

AS/NZS 4164:1994
ISO/IEC TR 10023:1992

Australian/New Zealand Standard

Information technology— Telecommunications and information exchange between systems—Formal description of AS 2911 in LOTOS

[ISO/IEC TR title: Information technology—Telecommunications and information exchange between systems—Formal description of ISO 8072 in LOTOS]



AS/NZS 4164:1994

This Joint Australian/New Zealand Standard was prepared by Joint Technical Committee IT/1, Information Systems—Interconnection. It was approved on behalf of the Council of Standards Australia on 12 October 1993 and on behalf of the Council of Standards New Zealand on 22 April 1993. It was published on 21 February 1994.

The following interests are represented on Committee IT/1:

Australian Association of Permanent Building Societies
 Australian Bankers Association
 Australian Bureau of Statistics
 Australian Chamber of Commerce and Industry
 Australian Committee of Directors and Principals
 Australian Computer Society
 Australian Computer Users Association
 Australian Information Industry Association
 Australian Telecommunications Users Group
 Australian Vice Chancellors Committee
 Department of Defence, Australia
 Department of Industry, Technology and Commerce, Australia
 Information Exchange Steering Committee, Australia
 Institute of Information and Communication Technologies, CSIRO, Australia
 Life Insurance Federation of Australia
 OPTUS Communications, Australia
 Telecom Australia

Review of Standards. To keep abreast of progress in industry, Joint Australian/New Zealand Standards are subject to periodic review and are kept up to date by the issue of amendments or new editions as necessary. It is important therefore that Standards users ensure that they are in possession of the latest edition, and any amendments thereto.

Full details of all Joint Standards and related publications will be found in the Standards Australia and Standards New Zealand Catalogue of Publications; this information is supplemented each month by the magazines 'The Australian Standard' and 'Standards New Zealand', which subscribing members receive, and which give details of new publications, new editions and amendments, and of withdrawn Standards.

Suggestions for improvements to Joint Standards, addressed to the head office of either Standards Australia or Standards New Zealand, are welcomed. Notification of any inaccuracy or ambiguity found in a Joint Australian/New Zealand Standard should be made without delay in order that the matter may be investigated and appropriate action taken.


Documents

Document Sharing Hub

AS/NZS 4164:1994

Australian/New Zealand Standard

Information technology— Telecommunications and information exchange between systems—Formal description of AS 2911 in LOTOS

First published as Joint Standard AS/NZS 4164:1994.

PUBLISHED JOINTLY BY:

STANDARDS AUSTRALIA
1 The Crescent,
Homebush NSW 2140 Australia

STANDARDS NEW ZEALAND
6th Floor, Wellington Trade Centre,
181-187 Victoria Street,
Wellington 1 New Zealand

PREFACE

This Standard was prepared by the Joint Australia/New Zealand Standards Committee IT/1, on Information Systems—Interconnection. It is identical with, and has been reproduced from, ISO/IEC TR 10023:1992, *Information technology—Telecommunications and information exchange between systems—Formal description of ISO 8072 in LOTOS*.

This Joint Standard is one of a series of Open Systems Interconnection (OSI) Standards which are currently under development. Since OSI Standards are developmental, there may be some minor difficulties encountered in their implementation. For this reason, Standards Australia will be providing, through the OSI Help Desk, a service to coordinate and disseminate information concerning difficulties which are identified in using this Joint Standard.

Under arrangements made between Standards Australia/Standards New Zealand and the international Standards bodies, ISO and IEC, as well as certain other Standards organizations, users of this Joint Standard are advised of the following:

- (a) In Australia, copyright is vested in Standards Australia. In New Zealand, copyright is vested in Standards New Zealand.
- (b) The number of this Joint Standard is not reproduced on each page; its identity is shown only on the cover and title pages.

For the purpose of this Joint Standard, the ISO/IEC text should be modified as follows:

- (i) *Terminology* The words ‘Australian Standard’, ‘New Zealand Standard’ or ‘Joint Australian/New Zealand Standard’ should replace the words ‘International Standard’ wherever they appear.
- (ii) *References* The references to International Standards should be replaced by references to the following Australian/New Zealand Standards:

<i>Reference to International Standard or other Publication</i>		<i>Australian/New Zealand Standard</i>	
ISO		AS	
7498	Information Processing Systems—Open Systems Interconnection— Basic Reference Model	2777	Information processing systems— Open Systems Interconnection— Basic reference model
8072	Information Processing Systems—Open Systems Interconnection—Transport service definition	2911	Information processing systems— Open Systems Interconnection— Transport service definition
8072/Add.1	Addendum 1: Connectionless-mode transmission	—	
8072/Cor.1	Technical Corrigendum 1	—	

