文献综述

——数据建模技术比较

摘要：本文对目前常用的数据建模技术进行比较，并概述了各种建模技术的历史背景，描述了其在实体、属性、唯一标识符、关系、子类型、关系间约束的不同表示方法，并叙述了各项建模技术的特点。

关键词：数据建模技术, 表示方法, 特点

第一章 引言

数据建模将现实生活中的业务抽象化，确定数据库需管辖的范围、数据的组织形式等直至转化成现实的数据库。数据建模为了描述数据特征、数据间关系。在实际使用中，可供选择的数据建模技术有多种，各种技术的不同点在于被描述系统的完整程度，表达方式是否直观、美观。每种建模技术都有自己的规则，主要体现在语法规则、布局规则、语义规则，而共同点是所有技术都会保证每种符号只有一种含义，每个概念只有一种符号。

对建模效果评估标准在于直观性和完整性。建模受众为用户和设计师，用户社区使用模型和说明来证实分析师是理解他们的业务场景和需求的，分析师使用的模型必须清晰且易读。意味着模型描述尽可能少地展示细节的完整内容。而系统设计者集 使用模型暗示的业务规则作为计算机系统设计的基础，聚焦在确保能描述所有可能的约束上，因为复杂，经常以可读性缺乏为代价。因此针对不同受众需求使用的建模技术才能达到更好的效果。

文中主要描述常见的 7 种数据建模技术，在实体、属性、唯一标识符、关系、子类型、关系间约束重点描述。

第二章 数据建模技术

2.1 ER 模型

Peter Chen 在 1976 年首次提出 ER 建模的概念，并建立了陈式表达法，因此 ER 模型也可称为陈氏模型。

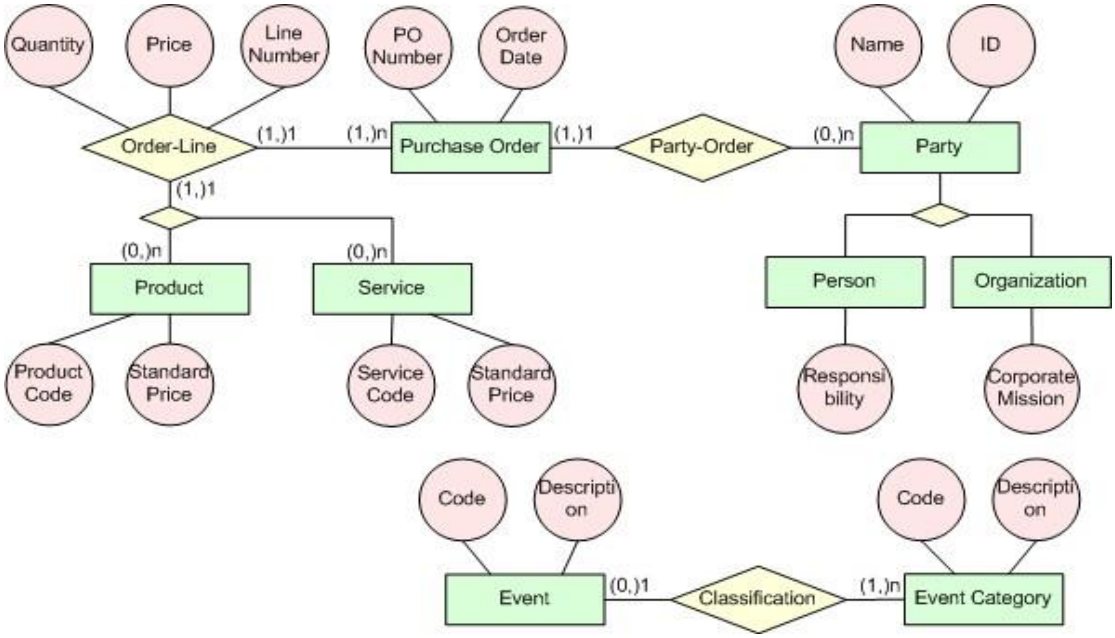

图 1 ER 模型-陈氏模型的案例

ER（Entity Relationship Model）为实体关系模型。实体（Entity）用方框表示属性（Attribute）用圆形或椭圆表示，并用线条连接到实体上。关联关系（Relationship）用菱形表示。关联关系也可以有属性。关联关系不仅是个二元关系，即，可以在两个实体间，还可以在两个以上实体间。

