

Release 3.1 Change Orders

Change Order #	Description	NPAC Effort	SOA/LSMS
NANC 179	TN Range Notifications	Medium	Medium High / N/A
NANC 240	No cancellation of SVs based on expiration of T2 timer	Low	Low / N/A
NANC 294	Changing Due Date Edit Functionality in the NPAC SMS for 7pm on Due Date Problems	Medium	Medium /N/A
NANC 328	Tunable for Long and Short Business Days	Low	N/A /N/A
NANC 329	Prioritization of SOA Notifications	Medium	N/A /N/A

Table of Contents

Change Order Number: NANC 179.....	3
Change Order Number: NANC 240.....	36
Change Order Number: NANC 294.....	45
Change Order Number: NANC 328.....	51
Change Order Number: NANC 329.....	53

Origination Date: 11/25/1997

Change Order Number: NANC 179

Description: TN Range Notifications

Cumulative SP Priority, Weighted Average: 14.42

Pure Backwards Compatible: NO

Functional Backwards Compatible: YES

IMPACT/CHANGE ASSESSMENT

FRS	IIS	GDMO	ASN.1	NPAC	SOA	LSMS
Y	Y	Y		Medium	Medium High	N/A

Business Need:

Currently SOA systems receive notifications on a TN by TN basis. This change order will allow them to receive a single notification from the NPAC for a TN range instead of individual notifications for each TN. This will reduce the number of messages received and time required by SPs to compare outputs to their internal service order processing systems.

Description of Change:

Currently notifications for TN range related operations come as individual notifications for each TN in the range. It has been suggested that the notifications for all TN's in a range be combined into one notification.

After further analysis, it was determined that this should be revised to include all appropriate status attribute value changes and attribute value changes, plus return to donor notifications.

Additional Write-Up:

Currently notifications for TN range related operations come as individual notifications for each TN in the range. It has been suggested that the notifications for all TNs in a range be combined into one notification.

This would include the following notifications:

- Object Creation (1 set of info for the TN range, plus a list of TN/SV-ID pairing)
- Attribute Value Change (1 set of info for the TN range, plus list of SV-ID)
- Status Attribute Value Change (1 set of info for the TN range, plus list of SV-ID)
- Return to Donor Notifications (1 set of info for the TN range, plus list of SV-ID)
- Subscription Version Cancellation Resolution Request
- Subscription Version New SP Create Request
- Subscription Version Old SP Concurrence Request
- Subscription Version Old SP Final Concurrence Timer Expiration

Any action that is performed in a TN range basis that results in above notifications will be sent in a range format.

Benefits of NANC 179:

- Large reduction in number of messages sent to the SOA and NPAC SMS.
- Reduces NPAC SMS database activity due to less logging and processing.
- Improved interface performance for both NPAC SMS and SOA.

The range notifications would not be emitted from a specific subscription version in the range, as these notifications are to date. They would have to be emitted from the container object for the subscription versions (InpSubscriptions).

Requirements:

The backwards compatibility sunset period for the TN Range Notification Indicator is two major NPAC SMS Releases (i.e., if implemented in R4, it is only guaranteed to be available through R5, and may be unavailable starting with R6).

Req 1 NPAC Customer TN Range Notification Indicator

NPAC SMS shall provide a mechanism to indicate whether a Service Provider supports receiving TN Range Notifications via the SOA to NPAC SMS Interface.

Req 2 NPAC Customer TN Range Notification Indicator – Default

NPAC SMS shall default the TN Range Notification Indicator to **FALSE**.

Req 3 NPAC Customer TN Range Notification Indicator – Modification

NPAC SMS shall allow NPAC Personnel, via the NPAC Administrative Interface, to modify the TN Range Notification Indicator on the NPAC Customer record.

Req 4 TN Range Notification Information – Service Provider TN Range Notification Indicator Sending of TN Range Notifications

NPAC SMS shall send TN Range Notifications, via the SOA to NPAC SMS Interface, if the Service Provider's TN Range Notification Indicator is **TRUE**.

Req 5 TN Range Notification Information – Service Provider TN Range Notification Indicator Suppression of TN Range Notifications

NPAC SMS shall suppress TN Range Notifications and send individual TN Notifications, via the SOA to NPAC SMS Interface, if the Service Provider's TN Range Notification Indicator is **FALSE**.

Req 6 TN Range Notification Information – Single TN Range Notifications

NPAC SMS shall send a single TN Range Notification when the same data applies to all TNs in the range.

Req 7 TN Range Notification Information – Breakup of TN Range Notifications

NPAC SMS shall send more than one TN Range Notification when the same feature data does NOT apply to all TNs in the range, by breaking up the TN Range and sending TN Range Notifications such that the same feature data applies to all TNs in the smaller broken up TN Ranges.

Req 8 TN Range Notification Information – Recovery of TN Range Notifications

NPAC SMS shall send TN Range Notifications during recovery that mimic the same TN Range Notifications that would have been received by the Service Provider had they been associated during the original broadcast of the TN Range Notifications.

Req 9 TN Range Notification Information – Single NPA-NXX

NPAC SMS shall only allow a TN Range Notification to be inclusive within a single NPA-NXX.

RR6-29 (Modified) Notification Recovery

NPAC SMS shall support recovery of all CMIP notifications defined in the IIS that are emitted over the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces.

Examples of notifications to be recovered include:

- subscriptionVersionNewNPA-NXX
- subscriptionVersionDonorSP-CustomerDisconnectDate
- subscriptionVersionAudit-DiscrepancyRpt
- -
- -
- -
- subscriptionVersionRangeStatusAttributeValueChange
- subscriptionVersionRangeStatusAttributeValueChange
- subscriptionVersionRangeStatusAttributeValueChange
- subscriptionVersionRangeAttributeValueChange
- subscriptionVersionRangeObjectCreation
- subscriptionVersionRangeDonorSP-CustomerDisconnectDate
- subscriptionVersionRangeCancellationAcknowledgeRequest
- subscriptionVersionRangeNewSP-CreateRequest
- subscriptionVersionRangeOldSP-ConcurrenceRequest
- subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration
- subscriptionVersionRangeNewSPFinalCreateWindowExpiration

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

For a complete list of notifications reference the IIS.

R4-8 (Modified) Service Provider Data Elements

NPAC SMS shall require the following data if there is no existing Service Provider data:

1. Service Provider name, address, phone number, and contact organization.
2. NPAC customer type.
3. Service Provider allowable functions.
 -
 -
 -

18. SOA Notification Priority for each SOA notification. Separate values may be set for Status Attribute Value Change notifications based on whether the Service Provider is acting as the Old Service Provider or as the New Service Provider for the port as indicated in Appendix C, Table C-7 – SOA Notification Priority Tunables.

(NOTE: This is the modification necessary from NANC 329.)

19. TN Range Notification Indicator

IIS:

The following text is an excerpt from the end of section 5.2.3 “*Data Origination Authentication*” on how access control is handled for the Object Creation, Object Deletion and Attribute Value Change Notifications:

ObjectCreation, ObjectDeletion, and AttributeValueChange should use the “information” attribute (*i.e.*, sub-index 6.1.7.3, 7.1.6.3, and 8.1.6.3 in section 9.21.5, *subscriptionVersionNPACNotifications*, Exhibit 83), which is an ANY DEFINED BY to contain the access control field. The values and authentication for the notification access control fields are the same as above.

This section would need to be modified to add the following text:

For range ObjectCreation and AttributeValueChange notifications the access control would not be placed in the information attribute but rather in the access control attribute defined. This would allow for the access control information to only be present once in the range notifications.

GDMO:

-- 14.0 LNP Subscriptions Managed Object Class

```
lnpSubscriptions MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY
    lnpSubscriptionsPkg,
    subscriptionVersionLocalSMS-CreatePkg;
  CONDITIONAL PACKAGES
  lnpDownloadPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionOldSP-CreatePkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionNewSP-CreatePkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionDisconnectPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionModifyPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionActivatePkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionCancelPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionOldSP-CancellationPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionNewSP-CancellationPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionRemoveFromConflictPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!,
  subscriptionVersionRangeStatusAttributeValueChangePkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeAttributeValueChangePkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeObjectCreationPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeDonorSP-CustomerDisconnectDatePkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeCancellationAcknowledgeRequestPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeNewSP-CreateRequestPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeOldSP-ConcurrenceRequestPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationPkg PRESENT
IF
    !the object is instantiated on the NPAC SMS!;
  subscriptionVersionRangeNewSPFinalCreateWindowExpirationPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  (NOTE: This item is to support the TN Range notification for the
NewSPFinalCreateWindowExpiration notification that is being created in
NANC 240)

  REGISTERED AS {LNP-OIDS.lnp-objectClass 14};

lnpSubscriptionsPkg PACKAGE
  BEHAVIOUR
```

```

        lnpSubscriptionsDefinition,
        lnpSubscriptionsBehavior;
ATTRIBUTES
        lnpSubscriptionsName GET;
NOTIFICATIONS
        subscriptionVersionLocalSMS-ActionResults
;

lnpSubscriptionsDefinition BEHAVIOUR
DEFINED AS !
    Local SMS and NPAC SMS Managed Object for the SOA to NPAC SMS and
    the Local SMS to NPAC SMS interface.

    The lnpSubscriptions class is the managed object that is used as
    the container object for the subscription version objects on the
    NPAC SMS and the Local SMS.

    Local SMS interfaces must be able to support scope/filtered M-SETs
    and M-DELETes with a TN range as the primary filter.

!;

lnpSubscriptionsBehavior BEHAVIOUR
DEFINED AS !
    Local SMS and NPAC SMS Managed Object

    The Local SMS (Data Download Association Function) and the service
    provider SOA (SOA Management Association Function) can M-GET any
    lnpSubscriptions object. The lnpSubscriptionsName attribute is
    read only and cannot be changed via the Local SMS Interface once
    the object has been created. The value of lnpSubscriptionsName
    will always be "lnpSubscriptions".

    Only one of these objects will exist and it will only be created
    at startup of the CMIP agent software on the NPAC SMS or the Local
    SMS.

    The lnpDownloadPkg will only be used for an lnpSubscriptions
    object instantiated on the NPAC SMS. This package is used to used
    for initiating downloading of subscriptionVersions object
    creation, deletion, or modifications to the Local SMS (Data
    Download Association Function).

    The subscriptionVersionOldSP-CreatePkg will only be used for an
    lnpSubscriptions object instantiated on the NPAC SMS. This
    package is used for creation of subscription versions for porting
    TNs by the old service provider.

    The subscriptionVersionNewSP-CreatePkg will only be used for an
    lnpSubscriptions object instantiated on the NPAC SMS. This
    package is used for creation of subscription versions for porting
    TNs by the new service provider.

    The subscriptionVersionDisconnectPkg will only be used for an
    lnpSubscriptions object instantiated on the NPAC SMS. This
    package is used for disconnection of a ported TN by the current
    service provider.

```


The `subscriptionVersionModifyPkg` will only be used for an `lnpSubscriptions` object instantiated on the NPAC SMS. This package is used for modification of a ported TN by a service provider.

The `subscriptionVersionActivatePkg` will only be used for an `lnpSubscriptions` object instantiated on the NPAC SMS. This package is used for activation of a ported TN by a new service provider.

The `subscriptionVersionCancelPkg` will only be used for an `lnpSubscriptions` object instantiated on the NPAC SMS. This package is used for cancellation of a ported TN by a service provider.

The `subscriptionVersionOldSP-CancellationPkg` will only be used for an `lnpSubscriptions` object instantiated on the NPAC SMS. This package is used for acknowledgment of subscription versions with status values of cancel-pending. Acknowledgments from both old and new service provider SOAs take a version from cancel-pending and to a canceled state. This action is used by the old service provider SOA.

The `subscriptionVersionNewSP-CancellationPkg` will only be used for an `lnpSubscriptions` object instantiated on the NPAC SMS. This package is used for acknowledgment of subscription versions with status values of cancel-pending. Acknowledgments from both old and new service provider SOAs take a version out of cancel-pending and to a canceled state. This action is used by the new service provider SOA.

The `subscriptionVersionRemoveFromConflictPkg` will only be used for an `lnpSubscriptions` object instantiated on the NPAC SMS. This package is used for setting the status of subscription versions with status values of conflict to pending. This action is used by either the new or old service provider SOA.

