**用于建模参数化和约束的新ISO 10303（STEP）资源**

迈克尔·J·普拉特

电子邮件：mike@lmr.clara.co.uk

LMR系统，

21 High Street, Carlton, Bedford, MK43 7LA, UK

目录

[作者信息 4](#_Toc719417165)

[用于建模参数化和约束的新ISO 10303（STEP）资源 4](#_Toc911359545)

[1简介 4](#_Toc1366502411)

[2 ISO 10303-108 的基本原则 5](#_Toc1615900659)

[3 ISO 10303-108 模式 6](#_Toc1364175851)

[3.1 参数化-模式 6](#_Toc1168112385)

[3.2显式–约束–模式 11](#_Toc169353821)

[3.3显式-几何-约束-模式 15](#_Toc913837272)

[3.3.1固定-元素-几何-约束 15](#_Toc59987160)

[3.3.2平行-几何-约束 16](#_Toc1034367677)

[3.3.3点-距离-几何-约束 16](#_Toc737424874)

[3.3.4倾斜-直线-距离-几何-约束 16](#_Toc771730481)

[3.3.5曲线-距离-几何-约束 17](#_Toc1820449934)

[3.3.6表面-距离-几何-约束 17](#_Toc1102521929)

[3.3.7半径-几何-约束 17](#_Toc1597154387)

[3.3.8曲线-长度-几何-约束 17](#_Toc1975678456)

[3.3.9平行-偏移-几何-约束 17](#_Toc835660078)

[3.3.10角度-几何-约束 17](#_Toc395879566)

[3.3.11垂直-几何-约束 17](#_Toc643527356)

[3.3.12入射角-几何-约束 18](#_Toc1036626000)

[3.3.13同轴-几何-约束 19](#_Toc38353889)

[3.3.14切线-几何-约束 19](#_Toc368718323)

[3.3.15对称-几何-约束 19](#_Toc1558533066)

[3.3.16扫掠-点-曲线-几何-约束 19](#_Toc1407197803)

[3.3.17扫掠-曲线-曲面-几何-约束 19](#_Toc536070610)

[3.3.18曲线-平滑度-几何-约束 19](#_Toc1044843105)

[3.3.19曲面-平滑度-几何-约束 19](#_Toc704284216)

[3.3.20平行-几何-约束-带尺寸 20](#_Toc2122439695)

[3.3.21点-距离-几何-约束-带尺寸 20](#_Toc2140577195)

[3.3.22曲线-距离-几何-约束-带尺寸 20](#_Toc2034861821)

[3.3.23表面-距离-几何-约束-带尺寸 20](#_Toc1245547072)

[3.3.24半径-几何-约束-带尺寸 20](#_Toc239048148)

[3.3.25曲线-长度-几何-约束-带尺寸 20](#_Toc1887803546)

[3.3.26平行-偏移-几何-约束-带尺寸 20](#_Toc1390796844)

[3.3.27角度-几何-约束-带尺寸 20](#_Toc1910543160)

[3.4变分表示模式 22](#_Toc1323400176)

[3.5草图-模式 24](#_Toc898626053)

[4 ISO 10303-108及其配套资源在标准化过程中的地位 28](#_Toc2103067067)

[4.1 ISO 10303-108的实验验证 29](#_Toc814849096)

[5 ISO 10303-108与ISO 10303-55结合使用的示例 30](#_Toc665539553)

[6 总结 35](#_Toc1628433695)

[致谢 35](#_Toc1553514497)

[参考 36](#_Toc811970853)

