coderen van een meerdere, aan dat PU2.1s intelligenter is en meer informatie te hebben om uit te wisselen dan een vast PU2.0 apparaat. Dit is belangrijk bij het configureren van een omgeving met meerdere aansluitingen, omdat u de lijn als primair voor het PU2.0-apparaat moet coderen. U moet ook de **xid-enquête** toevoegen voor het SDLC-adres van het PU2.1-apparaat, zodat de code begrijpt wat te doen met elk van de controllers. Dit is een voorbeeld van de configuratie.



Papaya	Mofongo
<pre> source-bridge ring- group 100 dlsw local-peer peer-id 1.1.1.1 dlsw remote-peer 0 tcp 1.1.2.1 ! interface serial 0 ip address 1.1.10.1 255.255.255.0 ! interface tokenring 0 ip address 1.1.1.1 255.255.255.0 ring-speed 16 source-bridge 1 1 100 </pre>	<pre> dlsw local-peer peer-id 1.1.2.1 dlsw remote-peer 0 tcp 1.1.1.1 ! interface loopback 0 ip address 1.1.2.1 ! interface serial 0 ip address 1.1.10.2 255.255.255.0 ! interface serial 1 no ip address encapsulation sdhc sdhc role primary sdhc vmac 4000.3174.0000 sdhc address c1 xid-poll </pre>

source-bridge spanning	sdhc partner 4000.9404.0001 c1 sdhc address c2 01767890 sdhc partner 4000.9404.0001 c2 sdhc dlsw c1 c2
------------------------	-----------------------------------------------------------------------------------------------------------------

Opdrachten tonen

Raadpleeg [Data-Link Switching Plus](#) voor meer informatie over de showopdrachten die voor DLSw-mediaconverting worden gebruikt.

SDLC-pakketten reinigen tijdens DLSw/SDLC voor PU2.1

```
%LINK-3-UPDOWN: Interface Serial2, changed state to up
```

Het eerste wat er gebeurt is een **XID** of **BF** naar het SDLC-adres van **FF**.

```
Serial2 SDLC output      FFBF
```

Daarna wordt er een **XID** ontvangen van de 5494. Dit is een **XID** formaat 2 type 3, dat in deze **debug sdhc**-pakketuitvoer wordt weergegeven:

```
Serial2 SDLC input
0046C930: DDBF3244 07300DD 0000B084 00000000 .....d....
0046C940: 00000001 0B000004 09000000 00070010 .....
0046C950: 17001611 01130012 F5F4F9F4 F0F0F2F0 .....54940020
0046C960: F0F0F0F0 F0F0F0F0 0E0CF4D5 C5E3C14B 00000000..4NETA.
0046C970: C3D7F5F4 F9F4                      CP5494
```

Dit zijn verklaringen van verschillende velden uit deze opdracht:

- **07300DD** — Dit veld bestaat uit het blok-ID en het ID-nummer dat in de 5494 is geconfigureerd. Blok-ID en ID-nummer worden **XID** genoemd en worden door de 5494 tijdens de sessieonderhandeling naar de peer verzonden.
- **NETA** — Dit veld is de Advanced Peer-to-Peer Network Identifier (APPN) die wordt gebruikt. Normaal gesproken komt dit veld overeen met de NETID die in de peer is ingesteld. In dit geval is de peer een AS/400.
- **CP5494** — Dit veld is de naam van het Control Point (CP) van 5494.
- **DD** — Dit veld is het SDLC-adres.

