

Downlink Data Notification

- Feature Summary and Revision History, on page 1
- Feature Description, on page 2
- DDN Message Handling, on page 2
- Control Messages Triggered DDN Support, on page 10
- DDN Advance Features, on page 12

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	cnSGW-C
Applicable Platform(s)	SMI
Feature Default Setting	DDN Message Handling Support: Enabled - Always-on Control Messages Triggered DDN Support: Disabled - Configuration required to enable DDN Advance Features: Enabled - Always-on
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
Enhancement introduced. Added support for DDN Advance Features.	2021.02.0

Revision Details	Release
First introduced.	2021.01.0

Feature Description

The following sub-features are associated with this feature:

- DDN Message Handling
- Control Messages Triggered DDN
- Downlink Data Notification Delay
- High Priority Downlink Data Notification
- DDN Throttling

DDN Message Handling

Feature Description

cnSGW-C supports handling of the Downlink Data Notification (DDN) functionality that includes:

- Generating a DDN message towards the MME to page the UE on arrival of downlink data when UE is in IDLE state.
- Handling DDN ACK/DDN failure indication.

How it Works

This section describes how this feature works.

Call Flows

This section describes the key call flows for this feature.

Downlink Data Notification Success Call Flow

This section describes the Downlink Data Notification Success call flow.

Figure 1: Downlink Data Notification Success Procedure Call Flow

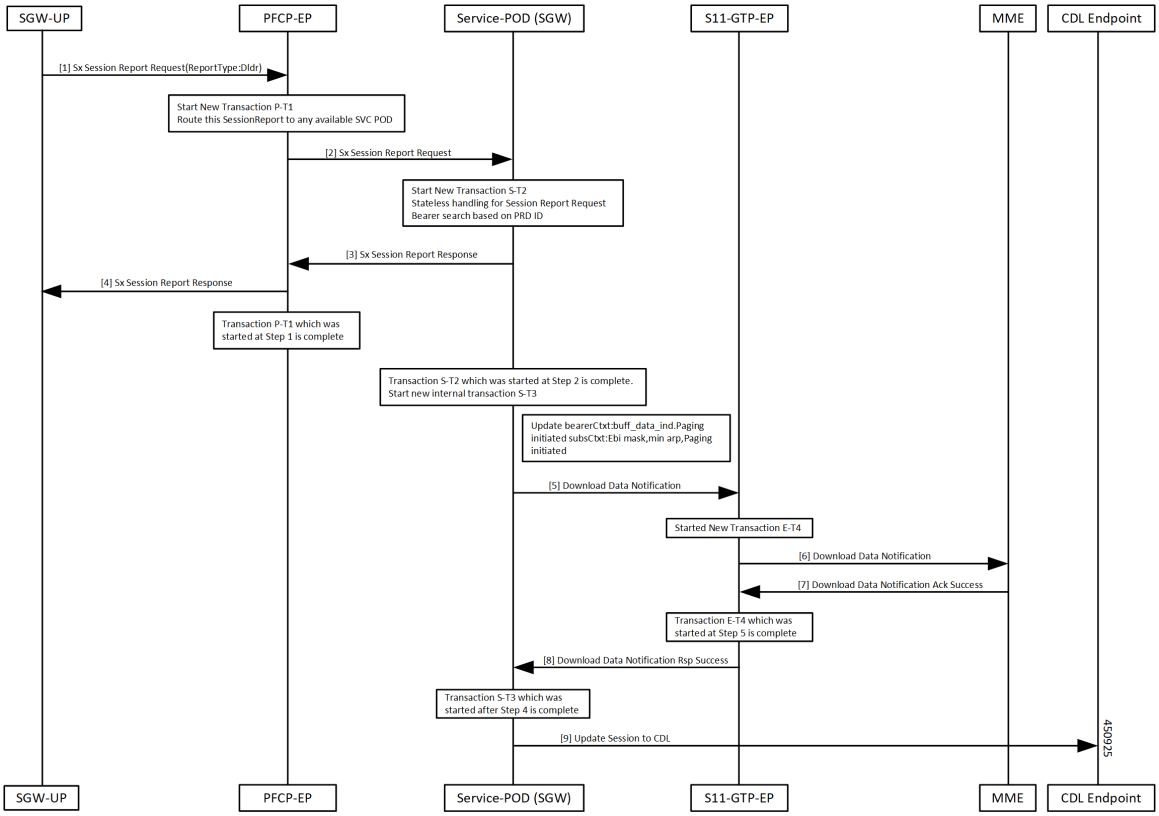


Table 3: Downlink Data Notification Success Procedure Call Flow Description

Step	Description
1	SGW-UP sends the Session Report Request to the PFCP-EP pod.
2	After receiving the Session Report Request, the PFCP-EP performs the following: • Starts a new P-T1 transaction. • Checks for the interface type. If its Sxa interface, it finds the available SGW-service pod and routes the request accordingly. • Sends the Sx Session Report Request to the SGW-service pod.

Step	Description
3	Upon reception of the Sx Session Report Request, SGW-service pod: • Creates a new S-T2 transaction. • Based on the message type received, updates the state processing not required for this message. • Handles the non-state processing transaction. (High priority is given to handle such messages). • Searches for the bearer based on PDR ID. If the bearer isn't found, the SGW-service pod fills the cause as **request rejected** in the Session Report Response. • If the received report type in the request isn't valid/supported, SGW-service pod fills the cause as **request rejected** and sends the Sx Session Report Response.
4	PFCP-EP forwards the Sx Session Report Response to the SGW-UP.
5	P-T1 transaction which is started at step one is completed. At SGW-service pod: • S-T2 transaction which is started at step two is completed. • If the Sx Session Report Response is success, a new internal transaction S-T3 is started with the same buffer as of the Session Report Request. • A DDN procedure for DLDR report type is initiated. • Bearer information is extracted from the received PDR ID. • Bearer context is updated with buffer-data_ind. • Initiated the DDN with EBI of bearers, which has downlink data, and minimum ARP among these bearers. • Sends the DDN to the S11-GTP-EP pod.
6	After receiving the DDN, S11-GTP-EP: • Creates a new E-T4 transaction. • Sends the DDN to the MME.
7	MME sends the DDN ACK Success to the S11-GTP-EP.
8	The transaction S-T3 which is started after step four is complete. S11-GTP-EP sends the DDN Response success to SGW-service pod.
9	SGW-service pod updates the CDL.

