

## 附录C 缩略语和符号

- ACID (atomicity/consistency/isolation/durability) 原子性/一致性/隔离性/持久性
- ACM (Association for Computing Machinery) 美国计算机学会
- ADT (abstract data type) 抽象数据类型
- ANSI (American National Standards Institute) 美国国家标准化组织
- ANSI/SPARC (literally, ANSI/Systems Planning and Requirements Committee; used to refer to the three-level database system architecture described in Chapter 2) 字面含义为ANSI/系统规划与需求委员会；用于指第2章中提到的数据库系统三级体系结构
- ARIES(algorithm for recovery and isolation exploiting semantics) 恢复和隔离语义的算法
- BB(same as GB) 同GB
- BCNF(Boyce/Codd normal form) BC范式
- BCS(British Computer Society) 英国计算机学会
- BLOB(binary large object) 二进制大对象
- BNF(Backus-Naur form or Backus normal form) 巴科斯范式
- CACM(Communications of the ACM (ACM publication)) ACM通信(ACM出版物)
- CAD/CAM(computer-aided design/computer-aided manufacturing) 计算机辅助设计/计算机辅助制造
- CASE(computer-aided software engineering) 计算机辅助软件工程
- CDO(class-defining object) 类定义对象
- CIM(computer-integrated manufacturing) 计算机集成制造
- CLI(Call-Level Interface (part of the SQL standard)) 调用级接口(SQL标准的一部分)
- CLOB(character large object) 字符大对象
- CNF(conjunctive normal form) 合取范式
- CODASYL(literally, Conference on Data Systems Languages; used to refer to certain prerelational (network) systems such as IDMS) 字面含义为数据系统语言会议；用于指网状数据库系统，如IDMS
- CPU(central processing unit) 中央处理器
- CS(cursor stability (DB2)) 游标稳定性(DB2)
- CWA(Closed World Assumption) 封闭世界假设
- DA(data administrator) 数据管理员(器)
- DB/DC(database/data communications) 数据库/数据通信
- DBA(database administrator) 数据库管理员
- DBMS(database management system) 数据库管理系统
- DBP&D(Database Programming & Design (magazine, now online)) 数据库编程和设计(杂志，联机)

DBTG(literally, Data Base Task Group; used interchangeably with CODASYL (in database contexts)) 字面含义为数据库任务组; 与 CODASYL同义(在数据库上下文中)

DC(data communications) 数据通信

DCO(domain check override) 域检查过载

DDB(distributed database) 分布式数据库

DDBMS(distributed DBMS) 分布式数据库管理系统

DDL(data definition language) 数据定义语言

DES(Data Encryption Standard) 数据加密标准

DK/NF(domain-key normal form) 域码范式

DML(data manipulation language) 数据操作语言

DNF(disjunctive normal form) 析取范式

DRDA(Distributed Relational Database Architecture (IBM)) 分布式关系型数据库体系结构(IBM)

DSL(data sublanguage) 数据子语言

DSS(decision support system) 决策支持系统

DUW(distributed unit of work) 分布工作单元

E/R(entity/relationship) 实体/关系

EB(exabyte (1024PB))

EDB(extensional database) 外延数据库

EKNF(elementary key normal form) 基本码范式

EMVD(embedded MVD) 嵌入式MVD

FD(functional dependence) 函数依赖

GB(gigabyte (1024MB)) 吉字节

GIS(geographic information system) 地理信息系统

HOLAP(hybrid OLAP) 混合OLAP

I/O(input/output) 输入/输出

IDB(intensional database) 内涵数据库

IDMS(Integrated Database Management System) 集成的数据库管理系统

IEEE(Institute for Electrical and Electronics Engineers) 电气和电子工程师协会

IMS(Information Management System) 信息管理系统

IND(inclusion dependence) 内含依赖

IS(intent shared (lock); information systems) 意向共享锁; 信息系统

ISBL(Information System Base Language) 信息系统(库)语言

ISO(International Organization for Standardization) 国际标准化组织

IT(information technology) 信息技术

IX(intent exclusive (lock)) 意向排它锁

JACM(Journal of the ACM (ACM publication)) ACM学报(ACM出版物)

JD(join dependence) 连接依赖

JDBC(Java Database Connectivity) 基于Java的数据库互连

K(1024 (sometimes 1000))