The SOA receives

`subscriptionVersionRangeStatusAttributeValueChange` notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The `subscriptionVersionRangeStatusAttributeValueChangePkg` is used to send the `subscriptionVersionRangeStatusAttributeValueChange` notification. When this package is sent, it will include one set of information for the TN range, plus a list of Subscription Version IDs. If the feature data does not apply to all TNs in the original range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives `subscriptionVersionRangeAttributeValueChange` notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The `subscriptionVersionRangeAttributeValueChangePkg` is used to send the `subscriptionVersionRangeAttributeValueChange` notification. When this package is sent, it will include one set of information

for the TN range, plus a list of Subscription Version IDs. If the feature data does not apply to all TNs in the original range, notifications will be broken up into smaller TN Range notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives `subscriptionVersionRangeObjectCreation` notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The `subscriptionVersionRangeObjectCreationPkg` is used to send the `subscriptionVersionRangeObjectCreation` notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives `subscriptionVersionRangeDonorSP-CustomerDisconnectDate` notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The `subscriptionVersionRangeDonorSP-CustomerDisconnectDatePkg` is used to send the `subscriptionVersionRangeDonorSP-CustomerDisconnectDate` notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives `subscriptionVersionRangeCancellationAcknowledgeRequest` notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The `subscriptionVersionRangeCancellationAcknowledgeRequestPkg` is used to send the `subscriptionVersionRangeCancellationAcknowledgeRequest` notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives `subscriptionVersionRangeNewSP-CreateRequest` notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The `subscriptionVersionRangeNewSP-CreateRequestPkg` is used to send the `subscriptionVersionRangeNewSP-CreateRequest` notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that

the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives subscriptionVersionRangeOldSP-ConcurrenceRequest notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeOldSP-ConcurrenceRequestPkg is used to send the subscriptionVersionRangeOldSP-ConcurrenceRequest notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The SubscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationPkg is used to send the subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA receives subscriptionVersionRangeNewSP-FinalCreateWindowExpiration notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The SubscriptionVersionRangeNewSP-FinalCreateWindow-ExpirationPkg is used to send the subscriptionVersionRangeNewSP-FinalCreateWindowExpiration notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

!;

-- Parameter Definitions
-- 15.0 Notification Recovery Action

```

lnpNotificationRecovery ACTION
  BEHAVIOUR
    lnpNotificationRecoveryDefinition,
    lnpNotificationRecoveryBehavior;
MODE CONFIRMED;
WITH INFORMATION SYNTAX LNP-
ASN1.NetworkNotificationRecoveryAction;
WITH REPLY SYNTAX LNP-ASN1.NetworkNotificationRecoveryReply;
REGISTERED AS {LNP-OIDS.lnp-action 15};

lnpNotificationRecoveryDefinition BEHAVIOUR
  DEFINED AS !
    The lnpNotificationRecovery action is the action that can be
    used by the SOA or LSMS to recover notification information
    that cannot be recovered by other means.

  !;

lnpNotificationRecoveryBehavior BEHAVIOUR
  DEFINED AS !
    Preconditions: This action is issued from an lnpNPAC-SMS
    object from a SOA or LSMS that specified the recovery mode
    flag in the access control as true at association
    establishment.

    Postconditions: After this action has been executed by the
    SOA or LSMS specifying recovery, the NPAC SMS will forward
    the notifications that occurred in the time range specified
    for the requesting system (SOA or LSMS) for the primary or
    associated SPID specified in the access control.
    Notifications are forwarded in the action reply.

    Notifications to be recovered are requested by time range.
    Time range requests will be limited to a tunable range
    specified in the NPAC SMS. All data in the download time
    period, regardless of the amount of data, will be returned.

    The recovery of the SOA and LSMS notifications is
    independent requests. Notifications can be recovered until
    they are purged from the database. The tunable used to
    determine when to purge the notifications is "Notify Log
    Retention Period" which defaults to 90 days.

    For all download requests, the Local SMS or SOA should
    behave as follows in response to the possible download M-
    ACTION response from the NPAC SMS:

        Success - process the data received from the NPAC SMS,
        continue processing.

        No-data-selected -- no data was found, continue
        processing.

        Criteria-too-large (using the MaxNotificationRecovery
        tunable) - break up the request into a smaller time
        range and re-issue the request to the NPAC SMS.

```

Time-range-invalid (using the Maximum Download Duration tunable) - break up the request into shorter time ranges and re-issue the request to the NPAC SMS.

Failed - go into retry mode. Re-issue the request a configurable number of additional retry attempts with an "x" amount of delay between requests ("x" is a configurable amount of time after receiving the failure for each request). If a failed response is received for the final retry request, abort the association and re-start the recovery process. Note: It is recommended that the Local SMS or SOA use the same value that the NPAC SMS uses for the retry interval. It is also recommended that the Local SMS use a value of at least two (2) for configurable number of additional retry attempts.

For activities that specify "continue processing" the Local SMS or SOA should send the NPAC SMS, either the next lnpDownload action for a different type of data, or an lnpRecoveryComplete request, depending on where the response appears in the flow.

The SOA or LSMS is capable of recovering data based on the association functions. The SOA recovers network data and notification data using the network data management association function (networkDataMgmt). The LSMS recovers network data and subscription data using the data download association function (dataDownload) and recovers notification data using the network data management association function (networkDataMgmt).

The SOA recovers subscriptionVersionRangeStatusAttributeValueChange notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeStatusAttributeValueChangePkg is used to send the subscriptionVersionRangeStatusAttributeValueChange notification. When this package is sent, it will include one set of information for the TN range, plus a list of Subscription Version IDs. If the feature data does not apply to all TNs in the original range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeAttributeValueChange notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeAttributeValueChangePkg is used to send the subscriptionVersionRangeAttributeValueChange notification. When this package is sent, it will include one set of information for the TN range, plus a list of Subscription Version IDs. If the feature data does not apply to all TNs in the original range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range.

The SOA recovers subscriptionVersionRangeObjectCreation notifications if their Service Provider TN Range Notification

Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeObjectCreationPkg is used to send the subscriptionVersionRangeObjectCreation notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeDonorSP-CustomerDisconnectDate notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeDonorSP-CustomerDisconnectDatePkg is used to send the subscriptionVersionRangeDonorSP-CustomerDisconnectDatePkg notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeCancellationAcknowledgeRequest notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeCancellationAcknowledgeRequestPkg is used to send the subscriptionVersionRangeCancellationAcknowledgeRequest notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeNewSP-CreateRequest notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeNewSP-CreateRequestPkg is used to send the subscriptionVersionRangeNewSP-CreateRequest notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeOldSP-ConcurrenceRequest notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeOldSP-ConcurrenceRequestPkg is used to send the subscriptionVersionRangeOldSP-

ConcurrenceRequest notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationPkg is used to send the subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

The SOA recovers subscriptionVersionRangeNewSP-FinalCreateWindowExpiration notifications if their Service Provider TN Range Notification Indicator is set to TRUE on the NPAC SMS. The subscriptionVersionRangeNewSP-FinalCreateWindowExpirationPkg is used to send the subscriptionVersionRangeNewSP-FinalCreateWindowExpiration notification. When this package is sent, it will include one set of information for the TN range, plus a paired list of TN/Subscription Version ID combinations. If the feature data does not apply to all TNs in the range, notifications will be broken up into smaller TN Range Notifications such that the feature data applies to all TNs in the smaller TN Range, and will be sent in separate messages.

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

!;

-- XXX.0 LNP Log Record for the Subscription Version Range Attribute Value Change Notification

```
lnpLogRangeAttributeValueChangeRecord MANAGED OBJECT CLASS
    DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
    CHARACTERIZED BY
        lnpLogRangeAttributeValueChangePkg;
    REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};
```

lnpLogRangeAttributeValueChangePkg PACKAGE

```

BEHAVIOUR
    lnpLogRangeAttributeValueChangeDefinition,
    lnpLogRangeAttributeValueChangeBehavior;
ATTRIBUTES
    subscriptionVersionRangeAttributeValueChangeInfo GET,
    accessControl GET;
    ;

lnpLogRangeAttributeValueChangeDefinition BEHAVIOUR
    DEFINED AS !
        The lnpLogAttributeValueChangeRecord class is the managed object
        that is used to create log records for the
        subscriptionVersionRangeAttributeValueChange Notification.
    !;

lnpLogRangeAttributeValueChangeBehavior BEHAVIOUR
    DEFINED AS !
        This log record can be used by any CME wanting to log the
        subscriptionVersionRangeAttributeValueChange Notification.
    !;

-- XXX.0 LNP Log Record for the Subscription Version Range Object Creation
Notification

lnpLogRangeObjectCreationRecord MANAGED OBJECT CLASS
    DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
    CHARACTERIZED BY
        lnpLogRangeObjectCreationPkg;
    REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeObjectCreationPkg PACKAGE
    BEHAVIOUR
        lnpLogRangeObjectCreationDefinition,
        lnpLogRangeObjectCreationBehavior;
    ATTRIBUTES
        subscriptionVersionRangeObjectInfo GET,
        accessControl GET;
    ;

lnpLogRangeObjectCreationDefinition BEHAVIOUR
    DEFINED AS !
        The lnpLogRangeObjectCreationRecord class is the managed object
        that is used to create log records for the
        subscriptionVersionRangeObjectCreation Notification.
    !;

lnpLogRangeObjectCreationChangeBehavior BEHAVIOUR
    DEFINED AS !
        This log record can be used by any CME wanting to log the
        subscriptionVersionObjectCreation Notification.
    !;

-- XXX.0 LNP Log Record for the Subscription Version Range Status Attribute
Value Change Notification

lnpLogRangeStatusAttributeValueChangeRecord MANAGED OBJECT CLASS

```



```

DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
CHARACTERIZED BY
    lnpLogRangeStatusAttributeValueChangePkg;
REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeStatusAttributeValueChangePkg PACKAGE
BEHAVIOUR
    lnpLogRangeStatusAttributeValueChangeDefinition,
    lnpLogRangeStatusAttributeValueChangeBehavior;
ATTRIBUTES
    subscriptionVersionRangeStatusAttributeValueChangeInfo GET,
    accessControl GET;
;

lnpLogRangeStatusAttributeValueChangeDefinition BEHAVIOUR
DEFINED AS !
    The lnpLogStatusAttributeValueChangeRecord class is the managed
    object that is used to create log records for the
    subscriptionVersionRangeStatusAttributeValueChange Notification.
!;

lnpLogRangeStatusAttributeValueChangeBehavior BEHAVIOUR
DEFINED AS !
    This log record can be used by any CME wanting to log the
    subscriptionVersionRangeStatusAttributeValueChange Notification.
!;

-- XXX.0 LNP Log Record for the Subscription Version Range Donor Disconnect
Notification

lnpLogRangeDonorSP-CustomerDisconnectDateRecord MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
CHARACTERIZED BY
    lnpLogRangeDonorSP-CustomerDisconnectDatePkg;
REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeDonorSP-CustomerDisconnectDatePkg PACKAGE
BEHAVIOUR
    lnpLogRangeDonorSP-CustomerDisconnectDateDefinition,
    lnpLogRangeDonorSP-CustomerDisconnectDateBehavior;
ATTRIBUTES
    subscriptionVersionRangeDonorSP-CustomerDisconnectDateInfo GET,
    accessControl GET;
;

lnpLogRangeDonorSP-CustomerDisconnectDateDefinition BEHAVIOUR
DEFINED AS !
    The lnpLogRangeDonorSP-CustomerDisconnectDateRecord class is the
    managed object that is used to create log records for the
    subscriptionVersionRangeDonorSP-CustomerDisconnectDate
    Notification.
!;

lnpLogRangeDonorSP-CustomerDisconnectDateBehavior BEHAVIOUR
DEFINED AS !

```

This log record can be used by any CME wanting to log the
subscriptionVersionRangeDonorSP-CustomerDisconnectDate
Notification.