Daarna wordt **XID** ontvangen van de AS/400:

```
Serial2 SDLC output
004BC070:      FFBF 324C0564 52530000 000A0800    ...<.....
004BC080: 00000000 00010B30 0005BA00 00000007    .....
004BC090: 000E0DF4 D5C5E3C1 4BD9E3D7 F4F0F0C1    ...4NETA.RTP400A
004BC0A0: 1017F116 11011300 11F9F4F0 F4C6F2F5    ..1.....9404F25
004BC0B0: F1F0F0F0 F4F5F2F5 F3460505 80000000    100045253.....
004BC0C0:
Serial2 SDLC input
0046C270:                      DDBF3244 07300DD    .....
0046C280: 0000B084 00000000 00000001 0B000004    ...d.....
0046C290: 09000000 00070010 17001611 01130012    .....
```



```

0046C2A0: F5F4F9F4 F0F0F2F0 F0F0F0F0 F0F0F0F0 5494002000000000
0046C2B0: 0E0CF4D5 C5E3C14B C3D7F5F4 F9F4 ..4NETA.CP5494
Serial2 SDLC output
004C0B10: FFBF 324C0564 52530000 00F6C800 ...<.....6H.
004C0B20: 00000080 15010B10 0005BA00 00000007 .....
004C0B30: 000E0DF4 D5C5E3C1 4BD9E3D7 F4F0F0C1 ..4NETA.RTP400A
004C0B40: 1017F116 11011300 11F9F4F0 F4C6F2F5 ..1.....9404F25
004C0B50: F1F0F0F0 F4F5F2F5 F3460505 80150000 100045253.....
004C0B60:
Serial2 SDLC input
0046BBC0: DDBF3244 073000DD 0000B084 00000000 .....d....
0046BBD0: 00000001 0B000004 09000000 00070010 .....
0046BBE0: 17001611 01130012 F5F4F9F4 F0F0F2F0 .....54940020
0046BBF0: F0F0F0F0 F0F0F0F0 0E0CF4D5 C5E3C14B 00000000..4NETA.
0046BC00: C3D7F5F4 F9F4 CP5494

```

- **05645253** — Dit veld bestaat uit de blokID en het ID-nummer van de AS/400.
- **RTP400A** — Dit veld is de CP-naam van de AS/400. De CP-naam wordt gevonden in het DSPNETA-bestand (Display Network Attributes) in het AS/400-bestand.

Vervolgens worden de SNRM (93) en UA (73) op de lijn weergegeven. Vóór SNRM, gebruikt de router altijd het uitzending adres. Vanaf dit punt op, gebruikt de router altijd het eigenlijke stemadres van de DD.

```

Serial2 SDLC output DD93
Serial2 SDLC input DD73
Serial2 SDLC output DD11
Serial2 SDLC input DD11

```

Op dit punt schorst de verbinding vanwege de status Secure Receiver Ready (RR) tussen de router en de 5494.

Opmerking: Als de router waarop u de debug moet uitvoeren andere SDLC interfaces heeft en u niet logt als gebufferd, kan de router schorsen. Het begrip wanneer u een debug aan de terminal tegen houtkap kunt lopen komt met ervaring. Als u het niet zeker weet, gebruikt u altijd de houtkap gebufferd en de opdracht **weblog** om SDLC-debuggs weer te geven

Schakel de controller uit op de AS/400. Dit stelt u in staat om DISK (53) en UA (73) te zien die resulteren aan de SDLC kant van de sessie.

```

Serial2 SDLC output DD53
Serial2 SDLC input DD73

```

DLSw-mediavertaalvoorbeeld



Nadat de interface op en neer komt, begint de router het proces door de locatie van de afstandsbediening te bepalen.

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial4, changed state to up
DLSW Received-ctlQ : CLSI Msg : ID_STN.Ind  dlen: 46
CSM: Received CLSI Msg : ID_STN.Ind  dlen: 46 from Serial4
CSM:  smac 4000.5494.00dd, dmac 4000.9404.0001, ssap 4 , dsap 4
%DLSWC-3-RECVSSP: SSP OP = 4( ICR ) -explorer from peer 10.17.2.198(2065)
DLSw: new_ckt_from_clsi(): Serial4 4000.5494.00dd:4->4000.9404.0001:4
```

Na het ontvangen van het ICR-frame start DLSw de finite state-machine (FSM) voor deze sessie. Dit wordt uitgevoerd door de berichten **REQ_OPNSTN.Reg** en **REQ_OPNSTN.Cfm** tussen DLSw en Cisco Link Services Interface (CLSI).