ISO

8807 Information Processing —
Systems—Open Systems
Interconnection—LOTOS—A
formal description technique
based on the temporal
ordering of observational
behaviour

ISO/TR

8509 Information Processing 3620 Information processing systems—
Systems—Open Systems Open Systems Interconnection—
Interconnection—Service Service conventions
conventions

ISO/IEC/TR

10024 Information technology—
Telecommunications and
information exchange between
systems—Formal description
of ISO 8073 (Classes 0, 1, 2,
3) in LOTOS

AS/NZS

4165 Information technology—Tele-
communications and information
exchange between systems—Formal
description of AS/NZS 2912 (Classes
0, 1, 2, 3) in LOTOS

© Copyright – STANDARDS AUSTRALIA/STANDARDS NEW ZEALAND

Users of Standards are reminded that copyright subsists in all Standards Australia and Standards New Zealand publications and software. Except where the Copyright Act allows and except where provided for below no publications or software produced by Standards Australia or Standards New Zealand may be reproduced, stored in a retrieval system in any form or transmitted by any means without prior permission in writing from Standards Australia or Standards New Zealand. Permission may be conditional on an appropriate royalty payment. Australian requests for permission and information on commercial software royalties should be directed to the head office of Standards Australia. New Zealand requests should be directed to Standards New Zealand.

Up to 10 percent of the technical content pages of a Standard may be copied for use exclusively in-house by purchasers of the Standard without payment of a royalty or advice to Standards Australia or Standards New Zealand.

Inclusion of copyright material in computer software programs is also permitted without royalty payment provided such programs are used exclusively in-house by the creators of the programs.

Care should be taken to ensure that material used is from the current edition of the Standard and that it is updated whenever the Standard is amended or revised. The number and date of the Standard should therefore be clearly identified.

The use of material in print form or in computer software programs to be used commercially, with or without payment, or in commercial contracts is subject to the payment of a royalty. This policy may be varied by Standards Australia or Standards New Zealand at any time.

CONTENTS

	<i>Page</i>
1 Scope	1
2 Normative references	1
3 Definitions	1
4 Symbols and abbreviations	1
5 Conventions	1
6 Requirements	2
7 Introduction of the formal description	2
8 Interface data types	2
9 Global Constraints	17
10 Provision of a transport connection	18
11 Local constraints for a TCEP	19
12 End-to-end constraints for a single TC	21
13 Identification of transport connections	24
14 Acceptance of transport connections	25
15 Flow control by backpressure	26
16 Connectionless-mode transmission	26

Information technology—Telecommunications and information exchange between systems—Formal description of AS 2911 in LOTOS

1 Scope

This Technical Report describes the OSI Transport service, defined in ISO 8072, using the formal description technique LOTOS, which is defined in ISO 8807.

NOTE – The formal definitions of data types and processes presented in this Technical Report are applicable for use in formal descriptions, using LOTOS, of the OSI Transport and Session protocols.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this Technical Report. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Technical Report are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of the currently valid International Standards.

ISO 7498:1984, *Information Processing Systems – Open Systems Interconnection – Basic Reference Model*.

ISO 8072:1986, *Information Processing Systems – Open Systems Interconnection – Transport service definition*.

ISO 8072:1986/Addendum 1:1986, *Information Processing Systems - Open Systems Interconnection - Transport service definition - Addendum 1 - Connectionless-mode transmission*.

ISO 8072:1986/Technical Corrigendum 1:1991, *Information Processing Systems - Open Systems Interconnection - Transport service definition - Technical Corrigendum 1*.

ISO/TR 8509:1987, *Information Processing Systems – Open Systems Interconnection – Service conventions*

ISO 8807:1989, *Information Processing Systems – Open Systems Interconnection – LOTOS – A formal description technique based on the temporal ordering of observational behaviour*.

ISO/IEC TR 10024:1992, *Information technology – Telecommunications and information exchange between systems – Formal description of ISO 8073 (Classes 0,1,2,3) in LOTOS*

3 Definitions

For the purposes of this Technical Report the definitions given in ISO 8072 apply.

4 Symbols and abbreviations

This Technical Report uses the symbols defined in clause 6 (formal syntax) and Annex A (data type library) of ISO 8807.

The abbreviations contained in clause 4 of ISO 8072 are employed in this Technical Report. Usage of other symbols and abbreviations is explained upon their first occurrence.

5 Conventions

Informal explanations precede the formal definitions to which they refer, separated from them by a line identical to the first line below. Separation of formal definitions from subsequent informal explanations is indicated by a line identical to the second line below.

