Title: USA Comments on SC6 N6387 (Inter-domain Routeing Protocol)

To: X3S3

Source: X3S3.3

Reference: SC6 N6387 (WD on Inter-domain Routeing Protocol)

Task group X3S3.3 has reviewed the working draft text on inter-domain routeing (SC6 N6387) , and recommends that the USA support its progression to CD-ballot status out of the Berlin SC6/WG2 meeting. In accordance with SC6 Sydney Resolution 29 (2.4):1990, X3S3.3 also requests that the attached detailed comments be submitted to the SC6/WG2 secretariat for discussion at the July 1991 SC6/WG2 meeting in Berlin,

ISO/IEC JTC1/SC6 WG2_____

Project: 06.41.05
Date: 1991-04-25

ISO
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/IEC JTC1/SC6 WG2

TELECOMMUNICATIONS AND INFORMATION EXCHANGE BETWEEN SYSTEMS

SECRETARIAT: UK (BSI)

TITLE: Comments on SC6 N6387 (IS-IS Inter-domain Routeing Exchange Protocol)

SOURCE: USA

STATUS: The USA has reviewed the working draft text (SC6 N6387) on the inter-domain routeing information exchange protocol, and supports its progression to CD-ballot status out of the Berlin SC6/WG2 meeting. In accordance with SC6 Sydney Resolution 29 (2.4):1990, the USA submits the attached comments for discussion at this meeting.

These comments are organized into three parts:

1. Responses to questions posed in Editor's notes in SC6 N6387
2. Comments of a technical nature
3. Comments of an editorial nature

Part I: Responses to Calls for Comment

Document SC6 N6387 contains several ″Editor′s Notes″ to which member bodies were asked to respond. The USA responses are as follows:

1. **Information about Systems Inside a Routeing Domain (Clause 7.3, page 20):**

   The protocol needs this information to operate correctly, but correct operation does not depend critically on the manner in which it is obtained. This information is used simply to list all the systems located in a given routeing domain, but not whether each system is ″up″ or ″down″. Since it conveys a system′s identity, not its operational status, it is not expected to change at a very rapid rate. Therefore, the current text in SC6 N6387 is sufficient, and there is no need to expand upon it.

   However, the USA notes that further work on the viability of acquiring this information dynamically would be extremely useful, but does not properly fall within the scope of SC6 N6387.

2. **Authentication Methods (Clause 7.9, page 27, and Annex B):**

   Normative material should be included in SC6 N6387 to define an additional authentication mechanism. A crypto-based authentication technique based on the MD4 Message Digest Algorithm (as described in RFC 1186, which is attached as Appendix D, "Text of RFC 1186" on page 31) is proposed. A single field in the BISPDUs will be used to carry both checksum and authentication information, and the IDRP checksum will be based on the MD4 algorithm rather than the ISO 8473 (Fletcher) algorithm. Details are provided in Appendix A, "Authentication based on MD4 Message Digest Algorithm" on page 21.

3. **CO/CL Attribute (Clause 7.11.9, page 31):**

   Since CO/CL interworking function units are outside the scope of the OSI Reference Model and are not within the scope of any international standard, this attribute should be dropped from SC6 N6387.

4. **TRANSIT DELAY Attribute (Clause 7.11.10, page 31):**

   A single value of TRANSIT DELAY per routeing domain is sufficient. The benefits from introducing ″path specific delay″ for each possible path between each pair of BISs in a routeing domain would be minimal, and would be far outweighed by the complexity that would be added.

5. **RESIDUAL ERROR Attribute (Clause 7.11.11, page 32):**

   A single RDLRE value per routeing domain is sufficient. The benefits from introducing ″path specific error rate″ for every possible path across a routeing domain would be minimal, and would be far outweighed by the complexity that would be added.

6. **EXPENSE Attribute (Clause 7.11.12, page 32):**

   A single EXPENSE value per routeing domain is sufficient. The benefits from introducing ″path specific expense″ for every possible path across a routeing domain would be minimal, and would be far outweighed by the complexity that would be added.

7. **HIERARCHICAL RECORDING Attribute (Clause 7.11.15, page 33):**

   A new technical approach for syntax, semantics, and usage of the HIERARCHICAL RECORDING attribute is contained in Appendix B, "HIERARCHICAL RECORDING Attribute" on page 23. The newly defined attribute will be used in conjunction with routeing domain confederations to provide a method for controlling transitivity of external traffic through a routeing domain. It is recommended that this approach be incorporated into SC6 N6387.

8. **Support for Forwarding of 8208 CR Packets (Clause 7.16, page 42):**

   Although the relaying and routeing functions are tightly coupled to one another in connectionless services, they are largely decoupled from one another in connection-oriented services. A known problem in connection-oriented routeing is that the effects of transient looping during connection-setup time will persist for the duration of the connection, affecting all subsequent data PDUs.

   None of the existing OSI routeing protocols (whether connection-oriented or connectionless) provide mechanisms to prevent the occurrence of transient loops. In particular, although the mechanisms of either DIS 10589 or SC6 N6387 could be applied to the routeing of CR packets, neither protocol provides mechanisms to prevent transient looping.

   The USA believes that the connection-oriented protocols themselves are the proper place to develop solutions to the transient looping problem, and is aware that work of this nature is under way for the X.25 and X.75 protocols.

   The actual description of the forwarding functions needed in the CONS environment most likely belongs as part of the CONS relaying specification (DIS 10029). This will ensure that all aspects of IS relaying for CONS are covered in one place. The specification can make use of routes computed by any of the ISO routeing protocols, such as ISO 9542, 10030, 10589, and IDRP.

   Thus, the USA recommends that support for forwarding of 8208 CR packets be dropped from SC6 N6387, and a new ″Non-goal″ be added to clause 5.13.2 to state that the solution of the transient looping problem for CR packets is outside the scope of IDRP.

Part II: Technical Comments

9. **Source Routeing of 8473 NPDUs**:

   The text on the forwarding process should be expanded to address the source route parameters of 8473 NPDUs. For example, there should be normative text stating that a complete source route (if present) takes precedence over the next IS contained in IDRP′s FIB.

10. **NEXT_HOP attribute:**

    NEXT_HOP is listed as a well-known mandatory attribute whose value is the NET of the BIS that originates an UPDATE PDU (see clause 7.11.3). Since the NET is known at the time the BIS-BIS connection is established or may be ascertained from the NUNITDATA.INDICATION primitive, the presence of the NEXT_HOP attribute in the UPDATE PDU is redundant.

    However, rather than dropping this attribute, we recommend that its syntax and semantics be redefined, as explained in Appendix C, “The NEXT_HOP Attribute” on page 26. The redefined attribute will provide an optional ″route server″ capability in IDRP, and will also permit a BIS to inform neighbor BISs of its preferences about which SNPAs should be used for inbound traffic.

11. **Use of ranges:**

    The benefit from the support of ranges in SC6 N6387 does not justify the additional complexity or the incompatibility with DIS 10589, which supports only address prefixes. Therefore, support for ranges should be deleted from SC6 N6387.

    However, support for prefixes should be retained in order to provide consistency with the intra-domain routeing protocol (DIS 10589), and to allow for convenient exchange of information between these two protocols.

12. **Handling of the 8473 Security parameter in NPDUs:**

    SC6 N6387 should be expanded to address handling of the 8473 Security parameter by defining a new distinguishing attribute and its associated usage rules.

    An approach is outlined below:

    > The usage rules should be similar to those already defined for SS-QOS and DS-QOS routeing: that is, it can parallel the text of clause 7.11.13-14 of SC6 N6387. The new path attributes would be type-value specific, and process for matching the attribute to the parameters in the 8473 NPDU would be based on the methods of clause 7.15.3. It would be necessary to expand clause 7.15.2 so that NPDU-derived Distinguishing Attributes could be obtained from the 8473 security parameter as well as from its QOS-Maintenance parameter.

13. **Controlling Inter-Domain Routeing Traffic Overhead**

    The inter-domain routeing protocol should provide methods that limit the amount of routeing traffic (that is, BISPDUs). Such constraints will place limits on both the link bandwidth needed to advertise BISPDUs and the processing power needed by the Decision Process to digest the information contained in the BISPDUs are required. Several measures are suggested below:

    - Frequency of Route Selection

      To limit the amount of Inter-Domain routeing traffic generated by a given BIS in response to updates received from other routeing domains, a new architectural constant (**MinRouteSelectionInterval**) and an associated timer should be defined. The amount of elapsed time between successive advertisements of better routes (as determined by its local policies) must be equal to or greater than this constant.

Since fast convergence is needed within an RD, this constant does not apply to advertisement of better routes chosen as a result of updates from BISs located in the advertising BIS′s own RD. To avoid long-lived black holes, it does not apply to advertisement of previously selected routes which have become unreachable. In both of these situations, the local BIS must advertise such routes immediately.

If a BIS has selected new routes based on updates from BISs in adjacent RDs, but have not yet advertised them because the **MinRouteSelectionInterval** has not yet expired, the reception of any routes from other BISs in its own RD forces the **MinRouteSelectionInterval** 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.

- Frequency of Route Origination

  The amount of update traffic generated by a BIS as a result of changes within its own routeing domain can be limited by requiring that there should be a minimum amount of time that must elapse before such changes can be advertised. This can be accomplished by a new architectural constant **MinRDOriginationInterval** whose value specifies the minimum allowable time interval between such updates.

- Introduction of Jitter

  There is danger that the distribution of BISPDUs generated by a given BIS will contain peaks. If there are a large number of BISs, this can cause overloading of both the transmission medium and other BISs.

  To prevent this, ″jitter″ (as defined in DIS 10589) should be applied to the timers associated with **MinRouteSelectionInterval** and **MinRDOriginationInterval**. A given BIS will apply the same ″jitter″ to each of these quantities regardless of the destinations to which the updates are being sent: that is, jitter will not be applied on a ″per peer″ basis.

14. **Length Field in UPDATE PDUs:**

    The additional complexity and the level of indirection involved in the use of an ″Extended Length″ bit in the path attributes flags (see clause 6.3, page 13) seems unnecessary. The ″Extended Length″ bit should be deleted, and that the ″Length″ field itself should be fixed at a length of 2 octets.

15. **LOCAL_PREF Attribute:**

    The usage rules for this attribute (clauses 7.11.8 and 7.12.7) should be expanded to state that if this attribute is present in an UPDATE PDU received from a BIS located in another routeing domain, then the receiving BIS shall ignore this attribute.

16. **Sequence Number Rollover:**

    Clause 7.4.2 should be expanded to state that the sequence numbers are linear, do not wrap around, and have values in the range from 1 to $2^{32} - 1$. The number 0 is not a valid sequence number.

    Upon initializing the FSM for a given BIS-BIS connection, the first sequence number to be used will be ″1″. If a BIS-BIS connection is broken down and then re-established, the next sequential sequence number will be used. Finally, if the maximum permissible sequence number would be exceeded, then the connection will be torn down and a new connection will be established, using the number ″1″ as its initial sequence number.

17. **8473 ER PDUs**

Clause 7.5 needs to contain material that describes how a BIS will handle an incoming 8473 ER PDU.

18. **Finite State Machines**

Material should be added to clause 7.5.3.3 to clarify the actions to be taken when a BIS receives an OPEN PDU while it is in the ESTABLISHED state: if a valid OPEN_PDU is received, the BIS shall ignore it. This will prevent the occurrence of delayed duplicate OPEN PDUs.

