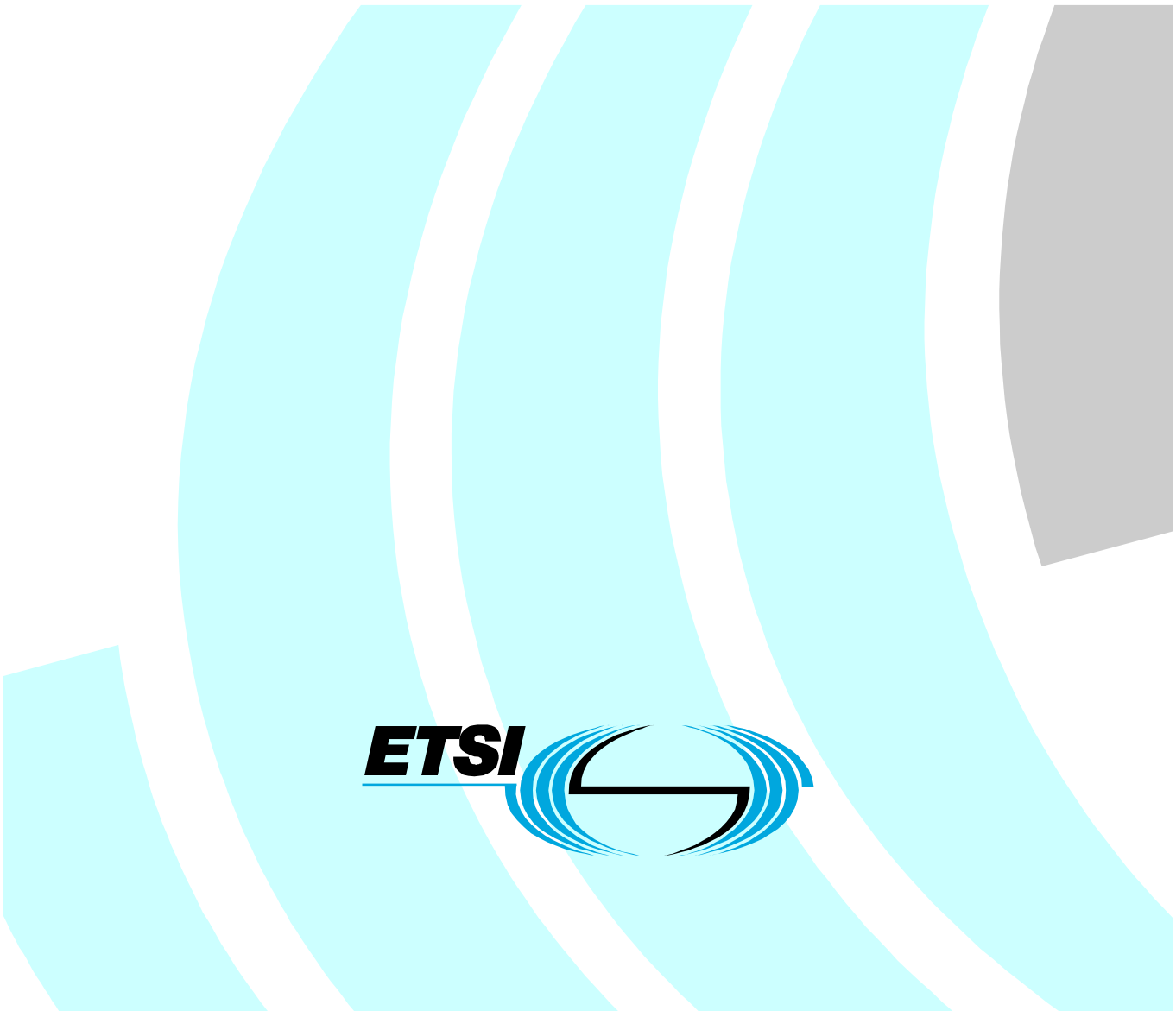
ETSI TS 101 882-5 V4.1.1 (2003-11)

Technical Specification

Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Protocol Framework Definition; Part 5: Transport control service



Reference
DTS/TIPHON-03016-5R4

Keywords
control, interface, IP, meta-protocol,
protocol, transport

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI Project Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON).

The present document is part 5 of a multi-part deliverable. Full details of the entire series can be found in part 1 [1].

Introduction

The present document is a product in TIPHON Release 4 (see TR 101 301) of step C of the TIPHON development process described in TR 101 835.

1 Scope

The present document defines the stage 1 and stage 2 (as defined by ITU-T Recommendation I.130 [6]) requirements for the transport control service.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

[1] ETSI TS 101 882-1: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Protocol Framework Definition; Part 1: Meta-protocol design rules, development method, and mapping guideline".

[2] ETSI TS 101 314: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Abstract Architecture and Reference Points Definition; Network Architecture and Reference Points".

[3] ETSI TS 101 878: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Service Capability Definition; Service Capabilities for TIPHON Release 4".

[4] ITU-T Recommendation Z.100: "Specification and description language (SDL) with corrigendum 1".

[5] ITU-T Recommendation X.680: "Information technology - Open Systems Interconnection - Abstract Syntax Notation One (ASN.1): Specification of basic notation".

[6] ITU-T Recommendation I.130: "Integrated Services Digital Network (ISDN); Method for the characterization of telecommunications services supported by an ISDN and network capabilities of an ISDN".

[7] ETSI TR 101 301: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 3; Release Definition; TIPHON Release 3 Definition".

[8] ETSI TR 101 835. "Telecommunications and Internet Protocol Harmonization over Networks (TIPHON); Project method definition".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 101 878 [3] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TS 101 878 [3] and the following apply:

ASN.1	Abstract Syntax Notation 1
ETF	Egress Transport Flow
FE	Functional Entity
ITF	Ingress Transport Flow
MSC	Message Sequence Chart
PDU	Protocol Data Unit
QoS	Quality of Service
SDL	Specification and Description Language
TC	Transport Control
TFE	Transport Functional Entity
TRM	Transport Resource Management
TU	Transport User
UML	Unified Modelling Language

4 Transport control service

4.1 Purpose

The Transport Control (TC) service provides a means of reserving, assigning and releasing specific transport capabilities to control inter-domain transport connections.

4.2 Description

The Transport Control service establishes the transport capabilities to support the IP Telephony Application plane. The TC service allows the TC service user to reserve and allocate transport resources, fulfilling specific QoS requirements and thereby establish QoS specific inter-domain transport connections. Release of allocated transport resources and connections is also controlled via the TC service.

4.3 Procedures

4.3.1 Provision/withdrawal

The transport control service shall be available to all transport service users in a TIPHON system.

4.3.2 Normal procedures

4.3.2.1 Activation/deactivation

The transport control service shall be permanently activated.

4.3.2.2 Invocation and operation

The TC service shall be invoked by a transport user agent requesting reservation of a transport resource with specific transport capabilities.

Allocated transport resources shall be released upon request from the transport user.

4.3.3 Exceptional procedures

If a transport resource reservation or activation request fails the transport user agent shall be notified. The following causes of failing to reserve requested transport resources may be reported:

- transport resource not available;
- requested transport resource not supported;
- previous or next domain connection address can not be identified.

The activation of reserved transport resource fails if the resources have been released due to reservation time expiration.

4.4 Interaction with other services or service capabilities

In the transport layer domain TIPHON Release 4 defines the following additional service capabilities.

4.4.1 Media control service

No interaction.

4.5 Service capabilities used in service definition

Although not explicitly identified, aspects of the following services and service capabilities are used in definition of the transport control service:

- simple call service.

The TIPHON Release 4 service capabilities are defined in [3].

4.6 Overall behaviour

The UML activity diagram in figure 1 shows the dynamic transport service signalling for a TIPHON system providing transport control service.

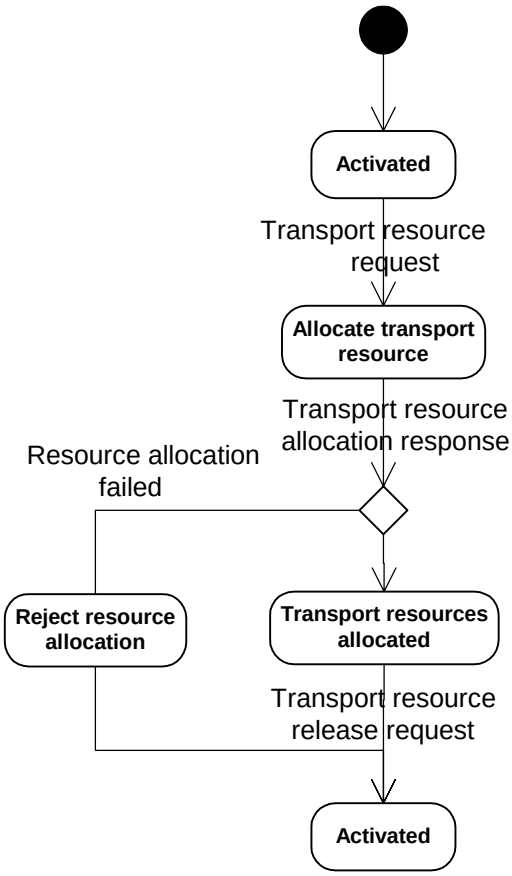


Figure 1: Overall behaviour of transport control service signalling

5 Functional entity model and information flows

5.1 Functional entity model

5.1.1 Description of model

The functional model of the transport control service shall comprise of the following transport control service functional entities:

- TU the Transport User that instigates the transport service request;
- $TFE1_{TRM}$ the Transport Resource Management functional entity;
- $TFE2_{ITF}$ the Transport Flow functional entity handling ingress connection requests;
- $TFE3_{ETF}$ the Transport Flow functional entity handling egress connection requests.

The following functional relationships shall exist between these MFEs:

- ra between the TU and the transport resource management functional entity ($TFE1_{TRM}$);
- rb between the transport resource management functional entity ($TFE1_{TRM}$) and the ingress transport flow functional entity ($TFE2_{ITF}$);
- rc between the transport resource management functional entity ($TFE1_{TRM}$) and the egress transport flow functional entity ($TFE4_{ETF}$).

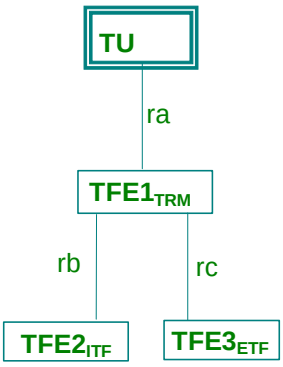


Figure 2: Transport control service functional entity model

5.1.2 Description of functional entities

5.1.2.1 Transport User

The TU acts on behalf of a media control entity to request or respond to reservation, allocation, or release of transport capabilities.

5.1.2.2 $TFE1_{TRM}$

The transport resource management functional entity controls reservation, allocation, and release of transport resources based on local state information.

5.1.2.3 $TFE2_{ITF}$

The transport flow functional entity controlling the ingress points of the transport domain.

5.1.2.4 $TFE3_{ETF}$

The transport flow functional entity controlling the egress points of the transport domain.

5.2 Information flows

5.2.1 Definition of information flows

NOTE: In the tables within this clause, the following convention is used in the "value" columns. Un-bulleted lists of values indicate that all items in the list are included in the associated information element; bulleted lists of values indicate that only one item in the list is included in the information element.

5.2.1.1 Relationship ra

5.2.1.1.1 TransportReserve

`TransportReserve` is a confirmed information flow that shall be sent across relationship ra from the transport user to $TFE1_{TRM}$ to reserve transport resource. Table 1 lists the information elements in the `TransportReserve` information flow.

Table 1: Contents of `TransportReserve`

TransportReserve			
Information element	**Value**	**Request**	**Response**
BearerIdentifier	Alphanumeric "handle"	M	
Transport QoS parameters	Maximum delay, Maximum packet delay variation, Maximum mean packet loss	M	O (see note1)
Transport parameters qualifier	- Transport QoS parameters indicate total remaining budget - Transport QoS parameters indicate budget available per domain	M	
Traffic descriptor	Media peak rate, Maximum media frame size	M	
PreviousDomainEgressAddress (forward path)	Network specific address	M	
NextDomainAddress	Network domain address	O (see note 2)	
UserDomainAddress	Network specific address	O (see note 2)	
Egress Point (forward path)	Network specific address		O(see note 3)
Result	- Transport resource reserved - Rejection cause - resource unavailable - destination unknown		M
NOTE 1: This information element shall be included if the value of the transport parameters qualifier in the request is "Transport QoS parameters indicate total remaining budget". NOTE 2: Exactly one of these information elements must be present. NOTE 3: The EgressDomainAddress shall be included if information element value "result" is "transport resource reserved".			

5.2.1.1.2 TransportAssign

`TransportAssign` is a confirmed information flow that shall be sent across relationship ra from the transport user to $TFE1_{TRM}$ to allocate a reserved transport resource. Table 2 lists the information elements in the `TransportAssign` information flow.

Table 2: Contents of `TransportAssign`

TransportAssign			
Information element	**Value**	**Request**	**Response**
BearerIdentifier	Alphanumeric "handle"	M	
Next Domain Egress point (backward path)	Network specific address	M	
Egress point (backward path)	Network specific address		O (see note)
Result	- Reserved Connection completed - Rejection cause - unable to complete connection		M
NOTE: Shall be present if result is "transport resource assigned".			

5.2.1.1.3 TransportRelease

`TransportRelease` is an unconfirmed information flow that shall be sent across relationship ra from the transport user to $TFE1_{TRM}$ to release an allocated transport resource. Table 3 lists the information elements in the `TransportRelease` information flow.

Table 3: Contents of `TransportRelease`

TransportRelease		
Information element	**Value**	**Request**
BearerIdentifier	Alphanumeric "handle"	M

5.2.1.2 Relationship rb

5.2.1.2.1 AssignIngressPoint

`AssignIngressPoint` is a confirmed information flow that shall be sent across relationship rb from $TFE1_{TRM}$ to $TFE2_{ITF}$ to identify and connect the ingress point and to connect it to the intra domain connection. Table 4 lists the information elements in the `AssignIngressPoint` information flow.

Table 4: Contents of `AssignIngressPoint`

AssignIngressPoint			
Information element	**Value**	**Request**	**Response**
BearerIdentifier	Alphanumeric "handle"	M	
Transport QoS parameters	Maximum delay, Maximum packet delay variation, Maximum mean packet loss	M	O (see note)
Traffic descriptor	Media peak rate, Maximum media frame size	M	
PreviousDomainEgressAddress	Network specific address	M	
ConnectionAddress (forward path)	Network specific address	M	
ConnectionAddress (backward path)	Network specific address	M	

Result	- IngressAssigned
	- Rejection cause
	  - resource unavailable
	  - Address not found
			M

NOTE: This information element shall be included if Result is "IngressAssigned". It reports the resource allocated by the egress connection.			

5.2.1.2.2 ReleaseIngressPoint

`ReleaseIngressPoint` is an unconfirmed information flow that shall be sent across relationship rb from $TFE1_{TRM}$ to $TFE2_{ITF}$ to release the transport resources allocated by the specified bearer. Table 5 lists the information elements in the `ReleaseIngressPoint` information flow.

Table 5: Contents of `ReleaseIngressPoint`

ReleaseIngressPoint		
Information element	**Value**	**Request**
BearerIdentifier	Alphanumeric "handle"	M

5.2.1.2.3 ConnectIngressPoint

`ConnectIngressPoint` is a confirmed information flow that shall be sent across relationship rb from $TFE1_{TRM}$ to $TFE2_{ITF}$ to connect the reserved ingress point and the next domain egress point for the backward transport path. Table 6 lists the information elements in the `ConnectIngressPoint` information flow.

Table 6: Contents of `ConnectIngressPoint`

ConnectIngressPoint			
Information element	**Value**	**Request**	**Response**
Bearer Identifier	Alphanumeric "handle"	M	
Next Domain Egress point (backward path)	Network specific address	M	
Result	- Connection made - Rejection cause - unable to connect		M

5.2.1.3 Relationship rc

5.2.1.3.1 AssignEgressPoint

`AssignEgressPoint` is a confirmed information flow that shall be sent across relationship rc from $TFE1_{TRM}$ to $TFE3_{ETF}$ to connect an egress point to the next domain and the egress point to the specified intra domain connection. Table 7 lists the information elements in the `AssignEgressPoint` information flow.

Table 7: Contents of `AssignEgressPoint`

AssignEgressPoint			
Information element	**Value**	**Request**	**Response**
BearerIdentifier	Alphanumeric "handle"	M	
Transport QoS parameters	Maximum delay, Maximum packet delay variation, Maximum mean packet loss	M	O (see note 1)
Traffic descriptor	Media peak rate, Maximum media frame size	M	
NextDomainAddress	Network domain address	O(see note 2)	
UserDomainAddress	Network specific address	O(see note 2)	
ConnectionAddress (forward path)	Network specific address	M	
ConnectionAddress (backward path)	Network specific address	M	
EgressAddressFw	Network specific address		O(see note 3)
EgressAddressBw	Network specific address		O(see note 3)

Result	- EgressAssigned
	- Rejection cause
	  - resource unavailable
	  - address not found
			M

NOTE 1: The Transport QoS parameters information element shall be present if the Result element returns value "EgressAssigned" and information element UserDomainAddress was present in the request. It reports the actual resource allocated by the egress connections. NOTE 2: Exactly one of these information elements shall be present. NOTE 3: The forward and backward path EgressAddress information elements shall be present if the Result element returns value "EgressAssigned".			

