

Content

1	Group Termination Points and Bundled SNCs.....	1
1.1	Rationale.....	1
1.2	GTP Definition and Usage.....	1
1.2.1	GTP Creation, Modification and Deletion.....	1
1.3	Bundled SNC Service Definition and Usage.....	2
1.3.1	Bundled SNC Service Creation and Deletion.....	2
1.3.2	Applicability of Other SNC-Related Operations to Bundled SNC Service.....	4
1.3.2.1	Operations in the ManagedElementMgr Interface.....	4
1.3.2.2	Operations in the MultLayerSubnetworkMgr Interface.....	4
1.3.2.3	Relationship to v3.0 Operations.....	5

1 Group Termination Points and Bundled SNCs

1.1 Rationale

Bundled SNC service involves the establishment of a bundle of connections (i.e., not necessarily of the same layerRate) from one point in a subnetwork to another. The bundled SNC service is treated as a single SNC. Currently, the MTNM interface does not support an efficient way to establish a bundle of SNCs. Using the capabilities available in MTNM Version 2.1, the NMS would need to make individual SNC requests for each SNC in the bundle, and might also need to provide routing constraints to request that the component SNCs follow the same route. This becomes impractical for large bundles, e.g., a bundle of 48 STS-1s carried in the form of an STS-48 (un-concatenated).

1.2 GTP Definition and Usage

A *Group Termination Point* (GTP) is a sequence of CTPs (with a specific order) in the same ME. Further details and requirements concerning GTPs can be found in TMF 513. Information modeling concepts concerning GTPs can be found in TMF 608.

1.2.1 GTP Creation, Modification and Deletion

The NMS create may create a GTP in two different ways

- by specifying the list of CTPs comprising the GTP (the list is placed in the listOfTPs parameter of the createGTP operation) or,
- in the case of contiguous CTPs of the same layerRate, the NMS may list the first CTP (in the initialCTPname parameter) and the number of following CTPs (in the numberOfCTPs parameter).

The NMS may also provide the EMS with the flexibility to decide the list of CTPs by using the gtpEffort parameter. In this flexible mode, the EMS uses the listOfTPs parameter or the initialCTPname/numberOfCTPs parameters (provided by the NMS) to determine the total requested bandwidth but it (the EMS) may return a GTP (newGTP parameter) with a different set of CTPs than those indicated in the original createGTP request. Note that this mode allows for the GTP components to be instantiated at a later time by the ME (e.g., upon detection of user's signal). Therefore the operation may successfully return a newGTP with an empty listOfTPs attribute (to be updated at a later time once the component CTPs are created in the ME).

The EMS may also create GTPs and inform the NMS via an object creation notification.

Using the modifyGTP operation, the NMS can add or delete CTPs, not both. It is not possible to add a CTP that is already involved in a cross connection or SNC, or that is part of another GTP. Attempts to modify a GTP that is involved in a cross connection (or SNC) should be rejected by the EMS. The operation is best-effort, i.e., the EMS will add or delete as many of the identified CTPs as possible. If the service is called with the name of a non-existent GTP or CTP, it will fail.

If the NMS created a GTP with gtpEffort equal to EFFORT_SAME, this also implies that the EMS will not modify on its own, i.e., the EMS will only modify the GTP if requested by the NMS via the modifyGTP operation.

The EMS may modify a GTP (by adding or deleting CTPs) and then inform the NMS via an AVC notification. As was the case for NMS requested modification of GTPs, the EMS should not add a CTP that is already involved in a cross connection or SNC, or that is part of another GTP.

The NMS can delete a GTP either directly via the deleteGTP operation or indirectly when deleting a Bundled SNC service. The latter approach is described in Section 1.3.2.2. The EMS may also delete GTPs and then inform the NMS via an object deletion notification. In any case, a GTP should not be deleted if it is being used in support of a Bundled SNC service.

1.3 Bundled SNC Service Definition and Usage

A **bundled SNC service** is comprised of a collection of cross connected GTPs. Figure 1 depicts a Bundled SNC service (see the dotted line). Note that CTP ordering is preserved between the aEnd and zEnd, i.e., the i^{th} CTP in the left GTP in Managed Element #1 is mapped to the i^{th} CTP in the right GTP in Managed Element #3. For a Bundled SNC service spanning a non-singleton subnetwork, the aEnd and zEnd attributes are GTPs such that the CTPs comprising each GTP are contained in the same PTP or FTP.