KB(kilobyte (1024 bytes)) 千字节  
LAN(local area network) 局域网  
LOB(large object) 大对象  
MB(megabyte (1024KB)) 兆字节  
MLS(multi-level secure) 多级安全  
MOLAP(multi- dimensional OLAP) 多维OLAP  
MVD(multi-valued dependence) 多值依赖  
NCITS(National Committee on Information Technology Standards (previously known as X3)) 国家信息技术标准委员会(以前称为X3)  
NCITS/H2(NCITS database committee (previously known as X3H2)) NCITS数据库委员会(以前称为X3H2)  
NF<sup>2</sup>(“NF squared” = NFNF = non first normal form)(?) “NF平方” =NFNF=非第一范式(?)  
ODBC(Open Database Connectivity) 开放数据库互连  
ODMG(Object Data Management Group) 对象数据管理组  
ODS(operational data store) 操作型数据存储  
OID(object ID) 对象标识  
OLAP(online analytic processing) 联机分析处理  
OLCP(online complex processing) 联机复杂处理  
OLDM(online decision management) 联机决策管理  
OLTP(online transaction processing) 联机事务处理  
OMG(Object Management Group) 对象管理组  
OO(object-oriented; object orientation) 面向对象  
OODB(object-oriented database) 面向对象数据库  
OOPL(object-oriented programming language) 面向对象的程序语言  
OQL(Object Query Language (part of ODMG)) 对象查询语言(OOMG的一部分)  
OSI(Open Systems Interconnection) 开放式系统互联  
OSQL(Object SQL) 对象SQL  
PB(petabyte (1024TB))  
PC(personal computer) 个人计算机  
PJ/NF(projection-join normal form) 投影连接范式  
PODS(Principles of Database Systems (ACM conference)) 数据库系统原理  
PRTV(Peterlee Relational Test Vehicle) Peterlee关系测试  
PSM(Persistent Stored Modules (part of the SQL standard) 持久存储模块(SQL标准的一部分)  
QBE(Query-By-Example) 按例查询  
QUEL(Query Language) 查询语言  
RAID(redundant array of inexpensive disks) 冗余廉价磁盘阵列  
RDA(Remote Data Access) 远程数据存取  
RDB(relational database) 关系数据库  
RDBMS(relational DBMS) 关系数据库系统

RID((stored) record ID or row ID) 记录标识或行标识

ROLAP(relational OLAP) 关系OLAP

RM/T(relational model/Tasmania) 关系模型/Tasmania

RM/V1(relational model/Version 1) 关系模型/版本1

RM/V2(relational model/Version 2) 关系模型/版本2

RPC(remote procedure call) 远程过程调用

RR(repeatable read (DB2)) 可重复读(DB<sup>2</sup>)

RUW(remote unit of work) 远程工作单元

RVA(relation-valued attribute) 关系值属性

S(shared (lock)) 共享锁

SIGMOD(Special Interest Group on Management of Data (ACM special interest group)) 数据管理特别兴趣组(ACM特别兴趣组)

SIX(shared intent exclusive (lock)) 共享意向排它锁

SPARC(see ANSI/SPARC) 见ANSI/SPARC

SQL((originally) Structured Query Language; sometimes Standard Query Language) 结构化查询语言;有时也理解为标准查询语言

TB(terabyte (1024GB)) 太字节

TCB(Trusted Computing Base) 可信计算基

TCP/IP(Transmission Control Protocol/Internet Protocol) 传输控制协议/网际协议

TID((stored) tuple ID) 元组标识

TODS(Transactions on Database Systems (ACM publication)) (ACM出版物)

TPC(Transaction Processing Council) 事务处理委员会

U(update (lock)) 更新锁

UDT(user-defined type) 用户定义类型

UML(Unified Modeling Language) 通用建模语言

unk(unknown (truth value)) 未知(真值)

UNK(unknown (null)) 未知(空值)

UOW(unit of work) 工作单元

VLDB(very large database; Very Large Data Bases (annual conference)) 超大规模数据库(每年一次学术会议)