5.2.1.3.2 ReleaseEgressPoint

`ReleaseEgressPoint` is an unconfirmed information flow that shall be sent across relationship rc from $TFE1_{TRM}$ to $TFE3_{ETF}$ to release the transport resources allocated to the specified bearer. Table 8 lists the information elements in the `ReleaseEgressPoint` information flow.

Table 8: Contents of `ReleaseEgressPoint`

ReleaseEgressPoint		
Information element	**Value**	**Request**
BearerIdentifier	Alphanumeric "handle"	M

5.2.2 Timers

5.2.2.1 Reservation hold timer

A transport Reservation Hold Timer in $TFE1_{TRM}$ is used to ensure that reserved transport resources are not held indefinitely if a `TransportAssign request` information flow is not received within a certain time after reserving the transport resources. The period of the Reservation Hold Timer is implementation dependent but shall be in the range of 8 seconds to 15 seconds.

5.2.3 Information flow sequences

A standard specifying TIPHON meta-protocols for transport service signalling shall provide signalling procedures in support of the information flow sequences specified below.

In the figures, transport service signalling information flows are represented by solid arrows. Within a column representing a media control signalling functional entity, the numbers refer to functional entity actions listed in clause 5.3.

The following abbreviations are used:

- req request;
- resp response.

5.2.3.1 Normal operation

Figure 3 shows the information flows for successful reservation and assignment of transport resources.

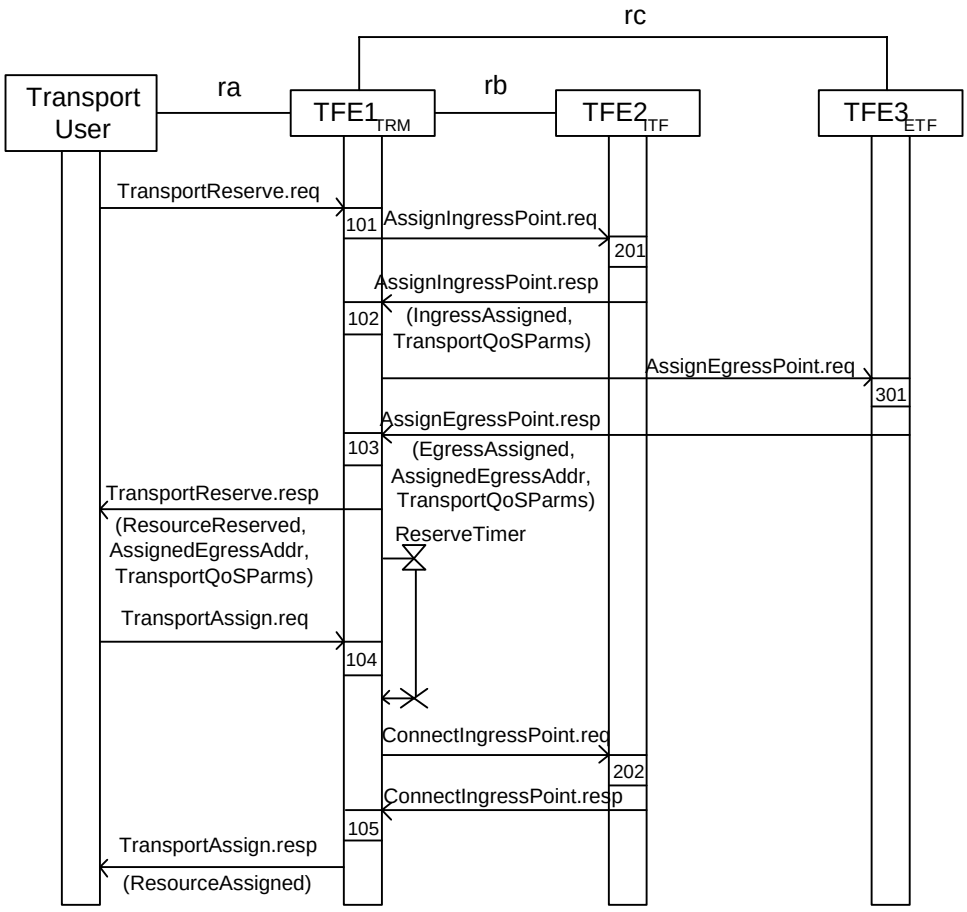


Figure 3: Successful transport resource allocation

Figure 4 shows the information flows for release of transport resources.

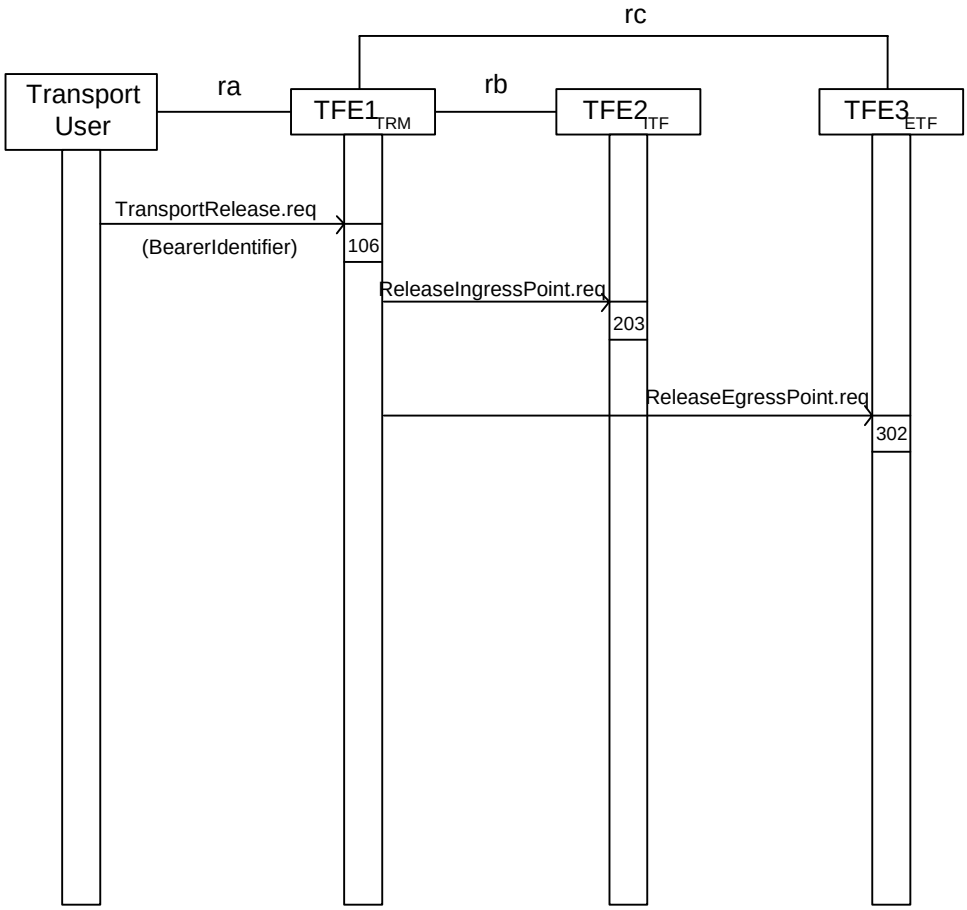


Figure 4: Transport resource release

5.2.3.2 Exceptional behaviour

Figure 5 shows unsuccessful transport resource assignment due to media resource reserve time expiration.

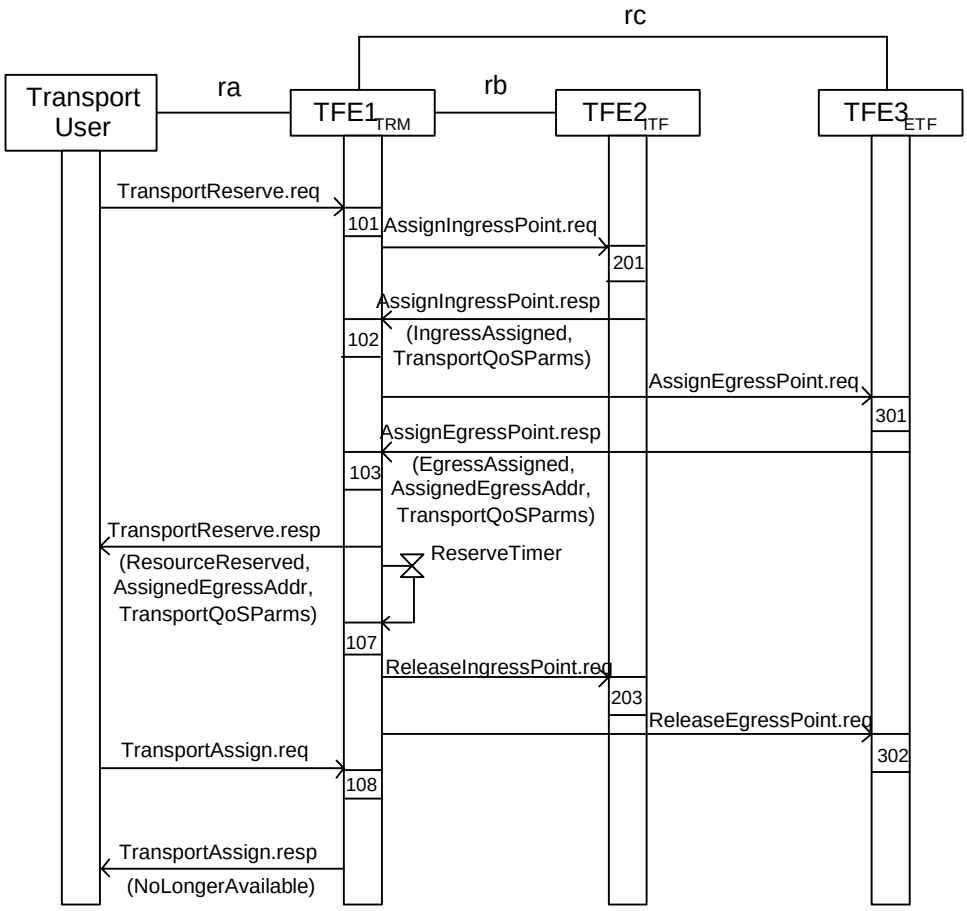


Figure 5: Unsuccessful transport allocation due to reservation timeout

Figure 6 shows unsuccessful transport reservation due to required transport resource not being available. Similar scenarios exist when in $TFE1_{TRM}$ or $TFE2_{ITF}$ sufficient resource is not available.

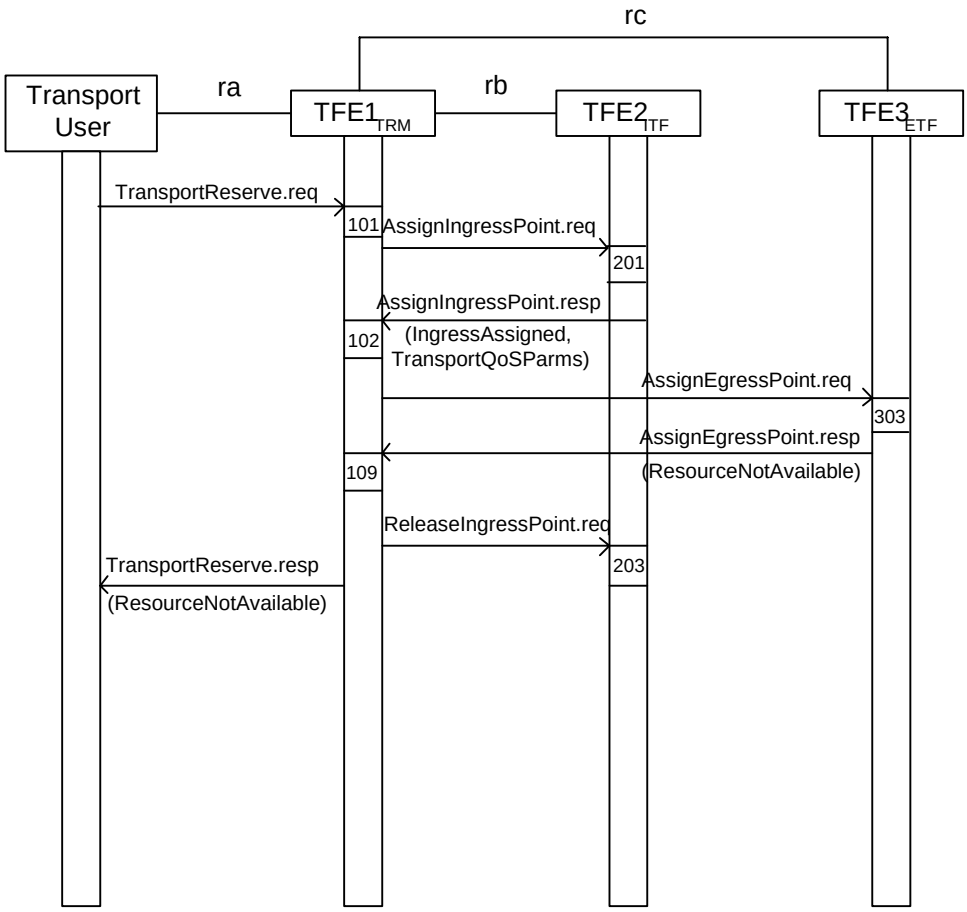


Figure 6: Unsuccessful transport reservation due to resource not being available.

Figure 7 shows unsuccessful transport resource reservation due to next domain address could not be identified. Similar cases exist when the previous domain can not be determined.

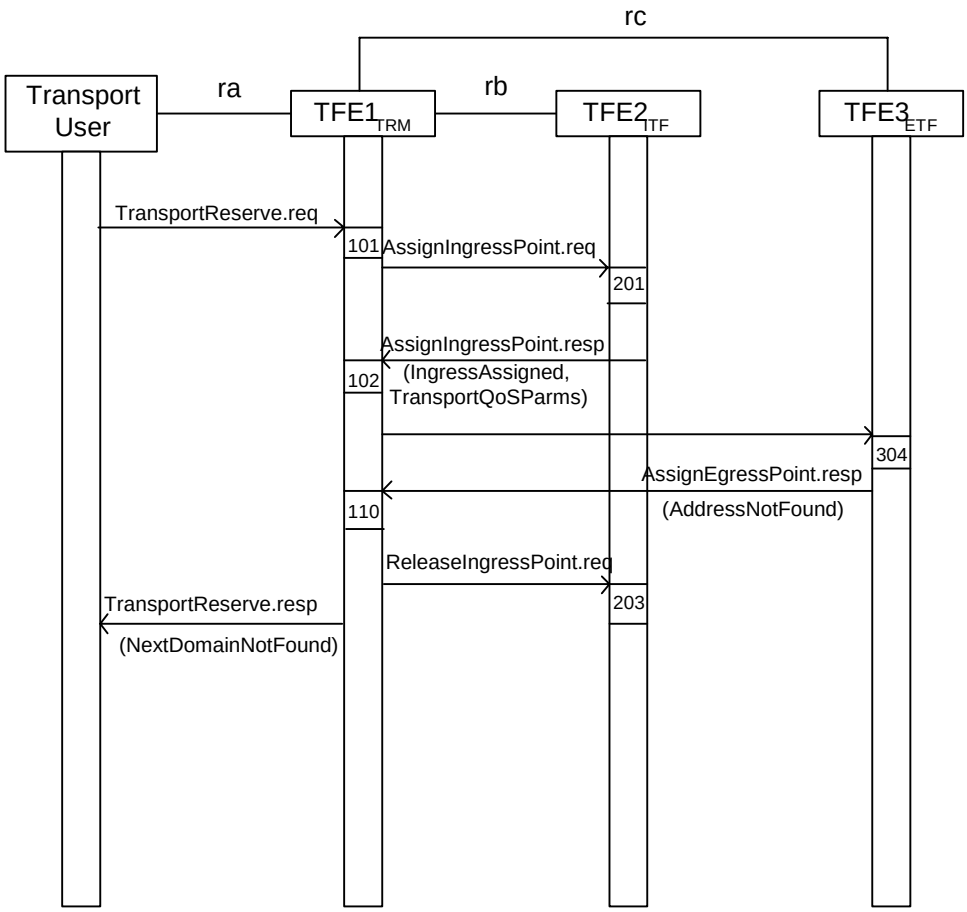


Figure 7: Unsuccessful transport reservation due to next domain address not found

5.3 Transport service functional entity actions

The following conventions are used to identify information flows in the descriptions of TFE actions:

- an information flow is referred to as a "request" at the TFE that sends it and as an "indication" at the TFE that receives it;
- the corresponding confirmation is referred to as a "response" at the TFE that sends it and as a "confirmation" at the TFE that receives it.

The following TFE actions shall occur at the points indicated in the figures of clause 5.2.3.