!;

-- XXX.0 LNP Log Record for the Subscription Version Range Cancellation
Acknowledge Request Notification

lnpLogRangeCancellationAcknowledgeRequestRecord MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
CHARACTERIZED BY
lnpLogRangeCancellationAcknowledgeRequestPkg;
REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeCancellationAcknowledgeRequestPkg PACKAGE
BEHAVIOUR
lnpLogRangeCancellationAcknowledgeRequestDefinition,
lnpLogRangeCancellationAcknowledgeRequestBehavior;
ATTRIBUTES
subscriptionVersionRangeCancellationAcknowledgeRequestInfo GET,
accessControl GET;
;

lnpLogRangeCancellationAcknowledgeRequestDefinition BEHAVIOUR
DEFINED AS !
The lnpLogRangeCancellationAcknowledgeRequestRecord class is the
managed object that is used to create log records for the
subscriptionVersionRangeCancellationAcknowledgeRequest
Notification.
!;

lnpLogRangeCancellationAcknowledgeRequestBehavior BEHAVIOUR
DEFINED AS !
This log record can be used by any CME wanting to log the
subscriptionVersionRangeCancellationAcknowledgeRequest
Notification.
!;

-- XXX.0 LNP Log Record for the Subscription Version Range New SP Create
Request Notification

lnpLogRangeNewSP-CreateRequestRecord MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
CHARACTERIZED BY
lnpLogRangeNewSP-CreateRequestPkg;
REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeNewSP-CreateRequestPkg PACKAGE
BEHAVIOUR
lnpLogRangeNewSP-CreateRequestDefinition,
lnpLogRangeNewSP-CreateRequestBehavior;
ATTRIBUTES
subscriptionVersionRangeNewSP-CreateRequestInfo GET,
accessControl GET;
;

```

lnpLogRangeNewSP-CreateRequestDefinition BEHAVIOUR
  DEFINED AS !
    The lnpLogRangeNewSP-CreateRequestRecord class is the managed
    object that is used to create log records for the
    subscriptionVersionRangeNewSP-CreateRequest Notification.
  !;

lnpLogRangeNewSP-CreateRequestBehavior BEHAVIOUR
  DEFINED AS !
    This log record can be used by any CME wanting to log the
    subscriptionVersionRangeNewSP-CreateRequest Notification.
  !;

-- XXX.0 LNP Log Record for the Subscription Version Range Old SP Concurrence
Request Notification

lnpLogRangeOldSP-ConcurrenceRequestRecord MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
  1992":eventLogRecord;
  CHARACTERIZED BY
    lnpLogRangeOldSP-ConcurrenceRequestPkg;
  REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeOldSP-ConcurrenceRequestPkg PACKAGE
  BEHAVIOUR
    lnpLogRangeOldSP-ConcurrenceRequestDefinition,
    lnpLogRangeOldSP-ConcurrenceRequestBehavior;
  ATTRIBUTES
    subscriptionVersionRangeOldSP-ConcurrenceRequestInfo GET,
    accessControl GET;
  ;

lnpLogRangeOldSP-ConcurrenceRequestDefinition BEHAVIOUR
  DEFINED AS !
    The lnpLogRangeOldSP-ConcurrenceRequestRecord class is the managed
    object that is used to create log records for the
    subscriptionVersionRangeOldSP-ConcurrenceRequest Notification.
  !;

lnpLogRangeOldSP-ConcurrenceRequestBehavior BEHAVIOUR
  DEFINED AS !
    This log record can be used by any CME wanting to log the
    subscriptionVersionRangeOldSP-ConcurrenceRequest Notification.
  !;

-- XXX.0 LNP Log Record for the Subscription Version Range Old SP Final
Concurrence Window Expiration Notification

lnpLogRangeOldSPFinalConcurrenceWindowExpirationRecord MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
  1992":eventLogRecord;
  CHARACTERIZED BY
    lnpLogRangeOldSPFinalConcurrenceWindowExpirationPkg;
  REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeOldSPFinalConcurrenceWindowExpirationPkg PACKAGE

```

```

BEHAVIOUR
    lnpLogRangeOldSPFinalConcurrenceWindowExpirationDefinition,
    lnpLogRangeOldSPFinalConcurrenceWindowExpirationBehavior;
ATTRIBUTES
    subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationInfo
    GET,
    accessControl GET;
;

lnpLogRangeOldSPFinalConcurrenceWindowExpirationDefinition BEHAVIOUR
DEFINED AS !
    The lnpLogRangeOldSPFinalConcurrenceWindowExpirationRecord class
    is the managed object that is used to create log records for the
    subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration
    Notification.
!;

lnpLogRangeOldSPFinalConcurrenceWindowExpirationBehavior BEHAVIOUR
DEFINED AS !
    This log record can be used by any CME wanting to log the
    subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration
    Notification.
!;

-- XXX.0 LNP Log Record for the Subscription Version Range New SP Final Create
Window Expiration Notification

lnpLogRangeNewSP-FinalCreateWindowExpirationRecord MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
CHARACTERIZED BY
    lnpLogRangeNewSP-FinalCreateWindowExpirationPkg;
REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};

lnpLogRangeNewSP-FinalCreateWindowExpirationPkg PACKAGE
BEHAVIOUR
    lnpLogRangeNewSP-FinalCreateWindowExpirationDefinition,
    lnpLogRangeNewSP-FinalCreateWindowExpirationBehavior;
ATTRIBUTES
    subscriptionVersionRangeNewSP-FinalCreateWindowExpirationInfo GET,
    accessControl GET;
;

lnpLogRangeNewSP-FinalCreateWindowExpirationDefinition BEHAVIOUR
DEFINED AS !
    The lnpLogRangeNewSP-FinalCreateWindowExpirationRecord class is
    the managed object that is used to create log records for the
    subscriptionVersionRangeNewSP-FinalCreateWindowExpiration
    Notification.
!;

lnpLogRangeNewSP-FinalCreateWindowExpirationBehavior BEHAVIOUR
DEFINED AS !
    This log record can be used by any CME wanting to log the
    subscriptionVersionRangeNewSP-FinalCreateWindowExpiration
    Notification.

```

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

-- XXX.0 LNP Subscription Version Range Attribute Value Change Package

```
subscriptionVersionRangeAttributeValueChangePkg PACKAGE
  BEHAVIOUR subscriptionVersionRangeAttributeValueChangePkgBehavior;
  NOTIFICATIONS
    subscriptionVersionRangeAttributeValueChange;
  REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```
subscriptionVersionRangeAttributeValueChangePkgBehavior BEHAVIOUR
  DEFINED AS !
    This package provides for conditionally including the
    subscriptionVersionRangeAttributeValueChange notification.
  !;
```

-- XXX.0 LNP Subscription Version Range Object Creation Package

```
subscriptionVersionRangeObjectCreationPkg PACKAGE
  BEHAVIOUR subscriptionVersionRangeObjectCreationPkgBehavior;
  NOTIFICATIONS
    subscriptionVersionRangeObjectCreation;
  REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```
subscriptionVersionRangeObjectCreationPkgBehavior BEHAVIOUR
  DEFINED AS !
    This package provides for conditionally including the
    subscriptionVersionRangeObjectCreation notification.
  !;
```

-- XXX.0 LNP Subscription Version Range Status Attribute Value Change Package

```
subscriptionVersionRangeStatusAttributeValueChangePkg PACKAGE
  BEHAVIOUR subscriptionVersionRangeStatusAttributeValueChangePkgBehavior;
  NOTIFICATIONS
    subscriptionVersionRangeStatusAttributeValueChange;
  REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```
subscriptionVersionRangeStatusAttributeValueChangePkgBehavior BEHAVIOUR
  DEFINED AS !
    This package provides for conditionally including the
    subscriptionVersionRangeStatusAttributeValueChange notification.
  !;
```

-- XXX.0 LNP Subscription Version Range Donor Disconnect Package

```
subscriptionVersionRangeDonorSP-CustomerDisconnectDatePkg PACKAGE
  BEHAVIOUR subscriptionVersionRangeDonorSP-
  CustomerDisconnectDatePkgBehavior;
  NOTIFICATIONS
    subscriptionVersionRangeDonorSP-CustomerDisconnectDate;
  REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```

subscriptionVersionRangeDonorSP-CustomerDisconnectDatePkgBehavior
BEHAVIOUR
    DEFINED AS !
        This package provides for conditionally including the
        subscriptionVersionRangeDonorSP-CustomerDisconnectDate
        notification.
    !;

-- XXX.0 LNP Subscription Version Range Cancellation Acknowledge Package

subscriptionVersionRangeCancellationAcknowledgeRequestPkg PACKAGE
BEHAVIOUR
    subscriptionVersionRangeCancellationAcknowledgeRequestPkgBehavior;
NOTIFICATIONS
    subscriptionVersionRangeCancellationAcknowledgeRequest;
REGISTERED AS {LNP-OIDS.lnp-package XXX};

subscriptionVersionRangeCancellationAcknowledgeRequestPkgBehavior
BEHAVIOUR
    DEFINED AS !
        This package provides for conditionally including the
        subscriptionVersionRangeCancellationAcknowledgeRequest
        notification.
    !;

-- XXX.0 LNP Subscription Version Range New SP Create Package

subscriptionVersionRangeNewSP-CreateRequestPkg PACKAGE
BEHAVIOUR subscriptionVersionRangeNewSP-CreateRequestPkgBehavior;
NOTIFICATIONS
    subscriptionVersionRangeNewSP-CreateRequest;
REGISTERED AS {LNP-OIDS.lnp-package XXX};

subscriptionVersionRangeNewSP-CreateRequestPkgBehavior BEHAVIOUR
    DEFINED AS !
        This package provides for conditionally including the
        subscriptionVersionRangeNewSP-CreateRequest notification.
    !;

-- XXX.0 LNP Subscription Version Range Old SP Concurrence Package

subscriptionVersionRangeOldSP-ConcurrenceRequestPkg PACKAGE
BEHAVIOUR subscriptionVersionRangeOldSP-
ConcurrenceRequestPkgBehavior;
NOTIFICATIONS
    subscriptionVersionRangeOldSP-ConcurrenceRequest;
REGISTERED AS {LNP-OIDS.lnp-package XXX};

subscriptionVersionRangeOldSP-ConcurrenceRequestPkgBehavior BEHAVIOUR
    DEFINED AS !
        This package provides for conditionally including the
        subscriptionVersionRangeOldSP-ConcurrenceRequest
        notification.
    !;

-- XXX.0 LNP Subscription Version Range Old SP Final Concurrence Window
Expiration Package

```

```
subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationPkg PACKAGE
  BEHAVIOUR
  subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationPkgBehav
  ior;
  NOTIFICATIONS
    subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration;
  REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```
SubscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationPkgBehavior
BEHAVIOUR
  DEFINED AS !
    This package provides for conditionally including the
    subscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration
    notification.
  !;
```

```
-- XXX.0 LNP Subscription Version Range New SP Final Create Window  
Expiration Package
```

```
subscriptionVersionRangeNewSP-FinalCreateWindowExpirationPkg PACKAGE  
BEHAVIOUR subscriptionVersionRangeNewSP-  
FinalCreateWindowExpirationPkgBehavior;  
NOTIFICATIONS  
  subscriptionVersionRangeNewSP-FinalCreateWindowExpiration;  
REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```
SubscriptionVersionRangeNewSP-FinalCreateWindowExpirationPkgBehavior  
BEHAVIOUR  
  DEFINED AS !  
    This package provides for conditionally including the  
    subscriptionVersionRangeNewSP-FinalCreateWindowExpiration  
    notification.  
  !;
```

```
(NOTE: This item is to support the TN Range notification for the  
NewSPFinalCreateWindowExpiration notification that is being created in NANC  
240)
```

```
-- XXX.0 LNP Subscription Version Range Status Attribute Value Change  
Information
```

```
subscriptionVersionRangeStatusAttributeValueChangeInfo ATTRIBUTE  
  WITH ATTRIBUTE SYNTAX Attribute-  
  ASN1Module.RangeStatusAttributeValueChangeInfo;  
  MATCHES FOR EQUALITY;  
  BEHAVIOUR  
  subscriptionVersionRangeStatusAttributeValueChangeInfoBehavior;  
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};
```

```
subscriptionVersionRangeStatusAttributeValueChangeInfoBehavior BEHAVIOUR  
  DEFINED AS !  
    This attribute is used to store the range status attribute value  
    change information for subscription version range status attribute  
    value change notifications in a log record.  
  !;
```

```

-- XXX.0 LNP Subscription Version Range Attribute Value Change Information

subscriptionVersionRangeAttributeValueChangeInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-
  ASN1Module.RangeAttributeValueChangeInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR subscriptionVersionRangeAttributeValueChangeInfoBehavior;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeAttributeValueChangeInfoBehavior BEHAVIOUR
  DEFINED AS !
    This attribute is used to store the range attribute value change
    information for subscription version range attribute value change
    notifications in a log record.
  !;

-- XXX.0 LNP Subscription Version Object Creation Information

subscriptionVersionRangeObjectCreationInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.RangeObjectCreationInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR subscriptionVersionRangeObjectCreationInfoBehavior;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeObjectCreationInfoBehavior BEHAVIOUR
  DEFINED AS !
    This attribute is used to store the range object creation
    information for subscription version object creation information
    in a log record.
  !;

-- XXX.0 LNP Subscription Version Donor Disconnect Information

subscriptionVersionRangeDonorSP-CustomerDisconnectDateInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.RangeDonorDisconnectInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR subscriptionVersionRangeDonorSP-
  CustomerDisconnectDateInfoBehavior;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeDonorSP-CustomerDisconnectDateInfoBehavior BEHAVIOUR
  DEFINED AS !
    This attribute is used to store the range donor disconnect
    information for subscription version donor disconnect information
    in a log record.
  !;

-- XXX.0 LNP Subscription Version Cancellation Acknowledge Information

subscriptionVersionRangeCancellationAcknowledgeRequestInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-
  ASN1Module.RangeCancellationAcknowledgeInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR
  subscriptionVersionRangeCancellationAcknowledgeRequestInfoBehavior;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