(*)

NOTE – This convention is consistent with the rules on delimitation of comments defined for LOTOS in ISO 8807. The formal text is set in Courier and the LOTOS keywords and operators are in boldface. Identifiers from the formal text are set in italics when used in the non-formal text.

The conventions defined in ISO/TR 8509 are adopted, but together with the following one: the term "Request" refers both to request and to response service primitives, and the term "Indication" refers both to indication and to confirm service primitives.

6 Requirements

This Technical Report complies with the requirements defined in clause 3 of ISO 8807 (see annex C for further details).

No conformity requirements are defined by this Technical Report.

7 Introduction of the formal description

The whole service boundary is formally represented by the single gate t . The event structure at t consists of a triple of values, resp. of sorts $TAddress$, $TCEI$, TSP (see clause 8). The first value identifies the TSAP where the interaction occurs. The second value identifies the TCEP, within that TSAP, where the interaction occurs. The third value is the Transport Service Primitive (TSP) executed in the interaction. Full account is taken of the multiplicity aspects of the service.

The behaviour of a never terminating service provider is described. The specification is not parameterized.

A constraint-oriented specification style is adopted since it is the most appropriate with respect to the definitive character of a service standard. The description is aimed at describing modalities of behaviour of the service provider in terms of the history of executed TSPs only, i.e. with no reference to nor suggestion of any internal structure of the provider itself.

The following separate constraints are found in the first decomposition of the specified behaviour (see clause 9):

- a) The service provider may offer a multiplicity of possibly concurrent connections (represented by process $TCConnections$, see clause 9), together with
- b) a means to distinguish among concurrent connections (represented by process $TCIdentification$, see clause 13), but under
- c) the possibility of internal non-determinacy in accepting to establish any additional connection, if at least one connection is being supported, (represented by process $TCAcceptance$, see clause 14), and
- d) the possibility of internal non-determinacy in accepting to transfer data (represented by process $TBackpressure$, see clause 15); by means of this, the effect of the flow control exerted by the receiving user is transferred back to the sending user.

The specification of the dynamic behaviour is preceded by that of the interface data types (see clause 8). Such definitions are to be shared with the formal descriptions that may interwork with the present one, viz. those of the transport and session protocols.

The order of presentation of the further definitions follows a desire of adherence to ISO 8072, which mainly addresses the provision of a single TC. The description of the requirements that are local to the provision of a single TC (presented in clauses 10, 11 and 12) precedes that of the global requirements respectively mentioned in b), c) and d) above.

The data types referred to in the *library* construct are imported from the LOTOS library of data types.

```
specification TransportService[t] : noexit
```

```
library Set, Element, OctetString, NatRepresentations, NaturalNumber, Boolean,
  FBoolean, DecNatRepr
```

```
endlib
```

```
(*
```

8 Interface data types

8.1 General

According to the representation of interactions at the TS boundary (see clause 7), the interface data types consist of three main definitions that respectively construct the sorts $TAddress$ (see 8.2), $TCEI$ (see 8.3), and TSP (see 8.4). The quality of service TSP parameter is defined in 8.5, and other parameters in 8.6. Auxiliary definitions are presented in 8.7.

8.2 Transport address

No structure of transport addresses is defined by ISO 8072. The following definition uses the definition of *GeneralIdentifier* (see 8.7) and allows to represent an infinite number of transport addresses.

```
type TransportAddress
```

```
is GeneralIdentifier renamedby
```

```
sortnames
```

```
  TAddress for Identifier
```

```
opnnames
```

```
  SomeTAddress for SomeIdentifier
```

```

    AnotherTAddress for AnotherIdentifier
endtype (* TransportAddress *)

```

8.3 Transport connection endpoint identifier

No structure of transport connection endpoint identifiers is defined by ISO 8072. The first definition below allows to represent an infinite number of them. The second definition presents TC endpoint identifiers that are global to the whole TS boundary: each of them is a pair *TAddress* x *TCEI* (see 8.7 for the generic definition *Pair* and *GeneralIdentifier*).

```

type TCEndpointIdentifier
is GeneralIdentifier renamedby
sortnames
    TCEI for Identifier
opnnames
    SomeTCEI for SomeIdentifier
    AnotherTCEI for AnotherIdentifier
endtype (* TCEndpointIdentifier *)

type TCEIdentification
is Pair actualizedby TransportAddress, TCEndpointIdentifier using
sortnames
    TAddress for Element
    TCEI for Element2
    Bool for FBool
    TId for Pair
opnnames
    TId for Pair
    TA for First
    TCEI for Second
endtype (* TCEIdentification *)

```

8.4 Transport service primitive

8.4.1 General

The TSP data type is presented starting with the basic construction of values of sort *TSP* (see below). This construction is a direct formulation of the definition given in table 3 of ISO 8072. The functions that yield TSP values are referred to as "TSP constructors". This definition imports the definitions that relate to TSP parameters (see 8.6).