Downlink Data Notification Failure Call Flow

This section describes the Downlink Data Notification Failure call flow.

Figure 2: Downlink Data Notification Failure Call Flow

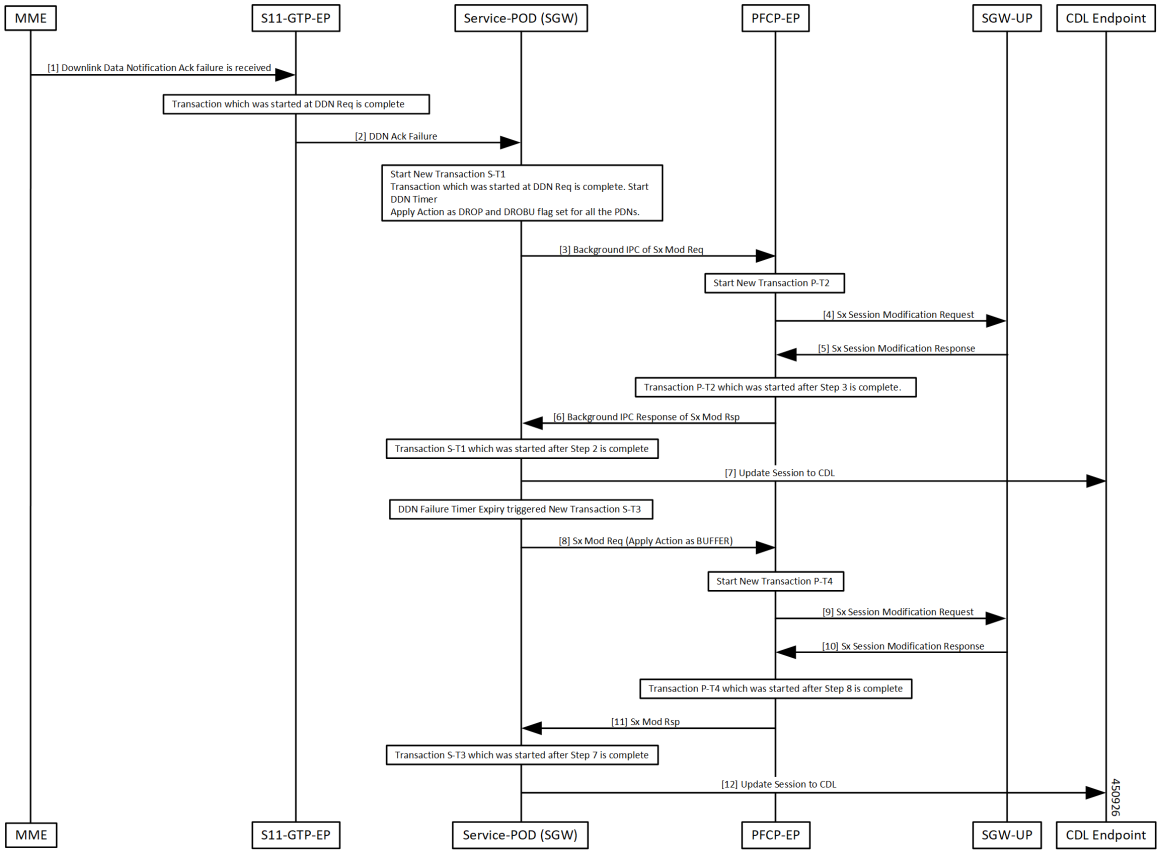


Table 4: Downlink Data Notification Failure Procedure Call Flow Description

Step	Description
1	S11-GTP-EP pod receives DDN ACK Failure.
2	The transaction started while sending the DDN Request ends. S11-GTP-EP forwards the DDN ACK Failure to the SGW-service pod.

Step	Description
3	After receiving the DDN ACK Failure at the SGW-service pod: • Decides the paging state based on the cause received: • EGTP_CAUSE_CONTEXT_NOT_FOUND: Submit internal transaction for call deletion. • EGTP_CAUSE_UNABLE_TO_PAGE_UE • EGTP_CAUSE_UNABLE_TO_PAGE_UE_DUE_TO_SUSPENSION • EGTP_CAUSE_UE_ALREADY_REATTACHED • EGTP_CAUSE_TEMP_REJECTED_DUE_TO_HANDOVER_IN_PROGRESS • Checks if the PDNs are in connected state to initiate the Sx Modify Request. Minimum one one PDN should be in the CONNECTED state. • Submits internal transactions to handle these paging failure causes. • Ends the current procedure and transaction. • In the new transaction of handling paging failures, derives all the PDNs for which you want to send Sx Modify request. • Based on the paging state, derives paging action and send Sx Modify Request based on the action required. Sends background IPC request for Sx Modification Request to PFCP-EP pod. Create a new transaction P-T2.
4	After receiving background IPC request for Sx Modification request, PFCP-EP: • Starts a new P-12 transaction. • Sends the o the SGW-UP.
5	PFCP-EP receives the Sx Modification Response from the SGW-UP.
6	The transaction P-T2 started at step three is complete. PFCP-EP pod sends background IPC response to the SGW-service pod.
7	The transaction S-T1 started at step two is complete. SGW-service pod updated the CDL with buff_data_ind at bearer level flag.
8	On DDN Failure timer expiry, a new transaction S-T3 is started. SGW-service pod sends background IPC request for the Sx Modification Request to the PFCP-EP pod with Apply Action as BUFFER.
9	A new P-T4 transaction is created. PFCP-EP pod sends the Sx Modification Request to the SGW-UP.
10	SGW-UP sends the Sx Modification Response to the PFCP-EP pod.

