The USA casts a vote of DISAPPROVAL of CD 10747, based upon the major technical comments in items 1 and 2. The remainder of the comments are not considered to be major. Upon satisfactory resolution of comments 1 and 2, the USA will change its vote to APPROVAL.

Notwithstanding its vote of DISAPPROVAL, the USA reiterates its strong support for progression of CD 10747. In developing the proposed solutions for items 1 and 2, the USA found that they have no adverse impact on other functions within CD 10747, and thus are believed to be non-contentious.

The USA has classified each of its comments as follows: (M)=major technical, (m)=minor technical, (E)=major editorial, and (e)=minor editorial.

1. ***Precedence of Match with Longest Prefix (M):***

   CD 10747 contains no normative text in 8.1.2.2 stating that a match of an NPDU with a longer prefix should take precedence over a match with a shorter prefix. The USA believes that this omission is an unintentional oversight, since the precedence of matches with longer prefixes is well-known and is an integral part of IS 10589. However, since we found no explicit normative text and the function is essential for correct and unambiguous operation of CD 10747, we classify this as a major technical comment, which can be satisfied by inserting the following sentence as a new last paragraph of clause 8.1.2.2:

   > In cases where a given NPDU matches several address prefixes, the match with the longest prefix shall take precedence.

   Further clarity can also be obtained by adding the following sentence to the first paragraph of related clause 9.4 (which describes how the matching process is used for forwarding NPDUS):

   > ...matches the NPDU-derived Distinguishing Attributes of the incoming NPDU. The incoming NPDU shall be forwarded based on the longest NSAP address prefix that matches (as in 8.1.2.2) the destination NSAP address of the incoming NPDU:

2. ***Handling of Overlapping Routes (M):***

   Within IDRP, a BIS can advertise a set of overlapping routes. Overlapping routes have the same distinguishing attributes, and the respective NLRI fields contain destinations in common. Since NLRI depicts destinations by means of NSAP address prefixes, this means that some prefixes, of different lengths, will be nested inside one another.

   Since the forwarding process selects a next hop for an NPDU based on matching the NPDU′s destination address with an address prefix, there is no ambiguity about what IDRP′s forwarding process will do if a given NPDU matches several address prefixes: IDRP will forward the NPDU based on the match with the longest prefix. However, IDRP contains no normative text to insure that the Decision Process will handle overlapping routes in a way that accurately portrays the actions that will be taken by the Forwarding Process. For example, nothing in CD 10747 prevents a BIS from accepting from a given neighbor BIS only the route with the shortest NSAP address prefix, while rejecting routes from the same neighbor that have longer nested prefixes.

Therefore, we recommend that normative text should be added to CD 10747′s description of the Decision Process to define how a BIS will handle overlapping routes. The following text is proposed:

8.15.3 Route Replacement

If an UPDATE PDU is received carrying a route that matches an earlier route received from the same BIS (identical NLRI and distinguishing attributes), the new route replaces the old route, which becomes unfeasible and shall be deleted from the appropriate Adj-RIB-In.

If an UPDATE PDU is received carrying a route whose NSAP prefix is nested within the prefix of an earlier route advertised by the same adjacent BIS, and the path attributes of both routes are identical, then the newly advertised route shall be placed in the appropriate Adj-RIB-In, and no further actions will be necessary.

8.17.2.1 Overlapping Routes

A BIS may transmit overlapping routes to another BIS (routes with overlapping NLRI). NLRI overlap occurs when a set of destinations are identified in non-matching multiple routes with the same set of distinguishing attributes. Since IDRP encodes NLRI using prefixes, overlaps will always exhibit subset relationships. A route describing a smaller set of destinations (a longer prefix) is said to be *more specific* than a route describing a larger set of destinations (a shorter prefix).

When overlapping routes are transmitted from one BIS to another, the more specific routes shall take precedence, in order from most specific to least specific.

This precedence relationship effectively decomposes less specific routes into two parts:

- a set of destinations described only by the less specific route, and
- a set of destinations described by the overlap of the less specific and the more specific routes

The set of destinations described by the overlap comprise a feasible route that is not in use. If a more specific route is later advertised as being unreachable, the set of destinations described by the overlap will then be reachable using the less specific route.

If a BIS receives overlapping routes, the Decision Process shall not alter the semantics of the overlapping routes. In particular, a BIS shall not accept the less specific route while rejecting the more specific route, because the destinations represented by the overlap will not be forwarded along that route. Therefore, a BIS has the following choices:

1. Install both the less and more specific routes
2. Install the more specific route only
3. Install the non-overlapping part of the less specific route only
4. Aggregate the two routes and install the aggregated route
5. Install neither route

3. ***Breaking Ties (m)***

   The procedure in clause 8.17.1 defines an unambiguous method for breaking ties within the Decision Process, but this method is inconsistent with the intended use of the MULTI-EXIT_DISC attribute described in 8.13.7. The text proposed in Appendix A, “Proposed Text for Breaking Ties” on page 16 eliminates this inconsistency by explicitly describing the role of the MULTI-EXIT_DISC attribute within the tie-breaking process.

   The USA also notes that a need to break ties can also arise within IDRP′s internal update process (8.16.1). Therefore, Appendix A, “Proposed Text for Breaking Ties” on page 16 also outlines similar tie-breaking procedures for use in the internal update process.

4. ***Inputs to the Degree of Preference Function (m)***

   Early discussions in SC6 on inter-domain routeing at the Ottawa meeting in May 1989 noted that if the feasibility of a given route depended on properties of other routes, then oscillations could occur in the route selection process. This premise is already accommodated in IDRP′s clause 8.17, which says that ″The selection process is formalized by defining a function that takes the attributes of a given path as an argument and returns a non-negative integer denoting the degree of preference for the path″.

   This important concept can be further emphasized by adding the following statement to clause 8.17 after the first sentence of the second paragraph: ″The degree of preference function for a given path shall not use as its inputs any of the following: the existence of other routes, the non-existence of other routes, or the path attributes of other routes″.

   Then, the next sentence should be changed as follows: ″Path selection then consists of individual application of the degree of preference function to each feasible path, followed by the choice of the one with the highest degree of preference.″

5. ***Stability of Routes (m)***

   The method outlined in clause 8.11 is limited to the detection of a situation in which a BIS receives a route from an adjacent BIS whose RD_PATH attribute contains one of the RDIs associated with the local BIS. Such a condition typically arises when the advertising BIS is operating with outdated routeing information, and the situation will rectify itself as soon as the advertising BIS receives current routeing information.

   Oscillation in the route selection process can occur if:

   - The receiving BIS decides to advertise a new route in response to the received UPDATE PDU (that contained the outdated information), and
   - The advertising BIS decides to change its selected route before the new UPDATE PDU arrives from the receiving BIS

   These conditions can not persist indefinitely unless there is perfect synchronism between the UPDATE PDUs exchanged by the two BISs--that is, the UPDATE PDUs always cross in transit. However, CD 10747 specifies the addition of ″jitter″ (25%) to the timers used to control the propagation of UPDATE PDUS, thus providing protection against a long-lived situation where the UPDATE PDUs continuously cross in transit.

   Thus, although not harmful, the methods of clause 8.11 appear largely superfluous. Hence, the USA recommends that clause 8.11 be deleted in its entirety. As a byproduct of its deletion, Annex K is no longer pertinent, and should hence be deleted as well.

6. ***External Updates (m)***

   Clause 8.16.2 (External Updates) requires that the procedures of clause 8.16.1 (Internal Updates) must be done before propagating an UPDATE PDU out of a given routeing domain. In reviewing these procedures, the USA notes that there is no need to impose the restriction in 8.16.2 that internal advertisements must be *acknowledged* before the external advertisements can take place.

   The second paragraph of 8.16.2 says that acknowledgement is needed "to insure that consistent information is will be propagated externally". However, since the protocol (see 8.15.2) specifies no corrective action to be taken even if an inconsistency is detected, the mere receipt of an acknowledgement (or lack thereof) does not guarantee consistency. Furthermore, the necessity to wait for an acknowledgement will actually slow down the convergence time of the protocol.

   Since waiting for an acknowledgement offers no guarantee of consistent information but does slow down protocol convergence, the USA recommends that the second paragraph of clause 8.16.2 should be deleted in its entirety.

7. ***Destinations within the RD (m)***

   IDRP contains mechanisms which make it counter-productive for a BIS's Decision Process to select a route to destinations within its own routeing domain if that route would leave the routeing domain and them later re-enter it. For example,

   - The IDRP CLNS Forwarding Process will not forward NPDUs along such a route. or destinations within the RD, IDRP will hand the NPDU over to the intra-domain protocol for forwarding, rather forwarding it out of the routeing domain.

   - If such a path were advertised externally, BISs in other RDs would detect a BISPDU looping error since the RDI of the advertising BIS's domain would appear twice. Therefore the advertised route can not be used by BISs located in other RDs.

   Hence, the USA recommends that there should be a warning note inserted in clause 8.17 stating that selection of such routes is deprecated, as follows:

   Note: The decision process should not select a route to destinations located within the local routeing domain if that route would exit the local local routeing domain and later re-enter it. Such routes would be rejected by other RDs due to the existence of an RD-loop. Furthermore, the IDRP CLNS Forwarding Process will not forward NPDUs (destined to internal destinations) outside of the local RD, but will instead hand them over to the intra-domain routeing protocol.

8. ***Maximum PDU Size of OPEN PDU (m)***

   Until a given BIS receives a valid OPEN PDU from a peer BIS, it does not know the maximum size PDU that its neighbor is willing to accept. Therefore, it is possible that the first OPEN PDU is too large for the neighbor BIS to accept. To avoid this problem, it is suggested that IDRP should define a size for the OPEN PDU that every BIS must be able to handle. Recall that BISPDUs are encapsulated within ISO 8473 NPDUs, and that ISO 8473 can fragment the NPDU as appropriate for specific types of subnetworks.

   It is recommended that IDRP require every BIS to be able to handle all OPEN PDUs whose length is less than or equal to 3000 octets, regardless of the value that the BIS has chosen for its managed object **maximumPDUsize**.

9. ***Retransmission (m)***

   Clause 8.5.3 is deficient in not defining the condition upon which retransmission should be stopped and the BIS-BIS connection should be aborted. We suggest adding the following text:

   > However, if no acknowledgement is received within the time specified in the Hold Time field of the adjacent BIS's OPEN PDU, then the local BIS shall issue a Stop Event, send a CEASE PDU, and enter the CLOSE-WAIT state."

10. ***Route Advertisement Intervals (m):***

    The text of 8.16.3.1 introduces a constant named **MinRouteSelectionInterval**, although in fact the text deals with the frequency with which a route can be advertised. For clarity, the constant should be renamed **MinRouteAdvertisementInterval**, and the procedure should be explained in terms of limiting the rate of advertisement, not selection. Other references within the document should be changed to conform to the new name of this constant.