```
DLSw: START-FSM (488636): event:DLC-Id state:DISCONNECTED
DLSw: core: dlsw_action_a()
DISP Sent : CLSI Msg : REQ_OPNSTN.Reg  dlen: 106
DLSw: END-FSM (488636): state:DISCONNECTED->LOCAL_RESOLVE
```

```
DLSW Received-ctlQ : CLSI Msg : REQ_OPNSTN.Cfm CLS_OK dlen: 106
DLSw: START-FSM (488636): event:DLC-ReqOpnStn.Cnf state:LOCAL_RESOLVE
DLSw: core: dlsw_action_b()
CORE: Setting lf size to FF
```

Na het gesprek met CLSI, stuurt DLSw **sessie opstarten CUR** beelden naar de afstandsrouter. Deze vinden alleen plaats tussen de twee routers.

```
%DLSWC-3-SENDSSP: SSP OP = 3( CUR ) to peer 10.17.2.198(2065) success
DLSw: END-FSM (488636): state:LOCAL_RESOLVE->CKT_START
```

```
%DLSWC-3-RECVSSP: SSP OP = 4( ICR ) from peer 10.17.2.198(2065)
DLSw: 488636 recv FCI 0 - s:0 so:0 r:0 ro:0
DLSw: recv RWO
DLSw: START-FSM (488636): event:WAN-ICR state:CKT_START
DLSw: core: dlsw_action_e()
DLSw: sent RWO
DLSw: 488636 sent FCI 80 on ACK - s:20 so:1 r:20 ro:1
%DLSWC-3-SENDSSP: SSP OP = 5( ACK ) to peer 10.17.2.198(2065) success
DLSw: END-FSM (488636): state:CKT_START->CKT_ESTABLISHED
```

Zodra het circuit is geactiveerd, stuurt de router de **XID** die is opgeslagen en start de **XID**-uitwisseling. Het is belangrijk te begrijpen waar de **XID**'s zijn. In dit voorbeeld betekent de data-link control (DLC)-ID dat de **XID** uit het lokale DLC-station kwam en de WAN-**XID** uit de afstandsrouter of het externe station kwam.

```
DLSw: START-FSM (488636): event:DLC-Id state:CKT_ESTABLISHED
DLSw: core: dlsw_action_f()
DLSw: 488636 sent FCA on XID
%DLSWC-3-SENDSSP: SSP OP = 7( XID ) to peer 10.17.2.198(2065) success
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED
```

```
%DLSWC-3-RECVSSP: SSP OP = 7( XID ) from peer 10.17.2.198(2065)
DLSw: 488636 recv FCA on XID - s:20 so:0 r:20 ro:0
DLSw: START-FSM (488636): event:WAN-XID state:CKT_ESTABLISHED
DLSw: core: dlsw_action_g()
DISP Sent : CLSI Msg : ID.Rsp  dlen: 12
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED
```

```
%DLSWC-3-RECVSSP: SSP OP = 7( XID ) from peer 10.17.2.198(2065)
DLSw: START-FSM (488636): event:WAN-XID state:CKT_ESTABLISHED
DLSw: core: dlsw_action_g()
DISP Sent : CLSI Msg : ID.Reg  dlen: 88
```

```
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

DLsw Received-ctlQ : CLSI Msg : ID.Ind   dlen: 82
DLSw: START-FSM (488636): event:DLC-Id state:CKT_ESTABLISHED
DLSw: core: dlsw_action_f()
%DLSWC-3-SENDSSP: SSP OP = 7( XID ) to peer 10.17.2.198(2065) success
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

%DLSWC-3-RECVSSP: SSP OP = 7( XID ) from peer 10.17.2.198(2065)
DLSw: START-FSM (488636): event:WAN-XID state:CKT_ESTABLISHED
DLSw: core: dlsw_action_g()
DISP Sent : CLSI Msg : ID.Rsp   dlen: 88
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