19. **Internal Updates**:

Clause 7.12.10 describes how a BIS can propagate updates to other BISs located in its own routeing domain. Although it discusses the handling of updates received from BIS located in adjacent RDs, it failed to discuss how to handle updates received from BISs in its own RD. This can be corrected by adding the following normative text:

> If a given BIS receives an UPDATE PDU from another BIS located in its own routeing domain, the receiving BIS shall not re-distribute the routeing information contained in that UPDATE PDU to other BISs located in its own routeing domain.

Note that this does not apply to updates that are received from BISs located in other routeing domains.

20. **Partition Repair**

Since IDRP does not support repair of either partitioned routeing domain or partitioned confederations, this should be noted in clause 5.13.2 as being outside its scope, and not one of its design goals.

21. **Rejection of ″Packet Bombs″**:

SC6 N6387 should define mechanisms for rejecting ″packet bombs″, which are defined to be bogus BISPDUs delivered to a BIS from an improper source. IDRP can provide this function by requiring that the receive process in clause 7.14 be expanded. The required additional check can be accomplished by amending the second dashed item under ″c)″, as follows (new text is shown in italics):

> If the SPI identifies IDRP *and the source address of the outer 8473 NPDU identifies any of the systems listed in either managed object* **INTERNAL-BIS** *or managed object* **EXTERNAL-BIS-NEIGHBORS**, then the inner BISPDU shall be extracted....
>
> *However, if the source address of the outer 8473 NPDU does not identify a system listed in these managed objects, then the NPDU shall be rejected by the receiving BIS.*

22. **Breaking Ties for Routes with Equal Preference:**

It is possible for a local policy to assign the same degree of preference to several routes to a given destination; hence, several routes to the same destination could be advertised with the same value of the LOCAL_PREF attribute. To provide determinable performance, the tie can be broken by requiring a BIS to select the route that was advertised by the BIS with the numerically lowest NET. If the local BIS also has a route to the same destination and its own NET is the lowest, then it may will select its own route. Appropriate normative text should be added to clause 7.13.1.

23. **RDCs and Distribution Lists:**

    There are several clarifications that apply to the DIST_LIST attributes:

    - The DIST_LIST_INCL and DIST_LIST_EXCL attributes contain lists of RDIs. When a BIS belongs to an RD which is a member of one or more confederations, it has several RDIs associated with itself: namely, the RDI of its routeing domain and the RDIs of all confederations listed in managed object **RDC-Config**. Note that the OPEN PDU makes this information available to BISs in an adjacent RDs.

      To accommodate this case (several RDIs associated with a given BIS), the rules for DIST_LIST_INCL should allow the update to be sent to an adjacent BIS whenever one or more of its RDIs are contained in the list. Similarly, the rules for DIST_LIST_EXCL should prevent an UPDATE PDU from being sent to an adjacent BIS whenever one or more of its RDIs are contained in the list.

    - If a BIS (pursuant to its own local policy) elects to restrict the distribution list of a previously advertised route, the BIS shall advertise the previous route as unfeasible to all BISs in adjacent RDs that have been excluded from receiving the new route.

Terminology

In the remainder of this paper, the terms *Loc-RIB*, *Adj-RIB-In*, and *Adj-RIB-Out* have the meanings defined in comment 35 on page 14. Use of this conceptual model clarifies the specific RIBs which are the subject of each of these comments.

24. **Abbreviated form of an unfeasible route:**

    SC6 N6387 advertises an unfeasible route by attaching the UNREACHABLE attribute to the previously feasible route. Thus, the UPDATE PDU that advertises the unfeasible route must contain all the path attributes that were present in the feasible route. An Adj-RIB-In or an Adj-RIB-Out holds at most one route to a particular destination (as specified in the Network Layer Reachability Information), and the distinguishing attributes of the route unambiguously identify a particular Adj-RIB.

    Since the combination of the distinguishing attributes and the NLRI unambiguously identifies a previously feasible route, the following shortcut is possible: to announce that a previously feasible route has become unfeasible, it is sufficient to include the UNREACHABLE attribute in an UPDATE PDU whose path attributes contain only the NLRI and the distinguishing attributes (rather than all the attributes) of the previously feasible route.

25. **Interaction between Decision and Update Processes:**

    SC6 N6387 should require the Decision Process to run to completion based on currently available routes before any newly arrived routes are used in the computation. The following suggested text takes an approach similar to that of DIS 10589:

    ″Since the Adj-RIBs-In are used both to receive inbound UPDATE PDUs and to provide input to the Decision Process, care must be taken that their contents are not modified while the Decision Process is running. That is, the input to the Decision Process shall remain stable while a computation is in progress.

    There are two approaches that could be taken:

    1. The Decision Process can signal when it is running. During this time, any incoming UPDATE PDUs will be queued and will not be written into the Adj-RIBs-In. If more

UPDATE PDUs arrive than can be fit into the allotted queue, they will be dropped and will not be acknowledged.

2. A BIS can maintain two copies of the Adj-RIBs-In—one used by the Decision Process for its computation (call this the Comp-Adj-RIB) and the other to receive inbound UPDATE PDUs (call this the Holding-Adj-RIB). Each time the Decision begins a new computation, the contents of the Holding-Adj-RIB will be copied to the Comp-Adj-RIB: that is, the a snapshot of the Comp-Adj-RIB is used as the input for the Decision Process. The contents of the Comp-Adj-RIB remain stable until a new computation is begun.

The advantage of the first approach is that it takes less memory; the advantage of the second is that inbound UPDATE PDUs will not be dropped. This international standard does not require that either of these methods be used. Any method that guarantees that the input data to the Decision Process will remain stable while a computation is in progress and that is consistent with the conformance requirements of this international standard may be used.″

26. **Checksum Considerations**

a. *Individual Checksums*

A BIS computes a single checksum over the portion of its Loc-RIB(s) that has been advertised to the neighbor BIS, and then sends the checksum to its neighbor BIS as part of a CHECKSUM PDU. (In terms of the more precise conceptual model of comment 35 on page 14, this is equivalent to stating that a ″BIS computes a separate checksum over the Adj-RIBs-Out that have been advertised to the adjacent BIS″.) Checksum verification is achieved when the checksum over the receiving BIS′s Adj-RIB-Ins matches the checksum field of the received CHECKSUM PDU. Thus, the protocol mechanisms expect a BIS to maintain multiple Adj-RIBs-In, one for each of the Adj-RIBs-Out which its partner BIS used when it generated the CHECKSUM PDU. If the receiving BIS does not maintain all the requisite Adj-RIBS-In, then the database integrity check will fail.

However, it is conceivable that the receiving BIS, for reasons of its own (such as local policy or memory constraints) may not wish to provide an Adj-RIB-In for each of its neighbor′s Adj-RIBs-Out: for example, the local BIS′s policies may ignore the information from one of its neighbor′s Adj-RIBs-Out when it runs its local Decision Process. However, if these Adj-RIBs-In were not present (even though their contents will not be used for route computation), it would not be possible to compute a checksum that agrees with the one contained in the CHECKSUM PDUs received from its partner BIS.

Therefore, SC6 N6387 should be modified to allow a BIS to support only those Adj-RIBs-In that actually will be used for route computation. This can be accomplished by having a BIS compute individual checksums over the portion of each of its Adj-RIBs-Out that it advertises to a neighbor BIS, and passing each each checksum passed to its neighbor BIS separately (either in a single CHECKSUM PDU, or in multiple CHECKSUM PDUs).

Since each RIB (Adj-RIB-In, Adj-RIB-Out, and Loc-RIB) is unambiguously identified by its RIB-Att (which consists of a set of distinguishing attributes), the format of the CHECKSUM PDU should be amended to associate each individual checksum value with the RIB-Att of the Adj-RIB-Out over which the checksum was computed. Several <RIB-Att, checksum> pairs can be included in a single CHECKSUM PDU; each RIB-Att field is encoded exactly the same as in an OPEN PDU. (The length of 16 octets for each checksum assumes that the MD4 algorithm is used to generate it.)

Fixed Header
First RIB-Att
First Checksum (16 octets)
Second RIB-Att
Second Checksum (16 octets)
....
Last RIB-Att
Last Checksum (16 octets)

b. *Computation Details*

SC6 N6387 does not provide clear procedural rules for computing the checksums. The following rules are suggested, and should be integrated into the normative text of SC6 N6387:

> When a BIS computes a checksum over an individual information base (Adj-RIB-Out when the BIS is generating a CHECKSUM PDU, or an Adj-RIB-In when a BIS is computing a checksum to be compared with the value contained in an incoming CHECKSUM PDU), the following rules shall be observed:
>
> 1. A sequence number shall be associated with each route in the information base: it shall be the sequence number of the BISPDU used to transmit (for Adj-RIB-Out) or receive (for Adj-RIB-In) the UPDATE PDU that contains the given route.
> 2. For purposes of computing a checksum over the routes contained in a single information base, the following sorting sequence shall be observed:
>   a. Routes shall be sorted in a non-decreasing order of their sequence numbers:
>   1) Within each route, path attributes shall be sorted in a non-decreasing order based on their type codes
>   2) Within each route, Network Layer Reachability Information shall be sorted in lexicographical order, based on the binary value of the prefixes that it contains.
> 3. A checksum shall be computed according to the MD4 algorithm. The algorithm shall be applied to the data as sorted by the previous rules, and the sorted data shall be treated as a sequence of octets.

27. **Handling BIS Overload**

Due to misconfiguration or certain transitory conditions, it is possible that there may be insufficient resources available at a particular BIS to correctly implement the procedures of IDRP. There are two different overload conditions: memory overload and CPU overload. For IDRP-based information exchange methods, CPU overloads are potentially more harmful than memory overloads because the IDRP route computation phase must precede its route distribution phase. That is, in a CPU-overload situation, the UPDATE process should be halted and priority should be given to the completion of the Decision Process's computation.

Since the remedies suggested for memory overload could be construed as enactment of local policies, this material is advisory in nature, and should be located in a new informative annex.

However, the material dealing with CPU overload should be part of the normative text of SC6 N6387. A general discussion of how to handle overload cases follows below:

- Memory Overload:

  We say that a BIS becomes memory-overloaded when there is not enough memory to maintain both the Adj-RIBs-In (which store the routeing information as received from other BISs), the Loc-RIBs (that are derived by the decision process, using the information in the Adj-RIBs-In as one of the inputs), and the Adj-RIBs-Out which (which hold information to be transmitted to neighbor BISs).

  Since the Loc-RIBs form a subset of the Adj-RIBs-In, the amount of memory needed to store the Adj-RIBs-In is greater than or equal to the amount of memory needed to store the Loc-RIBs. Therefore, the first step to alleviate the memory overload condition would be to reduce the amount of information that is stored in Adj-RIBs-In. That can be accomplished by removing routes that are not in the Loc-RIBs (that is, those routes that have not been selected by the Decision Process for advertisement to other BISs).

  Clearly, all routes in Adj-RIBs-In to destinations that are not in the Loc-RIB may be removed with no negative impact. Even if Adj-RIBs-In have routes to destinations that are in the Loc-RIBs as well, it still may be possible to remove some of these routes. Since these routes may potentially be used as a fallback routes (if the current route that is in the Loc-RIB becomes unfeasible), removing them from the Adj-RIBs-In may cause at some point in the future suboptimal connectivity.

