

Implementation of ISO/IEC 9075 : 1992

**Information technology —
Database languages — SQL**

National foreword

This British Standard reproduces verbatim ISO/IEC 9075 : 1992 and implements it as the UK national standard. This standard supersedes BS 6964 : 1990, which is withdrawn.

This British Standard is published under the direction of the Information Systems Technology Standards Policy Committee whose Technical Committee IST/21 has the responsibility to:

- aid enquirers to understand the text;
- present to the responsible international committee any enquiries on interpretation, or proposals for change, and keep UK interests informed;
- monitor related international and European developments and promulgate them in the UK.

NOTE. International and European Standards, as well as overseas standards, are available from BSI Sales Department, BSI, Linford Wood, Milton Keynes, MK14 6LE.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

**Implementation of Technical Corrigendum 1
published and effective from 15 June 1995
to BS ISO/IEC 9075 : 1992**

**Information technology —
Database languages — SQL**

National foreword

In line 1, after 'ISO/IEC 9075 : 1992' insert, 'including Technical Corrigendum 1'.

AMD 8627/June 1995

ISO text

Amend the text of the international standard in accordance with the details given on the attached page(s).

AMD 8627/June 1995



INTERNATIONAL STANDARD ISO/IEC 9075:1992
TECHNICAL CORRIGENDUM 1

Published 1994-11-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Database languages — SQL

TECHNICAL CORRIGENDUM 1

Technologies de l'information — Langages de bases de données — SQL

RECTIFICATIF TECHNIQUE 1

Technical corrigendum 1 to International Standard ISO/IEC 9075:1992 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Statement of purpose for rationale:

A statement indicating the rationale for each change to ISO/IEC 9075:1992 is included. This is to inform the users of that standard as to the reason why it was judged necessary to change the original wording. In many cases the reason is editorial or to clarify the wording; in some cases it is to correct an error or an omission in the original wording.

Notes on rule numbering:

This Corrigendum introduces some new Syntax, Access, General and Leveling Rules. The new Rules in this Corrigendum have been numbered as follows:

Rules inserted between, for example, Rules 7) and 8) (in ISO/IEC 9075:1992) are numbered 7.1), 7.2), etc. [or 7)a.1), 7)a.2), etc.]. Those inserted before Rule 1) are numbered 0.1), 0.2), etc.

UDC 681.3.04:519.76:800.92

Ref. No. ISO/IEC 9075:1992/Cor.1:1994(E)

Descriptors: data processing, text processing, factual data bases, information interchange, documentation, documents, logical structure, artificial languages, database languages, SQL.

© ISO/IEC 1994

Printed in Switzerland

Introduction

Rationale: In the list of significant new features, the wording incorrectly implies that all the examples listed in item 10) are referential integrity facilities.

On page xiv, in Significant new feature 10), replace "referential integrity" with "integrity".

2 Normative references

Rationale: Editorial

Add the following reference after the reference to "ISO 8601:1988":

-- ISO 8649:1988, *Information Processing Systems--Open Systems Interconnection--Service Definition for the Association Control Service Element*.

Add the following reference after the reference to ISO/IEC 8824:1990:

--ISO/IEC 9579-1:1993, *Information Technology--Open Systems Interconnection--Remote Database Access, Part 1: Generic Model, Service, and Protocol*.

Add the following reference after the reference to ISO/IEC 9899:

--ISO/IEC 10026-2¹, *Information Technology--Open Systems Interconnection--Distributed Transaction Processing--Part 2: Service Definition*.

3.1.3 Definitions provided in this International Standard

Rationale: Remove an unnecessary definition.

Delete Item x)

3.3.4.3 Terms denoting rule requirements

Rationale: The following unifies the SQLSTATE returned for the different ways of invoking an SQL statement.

In the first and second paragraphs, replace "syntax error or access rule violation (if this situation ... SQL statement)" with "syntax error or access rule violation".

4.2.1 Character strings and collating sequences

Rationale: The following changes make the definitions of character set and collation descriptors more precise.

Replace the text on page 17 that occurs after the first paragraph with the following:

A character set is described by a character set descriptor. A character set descriptor includes:

- the name of the character set or character repertoire,
- if the character set is a character repertoire, then the name of the form-of-use,
- an indication of what characters are in the character set, and

- whether or not the character set uses the DEFAULT collation for its character repertoire, and,
- if the character set does not utilize the DEFAULT collation for its character repertoire, then the <translation name> contained in the character set's <translation collation>, if any, the <collation name> contained in the character set's <collate clause> or <limited collation definition>, if any, and, whether or not DESC was specified in the reference to the collation

For every character set, there is at least one collation. A collation is described by a collation descriptor. A collation descriptor includes:

- the name of the collation,
- the name of the character repertoire on which the collation operates,
- whether the collation has the NO PAD or the PAD SPACE attribute, and
- whether or not this collation utilizes the DEFAULT collation for its character repertoire,
- if the character set does not utilize the DEFAULT collation for its character repertoire, then the <translation name> contained in the collation's <translation collation>, if any, the <collation name> contained in the collation's <collation source>, if any, and whether or not DESC was specified in the definition of the collation.

4.4 Numbers

Rationale: Clarification.

Add the following sentence immediately before the heading of Subclause 4.4.1 Characteristics of Numbers:

A value described by a numeric data type descriptor is always signed.

4.5 Datetimes and intervals

Rationale: Clarification.

Add the following sentence before the paragraph starting "Every datetime ...":

A value described by an interval data type descriptor is always signed.

4.5.2 Intervals

Rationale: Editorial

In Table 7, replace "<interval leading field precision" with "<interval leading field precision>" (two occurrences).

4.8 Columns

Rationale: Editorial

In the penultimate paragraph on page 28, replace "<row value constructor expression>" with "<row value constructor element>".

4.9 Tables

Rationale: There is no named derived table other than a viewed table.

After the paragraph that begins with "A derived table descriptor describes a derived table.", delete the first item ("– if the table is named, then the name of the table;").

4.10.2 Table constraints

Rationale: Editorial

In the **Note**, replace "<match option>" with "<match type>".

In the sentence that begins with "A <referential constraint> is satisfied", replace "<match option>" with "<match type>".

4.24 SQL dynamic statements

Rationale: Editorial

In the fourth paragraph, replace "<target specification>s" with "<simple value specification>s".

4.26 Privileges

Rationale: Editorial

In the fourth paragraph on page 52, replace "<module authorization identifier> is" with "<schema authorization identifier> is".

Rationale: Provide missing rules that cover the acquisition of the necessary privileges to acquire the WITH GRANT OPTION on views through a grant to PUBLIC.

Add the following before the antipenultimate paragraph of this Subclause:

The phrase *user privileges* refers to the set of privileges defined by the privilege descriptors whose grantee is either the identified <authorization identifier> or PUBLIC.

4.28 SQL-transactions

Rationale: Clarification

In the paragraph that begins "In some environments (e.g., remote database access)", replace all occurrences of "SQL-environment" with "SQL-implementation".

4.29 SQL-connections

Rationale: Clarification

In the last sentence of the penultimate paragraph, replace "SQL-environment" with "SQL-implementation".

4.31 Client-server operation

Rationale: Clarification

Replace the first sentence with the following:

As perceived by an SQL-agent, an SQL-implementation consists of one or more SQL-servers and one SQL-client through which SQL-connections can be made to them.

5.2 <token> and <separator>

Rationale: A <regular identifier> shall not contain any <quote> or <double quote>. Thus, a <delimited identifier> with a <delimited identifier body> containing a <quote> or <double quote> is not equivalent to any <regular identifier>.

In Syntax Rule 13), delete the expression "(with all occurrences of <quote> replaced by <quote symbol> and all occurrences of <doublequote symbol> replaced by <double quote>)".

Rationale: A <character representation> does not appear in a <regular identifier> or in a <delimited identifier body>.

Replace Leveling Rule 2)a) with the following:

- a) The number of <underscore>s plus the number of <identifier part>s contained in a <regular identifier> shall be less than 18.

Insert the following Leveling Rule 2)a.1) between Leveling Rules 2)a) and 2)b):

- a.1) The <delimited identifier body> of a <delimited identifier> shall not comprise more than 18 <delimited identifier part>s.

5.3 <literal>

Rationale: Ellipses were placed both in the definition of <separator> and in the usage of <separator> in several types of string literals. This leads to a small ambiguity when two separator characters follow one another. The following changes remove the ellipses in the latter of these two places:

In Format, in each of the productions for <character string literal>, <national character string literal>, <bit string literal>, <hex string literal>, replace "<separator> ..." with "<separator>".

In Syntax Rules 1) - 3) replace "<separator> ..." with "<separator>".

5.4 Names and identifiers

Rationale: Editorial

In Syntax Rule 1), replace "identified" with "indicated".

6.1 <data type>

Rationale: Clarify the error condition that should be returned if a datetime overflow occurs.

Delete General Rule 6).

6.2 <value specification> and <target specification>

Rationale: Identify possible upward incompatibility.

Add the following at the end of Syntax Rule 3):

Note: In an environment where the SQL-implementation conforms to Entry SQL, conforming SQL language that contains either:

- a) a specified or implied <comparison predicate> that compares the <value specification> USER with a <value specification> other than USER, or
- b) a specified or implied assignment in which the "value" (as defined in Subclause 9.2 Store assignment) contains the <value specification> USER

will become non-conforming in an environment where the SQL-implementation conforms to Intermediate SQL or Full SQL, unless the character repertoire of the implementation-defined character set in that environment is identical to the character repertoire of SQL_TEXT.

6.3 <table reference>

Rationale: Editorial

In Syntax Rule 2)a), replace "with no intervening" with "without an intervening".

In Syntax Rule 2)b), replace "with no intervening" with "without an intervening".

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 9):

- 9) Let *T* be the table identified by the <table name> immediately contained in <table reference>. If the <table reference> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*. If the <table reference> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*, or *S* shall include a <schema element> that creates the descriptor of *T*.

Rationale: Editorial

Delete the first sentence of Access Rule 1).

6.8 <datetime value function>

Rationale: The following change clarifies that <datetime value function> is effectively evaluated only once per SQL statement.

Replace General Rule 3) with the following:

- 3) If an SQL-statement causes the evaluation of one or more <datetime value function>s, then all such evaluations are effectively performed simultaneously. The time of evaluation of the <datetime value function> during the execution of the SQL-statement is implementation-dependent.

6.10 <cast specification>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 9):

- 9) If <domain name> is specified, then let *D* be the domain identified by the <domain name>. If the <cast specification> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <domain name> shall include the descriptor of *D*. If the <cast specification> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <domain name> shall include the descriptor of *D*, or *S* shall include a <schema element> that creates the descriptor of *D*.