DLsw Received-ctlQ : CLSI Msg : ID.Ind   dlen: 82
DLSw: START-FSM (488636): event:DLC-Id state:CKT_ESTABLISHED
DLSw: core: dlsw_action_f()
%DLSWC-3-SENDSSP: SSP OP = 7( XID ) to peer 10.17.2.198(2065) success
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED

%DLSWC-3-RECVSSP: SSP OP = 7( XID ) from peer 10.17.2.198(2065)
DLSw: START-FSM (488636): event:WAN-XID state:CKT_ESTABLISHED
DLSw: core: dlsw_action_g()
DISP Sent : CLSI Msg : ID.Rsp   dlen: 88
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED
```

```
DLsw Received-ctlQ : CLSI Msg : ID.Ind   dlen: 82
DLSw: START-FSM (488636): event:DLC-Id state:CKT_ESTABLISHED
DLSw: core: dlsw_action_f()
%DLSWC-3-SENDSSP: SSP OP = 7( XID ) to peer 10.17.2.198(2065) success
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CKT_ESTABLISHED
```

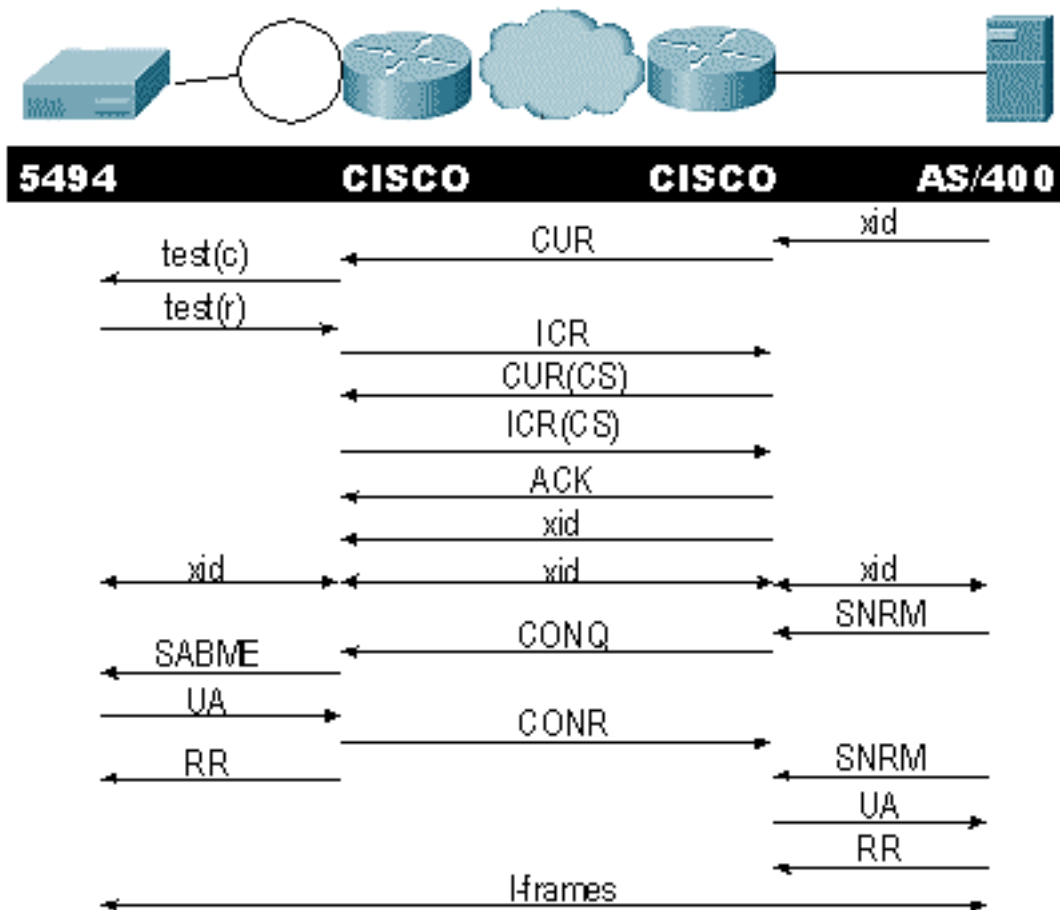
De router ontvangt **CONQ** van AS/400 (SABME). Dit wordt vertaald in de serielijn als een SNRM. Dan wacht de router op de UA op de seriële lijn (**CONNECT.Cfm**) en stuurt de **CONR** naar de andere kant. Hierdoor wordt de sessiestatus veranderd in **VERBONDEN**.