```



```

subscriptionVersionRangeCancellationAcknowledgeRequestInfoBehavior BEHAVIOUR
  DEFINED AS !
      This attribute is used to store the range cancellation acknowledge
      information for subscription version cancellation acknowledge
      information in a log record.
  !;

-- XXX.0 LNP Subscription Version New SP Create Information

subscriptionVersionRangeNewSP-CreateRequestInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.RangeNewSP-CreateInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR subscriptionVersionRangeNewSP-CreateRequestInfoBehavior;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeNewSP-CreateRequestInfoBehavior BEHAVIOUR
  DEFINED AS !
      This attribute is used to store the range new SP create
      information for subscription version new SP create information in
      a log record.
  !;

-- XXX.0 LNP Subscription Version Old SP Concurrence Information

subscriptionVersionRangeOldSP-ConcurrenceRequestInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.RangeOldSP-ConcurrenceInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR subscriptionVersionRangeOldSP-ConcurrenceRequestInfoBehavior;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeOldSP-ConcurrenceRequestInfoBehavior BEHAVIOUR
  DEFINED AS !
      This attribute is used to store the range old SP concurrence
      information for subscription version old SP concurrence
      information in a log record.
  !;

-- XXX.0 LNP Subscription Version Old SP Final Concurrence Window Expiration
Information

subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationInfo ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Attribute-
  ASN1Module.RangeOldSPFinalConcurrenceWindowExpirationInfo;
  MATCHES FOR EQUALITY;
  BEHAVIOUR
  subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationInfoBehavio
  r;
  REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationInfoBehavior
BEHAVIOUR
  DEFINED AS !
      This attribute is used to store the range old SP final concurrence
      window expiration information for subscription version old SP
      final concurrence window expiration information in a log record.
  !;

```

-- XXX.0 LNP Subscription Version New SP Final Create Window Expiration Information

subscriptionVersionRangeNewSP-FinalCreateWindowExpirationInfo ATTRIBUTE
WITH ATTRIBUTE SYNTAX Attribute-ASN1Module.RangeNewSP-
FinalCreateWindowExpirationInfo;
MATCHES FOR EQUALITY;
BEHAVIOUR subscriptionVersionRangeNewSP-
FinalCreateWindowExpirationInfoBehavior;
REGISTERED AS {LNP-OIDS.lnp-attribute XXX};

subscriptionVersionRangeNewSP-FinalCreateWindowExpirationInfoBehavior
BEHAVIOUR

DEFINED AS !

This attribute is used to store the range new SP final create
window expiration information for subscription version new SP
final create window expiration information in a log record.

!;

(NOTE: This item is to support the TN Range notification for the
NewSPFinalCreateWindowExpiration notification that is being created in NANC
240)

-- XXX.0 LNP Subscription Version Range Status Attribute Value Change Notification

subscriptionVersionRangeStatusAttributeValueChange NOTIFICATION
BEHAVIOUR subscriptionVersionRangeStatusAttributeValueChangeBehavior;
WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeStatusAttributeValueChange
AND ATTRIBUTE IDS
 range-status-attribute-value-change-info
 subscriptionVersionRangeStatusAttributeValueChangeInfo,
 access-control accessControl;
REGISTERED AS {LNP-OIDS.lnp-notification XXX};

subscriptionVersionRangeStatusAttributeValueChangeBehavior BEHAVIOUR
DEFINED AS !

This notification type is used to report changes to the subscriptionVersionStatus field for range operations. It uses the status attribute value change notification which is based on the attribute value change notification as defined in M.3100 with the addition of the list of failed service providers in cases where the version status is active, failed or partially failed and the subscriptionStatusChangeCauseCode if it is set.

The service provider supports the notification if the Service Provider TN Range Notification Indicator is set on the NPAC SMS.

!;

-- XXX.0 LNP Subscription Version Range Attribute Value Change Notification

subscriptionVersionRangeAttributeValueChange NOTIFICATION
BEHAVIOUR subscriptionVersionRangeAttributeValueChangeBehavior;
WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeAttributeValueChange

```

AND ATTRIBUTE IDS
    range-attribute-value-change-info
    subscriptionVersionRangeAttributeValueChangeInfo,
    access-control accessControl;
REGISTERED AS {LNP-OIDS.lnp-notification XXX};

subscriptionVersionRangeAttributeValueChangeBehavior BEHAVIOUR
DEFINED AS !
    This notification type is used to report changes to the attributes
    in the subscription versions for range operations. It uses the
    attribute value change notification, which is based on the
    attribute value change notification as defined in M.3100.

    The service provider supports the notification if the Service
    Provider TN Range Notification Indicator is set on the NPAC SMS.
!;

-- XXX.0 LNP Subscription Version Range Object Creation Notification

subscriptionVersionRangeObjectCreation NOTIFICATION
BEHAVIOUR subscriptionVersionRangeObjectCreationBehavior;
WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeObjectCreation
AND ATTRIBUTE IDS
    range-object-creation-info
    subscriptionVersionRangeObjectInfo,
    access-control accessControl;
REGISTERED AS {LNP-OIDS.lnp-notification XXX};

subscriptionVersionRangeObjectCreationBehavior BEHAVIOUR
DEFINED AS !
    This notification type is used to report creation of subscription
    versions for range operations. It uses the object creation
    notification as defined in M.3100.

    The service provider supports the notification if the Service
    Provider TN Range Notification Indicator is set on the NPAC SMS.
!;

-- XXX.0 LNP Subscription Version Range Donor Disconnect Notification

subscriptionVersionRangeDonorSP-CustomerDisconnectDate NOTIFICATION
BEHAVIOUR subscriptionVersionRangeDonorSP-
CustomerDisconnectDateBehavior;
WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeDonorSP-
CustomerDisconnectDate
AND ATTRIBUTE IDS
    range-donor-disconnect-info
    subscriptionVersionRangeDonorSP-CustomerDisconnectDateInfo,
    access-control accessControl;
REGISTERED AS {LNP-OIDS.lnp-notification XXX};

subscriptionVersionRangeDonorSP-CustomerDisconnectDateBehavior BEHAVIOUR
DEFINED AS !
    This notification informs the donor service provider SOA
    that a range of subscription versions is being disconnected.

```

The TN, the version id, customer disconnect date and effective release date (optional) values are sent.

The service provider supports the notification if the Service Provider TN Range Notification Indicator is set on the NPAC SMS.

!;

-- XXX.0 LNP Subscription Version Range Cancellation Acknowledge Notification

```
subscriptionVersionRangeCancellationAcknowledgeRequest NOTIFICATION
  BEHAVIOUR
  subscriptionVersionRangeCancellationAcknowledgeRequestBehavior;
  WITH INFORMATION SYNTAX LNP-
  ASN1.VersionRangeCancellationAcknowledgeRequest
  AND ATTRIBUTE IDS
    range-cancellation-acknowledge-info
    subscriptionVersionRangeCancellationAcknowledgeRequestInfo,
    access-control accessControl;
  REGISTERED AS {LNP-OIDS.lnp-notification XXX};
```

```
subscriptionVersionRangeCancellationAcknowledgeRequestBehavior BEHAVIOUR
  DEFINED AS !
```

This notification requests that a service provider send a cancellation acknowledgement for a range of subscription versions. The TNs and version ids are sent.

The service provider supports the notification if the Service Provider TN Range Notification Indicator is set on the NPAC SMS.

!;

-- XXX.0 LNP Subscription Version Range New SP Create Notification

```
subscriptionVersionRangeNewSP-CreateRequest NOTIFICATION
  BEHAVIOUR subscriptionVersionRangeNewSP-CreateRequestBehavior;
  WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeNewSP-CreateRequest
  AND ATTRIBUTE IDS
    range-new-sp-create-info
    subscriptionVersionRangeNewSP-CreateRequestInfo,
    access-control accessControl;
  REGISTERED AS {LNP-OIDS.lnp-notification XXX};
```

```
subscriptionVersionRangeNewSP-CreateRequestBehavior BEHAVIOUR
  DEFINED AS !
```

This notification requests that a new service provider send a create request for a range of subscription versions for which concurrence for porting the number has not been received. The TNs, the version ids and the old service provider id, authorization flag and authorization timestamp values are sent. If the new service provider supports timer type, it will be sent. If the new service provider supports business type, it will be sent.

The service provider supports the notification if the Service Provider TN Range Notification Indicator is set on the NPAC SMS.

!;

-- XXX.0 LNP Subscription Version Range Old SP Concurrence Notification

```
subscriptionVersionRangeOldSP-ConcurrenceRequest NOTIFICATION
  BEHAVIOUR subscriptionVersionRangeOldSP-ConcurrenceRequestBehavior;
  WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeOldSP-ConcurrenceRequest
  AND ATTRIBUTE IDS
    range-old-sp-concurrence-info
    subscriptionVersionRangeOldSP-ConcurrenceRequestInfo,
    access-control accessControl;
  REGISTERED AS {LNP-OIDS.lnp-notification XXX};
```

```
subscriptionVersionRangeOldSP-ConcurrenceRequestBehavior BEHAVIOUR
  DEFINED AS !
  This notification requests that an old service provider send
  a create request for a range of subscription versions for
  which concurrence for porting the number has not been
  received. The TNs, the version ids, and the new service
  provider id, authorization flag and creation timestamp
  values are sent. If the old service provider supports timer
  type, it will be sent. If the old service provider supports
  business type, it will be sent.

  The service provider supports the notification if the Service
  Provider TN Range Notification Indicator is set on the NPAC SMS.
  !;
```

```
-- XXX.0 LNP Subscription Version Range Old SP Final Concurrence Window
Expiration Notification
```

```
SubscriptionVersionRangeOldSPFinalConcurrenceWindowExpiration NOTIFICATION
  BEHAVIOUR
  subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationBehavior;
  WITH INFORMATION SYNTAX LNP-
  ASN1.VersionRangeOldSPFinalConcurrenceWindowExpiration
  AND ATTRIBUTE IDS
    range-old-sp-final concurrence-window-expiration-info
    subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationInfo,
    access-control accessControl;
  REGISTERED AS {LNP-OIDS.lnp-notification XXX};
```

```
subscriptionVersionRangeOldSPFinalConcurrenceWindowExpirationBehavior
BEHAVIOUR
  DEFINED AS !
  This notification will be sent by the NPAC SMS upon
  expiration of the Final Concurrence Timer to the old service
  provider via the SOA to NPAC SMS interface to inform them of
  the timer expiration for a range of TNs. If the old service
  provider supports timer type, it will be sent. If the old
  service provider supports business type, it will be sent.

  The service provider supports the notification if the Service
  Provider TN Range Notification Indicator is set on the NPAC SMS.
  !;
```

```
-- XXX.0 LNP Subscription Version Range New SP Final Create Window Expiration
Notification
```

```
SubscriptionVersionRangeNewSP-FinalCreateWindowExpiration NOTIFICATION
```

BEHAVIOUR subscriptionVersionRangeNewSP-FinalCreateWindowExpirationBehavior;
WITH INFORMATION SYNTAX LNP-ASN1.VersionRangeNewSP-FinalCreateWindowExpiration
AND ATTRIBUTE IDS
range-new-sp-final-create-window-expiration-info
subscriptionVersionRangeNewSP-FinalCreateWindowExpirationInfo,
access-control accessControl;
REGISTERED AS {LNP-OIDS.lnp-notification XXX};

subscriptionVersionRangeNewSP-FinalCreateWindowExpirationBehavior BEHAVIOUR
DEFINED AS !

This notification indicates the final create window has expired for concurrence of a subscription version. This occurs when an old service provider has sent a create request for a subscription version and the new service provider has not concurred to the port within the period of time of both the initial and final concurrence timers. Once the final create window has expired, this notification is sent to both the old and new service providers that support this notification.

On the NPAC SMS, the Final Create Window has expired, but the NPAC SMS has not updated the status to cancel.

The TN, the version id and the new and old service provider id, cause code, authorization flag and authorization timestamp values are sent. If the new service provider supports timer type, it will be sent. If the new service provider supports business type, it will be sent.

The subscription version remains in a status of pending or conflict.

The service provider supports the notification if the Service Provider TN Range Notification Indicator is set on the NPAC SMS.

