INTERNATIONAL STANDARD ISO 13584-20:1998
TECHNICAL CORRIGENDUM 1

Published 2014-07-01

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION

Industrial automation systems and integration — Parts library —

Part 20:
Logical resource: Logical model of expressions

TECHNICAL CORRIGENDUM 1

Systèmes d'automatisation industrielle et intégration — Bibliothèque de composants —

Partie 20: Ressource logique: Modèle logique d'expressions

RECTIFICATIF TECHNIQUE 1

Technical corrigendum 1 to International Standard ISO 13584-20:1998 was prepared by Technical Committee ISO/TC 184,*Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

The purpose of the modifications to the text of ISO 13584-20:1998 is to integrate the various SEDS reports that have been collected since its publication.

ICS 25.040.40 **Ref. No. ISO 13584-20:1998/Cor.1:2014(E)**

Published in Switzerland

Modifications to the text of ISO 13584-20:1998

Clause 6 Schema name, p8

The schema name shall be written using lowercase letters. Update the EXPRESS schema name as follows:

```
SCHEMA iso13584_generic_expressions_schema;
```

Additionally, all the references to the schema name (both in the text and in the EXPRESS code) is also updated.

Clause 6.3.1 Is_acyclic function, p13

A RETURN statement is not included between the final END_IF statement and the end of the function, resulting in possible invalid results. Consequently, the 'result' variable shall be initialized. Update the EXPRESS specification with the following:

```
FUNCTION acyclic (arg1: generic_expression;
                  arg2: SET OF generic_expression): BOOLEAN;

LOCAL
      result: BOOLEAN := TRUE;
END_LOCAL;
…
      REPEAT i := 1 TO
                  SIZEOF
                  (arg1\multiple_arity_generic_expression.operands);
            result := result AND
            acyclic(arg1\multiple_arity_generic_expression.operands[i],
                  arg2+[arg1]);
      END_REPEAT;

      RETURN (result);
END_IF;

RETURN (result);

END_FUNCTION; -- acyclic
```

Clause 7 Schema name, p15

The schema name shall be written using lower case letters. Update the EXPRESS schema name as follows:

```
SCHEMA iso13584_expressions_schema;
```

Additionally, all the references to the schema name (both in the text and in the EXPRESS code) is also updated.

Clause 7.4.6 Odd_Function, WR1, p35

*The EXPRESS declaration for **WR1** is not in compliance with the English description. Replace the WR1 specification with the following:*


```
WR1: is_int_expr(operand);
```

Clause 7.4.13,Comparison_expression, WR1, p38

*There is an error in the express for rule **WR1** of entity **comparison_expression**. Replace the rule with the following:*

```
WR1: (('ISO13584_EXPRESSIONS_SCHEMA.NUMERIC_EXPRESSION'
               IN TYPEOF(SELF\binary_generic_expression.operands[1]))
          AND
               ('ISO13584_EXPRESSIONS_SCHEMA.NUMERIC_EXPRESSION'
               IN TYPEOF(SELF\binary_generic_expression.operands[2])))
OR
(('ISO13584_EXPRESSIONS_SCHEMA.BOOLEAN_EXPRESSION'
     IN TYPEOF(SELF\binary_generic_expression.operands[1]))
          AND
               ('ISO13584_EXPRESSIONS_SCHEMA.BOOLEAN_EXPRESSION'
     IN TYPEOF(SELF\binary_generic_expression.operands[2])))
OR
(('ISO13584_EXPRESSIONS_SCHEMA.STRING_EXPRESSION'
     IN TYPEOF(SELF\binary_generic_expression.operands[1]))
          AND
               ('ISO13584_EXPRESSIONS_SCHEMA.STRING_EXPRESSION'
               IN TYPEOF(SELF\binary_generic_expression.operands[2])))
;
```