```
%DLSWC-3-RECVSSP: SSP OP = 8( CONQ ) from peer 10.17.2.198(2065)
DLSw: START-FSM (488636): event:WAN-CONQ state:CKT_ESTABLISHED
DLSw: core: dlsw_action_i()
DISP Sent : CLSI Msg : CONNECT.Req   dlen: 16
DLSw: END-FSM (488636): state:CKT_ESTABLISHED->CONTACT_PENDING
```

```
DLsw Received-ctlQ : CLSI Msg : CONNECT.Cfm CLS_OK dlen: 8
DLSw: START-FSM (488636): event:DLC-Connect.Cnf state:CONTACT_PENDING
DLSw: core: dlsw_action_j()
%DLSWC-3-SENDSSP: SSP OP = 9( CONR ) to peer 10.17.2.198(2065) success
DISP Sent : CLSI Msg : FLOW.Req   dlen: 0
DLSw: END-FSM (488636): state:CONTACT_PENDING->CONNECTED
```

[DLSw: uitvoeren van omgekeerde mediaconversie](#)

Een andere algemene instelling is **reverse-sdlc**. In omgekeerde SDLLC wordt het primaire station via een lijn SDLC aan de router bevestigd. Dit wordt meestal in host omgevingen gezien wanneer gebruikers de host willen migreren naar een Token Ring-bijlage. Omgekeerd SDLLC verandert de manier waarop DLSw de SDLC-lijn verwerkt omdat het vaak niet duidelijk is of de afstandsbediening actief is of niet.



Ten eerste, omdat de AS/400 in dit geval primair is, of in de rol onderhandelbaar is, moet zij de sessie starten. Wanneer de AS/400 de eerste XID verstuurt nadat de seriëlij gebruiksklaar is, start de router het zoekproces naar de afstandsbediening. Nadat het circuit is geactiveerd, kan de XID-onderhandeling in de regel worden gestart.

Wanneer de XID-onderhandeling is voltooid, verstuurt de AS/400 SNRM de router. Dit veroorzaakt de router om CONQ te verzenden en verwacht de CONR van de verre router. De router kan niet met de UA reageren tot het een SNRM ziet, en nadat het de CONR ontvangt. In bijna alle versies van de code, wacht de router 30 seconden tot de sessie is uitgelopen. Dit betreft het ontvangen van SNRMs van het primaire apparaat wanneer het primaire apparaat de CONR van de afstandsbediening ontvangt.

In de meest recente code van Cisco IOS 11.1, veranderen de standaardwaarden in één minuut in plaats van 30 seconden. In de AS/400 wordt deze tijd de **niet-productieve respontimer** genoemd en is de standaardinstelling op 32 seconden.

Lokale DLSw-mediaomzetting



```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2, changed state to up
%SYS-5-CONFIG_I: Configured from console by console
DLSW Received-ctlQ : CLSI Msg : ID_STN.Ind  dlen: 46
CSM: Received CLSI Msg : ID_STN.Ind  dlen: 46 from Serial2
```

Het eerste wat u in DLSw lokaal opmerkt is de XID van de seriële kant. Deze XID moet worden opgeslagen totdat de router de LLC-testframes/responsen doorstuurt.