Rationale: Clarify the error condition that should be returned if a datetime overflow occurs.

Replace General Rule 9)a)ii) with the following:

ii) Otherwise,

- 1) If a <datetime value> does not conform to the natural rules for intervals according to the Gregorian calendar, an exception condition is raised: *data exception - invalid interval format*.
- 2) Otherwise, an exception condition is raised: *data exception - invalid character value for cast*.

Replace General Rule 10)a)ii) with the following:

ii) Otherwise,

- 1) If a <datetime value> does not conform to the natural rules for dates or times according to the Gregorian calendar, an exception condition is raised: *data exception - invalid datetime format*.
- 2) Otherwise, an exception condition is raised: *data exception - invalid character value for cast*.

Replace General Rule 11)a)ii) with the following:

ii) Otherwise,

- 1) If a <datetime value> does not conform to the natural rules for dates or times according to the Gregorian calendar, an exception condition is raised: *data exception - invalid datetime format*.
- 2) Otherwise, an exception condition is raised: *data exception - invalid character value for cast*.

Replace General Rule 12)b)ii) with the following:

ii) Otherwise,

- 1) If a <datetime value> does not conform to the natural rules for dates or times according to the Gregorian calendar, an exception condition is raised: *data exception - invalid datetime format*.
- 2) Otherwise, an exception condition is raised: *data exception - invalid character value for cast*.

6.12 <numeric value expression>

Rationale: Editorial

In General Rule 2), replace "then the value of" with "then the result of".

6.14 <datetime value expression>

Rationale: The expression "X AT TIME ZONE Y - Z" may be parsed as either "(X AT TIME ZONE Y) -

ISO/IEC 9075:1992/Cor.1:1994(E)

Z" or "X AT TIME ZONE (Y - Z)". The following clarifies that it should be the former.

Replace the Format for <time zone specifier> with the following:

```
<time zone specifier>::=
    LOCAL
    | TIME ZONE <interval primary>
```

Rationale: Editorial

In Syntax Rule 4), replace "<interval value expression>" with "<interval primary>".

Rationale: If <time zone> is implied, then <time zone specifier> shall not be specified.

In General Rule 2), replace "specified or implied" with "specified".

Rationale: Editorial

In General Rule 2), replace "<interval value expression>" with "<interval primary>".

In General Rule 4)b), replace "<interval value expression>" with "<interval primary>".

7.1 <row value constructor>

Rationale: Editorial

In Leveling Rule 1)a), replace "<table value constructor>" with "<table value constructor> or an <overlaps predicate>".

7.4 <from clause>

Rationale: Editorial

In Syntax Rule 1)a), replace "with no intervening" with "without an intervening".

In Syntax Rule 1)b), replace "with no intervening" with "without an intervening".

In General Rule 1)a), replace "with no intervening" with "without an intervening".

In General Rule 1)b), replace "with no intervening" with "without an intervening".

Rationale: Editorial

In Leveling Rule 2)a), replace "<table name>" with "<table reference>".

7.9 <query specification>

Rationale: Editorial

In Syntax Rule 9)a), replace the two occurrences of "C" with "CN".

In Syntax Rule 9)b), replace "C" with "the <column name> of the column designated by the <column reference>".

Rationale: Editorial

In General Rule 1)a)ii)1), replace "the table is the result of the <query specification>" with "the result of the <query specification> is that table".

In General Rule 1)b)ii)1), replace "the table is the result of the <query specification>" with "the result of the <query specification> is that table".

7.10 <query expression>

Rationale: Editorial

Replace Syntax Rule 14) with the following:

14) The *simply underlying tables* of a <query expression> are the tables identified by those <table name>s, <query specification>s, and <derived table>s contained in the <query expression> without an intervening <derived table> or an intervening <join condition>.

7.11 <scalar subquery>, <row subquery>, and <table subquery>

Rationale: Editorial

Replace Leveling Rule 1) with the following:

1) The following restrictions apply for Intermediate SQL:

None.

8.3 <between predicate>

Rationale: Editorial

In Syntax Rule 2), replace "two" with "three".

8.7 <quantified comparison predicate>

Rationale: Editorial

In Syntax Rule 2), replace "columns" with "corresponding columns".

10.1 <interval qualifier>

Rationale: Editorial

Replace "Syntax rules" with "Syntax Rules".

10.4 <character set specification>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 4):

4) Let *C* be the <character set name> contained in the <character set specification>. If the <character set specification> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <character set name> shall include the descriptor of *C*. If the

ISO/IEC 9075:1992/Cor.1:1994(E)

<character set specification> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <character set name> shall include the descriptor of *C*, or *S* shall include a <schema element> that creates the descriptor of *C*.

10.5 <collate clause>

Rationale: Clarify that references to non-existing objects are only allowed within the same <schema definition>.

Add the following as Syntax Rule 1):

- 1) Let *C* be the <collation name> contained in the <collate clause>. If the <collate clause> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <collation name> shall include the descriptor of *C*. If the <collate clause> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <collation name> shall include the descriptor of *C*, or *S* shall include a <schema element> that creates the descriptor of *C*.

10.6 <constraint name definition> and <constraint attributes>

Rationale: Editorial

In Leveling Rule 2)a), replace "Intermediate" with "Entry".

11.1 <schema definition>

Rationale: Editorial

In Leveling Rule 2)a), replace "Intermediate" with "Entry".

11.2 <drop schema statement>

Rationale: Editorial

In General Rule 5), replace "collation definition" with "collating sequence".

11.4 <column definition>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 4.1), between Syntax Rules 4) and 5):

- 4.1) If the <column definition> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <domain name> shall include the descriptor of *D*. If the <column definition> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <domain name> shall include the descriptor of *D*, or *S* shall include a <schema element> that creates the descriptor of *D*.

Rationale: Editorial

In Access Rule 1), replace "*D*.." with "*D*".

Rationale: Editorial

In Leveling Rule 2)c), replace "Intermediate" with "Entry".

11.5 <default clause>

Rationale: Editorial

In Syntax Rule 1), replace "<alter domain definition>" with "<alter domain statement>".

Rationale: The following changes correct the fact that the current rules cause the value of, for example, CURRENT_USER to be evaluated at the time of the <table definition> or <column definition> and forever more used as the default value for the associated column in the table. The value of CURRENT_USER, etc., and the <datetime value function>s are being put directly into the column or domain descriptor at the time that the <default clause> is processed. It was intended that those things be evaluated at the time that the row was inserted into the table (or updated in the table).

Replace General Rules 1) and 2) with the following General Rules 1) through 3):

1) The default value inserted in the column descriptor, if the <default clause> is to apply to a column, or in the domain descriptor, if the <default clause> is to apply to a domain, is the <default option>. If the subject data type is bit string with fixed length, the <default clause> specifies a <bit string literal>, and the length of the <bit string literal> is less than the fixed length of the column, then a completion condition is raised: *warning--implicit zero-bit padding*.

2) The default value of a column is

Case:

- a) If the column descriptor of a column indicates a default value derived from a <default option>, then the value in the column descriptor;
- b) If the column descriptor includes a domain name that identifies a domain descriptor that includes a default value derived from a <default option>, then the value in the domain descriptor;
- c) Otherwise, the null value.

3) When a default value is required for a column in a row of a table, the default value for the column is derived from the default value in the column descriptor or domain descriptor for the column as follows:

- a) If the column or domain descriptor contains NULL, then the null value.
- b) If the column or domain descriptor contains a <literal>, then

Case:

- i) If the subject data type is numeric, then the numeric value of the <literal>.
- ii) If the subject data type is character string with variable length, then the value of the <literal>.
- iii) If the subject data type is character string with fixed length, then the value of the <literal>, extended as necessary on the right with <space>s to the length in characters of the subject data type.
- iv) If the subject data type is bit string with variable length, then the value of the <literal>.

- v) If the subject data type is bit string with fixed length, then the value of the <literal> extended as necessary on the right with 0-valued bits to the length of the subject data type.
- vi) If the subject data type is datetime or interval, then the value of the <literal>.
- c) If the column or domain descriptor specifies CURRENT_USER, SESSION_USER, or SYSTEM_USER, then

Case:

- i) If the subject data type is character string with variable length, then the value obtained by an evaluation of CURRENT_USER, SESSION_USER, or SYSTEM_USER at the time that the default value is required.
- ii) If the subject data type is character string with fixed length, then the value obtained by an evaluation of CURRENT_USER, SESSION_USER, or SYSTEM_USER at the time that the default value is required, extended as necessary on the right with <space>s to the length in characters of the subject data type.
- d) If the column or domain descriptor contains a <datetime value function>, then the value of an evaluation of the <datetime value function> at the time that the default value is required.

Rationale: The following helps define the action taken when a default value cannot be represented in the Information Schema:

Add the following as General Rule 4):

- 4) If the <default clause> is contained in an <schema statement>, and if the character representation of the <default option> cannot be represented in the Information Schema without truncation, then a completion indication is raised: *warning--default option too long for information schema*.

11.6 <table constraint definition>

Rationale: Editorial

In Leveling Rule 2)a), replace "Intermediate" with "Entry".

11.8 <referential constraint definition>

Rationale: Clarify that references to non-existing objects are only allowed within the same <schema definition>.

Add the following as Syntax Rule 10):

- 10) Let *T* be the referenced table. If the <referential constraint definition> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*. If the <referential constraint definition> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*, or *S* shall include a <schema element> that creates the descriptor of *T*.

11.11 <add column definition>

Rationale: Editorial

Add the following as General Rule 0.1):

- 0.1) Let *T* be the table identified by the <table name> immediately contained in the containing <alter table statement>.

11.18 <drop table statement>

Rationale: A <drop behavior> of RESTRICT shall not be allowed if there are referential constraints that reference the table being dropped, and shall not consider check constraints of the table being dropped.

Replace Syntax Rule 4) with the following:

- 4) If RESTRICT is specified, then *T* shall not be referenced in the <query expression> of any view descriptor, the <search condition> of any table check constraint descriptor of any table other than *T*, or the referenced table of any referential constraint descriptor of any table other than *T*.

Note: If CASCADE is specified, then such referencing objects will be dropped by the execution of the <revoke statement> specified in the General Rules of this Subclause.

11.19 <view definition>

Rationale: A predicate specified for LOCAL CHECK OPTION is tested only for rows that would pass all lower level predicates. This does not reflect the correct semantics for LOCAL.

Replace General Rule 9) with the following:

- 9) Let *V1* be an updatable view. Let *V2* be the simply underlying table of the simply underlying table of the <query expression> of *V1*. Let *S* be the set of *check conditions* of *V1* that is used for the validation of WITH CHECK OPTION. Each check condition of *S* is a *primary <search condition>* of a view. The primary <search condition> of a view is the <search condition> specified in the <where clause> of the <table expression> directly contained in the <query expression> of the view. If the <table expression> does not contain a <where clause>, then the view has an implicit primary <search condition> that is always true.