  If several Adj-RIBs-In (that have the same RIB attribute) have routes to the same destination, then routes with higher degree of preference (as computed by the local BIS) should be retained, while routes with lower degree of preference may be deleted, thus reducing the amount of memory needed for the Adj-RIBs-In. To ensure routeing consistency within an RD, the above procedure may be applied only to the Adj-RIBs-In associated with BISs in adjacent RDs.

  (Note that if a BIS unilaterally deletes a route, then it will not be able to compute a correct checksum for comparison to the CHECKSUM PDU received from a neighbor. Note also that any solicited refresh (see comment 28 on page 10) will only serve to reinstate the deleted route. Hence, if the condition persists, the memory-overloaded BIS should close the IDRP connection, and then take corrective action, such as re-opening it with an OPEN PDU that indicates support for a smaller **RIB-ATTsSet**, for example.

  A more drastic measure would be to terminate one or more of the IDRP sessions with other BISs. That would result in releasing the memory that was previously used to store the Adj-RIB-Ins and the Adj-RIBs-Out associated with that BIS. To ensure routeing consistency within an RD this measure may be applied only to the IDRP sessions with BISs in adjacent RDs. If the above measures do not alleviate the memory overload condition, the local BIS terminates all of its IDRP sessions.

- CPU Overload:

  We say that a BIS becomes CPU overloaded when there is not enough CPU processing power to process incoming BISPDUs received from other BISs. In this situation BIS must continue to update the Adj-RIBs-In with information contained in BISPDUs received from other BISs, but may not run the Decision Process using this information except for routes received with the UNREACHABLE path attribute.

If a route received in the UPDATE_PDU has the UNREACHABLE path attribute, the local BIS checks whether this route is currently installed in one of the Loc-RIBs; if so, it removes it from the appropriate Loc-RIB, updates the appropriate FIB, and generates (if necessary) an UPDATE-PDU to inform other BIS′s of the change in its Loc-RIBs and its Adj-RIBs-Out. The Decision Process on the local BIS does not select another to replace the one that becomes unfeasible.

Since this procedure decreases the size of the Loc-RIB, persistence of a CPU overload condition can eventually deplete the entire Loc-RIB, thus making the BIS unavailable as an intermediate system. If the CPU overload condition disappears, then the Decision Process and Update Process should be run over all the new routes that were installed into the Adj-RIBs but have not yet been processed by the Decision Process. If the CPU overload condition persists for more than the predefined architectural constant **MaxCPUOverloadPeriod**, the local BIS terminates its IDRP sessions.

The order of termination of the IDRP sessions is significant. First the BIS may terminate one or more of the IDRP sessions with BISs in adjacent RDs. If after terminating IDRP sessions with all of the BISs in adjacent RDs the CPU overload still persists, the BIS terminates the rest of its IDRP sessions (with all the BISs within its own RD).

28. **Solicited and unsolicited refresh of Adj-RIBs:**

In certain situations (e.g. memory overload) a BIS may have to purge some of the routeing information stored in its Adj-RIBs-In. If such purges occur, the Database integrity scheme (CHECKSUM PDU) will not work correctly. Therefore, we propose to add the solicited and unsolicited Adj-RIB-In refresh capability to the IDRP. Addition of such capability requires a new type of BISPDU, called the RIB-REFRESH PDU, with the following format:

Fixed Header
OpCode (1 octet)
Variable part

Currently defined OpCode are:
1 - RIB-Refresh-Request
2 - RIB-Refresh-Start
3 - RIB-Refresh-End

- *Solicited Refresh*

  A BIS may request the refresh of one or more of its Adj-RIBs-In by sending a RIB-REFRESH PDU that contains the OpCode for RIB-Refresh-Request, and it can restrict the scope of refresh by using its variable part to specify the RIB-Att of the Adj-RIB-In that it wants to refresh.

  When a BIS receives a RIB-REFRESH PDU with OpCode RIB-Refresh-Request, it sends back RIB-REFRESH PDU with OpCode RIB-Refresh-Start, followed by a sequence of UPDATE PDUs that contain the information in its Adj-RIBs-Out that have been advertised to the requesting BIS. The completion of the refresh procedure is indicated by sending the RIB-REFRESH PDU with OpCode RIB-Refresh-End.

- *Unsolicited Refresh*

  A BIS may initiate an unsolicited refresh by sending a RIB-REFRESH PDU with OpCode RIB-Request-Start, followed by a sequence of UPDATE PDUs that contain the information in its Adj-RIBs-Out that been advertised to a given BIS. The completion of the refresh is indicated by sending the RIB-REFRESH PDU with OpCode RIB-Refresh-End.

If the refreshing BIS receives a RIB-Refresh-Request while it is in the middle of refresh (after sending RIB-REFRESH PDU with OpCode RIB-Refresh-Start, but before sending RIB-REFRESH PDU with OpCode RIB-Refresh-End), then the current refresh is aborted and the new refresh is initiated.

If the BIS being refreshed receives a RIB-Refresh-Start in the middle of refresh (after receiving a RIB-REFRESH PDU with OpCode RIB-Refresh-Start, but before receiving the RIB-REFRESH PDU with OpCode RIB-Refresh-End), then the current refresh is aborted and the new refresh is initiated. That is, a refresh cycle may be terminated either by receipt of RIB-Refresh-End or by receipt of a new RIB-Refresh-Start.

If the OpCode is RIB-Refresh-Request, then the variable part of the RIB-REFRESH PDU contains the RIB-Atts of the Adj-RIBs for which a refresh is being requested. For all other OpCodes the Variable Part is empty.

29. **Updating RD_PATH for Confederations**

Clause 7.17.7 contains several errors and inconsistencies, caused by inaccurate editing of the material presented in Sydney as temporary document 2S36. Therefore, the current text of clause 7.17.7 b3 (right hand column) should be amended as follows:

> 3) If an intervening 2-tuple (for example, <Entry Marker><RDC-Y>)
> is found, and RDC-X is known to be nested within RDC-Y, then:
>
> -if there are no RDIs listed between the 2-tuple for RDC-X and the
> 2-tuple for RDC-Y, then RDC-Y and RDC-X were entered simultaneously.
> In this case, the contents of the RD_PATH attribute from the end
> back to the 2-tuple for RDC-X shall be replaced by the sequence
> <Entry Marker><RDC-Y>.
>
> -if there are any RDIs listed between the 2-tuple for RDC-X and the
> the 2-tuple for RDC-Y, this indicates that RDC-X is not in fact
> nested within RDC-Y, and the route is in error. The local BIS shall
> send an IDRP_ERROR PDU that reports a Misconfigured_RDCs error.

30. **Stability of Routes**

The description of the process should be expanded to recognize that an RDI contained in the RD_PATH attribute can refer to either a routeing domain or a confederation. This can be accomplished by the following changes to the second ″dashed″ item under ″b″ of clause 7.13.2:

- Change ″when the local RD is included in the RD_PATH attribute...″ to ″when the RDI of either the local RD or any confederation to which it belongs is included in the RD_PATH attribute...″
- Clarify the procedure for finding the suffix of an RD_PATH by adding text to say that the suffix of an RD_PATH is obtained by deleting all leading path segments of the RD_PATH attribute, up to and including the path segment that contains the RDI of the local RD or any confederation to which it belongs.

- Explain that inconsistent policies exist if the path segments of the suffix do not match the trailing path segments of the currently selected route. (This accounts for the fact that the RD_PATH attribute of UPDATE PDUs advertised to destinations within the local BIS′s lowest level RDC may contain some RDIs that later will not be listed explicitly when the route is advertised outside of a higher level confederation, using the procedures of clause 7.17.7.)
- Provide an informative example showing typical cases of consistent or inconsistent policies, for representative mixes of RDs and RDCs.

31. **Detecting Inconsistencies**

The procedures outlined in item c of clause 7.12.7 are of limited value, since they can not definitively discriminate between inconsistencies of a transient nature (due to ongoing topology changes) and of a permanent nature (due to flawed operations over a stable topology).

We recommend that item c be deleted, and that in its place the following informative note be inserted:

Note: If, for a given value of NLRI, all UPDATE PDUs received from BISs located in a single adjacent routeing domain do not contain identical path attributes, except for NEXT_HOP and LOCAL_PREF, then there is an external inconsistency. Such inconsistencies may be permanent (for example, because of a flawed decision process) or transient (for example, the UPDATE PDUs from all BISs in an adjacent RD can not all be received simultaneously). The detection of, and response to, external inconsistencies is left as a local policy matter.

32. **RD_HOP_COUNT**

Knowledge of an RD-hop-count is potentially an important input to many routeing policies, but SC6 N6387 does not currently provide this information. Since the RD_PATH attribute may contain RDIs for both routeing domains and confederations, it is not possible to infer the number of routeing domains that are traversed by examining only the RD_PATH attribute. Therefore, the text of SC6 N6387 should be amended to define a new well-known mandatory attribute, RD_HOP_COUNT. Its value field will be 1 octet long, and the integer value it contains will be the cumulative total of routeing domain that an UPDATE PDU has traversed.

The usage rules will be as follows:

a. The default value of this attribute is 0.

b. Before sending an UPDATE PDU to a BIS located in an adjacent routeing domain, a BIS shall increment the value of this attribute by 1, and shall place the result in the RD_HOP_COUNT field of the outbound UPDATE PDU.

c. A BIS shall not increment the value of this attribute when it sends an UPDATE PDU to another BIS located in its own routeing domain.

Note: ISO 8473 limits the maximum lifetime of an NPDU to 256 counts, and requires each Network entity processing a given NPDU to decrement that NPDU′s lifetime by at least 1 count. In the limiting case of one BIS per routeing domain, this implies that a NPDU′s lifetime will expire before it can reach the 257th RD. Hence, there is no need to address an RD_HOP_COUNT greater than 256.

Part III: Editorial Comments

33. **More Descriptive Title**:

The current title for SC6 N6387 can in many cases lead to confusion on the part of new readers: it is long, awkwardly constructed, and almost identical to the title of DIS 10589 (″inter″ vs. ″intra″). Since SC6 N6387 deals primarily with the protocol for exchanging routeing information, and it defines the rules which must be followed for advertising locally selected routes to other systems that are participating in an instance of IDRP, the USA suggests that the title of SC6 N6387 be changed to *Protocol for the Exchange of Inter-domain Routeing Information among Intermediate Systems*.

34. **Usage of the MULTI_EXIT_DISC attribute**

The MULTI_EXIT_DISC attribute can be used as a means for exchanging routeing information between the inter-domain protocol and an intra-domain protocol. However, SC6 N6387 provides no clarifying examples for the use of the MULTI_EXIT_DISC attribute. The USA submits the following illustrative example, and suggests that it be incorporated in a new informative annex to SC6 N6387:

EXAMPLE OF MULTI-EXIT_DISC USAGE

The MULTI-EXIT DISC attribute can be used to provide a limited form of multi-path (load-splitting), as is shown in the following examples.

- Example 1 (see Figure 1 on page 14):

  Consider the case when a BIS A located in routeing domain RD-A has two adjacent BISs (B1 and B2) that belong to the routeing domain RD-B. Assume that RD-B has Network Layer Reachability information about NSAPs N1, N2, ... Nk, and it wants to advertise this information to RD-A. By using the MULTI-EXIT_DISC attribute RD-B may do selective load splitting (based on NSAP addresses) between B1 and B2.