ISO 10303 (STEP) 是产品生命周期数据电子交换的标准。它自被ISO于 1994 年首次发布以来，通过添加新功能不断扩展。直到现在，该标准无法交换具有参数化和约束的模型，如现代CAD系统通常生成的模型。这使得在交换之后很难出于任何目的修改模型。然而，已经开发了新的STEP资源来纠正这种情况，本文报告了第一批达到国际标准地位的STEP资源。它提供了将显式表示的参数化和约束信息与CAD模型关联的功能。主要应用领域包括许多CAD模型构造过程中固有的二维轮廓或草图、三维模型中特征的定位和定向以及装配组件的定位和定向。本文最后简要讨论了其他补充性的新STEP资源。[DOI：10.1115/1.1814383]

|  |
| --- |
| **第 1 页** |

**1简介**

产品生命周期数据电子交换的国际标准ISO 10303已在其他地方进行了充分描述(例如[1,2]）。这方面的工作始于1984年，在1994年发布ISO 10303的第一个版本之前，花了10年时间为ISO的第一个计算机可解释标准奠定基础，并在此基础上构建适当的结构。

同时，由于ISO标准化过程的性质，CAD技术出现了重要的发展，在第一次发布标准时无法将其考虑在内。特别是，在CAD系统中普遍采用程序或施工历史建模，并包含参数化和几何约束功能，使得编辑模型比在早期系统中更容易。然而，由于ISO标准的技术内容在其开发的早期阶段就被冻结，这些新功能来得太迟，无法包含在STEP的第一个版本中。其结果是，虽然目前存在的标准显然允许CAD模型的交换取得了很大的成功，但交换后收到的模型缺乏关键信息，无法在接收系统中有效修改。尽管它们在几何和拓扑上与原始系统中的模型相对应，但本质上是惰性的。因此，在CAD模型转移后，操作员现在花费大量时间试图重建“设计意图”，以便这些模型可以针对设计下游的应用进行优化。

**2 ISO 10303-108 的基本原则**

在扩展标准以此来包括上述新CAD系统功能时，应遵守的第一个基本原则是，与ISO 10303的现有零件具有向上兼容性，不得对已发布的零件进行任何更改。ISO 10303-108 [3]，作为本文主要关注的新资源部分，可以引用进使用先前STEP资源定义的模型，但不允许从此类模型引用新资源中定义的实体，因为这需要被禁止的更改。

选择的解决方案是围绕非参数模型提供“包装器”。此外层包含显式定义的参数及其与内部模型图元的连接的详细信息，以及该模型图元之间显式定义的约束关系的详细信息。然后，内部模型本身被视为参数化模型族的代表性示例，该模型族对应于包装器中定义的一组特定参数值和约束。它被称为当前结果( current result )。

在交换模型时，就其嵌入组件而言，包装器中的信息是冗余的。只有在交换完成后，该信息才会发挥作用，其作用是控制接收系统中编辑模型的方式。首选的编辑方法是只更改参数值。包装器提供了可用于此目的的所有显式定义参数的详细信息。如果其中一个或多个模型发生了更改，则必须根据包装器中也存在的约束重新生成嵌入模型。

ISO 10303-108的大部分内容都与包装器的功能有关，但也提供了一个附加模式，规定了CAD系统中广泛使用的二维轮廓或草图的表示形式，作为构建三维体积元素的基础，通常通过使用扫掠操作，例如围绕轴拉伸或旋转。以下各节详细描述了构成ISO 10303-108的五个模式的内容。

ISO 10303中使用的信息建模语言称为EXPRESS，它是标准本身的一部分[4]。理解本文给出的示例不需要详细的EXPRESS知识；只需知道该语言提供了超类型和子类型，并且这些超类型和子类型以类似于面向对象建模的方式展示了属性的继承就足够了。EXPRESS中建模的实体数据类型及其属性在下文中用粗体表示。此外，还应注意，ISO 10303-108经常将新实体数据类型定义为本标准其他资源部分中定义的实体的子类型。这是“标准”实践，与以下事实相关： 如ISO 10303“（标准的）综合通用资源和综合应用资源规定了单一概念产品数据模型”的每个已发布部分的前言中所述。