唯一标识符(unique identifier)：ER 模型没有解决唯一标识符，而是使用一种标记方法。关联关系名称采用 E，箭头由被依赖方指向依赖方，且依赖实体外额外加一个方框。如图 2 中由 party 指向 purchase Order，关系名称用 E 表示

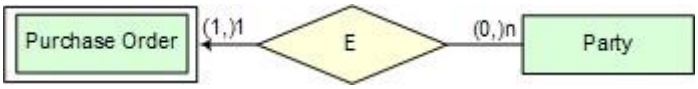

图 2 唯一标识符标记

子类型（sub-type）：陈氏表达法最初不包含子类型的描述，后来 Robert Brown 和 Mat Flavin 在陈氏表达法基础上增加了子类型。父类型（super-type）是子类型的子集，子类型发生，父类型一定发生。图 1 中，Party 是父类型，Person、Organization 是子类型。

关系间约束（Constrains between Relationship）：最初关系中的一端只存在 1 个数字。如：1 对多关系，只需要一端用 1，另一端表示 n。但是这种约束并不充分。因此模型改进后，关系的一端存在 2 个数字。

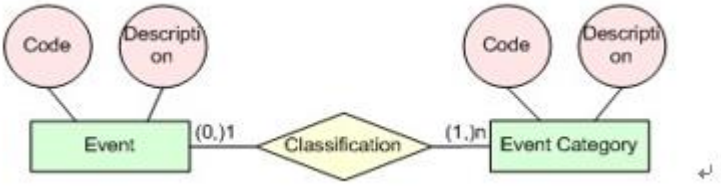

图 3 唯一标识符标记

由图 3，Event 和 Event Category 的关联关系中，Event 可以关联一个 Event Category 也可以不关联，而 Event Category 则必须关联一个或多个 Event。

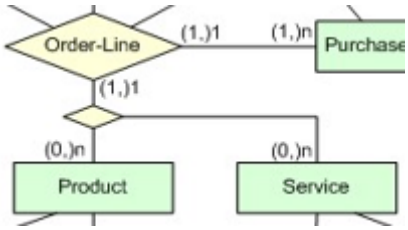

图 4 异或

由图 4，异或（exclusive or），或称互斥约束。Order Line 要么是 Product，要么是 Service，不能同时为 Product 和 Service。