5.3.1 Actions of $TFE1_{TRM}$

101: On receipt of a `TransportReserve` indication from the Transport User (TU), $TFE1_{TRM}$ shall check if the resource request complies with the transport policy and if an intra domain connection fulfilling the transport resource requirement is available. If so, $TFE1_{TRM}$ shall allocate the connection and send an `AssignIngressPoint` request to $TFE2_{ITF}$ to assign the ingress point and connect this to the allocated intra domain connection. The request parameters shall include also the remaining transport resource budget to enable $TFE2_{ITF}$ to check if the ingress point can fulfil the transport requirements;

102: When a successful `AssignIngressPoint` confirmation is received, $TFE1_{TRM}$ shall calculate the available transport resource budget for the egress point and send an `AssignEgressPoint` request to $TFE3_{ETF}$ to assign the egress point and connect this to the allocated intra domain connection;

103: When a successful `AssignIngressPoint` confirmation is received from $TFE3_{ETF}$, $TFE1_{TRM}$ shall prepare and send a `TransportReserve` response indicating successful reservation, the assigned egress address, and the remaining QoS transport budget in case total transport budget was indicated in the `TransportReserve` request. $TFE1_{TRM}$ shall also start the resource reservation timer;

104: On receipt of a `TransportAssign` indication from TU, the reservation timer shall be stopped and $TFE1_{TRM}$ shall prepare and send a `IngressConnect` request to $TFE2_{ITF}$ with the specified next domain forward path egress address to connect to the reserved ingress point;

105: Receiving a `ConnectIngresspoint confirmation from` $TFE2_{ITF}$ `with result "`Connection made`",` $TFE1_{TRM}$ shall send a `TransportAssign` confirmation to TU with result "Reserved Connection completed" and the Egress point address of the backward transport path;

106: Receiving a `TransportRelease` indication, $TFE1_{TRM}$ shall release the allocated transport resource from the connection associated to the bearer identifier, and send release requests to $TFE2_{ITF}$ and $TFE3_{ETF}$ to release the associated transport resource of the ingress and egress points;

107: If the reservation timer expires, $TFE1_{TRM}$ shall release the allocated transport resource from the connection and send release requests to $TFE2_{ITF}$ and $TFE3_{ETF}$ to release the associated transport resource;

108: On receipt of a `TransportAssign` indication from TU when the resource reservation timer has expired, $TFE1_{TRM}$ shall prepare a `TransportAssign` response with result "resource no longer available" and send it to TU;

109: On receipt of an `AssignEgressPoint` confirmation from $TFE3_{ETF}$ indicating transport resource not available, $TFE1_{TRM}$ shall release the connection resource, send a `ReleaseIngressPoint` request to $TFE2_{ITF}$ and send a `TransportReserve` response to TU indicating "transport resource not available";

110: On receipt of an `AssignEgressPoint` confirmation from $TFE3_{ETF}$ indicating next domain address not found, $TFE1_{TRM}$ shall release the assigned connection resource, send a `ReleaseIngressPoint` request to $TFE2_{ITF}$, and send a `TransportReserve` response to TU indicating "next domain address not found".

5.3.2 Actions of $TFE2_{ITF}$

201: On receipt of an `AssignIngressPoint` indication from $TFE1_{TRM}$, $TFE2_{ITF}$ shall check if the previous domain can be identified and if sufficient transport resource is available at the ingress point. If so, $TFE2_{ITF}$ shall allocate the ingress point and connect the ingress point and the connection address provided in the request, and send an `AssignIngressPoint` response to $TFE1_{TRM}$ indicating successful allocation of resource at the ingress point. In case of successful resource allocation also the remaining transport resource budget shall be passed in the assign response;

202: Receiving a `ConnectIngressPoint` indication from $TFE1_{TRM}$, $TFE2_{ITF}$ shall connect the ingress point reserved for the backback path and the specified next domain egress point for the backward transport path, and send a `ConnectIngressPoint` response to $TFE1_{TRM}$;

203: On receipt of a `ReleaseIngressPoint` indication, transport resource associated to the ingress point for this bearer identifier shall be released and the connection to the intra domain address removed.

5.3.3 Actions of $TFE3_{ETF}$

301: On receipt of an `AssignEgressPoint` indication from $TFE1_{TRM}$, $TFE3_{ETF}$ shall check if the next domain can be identified and an egress point with sufficient transport resource can be identified. If so $TFE3_{ETF}$ shall allocate the bearer identifier to this egress point, connect it to the intra domain connection address provided in the request, and prepare an `AssignEgressPoint` response to $TFE1_{TRM}$ indicating successful resource assignment and the assigned egress address;

302: Receiving a `ReleaseEgressPoint` indication, $TFE3_{ETF}$ shall release all transport resources allocated for this bearer identifier and remove the connection to the intra domain address;

303: When an `AssignEgressPoint` indication from $TFE1_{TRM}$ is received for which the transport resource requirement can not be fulfilled, $TFE3_{ETF}$ shall prepare an `AssignEgressPoint` response to $TFE1_{TRM}$ indicating that requested transport resource is not available;

304: On receipt of a `AssignEgressPoint` indication from $TFE1_{TRM}$, $TFE2_{ITF}$ shall check if the next domain can be identified, and if not, $TFE2_{ITF}$ shall prepare an `AssignEgressPoint` response to $TFE1_{TRM}$ indicating that the next domain address can not be identified.

5.4 Transport control service functional entity behaviour

The behaviour specified in this clause is intended to illustrate typical TFE behaviour in terms of information flows sent and received.

The behaviour of each TFE is shown using the Specification and Description Language (SDL) defined in ITU-T Recommendation Z.100 [4].

5.4.1 Information flows specified as ASN.1 operations

For the purposes of modelling transport control service signalling in SDL, the information flows have been specified using the Abstract Syntax Notation 1 (ASN.1) defined in ITU-T Recommendation X.680 [5]. The ASN.1 is shown in table 9**.**

Table 9: Transport control service information flows specified as ASN.1

```
TransportControlType DEFINITIONS ::=
BEGIN

-- Data structures for the transport control service signals --

TransportReserveReq_Type ::= SEQUENCE
{ bearerIdentifier       BearerIdentifierType,
  transportParmQualifier TransportParmQualifierType,
  transportQoSParms      TransportQoSParmsType,
  trafficDescr           TrafficDescrType,
  previousDomEgressFw    NetworkSpecificAddrType,
  nextDomainAddress      NetworkDomainAddrType OPTIONAL,
  userDomainAddress      NetworkSpecificAddrType OPTIONAL
}

TransportReserveResp_Type ::= SEQUENCE
{ remainingTransportBudget  TransportQoSParmsType OPTIONAL,
  egressPointFw             NetworkSpecificAddrType OPTIONAL,
  result                    ReservationResultType
}

TransportAssignReq_Type ::= SEQUENCE
{ bearerId           BearerIdentifierType,
  nextDomainEgressBw NetworkSpecificAddrType
}
```

```
TransportAssignResp_Type ::= SEQUENCE
{ egressPointBw  NetworkSpecificAddrType OPTIONAL,
  result         TransportAssignResultType
}

TransportReleaseReq_Type ::= BearerIdentifierType

AssignIngressPointReq_Type ::= SEQUENCE
{ bearerIdentifier            BearerIdentifierType,
  availableTransportQoSParms  TransportQoSParmsType,
  trafficDescr                TrafficDescrType,
  previousDomEgressFw         NetworkSpecificAddrType,
  connectionAddressFw         NetworkSpecificAddrType,
  connectionAddressBw         NetworkSpecificAddrType
}

AssignIngressPointResp_Type ::= SEQUENCE
{ result                      AssignIngressResultType,
  allocatedTransportQoSParms  TransportQoSParmsType OPTIONAL
}

ReleaseIngressPointReq_Type ::= BearerIdentifierType

ConnectIngressPointReq_Type ::= SEQUENCE
{ bearerIdentifier BearerIdentifierType,
  nextDomEgressBw  NetworkSpecificAddrType
}

ConnectIngressPointResp_Type ::= ConnectResultType;

AssignEgressPointReq_Type ::= SEQUENCE
{ bearerIdentifier            BearerIdentifierType,
  availableTransportQoSParms  TransportQoSParmsType,
  trafficDescr                TrafficDescrType,
  nextDomainAddress           NetworkDomainAddrType OPTIONAL,
  userDomainAddress           NetworkSpecificAddrType OPTIONAL,
  connectionAddressFw         NetworkSpecificAddrType,
  connectionAddressBw         NetworkSpecificAddrType
}

AssignEgressPointResp_Type ::= SEQUENCE
{ result                    AssignEgressResultType,
  egressAddressFw           NetworkSpecificAddrType OPTIONAL,
  egressAddressBw           NetworkSpecificAddrType OPTIONAL,
  allocatedTransportBudget  TransportQoSParmsType OPTIONAL
}

ReleaseEgressPointReq_Type ::= BearerIdentifierType


/* -- Information element types -- */

AssignEgressResultType ::= ENUMERATED
{ egressPointAssigned,
  egressResourceUnavailable,
  nextDomainAddrNotFound
}

AssignIngressResultType ::= ENUMERATED
{ ingressPointAssigned,
  ingressResourceUnavailable
}

BearerIdentifierType ::= Visiblestring

IPAddressType ::= CHOICE
{ ipv4Address  IPv4AddressType,
  ipv6Address  IPv6AddressType
}

IPv4AddressType ::= SEQUENCE
{ addr  FourOctetsType,
  port  OneOctetType
}

IPv6AddressType ::= SEQUENCE
{ addr  SixteenOctetsType,
```

```
  port  SixteenOctetsType
}

OneOctetType ::= Octet_String( SIZE(1) )

FourOctetsType ::= Octet_String( SIZE(4) )

SixteenOctetsType ::= Octet_String( SIZE(16) )

NetworkSpecificAddrType ::= CHOICE
{
  slotNumber SlotNumberType,   -- FOR FURTHER DISCUSSION
  ipAddress  IPAddressType
}

NetworkDomainAddrType ::= CHOICE
{ ipv4Domain  FourOctetsType,
  ipv6Domain  SixteenOctetsType
}

ReservationResultType ::= ENUMERATED
{ bandwidthReserved,
  bandwidthUnavailable,
  destinationUnknown
}

TransportQoSParmsType ::= SEQUENCE
{ maximumDelay      MicroSeconds,
  maxDelayVariation MicroSeconds,
  maxMeanPacketLoss PercentX1000
    -- Packet loss is specified as % x 1000 to avoid --
    -- the need for REAL numbers when loss is less   --
         -- than one percent                                --
}

MicroSeconds ::= Integer( 0 .. 10000000 )

PercentX1000 ::= Integer (0 : 100000 )

SlotNumberType ::= Integer

TrafficDescrType ::= SEQUENCE
{ peakFrameRate    FrameRateType,
  framesPerPacket  FrameCountType
}

FrameRateType ::= Integer( 1..255)

FrameCountType ::= Integer(0..maxFrameCount)

maxFrameCount Integer ::= 32

ConnectResultType ::= ENUMERATED
{ connectionMade,
  unableToConnect
}

TransportAssignResultType ::= ConnectResultType

TransportParmQualifierType ::= ENUMERATED
{ totalRemainingBudget,
  budgetAvailableForDomain
}

END
```

5.4.2 Behaviour of $TFE1_{TRM}$

The behaviour of $TFE1_{TRM}$ is shown in the SDL process diagram in figure 8 to figure 18.

Figure 8: SDL process diagram for functional entity $TFE1_{TRM}$ (1 of 12)

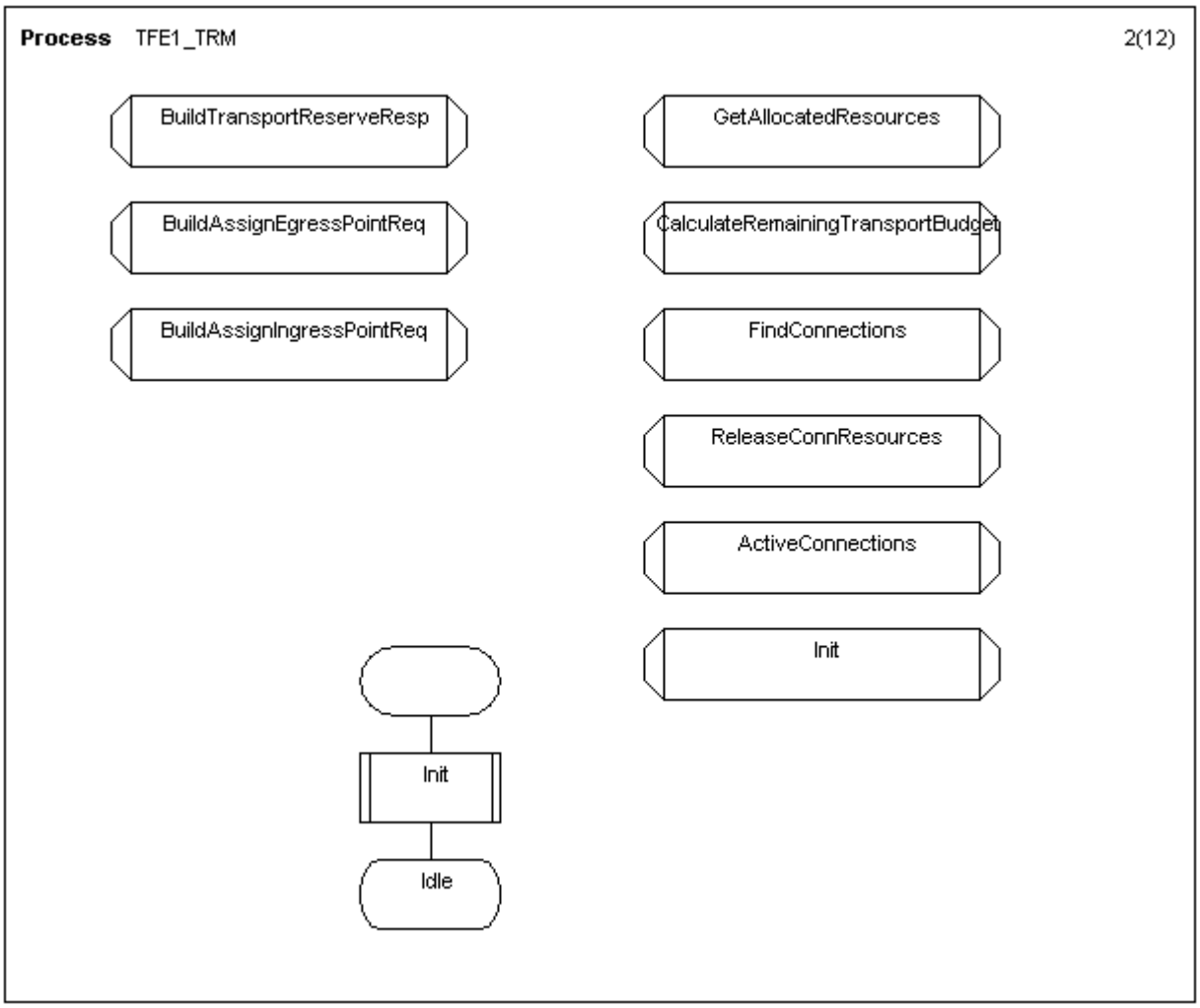


Figure 9: SDL process diagram for functional entity $TFE1_{TRM}$ (2 of 12)

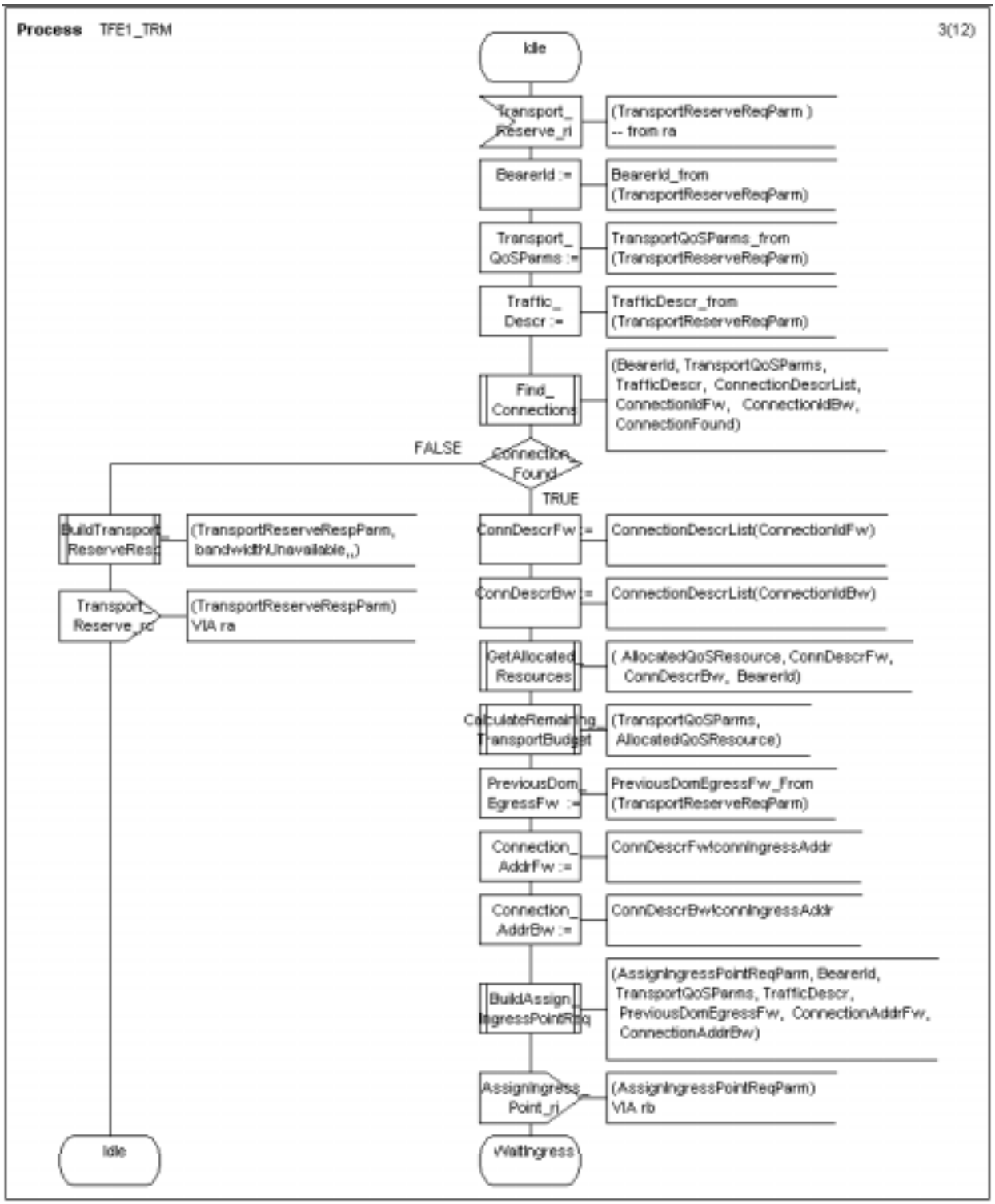


Figure 10: SDL process diagram for functional entity $TFE1_{TRM}$ (3 of 12)