Clause 7.4.21 Interval_expression, WR2, p41

*The types of the expressions to be compared in the **interval_expression** shall evaluate to comparable expressions. But, when an **interval_expression** is specified based on numeric expressions, **WR2** restricts the type of the **interval_low** attribute to be a string expression, what is erroneous. Change the **WR2** specification with the following:*

```
WR2:(('ISO13584_EXPRESSIONS_SCHEMA.STRING_EXPRESSION'
                                IN TYPEOF (SELF.interval_low))
     AND ('ISO13584_EXPRESSIONS_SCHEMA.STRING_EXPRESSION'
                                IN TYPEOF (SELF.interval_high))
     AND ('ISO13584_EXPRESSIONS_SCHEMA.STRING_EXPRESSION'
                                IN TYPEOF (SELF.interval_item)))
OR
     (('ISO13584_EXPRESSIONS_SCHEMA.NUMERIC_EXPRESSION'
                                IN TYPEOF(SELF.interval_low))
     AND ('ISO13584_EXPRESSIONS_SCHEMA.NUMERIC_EXPRESSION'
                                IN TYPEOF(SELF.interval_item))
AND ('ISO13584_EXPRESSIONS_SCHEMA.NUMERIC_EXPRESSION'
IN TYPEOF(SELF.interval_high)));
```

Clause 7.6.1 Is_int_expr, p48

*The local variable **i** is declared, but never used. It shall be removed from the EXPRESS specification.*

Clause 7.6.2 Is_SQL_mappable, p50

*The local variable **i** is declared, but never used. It shall be removed from the EXPRESS specification.*

Clause 7.6.3, used_functions function, p53

*When the type of the **arg** formal parameter is an 'ISO13584_EXPRESSIONS_SCHEMA.LIKE_EXPRESSION', the function is recursively called twice with some bad partial entity instance references. Replace with the following:*

```
FUNCTION used_functions (arg : expression) : SET OF defined_function;
...
IF 'ISO13584_EXPRESSIONS_SCHEMA.LIKE_EXPRESSION' IN TYPEOF (arg)
THEN
     RETURN (used_functions (arg\comparison_expression.operands[1])
          + used_functions (arg\comparison_expression.operands[2]));
END_IF;
```

Annex A, Table A.1, p56

In Table A.1, some short names are missing for the entities whose first letter is between 'i' and 'r'. Replace Table A.1 by the following:

Table A.1 — Short names of entities

Long name	Short name
ABS_FUNCTION	ABSFNC
ACOS_FUNCTION	ACSFNC
AND_EXPRESSION	ANDEXP
ASIN_FUNCTION	ASNFNC
ATAN_FUNCTION	ATNFNC
BINARY_BOOLEAN_EXPRESSION	BNBLEX
BINARY_FUNCTION_CALL	BNFNCL
BINARY_GENERIC_EXPRESSION	BNGNEX
BINARY_NUMERIC_EXPRESSION	BNNMEX
BOOLEAN_DEFINED_FUNCTION	BLDFFN
BOOLEAN_EXPRESSION	BLNEXP
BOOLEAN_LITERAL	BLNLTR
BOOLEAN_VARIABLE	BLNVRB
COMPARISON_EQUAL	CMPEQL
COMPARISON_EXPRESSION	CMPEXP
COMPARISON_GREATER	CMPGRT
COMPARISON_GREATER_EQUAL	CMGREQ
COMPARISON_LESS	CMPLSS
COMPARISON_LESS_EQUAL	CMLSEQ
COMPARISON_NOT_EQUAL	CMNTEQ
CONCAT_EXPRESSION	CNCEXP
COS_FUNCTION	CSFNC
DEFINED_FUNCTION	DFNFNC

DIV_EXPRESSION	DVEXP
ENVIRONMENT	ENVRNM
EQUALS_EXPRESSION	EQLEXP
EXP_FUNCTION	EXPFNC
EXPRESSION	EXPRSS
FORMAT_FUNCTION	FRMFNC
GENERIC_EXPRESSION	GNREXP
GENERIC_LITERAL	GNRLTR
GENERIC_VARIABLE	GNRVRB

Table A.1 (continued)