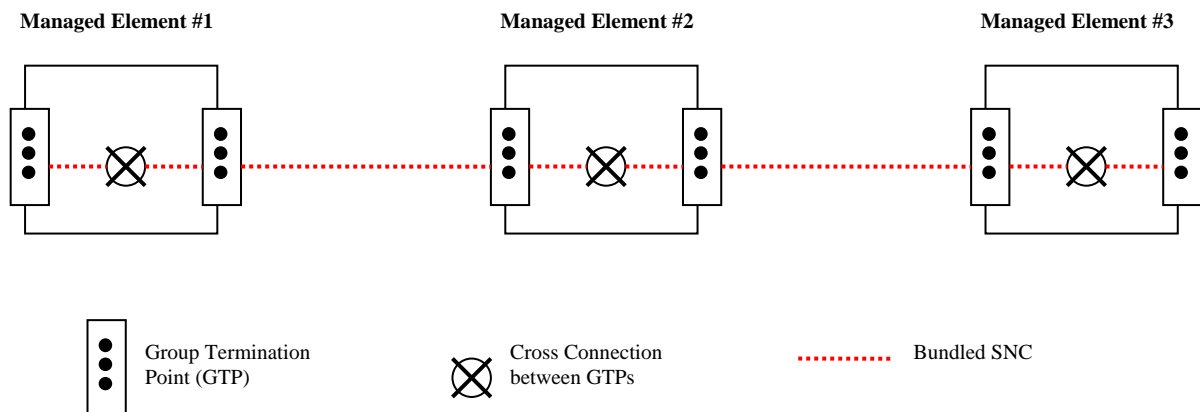


Figure 1. Bundled SNC service

Further details and requirements concerning Bundled SNC service can be found in TMF 513. Information modeling concepts concerning Bundled SNC service can be found in TMF 608.

1.3.1 Bundled SNC Service Creation and Deletion

The SNC establishment operations (createSNC, activateSNC, and createAndActivateSNC) are used to request the creation and activation of a Bundled SNC service. The parameters of the createAndActivate operation would be populated as shown in Table 1. The NMS would only get a response from the EMS concerning the Bundled SNC service and would not get any information about the component SNCs in the connection establishment response. Alarms and notifications would be issued against the Bundled SNC service.

Table 1. Usage of createAndActivateSNC for Bundled SNC service Establishment

Parameter	Usage
createData	
<i>userLabel</i>	Same usage as “regular” SNC establishment
<i>forceUniqueness</i>	
<i>owner</i>	
<i>direction</i>	
<i>staticProtectionLevel</i>	
<i>protectionEffort</i>	
<i>rerouteAllowed</i>	
<i>networkRouted</i>	
<i>sncType</i>	Whatever can be supported by the network.
<i>layerRate</i>	Use LR_Not_Applicable. The EMS determines the layerRate(s) by looking at the CTPs contained in the aEnd and zEnd GTPs.

Parameter	Usage
<i>ccInclusions</i>	Same usage as “regular” SNC establishment, except that the CCs are between GTPs .
<i>neTpInclusions</i>	Works similar to “regular” SNC neTpInclusions with the following exceptions: <ol style="list-style-type: none"> 1. The types of resources that can be used in the inclusion list should be expanded to allow for GTPs, 2. For each PTP used in the inclusion list, the EMS should use a GTP that contains at least one CTP from the given PTP, 3. For each CTP used in the inclusion list, the EMS should use a GTP that contains the given CTP.
<i>fullRoute</i>	Same usage as “regular” SNC establishment
<i>neTpSncExclusions</i>	Works similar to “regular” SNC neTpSncExclusions with the following exceptions: <ol style="list-style-type: none"> 1. The types of resources that can be used in the exclusion list should be expanded to allow for GTPs 4. For each PTP used in the exclusion list, the EMS should not use any GTP that contains any CTP from the given PTP, 2. For each CTP used in the exclusion list, the EMS should not use any GTP that contains the given CTP.
<i>aEnd</i>	Use a GTP rather than a CTP
<i>zEnd</i>	Use a GTP rather than a CTP
<i>additionalCreationInfo</i>	It is recommended that a bundledSNCIndicator be added. This parameter would be a boolean.
<i>tolerableImpact</i>	Same usage as “regular” SNC establishment
<i>emsFreedomLevel</i>	
<i>tpsToModify</i>	
<i>theSNC</i>	See Table 2
<i>errorReason</i>	

Table 2. Usage of SNC Structure for Bundled SNC service

Parameter	Usage
name	Name of Bundled SNC service
userLabel	Same usage as “regular” SNC establishment
nativeEMSName	
owner	
sncState	
direction	
rate	Use LR_Not_Applicable
staticProtectionLevel	Same usage as “regular” SNC establishment
sncType	No restriction (whatever the equipment can support)
aEnd	A GTP list rather than a CTP list is used. Note that TPData_T is used to represent an endpoint of an SNC. In the case of a GTP, only the tpName and transmissionParams apply. The tpMappingMode should be set to TM_NA, and traffic descriptors should not be used.
zEnd	A GTP list rather than a CTP list is used.
rerouteAllowed	Same usage as “regular” SNC establishment when such a capability is supported for Bundled SNC services
networkRouted	
	It should be noted that current control-plane signaling protocol do not support GTPs.

Parameter	Usage
additionalInfo	It is recommended that a bundledSNCIndicator be added to additionalInfo. This parameter would be a boolean.