|  |
| --- |
| **第2页** |

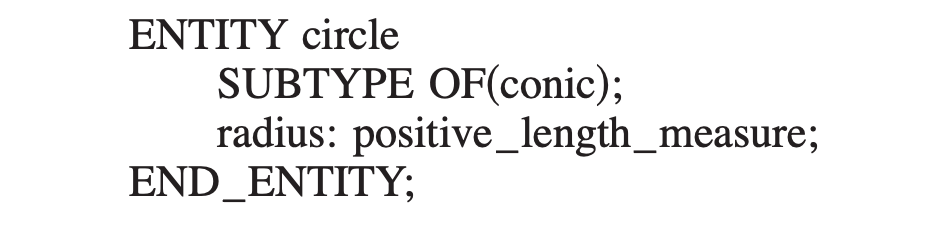
**3 ISO 10303-108 模式**

**3.1 参数化-模式**

**parameterization\_schema**涉及参数的定义及其与嵌入模型或当前结果中的数量的关联。事实上，“参数”一词在ISO 10303中已经有几种不同的含义，因此ISO 10303-108中的基本实体数据类型被命名为模型-参数（**model\_parameter**），以确保与其他用法有明显区别。它是一个抽象的或不可实例化的超类型(即，它的任何实例都不能出现在模型交换文件中)，但具有下面描述的可实例化子类型。模型-参数（**model\_parameter**）的子类型的实例被视为一个数学变量；它有一个有效域和一个当前值。定义了两个可实例化的子类型：绑定–模型–参数（**bound\_model\_parameter**）和未绑定–模型–参数（**unbound\_model\_parameter**）。

**绑定–模型–参数（bound\_model\_parameter）的实例与嵌入的当前结果模型中实体数据类型实例的属性相关联。其当前值是该属性的值。通过实体数据类型attribute\_reference（属性\_引用）进行必要的关联或绑定，该实体数据类型attribute\_reference（属性\_引用）本身有两个属性。第一个是**ISO 10303-43[5]**中定义的类型representation\_item（表示-项目），它允许它指向在构建当前结果时使用的元素的任何实例。第二个是一个专用文本字符串，它引用绑定到的指定实例的属性名称。**

**例如，考虑**ISO 10303-42 [6] **实体数据类型circle。具有**EXPRESS**定义**



这一点的意义是不言而喻的。圆的实例将以以下形式出现在ISO 10303-21交换文件 [7] 中



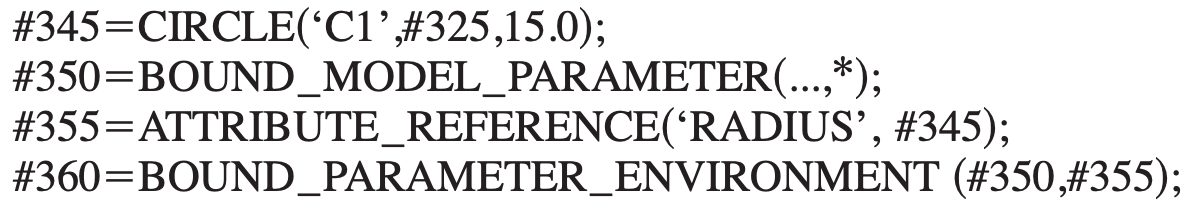
初始编号是文件中实例的参考编号。括号内的前两个值是**circle**从其超类型继承的属性值；第一个只是一个文本名称，第二个是对文件中由实例 #325表示的**axis\_placement**（轴-位置）或局部坐标系的引用（未显示）。第三个属性值15.0是圆实例的“半径”属性的值。

要将绑定–模型–参数（**bound\_model\_parameter**）与圆实例的半径属性相关联，必须首先创建**attribute\_reference（属性\_引用）**的实例。如上所述，这有两个属性；第一个是属性名（一个文本字符串），第二个是IISO 10303-43**类型representation\_item（表示-项目）**。在本例中，**attribute\_reference（属性\_引用）**实例将作为



出现在交换文件中，这里第一个属性是参数绑定到的**circle**属性的名称，第二个属性是对交换文件中指定的**circle**实例的引用。这是一个有效的引用，因为**circle**是**representation\_item（表示-项目）**的子类型。