Case:

- a) If *V2* is a base table and the descriptor of *V1* does not include WITH CHECK OPTION, then *S* is empty.
- b) If *V2* is a base table and the descriptor of *V1* includes any form of WITH CHECK OPTION, then *S* consists of the primary <search condition> of *V1*.
- c) If *V2* is a view and the descriptor of *V1* does not include WITH CHECK OPTION, then *S* is identical to the set of check conditions of *V2*.
- d) If *V2* is a view and the descriptor of *V1* contains WITH CASCADED CHECK OPTION, then *S* consists of the primary <search condition> of *V1* and the primary <search condition> of each view that is a generally underlying table of *V1*.
- e) If *V2* is a view and the descriptor of *V1* contains WITH LOCAL CHECK OPTION, then *S* consists of the primary <search condition> of *V1* and the set of check conditions of *V2*.

Replace General Rule 11) with the following:

- 11) During an update operation on *VI*, every check condition in *S* is applied to every inserted or updated row. If any check condition is not true, then an exception condition is raised: *with check option violation*.

11.21 <domain definition>

Rationale: In General Rule 3), a domain constraint descriptor is referenced, but it has not created.

Insert the following General Rule 2.1) between General Rules 2) and 3):

For every <domain constraint> immediately contained in the <domain definition>:

A domain constraint descriptor is created that describes the domain constraint being defined. The domain constraint descriptor includes the <constraint name> contained in the explicit or implicit <constraint name definition>. The domain constraint descriptor includes an indication of whether the constraint is deferrable or not deferrable, and whether the initial constraint mode of the constraint is *deferred* or *immediate*. The domain constraint descriptor includes the <search condition> immediately contained in the <check constraint definition> immediately contained in the <domain constraint>.

Rationale: The following change specifies that when overriding the collation of a contained character set, a grantable USAGE privilege on that overriding collation is not needed to acquire a grantable USAGE privilege on the domain.

Replace General Rule 4) with the following:

- 4) A privilege descriptor is created that defines the USAGE privilege on this domain to the <authorization identifier> of the <schema> or <module> in which the <domain definition> appears. This privilege is grantable if and only if the applicable privileges include a grantable REFERENCES privilege for each <column reference> included in the domain descriptor and a grantable USAGE privilege for each <domain name>, <collation name>, <character set name>, and <translation name> contained in the <search condition> of any domain constraint descriptor included in the domain descriptor. The grantor of the privilege descriptor is set to the special grantor value “_SYSTEM”.

11.28 <character set definition>

Rationale: Editorial

In Syntax Rule 3), replace "character set descriptor" with "character set".

11.27 <drop domain statement>

Rationale: When a domain is destroyed it is necessary to destroy its domain constraint descriptor.

Replace General Rule 3) with the following:

- 3) The identified domain is destroyed by destroying its descriptor, its data type descriptor, and its domain constraint descriptors.

11.29 <drop character set statement>

Rationale: The following change adds missing restrictions to deal with the convention that absence of a <drop behavior> production in the BNF of a schema manipulation statement indicates that the respective schema manipulation statement cannot drop objects while other objects depend on their existence.

Replace Syntax Rule 3) with:

- 3) *C* shall not be referenced in the <query expression> of any view descriptor or in the <search condition> of any constraint descriptor, nor be included in any collation descriptor or translation descriptor or schema descriptor, nor be referenced in the <module character set specification> of any module, nor be included in any column's or any domain's data type descriptor.

11.30 <collation definition>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 11):

- 11) If <translation name> is specified and the <collation definition> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <translation name> shall include the descriptor of *T*. If the <collation definition> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <translation name> shall include the descriptor of *T*, or *S* shall include a <schema element> that creates the descriptor of *T*.

11.31 <drop collation statement>

Rationale: The following changes address the problems that string manipulations are specified on descriptors that are not defined to contain character strings, and that a convention is violated that absence of a <drop behavior> production implies that the Schema Manipulation Language fails if it causes other objects to be dropped. To achieve this, the defined contents of descriptors must be altered. Then, General Rules are reordered such that the effectively executed revoke statement happens after any domain descriptors that depend upon the collation being dropped have been altered to no longer have that dependency.

Replace the General Rules with the following:

- 1) For every collation descriptor *CD* that includes *CN*, *CD* is modified such that it does not include *CN*. If *CD* does not include any translation name, then *CD* is modified to indicate that it utilizes the DEFAULT collation for its character repertoire.
- 2) For every character set descriptor *CSD* that includes *CN*, *CSD* is modified such that it does not include *CN*. If *CSD* does not include any translation name, then *CSD* is modified to indicate that it utilizes the DEFAULT collation for its character repertoire.
- 3) For every column descriptor or domain descriptor *DD* that includes *CN*, *DD* is modified such that it does not contain *CN*.

Note: This causes the column or domain that *DD* is the descriptor for to revert to the default collation for its character set.

- 4) Let *A* be the current <authorization identifier>. The following <revoke statement> is effectively executed with a current <authorization identifier> of “_SYSTEM” and without further Access Rule checking:

REVOKE USAGE ON COLLATION *CN* FROM *A* CASCADE

- 5) The descriptor of *C* is destroyed.

11.32 <translation definition>

Rationale: Editorial

In Syntax Rule 5), replace "identify a" with "identify some".

11.34 <assertion definition>

Rationale: Editorial

Replace Syntax Rule 4) with the following:

The <search condition> shall not contain a <parameter specification>, a <variable specification>, or a <dynamic parameter specification>.

11.36 <grant statement>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 4):

- 4) If the <grant statement> is not contained in a <schema definition>, then the schema identified by the explicit or implicit qualifier of the <object name> shall include the descriptor of *O*. If the <grant statement> is contained in a <schema definition> *S*, then the schema identified by the explicit or implicit qualifier of the <object name> shall include the descriptor of *O*, or *S* shall include a <schema element> that creates the descriptor of *O*.

Rationale: Editorial

In General Rule 7), replace "<grantor>" with "grantor" and replace "<object>" with "object".

Rationale: Provide missing rules that cover the acquisition of the necessary privileges to acquire the WITH GRANT OPTION on views through a grant to PUBLIC.

In General Rule 7), replace "if *G* has been granted" with "if the user privileges of *G* contain the".

Rationale: The following changes addresses the fact that <grant statement> and <revoke statement> are inconsistent with <collation definition> with regard to their treatment of the grantability of USAGE privilege on collations. The first change makes grantability of USAGE privilege against a domain more consistent with that of the grantability of USAGE privilege against a column, while the second change makes a collation USAGE privilege grantable only when the owner of that collation has a grantable USAGE privilege on both its contained translation, if any, and on its contained character set. For both changes, missing rules are provided that cover the acquisition of the necessary privileges to acquire the WITH GRANT OPTION on views through a grant to PUBLIC.

Replace General Rule 8) with the following:

- 8) For every <grantee> *G* and for every domain *D1* owned by *G*, if the user privileges of *G* contain the REFERENCES privilege WITH GRANT OPTION on every column referenced in the <search condition> included in a domain constraint descriptor included in the domain descriptor of *D1* and a grantable USAGE privilege on all domains, character sets, collations, and translations whose <domain name>s, <character set name>s, <collation name>s, and <translation name>s, respectively, are included in the domain descriptor, then for every privilege descriptor with <privileges> USAGE, a grantor of "_SYSTEM", object *D1*, and <grantee> *G* that is not grantable, the following <grant

statement> is effectively executed with a current <authorization identifier> of “_SYSTEM” and without further Access Rule checking:

GRANT USAGE ON DOMAIN *D1* TO *G* WITH GRANT OPTION

Replace General Rule 9) with the following:

- 9) For every <grantee> *G* and for every collation *C1* owned by *G*, if the user privileges of *G* contain a grantable USAGE privilege for the character set name included in the collation descriptor of *C1* and a grantable USAGE privilege for the translation name, if any, included in the collation descriptor of *C1*, then for every privilege descriptor with a <privileges> USAGE, a grantor of “_SYSTEM”, object of *C1*, and <grantee> *G* that is not grantable, the following <grant statement> is effectively executed with a current <authorization identifier> of “_SYSTEM” and without further Access Rule checking:

GRANT USAGE ON COLLATION *C1* TO *G* WITH GRANT OPTION

In General Rule 10), replace "USAGE privilege of *G* for every character set identified by a <character set specification> contained in the <translation definition> of *T1* is grantable" with "user privileges of *G* contain a grantable USAGE privilege for every character set identified by a <character set specification> contained in the <translation definition> of *T1*".

Rationale: Editorial.

In General Rule 10), replace "<grantor>" with "grantor" and replace "<object>" with "object".

Replace General Rule 11) with the following:

- 11) If <table name> is specified, then for each view *V* owned by some <grantee> *G* such that *T* or some column *CT* of *T* is referenced in the <query expression> of *V*, let RT_i for *i* ranging from 1 to the number of tables identified by the <table reference>s contained in the <query expression> of *V*, be the <table name>s of those tables. For every column *CV* of *V*:
- a) Let CRT_{ij} for *j* ranging from 1 to the number of columns of RT_i that are underlying columns of *CV*, be the <column name>s of those columns.
 - b) If WITH GRANT OPTION was specified, then let WGO be “WITH GRANT OPTION”; otherwise, let WGO be a zero-length string.
 - c) If, following successful execution of the <grant statement>, the user privileges of *G* will contain a REFERENCES(CRT_{ij}) privilege for all *i* and for all *j*, and will contain a REFERENCES privilege on some column of RT_i , for all *i*, then the following <grant statement> is effectively executed as though the current <authorization identifier> were “_SYSTEM” and without further Access Rule checking:

GRANT REFERENCES (CV) ON *V* TO *G* WGO

11.37 <revoke statement>

Rationale: Editorial

In Syntax Rule 6), replace "A privilege descriptor is" with "A privilege descriptor *P* is".

*Rationale: The following change address problems in the definition of when a privilege descriptor *D* is allowed to be created by a grant permitted by *P*, so as to properly deal with REFERENCES privilege*

against views.

Replace Syntax Rule 7)b)iv)2) with the following:

- 2) *P* and *D* are both column privilege descriptors, the privilege descriptor *D* identifies a <column name> *CVN* explicitly or implicitly contained in the <view column list> of a <view definition> *V* and either:
- A) *V* is an updatable view. For each column *CV* identified by a <column name> *CVN*, there is a corresponding column in the underlying table of the <query expression> *TN*. Let *CTN* be the <column name> of the column of the <query expression> from which *CV* is derived. The action for *P* is UPDATE or INSERT and the identified column of *P* is *TN.CTN*; or
 - B) For every table *T* identified by a <table reference> contained in the <query expression> of *V* and for every column *CT* that is a column of *T* and an underlying column of *CV*, the action for *P* is REFERENCES and either the identified column of *P* is *CT* or the identified table of *P* is *T*.