Long name	Short name
INDEX_EXPRESSION	INDEXP
INT_LITERAL	INTLTR
INT_NUMERIC_VARIABLE	INNMVR
INT_VALUE_FUNCTION	INVLFN
INTEGER_DEFINED_FUNCTION	INDFFN
INTERVAL_EXPRESSION	INTEXP
LENGTH_FUNCTION	LNGFNC
LIKE_EXPRESSION	LKEXP
LITERAL_NUMBER	LTRNMB
LOG_FUNCTION	LGFNC
LOG10_FUNCTION	LG1FNC
LOG2_FUNCTION	LG2FNC
MAXIMUM_FUNCTION	MXMFNC
MINIMUM_FUNCTION	MNMFNC
MINUS_EXPRESSION	MNSEXP
MINUS_FUNCTION	MNSFNC
MOD_EXPRESSION	MDEXP
MULT_EXPRESSION	MLTEXP
MULTIPLE_ARITY_BOOLEAN_EXPRESSION	MABE
MULTIPLE_ARITY_FUNCTION_CALL	MAFC
MULTIPLE_ARITY_GENERIC_EXPRESSION	MAGE
MULTIPLE_ARITY_NUMERIC_EXPRESSION	MANE
NOT_EXPRESSION	NTEXP
NUMERIC_DEFINED_FUNCTION	NMDFFN
NUMERIC_EXPRESSION	NMREXP
NUMERIC_VARIABLE	NMRVRB
ODD_FUNCTION	ODDFNC
OR_EXPRESSION	OREXP
PLUS_EXPRESSION	PLSEXP
POWER_EXPRESSION	PWREXP
REAL_DEFINED_FUNCTION	RLDFFN
REAL_LITERAL	RLLTR
REAL_NUMERIC_VARIABLE	RLNMVR

Table A.1 (continued)

Long name	Short name
SIMPLE_BOOLEAN_EXPRESSION	SMBLEX
SIMPLE_GENERIC_EXPRESSION	SMGNEX
SIMPLE_NUMERIC_EXPRESSION	SMNMEX
SIMPLE_STRING_EXPRESSION	SMSTEX
SIN_FUNCTION	SNFNC
SLASH_EXPRESSION	SLSEXP
SQL_MAPPABLE_DEFINED_FUNCTION	SMDF
SQUARE_ROOT_FUNCTION	SQRTFN
STRING_DEFINED_FUNCTION	STDFFN
STRING_EXPRESSION	STREXP
STRING_LITERAL	STRLTR
STRING_VARIABLE	STRVRB
SUBSTRING_EXPRESSION	SBSEXP
TAN_FUNCTION	TNFNC
UNARY_BOOLEAN_EXPRESSION	UNBLEX
UNARY_FUNCTION_CALL	UNFNCL
UNARY_GENERIC_EXPRESSION	UNGNEX
UNARY_NUMERIC_EXPRESSION	UNNMEX
VALUE_FUNCTION	VLFNC
VARIABLE	VRBL
VARIABLE_SEMANTICS	VRBSMN
XOR_EXPRESSION	XREXP

Annex C – EXPRESS-G diagrams, p59

Some EXPRESS-G diagrams are not in line with the textual EXPRESS specification:

- *Figure C.7: replace integer_defined_function with (ABS) integer_defined_function;*
- *Figure C.7: replace real_defined_function with (ABS) real_defined_function;*
- *Figure C.9: replace binary_function_call with (ABS) binary_function_call;*
- *Figure C.13: replace comparaison_expression with comparison_expression.*

Consequently, EXPRESS-G diagrams in Annex C are replaced with the followings:

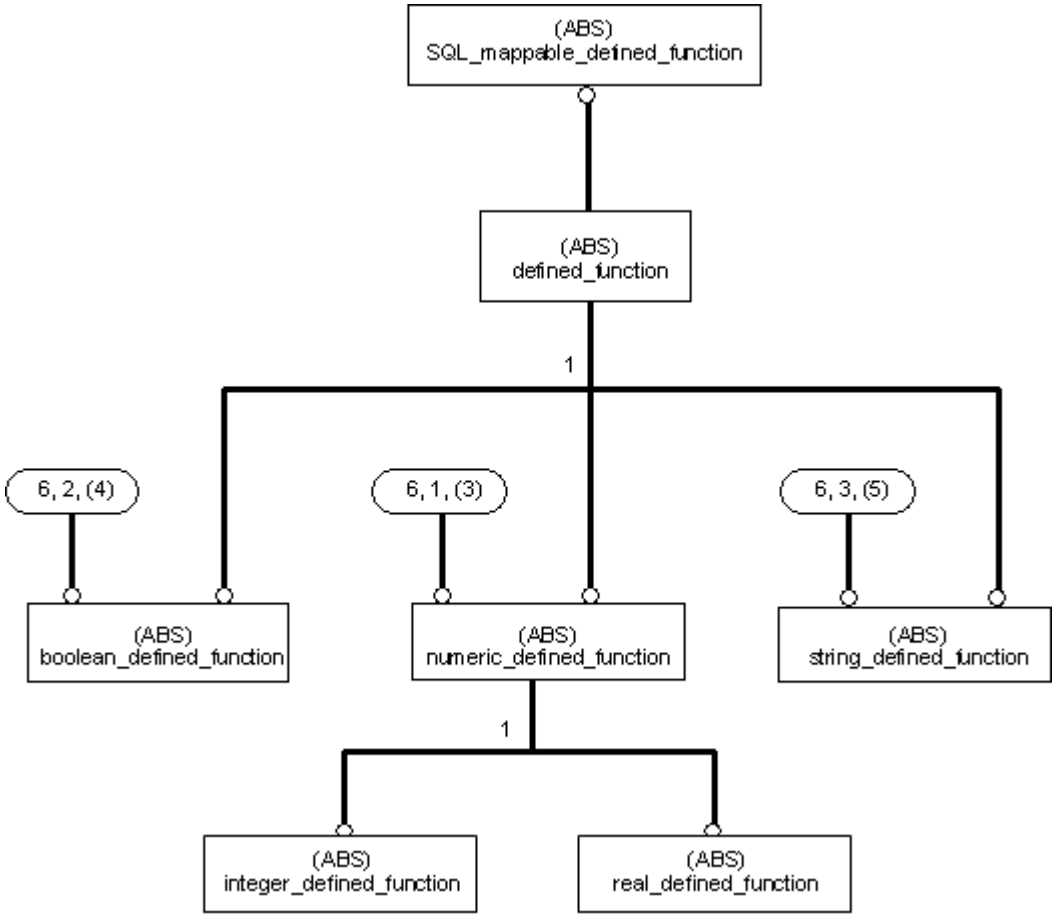

Figure C.1 — iso13584_expressions_schema - EXPRESS-G diagram 6 of 12 - defined_function