2.2 EER 模型

EER（Enhanced Entity Relationship Model）为扩展实体关系模型。在陈氏表达法基础上扩展 ER 模型弥补不足。

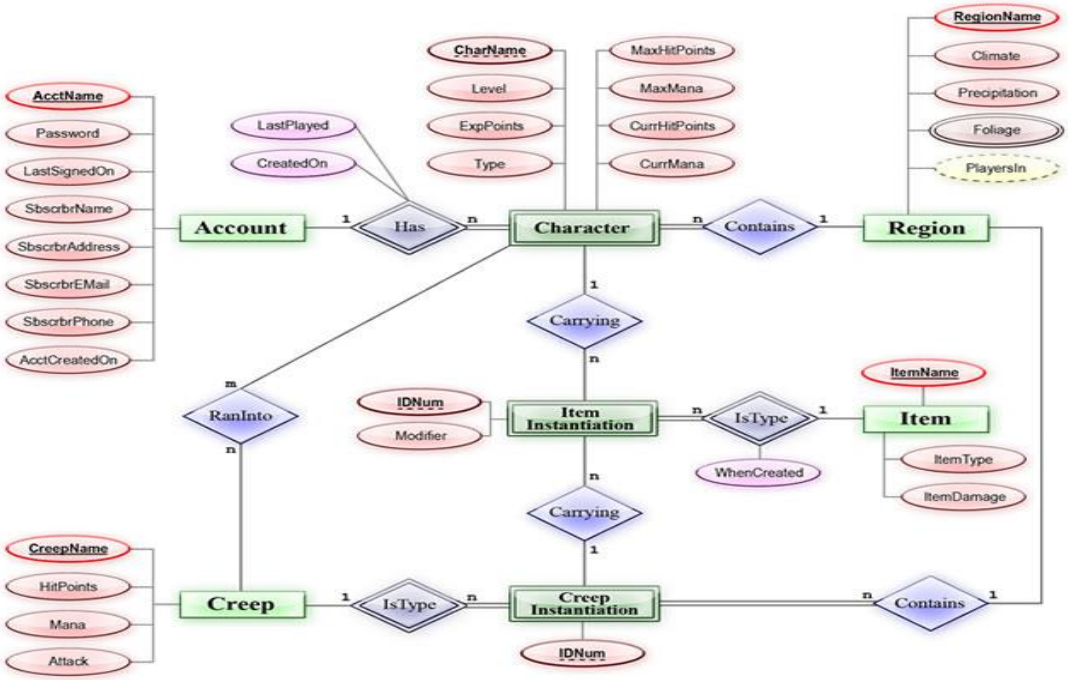

图 5 EER 模型案例

主键：在属性下加下划线。
关联关系端为 n（多数）：用两条线表示。
唯一标识符：Account has n(at least one) Character，即 Account 与 Character 为 1 对多关系，Account 的 AcctName 属性（主键）将成为 Has 的唯一标识的一员。

2.3 IE

IE（Information Engineer Relationship Model）信息工程，采用 Crows's Foot 鸭掌属性。关联关系的关联基数为多时，使用鸭掌形的三叉线来表示。

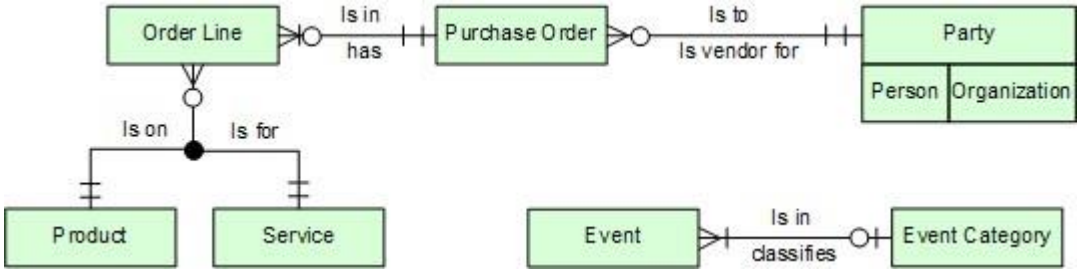

图 6 IE 模型案例

实体属性单独使用另外的文档记录，不出现在 IE 模型中。

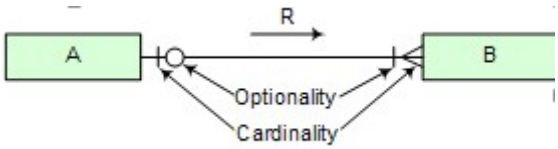

图 7 基数及可选性

图 7 中外侧表示基数（Cardinality），内侧表示可选性（optionality）。

可选性：表示关联关系是可选，还是必须。可选的关联关系表示为关联关系中的实体中的外键可以 null。必须的关联关系是指关联关系中的实体中的外键不能为 null。

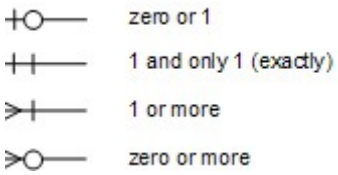

图 8 基数表示法

基数：表示关系实体的数量上限。如：1 个 A 必须关联到 1 个或多个 B，一个 B 可以关联 0 个或 1 个 A 关联。

Product 和 Service 通过一个圆连接到 Order Line。异或（exclusive or）用实心圆，Order Line 要么是 Product，要么是 Service，不能同时为 Product 和 Service。相容（inclusive or），Orderline 可以是 Product、Service 其中之一，或者同时为 Product 和 Service。