Step	Description
11	The transaction P-T4 started at step eight is complete. PFCP-EP pod forwards the Sx Modification Response to the SGW-service pod.
12	The transaction S-T3 started at step seven is complete. SGW-service pod updates the CDL.

No User Connect Retry Timer Call Flow

This section describes the No User Connect Retry Timer call flow.

Figure 3: No User Connect Retry Timer Call Flow

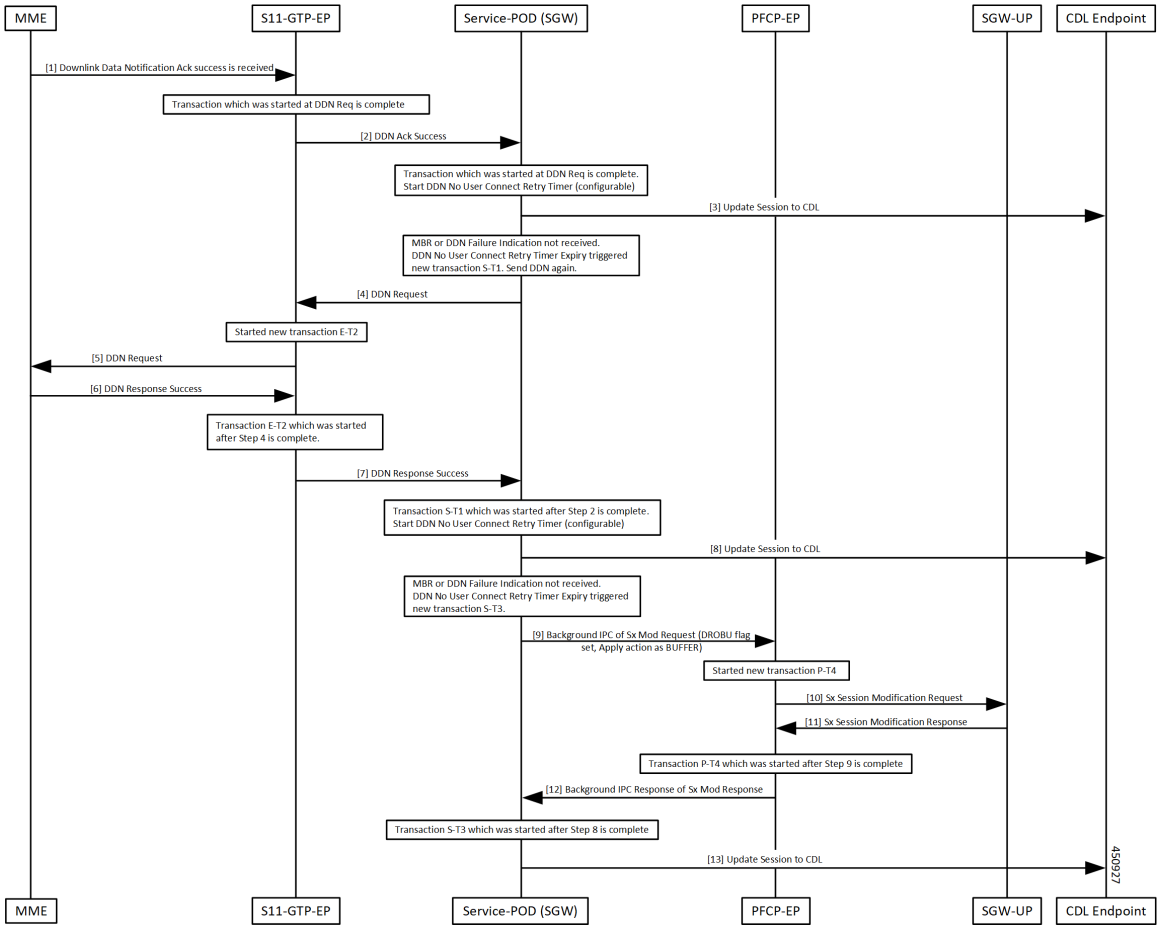


Table 5: No User Connect Retry Timer Call Flow Description

Step	Description
1	Received DDN ACK success at the S11-GTP-EP pod.

Step	Description
2	The transaction started while sending the DDN ends here. S11-GTP-EP sends the DDN ACK success to the SGW-service pod.
3	No User Connect Retry timer is started at the SGW-C pod. This timer is configurable. SGW-service pod updates the CDL.
4	SGW-service pod sends the DDN Request to the S11-GTP-EP pod, when: • The DDN Failure Indication/MBR is not received • No User Connect Retry timer expires. A new transaction S-T1 is created.
5	A new E-T2 transaction is created. S11-GTP-EP pod forwards the DDN Request to MME.
6	MME sends the DDN Response to the S11-GTP-EP.
7	The transaction E-T2 started at step four is complete. S11-GTP-EP forwards the DDN Response Success to the SGW-service pod.
8	S-T1 transaction started at step two is completed. No User Connect Retry timer is started at the SGW-C pod. This timer is configurable. SGW-service pod updates the CDL.
9	If DDN Failure Indication/MBR is not received, No User Connect Retry expiry triggered. A new transaction S-T3 is created. SGW-service pod sends the background IPC request for Sx Modification request to the PFCP-EP pod (DROBU flag and Apply Action as BUFFER).
10	A new transaction P-T4 is created. PFCP-EP pod sends the Sx Modification Request to the SGW-U pod.
11	PFCP-PE pod receives the Sx Modification Response.
12	The transaction P-T4 started at step nine is complete. PFCP-EP pod sends the background IPC response to the SGW-service pod.
13	The transaction S-T3 started at step eight is complete. CDL is updated.