最后，参数和属性之间的链接是通过实体数据类型**bound–parameter–environment（**绑定–参数–环境）的实例来实现的（该名称并不直观地表明其重要性，但为了与以前存在的资源兼容，它是必需的）。这个实体有两个属性；第一个是指一个变量，第二个是指它的解释（在某种意义上）。在本例中，变量是**bound\_model\_parameter（**绑定-模型-参数）实例，解释是**attribute\_reference（属性\_引用）**实例。下面的交换文件片段说明了这一原理。**circle**实例首先出现。接下来是**bound\_model\_parameter（**绑定-模型-参数）、**attribute\_reference（属性\_引用）**和**bound–parameter–environment（**绑定–参数–环境）的实例，最后一个实例提供了参数和**circle**实例的引用属性之间的关联。



**bound\_model\_parameter（**绑定-模型-参数）的一些属性被省略，包括名称、对有效域的引用和文本描述。显示的一个属性（用星号表示）表示参数的值，在实践中无法明确给出该值。实际值当然是15.0，即圆的半径属性的值，但EXPRESS语言无法提取该值并将其写入 #350中的适当位置。上述文件片段中的星号表示该值为“派生”——派生过程由翻译软件负责。

在上面给出的示例中，**circle**实例将属于嵌入的模型或当前结果，但**model\_parameter**（模型-参数）实例将属于围绕它的包装器。它提供了有关模型传输到接收系统后修改模型的有效方法的辅助信息。

顾名思义，未绑定–模型–参数（**unbound\_model\_parameter**）的实例未绑定到实体实例属性。相反，它参与了与其他参数的数学关系，其中一些可能是绑定-模型-参数（**bound\_model\_parameter**）的实例。在下一个要描述的模式中，即显式–约束–模式（**explicit\_constraint\_schema**）中规定了数学约束关系的规范。

为了说明未绑定–模型–参数（**unbound\_model\_parameter**）实例的使用，我们考虑一个矩形块。假设x、y和z表示与块的长度、宽度和高度关联的绑定模型参数。现在我们可以引入另一个参数t > 0，并定义关系x = 3t - 2， y = t2和z = t2 - 3。新参数可以由一个未绑定–模型–参数 （**unbound\_model\_parameter**）的实例表示。它不绑定到模型中的任何物理量，但用于控制绑定-模型-参数（**bound\_model\_parameter**）的三个实例的值。绑定和未绑定**model\_parameter**（模型-参数）之间的主要区别是，出于上述原因，第一个参数的值必须作为“派生”写入交换文件，但可以为第二个参数提供显式值。当然，在实际的模型交换中，这应该与它通过绑定-模型-参数实例的关系控制的属性值兼容。

应该注意的是，尽管**model\_parameter**（模型-参数）的一个属性是一个名称，但该名称在ISO 10303模型中几乎没有意义。如上所示，在文件交换中，参数实例是通过其在交换文件中的引用号而不是名称来引用的。在模型传输过程中，接收系统将为构成模型的元素生成自己的内部标识符。传输发送系统生成的内部系统标识符没有什么意义，因为该系统的命名约定与接收系统的命名约定不同。然而，参数的name属性可用于捕获和传输发送系统的用户与其关联的任何描述性名称。

**parameterization\_schema（**参数化-模式）中定义的另一种实体数据类型是**fixed\_instance\_attribute\_set**（固定的实例属性集）。这只是如上所述的**attribute\_reference（属性\_引用）**的一组实例。**fixed\_instance\_attribute\_set**（固定的实例属性集）实例的出现只是一个断言，即它所引用的单个属性集都具有固定值。等效地，其有效范围的域具有零范围。实际上，这是向接收系统发出的指示，指示属性的值在传输后对模型的任何修改中应保持不变。