Mr Finkelstein 没有命名关联关系。Mr LineMartin 用动词命名关联关系。

Mr Finkelstein 对每个子类使用单独的实体，使用 ISA 关联关系。Mr LineMartin 将子类放在父类实体中。如图 5 所示。

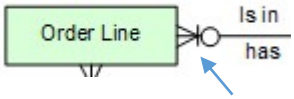

图 9 特殊符号

Finkelstein 的一个特殊符号，多了条竖线。Purchase Order 一开始有 0 或 n 个 Order Line，但最终必须有 1 或 n 个 Order。

IE 模型与 ER 模型的**区别**：Purchase Order 与 Party 是多对一 n:1 的关联关系。

IE 模型：n 被放置在了 Party 的左边

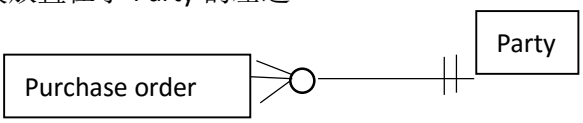

图 10 IE 模型表示多对一

等同于： (0),n (1),1

ER 模型：n 被放置在了 Purchase Order 的右边

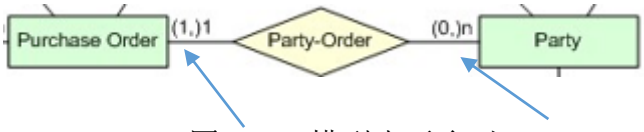

图 11 ER 模型表示多对一

因此，表示法（语法）不一样，但语义是一致的。这一点也只有 ER 模型是特殊的，其他模型表示法中都与 IE 模型一致。

2.4 Richard Barker's Notation

最早是英国咨询公司 CACI 发明，经过 Richard Barker 的推广，后来 Richard Barker 去了 Oracle，开发了相关的建模工具，因此也叫做 Oracle 表示法。

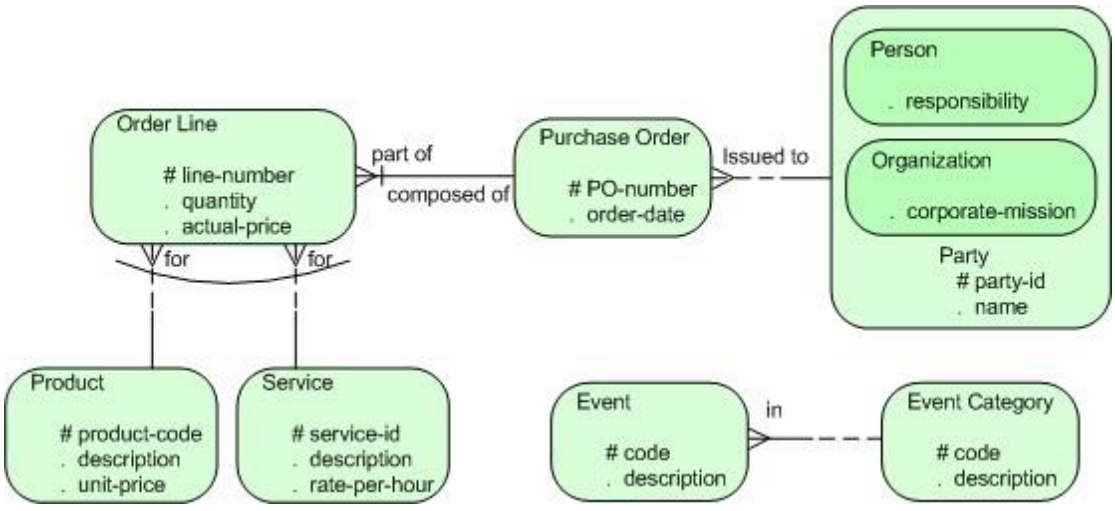

图 12 Barker's 模型案例

实体属性用圆角边框表示，属性出现在实体框中。可选属性（允许 null）在图中表示带空心圆。必须的属性（不允许 null）前面带一个实心圆。唯一标识属性前面带一个#符号。