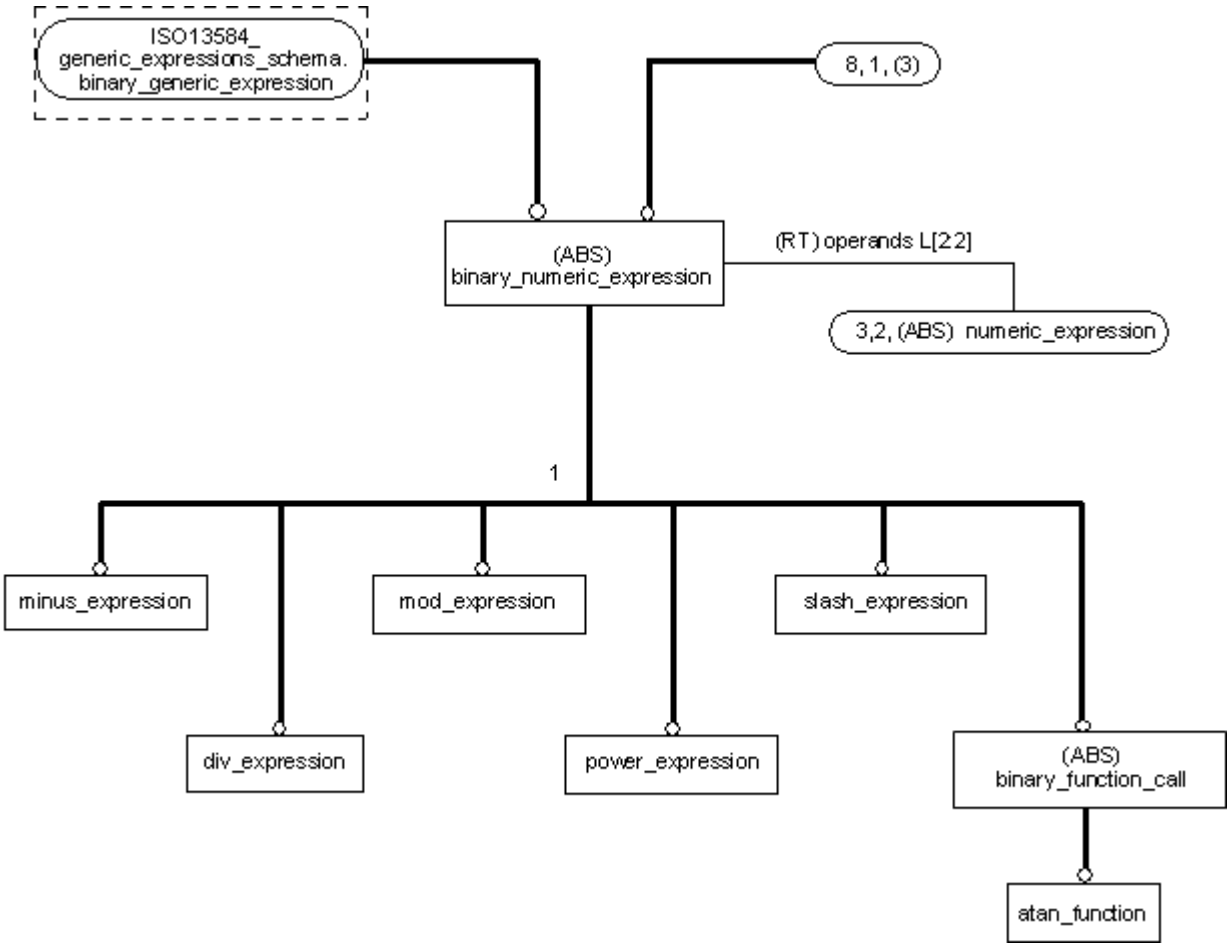

Figure C.9 — iso13584_expressions_schema - EXPRESS-G diagram 8 of 12- Binary numeric expressions