Rationale: The following change modifies the definition of "allowed to be created by" to reflect a collations dependency upon contained translations.

Replace Syntax Rule 7)b)iv)4) with the following:

- 4) The privilege descriptor *D* identifies the <collation name> of a <collation definition> *CO* and the identified character set name of *P* is included in the collation descriptor for *CO* or the identified translation name of *P* is included in the collation descriptor for *CO*.

*Rationale: The following change fixes the definition of when a privilege descriptor *D* is allowed to be created by a grant permitted by *P*, as that definition doesn't deal with USAGE privilege against domains properly.*

Modify Syntax Rule 7) by adding the following as Case c):

c) The following conditions hold:

- i) The privilege descriptor for *D* indicates that its grantor is the special grantor value “_SYSTEM”;
- ii) The grantee of *P* is the owner of the domain identified by *D*, or the grantee of *P* is PUBLIC, and
- iii) The privilege descriptor *D* identifies the <domain name> of a <domain definition> *DO* and either the column privilege descriptor *P* has an action of REFERENCES and identifies a column referenced in the <search condition> included in the domain descriptor for *DO*, or the privilege descriptor *P* has an action of USAGE and identifies a domain, collation, character set, or translation whose <domain name>, <collation name>, <character set name> or <translation name>, respectively, is contained in the <search condition> of the domain descriptor for *DO*.

Rationale: The following changes deal with the need that when determining whether or not the owner of a view has a grantable REFERENCES privilege on a particular column of a view, the owner must have a grantable REFERENCES privilege on every column that is an underlying column of that column.

*Rationale: This rule updates the definition of a modified privilege descriptor to deal with the case where revocation of REFERENCES privilege on one or more columns of a table *RTi* impacts one or more columns *CVN* of view *V*.*

Replace Syntax Rule 10)c) with the following:

c) Case:

i) The following conditions hold:

- 1) P is not a REFERENCES column privilege descriptor that identifies a <column name> CVN explicitly or implicitly contained in the <view column list> of a <view definition> V , and;
- 2) Let XO and XA respectively be the identifier of the object identified by a privilege descriptor X and the action of X . Within the set of privilege descriptors upon which P directly depends, there exists some XO and XA for which the set of identified privilege descriptors unioned with the set of modified privilege descriptors include all privilege descriptors specifying the grant of XA on XO WITH GRANT OPTION.

ii) The following conditions hold:

- 1) P is a REFERENCES column privilege descriptor that identifies a column CV named by a <column name> CVN explicitly or implicitly contained in the <view column list> of a <view definition> V , and;
- 2) Let SP be the set of privileges upon which P directly depends. For every table T identified by a <table reference> contained in the <query expression> of V , let RT be the <table name> of T . There exists a column CT whose <column name> is CRT , such that:
 - A) CT is a column of T and an underlying column of CV , and
 - B) Every privilege descriptor PD that is the descriptor of some member of SP that specifies REFERENCES on CRT WITH GRANT OPTION is either an identified privilege descriptor for CRT or a modified privilege descriptor for CRT .

and

Rationale: The following rule deals with the case where, due to a <revoke statement>, all REFERENCE privileges are lost against some underlying table of V , because in that instance all REFERENCE privileges should be lost against view V . The change above only deals with loss of GRANTability of REFERENCE against particular columns of V . To deal with this problem, it is necessary to define a new case where a privilege descriptor may become abandoned.

Modify Syntax Rule 11) by adding the following Case c):

c) The following conditions hold:

- i) P is a REFERENCES column privilege descriptor that identifies a <column name> CVN explicitly or implicitly contained in the <view column list> of a <view definition> V ; and,
- ii) GRANT OPTION FOR is not specified, and letting SP be the set of privileges upon which P directly depends, either:
 - 1) There exists some table name RT such that:
 - A) RT is the name of the table identified by some <table reference> contained in the <query expression> of V , and
 - B) For every column privilege descriptor CPD that is the descriptor of some member of SP

that specifies REFERENCES on *RT*, *CPD* is either an identified privilege descriptor for *RT* or an abandoned privilege descriptor for *RT*.

or

2) There exists some column name *CRT* such that:

A) *CRT* is the name of some column of the table of some <table reference> contained in the <query expression> of *V*, and

B) For every column privilege descriptor *CPD* that is the descriptor of some member of *SP* that specifies REFERENCES on *CRT*, *CPD* is either an identified privilege descriptor for *CRT* or an abandoned privilege descriptor for *CRT*.

Rationale: The following changes address an inconsistency in that to have grantable USAGE privilege on a column that specifies <collate clause>, its owner doesn't need grantable USAGE privilege on that overriding collation. But, to do the same for a domain, the owner needs grantable USAGE privilege on the overriding collation.

Delete Syntax Rules 17) and 18).

Rationale: The following changes provide a more consistent treatment of the <drop behavior> of the <revoke statement>.

Replace Syntax Rule 17) with the following:

17) For every domain descriptor *DO* contained in *S1*, *DO* is said to be *lost* if the destruction of all abandoned privileged descriptors, and, if GRANT OPTION FOR is not specified, all identified privilege descriptors would result in *A1* no longer having USAGE privilege on any character set included in the data type descriptor included in *DO*.

Add the following Syntax Rules:

18) For every table descriptor *TD* contained in *S1*, for every column descriptor *CD* included in *TD*, *CD* is said to be *lost* if either:

a) The destruction of all abandoned privilege descriptors, and, if GRANT OPTION FOR is not specified, all identified privilege descriptors would result in *A1* no longer having USAGE privilege on any character set included in the data type descriptor included in *CD*, or

b) The name of the domain *DN* included in *CD*, if any, identifies a lost domain descriptor and the destruction of all abandoned privilege descriptors, and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *A1* no longer having USAGE privilege on any character set included in the data type descriptor of the domain descriptor of *DN*.

19) For every module *MO*, let *G* be the <module authorization identifier> that owns *MO*. *MO* is said to be *lost* if the destruction of all abandoned privilege descriptors, and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *G* no longer having USAGE privilege on the character set referenced in the <module character set specification> of *MO*.

20) Let *SD* be the descriptor of the schema *S1*. *SD* is said to be *lost* if the destruction of all abandoned privilege descriptors, and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *A1* no longer having USAGE privilege on the default character set included in the schema descriptor *SD*.

Rationale: The following changes provide a more consistent treatment of the <drop behavior> of the <revoke statement>, as well as address an inconsistency in that to have grantable USAGE privilege on a column that specifies <collate clause>, its owner doesn't need grantable USAGE privilege on that overriding collation. But, to do the same for a domain, the owner needs grantable USAGE privilege on the overriding collation.

Add the following Syntax Rules:

- 21) For every domain descriptor *DO* contained in *S1*, *DO* is said to be *impacted* if *DO* is not lost, and the destruction of all abandoned privilege descriptors and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *A1* no longer having USAGE privilege on the collation whose <collation name> is contained in *DO*.
- 22) For every column descriptor *CD* contained in a table descriptor contained in *S1*, *CD* is said to be *impacted* if *CD* is not lost and the destruction of all abandoned privilege descriptors and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *A1* no longer having USAGE privilege on the collation whose <collation name> is contained in *CD*.
- 23) For every collation descriptor *CN* contained in *S1*, *CN* is said to be *impacted* if the destruction of all abandoned privilege descriptors and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *A1* no longer having USAGE privilege on the collation whose <collation name> is contained in *CN* as the name of the collation's <collation source>.
- 24) For every character set descriptor *CSD* contained in *S1*, *CSD* is said to be *impacted* if the destruction of all abandoned privilege descriptors and, if GRANT OPTION FOR is not specified, all identified privilege descriptors, would result in *A1* no longer having USAGE privilege on the collation whose <collation name> is contained in *CSD*.
- 25) If RESTRICT is specified, then there shall be no abandoned privilege descriptors, abandoned views, abandoned table constraints, abandoned assertions, abandoned domain constraints, lost domains, lost columns, lost modules, lost schema, impacted domains, impacted columns, impacted collations, or impacted character set descriptors.
- 26) If CASCADE is specified, then the impact on a module that is determined to be a lost module is implementation-defined.

Delete General Rule 7).

Rationale: The following changes specify the necessary actions against lost objects, given the changes above.

Add the following General Rules 7.1) - 7.3) between General Rules 7) and 8):

- 7.1) For every lost column descriptor *CD*, let *S1.TN* be the <table name> of the table whose descriptor includes the descriptor *CD* and let *CN* be the <column name> of *CD*. The following <alter table statement> is effectively executed without further Access Rule checking:

ALTER TABLE *S1.TN* DROP COLUMN *CN* CASCADE

- 7.2) For every lost domain descriptor *DO*, let *S1.DN* be the <domain name> of *DO*. The following <drop domain statement> is effectively executed without further Access Rule checking:

DROP DOMAIN *S1.DN* CASCADE

- 7.3) For every lost schema descriptor *SD*, the default character set of that schema is modified to

include the name of the implementation-defined <character set specification> that would have been this schema's default character set had the <schema definition> not specified a <schema character set specification>.

Rationale: The following changes describe the actions to take against impacted domains, columns, collations, and character sets.

Add the following General Rules 7.4) - 7.8) between General Rules 7.3) and 8):

- 7.4) If the object identified by *O* is a collation, let *OCN* be the name of that collation.
- 7.5) For every impacted domain descriptor *DO*, *DO* is modified such that it does not include *OCN*.
- 7.6) For every impacted column descriptor *CD*, *CD* is modified such that it does not include *OCN*.
- 7.7) For every impacted collation descriptor *CN*, *CN* is modified such that it does not include *OCN*. If *CN* does not include any translation name, then *CN* is modified to indicate that it utilizes the DEFAULT collation for its character repertoire.
- 7.8) For every impacted character set descriptor *CSD*, *CSD* is modified such that it does not include *OCN*. If *CSD* does not include any translation name, then *CSD* is modified to indicate that it utilizes the DEFAULT collation for its character repertoire.

12.3 <procedure>

Rationale: Resolve inconsistency regarding "no data" and exception condition.

Delete General Rules 8) and 9).

12.4 Calls to a <procedure>

Rationale: Editorial

In Syntax Rule 2)d), replace " $(L/(B+1))$ " with " (L/B) plus one".

13.1 <declare cursor>

Rationale: Editorial

Replace Syntax Rule 8) with the following:

- 8) The *simply underlying table* of the <cursor specification> is *T*.