!;

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

ASN.1:

Import Clause Modification:

-- DMI

```
AttributeValueChangeInfo, ObjectInfo
  FROM Notification-ASN1Module {joint-iso-ccitt ms(9) smi(3)
  part2(2)
  asn1Module(2) 2};
```

```
NetworkNotificationRecoveryReply ::= SEQUENCE {
  status ENUMERATED {
    success (0),
    failed (1),
```

```

        time-range-invalid (2),
        criteria-to-large (3),
        no-data-selected (4)
    },
    system-choice CHOICE {

    lsms [1] SET OF SEQUENCE {
        managedObjectClass ObjectClass,
        managedObjectInstance ObjectInstance,
        notification CHOICE {
            subscription-version-new-mpa-nxx [1]
                VersionNewMPA-NXX-Recovery,
            lnp-mpac-sms-operational-information [2]
                MPAC-SMS-Operational-InformationRecovery
        }
    },
    soa [2] SET OF SEQUENCE {
        managedObjectClass ObjectClass,
        managedObjectInstance ObjectInstance,
        notification CHOICE {
            subscription-version-new-mpa-nxx [1]
                VersionNewMPA-NXX-Recovery,
            subscription-version-donor-sp-customer-disconnect-date
            [2]
                VersionCustomerDisconnectDateRecovery,
            subscription-version-audit-discrepancy-report [3]
                AuditDiscrepancyRptRecovery,
            subscription-audit-results [4]
                AuditResultsRecovery,
            lnp-mpac-sms-operational-information [5]
                MPAC-SMS-Operational-InformationRecovery,
            subscription-version-new-sp-create-request [6]
                VersionNewSP-CreateRequestRecovery,
            subscription-version-old-sp-concurrence-request [7]
                VersionOldSP-ConcurrenceRequestRecovery,
            subscription-version-old-sp-final-window-expiration
            [8]
                VersionOldSPFinalConcurrenceWindowExpirationReco
                very,
            subscription-version-cancellation-acknowledge-request
            [9]
                VersionCancellationAcknowledgeRequestRecovery,
            subscriptionVersionStatusAttributeValueChange [10]
                VersionStatusAttributeValueChangeRecovery,
            attribute-value-change [11] AttributeValueChangeInfo,
            object-creation [12] ObjectInfo,
            object-deletion [13] ObjectInfo,
            numberPoolBlockStatusAttributeValueChange [14]
                NumberPoolBlockStatusAttributeValueChangeRecover
                Y,
            versionRangeAttributeValueChangeRecovery [15]
                VersionRangeAttributeValueChangeRecovery,
            versionRangeObjectCreationRecovery [16]
                VersionRangeObjectCreationRecovery,
            versionRangeStatusAttributeValueChangeRecovery [17]
                VersionRangeStatusAttributeValueChangeRecovery
        }
    }
}

```

```

versionRangeDonorSP-CustomerDisconnectDateRecovery
[18]
    VersionRangeDonorSP-
    CustomerDisconnectDateRecovery
versionRangeCancellationAcknowledgeRequestRecovery
[19]
    VersionRangeCancellationAcknowledgeRequestRecovery
versionRangeNewSP-CreateRequestRecovery [20]
    VersionRangeNewSP-CreateRequestRecovery
versionRangeOldSP-ConcurrenceRequestRecovery [21]
    VersionRangeOldSP-ConcurrenceRequestRecovery
versionRangeOldSPFinalConcurrenceWindowExpirationRecovery [22]
    VersionRangeOldSPFinalConcurrenceWindowExpirationRecovery
versionRangeNewSP-FinalCreateWindowExpirationRecovery
[23]
    VersionRangeNewSP-
    FinalCreateWindowExpirationRecovery

```

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

```

    }
} OPTIONAL
}

```

New ASN.1:

```

RangeStatusAttributeValueChangeInfo ::= SEQUENCE {
    version-id [0] SET OF LnpKey,
    value-change-info [1] AttributeValueChangeInfo,
    failed-service-provs [2] Failed-SP-List OPTIONAL,
    subscription-status-change-cause-code [3]
    SubscriptionStatusChangeCauseCode OPTIONAL
}

RangeAttributeValueChangeInfo ::= SEQUENCE {
    version-id SET OF LnpKey,
    value-change-info AttributeValueChangeInfo
}

RangeObjectCreationInfo ::= SEQUENCE {
    tn-version-id SET OF TN-VersionId,
    object-info SET OF ObjectInfo
}

RangeDonorSP-CustomerDisconnectDateInfo ::= SEQUENCE {
    version-id [0] SET OF LnpKey,
    customer-disconnect-date [1] GeneralizedTime,
    effective-release-date [2] GeneralizedTime OPTIONAL
}

```


RangeCancellationAcknowledgeRequest ::= SET OF
VersionCancellationAcknowledgeRequest

RangeNewSP-CreateRequestInfo ::= SEQUENCE {
tn-version-id SET OF TN-VersionId,
version-id SET OF LnpKey,
service-prov-id ServiceProvId,
service-prov-due-date GeneralizedTime,
service-prov-authorization-creation-time-stamp GeneralizedTime,
service-prov-old-authorization ServiceProvAuthorization,
subscription-status-change-cause-code
SubscriptionStatusChangeCauseCode,
subscription-timer-type [0] Integer OPTIONAL,
subscription-business-type [1] Integer OPTIONAL
}

RangeOldSP-ConcurrenceRequestInfo ::= SEQUENCE {
tn-version-id SET OF TN-VersionID,
version-id SET OF LnpKey,
service-prov-id ServiceProvId,
service-prov-due-date GeneralizedTime,
service-prov-authorization-creation-time-stamp GeneralizedTime,
subscription-timer-type [0] Integer OPTIONAL,
subscription-business-type [1] Integer OPTIONAL
}

RangeOldSPFinalConcurrenceWindowExpirationInfo ::= SEQUENCE {
tn-version-id SET OF TN-VersionID,
version-id SET OF LnpKey,
subscription-timer-type [0] Integer OPTIONAL,
subscription-business-type [1] Integer OPTIONAL
}

RangeNewSP-FinalCreateWindowExpirationInfo ::= SEQUENCE {
tn-version-id [0] SET OF TN-VersionID,
version-create-request [1] VersionCreateConcurrenceRequest,
new-service-prov-id [2] ServiceProvId,
service-prov-old-authorization [3] ServiceProvAuthorization,
subscription-status-change-cause-code [4]
SubscriptionStatusChangeCauseCode OPTIONAL,
}

(NOTE: This item is to support the TN Range notification for the
NewSPFinalCreateWindowExpiration notification that is being created in NANC
240)

VersionRangeAttributeValueChange ::= SEQUENCE {
range-attribute-value-change RangeAttributeValueChangeInfo,
access-control LnpAccessControl
}

VersionRangeAttributeValueChangeRecovery = RangeAttributeValueChangeInfo

VersionRangeObjectCreation ::= SEQUENCE {
range-object-creation RangeObjectCreationInfo,
access-control LnpAccessControl
}

```

VersionRangeObjectCreationRecovery ::= RangeObjectCreationInfo

VersionRangeStatusAttributeValueChange ::= SEQUENCE {
    Range-status-attribute-value-changes
    RangeStatusAttributeValueChangeInfo,
    access-control LnpAccessControl
}

VersionRangeStatusAttributeValueChangeRecovery ::=
RangeStatusAttributeValueChangeInfo

VersionRangeDonorSP-CustomerDisconnectDate ::= SEQUENCE {
    Range-donor-disconnect RangeDonorSP-CustomerDisconnectDateInfo,
    access-control LnpAccessControl
}

VersionRangeDonorSP-CustomerDisconnectDateRecovery ::= RangeDonorSP-
CustomerDisconnectDateInfo

VersionRangeCancellationAcknowledgeRequest ::= SEQUENCE {
    tn-version-id SET OF TN-VersionID,
    version-id SET OF LnpKey,
    access-control LnpAccessControl
}

VersionRangeCancellationAcknowledgeRequestRecovery ::=
RangeCancellationAcknowledgeRequestInfo

VersionRangeNewSP-CreateRequest ::= SEQUENCE {
    Range-new-sp-create RangeNewSP-CreateRequestInfo,
    access-control LnpAccessControl
}

VersionRangeNewSP-CreateRequestRecovery ::= RangeNewSP-CreateRequestInfo

VersionRangeOldSP-ConcurrenceRequest ::= SEQUENCE {
    Range-donor-disconnect RangeOldSP-ConcurrenceRequestInfo,
    access-control LnpAccessControl
}

VersionRangeOldSP-ConcurrenceRequestRecovery ::= RangeOldSP-
ConcurrenceRequestInfo

VersionRangeOldSPFinalConcurrenceWindowExpiration ::= SEQUENCE {
    Range-donor-disconnect
    RangeOldSPFinalConcurrenceWindowExpirationInfo,
    access-control LnpAccessControl
}

VersionRangeOldSPFinalConcurrenceWindowExpirationRecovery ::=
RangeOldSPFinalConcurrenceWindowExpirationInfo

VersionRangeNewSP-FinalCreateWindowExpiration ::= SEQUENCE {
    Range-donor-disconnect RangeNewSP-FinalCreateWindowExpirationInfo,
    access-control LnpAccessControl

```

1

VersionRangeNewSP-FinalCreateWindowExpirationRecovery ::= RangeNewSP-FinalCreateWindowExpirationInfo

(NOTE: This item is to support the TN Range notification for the NewSPFinalCreateWindowExpiration notification that is being created in NANC 240)

M&P:

No change required.

Origination Date: 10/15/1998

Change Order Number: NANC 240

Description: No cancellation of SVs based on expiration of T2 timer

Cumulative SP Priority, Weighted Average: 5.33

Pure Backwards Compatible: NO

Functional Backwards Compatible: YES

IMPACT/CHANGE ASSESSMENT

FRS	IIS	GDMO	ASN.1	NPAC	SOA	LSMS
Y	Y	Y	Y	Low	Low	N/A

Business Need:

Currently the NPAC cancels an SV as a result of the new SP failing to send up a matching create within 18 hours. Additional work effort and operational concerns are created when this occurs, including the new SP re-contacting the old SP, and having to re-do the LSR/FOC paper work. In addition to excessive contact between both SPs, the extra work created can delay the actual port and increase the likelihood of disconnects in error. This change order will eliminate the cancellation of the SV and reduce the level of effort and operational issues.

Description of Change:

During the discussion of NANC 198, it was mentioned that Service Providers end up doing more work if the NPAC cancels an SV, at the expiration of the T2 timer, when a New SP does NOT send up a matching Create message.

Therefore, this change order has been opened to explore the possibility of changing the NPAC to cancel the SV, "*at some later date*", than the expiration of T2, which is what the current functionality requires (R5-23.4 New Service Provider Fails to Authorize Transfer of Service).

Keep in mind that if we determine we do NOT want the NPAC to auto cancel at the expiration of T2 (and want some later date), then we need to separate this from the T2 timer. Need to add the option that we may need to incorporate this auto cancel into some type of housekeeping, and not have it scheduled like today's T1 and T2 timers.

The current proposed solution is to leave the SV as pending upon expiration of T2. This will be cleaned up by NPAC housekeeping in 30 days (Pending Subscription Retention parameter, (as indicated in RR5-51.1).

Requirements:

R5-23.4 New Service Provider Fails to Authorize Transfer of Service

Deleted.

Req 1 New Service Provider Final Create Window Expiration Notification

NPAC SMS shall upon expiration of the Final Concurrence Window, where a new Service Provider has not sent authorization for the transfer of service, send a notification to both the old Service Provider that supports the Final Create Window Expiration Notification and the new Service Provider that supports the Final Create Window Expiration Notification via the SOA to NPAC SMS Interface, to inform them of the timer expiration.

Req 2 New Service Provider Final Create Window Expiration Notification – Sending of Cause Code

NPAC SMS shall only send the Subscription Version Status Change Cause Code in the Final Create Window Expiration Notification when the old Service Provider authorization is **FALSE**.

Req 3 NPAC Customer No New SP Concurrence Notification Indicator

NPAC SMS shall provide a mechanism to indicate whether a Service Provider supports the Final Create Window Expiration Notification for a Subscription Version upon the expiration of the New Service Provider Final Create Window.

Req 4 NPAC Customer No New SP Concurrence Notification Indicator – Default

NPAC SMS shall default the No New SP Concurrence Notification Indicator to **FALSE**.

Req 5 NPAC Customer No New SP Concurrence Notification Indicator – Modification

NPAC SMS shall allow NPAC Personnel, via the NPAC Administrative Interface, to modify the No New SP Concurrence Notification Indicator on the NPAC Customer record.

Req 6 Subscription Version Information – Suppress Notification when Service Provider No New SP Concurrence Notification Indicator is False

NPAC SMS shall suppress the Final Create Window Expiration Notification, if the Service Provider's No New SP Concurrence Notification Indicator is **FALSE**.

Req 7 Subscription Version Information – Send Notification when Service Provider No New SP Concurrency Notification Indicator is True

NPAC SMS shall send the Final Create Window Expiration Notification, if the Service Provider's No New SP Concurrency Notification Indicator is **TRUE**.

RR6-29 (Modified) Notification Recovery

NPAC SMS shall support recovery of all CMIP notifications defined in the IIS that are emitted over the NPAC SMS to Local SMS and SOA to NPAC SMS interfaces.

Examples of notifications to be recovered include:

- subscriptionVersionNewNPA-NXX
- subscriptionVersionDonorSP-CustomerDisconnectDate
- subscriptionVersionAudit-DiscrepancyRpt
- subscriptionAuditResults
- InpNPAC-SMS-Operational-Information
- subscriptionVersionNewSP-CreateRequest (time sensitive T1 New SP)
- subscriptionVersionOld-SP-ConcurrencyRequest (time sensitive T1 Old SP)
- subscriptionVersionOldSPFinalWindowExpiration (time sensitive T2 Old SP)
- subscriptionVersionStatusAttributeValueChange
- numberPoolBlockStatusAttributeValueChange
- attributeValueChange
- objectCreation
- objectDeletion
- *subscriptionVersionNewSP-FinalCreateWindowExpiration (if supported by the recovering SOA)*

For a complete list of notifications reference the IIS.