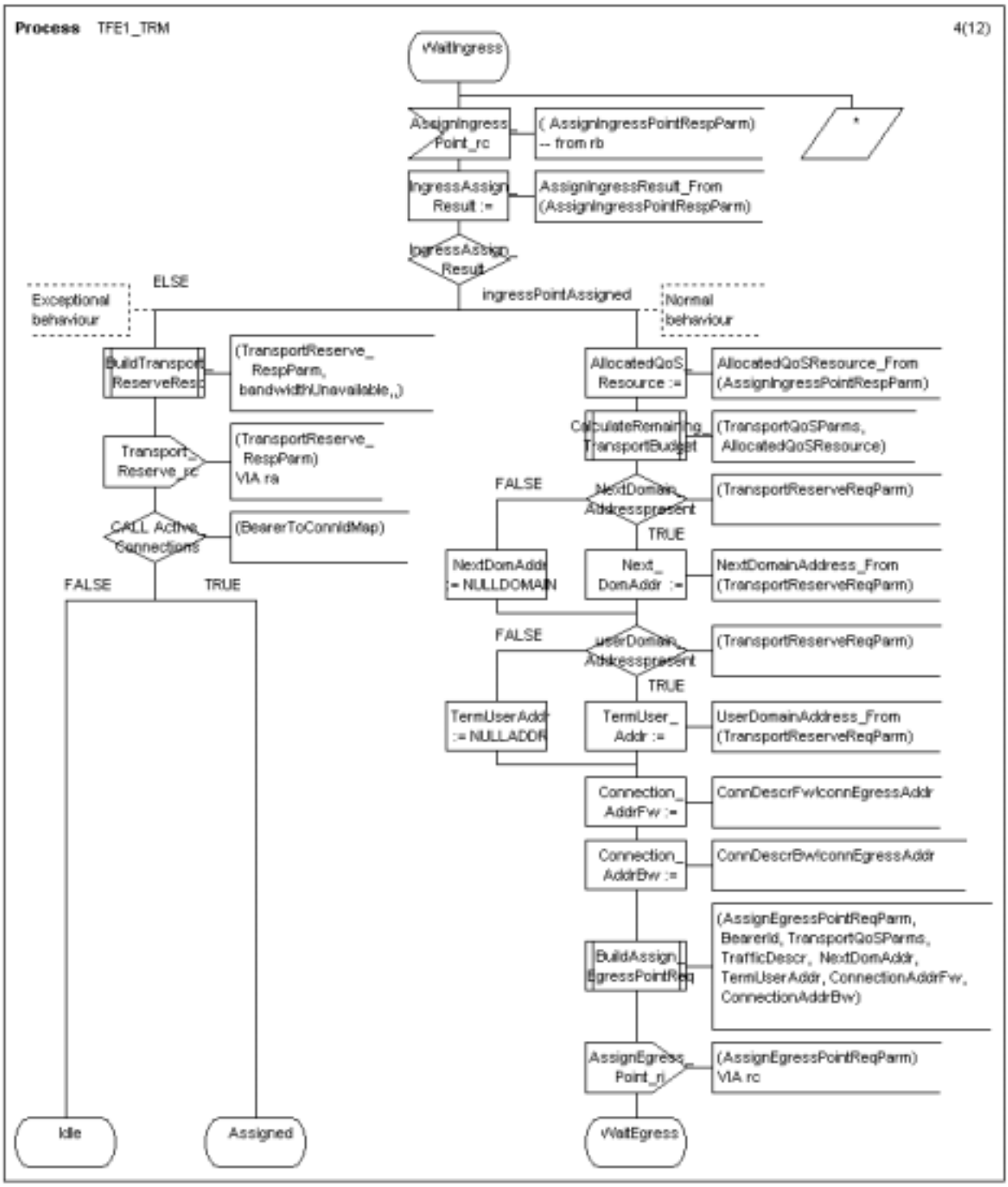


Figure 11: SDL process diagram for functional entity $TFE1_{TRM}$ (4 of 12)

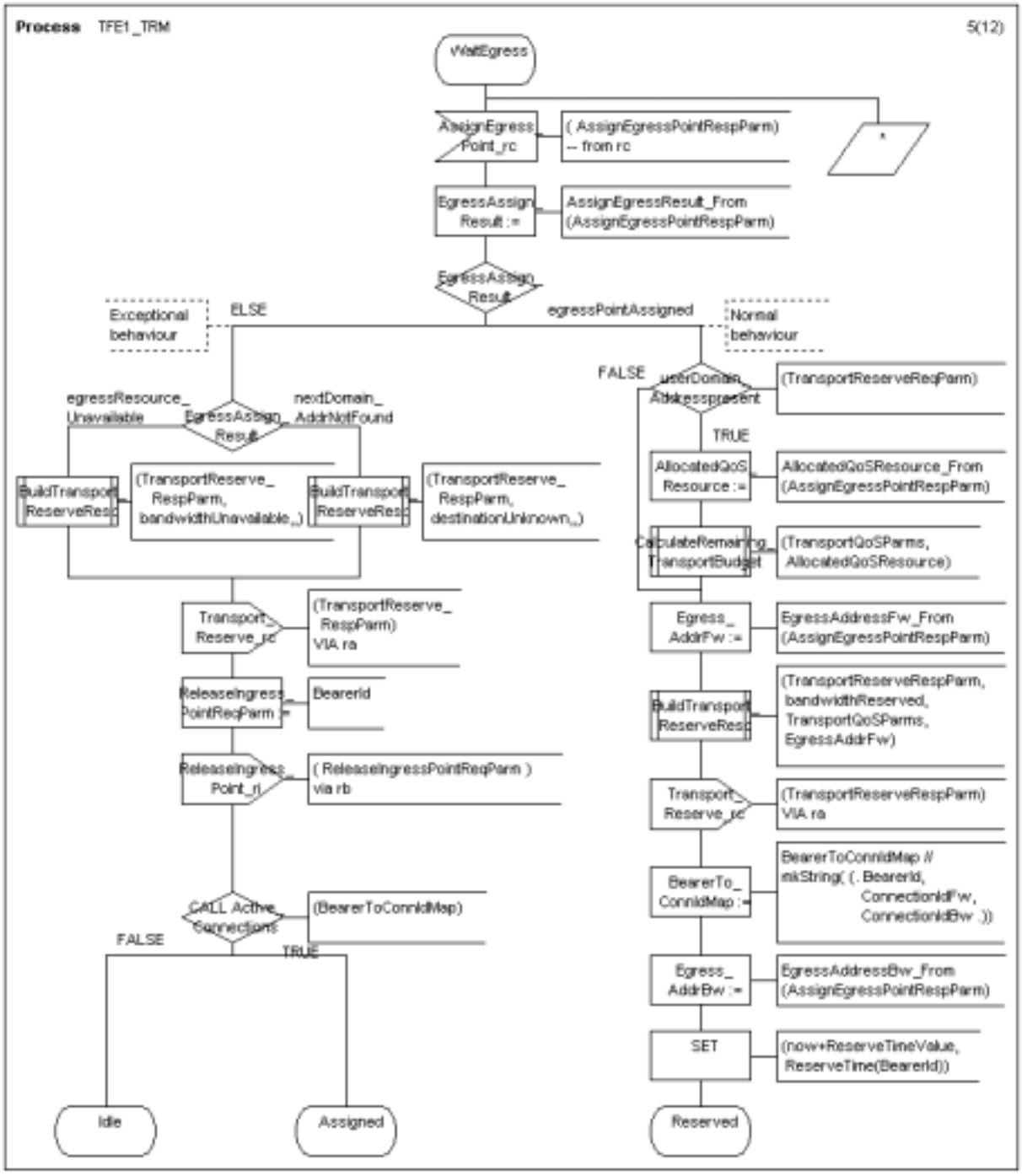


Figure 12: SDL process diagram for functional entity $TFE1_{TRM}$ (5 of 12)

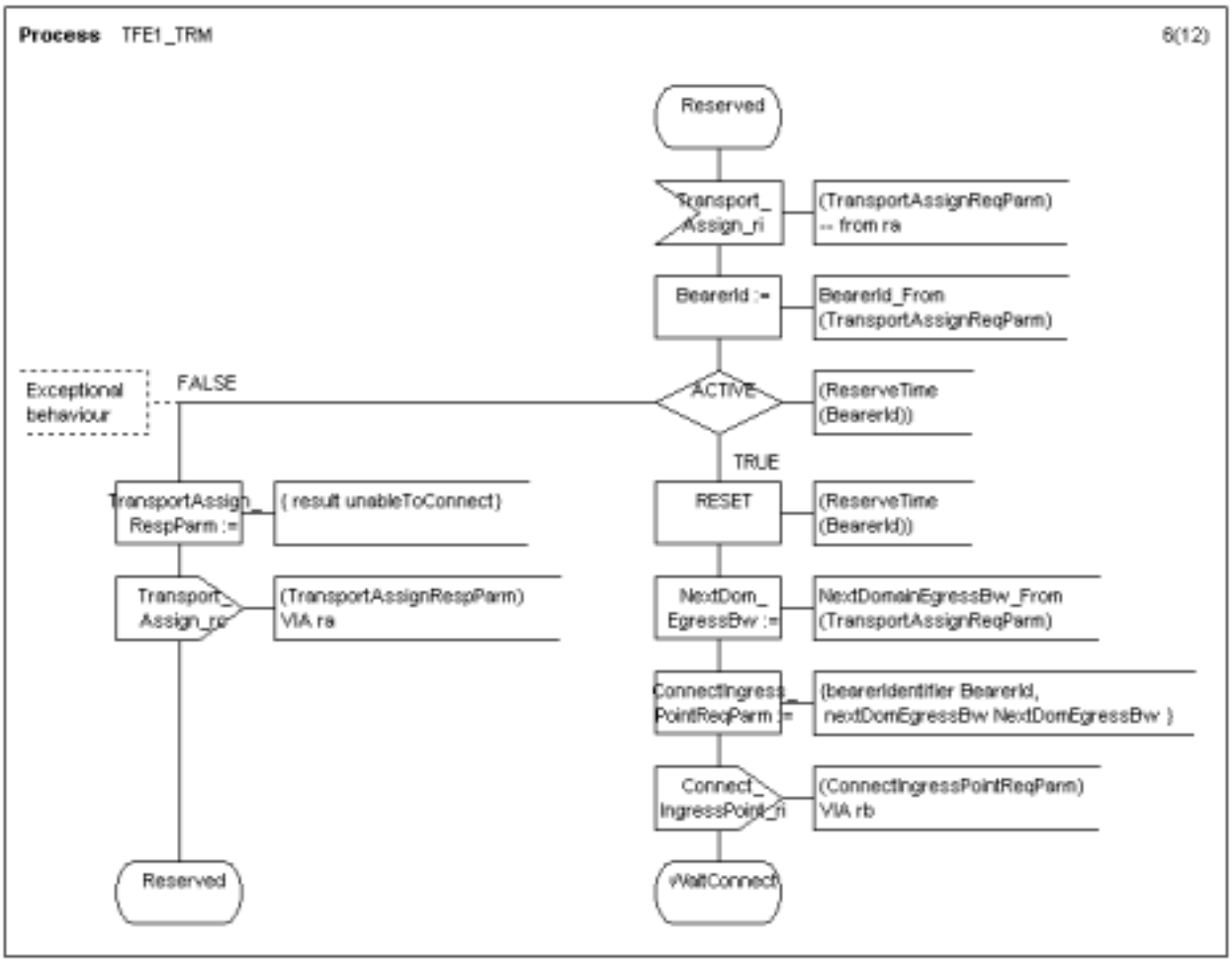


Figure 13: SDL process diagram for functional entity $TFE1_{TRM}$ (6 of 12)

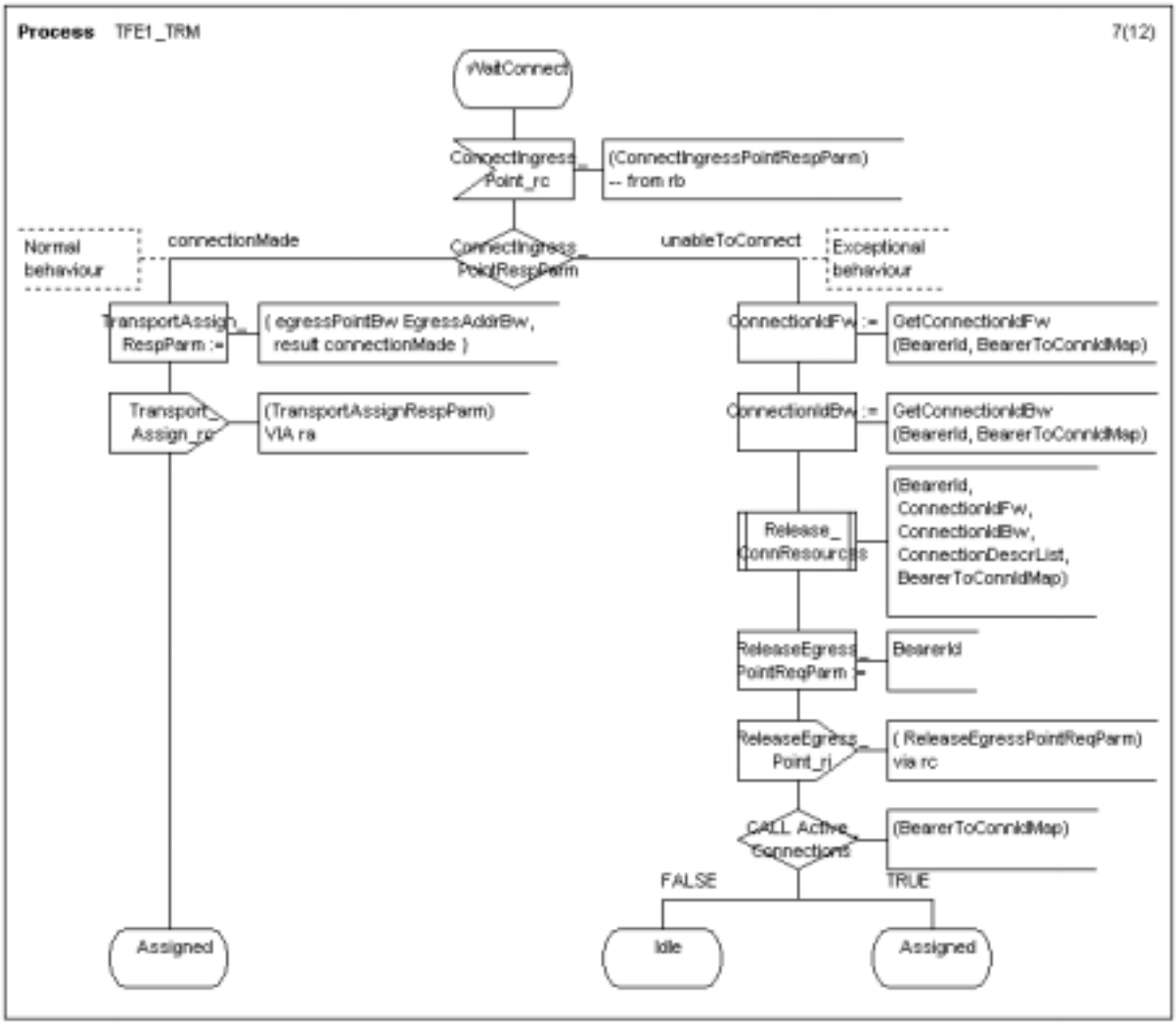


Figure 14: SDL process diagram for functional entity $TFE1_{TRM}$ (7 of 12)