```
CSM: smac 4000.5494.00dd, dmac 4000.9404.0001, ssap 4 , dsap 4
DISP Sent : CLSI Msg : TEST_STN.Req  dlen: 46
DISP Sent : CLSI Msg : TEST_STN.Req  dlen: 46
DISP Sent : CLSI Msg : TEST_STN.Req  dlen: 46
CSM: Write to all peers not ok - PEER_NO_CONNECTIONS
DLSW Received-ctlQ : CLSI Msg : TEST_STN.Ind  dlen: 43
CSM: Received CLSI Msg : TEST_STN.Ind  dlen: 43 from TokenRing0
CSM: smac c000.9404.0001, dmac 4000.5494.00dd, ssap 0 , dsap 4
```

Daarna verlaat het teststation de router en de reactiewinst van AS/400. Nu kan de router de lokale FSM creëren.

Opmerking: Denk eraan dat dit een lokale sessie is.

```
DLSw: csm_to_local(): Serial2-->TokenRing0 4000.5494.00dd:4->4000.9404.0001:4
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:ADMIN-START
DLSw: LFSM-A: Opening DLC station
DISP Sent : CLSI Msg : REQ_OPNSTN.Req  dlen: 106
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:DISCONNECTED ->OPN_STN_PEND
```

```
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:ADMIN-START
DLSw: LFSM-A: Opening DLC station
DISP Sent : CLSI Msg : REQ_OPNSTN.Req  dlen: 106
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:DISCONNECTED ->OPN_STN_PEND
```

```
DLSW Received-ctlQ : CLSI Msg : REQ_OPNSTN.Cfm CLS_OK dlen: 106
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-ReqOpnStn.Cnf
DLSw: LFSM-B: DLC station opened
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:OPN_STN_PEND ->ESTABLISHED
```

```
DLSW Received-ctlQ : CLSI Msg : REQ_OPNSTN.Cfm CLS_OK dlen: 106
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-ReqOpnStn.Cnf
DLSw: LFSM-B: DLC station opened
DLSw: processing saved clsi message
```

Nadat de router lokaal heeft bevestigd dat de FSM klaar is, kan het XID naar de partner sturen. In dit voorbeeld is de partner de AS/400 (**ID.Req**).

```
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Req  dlen: 12
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED
```

```
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:OPN_STN_PEND ->ESTABLISHED
```

```
DLSW Received-ctlQ : CLSI Msg : ID.Cfm CLS_OK dlen: 32
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp  dlen: 12
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED
```

Vervolgens wordt er een XID ontvangen van Token Ring. De **ID.Ind** heeft een lengte van 108. De

router zendt deze XID naar de partner in dit scenario, dat de lijn SDLC is. Dit wordt aangegeven door de **ID.Reg** die is verstuurd. Elke keer dat de router een pakje ontvangt, moet u de lineaire eindige staatsmachine (LFSM) starten. Dit is de sleutel tot het begrijpen van dit debug, omdat het je informeert waar het begint en welke punten het gaat.

```
DLSW Received-ctlQ : CLSI Msg : ID.Ind   dlen: 108
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Reg   dlen: 88
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED
```

Daarna wordt de XID-respons ontvangen van de serielijn en verzonden naar de partner (het Token Ring-station in dit voorbeeld). Dit duurt voort totdat de XID-uitwisseling voor dit PU2.1-apparaat is voltooid.

```
DLSW Received-ctlQ : CLSI Msg : ID.Ind   dlen: 82
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp   dlen: 80
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED
```

```
DLSW Received-ctlQ : CLSI Msg : ID.Ind   dlen: 108
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp   dlen: 88
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED
```

```
DLSW Received-ctlQ : CLSI Msg : ID.Ind   dlen: 82
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp   dlen: 80
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED
```

```
DLSW Received-ctlQ : CLSI Msg : ID.Ind   dlen: 108
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp   dlen: 88
DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->ESTABLISHED
```

```
%LINK-3-UPDOWN: Interface Serial2, changed state to up
DLSW Received-ctlQ : CLSI Msg : ID.Ind   dlen: 82
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Id
DLSw: LFSM-X: forward XID to partner
DISP Sent : CLSI Msg : ID.Rsp   dlen: 80
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->ESTABLISHED
```

Na de XID-uitwisseling ontvangt de router een SABME van de AS/400 via **CONNECT.Ind**. Dit vertelt de router om een **CONNECT.Reg** naar de lijn SDLC te sturen, wat SNRM is. Vervolgens wordt een bericht van **CONNECT.Cfm** (UA) ontvangen van de serielijn, waardoor de DLSw-code een **CONNECT.Rsp** (UA) naar de AS/400 veroorzaakt.