NOTE – In some TSPs the *Userdata* parameter is an *OctetString* that has a fixed bound, as defined in ISO 8072. For technical convenience, this requirement is formally represented by a process constraint (See 11.1 and 11.4) instead of a type constraint.

In 8.4.2 a classification of TSPs is defined, that enables both to enrich the basic construction with further functions in a simple way (see 8.4.3) and to conservatively extend the data type in the formal description of the transport protocol.

```

type BasicTSP
is TransportAddress, TEXOption, TSQuality, OctetString, TDISReason, TsClQuality
sorts
    TSP
opns
    TCONreq, TCONind      : TAddress, TAddress, TEXOption, TQOS, OctetString -> TSP
    TCONresp, TCONconf    : TAddress, TEXOption, TQOS, OctetString -> TSP
    TDTreq, TDTind        : OctetString -> TSP
    TEXreq, TEXind        : OctetString -> TSP
    TDISreq               : OctetString -> TSP
    TDISind               : TDISReason, OctetString -> TSP
    TUDTreq, TUDTind      : TAddress, TAddress, ClQOS, OctetString -> TSP
endtype (* BasicTSP *)

```

8.4.2 Transport service primitive classification

8.4.2.1 Basic classification

A basic classification of TSPs is defined by means of *TSPSubsort*, which consists of a set of constants, each denoting a TSP name in correspondence with table 3 of ISO 8072.

The type *TSPBasicClassifiers* is a functional enrichment of the basic construction in 8.4.1, where

- the *Subsort* function yields the TSP name;
- boolean functions on TSPs, termed "TSP (subsort) recognizers", are defined according to the basic classification introduced by *TSPSubsort*.

NOTE – The auxiliary function *h* that maps TSP names to natural numbers is defined in order to simplify the specification of the boolean operations of equality on TSP names (as well as on TSPs, in 8.4.3.3). The definitions of *IsRequest*, *IsIndication* (on TSP names) and *IsTReq*, *IsTInd* (on TSPs) reflect the convention introduced in clause 5

```

type TSPSubsort
is NaturalNumber
sorts
  TSPSubsort
opns
  TCONNECTrequest, TCONNECTindication, TCONNECTresponse, TCONNECTconfirm,
  TDATArequest, TDATAindication, TEXDATArequest, TEXDATAindication,
  TDISCONNrequest, TDISCONNindication, TUDATArequest, TUDATAindication : -> TSPSubsort
  h
    : TSPSubsort -> Nat
  Even, Odd
    : Nat -> Bool
  IsRequest, IsIndication
    : TSPSubsort -> Bool
  _eq_, _ne_
    : TSPSubsort, TSPSubsort -> Bool
eqns
forall
  s, s1 : TSPSubsort, n : Nat
ofsort Nat
  h(TCONNECTrequest) = 0;
  h(TCONNECTindication) = Succ(h(TCONNECTrequest));
  h(TCONNECTresponse) = Succ(h(TCONNECTindication));
  h(TCONNECTconfirm) = Succ(h(TCONNECTresponse));
  h(TDATArequest) = Succ(h(TCONNECTconfirm));
  h(TDATAindication) = Succ(h(TDATArequest));
  h(TEXDATArequest) = Succ(h(TDATAindication));
  h(TEXDATAindication) = Succ(h(TEXDATArequest));
  h(TDISCONNrequest) = Succ(h(TEXDATAindication));
  h(TDISCONNindication) = Succ(h(TDISCONNrequest));
ofsort Bool
  Even(0) = true;
  Even(Succ(0)) = false;
  Even(Succ(Succ(n))) = Even(n);
  Odd(n) = not(Even(n));
  IsRequest(s) = Even(h(s));
  IsIndication(s) = Odd(h(s));
  s eq s1 = h(s) eq h(s1);
  s ne s1 = not(s eq s1);
endtype (* TSPSubsort *)

type TSPBasicClassifiers
is BasicTSP, TSPSubsort
opns
  Subsort
    : TSP -> TSPSubsort
  IstCON, IstCON1, IstCON2, IstTDT, IstTEX, IstTDIS, IstTCONreq, IstTCONind,
  IstTCONresp, IstTCONconf, IstTDTreq, IstTDTind, IstTEXreq, IstTEXind, IstTDISreq,
  IstTDISind, IstTUDT, IstTUDTreq, IstTUDTind, IstTReq, IstTInd : TSP -> Bool
eqns
forall
  a, a1, a2 : TAddress, x : TEXOption, q : TQOS, d : OctetString, r : TDISReason,
  t : TSP, clq : CLQOS
ofsort TSPSubsort
  Subsort(TCONreq(a1, a2, x, q, d)) = TCONNECTrequest;
  Subsort(TCONind(a1, a2, x, q, d)) = TCONNECTindication;
  Subsort(TCONresp(a, x, q, d)) = TCONNECTresponse;
  Subsort(TCONconf(a, x, q, d)) = TCONNECTconfirm;
  Subsort(TDTreq(d)) = TDATArequest;
  Subsort(TDTind(d)) = TDATAindication;
  Subsort(TEXreq(d)) = TEXDATArequest;
  Subsort(TEXind(d)) = TEXDATAindication;
  Subsort(TDISreq(d)) = TDISCONNrequest;