Feature Configuration

Configuring this feature involves the following steps:

Configuring the DDN Failure Timer

DDN Failure Timer is configured under the `sgw-profile`.

To configure this feature, use the following configuration:

```
config
   profile sgw sgw_name
      ddn failure-action-drop-timer timer_value
      ddn timeout-purge-session { true | false }
      end
```

NOTES:

- **ddn failure-action-drop-timer** *timer_value*—Specify the duration of the DDN packet drop timer. During this specified timeframe, the DDN is not sent to the UE. This timer is used, when a notification of DDN ACK Failure or DDN Failure Indication is received. The default value is 300 seconds.

Note To disable the timer, set the timer value to zero.

- **ddn timeout-purge-session { true | false }**—Specify the option to enable or disable the DDN timeout purge session. The default value is false.

Configuration Example

The following is an example configuration.

```
config
profile sgw sgw1
ddn failure-action-drop-timer 60
ddn timeout-purge-session false
end
```

Configuration Verification

To verify the configuration:

```
show running-config profile sgw
profile sgw sgw1
locality LOC1
fqdn 209.165.201.1
ddn failure-action-drop-timer 60
ddn timeout-purge-session false
end
```

Configuring DDN No User Connect Retry Timer

This section describes how to configure the DDN No User Connect Retry Timer.

DDN No User Connect Retry Timer can be configured under sgw-profile.

To configure this feature, use the following configuration:

```
config
  profile sgw sgw_name
   ddn no-user-connect-retry-timer timer_value
   end
```

NOTES:

- **ddn no-user-connect-retry-timer** *timer_value* - Specify the DDN retry timer used when DDN Ack is received with Success and MBR is not received. Default value is 60 seconds.

  To disable the timer, set the value to 0.

Configuration Example

The following is the sample configuration.

```
config
 profile sgw sgw1
  ddn no-user-connect-retry-timer 120
  end
```

Configuration Verification

To verify the configuration:

```
show running-config profile sgw
profile sgw sgw1
locality LOC1
fqdn cisco.com.apn.epc.mnc456.mcc123
ddn failure-action-drop-timer 60
ddn no-user-connect-retry-timer 120
```

Control Messages Triggered DDN Support

Feature Description

This feature supports paging the UE for the PGW-initiated control procedures when the UE is in IDLE mode.

Note This feature is CLI controlled.

How it Works

This section describes how this feature works.

Call Flows

This section describes the key call flows for this feature.

Downlink Data Notification for PGW-initiated procedure with Cloud Native Call Flow

This section describes the DDN for the PGW-initiated procedure with Cloud Native call flow.

Figure 4: Downlink Data Notification for PGW initiated Procedure with Cloud Native Call Flow

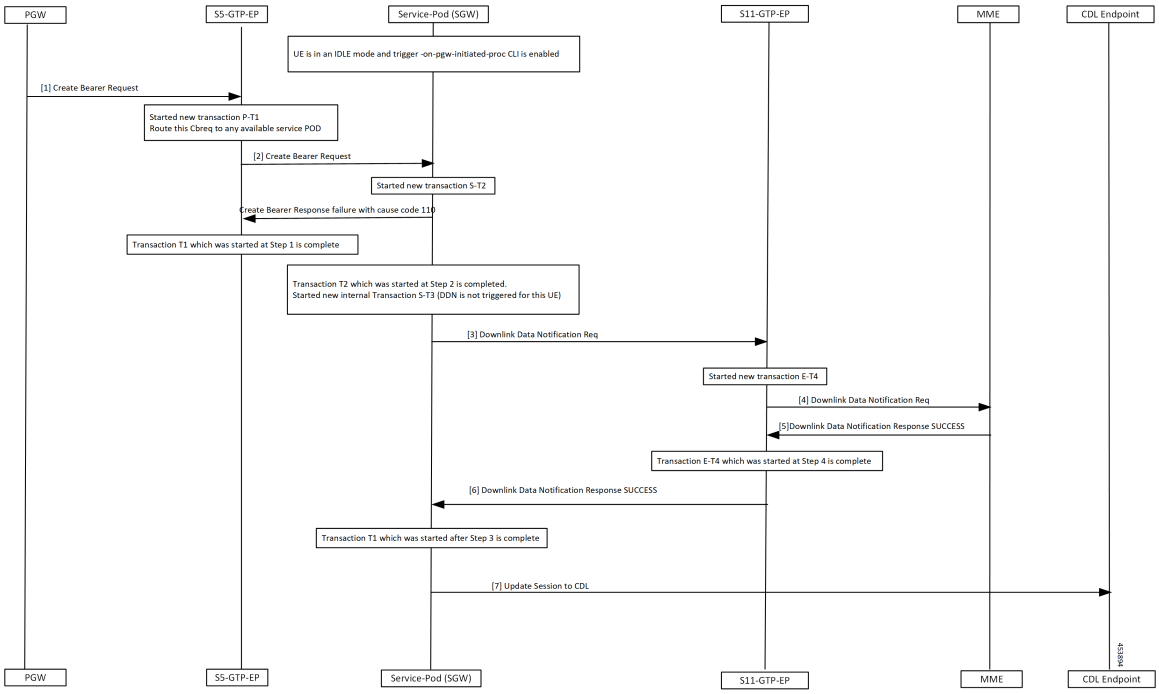


Table 6: Downlink Data Notification for PGW initiated procedure (CBR) with Cloud Native Call Flow Description