```
DLSW Received-ctlQ : CLSI Msg : CONNECT.Ind   dlen: 8
DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-Connect.Ind
DLSw: LFSM-C: starting local partner
DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:ADMIN-CONN
DLSw: LFSM-D: sending connect request to station
DISP Sent : CLSI Msg : CONNECT.Reg   dlen: 16
DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->CONN_OUT_PEND
```

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->CONN_IN_PEND

DLSW Received-ctlQ : CLSI Msg : **CONNECT.Cfm** CLS_OK dlen: 8

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Connect.Cnf

DLSw: LFSM-E: station accepted the connection

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:ADMIN-CONN

DLSw: LFSM-F: accept incoming connection

DISP Sent : CLSI Msg : **CONNECT.Rsp** dlen: 20

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:CONN_IN_PEND ->CONNECTED

DISP Sent : CLSI Msg : FLOW.Req dlen: 0

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CONN_OUT_PEND->CONNECTED

De sessie wanneer de controller (SDLC) wordt uitgeschakeld wordt weergegeven.

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2, changed state to down

%LINK-5-CHANGED: Interface Serial2, changed state to administratively down

DLSW Received-ctlQ : CLSI Msg : **DISCONNECT.Ind** dlen: 8

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Disc.Ind

DLSw: LFSM-Q: acknowledge disconnect

DISP Sent : CLSI Msg : **DISCONNECT.Rsp** dlen: 4

Daarna, verstuurt de router een DISK naar AS/400 (DISCONNECT.RSP). Dan begint het het lokale circuit af te breken.

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:ADMIN-STOP

DLSw: LFSM-Z: close dlc station request

DISP Sent : CLSI Msg : CLOSE_STN.Req dlen: 4

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:ESTABLISHED ->CLOSE_PEND

DISP Sent : CLSI Msg : CLOSE_STN.Req dlen: 4

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:ESTABLISHED ->CLOSE_PEND

DLSW Received-ctlQ : CLSI Msg : CLOSE_STN.Cfm CLS_OK dlen: 8

DLSw: START-LFSM TokenRing0 (4000.9404.0001->4000.5494.00dd) event:DLC-CloseStn.Cnf

DLSw: LFSM-Y: driving partner to close circuit

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:ADMIN-STOP

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CLOSE_PEND ->CLOSE_PEND

DLSw: END-LFSM (4000.9404.0001->4000.5494.00dd): state:CLOSE_PEND ->DISCONNECTED

DLSW Received-ctlQ : CLSI Msg : **DISCONNECT.Ind** dlen: 8

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-Disc.Ind

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CLOSE_PEND ->CLOSE_PEND

DLSW Received-ctlQ : CLSI Msg : CLOSE_STN.Cfm CLS_OK dlen: 8

DLSw: START-LFSM Serial2 (4000.5494.00dd->4000.9404.0001) event:DLC-CloseStn.Cnf

DLSw: LFSM-Y: removing local switch entity

DLSw: END-LFSM (4000.5494.00dd->4000.9404.0001): state:CLOSE_PEND ->DISCONNECTED

Nadat de router de **DISCONNECT.Ind** (UA) van AS/400 ontvangt, duurt het het verwijderen van de sessie en beweegt u naar een ontkoppelingsstaat.

[Gerelateerde informatie](#)

- [IBM-technologieën](#)
- [Technische ondersteuning en documentatie – Cisco Systems](#)