    The rate limitation applies only to advertisements sent between RDs (that is, over external IDRP connections). This allows internal IDRP connections to converge rapidly, and then only limits the rate at which changes can be advertised to other domains.

    Therefore, to clarify the intended functionality, we recommend that the existing text of clause 8.16.3.1 be replaced with the following text:

    > **8.16.3.1 Frequency of Route Advertisement**
    >
    > The managed object **MinRouteAdvertisementInterval** determines the minimum amount of time that must elapse between advertisements of routes to a particular destination from a single BIS. This rate limiting procedure applies on a per-destination basis, although the value of **MinRouteAdvertisementInterval** is set on a per-BIS basis.
    >
    > Two UPDATE PDUs sent from a single BIS that advertise feasible routes to some common set of destinations received from BISs in other routeing domains must be separated in time by at least **MinRouteAdvertisementInterval**.
    >
    > Since fast convergence is needed within an RD, this procedure does not apply for routes received from other BISs in the same routeing domain. To avoid long-lived black holes, the procedure does not apply to the advertisement of unfeasible routes (that is, containing the UNREACHABLE attribute).
    >
    > This procedure does not limit the rate of route selection, but only the rate of route advertisement. If new routes are selected multiple times while awaiting the expiration of **MinRouteAdvertisementInterval**, the last route selected shall be advertised at the end of **MinRouteAdvertisementInterval**.

11. ***Value of MinRouteAdvertisementInterval (m)***

    The constant value of 30 minutes for the MinRouteAdvertisementInterval shown in Table 4 will not cause the protocol to fail, but it will result in slow convergence--on the average, there will be a 15 minute "dead time" before a newly selected route is advertised. Since inter-domain routes will typically be long-lived, the need to advertise a newly selected route will occur relatively infrequently. However, when a new route is selected, it is desirable to advertise that fact on a timely basis.

    Since there is a trade-off between the frequency of route selection and the convergence time of the protocol, the USA recommends that **MinRouteAdvertisementInterval** should be a parameter rather than a constant. The value of the new parameter should be selectable from a range of values, with a requirement that the minimum acceptable value be 5 seconds. Appropriate GDMO descriptions of the new parameter should be generated.

12. ***IDRP ERROR PDU (m)***

The ERROR PDU is difficult to parse because the presence or absence of the Error Subcode and Data Fields is context sensitive. Therefore, the Error Subcode field should be made mandatory, with a value of 0 defined as ″No_Error_Subcode″.

An editorial correction is needed in 7.4 to change ″NOTIFICATION″ to ″IDRP ERROR PDU″ in the section describing ″Data″.

13. ***NSAP Prefix Encoding (m):***

Although not currently the case, it is possible that future families of NSAP addresses may benefit from allowing prefixes that do not fall on semi-octet boundaries. To provide maximum generality now and avoid having to amend IDRP later, it is beneficial to specify the length of NSAP prefixes in units of bits, rather than semi-octets. To do this, the following changes should be made:

- In clause 7.3, under header ″Network Layer Reachability Information″, subpart ″a″, replace current text with:

  > The length field indicates the length in bits of the following prefix. A length of zero indicates a prefix that matches all NSAPs.
  >
  > **Note:** Although IDRP can express prefixes with a granularity of bits, the use of a bit-level granularity may not be provide consistency with other protocols. For example, IS 10589 expresses its reachable address prefixes with a granularity of semi-octets.

- Strike clause 8.1.2.1 (″An NSAP prefix is said to be of length...″)
- Strike out the last sentence of the first paragraph of 8.1.2.2, and replace it with:

  > An NSAP address prefix which does not extend into the DSP, for which the AFI denotes as address with a fixed length IDI, shall be compared directly against the encoded NSAP address, including any padding characters that may be present. An NSAP address prefix which does not extend into the DSP, for which the AFI denotes an address with a variable-length IDI, shall be compared against NSAP′, which is obtained from the encoded NSAP address by removing all padding characters that were inserted by the binary encoding process of ISO 8348/Add.2.

- Replace item ″a″ of clause 8.1.2.2 with:

  > a) If the encoded NSAP (or NSAP′) contains fewer bits than the NSAP address prefix, then there is no match.

- Replace item ″b″ of clause 8.1.2.2 with:

  > b) If the NSAP (or NSAP′) contains at least as many octets as the NSAP address prefix, and all bits of the NSAP address prefix are identical to the corresponding leading bits of the encoded NSAP address (or NSAP′), there is a match. Otherwise, there is no match.

- Amend the descriptions of SS-QOS, DS-QOS, SS-SEC, and DS-SEC to reflect that the length of the NSAP is denominated in bits for these attributes as well.
- Bring the GDMO descriptions in line with these changes.

14. ***Permissible Sets of Distinguishing Attributes (m):***

The material in clause 8.12.2 is technically correct, but it does not present all the relevant information about permissible sets of distinguishing attributes. The constraints on permissible sets of distinguishing attributes arise from the header encodings of ISO 8473 NPDUs. Since the 8473 encodings limit the number and combinations of distinguishing attributes that can be encoded in a given NPDU, they also limit the distinguishing attributes that can appear in a given UPDATE PDU.

The USA suggests that the last two paragraphs of 8.12.2 be replaced with the following new text:

> A permissible set of distinguishing attributes is defined to be a set that can be derived from information that can be validly encoded in the header of an ISO 8473 NPDU, using the mappings described in 9.2. Therefore:
>
> 1. A single RIB-Atts may include either the SOURCE SPECIFIC SECURITY attribute or the DESTINATION SPECIFIC SECURITY attribute (but not both).
> 2. Only one of the following set of distinguishing attributes {Residual Error Probability, Transit Delay, Expense, Capacity, Source Specific QOS, Destination Specific QOS} may be contained in a given RIB-Att.
>
>   **Note:** The QOS maintenance parameter in the NPDU maps into a single NPDU-Derived Attribute in the set {Transit Delay, Expense, Residual Error, Capacity, Source Specific QOS, Destination Specific QOS}, as described in 9.4 ofCD 10747). This mapping is identical to the one used in IS 10589.
> 3. A single RIB-Atts may include the Priority Distinguishing Attribute.
>
>   **Note:** The Priority attribute is mapped from the 8473 priority parameter in the NPDU.
> 4. A single RIB-Atts may not include any instance of equivalent distinguishing attributes. (Two distinguishing attributes are equivalent if they are both type specific and have the same type or they are both type-value specific and have the same type and same value.)

15. ***Initializing FSM (m)***

To insure that the lifetimes of all BISPDUs have expired before a BIS FSM is re-started after a ″system crash″, clause 8.6.3.1 should require that the BIS must wait for a time period equal to **CloseWaitDelay** after receipt of a Start Event before it enters the OPEN-Sent state.

16. ***Combining Routes (e)***

The last paragraph of clause 6.6 presents only a partial picture of route aggregation, and hence can be confusing if read in isolation from other normative clauses of IDRP. Since this section of IDRP is not normative, it is suggested that this paragraph should be deleted entirely, and the following sentence be added at the end of the previous paragraph:

> For example, it is possible under certain constraints to aggregate path attributes, NLRI, or entire routes, as described more fully in clauses 8.17.5 and its subclauses.

17. ***Version Negotiation (e)***

For clarity, clause 8.7 should mention that the highest version number supported by the local BIS is contained in managed object **version**.

18. ***Handling of SSSEC and DSSEC (e)***

The material in clause 8.12.3, items ″a″ and ″b″ speaks of constraints on the SSQOS and DSQOS attributes. However, the text does not provide equivalent constraints for SS SECURITY and DS SECURITY, which are treated exactly the same way in IDRP. Therefore, two new items should be added to the list, using the text of existing ″a″ and ″b′, with SECURITY substituted for QOS.

19. ***Removal of References to CO/CL and IFU (m)***

At the Berlin meeting of SC6/WG2, it was decided that CD 10747 would address connectionless-mode operations only, routeing only ISO 8473 NPDUs. Therefore, the use of IFUs (see ISO TR 10172) will never be required, except possibly as Network Layer Relays. But in that mode, an IFU has no impact on the operation of CD 10747. Therefore, references in CD 10747 to ″CO/CL″ or ″IFU″ serve no useful purpose. The USA recommends that the editor remove all such references, and provides the following specific instances that we are aware of:

- Delete reference to TR 10172 from clause 2
- Delete clause 4.6 in its entirety
- Delete the acronyms ″CONS″ and ″CO″ from clause5.3
- Delete Type CO/CL 15 in figure 5
- Delete the CO/CL attribute from Table 1
- Delete references to X.25 and ISO 10030 from clause 8.13.3, third paragraph
- Delete clause 8.13.15 in its entirety
- Delete the item ″CO/CL″ from PICS (Table 13)

20. ***Annex L: Common Subnetworks (e)***

It was decided in the Berlin WG2 meeting that IDRP will be a CLNP inter-domain routeing protocol, and will not interface directly to CONS services. Since IDRP interfaces directly to ISO 8473, using it as a SNICP, the material in Annex L is no longer relevant, and it should be deleted.

However, believing it worthwhile to describe how one might ascertain that a pair of BISs are located on the same subnetwork, the USA proposes that the following note be added to clause 6.9.2:

Note: In the absence of an implementation specific method for ascertaining that a neighbor BIS listed in managed object **EXTERNAL-BIS-NEIGHBORS** is located on a common subnetwork with itself, a local BIS can include the ISO 8473 Complete Route Record parameter so that the recipient of the BISPDU can determine whether the sending BIS is adjacent to it.

21. ***Clause 8.17.3 and Clause 8.17.6 (e)***

These clauses are logically out of order with respect to the remainder of clause 8.17. Clause 8.16.3 (Path Selection) should be moved forward, so that it occurs immediately after current clause 8.17; for consistency, the words ″path selection″ should be changed to ″route selection″; and finally, clause 8.17.6 (Interaction with Update Process) should be moved immediately after current 8.17.2, ″Updating the Loc-RIBs″. Then, the clauses will be in a more logical order of presentation: Decision Process, Route Selection, Breaking Ties, Updating the Loc-RIBs, and Interactions with Update Process.

22. ***Clauses 8.17.4 to 8.17.5.6 (e)***

    The material presented in these clauses is not logically part of the Decision Process. They discuss ways to organize routeing information efficiently after it has been selected. Therefore, these clauses be located in a new 2nd level clause, to be entitled ″Efficient Organization of Routeing Information″.

23. ***Contents of Information Bases (e)***

    Although the contents of IDRP′s routeing and forwarding information bases can be inferred from the text, this material is not presented in a single place within the standard. It is suggested that text should be added to existing clause 6.8 (Selecting the Information Bases), and that it should present the table shown in Table 1 on page 10, which collects and summarizes information about the RIBs and FIBs.