  For example, BIS B1 advertises to BIS A Network Layer Reachability information N1, N2, ... Nm with the MULTI_EXIT_DISC set to X, and advertises N(m+1), ... Nk with the MULTI_EXIT_DISC set to X + 1.

  Similarly, BIS B2 advertises to BIS A Network Layer Reachability information N1, N2, ... Nm with the MULTI_EXIT_DISC set to X + 1, and advertises N(m+1), ... Nk with the MULTI_EXIT_DISC set to X.

  As a result, traffic from BIS A that destined to N1, N2, ... Nm will flow through BIS B1, while traffic from BIS A that destined to N(m+1), ... Nk will flow through BIS B2. This scenario illustrates the simplest way of doing limited multipath with IDRP.

- Example 2 (see Figure 2 on page 15):

  Next consider more complex case where there is a multihomed routeing domain RD-A that has only slow speed links. RD-A is connected at several points to a transit routeing domain RD-B that has only high speed links; BIS A1 is adjacent to BIS B1, and BIS A2 is adjacent to BIS B2. RD-A wants to minimize the distance that incoming NPDUs addressed to certain ESs—say ES(1) through ES(k)—will have to travel within RD-A.

  One way of doing this is by making BIS A1 to announce to BIS B1 destinations ES(1) – ES(k) with a lower MULTI_EXIT_DISC, as compared to the MULTI_EXIT_DISC that BIS A2 will use when announcing the same destinations to the BIS B2. Similarly, BIS A2 would announce to

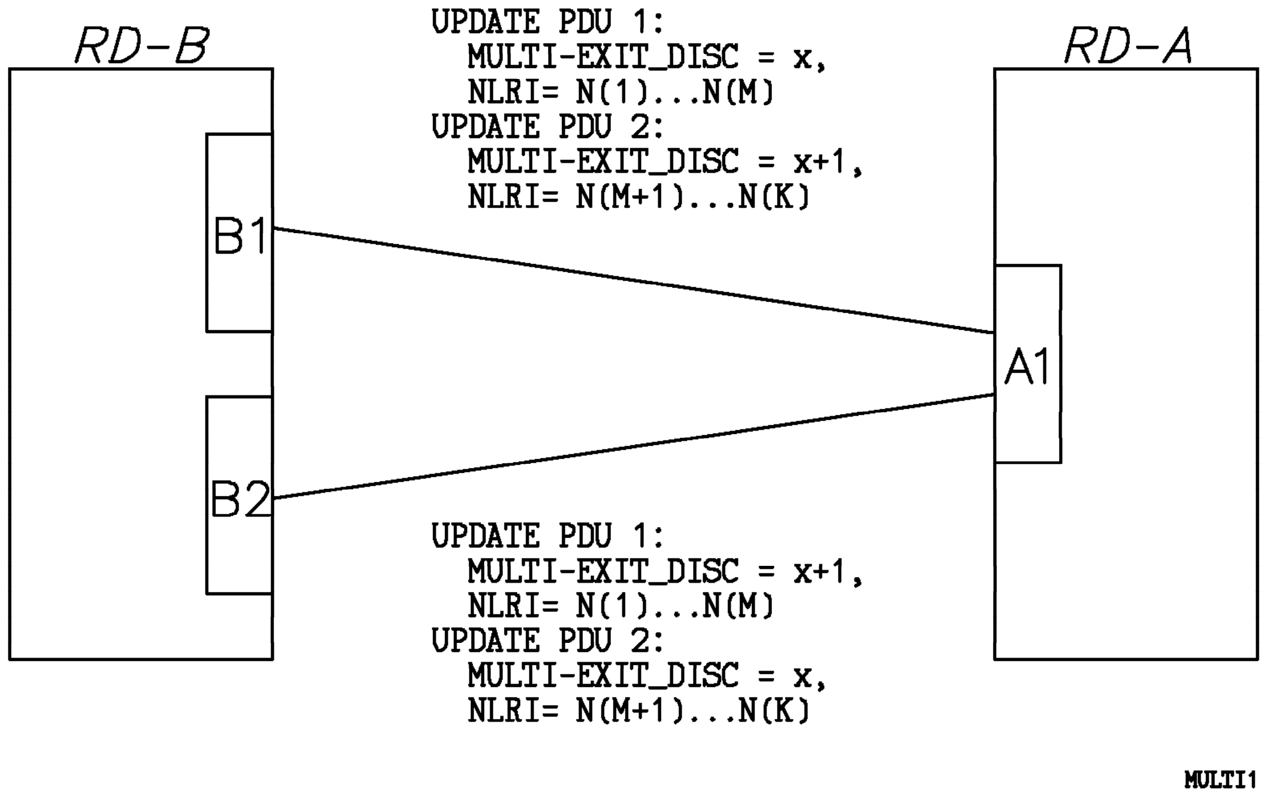


Figure 1. Example 1 Configuration

BIS B2 destinations ES(k=1)– , ES(n) within the RD-A that are closer to the BIS A2 (than to the BIS A1) with the lower MULTI_EXIT_DISC, as compared to the MULTI_EXIT_DISC that the BIS A1 will use when announcing the same destinations to the BIS B1.

When traffic that destined to some ES within RD-A enters RD-B on its way to RD-A via BIS X, X picks up the exit BIS that has the lowest MULTI_EXIT_DISC value for that destination. For example, X may pick up BIS A2 as an exit, even if the distance between A2 and X is greater than the distance between A1 and X.

35. **Conceptual Model for RIB(s)**

SC6 N6387 describes a routeing information base as being composed of two conceptual parts: the Loc-RIB(s) and the Adj-RIB(s). In this model, the Adj-RIB(s) are the repository for routeing information received from other BISs.

Note: The use of a conceptual model which distinguishes between Adj-RIBs-IN, Adj-RIBs-Out, and Loc-RIBs in no way implies that an implementation must maintain 3 separate copies of the routeing information. The choice of implementation (3 separate copies, one copy with pointers, etc.) is not constrained by this standard.

Additional clarity can be obtained by revising the conceptual model so that it contains three parts:

- Loc-RIB(s): This part of the model remains unchanged from that of SC6 N6387.

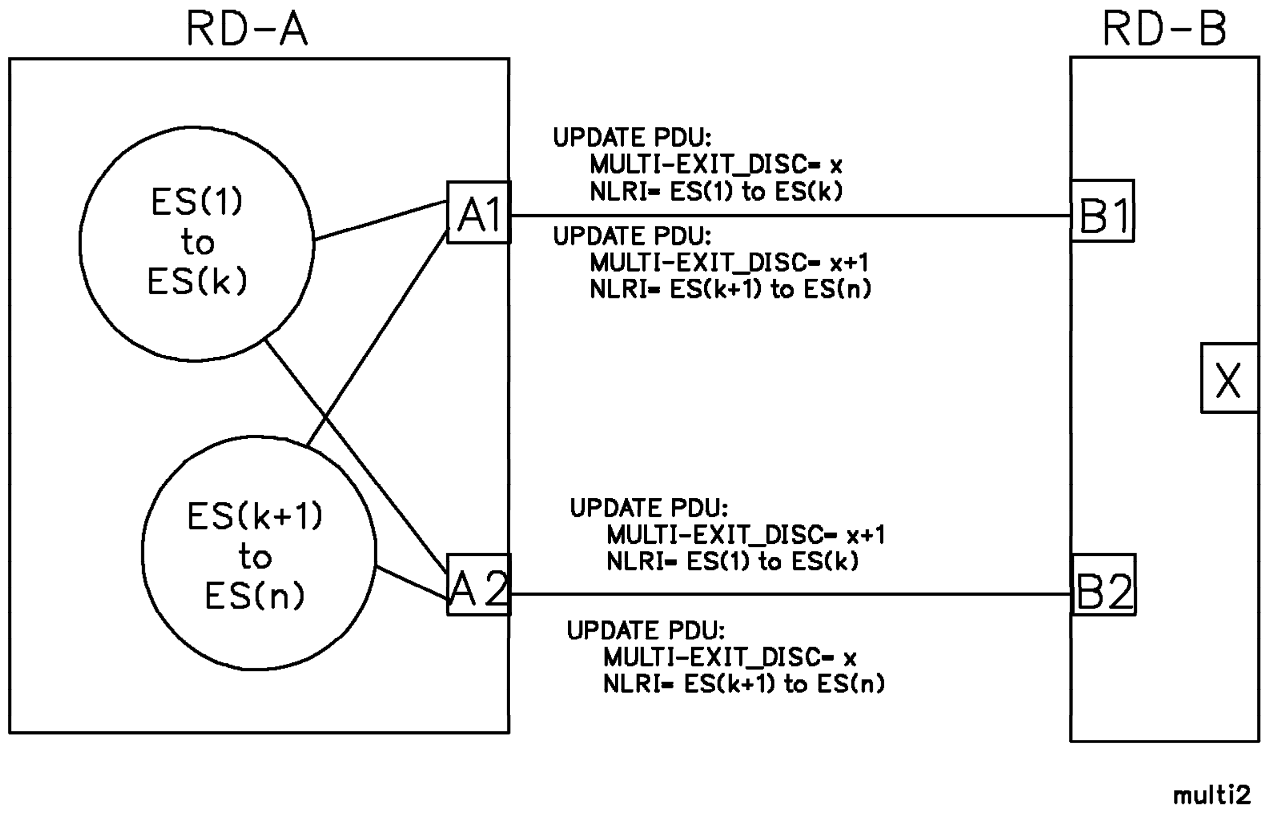

RD-A
RD-B
ES(1) to ES(k)
ES(k+1) to ES(n)
A1
A2
B1
B2
X
UPDATE PDU: MULTI-EXIT_DISC= x NLRI= ES(1) to ES(k)
UPDATE PDU: MULTI-EXIT_DISC= x+1 NLRI= ES(k+1) to ES(n)
UPDATE PDU: MULTI-EXIT_DISC= x+1 NLRI= ES(1) to ES(k)
UPDATE PDU: MULTI-EXIT_DISC= x NLRI= ES(k+1) to ES(n)
multi2

Figure 2. Example 2 Configuration

- Adj-RIB(s)-In: This part of the model identifies the storage used to hold incoming routeing information received from other BISs. (It corresponds to ″Adj-RIB(s)″ of SC6 N6387.)
- Adj-RIB(s)-Out: This is the new part of the model. It identifies the portion of storage used to hold routes which will be advertised to other BISs. Note from the picture that it is legitimate, for example, to have only one Adj-RIB-Out for a given neighbor even if there are several Loc-RIBs. This could occur, for example, if the receiving BIS only supported the Default Attribute while the advertising BIS supported the Default Attribute plus others.

Figures 4 and 8 of SC6 N6387 would be replaced with the Figure 3 on page 16 and Figure 4 on page 17

36. **Conformance-related Language:**

The following comments list several sections of SC6 N6387 in which the language needs to be amended to be consistent with the conformance requirements for the standard:

a. Clause 6.3: The ″must″ and ″may″ should be changed to ″shall″ in the descriptions of the various flag bits of the Path Attribute field.

b. Page 14, last sentence under DIST_LIST_INCL: ″can not″-->″shall not″

c. Clause 6.4: In items ″a″, ″b″, and ″c″ of the first paragraph: ″will″--> ″shall″

d. Clause 7.5.4, next-to-last paragraph: ″will″--> ″shall″.