Rationale: There is a contradiction between Syntax Rules 13) and 14) of Section 13.1 and Syntax Rule 8) and Access Rule 2) of Section 13.9.

Replace Syntax Rule 13) with the following:

- 13) If an <updatability clause> of FOR UPDATE without a <column name list> is specified or implicit, then a <column name list> that includes the <column name> of every column of the simply underlying table of the simply underlying table of *T* is implicit.

Replace Syntax Rule 14) with the following:

- 14) If an <updatability clause> of FOR UPDATE with a <column name list> is specified, then each

<column name> in the <column name list> shall be the <column name> of a column of the simply underlying table of the simply underlying table of *T*.

Delete Syntax Rule 9) and insert it before General Rule 1) as General Rule 0.1).

13.3 <fetch statement>

Rationale: Resolve inconsistency regarding "no data" and exception condition.

Replace General Rule 9) with the following:

- 9) If an exception condition occurs during the assignment of a value to a target, then the values of all targets are implementation-dependent and CR remains positioned on the current row.

13.5 <select statement: single row>

Rationale: Resolve inconsistency regarding "no data" and exception condition.

Delete General Rule 5) and add the following as General Rule 9):

- 9) If an exception condition is raised during the assignment of a value to a target, then the values of all targets are implementation-dependent.

13.6 <delete statement: positioned>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 4):

- 4) The schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*.

13.7 <delete statement: searched>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 3):

- 3) The schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*.

13.8 <insert statement>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 7):

- 7) The schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*.

Rationale: Editorial

ISO/IEC 9075:1992/Cor.1:1994(E)

In Access Rule 1)a), replace "<privileges>" with "privileges".

In Leveling Rule 2)a), replace "<insert statement>" with "<insert columns and source>".

Rationale: Editorial

In Leveling Rule 2)a), replace "<value specification>" with "<value specification> or a <null specification>".

13.9 <update statement: positioned>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 11):

11) The schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*.

Rationale: The following change is needed for the same reason that the changes to Subclause 11.5, "<default clause>" were necessary.

Replace General Rule 5) with the following:

5) The value associated with DEFAULT is the default value for the <object column> in the containing <set clause>, as indicated in the General Rules of Subclause 11.5, "<default clause>".

Rationale: There is a contradiction between Syntax Rules 13) and 14) of Section 13.1 and Syntax Rule 8) and Access Rule 2) of Section 13.9.

Delete Access Rule 2) and insert it as Syntax Rule 7.1), after Syntax Rule 7).

13.10 <update statement: searched>

Rationale: Clarify that references to non-existing objects is only allowed within the same <schema definition>.

Add the following as Syntax Rule 6):

6) The schema identified by the explicit or implicit qualifier of the <table name> shall include the descriptor of *T*.

14.1 <set transaction statement>

Rationale: Clarify the scope of the <set transaction statement>.

Add the following sentence to the end of the paragraph under Function:

Note: This statement has no effect on any SQL-transaction subsequent to the next SQL-transaction.

14.2 <set constraints mode statement>

Rationale: Clarify the scope of the <set transaction statement>.

Add the following sentence to the end of the paragraph under Function:

This statement has no effect on any SQL-transaction subsequent to this SQL-transaction.

14.3 <commit statement>

Rationale: Clarification

In General Rule 4), replace "were executed" with "were executed for each active SQL-connection".

15.1 <connect statement>

Rationale: Editorial

In General Rule 9)a), replace "SQL-server" with "SQL-session".

Rationale: Editorial

Replace General Rule 2) with the following:

- 2) If <user name> is specified, then let *S* be the character string that is the value of <user name> and let *V* be the character string that is the value of

TRIM (BOTH ' ' FROM *S*)

15.2 <set connection statement>

Rationale: Editorial

In General Rule 4), replace "SQL-server" with "SQL-session".

15.3 <disconnect statement>

Rationale: Editorial

In General Rule 5), replace "SQL-connection, if any" with "SQL-connection".

16.5 <set local time zone statement>

Rationale: Editorial

In Leveling Rule 1), replace "SQL;" with "SQL:".

In Leveling Rule 2), replace "SQL;" with "SQL, in addition to any Intermediate SQL restrictions:".

17.1 Description of SQL item descriptor areas

Rationale: Editorial

In Syntax Rule 3)e), change "date" to "data".

Rationale: Syntax Rule 3) applies only to data types that are not implementation-defined.

Add the following as Syntax Rule 2)l):

- l) TYPE indicates an implementation-defined data type and *T* satisfies the implementation-defined

ISO/IEC 9075:1992/Cor.1:1994(E)

rules for matching that data type.

Add the following as Syntax Rule 3)i):

i) TYPE indicates an implementation-defined data type.

17.2 <allocate descriptor statement>

Rationale: Editorial

In General Rule 2), replace "<scope clause>" with "<scope option>".

17.3 <deallocate descriptor statement>

Rationale: Editorial

In General Rule 1), replace "<scope clause>" with "<scope option>".

Rationale: Editorial

In Leveling Rule 2)a), replace "shall contain no" with "shall not contain any".

17.6 <prepare statement>

Rationale: The following unifies the SQLSTATE returned for the different ways of invoking an SQL statement.

In General Rule 4), replace "syntax error or access rule violation in dynamic SQL statement" with "syntax error or access rule violation".

Rationale: Editorial

In General Rule 4)g), replace "where *E1* is the" with "where the" and replace "of the <overlaps predicate>" with "of the <overlaps predicate> is *E1*".

In General Rule 4)i), replace "where *E1* is both the *i*-th" with "where both the *i*-th" and replace "of the <between predicate>" with "of the <between predicate> are *E1*".

In General Rule 4)k), replace "which *E1* is both" with "which both" and replace "of the <in value list>" with "of the <in value list> are *E1*".

In General Rule 8), replace "<simple target specification>" with "<simple value specification>".

17.11 <execute immediate statement>

Rationale: The following unifies the SQLSTATE returned for the different ways of invoking an SQL statement.

In General Rule 3), replace "syntax error or access rule violation in dynamic SQL statement" with "syntax error or access rule violation".

17.15 <dynamic fetch statement>

Rationale: Editorial

Replace Leveling Rule 1) with the following:

1) The following restrictions apply for Intermediate SQL:

None.

17.18 <dynamic update statement: positioned>

Rationale: Editorial

In Format, replace "<set clause> [{ <comma> <set clause> } ...]" with "<set clause list>".

Delete Syntax Rules 7), 8), and 11).

18.1 <get diagnostics statement>

Rationale: The following unifies the SQLSTATE returned for the different ways of invoking an SQL statement.

Replace the first paragraph of General Rule 3)g) with the following:

g) If the value of RETURNED_SQLSTATE corresponds to *syntax error or access rule violation*, then:

Replace the first paragraph of General Rule 3)j) with the following:

j) If the value of RETURNED_SQLSTATE does not correspond to *syntax error or access rule violation*, then:

Rationale: Editorial

In General Rule 3)n)i), replace "<SQL server specification>" with "<SQL-server name>".

Rationale: Editorial

In Leveling Rule 1), replace "SQL;" with "SQL:".

In Leveling Rule 2), replace "SQL;" with "SQL in addition to any Intermediate SQL restrictions:".

19.5 <embedded SQL COBOL program>

Rationale: Incorrect use of PICTURE B in COBOL syntax.

In Format, replace the BNF for <COBOL bit type> with the following:

```
<COBOL bit type> ::=
  { PIC | PICTURE } [ IS ]
  { X [ <left paren> <length> <right paren> ] } ...
  USAGE IS BIT
```

Replace Syntax Rule 5)b) with the following:

b) The syntax "USAGE IS BIT" shall be deleted in any <COBOL bit type>.

20.1 <direct SQL statement>

Rationale: The following unifies the SQLSTATE returned for the different ways of invoking an SQL statement.

In General Rule 5), replace "syntax error or access rule violation in direct SQL statement" with "syntax error or access rule violation".

21.1 Introduction

Rationale: The following clarifies how to handle <doublequote symbol> in identifiers stored in the Information Schema:

Replace the last paragraph with the following:

<identifier>s are represented in the tables of the Information Schema by the value of the <character string literal> corresponding to their <identifier body>s in the case of <regular identifier>s, or by their <delimited identifier body>s in the case of <delimited identifier>s. The corresponding <character string literal> and the comparison of <identifier>s is defined in Subclause 5.2, "<token> and <separator>". Where an <identifier> has many equivalent forms, the one encountered at definition time is stored, and any lower case letters appearing in a <regular identifier> are converted to the corresponding upper case letter, and any <doublequote symbol> appearing in a <delimited identifier> are replaced with a <double quote> before the <identifier> is stored in any table of the Information Schema.

Note: The value of the <character string literal> does not contain <quote>s that surround a <character string literal> or the "<introducer> <character set specification>" that can be specified with the <character string literal>.

21.2.2 INFORMATION_SCHEMA_CATALOG_NAME base table

Rationale: This base table is supposed to contain a single row; however, its primary key does not properly reflect this constraint.

Replace Definition with the following:

```
CREATE TABLE INFORMATION_SCHEMA_CATALOG_NAME
  ( CATALOG_NAME SQL_IDENTIFIER,
    CONSTRAINT INFORMATION_SCHEMA_CATALOG_NAME_PRIMARY_KEY
      PRIMARY KEY ( CATALOG_NAME ),
    CONSTRAINT INFORMATION_SCHEMA_CATALOG_NAME_CHECK
      CHECK ( ( SELECT COUNT (*) FROM INFORMATION_SCHEMA_CATALOG_NAME ) = 1 ) )
```

Rationale: Editorial

Replace the second occurrence of "**Definition**" with "**Description**".

21.2.3 INFORMATION_SCHEMA_CATALOG_NAME_CARDINALITY assertion

Rationale: Editorial

Delete the Description section.

Add the following at the end of the Subclause:

Leveling Rules

1) The following restrictions apply for Intermediate SQL:

None.

2) The following restrictions apply for Entry SQL in addition to any Intermediate SQL restrictions:

a) Conforming Entry SQL language shall not reference the Information Schema.

21.2.5 DOMAINS view

Rationale: In support of changes to 21.3.5.

In the Definition, replace "DATETIME_PRECISION, DOMAIN_DEFAULT" with "DATETIME_PRECISION, INTERVAL_TYPE, INTERVAL_PRECISION, DOMAIN_DEFAULT".

21.2.6 DOMAIN_CONSTRAINTS view

Rationale: Editorial

In Definition, replace "(S.CATALOG_NAME, SCHEMA_NAME S)" with "(S.CATALOG_NAME, S.SCHEMA_NAME)".

21.2.9 COLUMNS view

Rationale: In support of changes to 21.3.5.