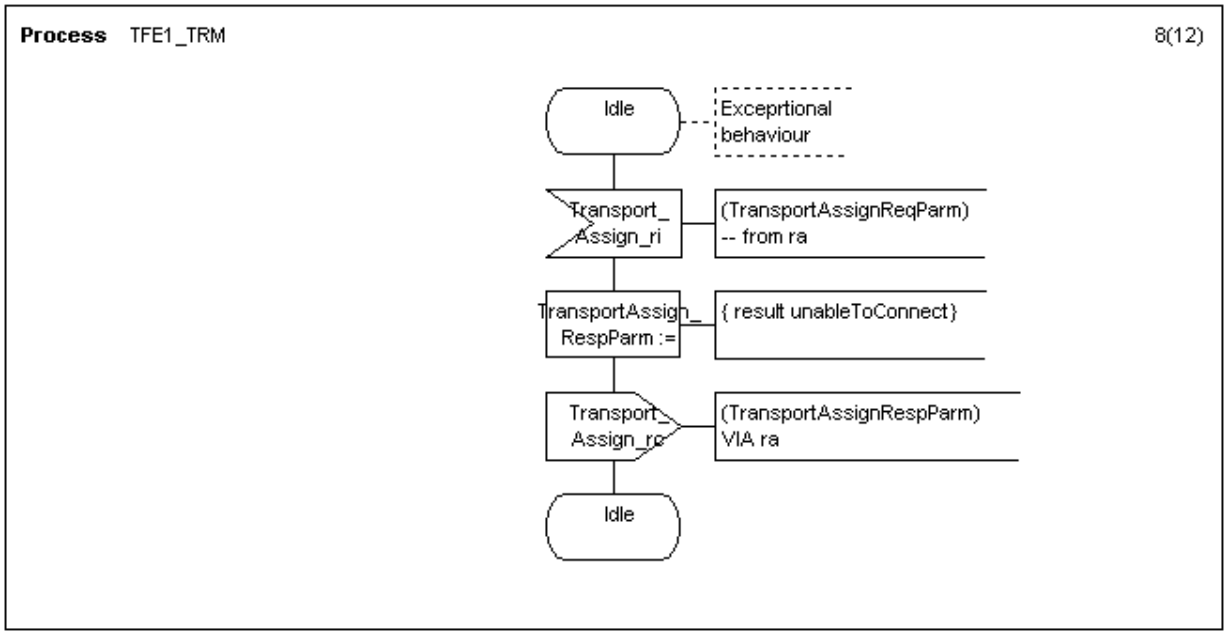


Figure 15: SDL process diagram for functional entity $TFE1_{TRM}$ (8 of 12)

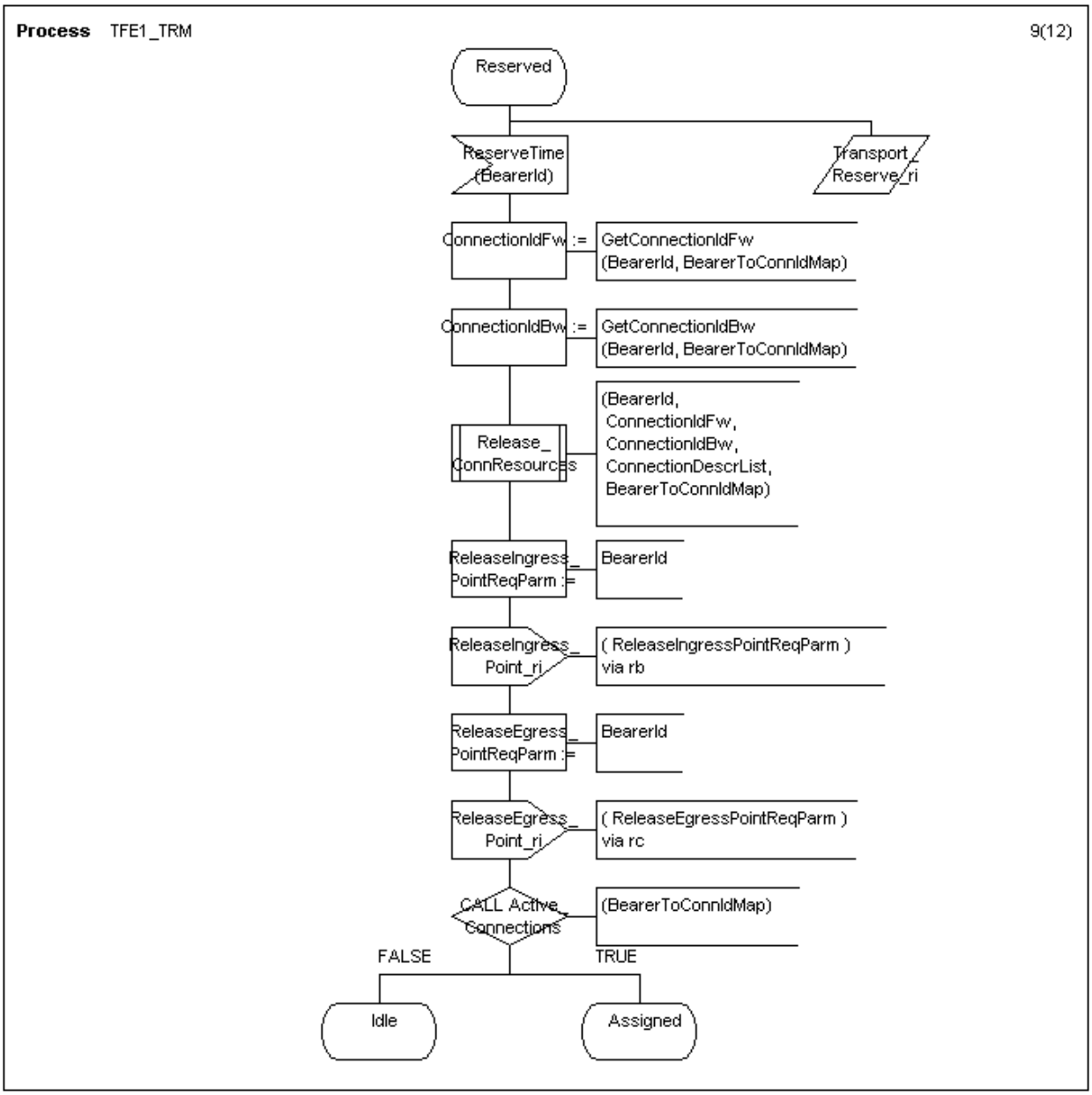


Figure 16: SDL process diagram for functional entity $TFE1_{TRM}$ (9 of 12)

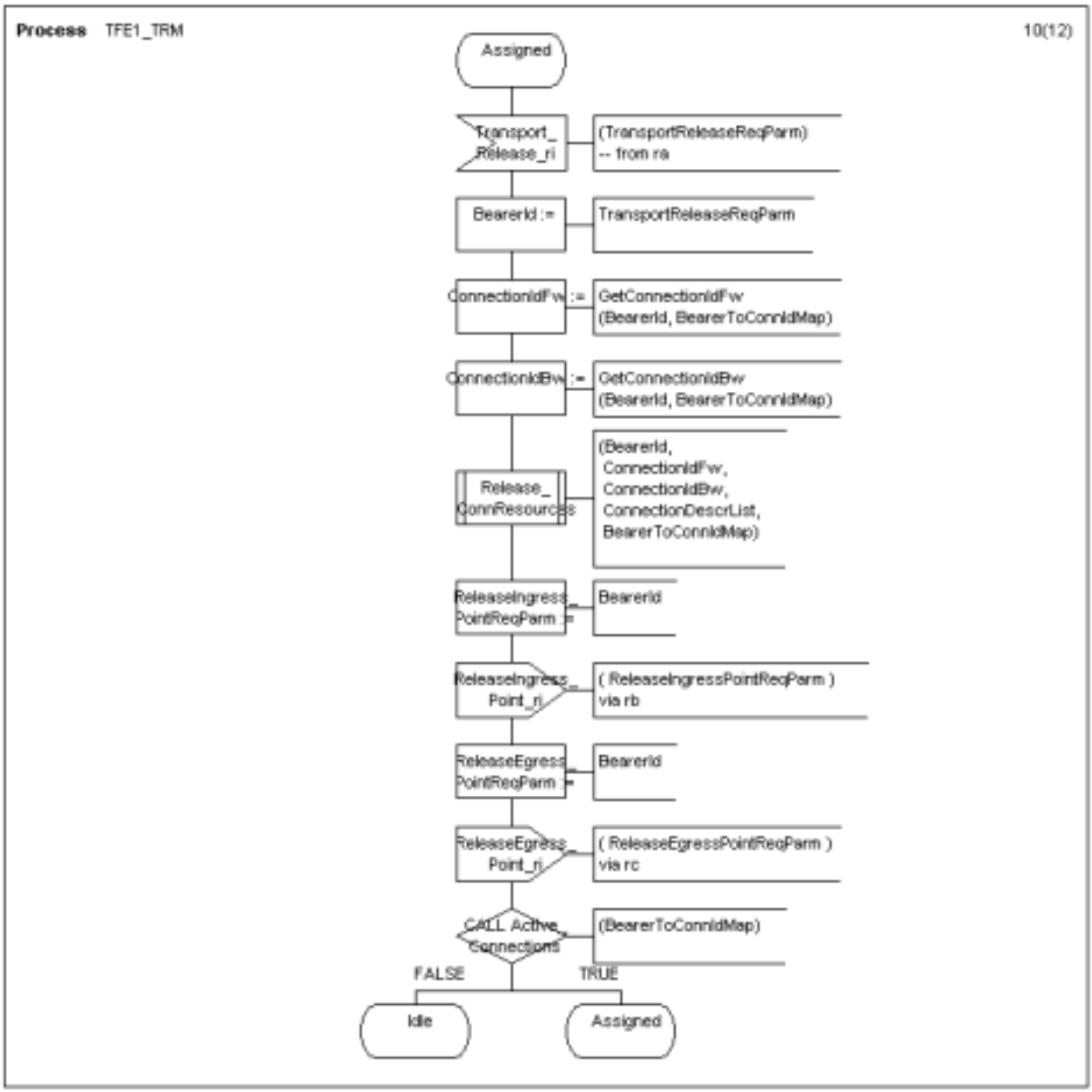


Figure 17: SDL process diagram for functional entity $TFE1_{TRM}$ (10 of 12)

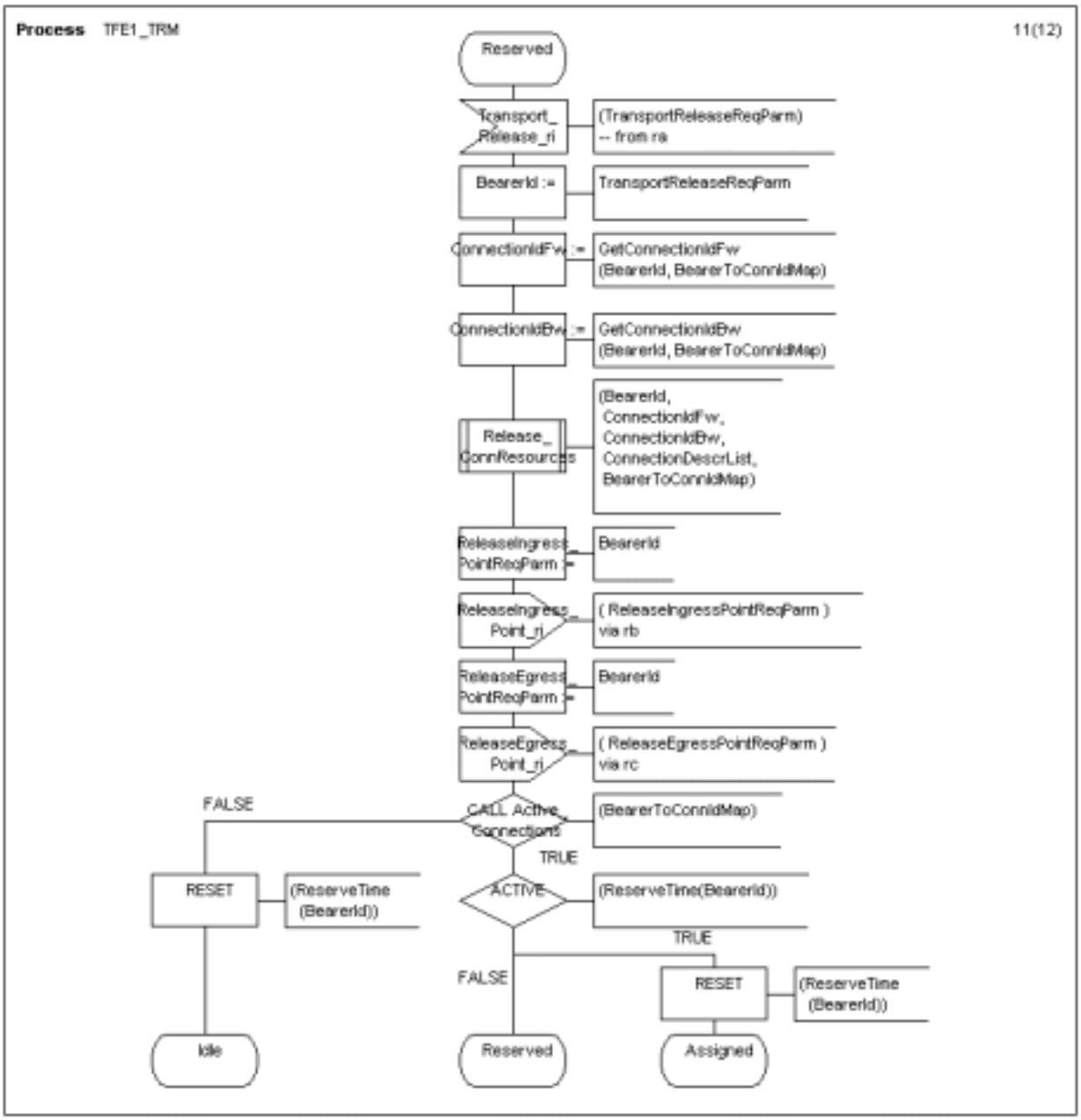


Figure 18: SDL process diagram for functional entity $TFE1_{TRM}$ (11 of 12)

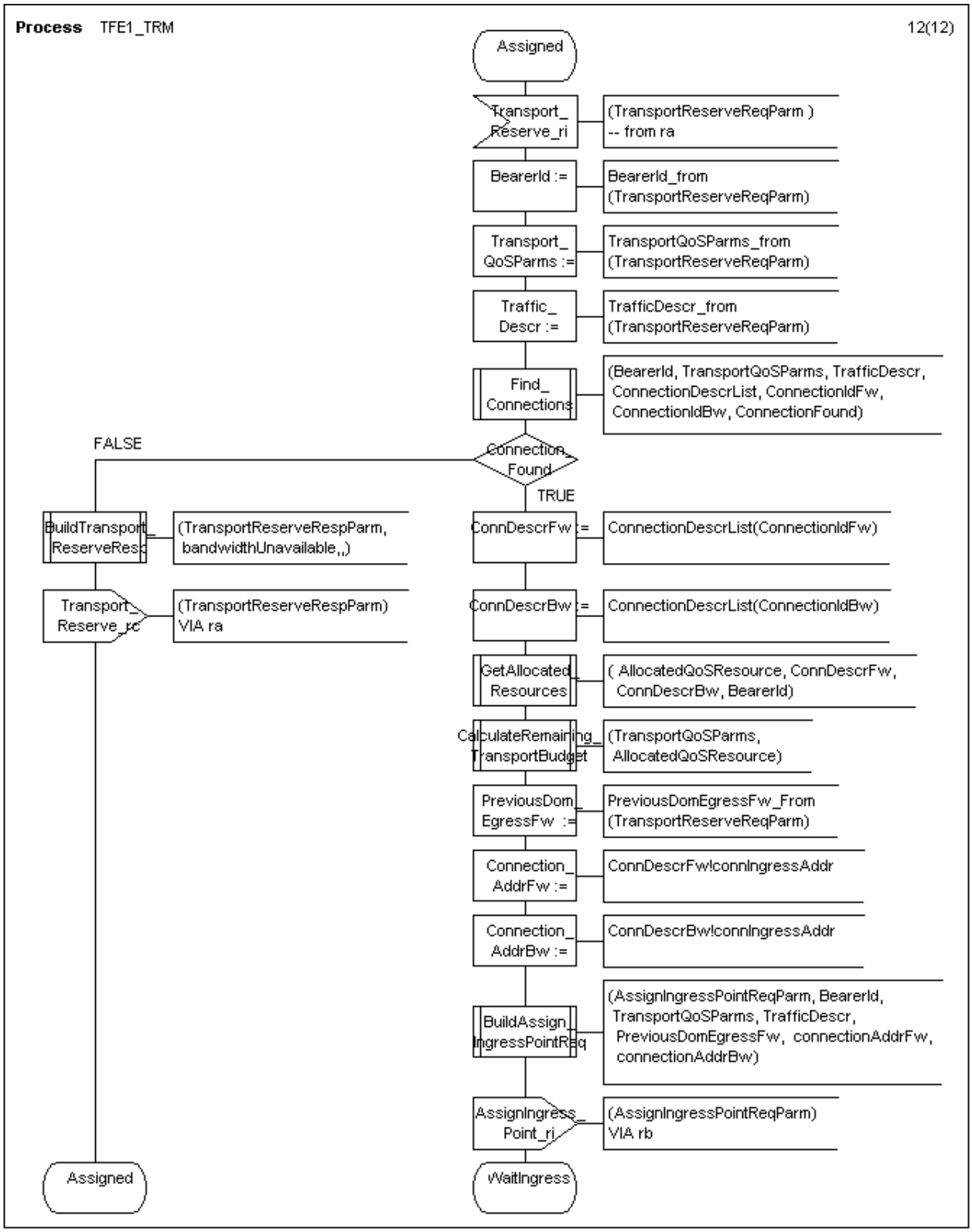


Figure 19: SDL process diagram for functional entity $TFE1_{TRM}$ (12 of 12)

5.4.3 Behaviour of $TFE2_{ITF}$

The behaviour of $TFE2_{ITF}$ is shown in the SDL process diagram in figure 20 to figure 25.