Step	Description
1, 2	Enabled trigger-on-pgw-initiated-proc CLI and state of the UE is in IDLE mode. S5-GTP-EP receives the CBR from the PGW and forwards it to the SGW-service pod. SGW-service pod starts a new S-T2 transaction. SGW-service pod sends failure response to the S5-GTP-EP with cause code 110.
3	The T2 transaction which started in step two is completed. A new S-T3 transaction is started for the UE for which DDN is not triggered. SGW-service pod initiates the DDN Request to the theS11-GTP-EP.
4	A new E-T4 transaction is started. S11-GTP-EP forwards the DDN Request to the MME.
5	S11-GTP-EP receives the DDN Response success from the MME.
6	Transaction E-T4 which started in step four is completed. S11-GTP-EP sends the DDN Response success to the SGW-service pod.
7	Transaction T1 which is started in step three is completed. SGW-service pod updates the session to CDL.

Feature Configuration

To configure this feature, use the following configuration:

```
config
 profile sgw sgw_name
  ddn trigger-on-pgw-initiated-proc
  end
```

NOTES:

- **ddn trigger-on-pgw-initiated-proc**—When UE is in IDLE mode, the DDN triggers paging for PGW-initiated procedures. SGW sends failure response to the PGW with cause code 110.

Configuration Example

The following is an example configuration.

```
config
 profile sgw sgw1
  ddn trigger-on-pgw-initiated-proc
  end
```

Configuration Verification

To verify the configuration:

```
show running-config profile sgw
profile sgw sgw1
locality LOC1
fqdn 209.165.201.1
ddn failure-action-drop-timer 60
ddn no-user-connect-retry-timer 120
ddn trigger-on-pgw-initiated-proc
exit
```

Disabling the DDN Control Procedure

Use `no ddn trigger-on-pgw-initiated-proc` to disable DDN Control Procedure feature.

DDN Advance Features

Feature Description

This feature supports the following:

- Downlink Data Notification Delay
- High Priority Downlink Data Notification
- DDN Throttling

How it Works

This section describes how this feature works.

Call Flows

This section describes the key call flows for this feature.

DDN Delay Call Flow

This section describes DDN Delay call flow.

Figure 5: DDN Delay Call Flow

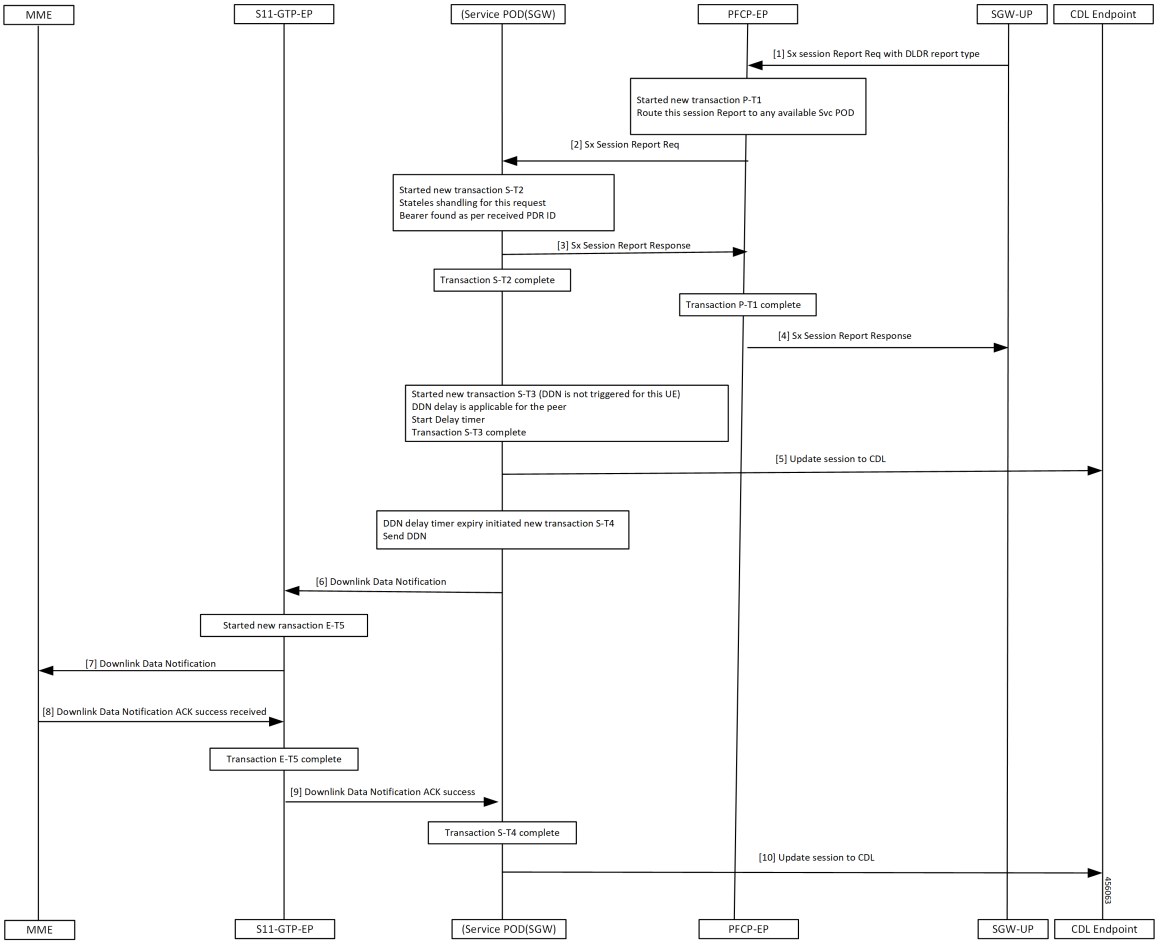


Table 7: DDN Delay Call Flow Description

Step	Description
1	Received downlink data when UE is in IDLE state. SGW-UP sends the Sx Report Request with report type as DLDR with corresponding PDR ID to the PFCP-EP.