|  |
| --- |
| **第 3 页** |

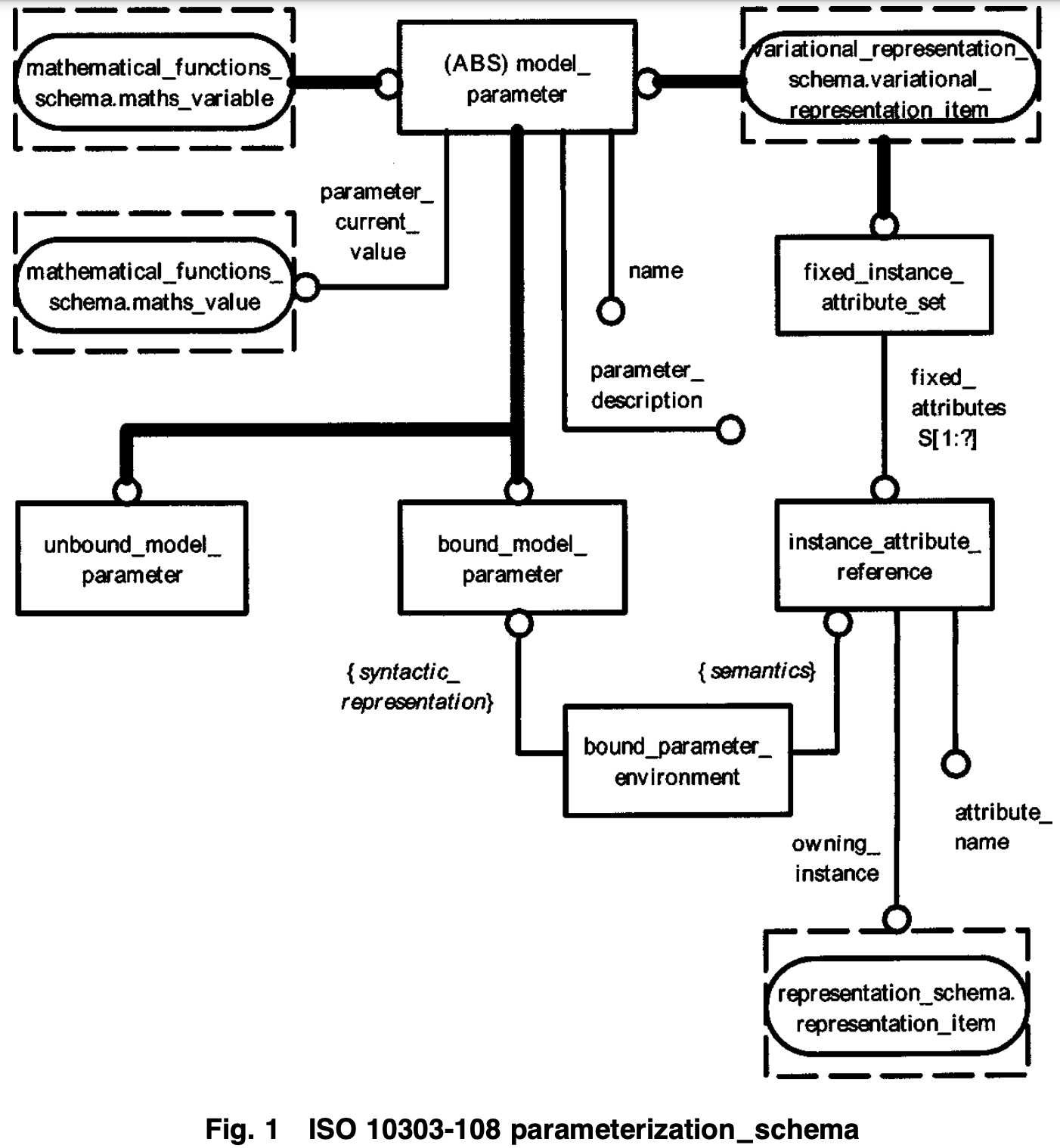


图1总结了ISO 10303-108 **parameterization\_schema（**参数化-模式），它使用EXPRESS语言的EXPRESS-G图形表示。EXPRESS-G在 [4] 中有完整定义。该图表并未充分利用其所有功能，为了清晰起见，对其进行了一些细微的简化。其解释如下：