24. ***MD4 References (e)***

    The reference to ″MD4″ in Figure 7 is inappropriate since the term ″MD4″ is not defined or mentioned anywhere in the IDRP text. To correct this, the following changes should be made:

    - Change ″MD4 Algorithm″ to ″IDRP Checksum Algorithm″ in figure 7
    - Insert a bibliographic reference to RFC 1186 in clause 3 (″Informative References″)
    - Provide a reference in Annex B to the algorithmic description part of RFC 1186.

25. ***Attribute Numbering (e)***

    There is inconsistent numbering of path attribute types in 7.4 and Table 1. The existing type numbers should be corrected as follows: SS SEC=17, DS SEC=18, CAPACITY=19, PRIORITY=20. The editor should also assure that the remainder of the text is also numbered consistently.

26. ***RDIs (e)***

    The use of length fields with respect to RDIs is not consistent. In particular, the OPEN PDU uses a length in octets, while the DIST_LIST_INCL and DIST_LIST_EXCL attributes use semioctets.

    Since RDIs must be valid NSAPs, they are always encoded as octets. Thus the description of the DIST_LIST_EXCL and DIST_LIST_INCL attributes on page 15 should be changed to say that the length is in octets.

    Furthermore, the term ″RDI prefix″ occurs at least in these same two places. This term is incorrect; RDIs are not prefixes, nor are they ever abbreviated. The term ″RDI″ should replace ″RDI prefix″ wherever it occurs.

27. ***Header Length of BISPU (e):***

    The length of the fixed header is 29 octets, but in several places in CD 10747, the value 31 is used in error. It should be changed in the following places:

    - first sentence on page 18
    - In 7.4, change ″...IDRP ERROR PDU is 32...″ to ″...IDRP ERROR PDU is 30...″.
    - In 7.5, change 31 octets to 29 octets
    - In 7.6, change 31 octets to 29 octets

28. ***GDMO Notation (m, E):***

To align the GDMO description in clause 12 with both DIS 10733 and IS 10589, the USA suggests that clause 12 should be amended as shown in Appendix B, "Revised GDMO for CD 10747" on page 18. This appendix reproduces the text of CD 10747, with additions and deletions marked with revision characters or strikethroughs, as appropriate. The bulk of the new technical material deals with importing communicationsAlarm notifications. The remainder of the changes are editorial in nature: for example, consecutive numbering of parameters, elimination of duplicate text, elimination of duplicate numbering of parameters, etc.

Table 1. The IDRP Information Bases. The indexing variables and contents of the RIBs and FIBs are shown.

Information Base	***Indexed by...***	***Contains...***
Adj-RIB-In	• NET of adjacent BIS • RIB-Atts	• Path attributes • NLRI
Loc-RIB	• RIB-Atts	• Path attributes • NLRI
Adj-RIB-Out	• NET of adjacent BIS • RIB-Atts	• Path attributes • NLRI
FIB	• RIB-Atts • NLRI	• NET of next hop BIS • Output SNPA of local BIS • Input SNPA of next hop BIS

Notes:

1. As a local option, a BIS may elect to apply information reduction techniques to path attributes and NLRI information.
2. For each adjacent BIS, a given BIS maintains an Adj-RIB-In for each RIB-Att (including the Empty RIB-Att) that it supports.
3. A BIS maintains a separate Loc-RIB for each RIB-Att (including the Empty RIB-Att) that it supports.
4. For each adjacent BIS, a given BIS maintains an Adj-RIB-Out for each set of RIB-Atts (including the Empty RIB-Att) that it advertises to that neighbor.
5. A given BIS maintains a separate FIB for each set of RIB-Atts (including the Empty RIB-Att) that it has advertised to its neighbor BISs—that is, each FIB corresponds to an Adj-RIB-Out.

   To facilitate the forwarding process, a BIS can organize each of its FIBS into two conceptual parts: one containing information for NLRI located within its own RD, and another for NLRI located in other RDs (see clause 9). For external NLRI, a BIS can further organize the FIB information based on whether the next-hop-BIS is located within its own RD or in another RD (see clause 9.4, items "a" and "b"). And finally, for those next-hop BISs located in its own RD, the local BIS can organize the information according to a specific forwarding mechanism (see clause 9.4, items "b1", "b2", and "b3").

29. ***Authentication of BISPDUs with Malformed Headers:*** In clause 8.9, expand the note to indicate that a BISPDU with a malformed header will be discarded, and therefore the authentication procedures will not be applied to it.

30. ***OPEN PDU Handling:***

    In clause 8.6.3.2 (OPEN-RCVD State), the remote BIS may also send an OPEN PDU with acknowledgement to acknowledge receipt of the local BIS′s OPEN PDU. Therefore, the following changes should be made to clause 8.6.3.2:

    > The local BIS shall wait in OPEN-RCVD state for the remote BIS to send an OPEN PDU, KEEPALIVE or IDRP ERROR PDU to acknowledge receipt of the local BIS′s OPEN PDU. Upon receipt of an acknowledgement, the local BIS shall take the following actions:
    >
    > 1. If the OPEN PDU received from the remote BIS acknowledges the local BIS′s OPEN PDU, then:
    >   - If the incoming OPEN PDU successfully passed local error checking, as defined in 8.19.2, the local system shall acknowledge the incoming OPEN PDU by sending a KEEPALIVE PDU. The local BIS shall change its state to ESTABLISHED.
    >   - If the incoming OPEN PDU has any of the errors described in 8.19.6, the local system shall send the IDRP ERROR PDU (if required by the local error checking procedure) to acknowledge the receipt of the OPEN PDU. The local BIS shall then change its state to CLOSE-WAIT
    > 2. If the OPEN PDU received from the remote BIS does not contain an acknowledgement of the OPEN PDU sent by the local system then:
    >   - If the incoming OPEN PDU successfully passed local error checking, then the local BIS shall resend its own OPEN PDU with the same sequence number, but shall also include an acknowledgement of the remote BIS′s OPEN PDU.
    >   - If the incoming OPEN PDU fails to pass the local error checking, the local system shall send the IDRP ERROR PDU (if required by the local error checking procedure) to acknowledge the receipt of the OPEN PDU. The IDRP ERROR PDU shall contain the same sequence number used for its previously issued OPEN PDU to that BIS, and shall contain an acknowledgement of the remote BIS′s OPEN PDU. The local BIS shall then change its state to CLOSE-WAIT.″

31. ***Additional BISPDU Error Handling (m):***

    In clause 8.19, there are no error handling procedures given for CEASE, KEEPALIVE and RIB REFRESH PDUs. Although the procedures are implied for KEEPALIVE and CEASE PDUs in clause 8.19.1, we suggest the following sections be added for clarity:

    > **8.19.6 KEEPALIVE PDU Error Handling**
    >
    > The KEEPALIVE PDU consists of only the BISPDU Header. Error conditions are handled according to 8.19.1.
    >
    > **8.19.7 CEASE PDU Error Handling**
    >
    > The CEASE PDU consists of only the BISPDU Header. Error conditions are handled according to 8.19.1.
    >
    > **8.19.8 RIB REFRESH PDU Handling**

If any of the following error conditions are detected, the BIS shall issue an IDRP ERROR PDU with the following error indications:

- Invalid OpCode not in Range 1 to 3: indicate RIB REFRESH error with error subcode ″Invalid OpCode″
- Receipt of an OpCode 3 (RIB Refresh End) without prior receipt of OpCode 2 (Rib Refresh Start): indicate FSM Error
- Receipt of an unsupported RIB-Att in the Rib-Atts variable length field in the RIB FRESH PDU for a RIB Refresh Start OpCode: indicate RIB REFRESH error with error subcode ″Unsupported RIB-Atts″

32. ***FSM Error Code (m):***

Add a new error code to the IDRP ERROR PDU (clause 7.4). Its name is ″FSM_Error″ and its code is 4. Also, add a new 1 octet long error subcode for this error: the first semi-octet should contain the type number of the BISPDU that generated the error condition, and the last semi-octet should contain the number of the state of the local BIS′s FSM when the error was detected (1=CLOSED, 2=OPEN_RCVD, 3=OPEN-SENT, 4=CLOSE-WAIT, and 5=ESTABLISHED). Adjust GDMO as appropriate.

33. ***FSMS: Actions on Receipt of BISPDUs (m):***

In clauses 8.6.3.1, 8.6.3.2, 8.6.3.3, 8.6.3.4 the states (CLOSED, OPEN-SENT, OPEN-RCVD, ESTABLISHED and CLOSE-WAIT) describe the actions for receipt of expected PDU types. There is no description of the actions to be performed for receipt of unexpected BISPDU types for each state (and in which the BISPDU contains no internal errors). It is suggested that the following material should be added to clarify the expected actions:

8.6.2 CLOSED State

The BIS shall remain in the CLOSED state until it receives a Start Event. It shall then send an OPEN PDU to the remote BIS, and shall change its state to OPEN-SENT.

Any IDRP ERROR PDUs, UPDATE PDUs, KEEPALIVE PDUs, CEASE PDUs, or RIB REFRESH PDUs received while the BIS is in the CLOSED state shall be discarded, and the BIS shall remain in the CLOSED state.

If an OPEN PDU PDU is received, the local BIS shall send a CEASE PDU and remain in the CLOSED state.

8.6.3.1 OPEN-SENT State

e) If the BIS receives a KEEPALIVE, UPDATE, or RIB REFRESH PDU, the BIS shall issue an IDRP ERROR PDU, indicating ″FSM Error″, and shall then enter the CLOSE-WAIT state.

f) If the BIS receives a CEASE PDU, the BIS shall change its state to CLOSE-WAIT.

g) If the BIS detects any OPEN PDU error conditions (see 8.19.2), it shall send the appropriate IDRP ERROR PDU to the remote BIS, and shall close the connection according to 8.6.4.

h) If the BIS detects any IDRP ERROR PDU error conditions (see 8.19.4), it shall send a CEASE PDU to the remote BIS, and shall then close the connection according to 8.6.4.

8.6.3.2 OPEN-RCVD State

-If the BIS receives a CEASE PDU, the BIS shall change its state to CLOSE-WAIT

-If the BIS detects any OPEN PDU error conditions (see 8.19.2), it shall send the appropriate IDRP ERROR PDU to the remote BIS, and shall then enter the CLOSE-WAIT state.

- If the BIS receives a RIB REFRESH or UPDATE PDU with correct acknowledgement, the BIS shall change its state to ESTABLISHED and send a KEEPALIVE PDU to the remote BIS.

- If the BIS receives an OPEN PDU with incorrect acknowledgement, the BIS shall resend its OPEN PDU with acknowledgement to the remote BIS.

- If the BIS detects any IDRP ERROR PDU error conditions (see 8.19.4), it shall send a CEASE PDU to the remote BIS, and shall then enter the CLOSE-WAIT state.

- If the BIS detects any KEEPALIVE PDU error conditions, they shall be handled according to 8.19.6.