Step	Description
2	Started a new P-T1 transaction. PFCP-EP pod: • Checks the available service pod. • Sends the Sx Session Report to the the SGW-service pod.
3	A new transaction S-T2 is stared. SGW-CP sends success response to the SGW-UP, when a bearer found at CP for this PDR-ID.
4	The S-T2, P-T1 transactions are completed. PFCP-EP sends the Sx Session Report Response to the SGW-UP.
5	A new transaction S-T3 is started when DDN is not triggered for this UE. Sgw-service pod gets the peer information to check if the peer configured with the DDN delay value. DDN delay timer is triggered, if DDN delay configured. S-T3 transaction is completed. SGW-service pod sends the CDL update.
6, 7	A new S-T4 transaction started. SGW-service pod sends the DDN to the S11-GTP-EP. A new E-T5 transaction is started. S11-GTP-EP forwards the DDN to the MME.
8, 9	MME sends the DDN ACK success to the S11-GTP-EP. Transaction E-T5 started in step seven is completed. S11-GTP-EP forwards the DDN ACK success towards the SGW-service pod.
10	Transaction S-T4 started in step six is completed. SGW-service pod updates session information to CDL.

High Priority DDN Call Flow

This section describes High Priority DDN call flow.

Figure 6: High Priority DDN Call Flow

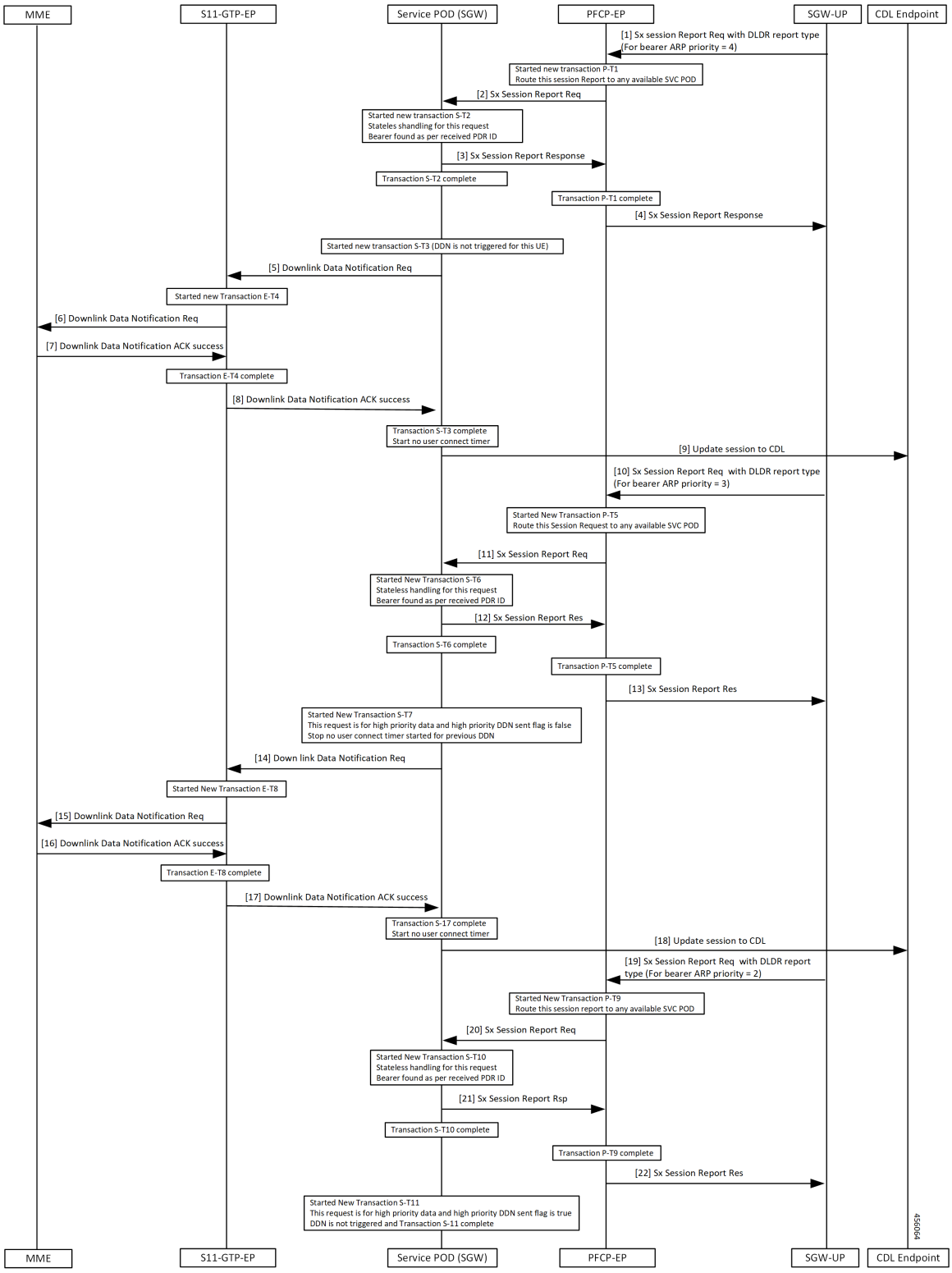


Table 8: High Priority DDN Call Flow Description

Step	Description
1	Bearer received downlink data with ARP priority value as four, when UE is in IDLE state. SGW-UP sends the Sx Report Request to the PFCP-EP with report type as DLDR with corresponding PDR ID.
2	New P-T1 transaction is started and routed the session report to all available service pods. PFCP-EP sends the Sx Session Report Request to the SGW-service pod.
3	New S-T2 transaction is started and the SGW-service pod sends Sx Session Report Response to the PFCP-EP.
4	Transaction P-T1 and S-T2 completed and the PFCP-EP forwards the Sx Session Report Response to the SGW-UP.
5	New S-T3 transaction is started for which the DDN isn't triggered. SGW-service pod sends the DDN Request to the S11-GTP-EP.
6	New E-T4 transaction is started and the S11-GTP-EP forwards the DDN Request to the MME.
7	MME sends the DDN ACK success to the S11-GTP-EP.