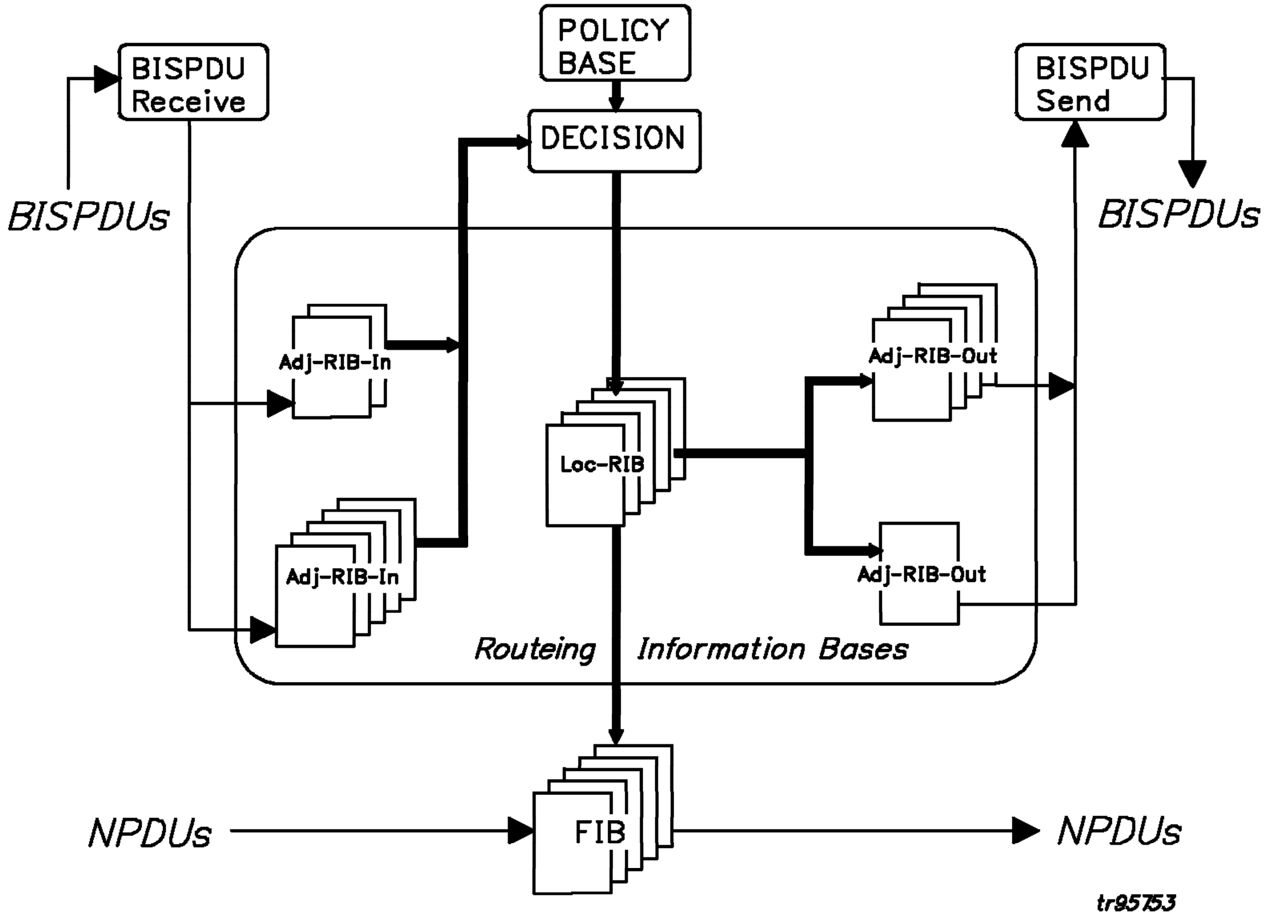


Figure 3. Replacement for SC6 N6387 Figure 4

e. Clause 7.9, first sentence:″must be all ones″--> ″shall consist of all one bits″.

f. Clause 7.11.2, first sentence (2 occurrences), ″must″-->″shall″.

g. Clause 7.12.1, second paragraph, ″must be able″ --> ″shall″ (2 places)

37. **Use of ″Notes″ for Parenthetical Material:**

There are several sections where the written material should be presented as a ″Note″ rather instead of being incorporated directly into the body of the standard:

a. Clause 7.1.3: The parenthetical material at the end of this clause should be presented as a Note.

b. The last paragraph of this 7.4.4 should be presented as a NOTE, because no particular algorithm is specified within IDRP.

c. The very last paragraph of 7.4.5 should be presented as a NOTE.

d. Clause 7.6.4: The end of this paragraph, beginning with ″Any error...″ should be presented as a NOTE.

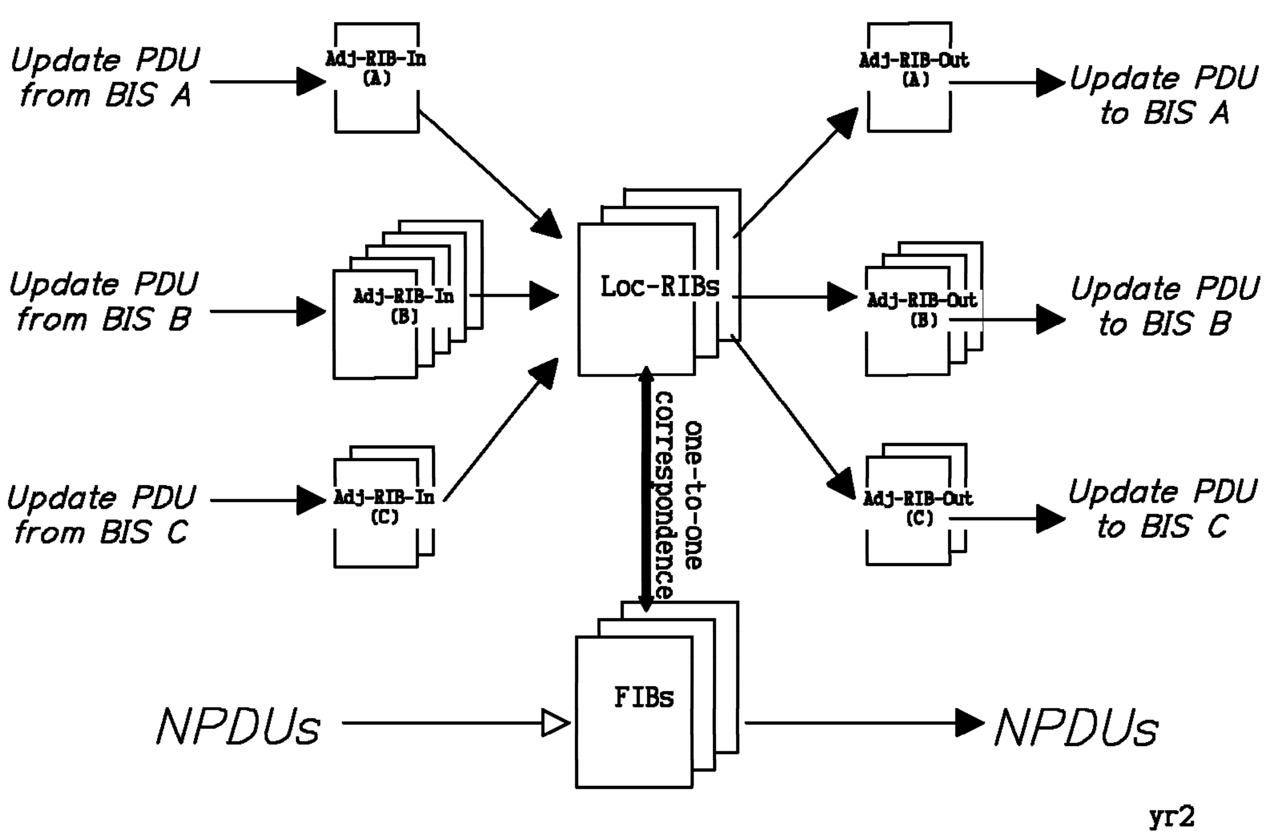


Figure 4. Replacement for SC6 N6387 Figure 8

e. Clause 7.11.1, last paragraph (top of page 29) should be presented in a NOTE.

38. **Miscellaneous:**

a. Include ″SPI″ in the abbreviations list.

b. Add a definition for ″Policy Information Base″ to clause 3.7.

c. Clause 5.2; Reword the first two sentences: ″The direct exchange of policy information is outside the scope of IDRP. Instead, IDRP communicates ...in its UPDATE PDUs which reflect the effects....″

Remove the words ″own local″ from the first sentence of the second paragraph.

d. Clause 5.4: Delete the last sentence under bullet ″a″.

e. Clause 5.5: delete the word ″allows″ in the first sentence

f. Clause 6.1: Page 11, first table: 2 octet==> 2 octets.

Delete the word ″currently″ from last sentence describing the TYPE field.

g. Clause 6.2, page 12, ″RIB-ATTSet″, 2nd paragraph: change ″all of the RDCs″ to read ″all of the RIB-Atts″

h. Figure 5: The attribute type 1 should be labelled ″EXT_INFO″

i. Clause 7.11.11: It needs to be noted that the values of RDLRE and RRE are positive integers: they are not the actual numerical value of the associated error rate. The numerical value of the associated error rate is obtained by dividing the integer value by $2^{32} - 1$.

The formula needs to be adjusted for consistency of units in order to account for this, as follows:

$$K \times (1 - [(1 - RRE/K) \times (1 - RDLRE/K)])$$

The words ″...and K is the constant $2^{32} - 1$″ also need to be added to the text following this formula to explain the notation *k*.

j. Page 15, MULTI-EXIT_DISC: The words ″unsigned non-negative″ in the first sentence under MULTI-EXIT_DISC are redundant: delete ″unsigned″.

k. Page 15,16, Source and Destination QOS: The last sentence of the descriptions of the address fields speaks of ″the indicated QOS type″. References to clauses 7.11.13 and 7.11.14 would serve to clarify what this phrase actually means.

l. Clause 6.4: Provide a reference to clause 7.4.2 to clarify how sequence numbers are chosen for this PDU.

m. To avoid confusion with management notifications, it would be desirable to change the name NOTIFICATION PDU″ to ″IDRP Error PDU″.

n. The material in clause 7.1.1 belongs in clause 7.1, not in a subclause.

o. Clause 7.1.3.1: characters--> octets.

p. Clause 7.2.1, second item: For clarity, reword as follows: ″support inter-domain links to at least two different routeing domains...″

q. Clause 7.2.2, item ″a″: second clause of first sentence, change to ″must reside in that routeing domain.″

r. Clause 7.3, item ″a″ on page 20: For clarity, this item should be be reworded to reflect the fact that an IS to which intra-domain traffic is handed can be either adjacent to the local BIS, or may be co-located with the local BIS. The following amended text is suggested for item ″a″:

> The managed object **INTRA-IS** lists the NETs of systems to which the local BIS may deliver an inbound NPDU whose destination lies within the BIS′s routeing domain. This managed object contains the NETs of any IS that supports the intra-domain routeing protocol and is located on the same common subnetwork as the local BIS. In particular, if the BIS participates in the intra-domain routeing protocol (that is, the protocol machines for both inter- and intra-domain routeing are located in the same open system), then the NET of the local BIS will be listed in the managed object **INTRA-IS**.

s. Clause 7.4.3: For clarity, reword the first sentence of second paragraph: ″Acknowledgements can be carried in the headers of any type of BISPDU.″

In the third sentence, ″last correctly received″ is sloppy wording. For example, it doesn′t consider the problem of mis-sequencing.