* 矩形框表示实体数据类型。在抽象（不可实例化）情况下，实体名称前面有 “(ABS)”。
* 粗线表示子类型关系，子类型由链接末端的小圆圈区分。
* 细线显示属性关系，属性类型也类似地通过一个小圆圈来区分。属性名称显示在连接旁边。L[x:y] 或S[x:y]表示属性值是所指示类型的列表或一组实例，并给出聚合成员数的上限和下限（“ ? ”用于不确定的上限）。
* 连接终止于小圆圈但不指向实体框的属性本质上是基本类型，如实数、整数、布尔值或字符串。
* 从其他模式导入的实体数据类型由带有实体名称的虚线框表示，前面是定义该实体的模式的名称，在一个封闭的圆框中。其中一些模式包含在ISO 10303-108中，本文对此进行了描述；在其他情况下，被引用实体的名称基本是不言自明的。
* 指定EXPRESS实体的代码可能包含对这些实体的实例的有效性施加限制的规则，但这些规则在EXPRESS-G图表中没有表示。
* 简化的链接以斜体属性名称显示（但请注意，这不是标准的EXPRESS-G实践）。

**3.2显式–约束–模式**

在CAD模型中，约束可以以两种形式出现：

* 隐式约束内置于构造操作的操作中。示例：在系统屏幕上创建矩形的操作会自动（*automatically*）创建一个四边形，其中对边平行，相邻边垂直。
* 相比之下，显式约束是系统用户故意添加的约束。示例：用户可以通过选取四个角点来创建四边形，然后应用两个平行度约束作为相对边之间的关系。然后，可以应用角度约束将生成的平行四边形的一个角固定为60度。CAD模型中指定的大多数尺寸都是显式约束的情况。

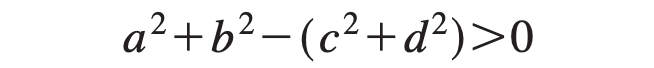
顾名思义，ISO 10303-108 **explicit\_constraint\_schema** (显式-约束-模式)只涉及显式约束。隐式约束由本文后面提到的过程表示功能处理。定义了一个抽象（不可实例化）超类型**explicit\_constraint**（显式-约束），它有两个子类型：**free\_form\_constraint**（自由形式约束）和**defined\_constraint**（定义约束）。

实体数据类型**free\_form\_constraint（**自由-形式-约束）同样是抽象的；它是根据**model\_parameter**（模型-参数）子类型（以下简称“参数”）的实例之间的显式数学关系定义的所有约束的超类型。它的可实例化子类型是

* **free\_form\_assignment**（自由形式赋值）——他的实体数据类型将一个或多个参数的值指定为等于包含其他参数的数学表达式的值。这种类型的约束将用于根据上一节给出的**unbound\_model\_parameter（**未绑定-模型-参数）示例中的未绑定参数t，对x、y和z表示的参数的赋值进行建模。
* **free\_form\_relation（**自由-形式-关系）——此实体数据类型指定两个或多个参数参与的数学方程或不等式。这些参数分为两个子类；一个集合被视为提供输入值，而另一个输出值则是第二个子类的成员，其值根据第一个子类成员的值受到约束。

|  |
| --- |
| **第 4 页** |

作为使用**free\_form\_relation（**自由-形式-关系）约束的一个例子，考虑以下由a、b、c和d表示的参数之间的关系：



此处，a和b可以是输入或参考参数(reference parameters)，以及用于控制c和d值的约束。然而，a和b本身可以是其他约束关系中的输出或约束参数，这为模型中的约束层次结构提供了可能性。

这些约束中使用的数学关系的表示由资源ISO 10303-50（“数学构造”）[8]定义。这种关系并不像在编程语言中那样以字符串的形式表示，而是以单个算术和关系运算符、标准函数和操作数构建的树结构表示。