1.3.2 Applicability of Other SNC-Related Operations to Bundled SNC Service

1.3.2.1 Operations in the ManagedElementMgr Interface

- [getAllCrossConnections](#) – the operation should also return cross connections between GTPs.
- [getContainedCurrentTPNames](#), [getContainedCurrentTPs](#), [getContainedInUseTPNames](#), [getContainedInUseTPs](#), [getContainedPotentialTPNames](#), [getContainedPotentialTPs](#) – these operations apply to TP and **not** GTPs.
- [getContainingTPNames](#), [getContainingTPs](#) – these operations apply to TP and **not** GTPs.
- [getTP](#) – this operation applies to TP and **not** GTPs, a new operation needs to be defined for the retrieval of GTPs (see the [getGTP](#) operation defined later in this contribution).
- [setTPData](#) – this operation applies to TP and **not** GTPs.

1.3.2.2 Operations in the MultLayerSubnetworkMgr Interface

The following operations are valid for Bundled SNC service: [checkValidSNC](#), [activateSNC](#), [createSNC](#), [deactivateAndDeleteSNC](#), [deactivateSNC](#) and [deleteSNC](#).

Concerning the deletion of a Bundled SNC service: A new attribute has been added to the additionalInfo of the SNC struct (to be named mustRemoveGTPs). The attribute is carried in the additionalCrationInfo parameter of the various SNC establishment operations. The attribute is only used for Bundled SNC service.

MustRemoveGTPs takes on two possible values, i.e., YES (the EMS must make a best-effort attempt to remove the interior GTPs that once belonged to the SNC) and NO (the NMS doesn't care if the EMS removes the interior GTPs or not). In the case that mustRemoveGTPs is set to "YES" and the EMS cannot remove all the interior GTPs, the EMS should return a list of the GTPs that could not be removed in the errorReason field of the deactivateAndDeleteSNC and deactivateSNC operations. The errorReason would be "not all interior GTPs could be deleted" – this would be followed by the names of the GTPs that could not be deleted.

- [getAllSubnetworkConnections](#), [getAllSubnetworkConnectionNames](#) – these operations should also return Bundled SNC services.
- [getAllSubnetworkConnectionsWithTP](#), [getAllSubnetworkConnectionNamesWithTP](#) – these operations also apply to GTPs and Bundled SNC services. This is an example of an operation that is not semantically backward compatible. Further, the following should be noted:
 - This operation also works if a GTP is supplied rather than a CTP. In this case, the layerRate is set to LR_Not_Applicable. If the GTP is part of a cross connection, the EMS returns the set of Bundled SNC services that use the provided GTP as either an end point or an interior point.
 - As input to the operation the NMS may use a CTP that is associated with a GTP. In this case, the EMS returns the set of Bundled SNC services that use the containing GTP as either an end point or an interior point.
- [getAssociatedTP](#) – this operation is **not** applicable to GTPs.
- [getRoute](#) – this operation also works for Bundled SNC services. In the case of a Bundled SNC service, the EMS returns a set of cross connections between GTPs."
- [getSNC](#) – this operation is applicable to Bundled SNC services.
- [getSNCsByUserLabel](#) – this operation is applicable to Bundled SNC services.

1.3.2.3 Relationship to v3.0 Operations

- `addRoute` – the relationship between Bundled SNC service and the SNC with Multiple Routes feature requires further study and is not applicable for v3.0.
- `createModifiedSNC` – this operation is applicable to Bundled SNC services.
- `getAllFixedSubnetworkConnectionNames` – this operation is applicable to Bundled SNC services.
- `getAllFixedSubnetworkConnectionNamesWithTP` – this operation is applicable to Bundled SNC services.
- `getAllFixedSubnetworkConnections` – this operation is applicable to Bundled SNC services.
- `getAllFixedSubnetworkConnectionsWithTP` – this operation is applicable to Bundled SNC services.
- `getBackupRoutes` – the relationship between Bundled SNC service and the SNC with Multiple Routes feature requires further study and is not applicable for v3.0.
- `getRouteAndTopologicalLinks` – this operation is applicable to Bundled SNC services.
- `modifySNC` – this operation is applicable to Bundled SNC services.
- `removeRoute` – the relationship between Bundled SNC service and the SNC with Multiple Routes feature requires further study and is not applicable for v3.0.
- `setIntendedRoute` – the relationship between Bundled SNC service and the SNC with Multiple Routes feature requires further study and is not applicable for v3.0.
- `setRoutesAdminState` – the relationship between Bundled SNC service and the SNC with Multiple Routes feature requires further study and is not applicable for v3.0.
- `swapSNC` – this operation is applicable to Bundled SNC services.
- `switchRoute` – the relationship between Bundled SNC service and the SNC with Multiple Routes feature requires further study and is not applicable for v3.0.

Revision History

Version	Date	Description of Change
3.0	June 2005	Conversion of BundledSNC into new template.

Acknowledgements

<FirstName>	<LastName>	<Company>

How to comment on the document

Comments and requests for information must be in written form and addressed to the contact identified below:

Keith	Dorking	CIENA
Phone:	+1 678 867 5007	
Fax:	+1 678 867 5010	
e-mail:	Kdorking@ciena.com	

Please be specific, since your comments will be dealt with by the team evaluating numerous inputs and trying to produce a single text. Thus we appreciate significant specific input. We are looking for more input than wordsmith" items, however editing and structural help are greatly appreciated where better clarity is the result.