In the Definition, insert the following line immediately after the line that reads "COALESCE (D1.DATETIME_PRECISION, D2.DATETIME_PRECISION) AS DATETIME_PRECISION,":

```
COALESCE ( D1.INTERVAL_TYPE, D2.INTERVAL_TYPE ) AS INTERVAL_TYPE,
COALESCE ( D1.INTERVAL_PRECISION, D2.INTERVAL_PRECISION ) AS
INTERVAL_PRECISION,
```

21.2.27 SQL_IDENTIFIER domain

Rationale: Editorial

In Function, replace "<identifier>s" with "<identifier>s.".

21.3.5 DATA_TYPE_DESCRIPTOR base table

Rationale: The descriptions of the existing columns do not allow for the fact that the interval type has two precisions: the fractional seconds precision and the interval leading field precision. In addition, the interval type also specifies the interval qualifier which must be encoded into the DATA_TYPE_DESCRIPTOR. Finally, numeric precision should not be allowed to be null for INTEGER, SMALLINT, NUMERIC, DECIMAL, DATE, and INTERVAL data types.

Replace Definition with the following:

```
CREATE TABLE DATA_TYPE_DESCRIPTOR
(
    TABLE_OR_DOMAIN_CATALOG          INFORMATION_SCHEMA.SQL_IDENTIFIER,
    TABLE_OR_DOMAIN_SCHEMA           INFORMATION_SCHEMA.SQL_IDENTIFIER,
    TABLE_OR_DOMAIN_NAME              INFORMATION_SCHEMA.SQL_IDENTIFIER,
    COLUMN_NAME                       INFORMATION_SCHEMA.SQL_IDENTIFIER,
    DATA_TYPE                         INFORMATION_SCHEMA.SQL_IDENTIFIER,
```

```

CONSTRAINT TABLE_OR_DOMAIN_DATA_TYPE_NOT_NULL NOT NULL,
CHARACTER_MAXIMUM_LENGTH      INFORMATION_SCHEMA.CARDINAL_NUMBER,
CHARACTER_OCTET_LENGTH        INFORMATION_SCHEMA.CARDINAL_NUMBER,
COLLATION_CATALOG              INFORMATION_SCHEMA.SQL_IDENTIFIER,
COLLATION_SCHEMA               INFORMATION_SCHEMA.SQL_IDENTIFIER,
COLLATION_NAME                 INFORMATION_SCHEMA.SQL_IDENTIFIER,
NUMERIC_PRECISION              INFORMATION_SCHEMA.CARDINAL_NUMBER,
NUMERIC_PRECISION_RADIX        INFORMATION_SCHEMA.CARDINAL_NUMBER,
NUMERIC_SCALE                  INFORMATION_SCHEMA.CARDINAL_NUMBER,
DATETIME_PRECISION             INFORMATION_SCHEMA.CARDINAL_NUMBER,
INTERVAL_TYPE                  INFORMATION_SCHEMA.CHARACTER_DATA,
INTERVAL_PRECISION             INFORMATION_SCHEMA.CARDINAL_NUMBER,
CONSTRAINT TABLE_OR_DOMAIN_CHECK_COMBINATIONS
CHECK ( DATA_TYPE IN ( 'CHARACTER', 'CHARACTER VARYING', 'BIT', 'BIT VARYING' )
      AND ( CHARACTER_MAXIMUM_LENGTH, CHARACTER_OCTET_LENGTH,
            COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME ) IS NOT NULL
      AND ( NUMERIC_PRECISION, NUMERIC_PRECISION_RADIX, NUMERIC_SCALE,
            DATETIME_PRECISION ) IS NULL
      AND ( INTERVAL_TYPE, INTERVAL_PRECISION ) IS NULL
OR
      DATA_TYPE IN ( 'REAL', 'DOUBLE_PRECISION', 'FLOAT' )
      AND ( CHARACTER_MAXIMUM_LENGTH, CHARACTER_OCTET_LENGTH,
            COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME ) IS NULL
      AND NUMERIC_PRECISION IS NOT NULL
      AND NUMERIC_PRECISION_RADIX = 2
      AND NUMERIC_SCALE IS NULL
      AND DATETIME_PRECISION IS NULL
      AND ( INTERVAL_TYPE, INTERVAL_PRECISION ) IS NULL
OR
      DATA_TYPE IN ( 'INTEGER', 'SMALLINT', 'NUMERIC', 'DECIMAL' )
      AND ( CHARACTER_MAXIMUM_LENGTH, CHARACTER_OCTET_LENGTH,
            COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME ) IS NULL
      AND ( NUMERIC_PRECISION, NUMERIC_PRECISION_RADIX ) IS NOT NULL
      AND NUMERIC_SCALE IS NOT NULL
      AND ( NUMERIC_SCALE <> 0 AND NUMERIC_PRECISION_RADIX = 10
            OR
            NUMERIC_SCALE = 0 AND NUMERIC_PRECISION_RADIX IN ( 2, 10 ) )
      AND DATETIME_PRECISION IS NULL
      AND ( INTERVAL_TYPE, INTERVAL_PRECISION ) IS NULL
OR
      DATA_TYPE IN ( 'DATE', 'TIME', 'TIMESTAMP', 'TIME WITH TIME ZONE',
            'TIMESTAMP WITH TIME ZONE' )
      AND ( CHARACTER_MAXIMUM_LENGTH, CHARACTER_OCTET_LENGTH,
            COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME ) IS NULL
      AND ( NUMERIC_PRECISION, NUMERIC_PRECISION_RADIX ) IS NULL
      AND NUMERIC_SCALE IS NULL
      AND DATETIME_PRECISION IS NOT NULL
      AND ( INTERVAL_TYPE, INTERVAL_PRECISION ) IS NULL
OR
      DATA_TYPE = 'INTERVAL'
      AND ( CHARACTER_MAXIMUM_LENGTH, CHARACTER_OCTET_LENGTH,
            COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME ) IS NULL
      AND ( NUMERIC_PRECISION, NUMERIC_PRECISION_RADIX ) IS NULL
      AND NUMERIC_SCALE IS NULL
      AND DATETIME_PRECISION IS NOT NULL

```

```

AND INTERVAL_TYPE IN ( 'YEAR', 'MONTH', 'DAY', 'HOUR', 'MINUTE', 'SECOND',
                        'YEAR TO MONTH', 'DAY TO HOUR', 'DAY TO MINUTE',
                        'DAY TO SECOND', 'HOUR TO MINUTE', 'HOUR TO SECOND',
                        MINUTE TO SECOND' )
AND INTERVAL_PRECISION IS NOT NULL
),

CONSTRAINT DATA_TYPE_DESCRIPTOR_PRIMARY_KEY
PRIMARY KEY ( TABLE_OR_DOMAIN_CATALOG, TABLE_OR_DOMAIN_SCHEMA,
              TABLE_OR_DOMAIN_NAME, COLUMN_NAME ),

CONSTRAINT DATA_TYPE_CHECK_REFERENCES_COLLATION
CHECK ( COLLATION_CATALOG <> ANY ( SELECT CATALOG_NAME FROM SCHEMATA )
      OR
      ( COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME ) IN
      ( SELECT COLLATION_CATALOG, COLLATION_SCHEMA, COLLATION_NAME
        FROM COLLATIONS ) ),

CONSTRAINT DATA_TYPE_DESCRIPTOR_CHECK_USED
CHECK ( ( TABLE_OR_DOMAIN_CATALOG, TABLE_OR_DOMAIN_SCHEMA,
          TABLE_OR_DOMAIN_NAME, COLUMN_NAME )
      IN ( SELECT TABLE_CATALOG, TABLE_SCHEMA, TABLE_NAME, COLUMN_NAME FROM
          COLUMNS
          UNION
          SELECT DOMAIN_CATALOG, DOMAIN_SCHEMA, DOMAIN_NAME, ' ' FROM DOMAINS )
      )
)

```

Rationale: Editorial. In Description 3), replace "numeric type, and" with "numeric type, the scale if it is a numeric type, and".

Add the following as Descriptions 4) and 5):

- 4) If DATA_TYPE is 'INTERVAL', then the values of INTERVAL_TYPE are the value for <interval qualifier> for the data type being described, excluding any <interval leading field precision> and an <interval fractional seconds precision>; otherwise, INTERVAL_TYPE is the null value.
- 5) If DATA_TYPE is 'INTERVAL', then the values of INTERVAL_PRECISION are the interval leading field precision of the data type being described; otherwise, INTERVAL_PRECISION is the null value.

21.3.6 DOMAINS base table

Rationale: The following helps define the action taken when a default value cannot be represented in the Information Schema:

In Description, replace the first sentence of Item 3) with the following:

The value of DOMAIN_DEFAULT is null if the domain being described has no default value; the value of DOMAIN_DEFAULT is TRUNCATED if the character representation of the <default option> cannot be represented without truncation.

Note: TRUNCATED is different from other values like CURRENT_USER, SYSTEM_USER, or CURRENT_DATE in that it is not an SQL keyword and does not correspond to a defined value in SQL.

21.3.8 TABLES base table

Rationale: Editorial

In Definition, in the last constraint, change "EXCEPTIONEXISTS" to "EXISTS".

21.3.10 COLUMNS base table

Rationale: The following helps define the action taken when a default value cannot be represented in the Information Schema:

In Description, replace the first sentence of Item 6) with the following:

The value of COLUMN_DEFAULT is null if the column being described has no default value or if its default value comes only from a domain; the value of COLUMN_DEFAULT is TRUNCATED if the character representation of the <default option> cannot be represented without truncation.

In Description, add the following at the end of Item 6):

Note: TRUNCATED is different from other values like CURRENT_USER, SYSTEM_USER, or CURRENT_DATE in that it is not an SQL keyword and does not correspond to a defined value in SQL.

21.3.15 REFERENTIAL_CONSTRAINTS base table

Rationale: Editorial

Change the first line of the last CHECK to the following:

```
CHECK ( UNIQUE_CONSTRAINT_CATALOG <> ANY ( SELECT CATALOG_NAME FROM SCHEMATA )
```

Rationale: Editorial

In Description 4), replace "update rule" with "<update rule>".

21.3.21 COLUMN_PRIVILEGES base table

Rationale: Editorial

In Description 4), for each of the four values for PRIVILEGE_TYPE, replace "CATALOG.TABLE", "SCHEMA.TABLE", "NAME.COLUMN" with "CATALOG, TABLE", "SCHEMA, TABLE", "NAME, and COLUMN", respectively.

21.3.22 USAGE_PRIVILEGES base table

Rationale: Editorial

In the Definition, replace "OBJECT_TYPE INFORMATION_SCHEMA.SQL_IDENTIFIER" with "OBJECT_TYPE INFORMATION_SCHEMA.CHARACTER_DATA".