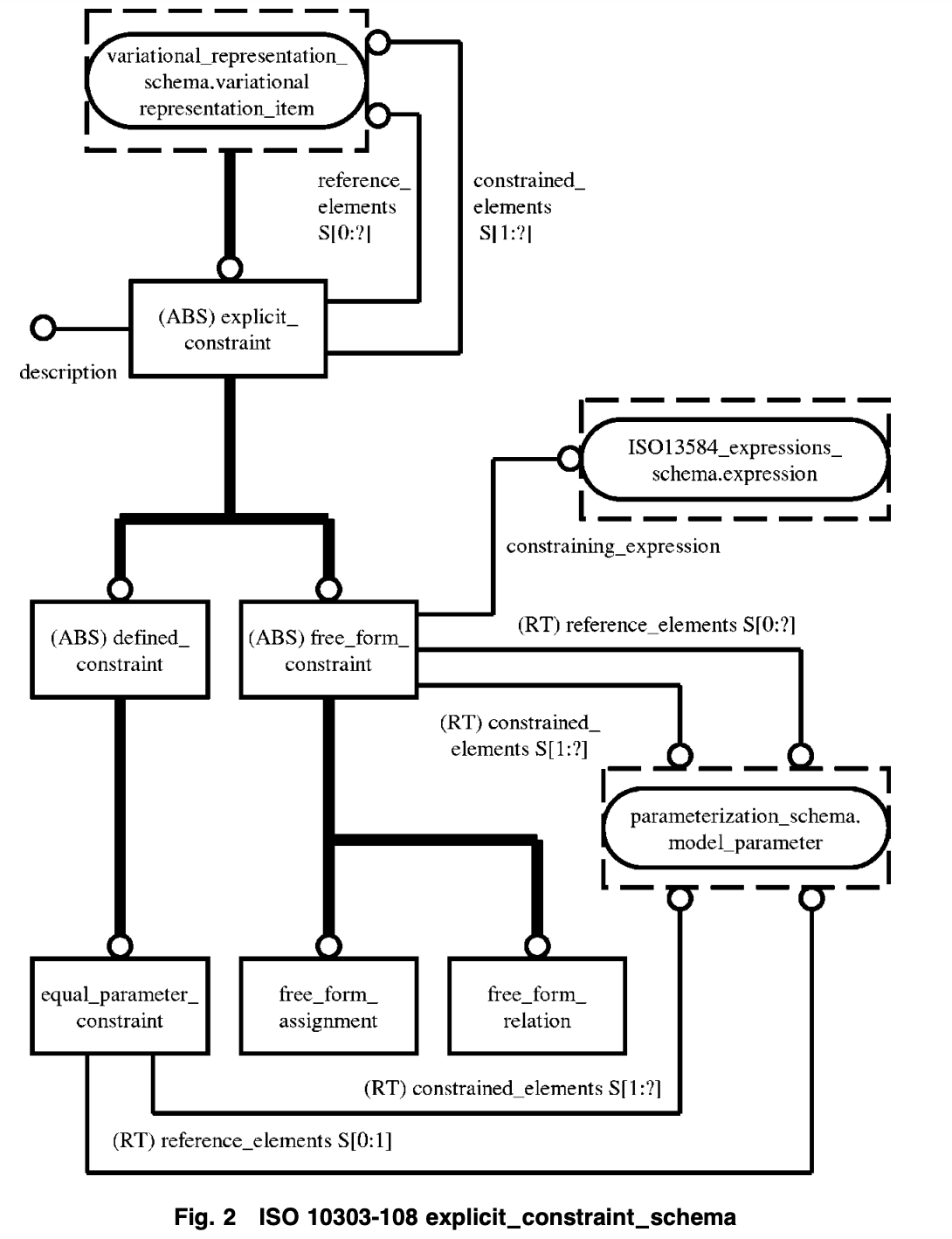
实体数据类型**explicit\_constraint**有第二个子类型**defined\_constraint**。这也是抽象的，在**explicit\_constraint\_schema（**显式-约束-模式）中只有一个可实例化的子类型。然而，在下一个要讨论的模式中，它还有另一个非常重要的子类型，即**explicit\_geometric\_constraint\_schema（**显式-几何-约束-模式）。