VSAM(Virtual Storage Access Method (IBM)) 虚拟存储存取方法

WAL(write-ahead log) 先写日志

WAN(wide area network) 广域网

WFF(well-formed formula) 合式公式

WORM(write once/read many times) 写一读多

WYSIWYG(what you see is what you get) 所见即所得

X(exclusive (lock)) 排它锁

X3(see NCITS) 见NCITS

X3H2(see NCITS/H2) 见NCITS/H2

1NF(first normal form) 第一范式  
2NF(second normal form) 第二范式  
2PC(two-phase commit) 两阶段提交  
2PL(two-phase locking) 两阶段封锁  
2VL(two-valued logic) 二值逻辑  
2 $\phi$ C(same as 2PC) 与2PC相同  
2 $\phi$ L(same as 2PL) 与2PL相同  
3GL(third generation language) 第三代语言  
3VL(three-valued logic) 三值逻辑  
3NF(third normal form) 第三范式  
4GL(fourth generation language) 第四代语言  
4NF(fourth normal form) 第四范式  
4VL(four-valued logic) 四值逻辑  
5NF(fifth normal form (same as PJ/NF)) 第五范式

(belongs to) 属于

(comparison operator (=, <, etc.)) 比较运算符

$\emptyset$ (the empty set) 空集

(functionally determines) 函数决定

(multi-determines) 多值决定

(is equivalent to) 等价于

$\Rightarrow$  (implies (logical connective)) 蕴含(逻辑上的)

$\vdash$  (implies (metalinguistic symbol)) 蕴含(元语言符号)

$\models$  (it is always the case that (metalinguistic symbol)) 可推出(元语言符号)

```

<table expression>
::= <join table expression>
    | <nonjoin table expression>

<join table expression>
::= <table reference>
    [ NATURAL ] JOIN <table reference>
    [ ON <conditional expression>
      | USING ( <column name commalist> ) ]
    | <table reference> CROSS JOIN <table reference>
    ( <join table expression> )

<table reference>
::= <table name> [ [ AS ] <range variable name>
                  [ ( <column name commalist> ) ] ]
    | ( <table expression> ) [ AS ] <range variable name>
    [ ( <column name commalist> ) ]
    | <join table expression>

<nonjoin table expression>
::= <nonjoin table term>
    | <table expression> UNION [ ALL ] [ CORRESPONDING
      [ BY ( <column name commalist> ) ] ]
      <table term>
    | <table expression> EXCEPT [ ALL ] [ CORRESPONDING
      [ BY ( <column name commalist> ) ] ]
      <table term>

<nonjoin table term>
::= <nonjoin table primary>
    | <table term> INTERSECT [ ALL ] [ CORRESPONDING
      [ BY ( <column name commalist> ) ] ]
      <table primary>

<table term>
::= <nonjoin table term>
    | <join table expression>

<table primary>
::= <nonjoin table primary>
    | <join table expression>

<nonjoin table primary>
::= TABLE <table name>
    | <table constructor>
    | <select expression>
    | ( <nonjoin table expression> )

<table constructor>
::= VALUES <row constructor commalist>

<row constructor>
::= <scalar expression>
    | ( <scalar expression commalist> )
    | ( <table expression> )

<select expression>
::= SELECT [ ALL | DISTINCT ] <select item commalist>
    FROM <table reference commalist>
    [ WHERE <conditional expression> ]
    [ GROUP BY <column name commalist> ]
    [ HAVING <conditional expression> ]

<select item>
::= <scalar expression> [ [ AS ] <column name> ]
    | [ <range variable name> . ] *

```

```

<conditional expression>
    ::=      <conditional term>
           | <conditional expression> OR <conditional term>

<conditional term>
    ::=      <conditional factor>
           | <conditional term> AND <conditional factor>

<conditional factor>
    ::=      [ NOT ] <conditional primary>

<conditional primary>
    ::=      <simple condition> | ( <conditional expression> )

<simple condition>
    ::=      <comparison condition>
           | <in condition>
           | <like condition>
           | <match condition>
           | <all or any condition>
           | <exists condition>
           | <unique condition>

<comparison condition>
    ::=      <row constructor>
           <comparison operator> <row constructor>

<comparison operator>
    ::=      = | < | <= | > | >= | <>

<in condition>
    ::=      <row constructor> [ NOT ] IN ( <table expression> )
           | <scalar expression> [ NOT ] IN
           ( <scalar expression commalist> )

<like condition>
    ::=      <character string expression>
           [ NOT ] LIKE <pattern> [ ESCAPE <escape> ]

<match condition>
    ::=      <row constructor>
           MATCH UNIQUE ( <table expression> )

<all or any condition>
    ::=      <row constructor>
           <comparison operator> ALL ( <table expression> )
           | <row constructor>
           <comparison operator> ANY ( <table expression> )

<exists condition>
    ::=      EXISTS ( <table expression> )

<unique condition>
    ::=      UNIQUE ( <table expression> )

```