```

```

Subsort(TDISind(r, d)) = TDISCONNindication;
Subsort(TUDTreq(a1, a2, clq, d)) = TUDATArequest;
Subsort(TUDTind(a1, a2, clq, d)) = TUDATAindication;
ofsort Bool
  IsTCOn(t) = IsTCOn1(t) or IsTCOn2(t);
  IsTCOn1(t) = IsTCOnreq(t) or IsTCOnind(t);
  IsTCOn2(t) = IsTCOnresp(t) or IsTCOnconf(t);
  IsTDT(t) = IsTDTreq(t) or IsTDTind(t);
  IsTEX(t) = IsTEXreq(t) or IsTEXind(t);
  IsTDis(t) = IsTDisreq(t) or IsTDisind(t);
  IsTCOnreq(t) = Subsort(t) eq TCONNECTrequest;
  IsTCOnind(t) = Subsort(t) eq TCONNECTindication;
  IsTCOnresp(t) = Subsort(t) eq TCONNECTresponse;
  IsTCOnconf(t) = Subsort(t) eq TCONNECTconfirm;
  IsTDTreq(t) = Subsort(t) eq TDATArequest;
  IsTDTind(t) = Subsort(t) eq TDATAindication;
  IsTEXreq(t) = Subsort(t) eq TEXDATArequest;
  IsTEXind(t) = Subsort(t) eq TEXDATAindication;
  IsTDisreq(t) = Subsort(t) eq TDISCONNrequest;
  IsTDisind(t) = Subsort(t) eq TDISCONNindication;
  IsTUDT(t) = IsTUDTreq(t) or IsTUDTind(t);
  IsTUDTreq(t) = Subsort(t) eq TUDATArequest;
  IsTUDTind(t) = Subsort(t) eq TUDATAindication;
  IsTReq(t) = IsRequest(Subsort(t));
  IsTInd(t) = IsIndication(Subsort(t));
endtype (* TSPBasicClassifiers *)

```

8.4.2.2 Auxiliary classification

TDATAAtomSubsort introduces a further classification of the atomic constituents of T-DATA primitives, viz. user data octets and TSDU terminators. However, in this specification these constituents are not specified.

NOTE – The only reason for presenting *TDATAAtomSubsort* in this description is that in this way the *TransportServicePrimitive* data type admits a conservative extension, where a more refined, non-atomic representation of T-DATA primitives can be introduced. Such an extension is needed in the formal description of the transport protocol for a correct formulation of the requirements relating to segmenting and flow control.

The boolean function *Terminates*, defined in *TSPClassifiers*, relates the atomic execution of T-DATA primitives, which characterizes the service definition, to the non-atomic one found in the formal description of the protocol.

```

type TDATAAtomSubsort
is FourTuple renamedby
sortnames
  TDTASubsort for Tuple
opnames
  TDATAOCTrequest for TheOne
  TDATAOCTindication for TheOther
  TEOTrequest for TheThird
  TEOTindication for TheFourth
endtype (* TDATAAtomSubsort *)

type TSPClassifiers
is TSPBasicClassifiers, TDATAAtomSubsort
opns
  _Terminates_ : TDTASubsort, TSP -> Bool
eqns
forall
  d : OctetString, s : TDTASubsort, t : TSP
ofsort Bool
  TEOTrequest Terminates TDTreq(d) = true;
  TEOTindication Terminates TDTind(d) = true;
  TEOTrequest Terminates TDTind(d) = false;
  TEOTindication Terminates TDTreq(d) = false;
  TDATAOCTrequest Terminates t = false;
  TDATAOCTindication Terminates t = false;
  not(IsTDT(t)) => s Terminates t = false;
endtype (* TSPClassifiers *)

```