In the middle of the second paragraph, the word ″prior″ should be defined, especially if the sequence numbers are permitted to wrap.

t. Clause 7.4.4, first and second sentences: ″must″--> ″shall″

u. Clause 7.4.5: Define ″left″ and ″right″.

Note that the third paragraph doesn′t apply to BISPDUs that don′t increment a sequence number.

v. Clause 7.5: Change the first sentence: ″The protocol described...on the underlying Network layer protocol to establish...between each pair of BISs.″

w. Clause 7.5.4: third item: delete ″then″, add ″state″ as the very last word.

x. Clause 7.6.2: first item: ″largest locally supported version number″ --> ″highest supported version number″.

first item in right hand column: ″different than″-->″different from that″

In the case of authentication failure, management should also be notified.

In the last paragraph in the second column, change ″UPDATE PDU″ to ″OPEN PDU″.

y. Clause 7.6.3, second item: NOTIFICATION PDU contains″--> ″IDRP Error PDU shall contain″

The note in the right hand column could be reworded for clarity, by deleting the current last sentence, and inserting the following as a new first sentence: ″It is permissible for an UPDATE PDU to contain neither the DIST_LIST_INCL nor the DIST_LIST_EXCL attributes.″

Delete reference to ″syntactic correctness″ in the last paragraph in the left-hand column of page 25 and in the first paragraph of page 26: IDRP assumes that addressing information is syntactically correct, but does not explicitly check it for correctness.

z. Clause 7.9: It would be preferable not to use zero as an authentication code. Any other value would be suitable.

Right-hand column, second dashed item: ″will define″--> ″specifies″

Last paragraph, last sentence: Change to ″Path attributes are summarized in Table 1; their encoding is described....″

aa. Clause 7.10.2, item ″c″, page 28; Define ″equivalent″.

ab. Clause 7.11.7: The text should state that the MULTI-EXIT_DISC attribute should be applied on a ″per QOS″ basis.

ac. Clause 7.11.14, second paragraph: ″SOURCE SPECIFIC″-->″DESTINATION SPECIFIC″

ad. Clause 7.12.8, second line at top of page 37: ″router″-->″BIS″

ae. Clause 7.12.9, first paragraph: In last two sentences: ″will perform″-->″performs″, ″will be advertised″-->″is advertised″, ″will aggregate″-->″may aggregate″.

In item 8 under **RD_PATH**, change ″steps 1 to 7″ to read ″steps 2 to 7″.

af. Clause 7.12.10: In several places, change ″external inter-domain links″ to ″inter-domain links″.

ag. Clause 7.13.2: Text is needed to describe what is meant by ″replacement route″, and how it would be distinguished from a new route.

ah. Clause 7.15.1: delete the parenthetical material

ai. Clause 7.17: The first paragraph needs to clarify the phrase ″Confederations can be disjoint...″. In this context, ″disjoint″ refers to confederations which are have no members in common. However, text needs to be added to reinforce the fact that within any given RDC, all of its members (which could be RDs or RDCs) must form a connected set: that is, ″disjointedness″ is allowed between confederations, but not within a confederation.

aj. Clause A.3.4.1, first paragraph: change ″appliesory,...″ to ″applies—mandatory,...″.

Appendix A. Authentication based on MD4 Message Digest Algorithm

A digital signature scheme for authentication should be included in the normative parts of SC6 N6387. RFC 1186 gives a generic description of an algorithm known as MD4, whose output is a 128-bit long ″message digest″ (or ″digital signature″) for an input string of arbitrary length.

In terms of the authentication approach outlined in IDRP′s Annex B, the MD4 algorithm can serve as the function F_{hash}. First, MD4 generates a 128-bit digital signature for the contents of a BISPDU; and for authentication, the cleartext signature (not the entire BISPDU) is encrypted and placed in the ″Marker″ field of the BISPDU. In security terminology, the digital signature provides ″integrity″ for the contents of the BISPDU.

The authentication procedure using MD4 does not provide confidentiality for the contents of the entire BISPDU. If confidentiality and integrity are needed for the entire BISPDU, then means outside of IDRP can be used, such as the emerging Network layer security protocol, for example.

In IDRP, the ″Marker″ field of the BISPDU′s fixed header is intended to be used by by an authentication function, which itself is described by the the ″Authentication Code″ and ″Authentication Data″ fields of the OPEN PDU: for example, the ″authentication code″ might point to a particular cryptographic association between a pair of BISs. Since the OPEN PDU describes the characteristics of a single BIS-BIS connection, the authentication code applies only on a pairwise basis. Thus, it is possible for a given BIS to implement several distinct authentication functions: for example, a different authentication method could be used with each of its neighbors.

Interaction with BISPDU Checksum:

Currently, IDRP defines a ″checksum″ field and a ″Marker″ field in the header of each BISPDU. Considering the fact that the MD4 algorithm is applied to the contents of the whole BISPDU, it is obvious that there is some redundancy between functions of these fields, since each of them provide an integrity check on the contents of the BISPDU. Therefore, it is recommended that the existing checksum field be deleted from the BISPDU header, and the ″Marker″ field be used to support the checksum function as well as the authentication function.

The semantics of the contents of the ″Marker″ field will be determined by the ″Authentication Code″ carried in the OPEN PDU for a given BIS-BIS connection:

- If the authentication code has a value of 1, then the ″Marker″ field shall contain the unencrypted output of the MD4 message digest algorithm. In this case, the ″Marker″ field is used to provide a checksum for the contents of the BISPDU

- If the authentication code has a value of 2, then the ″Marker″ field shall contain the encrypted output of the MD4 message digest algorithm, as described below. In this case, the ″Marker″ field provides an authentication function and a checksum function.

The proposed dual-function usage of the ″Marker″ field avoids the use of two checksum procedures for IDRP: to do this, the text should be amended to state that the IDRP checksum algorithm will be the MD4 message digest algorithm, rather than the currently specified Fletcher checksum algorithm.

Generating the Contents of the ″Marker″ Field:

When the authentication type code is equal to 2, then the contents of the ″Marker″ field are generated by the following process, illustrated in Figure 5:

1. Apply the MD4 algorithm as described in section 3 of RFC 1186 to the message consisting of all fields of the BISPDU, with the ″Marker″ field initially set to all 0′s. This will yield a 128-bit (16 octet) cleartext quantity which we will refer to as *mcksum*.
2. Encrypt the 16-octet ″mcksum″ to obtain the corresponding 16-octet quantity ″Marker″, which is placed in the ″Marker″ field of the BISPDU.

   Note: The encryption algorithm must be agreed upon in the cryptographic association set up by the two BISs involved in the authentication process. Different BIS-pairs are free to choose different encryption algorithms. IDRP does not mandate use of a specific encryption algorithm. Explicit indication of the specific algorithm to be used is outside the scope of IDRP, but the ″Authentication Code″ field of IDRP′s OPEN PDU can be used to specify an algorithm indirectly in accordance with the local agreements of the two communicating BISs.

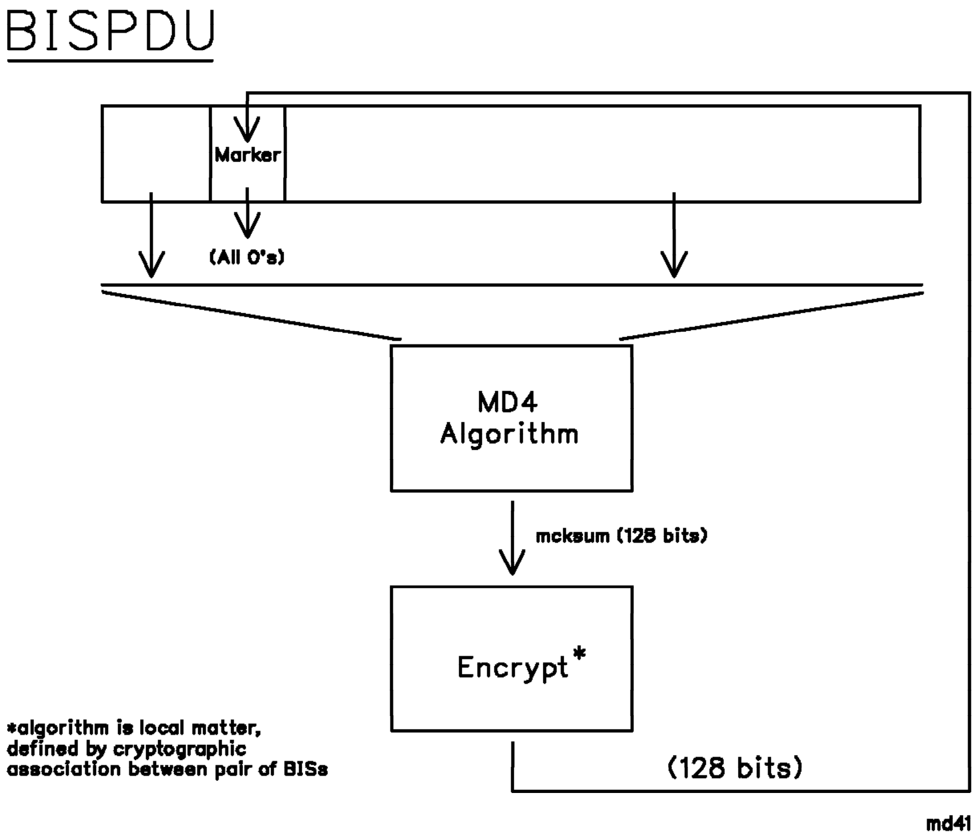

BISPDU
Marker
(All 0's)
MD4 Algorithm
mcksum (128 bits)
Encrypt*
*algorithm is local matter, defined by cryptographic association between pair of BISs
(128 bits)
md41

Figure 5. Generating the Marker Field

Validating a Received ″Marker″ Field:

The contents of the ″Marker″ field of an incoming BISPDU is validated by the following process, depending upon the ″authentication type code″ that was used in the OPEN PDU for a given BIS-BIS connection:

1. Generate a cleartext ″Reference Marker″ for the contents of the received BISPDU: that is, apply the MD4 message digest procedures to the entire BISPDU, setting the ″Marker″ field to all 0′s. Call this cleartext value the ″Reference Marker″.
2. Generate a cleartext ″Test Marker″:
   - If the authentication type is 1, the ″Test Marker″ is equal to the ″Marker″ field in the received BISPDU
   - If the authentication type is 2, the ″Test Marker″ is equal to the decrypted value of the ″Marker″ field of the received BISPDU.
3. If the ″Test Marker″ and the ″Reference Marker″ (both cleartext) agree with each other, accept the BISPDU.
4. If the ″Test Marker″ and the ″Reference Marker″ disagree, then:
   - If the authentication type is 1, discard the PDU because the checksum has failed (see clause 7.6.1 of SC6 N6387)
   - If the authentication type is 2, inform system management that an authentication failure has occurred. (An IDRP Error PDU is not sent, because the source of the received BISPDU has not been authenticated.)