- If the BIS detects any CEASE PDU error conditions, they shall be handled according to 8.19.7.

8.6.3.3 ESTABLISHED State

If the BIS receives a KEEPALIVE PDU, UPDATE PDU, or RIB REFRESH PDU, the BIS shall restart its Hold Timer.

If the BIS receives an UPDATE PDU, the BIS shall perform the actions provided in clause 8.15 to update the appropriate Adj-RIB-In with the new routing information, and shall restart its Hold Timer.

If the BIS receives a RIB REFRESH PDU, the BIS shall perform the actions provided in clause 8.10.3 to refresh the appropriate Adj-RIB-In for the local or remote BIS, and shall restart its Hold Timer.

-If the BIS detects any OPEN PDU error conditions (see 8.19.2), it shall send the appropriate IDRP ERROR PDU to the remote BIS, and shall then enter the CLOSE-WAIT state.

If the BIS receives an UPDATE PDU with internal errors, the BIS shall send an IDRP ERROR PDU to the remote BIS and shall change its state to CLOSE-WAIT.

- If the BIS detects any IDRP ERROR PDU error conditions (see 8.19.4), it shall send a CEASE PDU to the remote BIS, and shall then close the connection according to 8.6.4.

- If the BIS detects any KEEPALIVE PDU error conditions, they shall be handled according to 8.19.6.

- If the BIS detects any CEASE PDU error conditions, they shall be handled according to 8.19.7.

- If the BIS detects any RIB REFRESH PDU error conditions, they shall be handled according to 8.19.8.

8.6.3.4 CLOSE-WAIT State

If the BIS receives any BISPDU, the BIS shall ignore them and remain in the CLOSE-WAIT state.

34. ***Effect of Events on IDRP FSMs (e):***

The effects that various events have on the FSMs are described in CD 10747, but these descriptions are scattered throughout the document. Clarity can be improved if there is a single section which collects this information. The following informative text is suggested:

8.6.5 Event Effects on FSM

System generated events and timer expiration events may effect the state of a connection. Upon receipt of the following events, the following actions occur:

a. Start-Event: As defined in clause 8.6.3, upon delivery of a Start-Event, an OPEN PDU is sent to the remote BIS and the state changes to OPEN-SENT.

b. Stop-Event: As defined in clause 8.6.3.1, 8.6.3.2, 8.6.3.3 and 8.6.3.4, upon delivery of a Stop-Event for a BIS-BIS connection, the local BIS sends a CEASE PDU to the remote BIS, and enters the CLOSE-WAIT State.

c. CLOSE-WAIT Timer Expiration Event: An connection remains in the CLOSE-WAIT state for a given amount of time, and then returns to the CLOSED State. When a Close-Wait Timer expires, a Close-Wait Timeout Event occurs, and the connection returns to the CLOSED State.

d. Hold Timer Expiration Event: As defined in clause 8.19.5, if a system does not receive successive KEEPALIVE, RIB REFRESH, or UPDATE PDUs within the period specified in the Hold Time field of the OPEN PDU, then an IDRP ERROR PDU with error code Hold_Timer_Expired is sent to the remote BIS.

e. MinRouteAdvertisementInterval Timer Expiration Event: As defined in clause 8.16.3.1, the architectural constant MinRouteAdvertisementInterval determines the minimum amount of time that must elapse between advertisements of preferable routes received by a given BIS from systems located in other routing domains. If a BIS has selected new routes based on updates from BISs in adjacent RDs, but have not yet advertised them because this interval has not yet elapsed, the receipt of routes from other BISs in its own RD forces the MinRouteAdvertisementInterval timer to expire, and triggers a new selection process that will be based on both updates from BISs in the same RD and in adjacent RDs. No state changes occur as a result of this event.

f. MinRDOriginationInterval Timer Expiration Event: As defined in clause 8.16.3.2, the architectural constant MinRDOriginationInterval determines the minimum amount of time that must elapse between successive advertisements of UPDATE PDUs that report changes within the advertising BIS's own routing domain. No state changes occur as a result of this event.

35. ***Tabular Presentation of IDRP FSM (E):***

As an alternative (or a complement) to the text presented in the previous comments, the editor is asked to consider developing a table that succinctly presented the effects that receipt of inbound BISPDUs, events, and error conditions have on the FSMs.

36. ***Acknowledgement of CEASE PDUs (m):***

    There is no need to acknowledge CEASE PDUs, since the sender will have begun to terminate the connection. Therefore, the end of the first sentence of 7.6, beginning with ″..and then wait...″ should be deleted.

37. ***Miscellaneous (e):***

    a. Add the words ″or routeing domain confederation″ to the end of the first sentence of 8.1.2.
    b. **MaxRIBIntegrityCHeck**= = = > **MaxRIBIntegrityCheck**
    c. In Annex N, change ″Set STATE=CLOSED″ to ″Set STATE=CLOSED-WAIT″ (pages 88, 90, 93)
    d. In Annex G, adjust notation to reflect that last RD traversed is listed last in the RD_PATH attribute (several places):
        - Page 76, 2nd paragraph: ″right to left″ = = > ″ left to right″ and ″leftmost″ = = > ″ rightmost″
        - Notation <X.*>==> <.*X>, several places on pages 76 and 77 for several values of ″X″
        - server== routeServer
        - Clause 8.6.3.1, item ″a″: end item with the words ″by the local sytem:″, and delete the remainder
        - Clause 8.6.3.4, 3rd paragraph: ″at most″ = = > ″ for″
        - In 8.16.3.3, change ″1-J to J″ to ″1-J to 1″

Appendix A. Proposed Text for Breaking Ties

The following text is suggested as a replacement for the existing text of 8.17.1 (see comment #3):

8.17.1 Breaking Ties among Routes with Equal Degrees of Preference

In its Adj-RIBs-In, there may be several routes to the same destination that have the same degree of preference and also have an equivalent set of distinguishing attributes. The local BIS can select only one of these routes for inclusion in the associated Loc-RIB. The local BIS considers all equally preferable routes, both those received from BISs located in adjacent RDs and those received from other BISs located in the local BIS′s own RD.

Ties shall be broken according to the following rules:

1. If the candidate routes have identical path attributes or differ only in the NEXT_HOP attribute, select the route that was advertised by the BIS in an adjacent routeing domain whose NET has the lowest value. Otherwise, select the route that was advertised by the BIS in the local routeing domain whose NET has the lowest value.
2. If the candidate routes differ only in their NEXT_HOP and MULTI-EXIT_DISC attributes, and the local BIS′s managed object **Multiexit** is TRUE, select the route that has the lowest value of the MULTI-EXIT_DISC attribute.

   If the managed object **Multiexit** is false, select the route advertised by the BIS in an adjacent RD whose NET has the lowest value. Otherwise, select the route that was advertised by the BIS in the local routeing domain whose NET has the lowest value.
3. If the candidate routes differ in any path attributes other than NEXT_HOP and MULTI-EXIT_DISC, select the route that was advertised by the BIS whose NET has the lowest value.

For purposes of determining the lowest-valued NET, each binary-encoded NET shall be padded with trailing 0′s in order to bring its length up to 20 octets. The encoded (and possibly padded) NETs shall then be treated as unsigned binary integers.

To clarify the role of tie-breaking in the internal update process, the following changes are suggested in clause 8.16.1:

- Add a new item to the numbered list under item ″b″: ″the newly received route is selected as a result of breaking a tie between several routes, each of which have the highest degree of preference, the same destinations, and the same distinguishing attributes″
- Add the following new clause, numbering it as 8.16.1.1:

  8.16.1.1 Breaking Ties in the Internal Update Process

  When a local BIS has connections to several BISs in adjacent domains, there will be multiple Adj-RIBs-In associated with these neighbors. These Adj-RIBs-In might contain several equally preferable routes to the same destination, all of which have the same set of distinguishing attributes and all of which were advertised by BISs located in adjacent routeing domains. The local BIS shall select one of these routes, according to the following rules:

1. If the candidate routes differ only in their NEXT_HOP and MULTI-EXIT_DISC attributes, and the local BIS′s managed object **Multiexit** is TRUE, select the route that has the lowest value of the MULTI-EXIT_DISC attribute.
2. In all other cases, select the route that was advertised by the BIS whose NET has the lowest value.

For purposes of determining the lowest-valued NET, each binary-encoded NET shall be padded with trailing 0′s in order to bring its length up to 20 octets. The encoded (and possibly padded) NETs shall then be treated as unsigned binary integers.

Appendix B. Revised GDMO for CD 10747

The following is the marked-up text referred to in comment #28:

12.0 System Management and GDMO Definitions

The operation of the inter-domain routeing functions in a BIS may be monitored and controlled using System Management. This clause contains management specification for IDRP, expressed in the GDMO notation defined in ISO 10165-4.

12.1 Name Binding

iSOxxxx-NB **NAME BINDING**

SUBORDINATE OBJECT CLASS idrp_config
NAMED BY
SUPERIOR OBJECT CLASS ″ISO/IEC xxxx″: networkEntity;
WITH ATTRIBUTE ″ISO/IEC xxxx″: idrp_config_MO_Name
CREATE with-automatic-instance-naming iSO-xxxxx-NB-pl;
DELETE on-if-no-contained-objects;
REGISTERED AS {ISO xxxxx-IDRP.nboi ISOxxxx-NB (1)};

adjacentBIS **NAME BINDING**

SUBORDINATE OBJECT CLASS adjacentBIS
NAMED BY
SUPERIOR OBJECT CLASS idrp_config
WITH ATTRIBUTE BIS-NET;
DEFINED AS This name binding attribute identifies a BIS to BIS connection information block. One of these blocks of data should exist per remote BIS that this local BIS exchanges BISPDUs with.;
REGISTERED AS {ISO xxxx-IDRP.nboi adjacentBIS (2)};

12.2 Local BIS Managed Objects for IDRP

idrp_config **MANAGED OBJECT CLASS**

DERIVED FROM ″ISO/IEC xxxxxx″: top

CHARACTERIZED BY localbispackage
PACKAGE
BEHAVIOUR
iDRPBasicImportedAlarmNotifications-B
BEHAVIOUR DEFINED AS Imports the communicationsAlarm notification from ISO/IES 10165-2. It is used to report the following protocol events:

errorBISPDUsent: generated when a BISPDU is received with an error in its format. In addition to the parameters specified by ISO/IEC 10733, the following information will be reported in the AdditionalInformation field for the BIS Connection on which the error BISPDU was received:

1. RemoteBIS-NET for BIS-BIS connection—using the **notificationRemoteBIS-NET** parameter
2. BISPDU error code (see 7.4 and 8.19)—this reports the error code that will be sent in the ERROR PDU using the parameter **notificationBISpduerrorcode**.
3. BIS error subcode (see 7.4 and 8.19)—this reports the subcode that will be sent using the parameter **notificationBISerrorsubcode**.
4. BISPDU error information (see 7.4 and 8.19)—this reports the data from the received BISPDU that will be used to diagnose the problem for the Notification. The parameter **notificationBISpuderrorinfo** will be used to report this information.