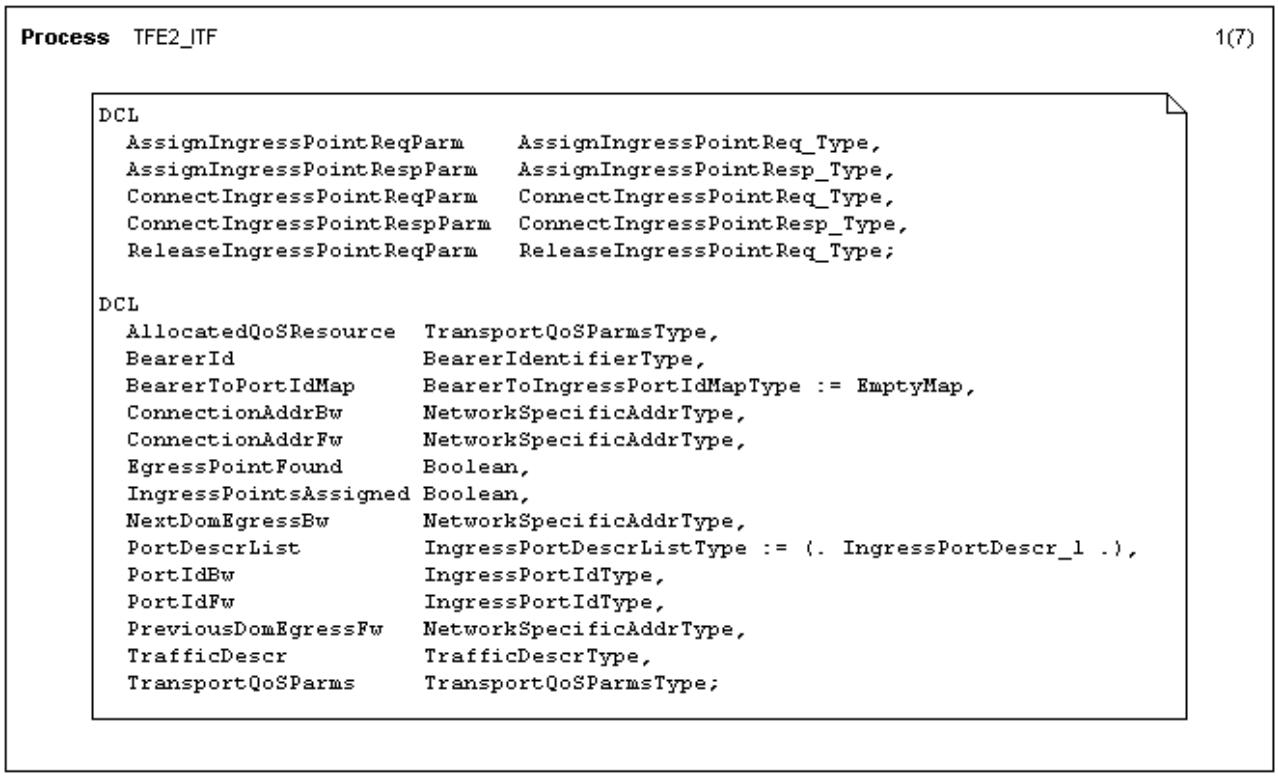


Figure 20: SDL process diagram for functional entity $TFE2_{ITF}$ (1 of 7)

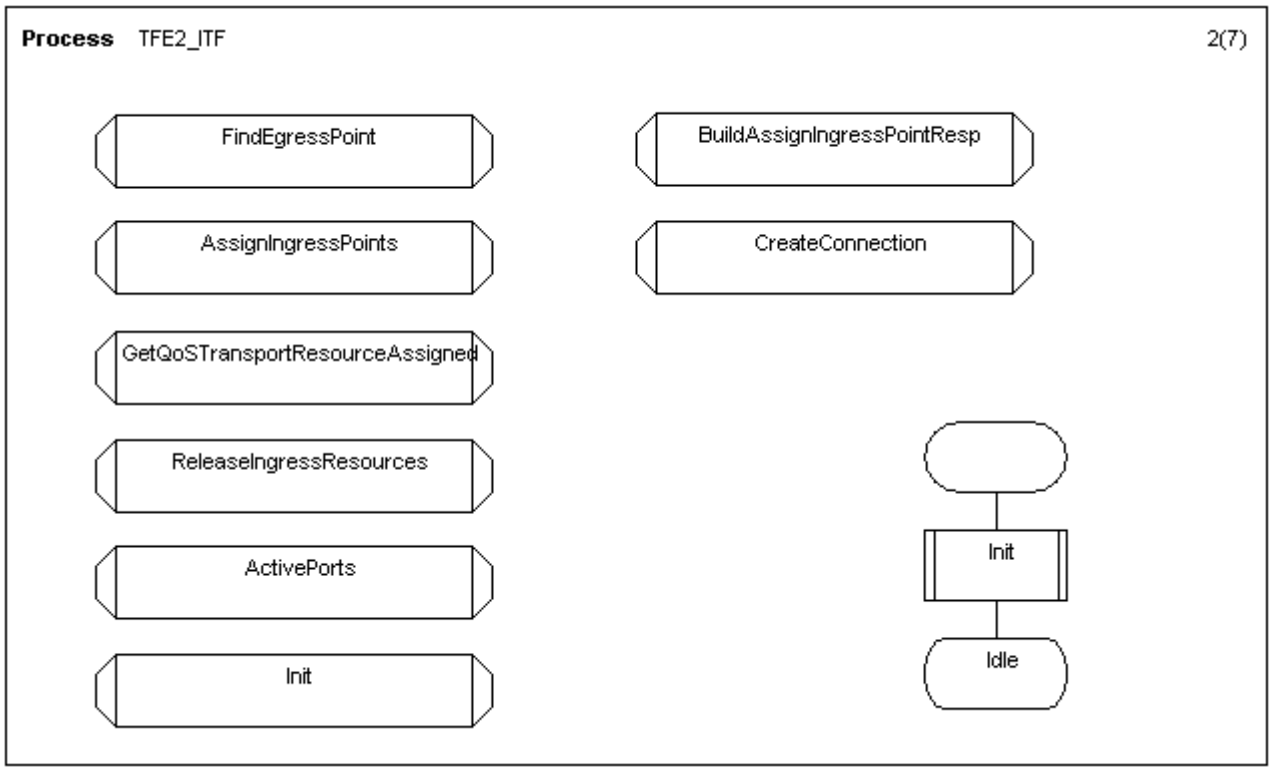


Figure 21: SDL process diagram for functional entity $TFE2_{ITF}$ (2 of 7)

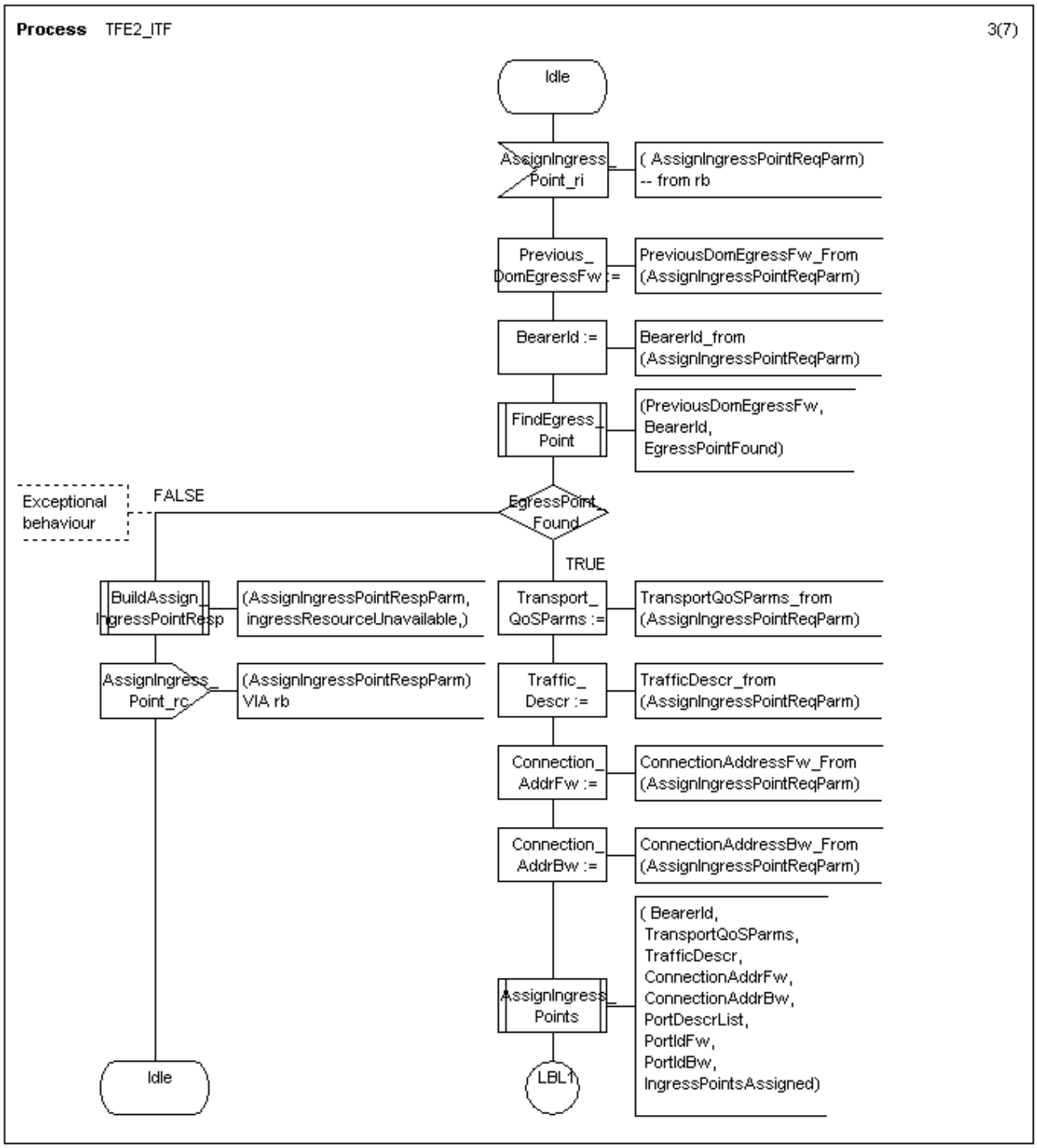


Figure 22: SDL process diagram for functional entity $TFE2_{ITF}$ (3 of 7)

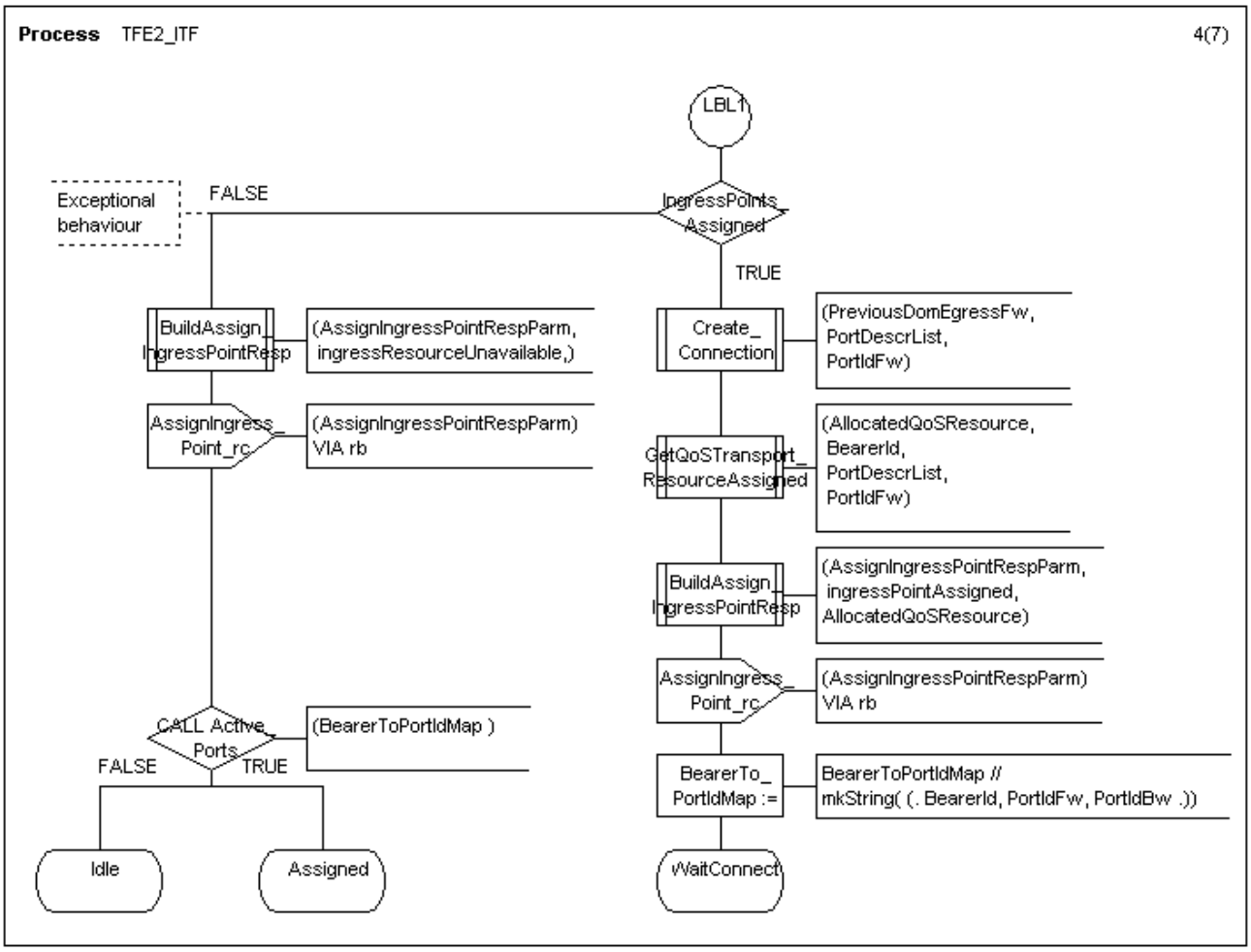


Figure 23: SDL process diagram for functional entity $TFE2_{ITF}$ (4 of 7)