Appendix B. HIERARCHICAL RECORDING Attribute

Currently, IDRP controls the transitivity of externally generated NPDUs through a routeing domain by means of the DIST_LIST_INCL and DIST_LIST_EXCL path attributes. These lists explicitly enumerate RDs that can (or can not) receive the routeing information contained in a given UPDATE PDU. These lists permits a given RD to control use of its resources by other RDs. Since data NPDUs flow in a direction opposite to the UPDATE PDUs, an RD indirectly controls the flow of user NPDUs by controlling where it allows its own UPDATE PDUS to be sent,

The HIERARCHICAL RECORDING attribute can be redefined to provide an additional optional means for controlling transitivity through a routeing domain. If the transitivity constraints imposed by the proposed optional method are not suitable for a particular RD, that RD can still employ the currently defined methods (DIST_LISTs).

Proposed Additional Method:

If the Routeing Domain Confederations are supported, then the associated hierarchical relationships form a basis for imposing transitivity constraints on traffic flow within a given RDC: the constraints will be based on the value of the HIERARCHICAL RECORDING attribute and the knowledge of whether it is necessary to enter or exit a confederation in order to reach an adjacent RD (See SC6 N6387, clause 7.17.4).

Proposed New Attribute Usage Rules:

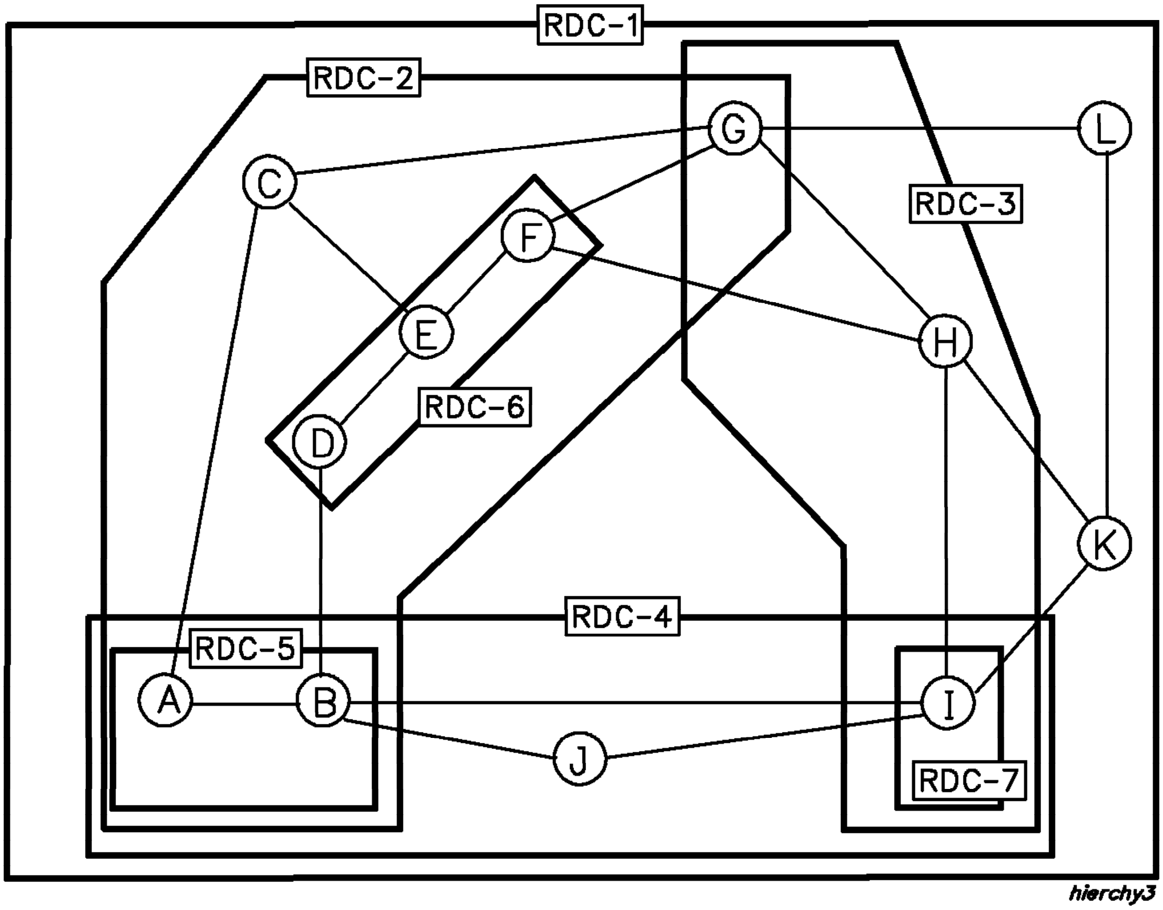


Figure 6. Transitivity and Nesting Relationships

The HIERARCHICAL RECORDING attribute constrains the destinations to which an UPDATE PDU may be propagated. It should remain as a well-known discretionary attribute. Its meaning is significant only within an RDC. Thus, it should not be included in an UPDATE PDU that is transmitted to a BIS located in a confederation that does not intersect or overlap with at least one of the RDC(s) to which the BIS belongs. It is a single octet in length, and its allowed values are 0 or 1.

If an RD wants to utilize partial order among confederations as a mechanism to control its transitivity, BISs within that RD should include the HIERARCHICAL_RECORDING path attribute in their UPDATE PDUs, according to the following rules:

1. *Destination BIS in Disjoint RDC:*

   A BIS shall not include the HIERARCHICAL_RECORDING attribute in an UPDATE PDU sent to a BIS which can be reached only by exiting all of the RDCs of which the advertising BIS is a member.

2. *Destination BIS in Same, Nested, or Overlapping RDC:*

   a. If a given BIS wishes to advertise routeing information learned from an inbound UPDATE PDU whose HIERARCHICAL RECORDING attribute is equal to 1, or if it is the originator of the information to be advertised in an UPDATE PDU, it may advertise the routeing information to BISs in any adjacent RD:

1) If it is necessary to enter a confederation in order to reach the destination BIS, then the advertising BIS shall set the HIERARCHICAL RECORDING attribute of its outbound UPDATE PDU to 0.

2) If the destination BIS can be reached without entering a confederation, the value of the HIERARCHICAL RECORDING attribute in the outbound UPDATE PDU shall be 1.

b. If a given BIS receives an inbound UPDATE PDU whose HIERARCHICAL RECORDING attribute is equal to 0, then it may propagate the routeing information only to those BISs that can be reached without exiting any confederation to which the advertising BIS belongs. The value of the HIERARCHICAL RECORDING attribute in the outbound UPDATE PDU shall be set to 0.

Complementary Operation with Distribution Lists

Hierarchical-based rules can be complemented with rules that govern propagation of the routeing information in presence of the DIST_LIST_INCL and DIST_LIST_EXCL path attributes. In certain cases, the constraints on propagation of UPDATE PDUs which are imposed by the HIERARCHICAL RECORDING attribute may not be sufficiently broad to support all of the policies that a given RD wishes to implement. Therefore, an RD is free to include the DIST_LIST_INCL(EXCL) attribute in its UPDATE PDUs as well. In such cases, the contents of the DIST_LIST_EXCL (or DIST_LIST_EXCL) shall take precedence over the propagation rules based on the HIERARCHICAL RECORDING attribute.

When aggregating routes (see Section 7.12.9), if at least one of the routes to be aggregated contains the Hierarchical Recording path attribute, then the aggregated route contains the Hierarchical Recording path attribute as well (with the value of the aggregated route). If several routes to be aggregated contain the Hierarchical Recording path attribute with different values, the value of the Hierarchical Recording attribute of the aggregated route is set to 0.

When announcing unfeasible routes, the Hierarchical Recording path attribute may be omitted. However, propagation of the unfeasible routes is governed by the rules associated with the corresponding feasible routes. Presence of the Hierarchical Recording path attribute in the unfeasible route is ignored.

Attribute Encoding:

Since the HIERARCHICAL RECORDING attribute is now a single 1-octet field, there is no need to maintain the semantics of either a ″level″ or an ″Up/Down Flag″.

Transitivity Constraints

The rules listed above will support several simple transitivity policies, for example:

- A BIS may freely propagate an UPDATE PDU to any adjacent RD that can be reached without exiting any confederation to which the advertising BIS belongs
- Systems located in a region where confederations overlap cannot be used to propagate UPDATE PDUs between systems located in the non-overlapped regions
- RDs that form a confederation can not be used as a transit for RDs outside the confederation.

Figure Figure 6 on page 24 will be used to illustrate the effect of the attribute usage rules. This figure shows 12 routeing domains which have been organized into 7 confederations, 6 of which are nested inside RDC-1. It also shows the links between the routeing domains.

Consider the following examples:

- Consider an UPDATE PDU that is originated by RD-A and travels to RD-C. The rules permit RD-A to advertise the update to RD-C, and since RD-C can be reached without entering a new RDC, the HIERARCHICAL RECORDING attribute remain equal to 1. Routeing information that corresponds to traversing a set of nested confederations from the inside out is permitted.
- Consider an UPDATE PDU that is originated by RD-G and travels along a path <RD-G, RD-F>. RD-G is permitted to advertise to RD-F, and it sets the HIERARCHICAL RECORDING attribute to a value of 0, because RDC-6 will be entered. Routeing information that corresponds to traversing a set of nested confederations from the outside in is permitted.
- Consider an UPDATE PDU originated by RD-B and travelling to RD-I along path <RD-B, RD-J, RD-I>. When RD-B advertises to RD-J, the HIERARCHICAL attribute will have a value of 1, because it was not necessary to enter any new RDC in going from B to J. When J subsequently advertises to I, it will set the HIERARCHICAL RECORDING attribute to 0, because RDC-3 and RDC-7 are entered. Routing information flow that corresponds to traversing a set of nested confederations from the inside out, and then traversing a different set of nested confederations from the outside in is permitted.
- Consider a potential UPDATE PDU originated by RD-J. It cannot travel along the path <RD-J, RD-I, RD-H>. The UPDATE PDU that arrives at I will have a HIERARCHICAL RECORDING attribute with value 0; therefore this update can not be advertised to H since it would be necessary to exit RDC-7 in order to reach RD-H. Thus, the proposed method provides a simple mechanism mechanism to control the transitivity of the overlap: that is, it allows the RDCs that form the overlap to constrain their transitivity in such a way that traffic between the non-overlapping parts of the hierarchy could not traverse the region of the overlap.
- RD-D can advertise any UPDATE PDU to RD-E and RD-F regardless of who originated it or what the value of the HIERARCHICAL RECORDING attribute was. This is so because both of these destinations can be reached without exiting RDC-6. Note also that the value of the attribute will remain unchanged because it is not necessary to enter a confederation anywhere along this path.
- It would not be permitted to advertise an UPDATE PDU along a path <RD-C, RD-A, RD-B, RD-D>. The path segment from C to A is allowed, and A will receive an UPDATE PDU with a HIERARCHICAL RECORDING attribute of 0, because it was necessary to enter confederation RDC-5. Flow from A to B is permitted also. However, flow from B to D is not allowed because the incoming value of the HIERARCHICAL RECORDING attribute was 0, and therefore the UPDATE PDU can not exit RDC-5.