21.3.24 Collations base table

Rationale: Editorial

In Description 1), replace "defined" with "described".

In Description 4), replace "A row always exists in this table for the collation SQL_TEXT." with "There is a row in this table for the collation INFORMATION_SCHEMA.SQL_TEXT."

21.3.26 SQL LANGUAGES base table

Rationale: Editorial

In Definition, replace "SQL_LANGUAGE_YEAR = '1989' AND AND" with "SQL_LANGUAGE_YEAR = '1989' AND".

22.1 SQLSTATE

Rationale: Editorial

Insert the following row into Table 23, after the row with Condition "data exception" and Subcondition "invalid escape character":

		invalid interval format	006
--	--	-------------------------	-----

Rationale: Editorial

In Table 23, change the Subcondition column for the entry with Condition "Remote Database Access" to read "(See Table 25 for subclass codes)".

Rationale: The following unifies the SQLSTATE returned for the different ways of invoking an SQL statement.

Delete the following rows from Table 23:

syntax error or access rule violation in direct SQL statement	2A (<i>no subclass</i>) 000
syntax error or access rule violation in dynamic SQL statement	37 (<i>no subclass</i>) 000

Rationale: The following helps define the action taken when a default value cannot be represented in the Information Schema:

Add the following rows to Table 23, between the second and third item with Condition "warning":

			default option too long for information 00B schema	
--	--	--	---	--

Rationale: Errors that prevent the execution of an SQL-statement may occur in RDA service elements, and should be reported to the SQL-agent through SQLSTATE and subclass codes.

Add a new subclass 22.3. "Remote Database Access SQLSTATE Subclasses":

22.3 Remote Database Access SQLSTATE Subclasses

This International Standard reserves SQLSTATE class 'HZ' for Remote Database Access errors, which may occur when an SQL-client interacts with an SQL-server across a communications network using an RDA Application Context. ISO/IEC 9579-1, ISO/IEC 9579-2, ISO 8649, and ISO/IEC 10026-2 define a number of exception conditions that must be detected in a conforming ISO RDA implementation.

ISO/IEC 9075:1992/Cor.1:1994(E)

This subclause defines SQLSTATE subclass codes for each such condition out of the set of codes reserved for International Standards.

If an implementation using RDA reports a condition shown in Table 25 for a given exception condition, then it shall use the SQLSTATE class code 'HZ' and the subclass codes shown, and shall set the values of CLASS_ORIGIN to 'ISO 9075' and SUBCLASS_ORIGIN as indicated in Table 25 when those exceptions are retrieved by a <get diagnostics statement>.

An implementation using client-server communications other than RDA may report conditions corresponding to the conditions shown in Table 25 using the SQLSTATE class code 'HZ' and the corresponding subclass codes shown. It may set the values of CLASS_ORIGIN to 'ISO 9075' and SUBCLASS_ORIGIN as indicated in Table 25. Any other communications error shall be returned with a subclass code from the implementation-defined range, with CLASS_ORIGIN set to 'ISO 9075' and SUBCLASS_ORIGIN set to an implementation-defined character string.

Table 25--SQLSTATE Subclasses for Class 'HZ'

RDA Generic Condition	SQLSTATE subclass
Subclass origin 'ISO/IEC 9579-1'	
(no subclass)	000
access control violation	010
bad repetition count	020
command handle unknown	030
control authentication failure	040
control services not allowed	230
data resource handle not specified	050
data resource handle unknown	060
data resource name not specified	070
data resource not available - premanent	080
data resource not available - transient	081
data resource already open	090
data resource unknown	100
dialogue id unknown	110
duplicate command handle	120
duplicate data resource handle	130
duplicate dialogue id	140
duplicate operation id	150
invalid sequence	160
* dialogue already active	161
* dialogue initializing	162
* dialogue not active	163
* dialogue terminating	164
* transaction not open	165
* transaction open	166
* transaction terminating	167
no data resource available	170
operation aborted - permanent	180
operation aborted - transient	181
operation cancelled	190
service not negotiated	200
transaction rolled back	210
user authentication failure	220

* Note: These are subcomponents of invalid sequence.

RDA SQL Specialization Condition	SQLSTATE subclass
Subclass origin 'ISO/IEC 9579-2'	
host identifier error	300
invalid SQL conformance level	310
RDA transaction not open	320
RDA transaction open	325
SQL access control violation	330
SQL database resource already open	340
SQL DBL argument count mismatch	350
SQL DBL argument type mismatch	360
SQL DBL no char set	365
SQL DBL transaction statement not allowed	370
SQL usage mode violation	380

Association Control Service Element Condition	SQLSTATE subclass
Subclass origin 'ISO 8649'	
abort failure service provider	410
abort failure service user	411
association failure - permanent	420
association failure - transient	421
release failure	430

Distributed Transaction Processing Condition	SQLSTATE subclass
Subclass origin 'ISO/IEC 10026-2'	
begin dialogue rejected provider	450
begin dialogue rejected user	451
heuristic hazard	460
heuristic mix	461
PAbort rollback false	470
PAbort rollback true	471
rollback	480
UAbort rollback false	490
UAbort rollback true	491
UError	4A0

Note: The error conditions given in this table correspond to ASN.1 error types formed by eliminating blanks between individual words and, in some cases, selecting one of a set of enumerated subparameters.

A Remote Database Access exception condition may also result in an SQL completion condition defined in Table 23 (such as '40000', Transaction Rollback); if such a condition occurs, then the 'HZ' class SQLSTATE value shall not be returned in the SQLSTATE parameter but may be returned in the Diagnostics Area.

23.3 Extensions and options

Rationale: Clarification.

In the first sentence of the first paragraph, replace "this International Standard" with "the level of this International Standard to which conformance is claimed".

A.1 Intermediate SQL Specifications

Rationale: Support changes to Subclause 7.11.

Delete Item 20).

A.2 Entry SQL Specifications

Rationale: Support changes to Subclause 7.11.

Add the following as Item 23)b):

b) The <query expression> contained in a <subquery> shall be a <query specification>.

Rationale: Support changes to Subclause 10.6.

Add the following as Item 31.1):

31.1) Subclause 10.6, "<constraint name definition> and <constraint attributes>":

a) Conforming Entry SQL language shall not contain any <constraint name definition>.

Rationale: Editorial

Replace Item 32)a) with the following:

a) Conforming Entry SQL language shall not contain any <domain definition>.

Replace Item 63)a) with the following:

a) The <query expression> that is contained in an <insert columns and source> shall be a <query specification> or it shall be a <table value constructor> that contains exactly one <row value constructor> of the form "<left paren> <row value constructor list> <right paren>", and each <row value constructor element> of that <row value constructor list> shall be a <value specification> or a <null specification>.

Annex B: Implementation-defined elements

Rationale: Clarification

In Item 24), replace "SQL-environment and create" with "SQL-server and initiate".

In Item 25), replace "both occurrences of "SQL-environment" with "SQL-server".

In Item 26), replace "environment" with "SQL-implementation".

Rationale: The above changes require a change in Annex B.

Add the following item 90.1) between items 90) and 91):

ISO/IEC 9075:1992/Cor.1:1994(E)

- 90.1) Subclause 11.37, "<revoke statement>": When loss of the USAGE privilege on a character set causes a module to be determined to be a lost module, the impact on that module is implementation-defined.

Rationale: Clarification

In Item 112), replace "SQL-environment" with "SQL-server".

In Item 113), replace "SQL-environment" with "SQL-server".

Annex C Implementation-dependent elements

Rationale: Editorial

Delete Item 41).

Add the following Item 42.1) between Items 42) and 43):

- 42.1) Subclause 12.1, "<module>": If the SQL-agent performs calls of <procedure>s from more than one Ada task, then the results are implementation-dependent.

Add the following Item 42.2) between Items 42.1) and 43):

- 42.2) Subclause 12.1, "<module>": After the execution of the last <procedure>, if an unrecoverable error has not occurred, and the SQL-agent did not terminate unexpectedly, and no constraint is not satisfied, then the choice of whether to perform a <commit statement> or a <rollback statement> is implementation-dependent. The determination of whether an SQL-agent has terminated unexpectedly is implementation-dependent.

Delete Item 44).

Insert the following as the first sentence of Item 72):

A <rollback statement> or a <commit statement> is effectively executed.

Rationale: Resolve inconsistency regarding "no data" and exception condition.

Replace Item 49) with the following:

- 49) Subclause 13.3 "<fetch statement>": If an exception condition occurs during the assignment of a value to a target, then the values of all targets are implementation-dependent.

Annex E Incompatibilities with ISO/IEC 9075:1992

Rationale: Editorial

In Item 10), delete "-- BETWEEN" and insert "-- CURRENT" between "-- CROSS" and "CURRENT_DATE".

INTERNATIONAL STANDARD

ISO/IEC
9075

Third edition
1992-11-01

Information technology — Database languages — SQL

Technologies de l'information — Langages de base de données — SQL



Reference number
ISO/IEC 9075:1992/E1



Documents
Document Sharing Hub

Contents

	Page
Foreword	xi
Introduction	xiii
1 Scope	1
2 Normative references	3
3 Definitions, notations, and conventions	5
3.1 Definitions	5
3.1.1 Definitions taken from ISO/IEC 10646	5
3.1.2 Definitions taken from ISO 8601	5
3.1.3 Definitions provided in this International Standard	5
3.2 Notation	7
3.3 Conventions	8
3.3.1 Informative elements	8
3.3.2 Specification of syntactic elements	8
3.3.3 Specification of the Information Schema	9
3.3.4 Use of terms	9
3.3.4.1 Exceptions	9
3.3.4.2 Syntactic containment	9
3.3.4.3 Terms denoting rule requirements	10
3.3.4.4 Rule evaluation order	10
3.3.4.5 Conditional rules	11
3.3.4.6 Syntactic substitution	11
3.3.4.7 Other terms	12
3.3.5 Descriptors	12
3.3.6 Index typography	13
3.4 Object identifier for Database Language SQL	13
4 Concepts	15
4.1 Data types	15
4.2 Character strings	16
4.2.1 Character strings and collating sequences	16