openBISpduRDCerror: generated when an OPEN BISPDU is received from another BIS in the same routeing domain, and the remote BIS is not a member of identically the same confederations as the local BIS. In addition to the parameters specified by ISO/IEC 10733, the following information will be reported by the AdditionalInformation field for the BIS Connection on which this OPEN PDU was received:

1. Remote BIS NET for this BIS-BIS connection—using the **notificationRemoteBIS-NET** parameter.
2. Remote BIS Routeing Domain Confederation (RDC) information using the **notificationRemoteRDCconfig** parameter.
3. Local BIS Routeing Domain Confederation (RDC) information using the **notificationLocalRDCconfig** parameter.

errorBISPDUconnectionclose: generated when an ERROR PDU has been received from a remote BIS. In addition to the parameters specified by ISO/IEC 10733, the following information will be reported by the AdditionalInformation field for the BIS Connection on which this OPEN PDU was received:

1. RemoteBIS-NET for BIS-BIS connection—using the **notificationRemoteBIS-NET** parameter
2. BISPDU error code (see 7.4 and 8.19)—this reports the error code that will be sent in the ERROR PDU using the parameter **notificationBISpduerrorcode**.
3. BIS error subcode (see 7.4 and 8.19)—this reports the subcode that will be sent using the parameter **notificationBISerrorsubcode**.
4. BISPDU error information (see 7.4 and 8.19)—this reports the data from the received BISPDU that will be used to diagnose the problem for the Notification. The parameter **notificationBISpuderrorinfo** will be used to report this information.

CorruptAdjRIBIn: generated when the local method of checking the Adj-RIB-In has found an error. All Adj-RIBs-In are being purged. In additon to the parameters specified by ISO/IEC 10733, the following information will be reported by the AdditionalInformation field:

1. **MaxRIBIntegrityCheck** attribute for this BIS
2. The remote BIS associated with this Adjacent RIB in the parameter **notificationRemoteBIS-NET**

packetBomb: generated when the local BIS has been presented with a BISPDU whose source is not one of the BISs adjacent to the local BIS. Such BISPDUs are rejected by the local BIS. In additon to the parameters specified by ISO/IEC 10733, the following information will be reported by the AdditionalInformation field with the parameters:

1. notificationSourceBIS
2. notificationSourceBISrdi
3. notificationSourceBISrdc

These parameters are created from the OPEN PDU values:

1. notificationSourceBIS—NET of remote BIS sending packet bomb
2. notificationSourceBISrdi—RDI of remote BIS sending packet bomb
3. notificationSourceBISrdc—RDC information for remote BIS sending packet bomb

enterFSMstateMachine: generated when the IDRP FSM state machine used to communicate with another BIS is started. The RemoteBis-N is reported in the additionalInformation field using the

notificationRemoteBis-NET parameter. The significance of the subparameter of each item of AdditionalInformation shall be set to the value FALSE (that is, not significant) so that a managing system that receives the event report will be less likely to reject it.

BEHAVIOUR
iDRPBasicImportedInfoNotifications-B
BEHAVIOUR DEFINED AS Imports the communicationsInformation notification from ISO/IES 10165-2. It is used to report the following protocol events:

enterFSMState: generated when a BIS starts the IDRP FSM state machine to establish a connection with a remote BIS. The RemoteBis-NET is reported in the AdditionalInformation field using the **notificationRemoteBis-NET** parameter. The significant subparameter of each item of AdditionalInformation shall be set to ″false″ (that is, not significant) so that a managing system receiving the event report will be less likely to reject it.

FSMStateChange: generated when the IDRP FSM used to communicate with another BIS transitions from one state to another. The RemoteBis-NET is reported in the AdditionalInformation field using the **notificationRemoteBis-NET** parameter. The significant subparameter of each item of AdditionalInformation shall be set to ″false″ (that is, not significant) so that a managing system receiving the event report will be less likely to reject it.

ATTRIBUTES
InternalBIS **GET**,
IntraIS **GET**,
ExternalBISNeighbor **GET**,
InternalSystems **GET**,
LocalRDI **GET**,
RDC-Config **GET**,
LocalSNPA **GET**,
MultiExit **GET**,
routeserver **GET**,
maximumPDUsize **GET**,
holdTime **GET**,
oustandingPDUs **GET**,
authenticationCode **GET**,
RetransmissionTimer **GET**,
CloseWaitDelayPeriod **GET**,
RDTransitDelay **GET**,
RDLRE **GET**,
LocExpense **GET**,
RIBAttsSet **GET**,
Capacity **GET**,
Priority **GET**;
version **GET**
maxRIBIntegrityCheck **GET**
maxIntegrityTimer **GET**
routeAdvertisementInterval

ACTIONS
startevent,
stopevent;

NOTIFICATIONS
~~enterFSMState,~~
~~FSMStateChange,~~
~~errorBISPDUsent,~~
~~openBISpduRDCerror,~~
~~errorBISPDUconnectionclose;~~
~~CorruptAdjRIBIn~~
~~packetbomb~~
″REC X.721 | ISO/IEC 10165-2:1992″:
communicationsAlarm
notificationRemotebis-NET
notificationBISpduerrorcode
notificationBISerrorssubcode
notificationBISpduerrorinfo
notificationRemoteRDCconfig
notificationLocalRDCconfig
maxAdjRibIntegritycheck
notificationSourceBis
″REC X.723 | ISO/IEC 10165-5: 1992″:
communicationsInformation

notificationRemotebis-NET

REGISTERED AS {ISOxxxx-IDRP.moi idrp_config (1) ;;;

12.3 Adjacent BIS Peer Managed Objects

adjacentBIS **MANAGED OBJECT CLASS**

DERIVED FROM ″ISO/IEC xxxxx″: top
CHARACTERIZED BY adjacentBIS **PACKAGE**
ATTRIBUTES
BIS_NET **GET**,
BIS_RDI **GET**,
BIS RDC **GET**,
BISnegotiatedversion **GET**,
BISpeerSNPAs **GET**,
Authentication_type **GET**,
State **GET**,
Lastseqnosent **GET**,
Lastseqnorecv **GET**,
Lastacksent **GET**,
Lastackrecv **GET**,
updatesIn **GET**,
updatesOut **GET**,
totalBISPDUsIn **GET**,
totalBISPDUsOut **GET**,
KeepalivesSinceLastUpdate **GET**,
closeWaitDelayTimer **GET**,
keepAliveTimer **GET**,
minRouteAdvertisementTimer **GET**,
maxCPUOverloadTimer **GET**,
minRDOriginationTimer **GET**,
ATTRIBUTE GROUPS
″REC X.723 | ISO/IEC 10165-5″: counters
updateIN
updateOUT
totalBISPDUsIN
totalBISPDUsOUT
KeepalivesSinceLastUpdate;
″REC X.723 | ISO/IEC 10165-5″: state
state
lastseqnosent
lastseqnorecv
lastacksent
lastackrecv;
″REC X.723 | ISO/IEC 10165-5″: timer
closeWaitDelayTimer **GET**;
keepALivetIMER **get**;
MinRouteAdvertisementTimer **GET**;
maxCPUOverloadTimer **GET**;
minRDUOriginationTimer **GET**;
REGISTERED AS [ISO xxxxx-IDRP.moi
adjacentBIS(2);

12.4 Attribute Definitions

InternalBIS **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOXXXX-IDRP.BIS_group;
MATCHES FOR Equality;
BEHAVIOUR InternalBIS-B
BEHAVIOUR DEFINED AS The set of NETs which identify the BISs in this routeing domain;
REGISTERED AS {ISOXXXX-IDRP.aoi
InternalBIS(1);

IntraIS **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOXXXX-IDRP.BIS_group;
MATCHES FOR Equality;
BEHAVIOUR IntraIS-B
BEHAVIOUR DEFINED AS The set of NETs of the ISs to which the local BIS may deliver an inbound NPDU whose destination lies within the BIS′s routeing domain. These ISs must be located on the same common subnetwork as this local BIS, and must be capable of delivering NPDUs to destinations that are located within the local BIS′s routeing domain.
REGISTERED AS {ISOXXXX-IDRP.aoi
IntraBIS(2);

ExternalBISNeighbor **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOXXXX-IDRP.BIS_group;
MATCHES FOR Equality;
BEHAVIOUR ExternalBISNeighborB
BEHAVIOUR DEFINED AS The set of NETs which identify the BISs in adjacent routeing domain that are reachable via a single subnetwork hop.
REGISTERED AS {ISOXXXX-IDRP.aoi
ExternalBISNeighbor (3);

InternalSystems **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOXXXX-IDRP.system_id_group
MATCHES FOR Equality;
BEHAVIOUR InternalSystems-B

BEHAVIOUR DEFINED AS The set of NETs and NSAPS which identify the systems in this routeing domain which the BIS uses to construct network layer reachability information;
REGISTERED AS ISOXXXX-IDRP.aoi InternalSystems (4);

LocalRDI **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOXXXX-IDRP.rdi
MATCHES FOR Equality;
BEHAVIOUR LocalRDI-B
BEHAVIOUR DEFINED AS The Routing Domain Identifier for the routeing domain where this BIS is located;
REGISTERED AS ISOXXXX-IDRP.aoi LocalRDI (5);

RDC-Config **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOXXXX-IDRP.rdc_group
MATCHES FOR Equality;
BEHAVIOUR RDC-Config-B
BEHAVIOUR DEFINED AS All of the Routing Confederations to which the RD of this BIS belongs and the nesting relationships that are in force between them. The nesting relationships are described as a sequence of sets of RDC Identifiers;
REGISTERED AS ISOXXXX-IDRP.aoi RDC-Config (6);

LocalSNPA **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOXXXX-IDRP.localSNPA
MATCHES FOR Equality;
BEHAVIOUR localSNPA-B
BEHAVIOUR DEFINED AS The list of SNPAs of this BIS;
REGISTERED AS ISOXXXX-IDRP.aoi LocalSNPA(7);