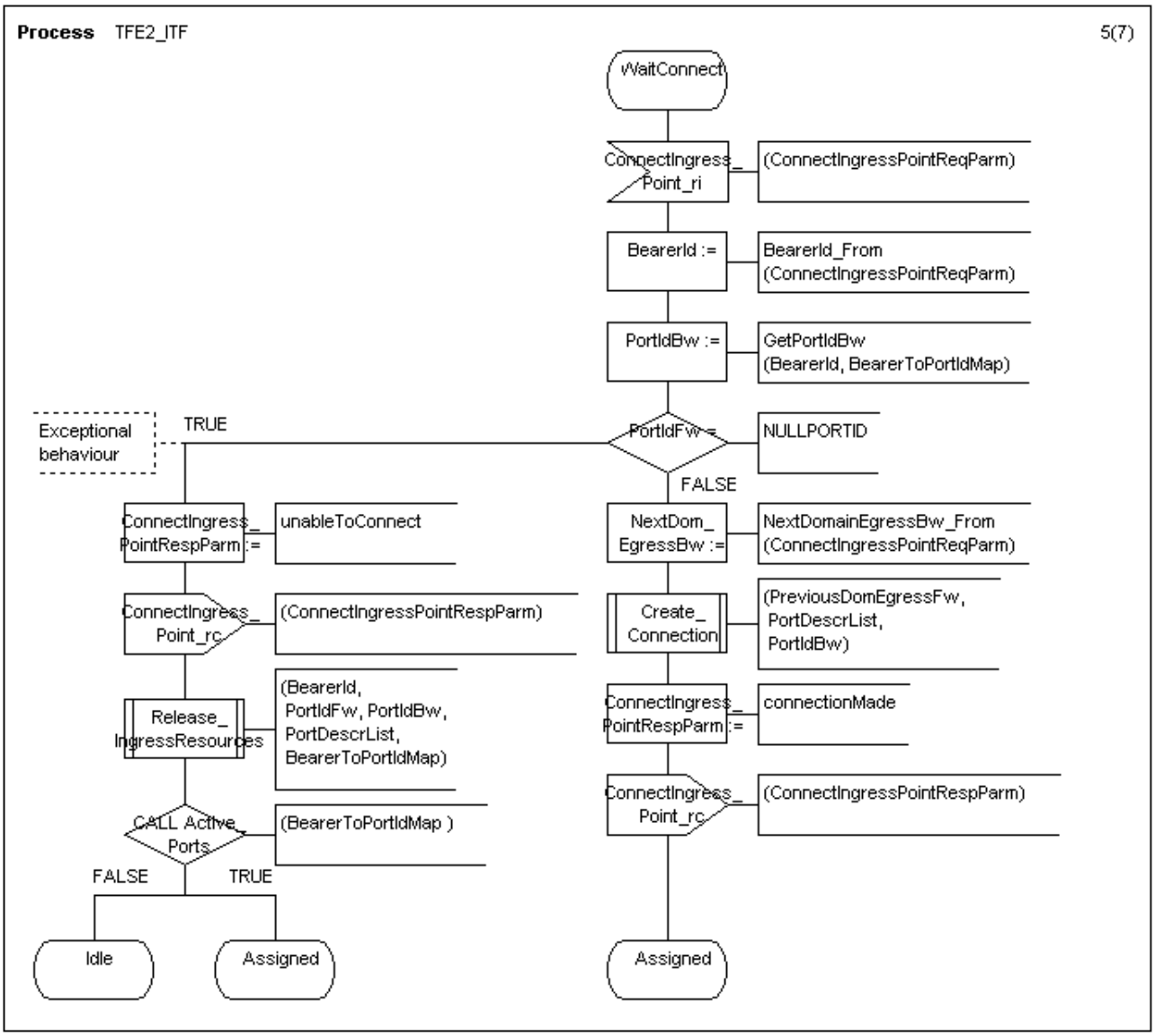


Figure 24: SDL process diagram for functional entity $TFE2_{ITF}$ (5 of 7)

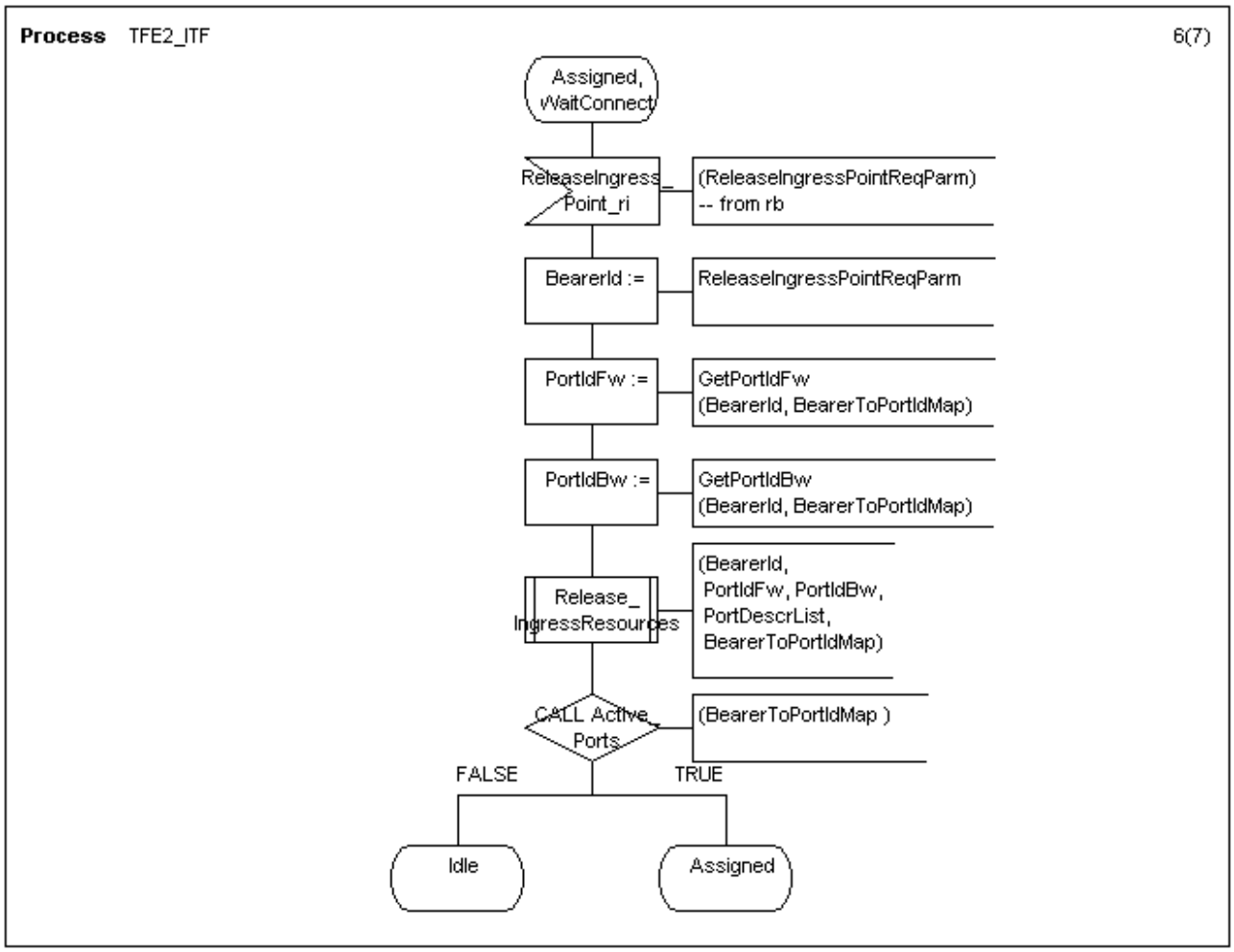


Figure 25: SDL process diagram for functional entity $TFE2_{ITF}$ (6 of 7)

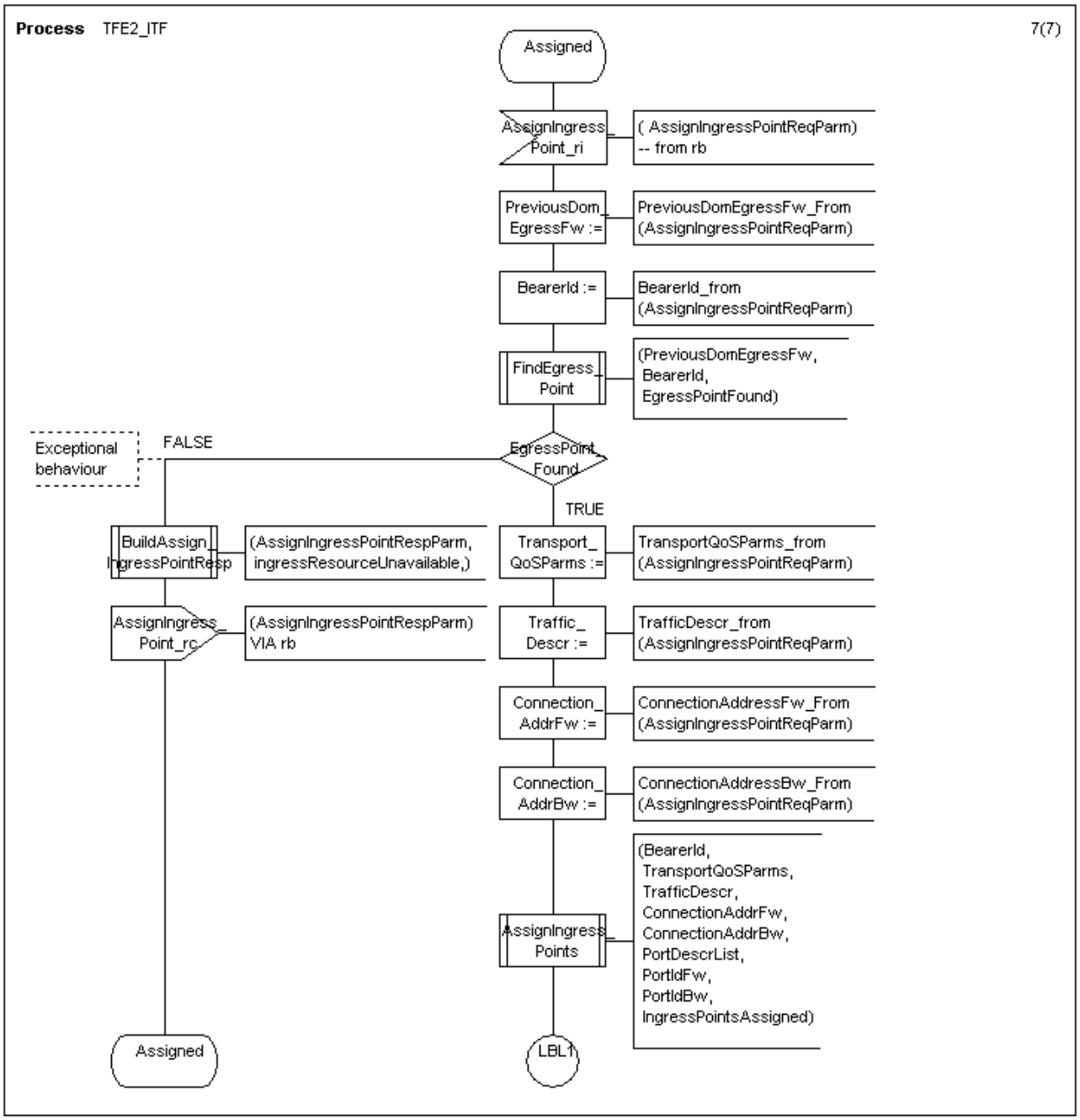


Figure 26: SDL process diagram for functional entity $TFE2_{ITF}$ (7 of 7)

5.4.4 Behaviour of $TFE3_{ETF}$

The behaviour of $TFE3_{ETF}$ is shown in the SDL process diagram in figure 27 to figure 32.

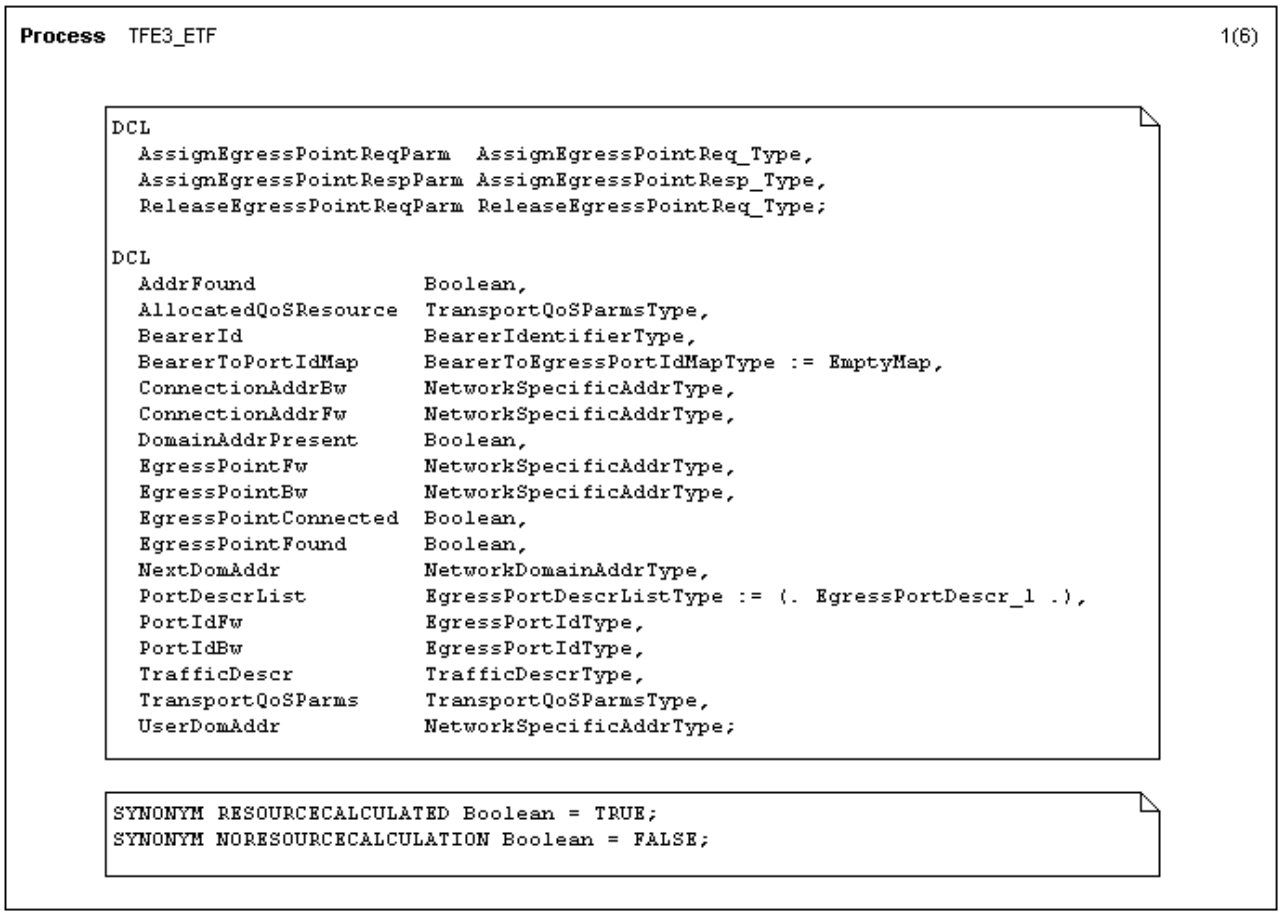


Figure 27: SDL process diagram for functional entity $TFE3_{ETF}$ (1 of 6)

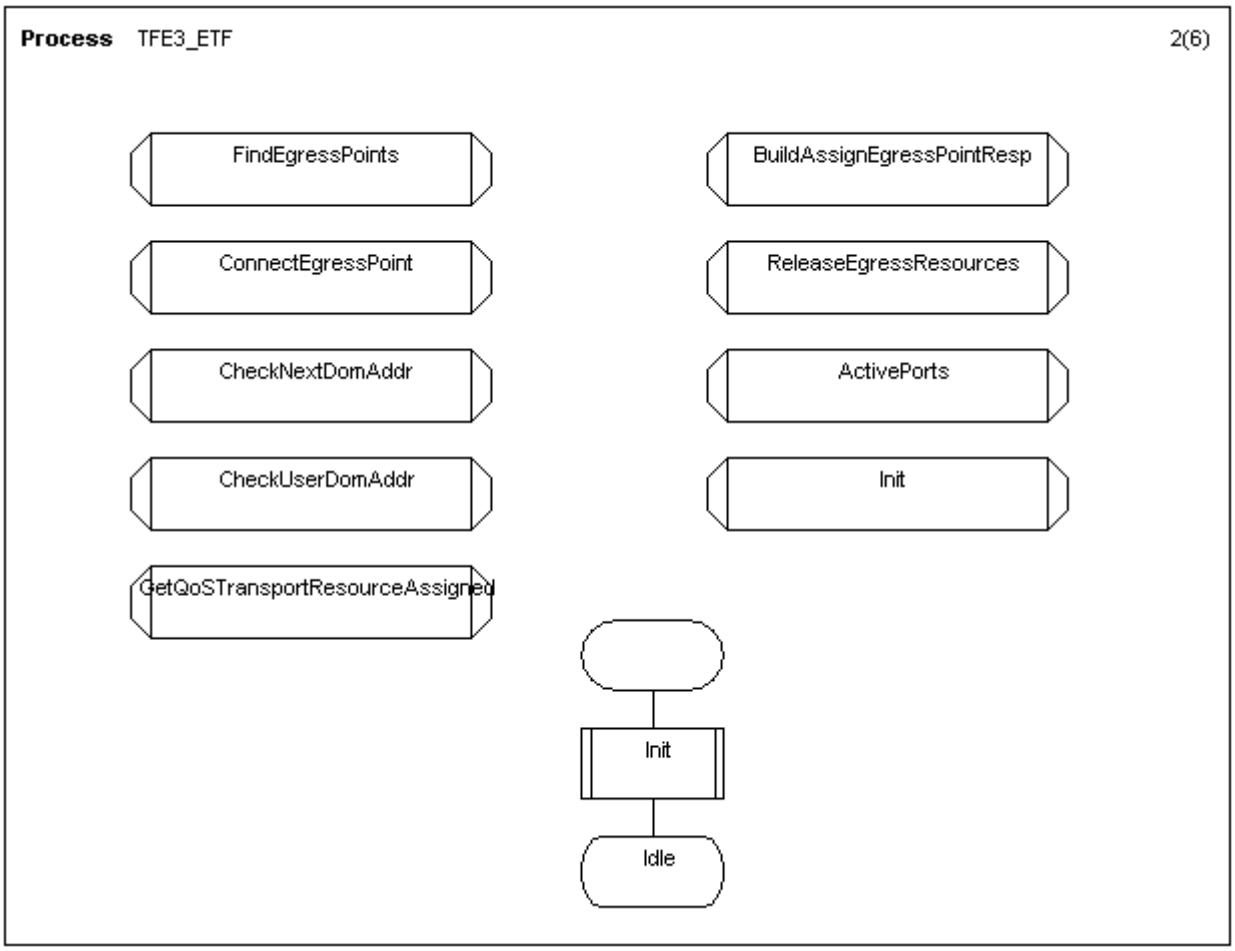


Figure 28: SDL process diagram for functional entity $TFE3_{ETF}$ (2 of 6)