© ISO/IEC 1992

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland
Printed in Switzerland

4.2.2	Operations involving character strings	17
4.2.2.1	Operators that operate on character strings and return character strings	17
4.2.2.2	Other operators involving character strings	18
4.2.3	Rules determining collating sequence usage	18
4.3	Bit strings	21
4.3.1	Bit string comparison and assignment	21
4.3.2	Operations involving bit strings	21
4.3.2.1	Operators that operate on bit strings and return bit strings	21
4.3.2.2	Other operators involving bit strings	21
4.4	Numbers	21
4.4.1	Characteristics of numbers	22
4.4.2	Operations involving numbers	23
4.5	Datetimes and intervals	23
4.5.1	Datetimes	23
4.5.2	Intervals	25
4.5.3	Operations involving datetimes and intervals	26
4.6	Type conversions and mixing of data types	27
4.7	Domains	28
4.8	Columns	28
4.9	Tables	29
4.10	Integrity constraints	31
4.10.1	Checking of constraints	32
4.10.2	Table constraints	32
4.10.3	Domain constraints	33
4.10.4	Assertions	33
4.11	SQL-schemas	34
4.12	Catalogs	34
4.13	Clusters of catalogs	35
4.14	SQL-data	35
4.15	SQL-environment	35
4.16	Modules	36
4.17	Procedures	36
4.18	Parameters	36
4.18.1	Status parameters	36
4.18.2	Data parameters	37
4.18.3	Indicator parameters	37
4.19	Diagnostics area	37
4.20	Standard programming languages	38
4.21	Cursors	38
4.22	SQL-statements	39
4.22.1	Classes of SQL-statements	39
4.22.2	SQL-statements classified by function	40
4.22.3	Embeddable SQL-statements	43
4.22.4	Preparable and immediately executable SQL-statements	44
4.22.5	Directly executable SQL-statements	46
4.22.6	SQL-statements and transaction states	47
4.23	Embedded syntax	48
4.24	SQL dynamic statements	49
4.25	Direct invocation of SQL	51

4.26	Privileges	51
4.27	SQL-agents	53
4.28	SQL-transactions	53
4.29	SQL-connections	56
4.30	SQL-sessions	57
4.31	Client-server operation	59
4.32	Information Schema	59
4.33	Leveling	60
4.34	SQL Flagger	60
5	Lexical elements	63
5.1	<SQL terminal character>	63
5.2	<token> and <separator>	66
5.3	<literal>	71
5.4	Names and identifiers	78
6	Scalar expressions	85
6.1	<data type>	85
6.2	<value specification> and <target specification>	91
6.3	<table reference>	94
6.4	<column reference>	96
6.5	<set function specification>	98
6.6	<numeric value function>	101
6.7	<string value function>	105
6.8	<datetime value function>	110
6.9	<case expression>	112
6.10	<cast specification>	114
6.11	<value expression>	124
6.12	<numeric value expression>	126
6.13	<string value expression>	128
6.14	<datetime value expression>	132
6.15	<interval value expression>	135
7	Query expressions	139
7.1	<row value constructor>	139
7.2	<table value constructor>	141
7.3	<table expression>	142
7.4	<from clause>	143
7.5	<joined table>	145
7.6	<where clause>	150
7.7	<group by clause>	151
7.8	<having clause>	153
7.9	<query specification>	155
7.10	<query expression>	159
7.11	<scalar subquery>, <row subquery>, and <table subquery>	165
8	Predicates	167
8.1	<predicate>	167
8.2	<comparison predicate>	169
8.3	<between predicate>	172

8.4	<in predicate>	173
8.5	<like predicate>	175
8.6	<null predicate>	178
8.7	<quantified comparison predicate>	180
8.8	<exists predicate>	182
8.9	<unique predicate>	183
8.10	<match predicate>	184
8.11	<overlaps predicate>	186
8.12	<search condition>	188
9	Data assignment rules	191
9.1	Retrieval assignment	191
9.2	Store assignment	193
9.3	Set operation result data types	195
10	Additional common elements	197
10.1	<interval qualifier>	197
10.2	<language clause>	201
10.3	<privileges>	203
10.4	<character set specification>	205
10.5	<collate clause>	207
10.6	<constraint name definition> and <constraint attributes>	208
11	Schema definition and manipulation	211
11.1	<schema definition>	211
11.2	<drop schema statement>	214
11.3	<table definition>	216
11.4	<column definition>	218
11.5	<default clause>	221
11.6	<table constraint definition>	224
11.7	<unique constraint definition>	226
11.8	<referential constraint definition>	228
11.9	<check constraint definition>	233
11.10	<alter table statement>	235
11.11	<add column definition>	236
11.12	<alter column definition>	237
11.13	<set column default clause>	238
11.14	<drop column default clause>	239
11.15	<drop column definition>	240
11.16	<add table constraint definition>	242
11.17	<drop table constraint definition>	243
11.18	<drop table statement>	244
11.19	<view definition>	245
11.20	<drop view statement>	249
11.21	<domain definition>	250
11.22	<alter domain statement>	252
11.23	<set domain default clause>	253
11.24	<drop domain default clause>	254
11.25	<add domain constraint definition>	255
11.26	<drop domain constraint definition>	256

11.27	<drop domain statement>	257
11.28	<character set definition>	259
11.29	<drop character set statement>	261
11.30	<collation definition>	262
11.31	<drop collation statement>	265
11.32	<translation definition>	267
11.33	<drop translation statement>	269
11.34	<assertion definition>	270
11.35	<drop assertion statement>	272
11.36	<grant statement>	273
11.37	<revoke statement>	276
12	Module	281
12.1	<module>	281
12.2	<module name clause>	284
12.3	<procedure>	285
12.4	Calls to a <procedure>	290
12.5	<SQL procedure statement>	303
13	Data manipulation	307
13.1	<declare cursor>	307
13.2	<open statement>	310
13.3	<fetch statement>	312
13.4	<close statement>	315
13.5	<select statement: single row>	316
13.6	<delete statement: positioned>	318
13.7	<delete statement: searched>	320
13.8	<insert statement>	322
13.9	<update statement: positioned>	325
13.10	<update statement: searched>	328
13.11	<temporary table declaration>	330
14	Transaction management	333
14.1	<set transaction statement>	333
14.2	<set constraints mode statement>	335
14.3	<commit statement>	337
14.4	<rollback statement>	339
15	Connection management	341
15.1	<connect statement>	341
15.2	<set connection statement>	344
15.3	<disconnect statement>	346
16	Session management	349
16.1	<set catalog statement>	349
16.2	<set schema statement>	350
16.3	<set names statement>	351
16.4	<set session authorization identifier statement>	352
16.5	<set local time zone statement>	354

17	Dynamic SQL	355
17.1	Description of SQL item descriptor areas	355
17.2	<allocate descriptor statement>	360
17.3	<deallocate descriptor statement>	362
17.4	<get descriptor statement>	363
17.5	<set descriptor statement>	366
17.6	<prepare statement>	369
17.7	<deallocate prepared statement>	375
17.8	<describe statement>	376
17.9	<using clause>	377
17.10	<execute statement>	383
17.11	<execute immediate statement>	385
17.12	<dynamic declare cursor>	387
17.13	<allocate cursor statement>	388
17.14	<dynamic open statement>	390
17.15	<dynamic fetch statement>	392
17.16	<dynamic close statement>	394
17.17	<dynamic delete statement: positioned>	395
17.18	<dynamic update statement: positioned>	396
17.19	<preparable dynamic delete statement: positioned>	398
17.20	<preparable dynamic update statement: positioned>	399
18	Diagnostics management	401
18.1	<get diagnostics statement>	401
19	Embedded SQL	411
19.1	<embedded SQL host program>	411
19.2	<embedded exception declaration>	418
19.3	<embedded SQL Ada program>	421
19.4	<embedded SQL C program>	424
19.5	<embedded SQL COBOL program>	428
19.6	<embedded SQL Fortran program>	431
19.7	<embedded SQL MUMPS program>	434
19.8	<embedded SQL Pascal program>	436
19.9	<embedded SQL PL/I program>	439
20	Direct invocation of SQL	443
20.1	<direct SQL statement>	443
20.2	<direct select statement: multiple rows>	447
21	Information Schema and Definition Schema	449
21.1	Introduction	449
21.2	Information Schema	450
21.2.1	INFORMATION_SCHEMA Schema	450
21.2.2	INFORMATION_SCHEMA_CATALOG_NAME base table	451
21.2.3	INFORMATION_SCHEMA_CATALOG_NAME_CARDINALITY assertion	452
21.2.4	SCHEMATA view	453
21.2.5	DOMAINS view	454
21.2.6	DOMAIN_CONSTRAINTS view	455
21.2.7	TABLES view	456

21.2.8 VIEWS view	457
21.2.9 COLUMNS view	458
21.2.10 TABLE_PRIVILEGES view	460
21.2.11 COLUMN_PRIVILEGES view	461
21.2.12 USAGE_PRIVILEGES view	462
21.2.13 TABLE_CONSTRAINTS view	463
21.2.14 REFERENTIAL_CONSTRAINTS view	464
21.2.15 CHECK_CONSTRAINTS view	465
21.2.16 KEY_COLUMN_USAGE view	466
21.2.17 ASSERTIONS view	467
21.2.18 CHARACTER_SETS view	468
21.2.19 COLLATIONS view	469
21.2.20 TRANSLATIONS view	470
21.2.21 VIEW_TABLE_USAGE view	471
21.2.22 VIEW_COLUMN_USAGE view	472
21.2.23 CONSTRAINT_TABLE_USAGE view	473
21.2.24 CONSTRAINT_COLUMN_USAGE view	474
21.2.25 COLUMN_DOMAIN_USAGE view	475
21.2.26 SQL_LANGUAGES view	476
21.2.27 SQL_IDENTIFIER domain	477
21.2.28 CHARACTER_DATA domain	477
21.2.29 CARDINAL_NUMBER domain	478
21.3 Definition Schema	479
21.3.1 Introduction	479
21.3.2 DEFINITION_SCHEMA Schema	480
21.3.3 USERS base table	481
21.3.4 SCHEMATA base table	482
21.3.5 DATA_TYPE_DESCRIPTOR base table	483
21.3.6 DOMAINS base table	485
21.3.7 DOMAIN_CONSTRAINTS base table	486
21.3.8 TABLES base table	488
21.3.9 VIEWS base table	489
21.3.10 COLUMNS base table	491
21.3.11 VIEW_TABLE_USAGE base table	493
21.3.12 VIEW_COLUMN_USAGE base table	494
21.3.13 TABLE_CONSTRAINTS base table	495
21.3.14 KEY_COLUMN_USAGE base table	497
21.3.15 REFERENTIAL_CONSTRAINTS base table	499
21.3.16 CHECK_CONSTRAINTS base table	501
21.3.17 CHECK_TABLE_USAGE base table	502
21.3.18 CHECK_COLUMN_USAGE base table	503
21.3.19 ASSERTIONS base table	504
21.3.20 TABLE_PRIVILEGES base table	505
21.3.21 COLUMN_PRIVILEGES base table	507
21.3.22 USAGE_PRIVILEGES base table	509
21.3.23 CHARACTER_SETS base table	511
21.3.24 COLLATIONS base table	513
21.3.25 TRANSLATIONS base table	515
21.3.26 SQL_LANGUAGES base table	517