Multiexit **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX Boolean
MATCHES FOR Equality
BEHAVIOUR Multiexit-B
BEHAVIOUR DEFINED AS The indication whether this BIS will use the MULTI_EXIT_DISC attribute to decide between otherwise identical routes. The Multiexit parameter is used as the default value for the "multi_exit_disc" function in policy decisions;;
REGISTERED AS ISOXXXX-IDRP.aoi MultiExit(8);

maximumPDUsize **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.MaximumPDUSize;
MATCHES FOR Equality, Ordering;
BEHAVIOUR maximumPDUsize-B
BEHAVIOUR DEFINED AS The maximum number of octets that this BIS is able to handle in an incoming BISPDU;
REGISTERED AS ISOXXXXX-IDRP.aoi maximumPDUsize(9);

holdtime **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.Holdtime;
MATCHES FOR Equality, Ordering;
BEHAVIOUR holdtime-B
BEHAVIOUR DEFINED AS The maximum number of seconds that may elapse between the receipt of two successive BISPDUs of any of the following types: KEEPALIVE, UPDATE, RIB CHECKSUM PDUs or RIB REFRESH PDUs;
REGISTERED AS ISOXXXX-IDRP.aoi holdtime(10);

outstandingPdus **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.OutstandingPdus;
MATCHES FOR Equality, Ordering;
BEHAVIOUR outstandingPdus-B
BEHAVIOUR DEFINED AS The maximum number of BISPDUs that may be sent to this BIS without receiving an acknowl-edgement;
REGISTERED AS ISOXXXX-IDRP.aoi outstandingPdus(11);

authenticationCode **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.AuthenicationCode;
MATCHES FOR Equality, Ordering;
BEHAVIOUR authenticationCode-B
BEHAVIOUR DEFINED AS Indication of which authentication mechanism will be used;
REGISTERED AS ISOXXXX-IDRP.aoi authenticationCode (12);

RetransmissionTimer **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.retransmissiontimer
MATCHES FOR Equality, Ordering;
BEHAVIOUR RetransmissionTimer-B
BEHAVIOUR DEFINED AS The Number of seconds of between KEEPALIVE messages if no other traffic is sent;
REGISTERED AS ISOXXXX-IDRP.aoi RetransmisionTimer (13);

CloseWaitDelayPeriod **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.closewaitdelayperiod
MATCHES FOR Equality, Ordering;
BEHAVIOUR CloseWaitDelayPeriod-B
BEHAVIOUR DEFINED AS The number of seconds the local system shall stay in the CLOSE-WAIT state prior to changing to the CLOSED stated.;
REGISTERED AS ISOXXXX-IDRP.aoi CloseWaitDelayPeriod (14);

RDTransitDelay **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.RDtransitdelay
MATCHES FOR Equality, Ordering;
BEHAVIOUR RDTRansitDelay-B
BEHAVIOUR DEFINED AS The estimated average delay across a Routeing Domain in units of 500ms.
REGISTERED AS ISOXXXX-IDRP.aoi RDTransitDelay (15);

RDLRE **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.rdlre
MATCHES FOR Equality, Ordering;
BEHAVIOUR RDLRE-B
BEHAVIOUR DEFINED AS The average error rate of a Routeing Domain in units of an integer which if divided by 2**32-1 will provided the actual probability of the error.
REGISTERED AS ISOXXXX-IDRP.aoi RDLRE(16);

LocExpense **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.locexpense
MATCHES FOR Equality, Ordering;
BEHAVIOUR LocExpense-B
BEHAVIOUR DEFINED AS The monetary expense of transiting this Routeing Domain. The attribute contains an indication of cost and the units in which it is calculated;
REGISTERED AS ISOXXXX-IDRP.aoi LocExpense(17);

RIBAttsSet **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.ribattsSet
MATCHES FOR Equality;
BEHAVIOUR RIBAttsSet-B
BEHAVIOUR DEFINED AS The set of Rib Attributes supported by this BIS.;
REGISTERED AS ISOXXXX-IDRP.aoi RIBAttsSet(18);

Capacity **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.capacity
MATCHES FOR Equality, Ordering;
BEHAVIOUR Capacity-B
BEHAVIOUR DEFINED AS The traffic carrying capacity of this Routeing Domain.
REGISTERED AS ISOXXXX-IDRP.aoi Capacity(19);

Priority **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX
ISOxxxx-IDRP.priority
MATCHES FOR Equality, Ordering;
BEHAVIOUR Priority-B

BEHAVIOUR DEFINED AS The lowest value of ISO 8473 priority parameter that this RD will provide forwarding services for;
REGISTERED AS ISOXXXX-IDRP.aoi Priority(20);

BIS_NET **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.bis_net;
MATCHES FOR Equality;
BEHAVIOUR BIS_NET-B
BEHAVIOUR DEFINED AS The NET of the remote BIS of this BIS to BIS connection.;
REGISTERED AS {ISO-IDRP.aoi BIS_NET (21)};

BIS_RDI **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.rdi;
MATCHES FOR Equality;
BEHAVIOUR BIS_RDI-B
BEHAVIOUR DEFINED AS The RDI of the remote BIS of this BIS to BIS connection.;
REGISTERED AS {ISO-IDRP.aoi BIS_RDI (22)};

BIS_RDC **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.rdc_group
MATCHES FOR Equality;
BEHAVIOUR BIS_RDC-B

BEHAVIOUR DEFINED AS The RDC the remote BIS belongs to in this BIS to BIS connection.;

REGISTERED AS {ISO-IDRP.aoi BIS_RDC (23)};

BISnegotiatedversion **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.bisnegotiatedvesion;
MATCHES FOR Equality, Ordering;
BEHAVIOUR BISnegotiatedversion-B
.
BEHAVIOUR DEFINED AS The negotiated version of IDRP protocol this BIS to BIS connection is using.;
REGISTERED AS {ISOxxxx-IDRP.aoi BISnegotiatedversion (24)};

BISpeerSNPAs **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.bispeersSNPAs
MATCHES FOR Equality;
BEHAVIOUR BISpeerSNPAs-B
BEHAVIOUR DEFINED AS The SNPAs announced by the remote BIS of this BIS to BIS connection.
REGISTERED AS {ISOxxxx-IDRP.aoi BISpeerSNPAs (25)};

Authentication_type **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.auth_type
MATCHES FOR Equality, Ordering;
BEHAVIOUR authentication_type-B
BEHAVIOUR DEFINED AS The authentication type the remote BIS sent in the OPEN BISPDU in this BIS to BIS connection.
REGISTERED AS {ISOxxxx-IDRP.aoi Authentication_type (26)};

State **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.state
MATCHES FOR Equality, Ordering;
BEHAVIOUR state-B
BEHAVIOUR DEFINED AS The current state of BIS to BIS communication in the local BIS.
REGISTERED AS {ISOxxxx-IDRP.aoi state (27)};

Lastseqnosent **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.lastseqnosent
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR Lastseqnosent-B
BEHAVIOUR DEFINED AS The last sequence number sent to the remote BIS

from this local BIS on this BIS to BIS connection.
REGISTERED AS {ISOxxxx-IDRP.aoi Lastseqnosent (28)};

Lastseqnorecv **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.lastseqnorecv
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR Lastseqnorecv-B
BEHAVIOUR DEFINED AS The last sequence number received from the remote BIS by this local BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxx-IDRP.aoi Lastseqnorecv (29)};

Lastacksent **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.lastacksent
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR Lastacksent-B
BEHAVIOUR DEFINED AS The number of the last ack sent to the remote BIS from this local BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxxx-IDRP.aoi Lastacksent (30};

Lastackrecv **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.lastackrecv
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR Lastacksent-B
BEHAVIOUR DEFINED AS The number of the last ack received from the remote BIS by this local BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxxx-IDRP.aoi Lastackrecv (31)};

updatesIn **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.updatesin
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR updatesIn-B
BEHAVIOUR DEFINED AS The number of UPDATE BISPDUs received by this BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxx-IDRP.aoi updatesIn (32)};

updatesOut **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.updatesout
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR updatesOut-B
BEHAVIOUR DEFINED AS The number of UPDATE BISPDUs sent by this BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxx-IDRP.aoi updatesOut (33)};

totalBISPDUsIn **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.totalbispdusin
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR totalBISPDUsIn-B
BEHAVIOUR DEFINED AS The number of BISPDUS received by this BIS from the remote BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxx-IDRP.aoi totalBISPDUsIn (34)};

totalBISPDUsOut **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.totalbispdusout
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR totalBISPDUsOut-B
BEHAVIOUR DEFINED AS The number of BISPDUS received by this BIS from the remote BIS on this BIS to BIS connection.
REGISTERED AS {ISO xxxx-IDRP.aoi totalBISPDUsOut (35)};

KeepalivesSinceLastUpdate **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxxx-IDRP.keepaliveSincelastupdate
DERIVED FROM nonWrappingCounter;
MATCHES FOR Equality, Ordering;
BEHAVIOUR KeepalivesSinceLastUpdate-B
BEHAVIOUR DEFINED AS The number of KEEPALIVE BISPDUS received by this BIS from the remote BIS since this last UPDATE BISPDU.
REGISTERED AS {ISO xxxx-IDRP.aoi KeepAlivesSinceLastUpdate (36)};

version **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.version
MATCHES FOR Equality, Ordering;
BEHAVIOUR version-B
BEHAVIOUR DEFINED AS The version of IDRP protocol this machine defaults to using.;
REGISTERED AS {ISO xxxx-IDRP.aoi version (37)};

maxRIBIntegrityCheck**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.maxribintegritycheck
MATCHES FOR Equality, Ordering;
BEHAVIOUR maxRIBIntegrityCheck-B
BEHAVIOUR DEFINED AS The maximum time in seconds between checking of the Adj-RIBs-In by a local mechanism. If corrupt Adj-RIB-In is detected, the BIS shall purge the offending Adj-RIB-In;
REGISTERED AS {ISO xxxx-IDRP.aoi MaxRIBIntegrityCheck(38)};

maxRIBIntegrityTimer**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.ribintegritytimer
DERIVED FROM timer
MATCHES FOR Equality, Ordering;
BEHAVIOUR RIBIntegritytimer-B
BEHAVIOUR DEFINED AS The timer that measures in seconds the time remaining until the Adj-RIBs-In must be checked by a local mechanism. If a corrupt Adj-RIB-In is detected, the BIS shall purge the offending Adj-RIB-In;
REGISTERED AS {ISO xxxx-IDRP.aoi MaxRIBIntegrityTimer(39)};

closeWaitDelayTimer**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.waitdelaytimer
DERIVED FROM timer
MATCHES FOR Equality, Ordering;
BEHAVIOUR CloseWaitDelaytimer-B
BEHAVIOUR DEFINED AS The timer that measures in seconds the time that has elapsed since the BIS FSM entered the CLOSE-WAIT state. Upon timer expiration, the BIS FSM will enter the CLOSED state;
REGISTERED AS {ISO xxxx-IDRP.aoi CloseWaitDelayTimer(40)};

keepAliveTimer**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.keepalivetimer
DERIVED FROM timer
MATCHES FOR Equality, Ordering;
BEHAVIOUR Keepalivetimer-B
BEHAVIOUR DEFINED AS The timer that measures in seconds the time that has elapsed since the previous KEEPALIVE PDU was received by the local BIS. Upon its expiration, the BIS will send a BISPDU to its peer BIS;
REGISTERED AS {ISO xxxx-IDRP.aoi KeepAliveTimer(41)};