可选项通过半边连接线的虚实线表示，若是必须关联，用实线表示。若是

可选关联，用虚线表示。Purchase Order 关联到一个 Party，所以关联线在 Party 一侧的那一半是实线表示。而 Party 关联到 0 或多个 Purchase Order，所以关联线在 Purchase Order 一侧的那一半是虚线表示。

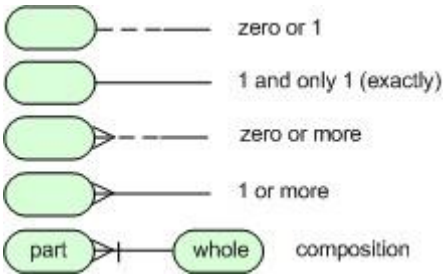

图 13 Barker's 关联基数

关联基数为 n 用三叉线表示。关联基数为 1 时线条末端没有符号。

子类型（sub-type）显示在父类型的实体框中，与 IE 中 Mr LineMartin 表示 sub-type 相同。

关系间约束（constrains between relationships）Barker 表示法仅支持 exclusive or 约束（互斥约束）。如图 14 用一条弧线划过 2 个关联关系。

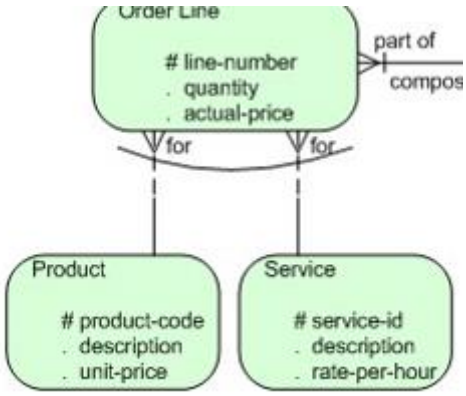

图 14 Barker's 模型 exclusive 约束

2.5 IDEF1X

是美国联邦政府广泛使用的一种模型。

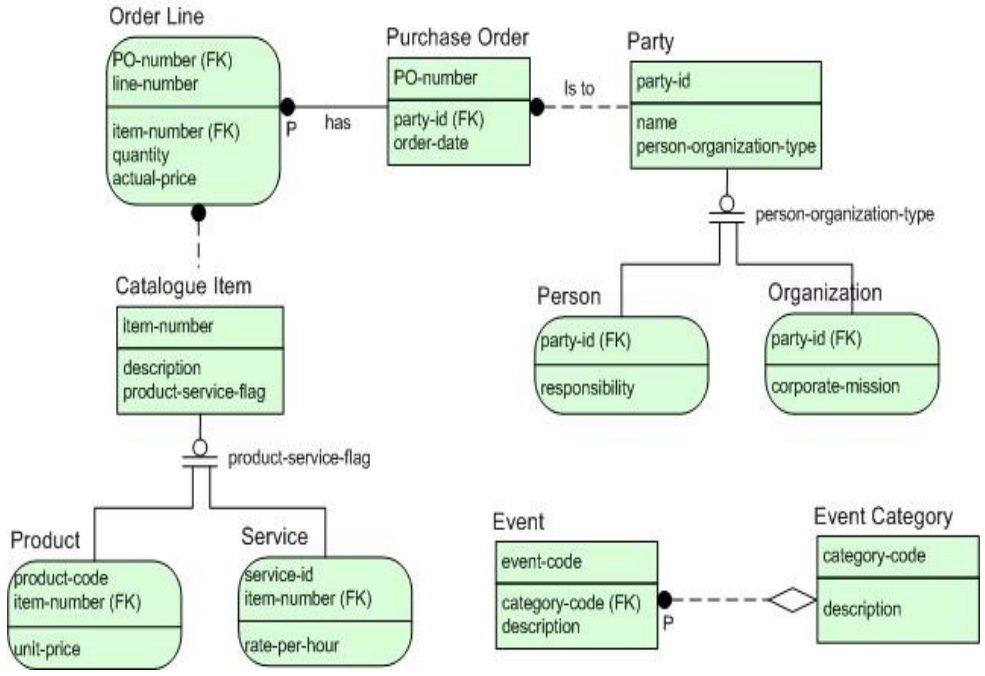

图 15 IDEF1X 模型案例

实体（Entity）分为圆角实体和方框。圆角实体是指主键不包含其他实体主键值，使用方框。非独立实体（dependent entities，主键包含其他实体主键值，使用圆角框。属性出现在实体框中，主键用线隔开。