IIS:

A new entry will be added in section 4.1.4, Notification Interface Functionality, to Exhibit 1, *The Notification Interface Functionality Table*.

B.5.1.6.4 Subscription Version Create: Failure to Receive Response from New SOA

This scenario shows ~~action taken by~~ the *process of the* NPAC SMS after not receiving any concurrence from the new service provider after the “Final Service Provider Concurrency ~~Failure~~ Window.”

The subscription version remains in the NPAC SMS with a status of pending.

Step-by-step message flow text is shown below:

1. NPAC SMS receives no *concurrency occurrence* from the new service provider SOA in “Service Provider Concurrency **Failure** Window” for the pending subscriptionVersionNPAC created by the old service provider SOA.
2. *NPAC SMS notifies the old service provider, if they support the notification according to their NPAC Customer No New SP Concurrency Notification Indicator in their service provider profile on the NPAC SMS, of the expiration of the final create window where the new service provider did not send up a Create action for this subscription version, with an M-EVENT-REPORT subscriptionVersionNewSP-FinalCreateWindowExpiration.*
3. *The old service provider SOA returns an M-EVENT-REPORT confirmation to the NPAC SMS.*
4. *NPAC SMS notifies the new service provider, if they support the notification according to their NPAC Customer No New SP Concurrency Notification Indicator in their service provider profile on the NPAC SMS, of the expiration of the final create window where the new service provider did not send up a Create action for this subscription version, with an M-EVENT-REPORT subscriptionVersionNewSP-FinalCreateWindowExpiration.*
5. *The new service provider SOA returns an M-EVENT-REPORT confirmation to the NPAC SMS.*
- ~~6.—NPAC SMS issues M-SET for subscriptionVersionStatus to set it to “cancel” and the subscriptionModifiedTimeStamp in the subscriptionVersionNPAC object.~~
- ~~7.—NPAC SMS responds to M-SET.~~
- ~~8.—If the subscriptionVersionNPAC object was modified, the NPAC SMS notifies the old service provider of the status change.~~
- ~~9.—The old service provider SOA returns an M-EVENT-REPORT confirmation to the NPAC SMS.~~
- ~~10.—If the subscriptionVersionNPAC object was modified, the NPAC SMS notifies new service provider SOA of the status change.~~
- ~~11.—The new service provider SOA returns an M-EVENT-REPORT confirmation to the NPAC SMS.~~

GDMO:

-- 21.0 LNP NPAC Subscription Version Managed Object Class

```
subscriptionVersionNPAC MANAGED OBJECT CLASS
  DERIVED FROM subscriptionVersion;
  CHARACTERIZED BY
    subscriptionVersionNPAC-Pkg;
  CONDITIONAL PACKAGES
    subscriptionVersionNewSP-FinalCreateWindowExpirationPkg PRESENT IF
    !the object is instantiated on the NPAC SMS!;
  REGISTERED AS {LNP-OIDS.lnp-objectClass 21};

subscriptionVersionNPAC-Pkg PACKAGE
  BEHAVIOUR
    subscriptionVersionNPAC-Definition,
    subscriptionVersionNPAC-Behavior;
  ATTRIBUTES
    subscriptionVersionStatus GET-REPLACE,
    subscriptionOldSP GET-REPLACE,
    subscriptionNewSP-DueDate GET-REPLACE,
    subscriptionNewSP-CreationTimeStamp GET-REPLACE,
    subscriptionOldSP-DueDate GET-REPLACE,
    subscriptionOldSP-Authorization GET-REPLACE,
    subscriptionStatusChangeCauseCode GET-REPLACE,
    subscriptionOldSP-AuthorizationTimeStamp GET-REPLACE,
    subscriptionBroadcastTimeStamp GET-REPLACE,
    subscriptionConflictTimeStamp GET-REPLACE,
    subscriptionCustomerDisconnectDate GET-REPLACE,
    subscriptionEffectiveReleaseDate GET-REPLACE,
    subscriptionDisconnectCompleteTimeStamp GET-REPLACE,
    subscriptionCancellationTimeStamp GET-REPLACE,
    subscriptionCreationTimeStamp GET-REPLACE,
    subscriptionFailed-SP-List GET-REPLACE,
    subscriptionModifiedTimeStamp GET-REPLACE,
    subscriptionOldTimeStamp GET-REPLACE,
    subscriptionOldSP-CancellationTimeStamp GET-REPLACE,
    subscriptionNewSP-CancellationTimeStamp GET-REPLACE,
    subscriptionOldSP-ConflictResolutionTimeStamp GET-REPLACE,
    subscriptionNewSP-ConflictResolutionTimeStamp GET-REPLACE,
    subscriptionPortingToOriginal-SPSwitch GET-REPLACE,
    subscriptionPreCancellationStatus GET-REPLACE,
    subscriptionTimerType GET,
    subscriptionBusinessType GET;
  NOTIFICATIONS
    subscriptionVersionOldSP-ConcurrenceRequest,
    subscriptionVersionNewSP-CreateRequest,
    subscriptionVersionOldSPFinalConcurrenceWindowExpiration,
    subscriptionVersionNewNPA-NXX,
    subscriptionVersionCancellationAcknowledgeRequest,
    subscriptionVersionDonorSP-CustomerDisconnectDate,
    subscriptionVersionStatusAttributeValueChange,
    "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":attributeValueChange
    accessControlParameter,
    "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":objectCreation
    accessControlParameter;
;
```



```
-- XXX.0 LNP Log Record for the LNP Subscription version New Service
-- Provider Final Create Window Expiration Notification
```

```
lnpLogNewSP-FinalCreateWindowExpiration Record MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 :
1992":eventLogRecord;
```

```
CHARACTERIZED BY
    lnpLogNewSP-FinalCreateWindowExpirationPkg;
REGISTERED AS {LNP-OIDS.lnp-objectClass XXX};
```

```
lnpLogNewSP-FinalCreateWindowExpirationPkg PACKAGE
BEHAVIOUR
```

```
    lnpLogNewSP-FinalCreateWindowExpirationDefinition,
    lnpLogNewSP-FinalCreateWindowExpirationBehavior;
ATTRIBUTES
```

```
    subscriptionTN GET,
    subscriptionVersionId GET,
    subscriptionNewSP GET,
    subscriptionOldSP GET,
    subscriptionOldSP-DueDate GET,
    subscriptionOldSP-Authorization, GET
    subscriptionOldSP-AuthorizationTimeStamp GET,
    subscriptionStatusChangeCauseCode GET,
    subscriptionTimerType GET,
    subscriptionBusinessType GET,
    accessControl GET;
```

```
;
```

```
lnpLogNewSP-FinalCreateWindowExpirationDefinition BEHAVIOUR
DEFINED AS !
```

```
    The lnpLogNewSP-FinalCreateWindowExpirationRecord class is the
    managed object that is used to create log records for the
    subscriptionVersionNewSP-FinalCreateWindowExpiration Notification.
```

```
!;
```

```
lnpLogNewSP-FinalCreateWindowExpirationBehavior BEHAVIOUR
DEFINED AS !
```

```
    This log record can be used by any CME wanting to log the
    subscriptionVersionNewSP-FinalCreateWindowExpiration Notification.
```

```
!;
```

```
-- XXX14.0 LNP Subscription Version New SP Final Create Window Expiration
Package
```

```
subscriptionVersionNewSP-FinalCreateWindowExpirationPkg PACKAGE
```

```
BEHAVIOUR subscriptionVersionNewSP-
FinalCreateWindowExpirationPkgBehavior;
NOTIFICATIONS
```

```
    subscriptionVersionNewSP-FinalCreateWindowExpiration;
REGISTERED AS {LNP-OIDS.lnp-package XXX};
```

```
subscriptionVersionNewSP-FinalCreateWindowExpirationPkgBehavior BEHAVIOUR
DEFINED AS !
```

```
    This package provides for conditionally including the
    subscriptionVersionNewSP-FinalCreateWindowExpiration notification.
```

!;

-- XXX.0 LNP Subscription Version New SP Final Create Window Expiration Notification

```
subscriptionVersionNewSP-FinalCreateWindowExpiration NOTIFICATION
  BEHAVIOUR subscriptionVersionNewSP-
  FinalCreateWindowExpirationPkgBehavior;
  WITH INFORMATION SYNTAX LNP-ASN1.VersionNewSP-
  FinalCreateWindowExpiration
  AND ATTRIBUTE IDS
    tn subscriptionTN,
    version-id subscriptionVersionId,
    new-service-prov-id subscriptionNewSP,
    old-service-prov-id subscriptionOldSP,
    service-prov-due-date subscriptionOldSP-DueDate,
    service-prov-old-authorization subscriptionOldSP-Authorization,
    service-prov-authorization-creation-time-stamp
    subscriptionOldSP-AuthorizationTimeStamp,
    status-change-cause-code subscriptionStatusChangeCauseCode,
    access-control accessControl,
    subscription-timer-type subscriptionTimerType,
    subscription-business-type subscriptionBusinessType;
  REGISTERED AS {LNP-OIDS.lnp-notification XXX};
```

subscriptionVersionNewSP-FinalCreateWindowExpirationPkgBehavior BEHAVIOUR
DEFINED AS !

This notification indicates the final create window has expired for concurrence of a subscription version. This occurs when an old service provider has sent a create request for a subscription version and the new service provider has not concurred to the port within the period of time of both the initial and final concurrence timers. Once the final create window has expired, this notification is sent to both the old and new service providers that support this notification.

On the NPAC SMS, the Final Create Window has expired, but the NPAC SMS has not updated the status to cancel.

The TN, the version id and the new and old service provider id, cause code, authorization flag and authorization timestamp values are sent. If the new service provider supports timer type, it will be sent. If the new service provider supports business type, it will be sent.

The subscription version remains in a status of pending or conflict.

!;

ASN.1:

```
NetworkNotificationRecoveryReply ::= SEQUENCE {
    status ENUMERATED {
        success (0),
        failed (1),
        time-range-invalid (2),
        criteria-to-large (3),
        no-data-selected (4)
    },
    system-choice CHOICE {

    lsms [1] SET OF SEQUENCE {
        managedObjectClass ObjectClass,
        managedObjectInstance ObjectInstance,
        notification CHOICE {
            subscription-version-new-npa-nxx [1] VersionNewNPA-NXX-Recovery,
            lnp-npac-sms-operational-information [2]
            NPAC-SMS-Operational-InformationRecovery
        }
    },
    soa [2] SET OF SEQUENCE {
        managedObjectClass ObjectClass,
        managedObjectInstance ObjectInstance,
        notification CHOICE {
            subscription-version-new-npa-nxx [1] VersionNewNPA-NXX-
            Recovery,
            subscription-version-donor-sp-customer-disconnect-date
            [2]
                VersionCustomerDisconnectDateRecovery,
            subscription-version-audit-discrepancy-report [3]
                AuditDiscrepancyRptRecovery,
            subscription-audit-results [4] AuditResultsRecovery,
            lnp-npac-sms-operational-information [5]
            NPAC-SMS-Operational-InformationRecovery,
            subscription-version-new-sp-create-request [6]
                VersionNewSP-CreateRequestRecovery,
            subscription-version-old-sp-concurrence-request [7]
                VersionOldSP-ConcurrenceRequestRecovery,
            subscription-version-old-sp-final-window-expiration [8]
                VersionOldSPFinalConcurrenceWindowExpirationRecovery,
            subscription-version-cancellation-acknowledge-request [9]
                VersionCancellationAcknowledgeRequestRecovery,
            subscriptionVersionStatusAttributeValueChange [10]
                VersionStatusAttributeValueChangeRecovery,
            attribute-value-change [11] AttributeValueChangeInfo,
            object-creation [12] ObjectInfo,
            object-deletion [13] ObjectInfo,
            numberPoolBlockStatusAttributeValueChange [14]
                NumberPoolBlockStatusAttributeValueChangeRecovery,
            subscription-version-new-sp-final-window-expiration [15]
                VersionNewSP-FinalCreateWindowExpirationRecovery
        }
    }
} OPTIONAL
}
```

```
VersionNewSP-FinalCreateWindowExpiration ::= SEQUENCE {  
    version-create-request [0] VersionCreateConcurrencyRequest,  
    new-service-prov-id [1] ServiceProvId,  
    service-prov-old-authorization [2] ServiceProvAuthorization,  
    subscription-status-change-cause-code [3]  
        SubscriptionStatusChangeCauseCode OPTIONAL,  
    access-control [4] LnpAccessControl  
}
```

```
VersionNewSP-FinalCreateWindowExpirationRecovery ::= SEQUENCE {  
    version-create-request [0]  
    VersionCreateConcurrencyRequestRecovery,  
    new-service-prov-id [1] ServiceProvId,  
    service-prov-old-authorization [2] ServiceProvAuthorization,  
    subscription-status-change-cause-code [3]  
        SubscriptionStatusChangeCauseCode OPTIONAL  
}
```

M&P:

No change required.