minRouteAdvertisementTimer**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.routeadvertisementtimer
DERIVED FROM timer
MATCHES FOR Equality, Ordering;
BEHAVIOUR Routeadvertisementtimer-B
BEHAVIOUR DEFINED AS The timer that measures in seconds the time that has elapsed since the advertisement by the local BIS of a better route that was received from a BIS located in another routeing domain. See clause8.16.3.1;
REGISTERED AS {ISO xxxx-IDRP.aoi MinRouteAdvertisementtimer(42)};

minRDOriginationTimer**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.rdoriginationtimer
DERIVED FROM timer
MATCHES FOR Equality, Ordering;
BEHAVIOUR RDOriginationtimer-B
BEHAVIOUR DEFINED AS The timer that measures in seconds the time that has elapsed since the advertisement by the local BIS of an UPDATE PDU that reported changes within the local BIS's routeing domain. See clause 8.16.3.2;
REGISTERED AS {ISO xxxx-IDRP.aoi MinRDOriginationtimer(43)};

maxCPUOverloadTimer**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISO xxxx-IDRP.maxcpuoverloadtimer
DERIVED FROM timer
MATCHES FOR Equality, Ordering;
BEHAVIOUR MaxCPUOverloadTimer-B
BEHAVIOUR DEFINED AS The timer that measures in seconds the time that has elapsed since the local BIS has detected that its CPU has become overloaded. See Annex F;
REGISTERED AS {ISO xxxx-IDRP.aoi MaxCPUOverloadtimer(44)};

routeserver **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX Boolean;
MATCHES FOR Equality
BEHAVIOUR routeserver-B
BEHAVIOUR DEFINED AS The indication whether this BIS may set the ″IDRP_Server_Allowed″ field in the NEXT_HOP attribute to X″FF″ for BIS to BIS UPDATE BISPDUs. If this variable is true then in accordance with local policy, the IDRP_Server_Allowed field may be set on some UPDATE BISPDUs that this BIS sends. If this attribute is set to false, then no UPDATE BISPDUs will be sent by this BIS with NEXT_HOP attributes containing an ″IDRP_Server flag″ equal to X″FF″.;
REGISTERED AS ISOXXXX-IDRP.aoi routeserver(45);

routeAdvertisementInterval**ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ISOxxxx-IDRP.RouteAdvertisementInterval;
MATCHES FOR Equality,Ordering;
BEHAVIOUR routeAdvertisementInterval-B
BEHAVIOUR DEFINED AS The minimum time in seconds between successive UPDATE PDUs advertising feasible routes learned from other RDs.
REGISTERED AS ISOXXXX-IDRP.aoi routeAdvertisementInterval(46);

12.5 Action Definitions

~~minRDOriginationTimer**ATTRIBUTE**~~

~~**WITH ATTRIBUTE SYNTAX** ISO xxxx-IDRP.rdoriginationtimer~~
~~**MATCHES FOR** Equality, Ordering;~~
~~**BEHAVIOUR** RDOriginationtimer-B~~
~~**BEHAVIOUR DEFINED AS** The timer that measures in seconds the time that has elapsed since the advertisement by the local BIS of an UPDATE PDU that reported changes within the local BIS's routeing domain. See clause – Heading '8' unknown –16.3.2;~~
~~**REGISTERED AS** {ISO xxxx-IDRP.aoi MinRDOriginationtimer(40)};~~

startevent **Action**

BEHAVIOUR
startevent **BEHAVIOUR**
MODE CONFIRMED;
CONTEXT ACTION-INFO;
WITH INFORMATION SYNTAX ISO xxxx-idrp.Actioninfo;
WITH REPLY SYNTAX ISO xxxx-idrp.Startevenreply;
DEFINED AS The request to start communication with a remote BIS peer;
PARAMETERS Remotebis-NET;
MODE CONFIRMED;
REGISTERED AS ISO xxxxx-IDRP.aci startevent (1);

Stopevent **Action**

BEHAVIOUR

stopevent **BEHAVIOUR**
MODE CONFIRMED;
CONTEXT ACTION-INFO;
WITH INFORMATION SYNTAX ISO xxxx-idrp.Actioninfo;
WITH REPLY SYNTAX ISO xxxx-idrp.Stopevenreply;
PARAMETERS Remotebis-NET;
MODE CONFIRMED;
DEFINED AS The request to stop communication with a remote BIS peer;
REGISTERED AS ISO xxxxx-IDRP.aci stopevent (2);

~~**12.6 Notification Definitions**~~

~~enterFSMstatemachine **NOTIFICATON**~~

~~**BEHAVIOUR** enterFSMstatemachine-B~~
~~**BEHAVIOUR DEFINED AS** The indication of starting the FSM state machine to establish a connection with a remote BIS.;~~
~~**MODE** NON-CONFIRMED~~
~~**PARAMETERS** Remotebis-NET;~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo~~
~~**REGISTERED AS** ISOxxxx-IDRP.noi enterFSMstatemachine (1);~~

~~FSMstatechange **NOTIFICATION**~~

~~**BEHAVIOUR** FSMstatechange-B~~
~~**BEHAVIOUR DEFINED AS** The indication of transiting from one state to another in the IDRP connection state machine in communication with another BIS.;~~
~~**MODE** NON-CONFIRMED~~
~~**PARAMETERS** remoteBIS-NET, state;~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo~~
~~**REGISTERED AS** ISOxxxx-IDRP.noi FSMstatechange(2);~~

~~errorBISPDUsent **NOTFICATION**~~

~~**BEHAVIOUR** errorBISPDUsent-B~~
~~**BEHAVIOUR DEFINED AS** The indication of an error in the format of BISPDU.;~~
~~**MODE** NON-CONFIRMED~~
~~**PARAMETERS** Remotebis-NET, BISpduerrorcode, BISerrorsubcode, BISpduerrorinfo;~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo **REGISTERED AS** ISOxxxx-IDRP.noi errorBISPDUsent (3);;~~

~~openBISpduRDCerror **NOTIFICATION**~~

~~**BEHAVIOUR** openBISpduRDCerror-B~~
~~**BEHAVIOUR DEFINED AS** The indication that an OPEN PDU has been received with the RDC Config for remote BIS and this BIS do not indicate that the two BIS trying to establish a connection are a part of the same confederations;~~
~~**MODE** NON-CONFIRMED~~
~~**PARAMETERS** Remotebis-NET, RemoteRDCconfig, LocalRDCConfig;~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo~~
~~**REGISTERED AS** ISOxxxx-IDRP.noi errorpduRDCerror(4);~~

~~errorBISPDUconnectionclose **NOTIFICATION**~~

~~**BEHAVIOUR** errorBISPDUconnectionclose-B~~
~~**BEHAVIOUR DEFINED AS** The indication that an ERROR BISPDU has been received from a remote BIS;~~
~~**MODE** NON-CONFIRMED~~
~~**PARAMETERS** Remotebis-NET, bispduerrorcode, bispduerrorsubcode,bispduinfo;~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo~~
~~**REGISTERED AS** ISOxxxx-IDRP.noi errorBISPDUconnectionclose(5);;~~

~~CorruptAdjRIBIn **NOTIFICATION**~~

~~**BEHAVIOUR** corruptAdjRIBIn-B~~
~~**BEHAVIOUR DEFINED AS** The indication that the local method of checking the Adj-RIB-In has found an error. All Adj-RIBs-In are being purged.~~
~~**MODE** NON-CONFIRMED~~
~~**PARAMETERS** maxAdjRibIntegritycheck;~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo~~

~~**REGISTERED AS** ISOxxxx-IDRP.noi corruptAdjRIBIn(6);;~~

~~packetBomb **NOTIFICATION**~~

~~**BEHAVIOUR** packetBomb-B~~
~~**BEHAVIOUR DEFINED AS** The indication that the local BIS has been presented with a BISPDU whose source is not one of the BISs adjacent to the local BIS. Such BISPDUs are rejected by the local BIS.~~
~~**MODE** NON-CONFIRMED~~
~~**WITH INFORMATION SYNTAX** ISOxxxx-IDRP.NotificationInfo~~
~~**REGISTERED AS** ISOxxxx-IDRP.noi packetBomb(7);;~~

12.7 Parameter Definitions

notificationRemoteBIS-NET **PARAMETER**

CONTEXT ACTION-REPLY;
WITH SYNTAX ISOxxxx-IDRP.remoteBIS-NET;
BEHAVIOUR RemoteBIS-NET-B
PARAMETER DEFINED AS The NET of the Remote BIS that this local BIS is starting IDRP protocol communication with.;
REGISTERED AS ISOxxx-IDRP.proi RemoteBIS-NET(1);

~~Remotebis-NET **PARAMETER**~~

~~**CONTEXT** EVENT-INFO;~~
~~**WITH SYNTAX** ISOxxxx-IDRP.remoteBIS-NET;~~
~~**BEHAVIOUR** Remotebis-NET-B~~
~~**PARAMETER DEFINED AS** The NET of the Remote BIS that this local BIS is starting IDRP protocol communication with.;~~
~~**REGISTERED AS** ISOxxxx-IDRP.proi Remotebis-NET(1);~~

notificationSTATE **PARAMETER**

CONTEXT EVENT-INFO;
WITH SYNTAX ISOxxxx-IDRP.state
BEHAVIOUR ISOxxx-IDRP.STATE-B
PARAMETER DEFINED AS The state of the local BIS Finite State machine.;
REGISTERED AS ISOxxxx-IDRP.prio STATE(1);

notificationBISpduerrorcode **PARAMETER**

CONTEXT EVENT-INFO;
WITH SYNTAX ISOxxxx-IDRP.bispduerrorcode
BEHAVIOUR ISOxxxx-IDRP.BISpduerrorcode-B
BEHAVIOUR DEFINED AS The error code indicating what type of error occurred in the BIS PDU.;
REGISTERED AS ISOxxxx-IDRP.prio BISpduerrorcode(2)

notificationBISpduerrorsubcode **PARAMETER**

CONTEXT EVENT-INFO;
WITH SYNTAX ISOxxxx-IDRP.bispduerrorsubcode
BEHAVIOUR ISOxxxx-IDRP.BISpduerrorcode-B
BEHAVIOUR DEFINED AS The error code indicating what type of error within the major error type occurred in the BIS PDU.;
REGISTERED AS ISOxxxx-IDRP.prio BISpduerrorsubcode(3)

notificationBISpduerrorinfo **PARAMETER**

CONTEXT EVENT-INFO;
WITH SYNTAX ISOxxxx-IDRP.bispduerrorinfo
BEHAVIOUR ISOxxxx-IDRP.BISpduerrorinfo-B
BEHAVIOUR DEFINED AS The additional information from original pdu that indicated an error in the BIS PDU.;
REGISTERED AS ISOxxxx-IDRP.prio BISpduerrorinfo(4);

notificationRemoteRDCconfig **PARAMETER**

CONTEXT EVENT-INFO;
WITH SYNTAX ISOxxxx-IDRP.remoteRDCconfig;
BEHAVIOUR ISOxxxx-IDRP.RemoteRDCconfig-B
BEHAVIOUR DEFINED AS The Routing Domain Confederation (RDC) information from the remote BIS on this BIS to BIS communication.;
REGISTERED AS ISOxxxx-IDRP.prio RemoteRDCconfig(5);

notificationLocalRDCconfig **PARAMETER**

CONTEXT EVENT-INFO;
WITH SYNTAX ISOxxxx-IDRP.localRDCconfig;
BEHAVIOUR ISOxxx-IDRP.LocalRDCconfig-B
BEHAVIOUR DEFINED AS The Routing Domain Confederation (RDC) information from this local BIS on this BIS to BIS communcation.;
REGISTERED AS ISOxxxx-IDRP.prio LocalRDCconfig(6);