Appendix C. The NEXT_HOP Attribute

The NEXT_HOP attribute as it is defined in SC6 N6387 (IDRP) is a well-known mandatory attribute whose value is the NET of the BIS that originates the UPDATE PDU in which the attribute is contained (see clause 7.11.3 of SC6 N6387). Since the NET is either known at the time the BIS-BIS connection is established or can be ascertained from the NUNITDATA.INDICATION primitive, the attribute is redundant, and should not be retained in IDRP in its present form.

It is proposed that the NEXT_HOP attribute be changed to permit IDRP to support an optional route server function which allows a given BIS to advertise a BIS other than itself in this field. Secondly, the redefined NEXT_HOP attribute should allow explicit listing of SNPAs.

Rationale for Route Servers

Large public data networks or broadcast subnetworks may contain hundreds or even thousands of BISs, each of which may want routeing information from any of the others. However, the number of connections among a set of fully interconnected BISs may become prohibitive. Thus, it may be more efficient to provide a small number of ″IDRP route servers″. A BIS can establish an IDRP connection with an IDRP route server, which will then re-distribute the routeing information which it has learned from other BISs or other route servers. This implies that a route server must be capable of advertising a next-hop other than itself. In all other respects, it functions as a conventional BIS.

Typical Application of NEXT_HOP

The route server function applies to systems that are located on transitive fully connected subnetworks. We say that a subnetwork is transitive with respect to system reachability if all of the following are true:

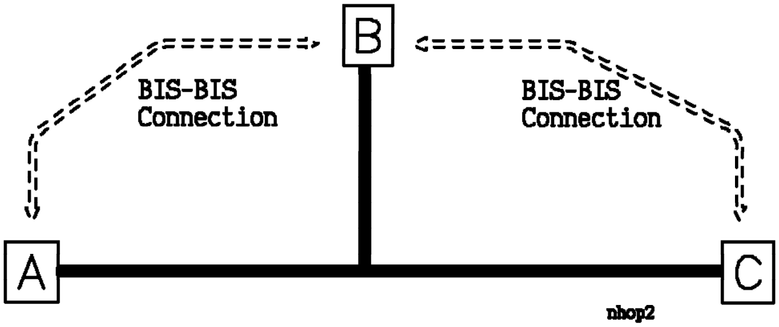
1. Systems A, B, and C are all attached to the same subnetwork,
2. When A can reach B directly, and B can reach C directly, it follows that A can reach C directly.

Verification of direct reachability can be accomplished by means outside of IDRP. For example, systems located on a common subnetwork could use an ES-IS protocol (such as IS 9542 or IS 10030) to ascertain if there is direct reachability between them. Examples of such media are IEEE 802.2, SMDS, and X.25.

Route Server Example

Consider three BISs attached to a common subnetwork that has the transitivity property. Assume that A has an IDRP connection with B, and B has an IDRP connection with C, but A and C do not have an IDRP connection with each other. Let us call BIS-C the source BIS, BIS-B the first recipient, and BIS-A the subsequent recipient of the UPDATE PDUs. Clearly, IDRP in its current definition allows the first recipient (B) to receive an UPDATE PDU with routeing information from the source (C), and then advertise it in its own UPDATE PDU to a subsequent recipient (A), subject to B′s routeing policies and any propagation constraints that the source included in the path attributes of its UPDATE PDU.

In SC6 N6387, if the source (C) informs B of a route to a destination X, and the first recipient (B) announces the same route to a subsequent recipient (A), the NPDUs from A destined to X will first be sent to B, and then B will forward them to C (subject of course to B′s own policies and to any constraints contained in the path attributes of C′s original UPDATE PDU).



Under the new usage rules for NEXT_HOP, the source (C) would inform B of a route to a destination X, indicating its own NET and an associated SNPA in the NEXT_HOP field of its UPDATE PDU. The first recipient (B) may then, at its option, elect to advertise the route to the subsequent BIS (A), but B would list the NET and SNPA of the source (C) in the UPDATE PDU that it sends to A. As a result of this advertisement, BIS-A would then send NPDUs destined for X directly to BIS-C , bypassing BIS-B.

Clearly, there are no compelling reasons why BIS-A should not send NPDUs directly to BIS-C, since this would not conflict with any path attributes that C had advertised in its UPDATE PDU. Sending NPDUs directly to C will avoid an extra hop at B, thus conserving network resources and eliminating the need for BIS-A and BIS-C to establish a BIS-BIS connection between themselves.

One might argue that if A is going to send NPDUs directly to C, then A and C should establish an IDRP connection between themselves. This may be feasible when the number of BISs involved is not large. However, in a public data network or a large broadcast subnetwork, this may not be practical due to the large number of BIS-BIS connections that would be needed if each BIS were required to have a BIS-BIS connection to each other BIS. (Note, for example, that DIS 10589 uses the concept of a ″LAN Designated IS″, or ″pseudonode″, to minimize the amount of routeing traffic that flows on a LAN.)

In some public data networks, it may be desirable to allow one IDRP server to advertise routes to another IDRP server, which in turn advertises it to yet another IDRP server, etc. This would be useful, for example, so that all BISs in a public data network would not have to converge on a single server. As long as all the IDRP servers are on a common subnetwork and are directly reachable from one another, the rules given below do not preclude this sort of operation.

Thus, both performance (elimination of an intermediate hop) and efficient use of resource (minimum BIS-BIS connections) offer compelling reasons to allow a BIS to advertise the NET of some other BIS in its UPDATE PDUs. When the NEXT_HOP attribute is used to provide this sort of redirection capability, it is necessary that all three BISs are located on a common full-duplex subnetwork, and that they are all directly reachable by one another.

Rationale for Advertising Multiple SNPAs

It is also suggested that the NEXT_HOP field should allow a BIS to explicitly list its SNPA(s). Once the SNPAs are listed explicitly, then it follows that all other path attributes advertised in an UPDATE PDU apply to them. This allows a BIS to inform its neighbors about which SNPAs it prefers its neighbors to send inbound traffic on.

Example Usage of Multiple SNPA Advertisement

For example, a given BIS may wish to receive inbound NPDUs over multiple SNPAs, perhaps as a form of load balancing. When multiple SNPAs are carried in a single UPDATE PDU, they are all considered to be equally preferable. In this case, a neighbor BIS may decide to send traffic to each of them on a round-robin basis.

However, if all its SNPAs are not equivalent, then the BIS may choose to advertise several UPDATE PDUs, where each lists a different set of equivalent SNPAs. For example, by using different values of the MULTI-EXIT_DISC attribute in each UPDATE PDU, the advertising system could also express its preference about which SNPAs are most desirable. A neighbor BIS would then have this information available to its Decision Process, and could factor the advertising BIS′s preferences into its own local decision-making process.

Thus, by allowing explicit advertisement of SNPAs, performance improvements can be obtained, since a preference among a set of SNPAs can now be expressed.

Changes to SC6 N6387

The proposed rules for the NEXT_HOP attribute accomplish two purposes:

1. They allow a BIS to unambiguously identify the preferred SNPA(s) to be used for inbound traffic.
2. They allow the source of an UPDATE PDU to specify whether or not the first recipient is allowed to advertise the source′s SNPA as part of its own UPDATE PDU.

The following changes should be made in SC6 N6387:

1. The encoding of the NEXT_HOP attribute (in clause 6.3, item ″c″, on page 14) should be changed as shown below. In addition to listing the NET of the next-hop BIS, this attribute may contain one or more SNPAs. If multiple SNPAs are listed, it is understood that they are all equivalent from the viewpoint of the BIS that advertises them.

   IDRP_Server_Allowed: This is a one octet field. The value X′FF′ indicates the recipient of this UPDATE PDU has the option of advertising (in its own outbound UPDATE PDUs) the NET and SNPA information learned from this UPDATE PDU. If the value is not X′FF′, then the recipient of this UPDATE PDU shall not advertise the NET and SNPA information learned from this UPDATE PDU.

   Length of NET: A 1 octet field whose value expresses the length of the ″NET of Next Hop″ field as measured in octets

   NET of Next Hop: A variable length field that contains the NET of the next BIS on the path to the destination system

   Number of SNPAs: A 1 octet field which contains the number of distinct SNPAs to be listed in the following fields

   Length of First SNPA: A 1 octet field whose value expresses the length of the ″First SNPA of Next Hop″ field as measured in semi-octets

   First SNPA of Next Hop: A variable length field that contains an SNPA of the BIS whose NET is contained in the ″NET of Next Hop″ field. The field length is an integral number of octets in length, namely the rounded-up integer value of one half the SNPA length expressed in semi-octets; if the SNPA has an contains an odd number of semi-octets, a value in this field will be padded with a trailing all-zero semi-octet.

   ⋮

:

Length of Last SNPA: A 1 octet field whose value expresses the length of the ″Last SNPA of Next Hop″ field as measured in semi-octets

Last SNPA of Next Hop: A variable length field that contains an SNPA of the BIS whose NET is contained in the ″NET of Next Hop″ field.

2. New usage rules should replace those currently listed in clause 7.11.3, as follows:

   NEXT_HOP is a well-known discretionary attribute that must be recognized upon receipt by all BISs.

   1. A BIS may choose not to include the NEXT_HOP attribute in its UPDATE PDU. In this case the SNPA must be learned from other means: for example, the value of the NET can be learned from the NUNITDATA.INDICATION, and IS 9542 can be used to associate an SNPA with that NET. When the NEXT_HOP field is not present, then it is understood that the BIS that advertises the UPDATE PDU is also the next-hop BIS.

   2. A BIS may choose to list its own NET and its own SNPAs in the appropriate fields of the NEXT_HOP attribute. It may set the value of the ″IDRP_Server_Allowed″ field in accordance with its local policies.

   3. A first recipient BIS that receives a route from a source BIS may advertise that route to a third subsequent recipient BIS. The first recipient BIS may list (in the NEXT_HOP field of its UPDATE PDU being sent to a subsequent recipient BIS) the NET of the source BIS and its associated SNPA (as contained in the UPDATE PDU received from the source BIS) when all of the following conditions are satisfied:

      a. The ″IDRP_Server_Allowed″ field of the inbound UPDATE PDU was equal to X′FF′.

      b. All three BISs (source, first recipient, and subsequent recipient) are located on a common subnetwork which is full-duplex.

      c. The subnetwork has the transitivity property with respect to reachability of all three BISs.

      d. The first recipient and subsequent recipient are located in different routeing domains.

      e. Advertisement of this route to the subsequent recipient BIS does not conflict with any of the path attributes that were contained in the UPDATE PDU from the source BIS.

      In this case, the advertising BIS may set the value of the ″IDRP_SERVER_Allowed″ field in the outbound UPDATE PDU in accordance with its own local policies.

   A note should also be added to make it clear that the rules outlined above do not remove the requirement that there must be an IDRP connection between each pair of BISs located within a given routeing domain.

3. The text in clause 7.11.2, item ″b″ should be expanded to note that a BIS will always update the RD_PATH attribute with its own RDI regardless of the contents of the NEXT_HOP field. Since a BIS that receives the UPDATE PDU has enough information to determine if the NEXT_HOP BIS is located in the same RD as the BIS from whom the UPDATE PDU was received, it can factor this into its decision process when it determines its own degree of preference for the route.

Appendix D. Text of RFC 1186

The text of RFC 1186, *The MD4 Message Digest Algorithm*, is attached for reference.