8	Transaction E-T4 is completed. S11-GTP-EP forwards the DDN ACK success to the SGW-service pod.
9	Transaction S-T3 completed. SGW-service pod triggers No User Connect timer and updates session to CDL.
10	SGW-UP sends the Sx Session Report Request to the PFCP-EP with report type as DLDR for bearer whose ARP priority value is three.
11	New P-T5 transaction is started and routed the session report to all the available service pods. PFCP-EP sends the Sx Session Report Request to the SGW-service pod.
12	New transaction S-T6 started SGW-service pod sends the Sx Session Report Response to the PFCP-EP when bearer found as per the received PDR ID.
13	Transaction S-T6 and P-T5 completed and PFCP-EP forwards the Sx Session Report Response to the SGW-UP.
14	New transaction S-T7 started and data, high priority DDN sent with the flag value as False. No User Connect timer is topped. SGW-service pod sends the DDN Request to the S11-GTP-EP.
15	New E-T8 transaction is started and the S11-GTP-EP forwards the DDN Request to the MME.
16	MME sends the DDN ACK success to the S11-GTP-EP.

Step	Description
17	S11-GTP-EP forwards the DDN ACK success to the SGW-service pod for this PDR ID.
18	Transaction S-17 completed. SGW-service pod triggers the No User Connect timer when received DDN ACK success and updated the session to CDL.
19	Bearer received the downlink data with ARP priority value as two. SGW-UP sends the Sx Report Request to the PFCP-EP with report type as DLDR with corresponding PDR ID.
20	New transaction P-T9 started and routed the session report to all the available service pods. PFCP-EP sends the Sx Session Report Request to the SGW-service pod.
21	New transaction S-T10 started and the SGW-service pod sends the Sx Session Report Response to the PFCP-EP when the bearer found as per the received PDR ID.
22	Transaction S-T10 and P-T9 completed and the PFCP-EP forwards the Sx Session Report Response to the SGW-UP. At SGW-service pod: • New transaction S-T11 started and data, high priority DDN sent with the flag value as True. SGW-service pod stops No User Connect timer. • SGW-service pod doesn't trigger DDN when high priority DDN already initiated. Transaction S-11 is completed.

DDN Throttling Call Flow

This section describes DDN Throttling call flow.