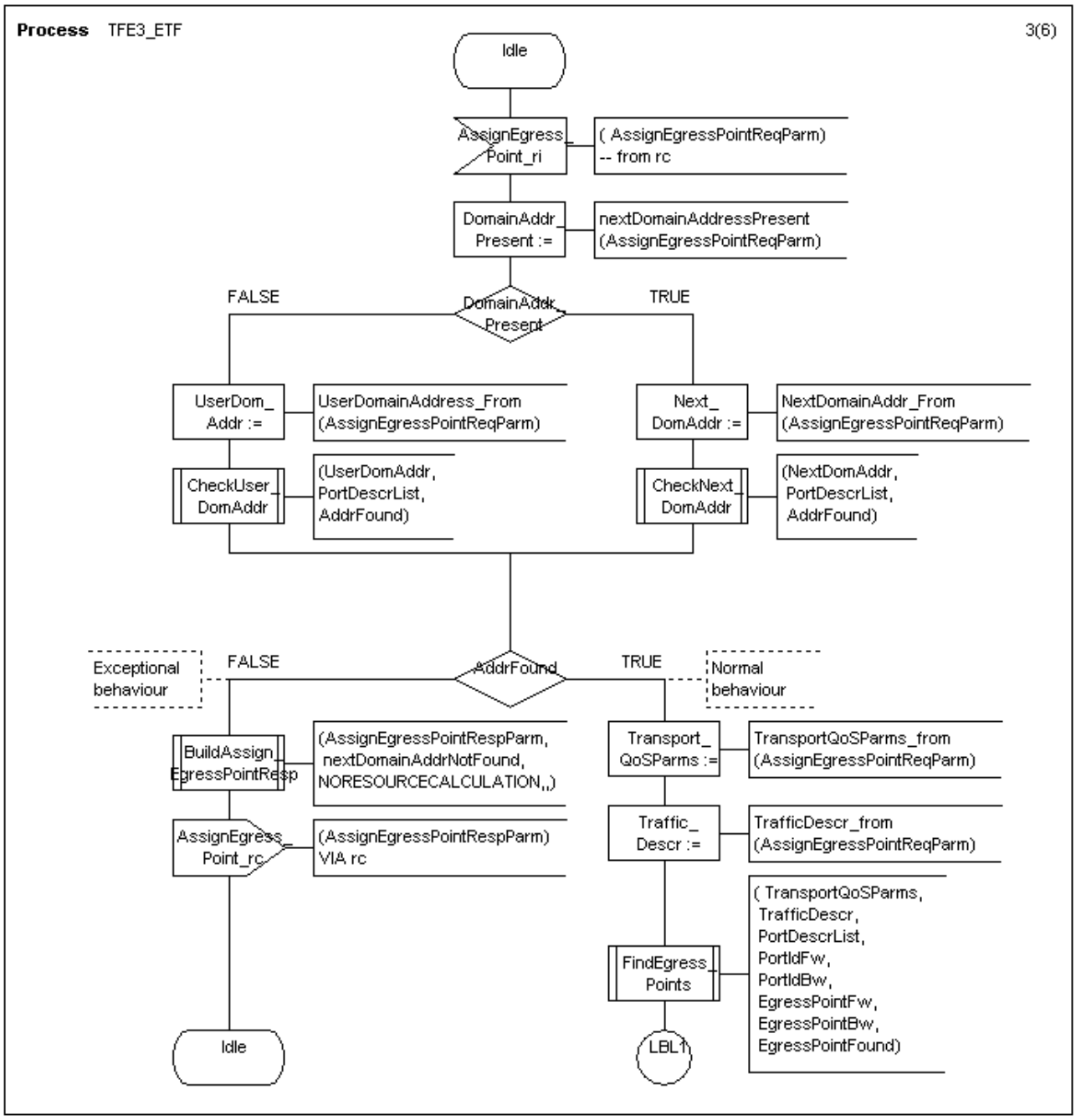


Figure 29: SDL process diagram for functional entity $TFE3_{ETF}$ (3 of 6)

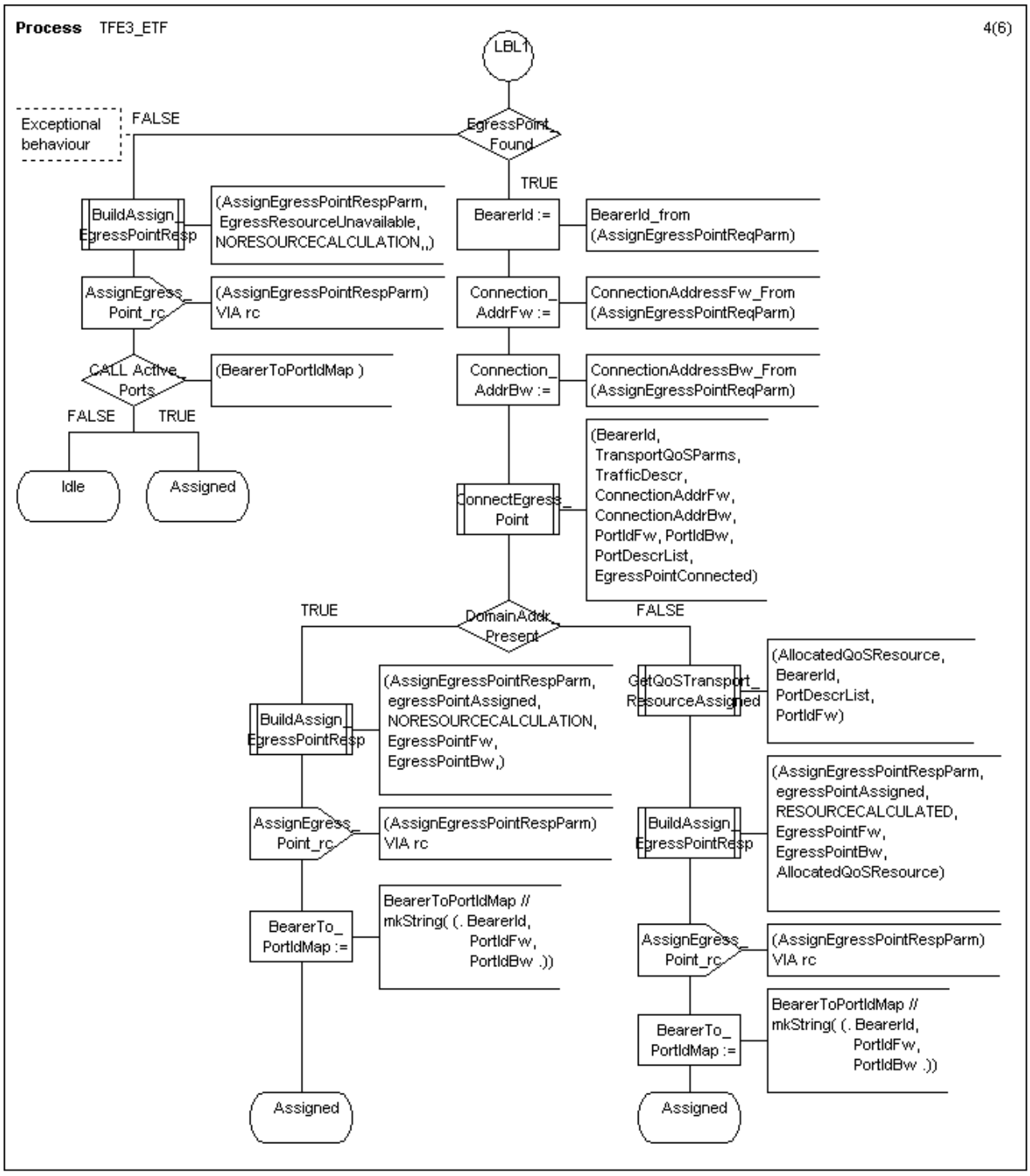


Figure 30: SDL process diagram for functional entity $TFE3_{ETF}$ (4 of 6)

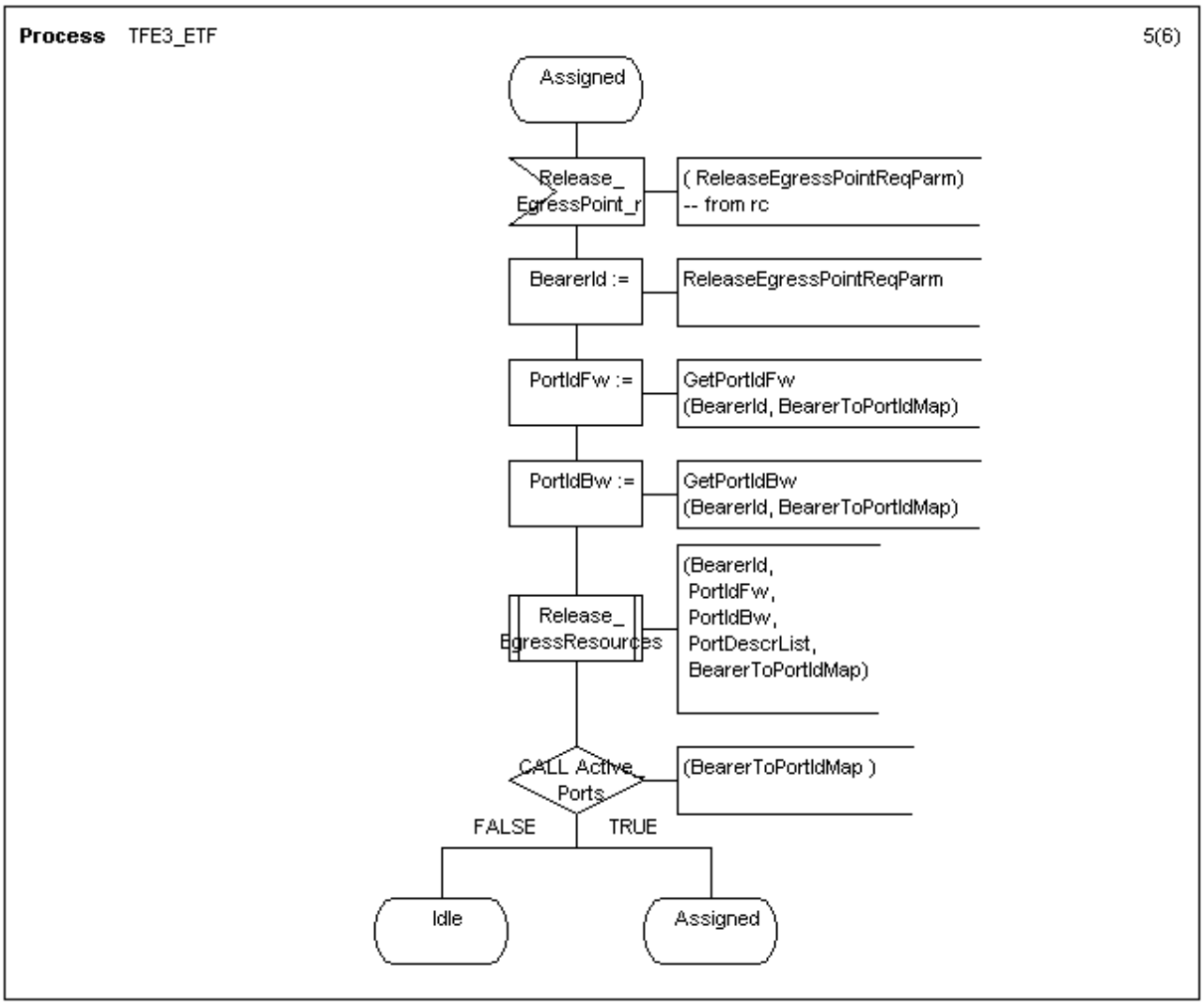


Figure 31: SDL process diagram for functional entity $TFE3_{ETF}$ (5 of 6)

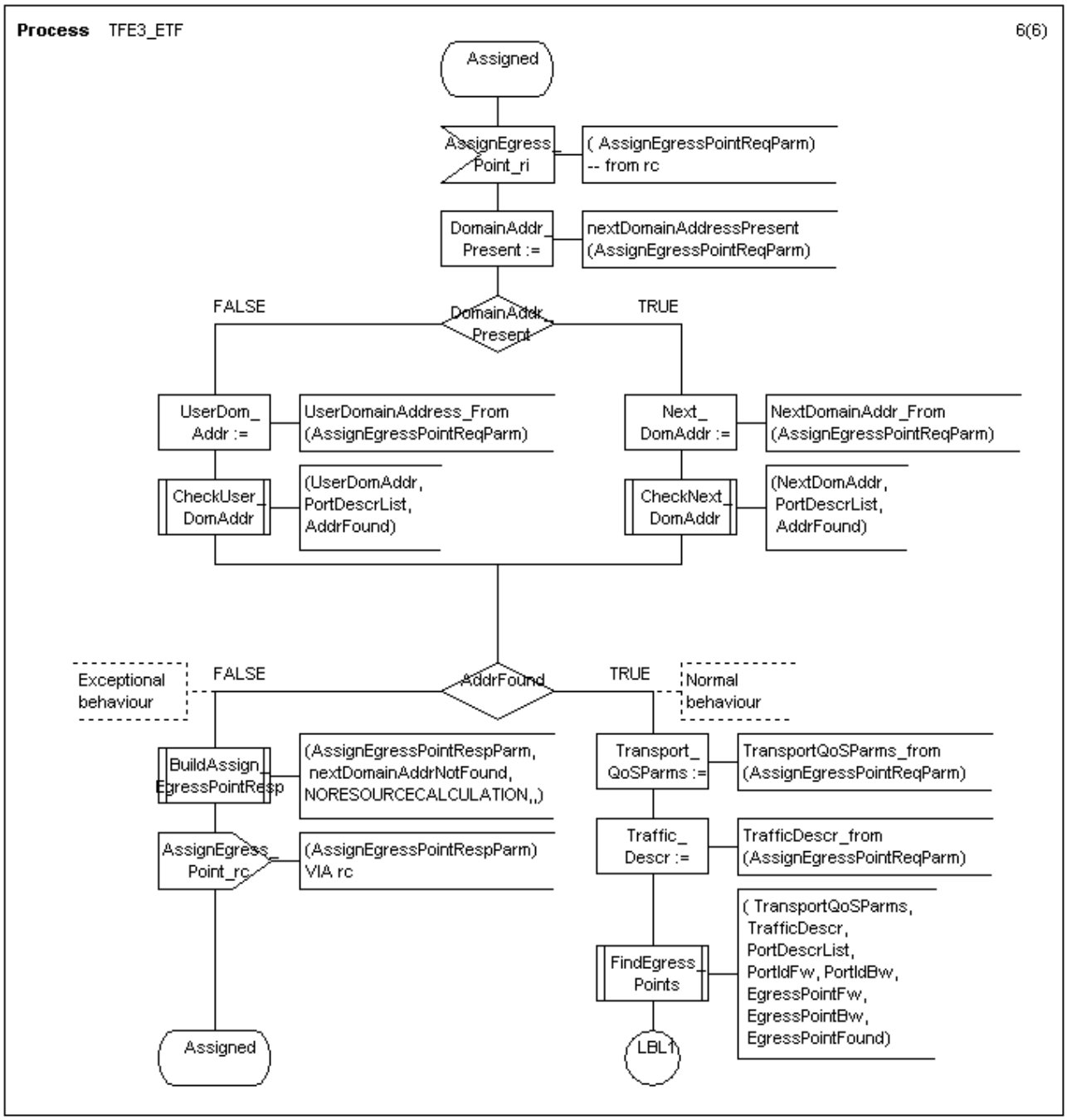


Figure 32: SDL process diagram for functional entity $TFE3_{ETF}$ (6 of 6)

5.5 Allocation of functional entities to domains

TS 101 314 [2] defines an abstract architecture for TIPHON-based on domains and functional groups. In the instantiations (scenarios) of the transport control functional model, the functional entities may be allocated to this architecture.

In all scenarios $TFE1_{TRM}$, $TFE2_{ITF}$ and $TFE3_{ETF}$ are allocated to the transport domain. This allocation may exist in the different functional groups, terminal, serving network, or home network functional group.

Annex A (informative): Simulation and validation SDL model

This annex contains the complete SDL model used for simulation and validation of the transport control service.

It is contained in the archive ts_10188205v040101p0.zip which accompanies the present document.

History

Document history		
V1.1.1	May 2002	Publication as TS 101 882
V4.1.1	November 2003	Publication