外键必须标注外键属性 FK，而不是使用关联线表示。

如果关联关系一方的唯一标识将作为另一方唯一标识的一部分，关联线使用实线，否则使用虚线。

关联关系分为可选和必须的，在可选项一端使用一个菱形。如图 16 所示。必须的关联关系，则在可选项一端直接将关联线与实体连接。

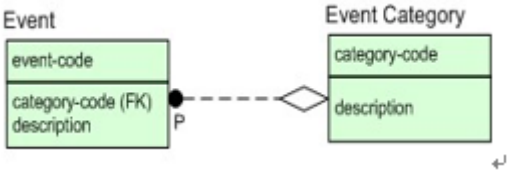

图 16 IDEF1X 模型可选关联关系

关联基数的图示都是出现在关联关系的左端或者上面。

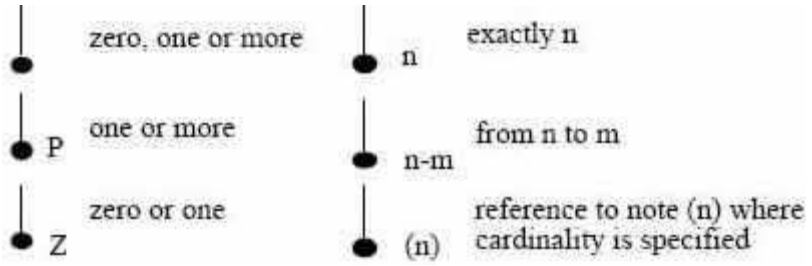

图 17 IDEF1X 模型关联基数图示

子类和关联约束图例中，如图 18 所示，圆圈下有 2 条横线，这表示模型中已经列举所有的子类和约束情况，如果模型只是部分列举子类和约束情况，则圆圈下只有 1 条横线。

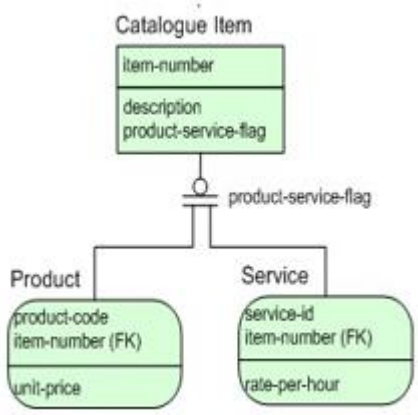

图 18 IDEF1X 模型子类和关联约束

2.6 Express-G

EXPRESS-G 是一个 ISO（International Standard Organzition 国际标准组织）标准，标准编号 ISO 10303-11。（省略了 Event、Event Category 部分）