Figure 7: DDN Throttling Call Flow

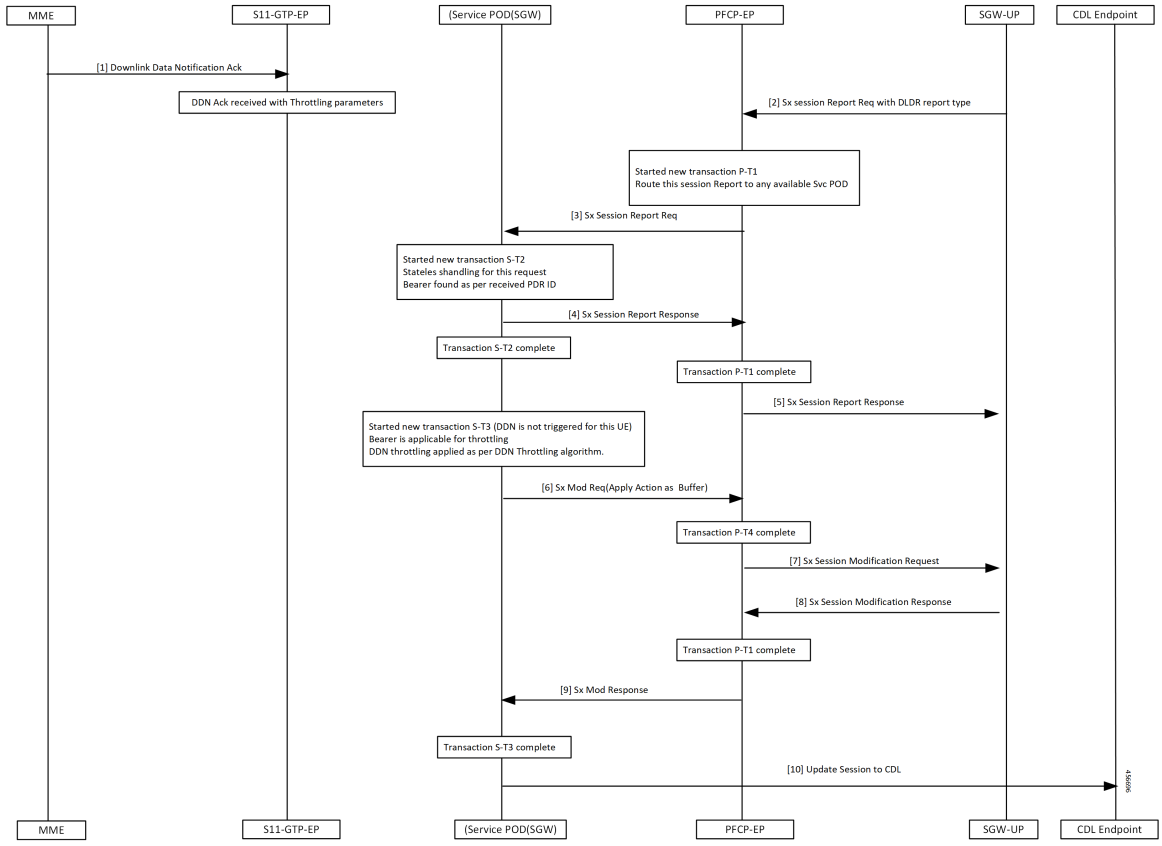


Table 9: DDN Throttling Call Flow Description

Step	Description
1	Received DDN with throttling parameters when UE is in IDLE state. MME sends the DDN ACK to the S11-GTP-EP.
2	SGW-UP sends the Sx Report Request with report type as DLDR with corresponding PDR ID to PFCP-EP.
3	PFCP-EP triggers a new P-T1 transaction and routes the Sx Report Request to available service pod. PFCP-EP sends the Sx Session Report Request to the SGW-service pod.
4	Started a new S-T2 transaction. Get peer information to check if DDN Throttle is active for this peer. Check if priority of this bearer is more than the configured ARP watermark SGW-service pod sends the Sx session Report Response to the PFCP-EP.

Step	Description
5	S-T2 transaction started in step four is completed. P-T1 transaction started in step threee is completed. When a bearer found at CP for this PDR ID, the PFCP-EP sends success response to the SGW-UP.
6	A new S-T3 transaction is started for the UE for which DDN is not triggered. Apply DDN algorithm to check if the DDN must be throttled. If DDN throttled, SGW-service pod sends the Sx Modification Request with Apply Action as BUFFER towards PFCP-EP.
7, 8	P-T4 transaction is completed. PFCP-EP sends the Sx Session Modification Request to the SGW-UP and receives the Sx Session Modification Response from the SGW-UP.
9	P-T1 transaction started in step five is completed. PFCP-EP sends Sx Modification Response to the SGW-service pod.
10	S-T3 transaction started in step six is completed. SGW-service pod updates the session to CDL.

Note cnSGW does not throttle outgoing downlink data notification messages for emergency or WPS sessions.

Standards Compliance

The Downlink Data Notification Support feature complies with the following standards:

- *3GPP TS 23.401, "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access"*
- *3GPP TS 23.402, "Architecture enhancements for non-3GPP accesses"*
- *3GPP TS 29.274, "3GPP Evolved Packet System (EPS); Evolved General Packet Radio Service (GPRS) Tunnelling Protocol for Control plane (GTPv2-C); Stage 3"*
- *3GPP TS 23.214, "Architecture enhancements for control and user plane separation of EPC nodes"*
- *3GPP TS 29.244, "Interface between the Control Plane and the User Plane nodes"*
- *3GPP TS 24.008, "Mobile radio interface Layer 3 specification; Core network protocols; Stage 3"*

Feature Configuration

By default, the DDN throttling is always enabled.

Note cnSGW-C handles DDN throttling parameters sent from the MME.

To configure this feature, use the following configuration:

```
config
 profile sgw sgw_name
  ddn throttle-arp-watermark arp_value
  end
```

NOTES:

- **ddn throttle-arp-watermark***arp_ value*—Specify the lowest priority ARP for DDN throttle.

  Throttling is applicable only for bearer having ARP PL value greater than the configured *value*. Must be an integer in the range of 0-15.

  By default, throttling is applicable for all bearers.

Configuration Example

The following is an example configuration.

```
config
 profile sgw sgw1
  ddn throttle-arp-watermark 3
  end
```

OAM Support

This section describes operations, administration, and maintenance information for this feature.

Bulk Statistics

The following statistics are supported for the DDN Advance feature.

```
sgw_ddn_stats{app_name="smf",cluster="cn",data_center="cn",
ddn_stats_type="control_proc_triggered",instance_id="0",service_name="sgw-service"}2
```

```
sgw_ddn_stats{app_name="smf",cluster="cn",data_center="cn",ddn_stats_type="data_triggered",
instance_id="0",service_name="sgw-service"} 18
```

```
sgw_ddn_stats{app_name="smf",cluster="cn",data_center="cn",ddn_stats_type="delayed",
instance_id="0",service_name="sgw-service"} 7
```

```
sgw_ddn_stats{app_name="smf",cluster="cn",data_center="cn",ddn_stats_type="high_priority_initiated",
instance_id="0",service_name="sgw-service"} 3
```

```
sgw_ddn_stats{app_name="smf",cluster="cn",data_center="cn",ddn_stats_type="high_priority_suppressed",
instance_id="0",service_name="sgw-service"} 1
```

```
sgw_ddn_stats{app_name="smf",cluster="cn",data_center="cn",ddn_stats_type="throttled",
instance_id="0",service_name="sgw-service"} 6
```

- high_priority_initiated - DDN initiated count, due to high priority paging trigger.
- high_priority_suppressed - DDN high priority count which is suppressed. When a UE is already working on the high priority DDN-initiated paging request. It suppresses the incoming high priority paging request.
- throttled - DDN throttled count.
- delayed - DDN initiated count after the DDN delay timer.
- control_proc_triggered - The received count of paging triggers from control procedure when UE is in IDLE state.
- data_triggered - The received count of paging triggers from UPF for downlink data when UE is in IDLE state.