Origination Date: 8/1/1999

Change Order Number: NANC 294

Description: Changing Due Date Edit Functionality in the NPAC SMS for 7pm on Due Date Problems

Cumulative SP Priority, Weighted Average: 10.50

Pure Backwards Compatible: YES

IMPACT/CHANGE ASSESSMENT

FRS	IIS	GDMO	ASN.1	NPAC	SOA	LSMS
Y	Y	Y		Medium	Medium	N/A

Business Need:

Currently the NPAC SMS operates on GMT (Greenwich Mean Time) and operating business hours (e.g., timers, due dates, etc.) are 7AM – 7 PM. When emergency porting conditions are encountered, the NPAC operating system prohibits the SP from initiating same day porting events due to the difference between GMT and SP time zones creating a variance in the date. There is a manual process requiring conversion of local time to GMT to ensure the dates on the SVs match. This change order would eliminate the manual process for these situations to ensure customers are not out of service any longer than necessary and omissions to large customer porting conversions are quickly resolved.

Description of Change:

Service Providers involved in last minute emergency porting situations, cannot create/concur/activate SVs that are created after 7pm (eastern standard time) on the due date. Since those created after 7pm EST, equate to after midnight GMT the next day on the NPAC SMS, the old SP cannot concur to the port, and the new SP cannot activate at this point in time since timers have not expired.

This problem exists for initial creates as well as concurs, if either one happens after 7pm EST.

Two options were discussed: 1.) change the NPAC SMS to run and store in central time; 2.) change the NPAC SMS edit to allow a concurrence in the past (i.e., local date/time concurrence). It was noted that the first option still has a problem with ports in the western region, west coast region, and Hawaii, albeit the problem window is smaller.

Upon further analysis, option #1 is a huge effort, and does not resolve the issue (it just narrows the window). Option #2 was deemed to be the best solution at this point. However, the local date/time needs to be limited to ensure this functionality does not open the window for “pamming” (port slamming).

Using option #2, a new tunable (“GMT Time Adjustment”) per region would only open the window for local date/time to the largest differential time zone in that region from the NPAC (i.e., from a map perspective, the left most time zone [“prevailing time zone”] in that specific region). The time zone would be adjusted for standard/daylight, and the tunable would have a valid range of 4-10 hours (4 hours is EDT, 10 hours is Hawaiian standard time).

Two tunables may be required to account for both standard time and daylight time. However, this is more of an implementation detail. This local date/time adjustment still allows an SP to send up the date with zeros in the time portion. This will accommodate SPs that always sends all zeros in SV create messages (even though this would be more than the 4-10 hour local date/time range). Discussion and analysis need to determine if the local date/time adjustment should be allows for both Creates or just a concurrence that happens to an existing port, but the current time is after 7pm EST on the due date.

Dec 99 LNPA-WG meeting, the group decided to forego the 29 hour day, the ability to have a due date mismatch, and the GMT Time Adjustment tunable. Instead it was agreed to just allow the concurrence to the port (2nd create) to be sent after the 7pm EST window, and specify the previous day (in NPAC GMT), which is still the current day (in SP’s local time). The due dates still need to match.

Requirements:

Req 1 Subscription Version – Due Date Validation for Second/Concurrence Create Message for a Subscription Version Inter-Service Provider Port

NPAC SMS shall allow the due date to be a past date upon Subscription Version concurrence (2nd create for this Subscription Version) for an Inter-Service Provider port.

IIS:

B.5.1.3 Subscription Version Create by Second SOA (New Service Provider)

In this scenario, the old service provider has already issued its request causing the subscriptionVersionNPAC to be created. The new service provider is now following with its own create action.

Step-by-step message flow text is shown below:

1. New service provider SOA personnel take action to create a new subscription version.
2. New service provider SOA sends M-ACTION subscriptionVersionNewSP-Create to NPAC SMS InpSubscriptions object to create a new subscriptionVersionNPAC. The new service provider SOA must specify the following valid attributes:

subscriptionTN or a valid subscriptionVersionTN-Range
subscriptionNewCurrentSP
subscriptionOldSP
subscriptionNewSP-DueDate (seconds set to zeros)
subscriptionLNPTType
subscriptionPortingToOriginal-SP Switch

The following items must be provided unless subscriptionPortingToOriginal-SP is true:

subscriptionLRN
subscriptionCLASS-DPC
subscriptionCLASS-SSN
subscriptionLIDB-DPC
subscriptionLIDB-SSN
subscriptionCNAM-DPC
subscriptionCNAM-SSN
subscriptionISVM-DPC
subscriptionISVM-SSN
subscriptionWSMSC-DPC - if supported by the Service Provider SOA
subscriptionWSMSC-SSN - if supported by the Service Provider SOA

The following attributes are optional:

subscriptionEndUserLocationValue
subscriptionEndUserLocationType
subscriptionBillingId

If a TN range is specified in the request, it would result in an M-SET request and M-EVENT-REPORT for each TN.

If the new service provider is not the new service provider specified in the initial create by the old service provider, an accessDenied error will be returned.

If any attribute is invalid, an action failure will be returned, indicating invalidArgumentValue. Other appropriate errors will be returned.

If the due date for the port is a previous date, the NPAC SMS accepts a value of a previous date from a service provider, in order to match the

due date of the port that was previously received from the Old Service Provider.

3. If successful, the NPAC SMS sets the subscriptionModifiedTimeStamp, subscriptionCreationTimeStamp, and all data specified in the M-ACTION.
4. NPAC SMS responds to M-SET.
5. NPAC SMS sends M-ACTION reply with success or failure and reasons for failure.
6. NPAC SMS issues the M-EVENT-REPORT with the following attributes to the old service provider when the subscriptionNewSP-DueDate changes value.

subscriptionNewSP-DueDate
subscriptionNewSP-CreationTimeStamp

7. Old service provider SOA issues M-EVENT-REPORT confirmation.
8. If the M-ACTION was successful, the NPAC SMS issues M-EVENT-REPORT to the new service provider for all attributes updated from the preceding list of modifiable attributes in addition to the following:

subscriptionNewSP-DueDate
subscriptionNewSP-CreationTimeStamp

9. New service provider SOA issues M-EVENT-REPORT confirmation.

B.5.1.4 Subscription Version Create by Second SOA (Old Service Provider) with Authorization to Port

In this scenario, the new service provider has already issued its request causing the subscriptionVersionNPAC to be created. The old service provider is now following with its own create action authorizing the port.

Note: This is an optional step.

Step-by-step message flow text is shown below:

1. Old service provider SOA personnel take action to create an old subscription version.
2. Old service provider SOA sends M-ACTION subscriptionVersionOldSP-Create to NPAC SMS InpSubscriptions object to create an old subscriptionVersionNPAC. The old service provider SOA must specify the following valid attributes:

subscriptionTN or a valid subscriptionVersionTN-Range
subscriptionNewCurrentSP
subscriptionOldSP
subscriptionOldSP-Authorization
subscriptionOldSP-DueDate (seconds set to zeros)
subscriptionLNPTType

If a TN range is specified in the request, it would result in an M-SET request and M-EVENT-REPORT for each TN.

If the old service provider is not the old service provider specified in the initial create request by the

new service provider, an accessDenied error will be returned.

If any attribute is invalid, an invalidArgumentValue will be returned, indicating invalid data values. Other appropriate errors will also be returned.

If the due date for the port is a previous date, the NPAC SMS accepts a value of a previous date from a service provider, in order to match the due date of the port that was previously received from the New Service Provider.

3. If the data is valid, the NPAC SMS sets the subscriptionOldSP-AuthorizationTimeStamp, subscriptionModifiedTimeStamp and all data specified in the M-ACTION.
4. NPAC SMS responds to M-SET.
5. NPAC SMS sends M-ACTION reply with success or failure and reasons for failure.
6. If the M-ACTION was successful, the NPAC SMS issues M-EVENT-REPORT attribute value change to the old service provider for all attributes updated from the following list:

- subscriptionOldSP-DueDate
- subscriptionOldSP-Authorization
- subscriptionOldSP-AuthorizationTimeStamp

7. Old service provider SOA issues M-EVENT-REPORT confirmation.
8. If the M-ACTION was successful, the NPAC SMS issues M-EVENT-REPORT attribute value change to the new service provider for all attributes updated from the preceding list. The following attributes are sent in the attributeValueChangeNotification:

- subscriptionOldSP-DueDate
- subscriptionOldSP-Authorization
- subscriptionOldSP-AuthorizationTimeStamp

9. New service provider issues M-EVENT-REPORT confirmation.

GDMO:

-- 21.0 LNP NPAC Subscription Version Managed Object Class

subscriptionVersionNPAC MANAGED OBJECT CLASS

...

subscriptionVersionNPAC-Behavior BEHAVIOUR
DEFINED AS !

...

Upon subscription version creation, the subscriptionOldSP-DueDate and subscriptionNewSP-DueDate must match. **If the due date for the port is a previous date, the NPAC SMS accepts a value of a previous date from a service provider, in order to match the due date of the port that was previously received from the other Service Provider (new or old).**

ASN.1:

No change required.

M&P:

No change required.

Origination Date: March 20, 2001

Originator: Wireless Operations Subcommittee

Change Order Number: NANC 328

Description: Tunable for Long and Short Business Days

Cumulative SP Priority, Weighted Average:

Pure Backwards Compatible: Yes

IMPACT/CHANGE ASSESSMENT

FRS	IIS	GDMO	ASN.1	NPAC	SOA	LSMS
Y				Low	N/A	N/A

Business Need: Currently, per RR3-30 in the FRS, the NPAC SMS has Long Business Days defined as Monday through Saturday excluding NPAC operations-defined holidays. This means that short timers only run Monday through Saturday. Wireless Service Providers need short timers to run on Sundays as well so they can port in a 2½-hour window on all days of the week. To meet this need Long Business Days need to be Monday through Sunday.

April 2001 LNPA WG Meeting: AT& T Broadband requested that the change order include making Short Business Days a tunable also. The LNPA WG accepted this request.

Description of Change: Wireless Service Providers are requesting that Long Business Days be defined as a tunable with a default value of Monday through Sunday.

April 2001 LNPA WG Meeting: The change order will be revised to make both Long and Short Business Days a tunable.

Requirements:

Req 1 Long Business Days Tunable Parameter

NPAC SMS shall provide a Long Business Days tunable parameter that defines the days of the week that are valid for operations involving business time calculation excluding NPAC operations-defined holidays.

Req 2 Long Business Days Tunable Parameter – Default Value

NPAC SMS shall default the Long Business Days tunable parameter to Monday through Sunday.

Req 3 Long Business Days Tunable Parameter – Valid Values

NPAC SMS shall use days of the week as valid values for the Long Business Days tunable parameter.

Req 4 Short Business Days Tunable Parameter

NPAC SMS shall provide a Short Business Days tunable parameter that defines the days of the week that are valid for operations involving business time calculation excluding NPAC operations-defined holidays.

Req 5 Short Business Days Tunable Parameter – Default Value

NPAC SMS shall default the Short Business Days tunable parameter to Monday through Friday.

Req 6 Short Business Days Tunable Parameter – Valid Values

NPAC SMS shall use days of the week as valid values for the Short Business Days tunable parameter.

IIS

No change required

GDMO

No change required

ASN.1

No change required

M&P

TBD

Origination Date: 06/13/2001

Originator: NeuStar

Change Order Number: NANC 329

Description: Prioritization of SOA Notifications

Pure Backwards Compatible: Yes

IMPACT/CHANGE ASSESSMENT

FRS	IIS	GDMO	ASN.1	NPAC	SOA	LSMS
Y	Y	Y	N	Medium	N/A	N/A

Business Need: With the deployment of the NPAC Release 3.0 in the Northeast region a SOA – NPAC Interface problem has surfaced. The improved performance of NPAC Release 3.0 and the faster hardware platform that this software is running on is resulting in transactions being processed for broadcast to the industry quicker than the SOA – NPAC interface can transmit them. ~~The SOA – NPAC interface has a specification of 2 CMIP transactions per second (sustained) and 5.2 CMIP transactions per second (peak).~~ During peak periods the interface cannot support the volumes of notifications that the NPAC SMS is generating, thus there is a long delay in notification delivery that results in operational issues. At the current time it is the ILEC that is primarily affected by this problem because the ILEC receives the largest volume of SOA notifications but the problem has the potential of affecting any Service Provider. The NAPM, LLC has decided not to go forward with the deployment of NPAC Release 3.0 until this interface problem has been ~~resolved~~mitigated. NeuStar is proposing that SOA Notifications be prioritized and transmitted over the interface based on priority. This would allow for more timely delivery of Service Providers high priority notifications.