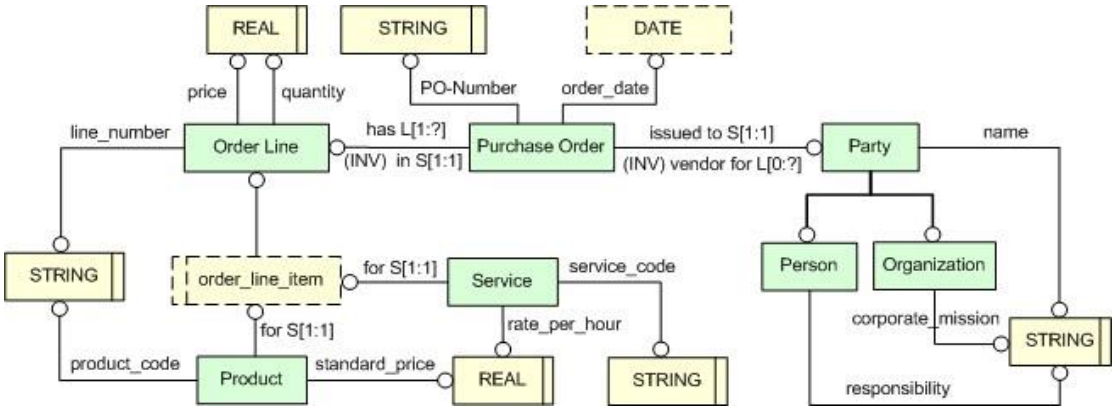

图 19 EXPRESS-G 模型案例

实体使用方框表示，实体名称出现在方框中。

属性通过空心圆结束的线条连接到属性值类型，属性名称出现在线条上。可选属性时，线条为虚线。必须属性时，需要连接到右边多一条竖线的方框表示。

EXPRESS-G 模型描述了数据类型。简单数据类型（String, Binary, Logical, Boolean, Number, Integer, Real 等）右边多一条竖线的方框表示。如图 20 所示。

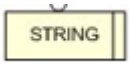

图 20 简单数据类型示例

扩展或者自定义的数据类型使用虚线框表示。如图 21 所示。

图 21 扩展或自定义数据类型示例

关联关系的名称出现在关联线上，关联基数在关联名称后面，第一个字符可以是 S、B、L、A，分别表示 Set、Bag、List、Array，后面中括号的内容即为关联基数，问号表示多。

默认情况下，关联基数都为[1:1]，且不用标注。

约束（exclusive or），EXPRESS-G 中使用 Select（可选类型）表示。

子类型（Sub-type）如图 22 中 Party、Person、Organization 所示，连接线使用粗线条

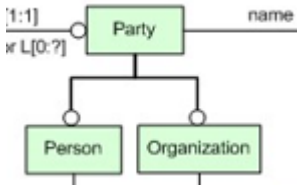

图 22 EXPRESS-G 模型子类型

2.7 UML

UML（Unified Modeling Language）统一建模语言。UML 被认为是一个对象建模技术，而不是数据建模技术。将实体称为对象类。UML 是面向对象社区中描述数据结构的标准。

UML静态图具有与其他数据建模相同的描述数据的功能。associations 就是relationships，objects就是指entity。优势在于能够描述更多的关系间的约束，包括包含、依赖、聚合、组合（符号各不相同）。

UML图除静态对象图外，还包括用例图、活动图等。

UML建模利用模型元素来组建整个系统模型，模型元素包括类、类间的关系、类的动态行为。类实例化为对象，对象有属性和操作。

UML类图，实体作为对象类，顶端包含实体名称，中间包含属性列表，低端包含实体行为。对象间不使用图形符号表示关系，而是使用行为字符。

唯一标识符在UML中很少提及，因为UML强调对象、对象属性、对象行为、对象关系，属性唯一没有太大意义。

2.8 XML

XML（The Extensible Markup Language）可扩展标记语言，不是一种建模语言，而是一种表示文本数据结构的方式。通过使用标记 tag 或标签 labels 描述数据的结构。XML 与 HTML 类似，都为超文本标记语言，浏览器（Browser）能够通过识别标签获取内容来解释 HTML 语言。

XML 能够描述数据结构，而且这种结构的数据能够在互联网上传输。另外，XML 允许用户自己定义标签。因此没有软件能够自动识别标签。但是某个行业定义一个通用标准的标签，这样对应软件能够识别标签的意义，并解释它。

XML 无法描述唯一标识和关系间约束，但可以通过设置标签清晰表达实体及属性。

第三章 总结

通过分析常见的数据建模技术的特点，并讲述了它们各自的表示法。能够帮助读者了解常见建模技术的背景、用法、特点，为在具体场景中选择合适的建模工具提供帮助。

参考文献

[1] A Comparison Of Data Modeling Techniques. David C. Hay[**C**], 1999