12.8 Attribute Groups

counters **ATTRIBUTE** group

DESCRIPTION The group of all counter per BIS connection
REGISTERED AS {ISO xxxxx-IDRP.agoi counters [1]};

stateinfo **ATTRIBUTE** group

DESCRIPTION The group of all state information per BIS connection
REGISTERED AS {ISO xxxx-IDRP.agoi stateinfo[2]};

bistimer **ATTRIBUTE** group

DESCRIPTION The group of all timers per BIS connection
REGISTERED AS {ISO xxxx-IDRP.agoi bistimer[2]};

12.9 ASN.1 MODULES

ISO 10747-IDRP(tbd1) **DEFINITIONS::=BEGIN**
-- object identifier definitions
sc6 **OBJECT IDENTIFIER** ::= {joint-iso-ccitt sc6(?)}
-- value to be assigned by SC21 secretariat
idrpoi **OBJECT IDENTIFIER** ::= {sc6 iso 10747(?)}
-- value to be assigned by SC6 secretariat
sseoi **OBJECT IDENTIFIER** ::{idrpoi standSpecificExtensions(0)}
moi **OBJECT IDENTIFIER** ::= {idrpoi objectClass (3)}
poi **OBJECT IDENTIFIER** ::= {idrpoi package (4)}
proi **OBJECT IDENTIFIER** ::={idrpoi parameter(5)}
nboi **OBJECT IDENTIFIER** ::={idrpoi nameBinding (6)}
aoi **OBJECT IDENTIFIER** ::={idrpoi attribute (7)}
agoi **OBJECT IDENTIFIER** ::={idrpoi attributeGroup (8)}
aoi **OBJECT IDENTIFIER** ::={idrpoi action (9)}
noi **OBJECT IDENTIFIER** ::={idrpoi action (10)}

--
--object identifiers for notification parameters
--

OBJECT IDENTIFIER ::={sseoi SpecificProblems(3)?}

errorBISPDUsent OBJECT IDENTIFIER ::= {se errorBISPDU(0)}
openBISpduRDCerror OBJECT IDENTIFIER ::= {se errorBISPDU(1)}
errorBISPDUconnectionclose OBJECT IDENTIFIER ::= {se errorBISPDU(2)}
CorruptAdjRIBIn OBJECT IDENTIFIER ::= {se errorBISPDU(3)}
packetBomb OBJECT IDENTIFIER ::= {se errorBISPDU(4)}
enterFSMstate OBJECT IDENTIFIER ::= {se errorBISPDU(5)}
FSMStateChange OBJECT IDENTIFIER ::= {se errorBISPDU(6)}

--
--ASN1 Types and Values
--

ActionInfo ::=**SET OF** Parameter
ActionReply ::= **SEQUENCE** {
responseCode **OBJECT IDENTIFIER**
responseArgs **SET** of Parameter
OPTIONAL}
AuthenticationCode ::=**ENUMERATED**{
integrityOnly(0),
integrityPlusAuthentication(1)}
auth_type ::=AuthenticationCode
BIS_group ::= **SET OF** {NetworkEntityTitle}
bis_net::= NetworkEntityTitle

bisnegotiatedversion ::=version
bispduerrorcode::= **ENUMERATED** {
OPEN_PDU_Error (1),
UPDATE_PDU_Error (2),
Hold_timer_Expired (3),
bispduerrorsubcode ::= **SET OF** {
openerrorsubcode,
updateerrorsubcode}
bispduerrorinfo ::=**OCTETSTRING**(1...50)
--50 bytes of original message are saved

Editor's Note: Comment is requested on the amount of data that should be saved.

bispeersSNPAs ::= SNPAAddresses
Boolean ::= **BOOLEAN**
capacity ::==**INTEGER**(1...255)
closewaitdelayperiod ::=**INTEGER**(150)
destinationspecificqos ::=ribattsec
destinationspecificsecurity ::=ribattsec
expensevalue ::=localexpense
Holdtime ::=**INTEGER**(1...65 535)
keepaliveSincelastupdupdate
::=**INTEGER**(1...4 294 967 295)
keepalivetimer ::= timer
lastseqnosent ::=**INTEGER**(1...(4 294 967 295))
lastseqnorecv ::=**INTEGER**(1...(4 294 967 295))
lastacksent ::=**INTEGER**(1...(4 294 967 295))
lastackrecv ::=**INTEGER**(1...(4 294 967 295))
locexpense ::= **INTEGER**(1...65 535)
localRDCconfig ::=rdc_group
local_SNPAs ::= SNPAaddresses
MaximumPDUSize ::=**INTEGER**(1..65 535)
Metriclength ::=**INTEGER(1...255)**
Metricvalue ::=**OCTETSTRING(SIZE(1...255))**
NSAPprefixLength ::=**INTEGER(1...160)**
NSAPprefix ::= BITSTRING(SIZE(1...160)
NetworkEntityTitle
::=**OCTETSTRING**(SIZE(1...20))
NotificationInfo ::=**SET OF** Parameter
openerrorsubcode ::=**ENUMERATED** {
UnsupportedVersion_number (1),
Bad_Max_PDU_size (2),
Bad_Outstanding_PDUs (3),
Bad_Peer_RD (4),
Unsupported_Authentication_code (5),
Authentication_Failure (6),
Bad_RIB-AttrsSet (7),
RDC_mismatch (8)}
OutstandingPdus ::=**INTEGER**(0...255)
Parameter ::= **SEQUENCE** {
paramID **OBJECTIDENTIFIER**
paramInfo **ANY DEFINED BY** ParmID}
priority ::=**INTEGER**(0...14)
priorityvalue ::=priority
QOSlength ::= **INTEGER**(1...255)
QOSvalue ::= **OCTETSTRING**(SIZE(1...255))
rdi ::=**OCTETSTRING**(SIZE(1...20));
--assigned from the NSAP address space
rdc_group::=**SEQUENCE**{**SEQUENCE**
rdc_set_id,**SET OF** {rdi}}}
rdc_set_id ::=**INTEGER**(1..255)
RDtransitDelay ::=**INTEGER**(0...65 535)
rdlre ::=**INTEGER**(0...(4 294 967 295))
retransmission_timer ::= **INTEGER**(0...65535)
remoteBIS-NET ::=NetworkEntityTitle
remoteRDCconfig ::=rdc_group
ribattsSet ::=SEQUENCE {
SEQUENCE {
ribsetid,
ribsetcount,
SET OF {rib_attributes}}
ribsetid ::=**INTEGER**(1..255)
ribsetcount ::=**INTEGER**(0..255)
rib_attributes ::= **SEQUENCE OF** {
rib_attribute,
rib_value}
rib_attribute ::= **ENUMERATED** {
TRANSIT_DELAY (9),
RESIDUAL_ERROR (10),
EXPENSE (11),
SourceSpecificQOS (12),
DestinationSpecificQOS (13),
SourceSpecificSecurity (17),
DestinationSpecificSecurity(18),
Capacity (19),
Priority (20)}
~~rib_value :: — **OCTETSTRING**~~
~~— This octetstring may vary according to the~~
~~— rib_attribute value. Source Specific QOS,~~
~~— Destination SPecific QOS, SOurce Specific~~
~~— Security, Destination Specific Security,~~
~~— may have varying lengths of rib attribute values.~~
~~— See the appropriate subclause of 8.12~~
~~— for more details~~
rib_value ::= **SEQUENCE OF**{ribattlength, ribattvalue}
ribattlength ::= **INTEGER**

```
ribattvalue ::= CHOICE OF{
    transitdelayvalue,
    residualerrorvalue,
    expensevalue,
    sourcespecificqos,
    destinationspecificqos,
    sourcespecificsecurity,
    destinationspecificsecurity,
    capacityvalue,
    priorityvalue}
ribattqos ::= SEQUENCE OF{
    NSAPprefixlength,
    NSAPprefix,
    QOSlength,
    QOSvalue,
    metriclength,
    metricvalue}
ribattsec ::= SEQUENCE OF{
    NSAPprefixlength,
    NSAPprefix,
    securitylength,
    securitylevel}
routeAdvertisementInterval
::=INTEGER(30...900)
 --IS 10589 imposes minimum value of 30
seconds
 --and maximum value of 900 seconds in
clause
 --12.2.3.4, part c)
securitylength ::= INTEGER(0...255)
securitylevel ::=
OCTETSTRING(SIZE(1...255))
~~routeadvertisementtimer ::= timer~~
~~rdoriginationtimer ::= timer~~
SNPAAddress ::=SET OF {
  SNPA_Type, SNPAaddress}
SNPAaddress ::= SEMIOCTET STRING
  (FROM
('1'H|'2'H|'3'H|'4'H|'5'H|'6'H|'7'H|'8'H|'9'H|
      'A'H|'B'H|'C'H|'D'H|'E'H|'F'H))
  --integral number of hexadecimal digits
SNPAaddresses ::= SET OF SNPAAddress
state ::= ENUMERATED {
  closed (0),
  open-recv(1),
  established(2),
  open-sent(3),
  close-wait(4)}
system_id_group ::= SEQUENCE OF {
  SET OF {NetworkEntityTitle},
  SET OF {EndSystemNSAPs}}
~~timer ::=SEQUENCE {~~
  ~~exponent [1] INTEGER (-62...63)~~
  ~~mantissa [2] INTEGER (0...65 535)~~
updateerrorsubcode ::=ENUMERATED {
  Malformed_Attribute_list (1),
  Unrecognized_Well-known_Attribute (2),
  Missing_Well-known_Attribute (3),
  Attribute_Flags_Error (4),
  Attribute_Length_Error (5),
  RD_Routeing_Loop (6),
  Invalid_NEXT_HOP_Attribute (7),
  Optional_Attribute_error (8),
  Invalid_Reachability_Information (9),
  Misconfigured_RDCs (10)}
updatesin ::=INTEGER(1...4 294 967 295)
updatesout ::=INTEGER(1...4 294 967 295)
totalbispdusin ::=INTEGER(1..4 294 967 295)
totalbispdusout ::=INTEGER(1..4 294 967 295)
version ::=INTEGER (1...255)
waitdelattimer ::= timer
```