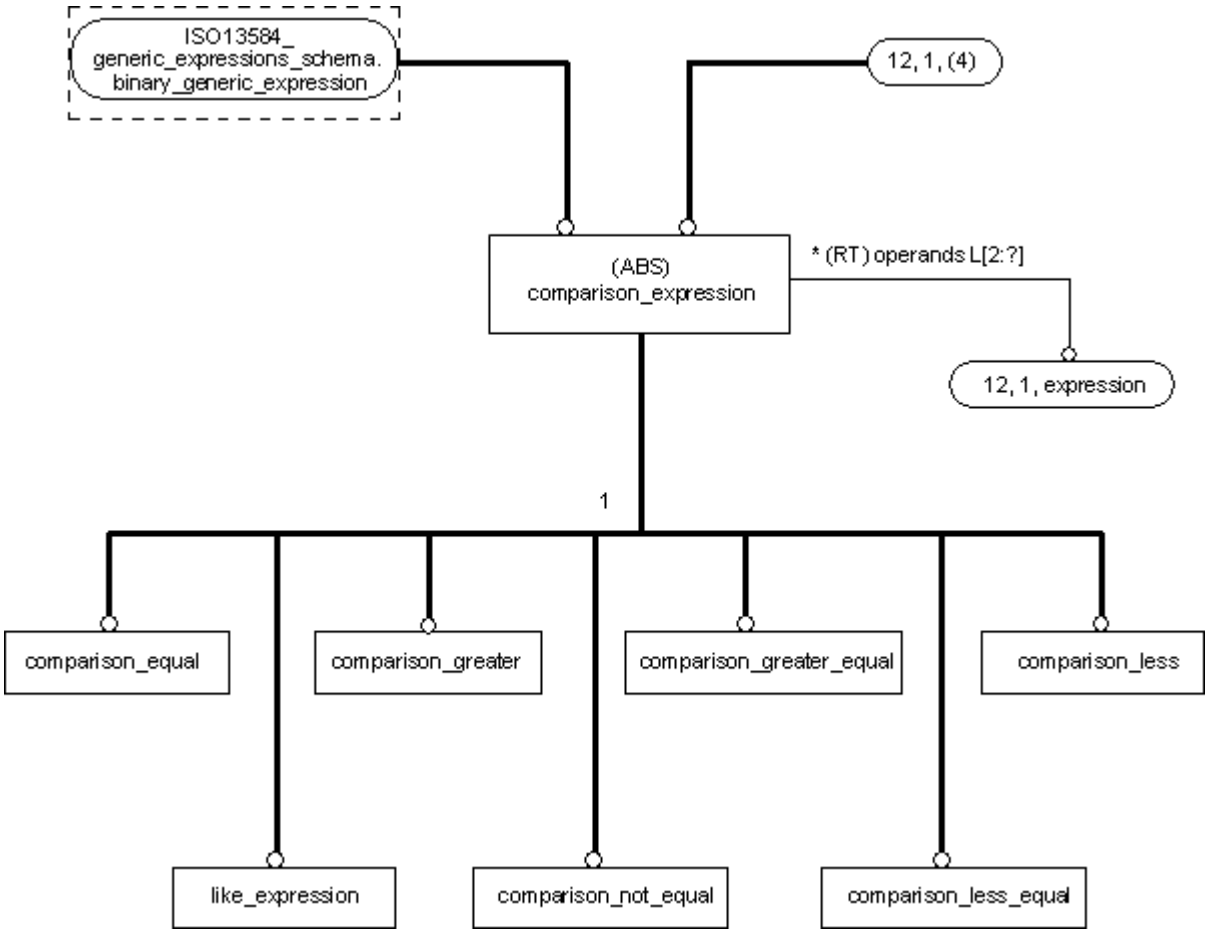

Figure C.13 — iso13584_expressions_schema - EXPRESS-G diagram 12 of 12- Comparison expressions

Annex C, Caption of Figure C.2, p61

The phrase "String typing of expressions" in the caption for Figure C.2 does not make sense. “String typing” shall be replaced by “Strong typing”.Replace with the following caption:

Figure C.2 — iso13584_expressions_schema - EXPRESS-G diagram 1 of 12- Strong typing of expressions

Annex D, Computer interpretable listings (new, to be included)

Annex D
(normative)

Computer interpretable listings

This annex references listings for the set of EXPRESS schemas documented in this part of ISO 13584. These schemas, without comments or explanatory text, are intended to be used as resources for the other parts of ISO 13584.

http://standards.iso.org/iso/13584/-20/-ed-1/tech/express

If there is difficulty accessing this site contact ISO Central Secretariat.

The following notice applies to the computer-interpretable files in this annex.

Table D.1 below, describes the URI of the schemas whose computer interpretable listings are provided by this annex.

Table D.1 — EXPRESS schemas documented in this part of ISO 13584

Description	URI
iso13584_generic_expressions_schema	urn:iso:std:iso:13584:-20:ed-2:tech:express:generic-expressions
iso13584_expressions_schema	urn:iso:std:iso:13584:-20:ed-2:tech:express:expressions