Description of Change: Currently SOA notifications are not prioritized so they are generated by the NPAC SMS and then transmitted on a ‘first in, first out’ basis. During a large porting volume peak this model can produce major delays in the transmission of notifications through the SOA – NPAC interface, resulting in operational issues. This change order would prioritize SOA notifications and allow requests and notifications with the highest priority to be transmitted first. The SOA notifications would have ~~five~~four categories: ~~very high, high, medium, low, and none.~~ ~~The category of very high would be reserved for Subscription Version Object Creation notifications with a due date less than or equal to today.~~ The category of **none** would indicate that a Service Provider did not want to receive a particular notification. One of the other three categories would be assigned to each notification on a per region basis. A Service Provider would have the option of overriding the default value.

Requirements:

Req 1 SOA Notification Priority Tunable Parameter

NPAC SMS shall provide a SOA Notification Priority tunable parameter for each SOA notification that defines the priority of the SOA notification for the given region.

Req 2 SOA Notification Priority Based on Attributes

NPAC SMS shall allow SOA Notifications to have separate priorities associated with the value of certain attributes based on the information contained in Appendix C, Table C-7 – SOA Notification Priority Tunables.

Note: The table referenced above is new and is appended to this document.

Req 3 SOA Notification Priority Tunable Parameter – Valid Values

NPAC SMS shall use ~~VERY HIGH~~, HIGH, MEDIUM, LOW, and NONE as valid values for the SOA Notification Priority tunable parameters.

Req 4 SOA Notification Priority Tunable Parameter – Default Value

NPAC SMS shall default the SOA Notification Priority tunable parameters to the values specified in Appendix C, Table C-7 – SOA Notification Priority Tunables.

Req 5 Modifying the SOA Notification Priority Tunable Parameter Value

NPAC SMS shall allow NPAC Personnel to modify the SOA Notification Priority tunable parameter values based on Service Provider requests.

~~**Req 6 SOA Notification Priority Tunable Parameter Value for Notifications Associated with Subscription Versions**~~

~~NPAC SMS shall set the SOA Notification Priority tunable parameter value to **VERY HIGH** for Subscription Version Object Creation notifications with a due date less than or equal to the current system date.~~

RR6-30 (Modified) Notification Recovery – Order of Recovery

NPAC SMS shall recover all notifications, failed or successful, in *the* order ~~they~~ *were the NPAC SMS attempted to sent-send them* when notification recovery is requested by the SOA or LSMS.

~~**Note: The SOA Notification Priority tunable parameter values will be ignored during notification recovery.**~~

Req 7 SOA Notification Priority Tunable Parameter based on Old or New Service Provider Status

NPAC SMS shall allow different SOA Notification Priority values for Status Attribute Value Change notifications based on whether the Service Provider is acting as the Old Service Provider or as New Service Provider for the port [as indicated in Appendix C, Table C-7 – SOA Notification Priority Tunables](#).

Req 98 SOA Notification Priority Tunable Parameter –Value Equal to NONE

NPAC SMS shall use the SOA Notification Priority tunable parameter equal to **NONE** to indicate that the notification is ~~not sent to~~[generated for](#) that Service Provider.

Req 109 Processing of SOA Notification Queues

NPAC SMS shall use the SOA notification priority when determining the order to send notifications to a Service Provider.

R4-8 (Modified) Service Provider Data Elements

NPAC SMS shall require the following data if there is no existing Service Provider data:

4. Service Provider name, address, phone number, and contact organization.
5. NPAC customer type.
6. Service Provider allowable functions.
 -
 -
 -
18. *SOA Notification Priority for each SOA notification. Separate values may be set for Status Attribute Value Change notifications based on whether the Service Provider is acting as the Old Service Provider or as the New Service Provider for the port [as indicated in Appendix C, Table C-7 – SOA Notification Priority Tunables](#).*

IIS

Need to add some notes to section 2.3.3 explaining the ~~prioritisation~~[prioritization](#) of SOA notifications.

GDMO

For each notification, we should add a note to the description text that indicates the notification is prioritised and transmitted according to the SOA Notification Priority tunable ~~and its Service Priority Profile SOA Notification Priority tunable~~.

ASN.1

No change required.

M&P

| ~~TBD~~ [Add the SOA Notification Priority Tunables table to Appendix I.](#)

SOA Notification Priority Tunables

#	Notification Name	Priority
L-1.0	NPAC SMS Operational Information Notification	LOW
L-2.0	Subscription Audit Discrepancy Report	LOW
L-3.0	Subscription Audit Results	LOW
L-4.0 A	Subscription Version Cancellation Acknowledge Request Scenario A: the OLD SP is requesting cancellation and no concurrence from New SP.	LOW
L-4.0 B	Subscription Version Cancellation Acknowledge Request Scenario B: the New SP is requesting cancellation and no concurrence from Old SP	LOW
L-6.0	Subscription Version - Donor SP - Customer Disconnect Date Notification	LOW
L-7.0	Subscription Version Local SMS Action Results	N/A
L-8.0	Subscription Version New NPA-NXX Notification	LOW (to SOA)
L-9.0	Subscription Version New SP Create Request Notification (T1 timer expiration for New SP concurrence)	LOW
L-10.0	Subscription Version Old SP Concurrence Request Notification (T1 timer expiration for Old SP concurrence)	LOW
L-11.0 A1	Subscription Version Status Attribute Value Change Notification - Activates When an INTER or INTRA SV has been created in the Local SMSs (or 'activated' by the SOA) and the SV status has been set to: <i>Active</i> or <i>Partial-Failure</i> . The notification is sent to both SOAs: Old and New. If the status has been set to <i>Partial-Failure</i> , this notification contains the list of Service Providers (SP) LSMSSs that have failed to receive the broadcast. Note: See L-11.0 E for Deletes and L-11.0 F for Modify Actives	LOW (to new SOA) LOW (to Old SOA)
L-11.0 A2	Subscription Version Status Attribute Value Change Notification - re-sends to fail list Intermediate Notification for Partial Failure. Every time one of these SPs is removed from the Failed SP-List, the NPAC re-sends the notification to both SOAs. This iteration happens until the last SP is cleared from the fail-list.	LOW (to Old SP) LOW (to New SP)
L-11.0 A3	Subscription Version Status Attribute Value Change Notification - clear Fail List Final Notification associated with a Partial Failure. Upon cleaning-clearing the Failed SP-FailHL list, the NPAC sends the same notification to	LOW (to New SP) LOW (to Old

	both SOAs but with an SV status of <i>active</i> and empty fail-list.	SP)
L-11.0 B	Subscription Version Status Attribute Value Change Notification - total failure When an SV has failed to be created (or ‘activated’) in ALL LSMSs and the SV status has been set to <i>Failed</i> . The notification is sent to both SOAs: Old and New.	LOW
<u>L-11.0 C</u>	<u>DELETED</u>	
L-11.0 D1	Subscription Version Status Attribute Value Change Notification - re-sends When the NPAC attempts to re-sends <u>Creates (or ‘activates’)/Modifies/Modify Active or Deletes</u> to the LSMSs for SVs with statuses of <i>Partial-Failure, Failed</i> <u><i>Active</i></u> or <i>Old</i> , and with a Fail SP List (the notification is sent regardless <u>of the final status of the SV</u>) and to the <u>The notification is sent to the Current (New) SOA, corresponding SOA/s involved in the requested operation).</u>	Same priorities as for 11.0 Ax scenarios <u>LOW</u>
L-11.0 E	Subscription Version Status Attribute Value Change Notification - set to OLD When the SV status has been set to <i>old</i> . (Port to Original, port-of-a port, port to original of a Pool TN (or snap back), disconnect, disconnect of a ported Pool TN). The notification is received only by those SOAs that actually have the SV in their local DB. It varies with the scenario. <u>Note: See L-11.0 A1 for Activates and L-11.0 F for Modify Actives</u>	LOW
L-11.0 F	Subscription Version Status Attribute Value Change Notification - Modify active When an <i>Active</i> SV has been modified in the LSMS and the status of the SV has been re-set to <i>Active</i> (with or without a Fail-SP-List). The notification is sent only to the current SOA. <u>Note: See L-11.0 A1 for Activates and L-11.0 E for Deletes</u>	LOW
L-11.0 G	Subscription Version Status Attribute Value Change Notification - cancel pending When a <i>Pending</i> SV has been cancelled by the Old SP and the NPAC SMS has set the SV status to <i>Cancel-Pending</i> . The notification is sent to both SOAs: Old and New.	LOW
L-11.0 H1	Subscription Version Status Attribute Value Change Notification - cancel When the NPAC SMS has set the status of a <i>pending, cancel-pending, or conflict</i> SV to <i>CANCEL</i> after: 1) concurrence and cancellation acknowledgment by both SOAs has been received in the NPAC	LOW
L-11.0 H2	Subscription Version Status Attribute Value Change Notification - cancel When the NPAC SMS has set the status of a <i>pending, cancel-pending, or conflict</i> SV to <i>CANCEL</i> after: 2) expiration of a tunable period without cancellation acknowledgment by one of the SOAs or	LOW
L-11.0	Subscription Version Status Attribute Value Change	LOW

H3	Notification - cancel When the NPAC SMS has set the status of a <i>pending, cancel-pending, or conflict</i> SV to <i>CANCEL</i> after: 3) cancellation request by the originating SOA with no concurrence from the other SOA. (Only one create action has been received in the NPAC and the same provider sends the cancellation request before the second provider send a create request.)	
L-11.0 H4	Subscription Version Status Attribute Value Change Notification - cancel When the NPAC SMS has set the status of a <i>pending, cancel-pending, or conflict</i> SV to <i>CANCEL</i> after: 4) the Conflict Cancellation Window expires, if no resolution has been reached for the conflict, the NPAC automatically cancels the <i>Conflict</i> SV.	LOW
L-11.0 I	Subscription Version Status Attribute Value Change Notification - Disconnect pending When an <i>active</i> SV is being disconnected with an Effective Release Date in the NPAC and the SV status is set to <i>Disconnect-Pending</i> . Only the current SOA receives the notification.	LOW
L-11.0 J	Subscription Version Status Attribute Value Change Notification - Fail disconnect When the NPAC attempts to delete an <i>Active</i> SV and the request fails to ALL LSMSs and the SV status is re-set to <i>Active</i> . Only the Current SOA receives the notification.	LOW
L-11.0 K	Subscription Version Status Attribute Value Change Notification - Conflict When the status of a <i>Pending</i> SV is set to <i>conflict</i> . The notification is sent to both SOAs: Old and New.	LOW
L-11.0 L	Subscription Version Status Attribute Value Change Notification After Conflict Resolution, when the status of the <i>Conflict</i> SV is re-set to <i>Pending</i> . The notification is sent to both SOAs: Old and New.	LOW
L-12.0	Subscription Version Old SP Final Concurrence Timer Expiration Notification (T2 expiration for Old SP concurrence)	LOW
L-13.0 A	Number Pool Block Status Attribute Value Change Notification The Pool Block has being created in the LSMSs (EDR and Non_EDR) and the Block Status has being set to Active or Partial Failure;	LOW
L-13.0 B	Number Pool Block Status Attribute Value Change Notification The creation of the Pool Block has failed in all the LSMSs (EDR and Non-EDR) and the Block Status has been set to Failed.	LOW
L-13.0 C	Number Pool Block Status Attribute Value Change Notification The NPAC attempts to re-send a failed Pool Block or failed SVs to SP in the fail-SP-List of the Block and the Block status changes to Active, Partial Failure or Failure.	LOW
L-13.0	Number Pool Block Status Attribute Value Change Notification	LOW

D	The attributes in the Pool Block have been modified in the LSMSs (EDR and Non-EDR) and the Block Status has been re-set to Active (with or without fail-sp-list).	
L-13.0 E	Number Pool Block Status Attribute Value Change Notification When a Pool Block has been ‘de-pooled’ from the LSMSs (EDR and Non-EDR) and the Block Status has been set to Old (with or without fail-sp-list).	LOW
L-13.0 F	Number Pool Block Status Attribute Value Change Notification When the NPAC SMS has attempted to ‘de-pool’ a block but the request has failed to ALL LSMSs (EDR and Non-EDR) and the Block Status has been reset to Active with a Failed-SP-list.	LOW
L-14.0	Subscription Version New SP Final Create Window Expiration Notification <u>New notification to be implemented with CO NANC 240.</u> It will be sent after T2 expiration to both SPs SOAs (old and new) to inform them that the T2 timer has expired and the new SP hasn’t send its create request yet. The SV will remain in status of Pending until the New SP sends the Create or the NPAC cancels it.	LOW
S-1.00	Object Creation	LOW
S-2.00	Object Deletion	LOW
S-3.00 A	Attribute Value Change For pending SVs	LOW
S-3.00 B	Attribute Value Change For Pool Blocks	LOW