**defined\_constraint（**定义–约束）是指未定义明确数学关系的约束。它用于发送和接收系统都能很好地理解约束语义的情况。在这种情况下，以描述性形式发送约束就足够了，允许接收系统以最适合其内部功能的任何方式对其进行数学表述。下一节将给出一个示例。当前模式中提供的**defined\_constraint**的单个可实例化子类型是**equal\_parameter\_group**，它只列出一组参数，这些参数的值都被约束为相等。这是一个非常普遍的要求，以至于人们觉得没有必要详细说明，哪怕是一个简单的数学关系，如相关参数值之间的相等。

最后，**explicit\_constraint\_schema**提供了构造**simultaneous\_constraint\_set**（同步约束集），它允许将需要在模型中同时保持的一组约束组合在一起。从数学上讲，这意味着需要同时解决接收系统中的一组约束关系。如果这些关系中有一个或多个是非线性的，那么这样的系统通常会有多个解。设计师在发送系统中选择的解决方案将通过前面提到的嵌入式或当前结果模型的配置来表示。

应注意，ISO 10303-108不包含约束关系顺序应用的表示，除非本节前面提到的约束层次系统暗示了此类顺序。序列通常意味着时间依赖性，它不在ISO 10303-108的范围内，但在补充资源ISO 10303-55（“程序和混合表示”）的范围内[9]，本文稍后简要讨论。

ISO 10303-108 **explicit\_constraint\_schema（**显式-约束-模式）的结构在图2中以EXPRESS-G形式示出。图1的说明性注释适用，并且在该图中有一个附加特征，即在超类型级别上进行的属性定义在某些情况下在子类型级别上进行了细化。这在重新定义的属性名称之前用“(RT)”表示，代表“重新定义类型”的缩写。



**3.3显式-几何-约束-模式**

前一节中描述的约束功能具有非常普遍的适用性，并且可以在任何类型的模型中使用。然而，一个非常重要的应用领域是由CAD系统生成的形状模型，其中在建模元素之间可以应用多种几何约束。**explicit\_geometric\_constraint\_schema（**显式-几何-约束-模式）指定了此类型的一系列约束。此类约束在CAD系统中广泛实施，因此无需对其进行数学定义，因为此类约束的语义对于发送和接收系统来说是通用的，因此描述形式的表示就足够了。因此，使用先前模式中定义的**defined\_constraint（**定义–约束）的子类型是合适的。

在最高级别，**explicit\_geometric\_constraint\_schema**定义了一个抽象超类型**explicit\_geometric\_constraint**，它是**defined\_constraint（**定义–约束）的子类型。此类型的所有约束共享两个属性，一个指定一组一个或多个约束元素（*constrained elements*），另一个指定一组零个或多个引用元素（*reference elements*）。这些元素必须是ISO 10303-42类型 **geometric\_representation\_item**（几何-表示-项），其中包括作为子类型的点、曲线和曲面，以及作为形状模型基础的单个几何元素。

如果引用元素存在于此类约束的实例中，则称该约束为定向（*directed*）约束；如果不是，它就是无向（*undirected*）的。举例来说，考虑两条线被约束为平行的情况。定向情况可表述为“线A平行于线B”，在这种情况下，线A相对于线B（引用元素）受到约束。无向情况对应于“线A和线B平行”，在这里，两条线在约束中扮演相同的角色；没有引用元素。这两种情况也可以被描述为不对称（*asymmetric*）和对称（*symmetric*），因为两条线在第一种情况下扮演不同的角色，但在第二种情况下扮演相同的角色。

**explicit\_geometric\_constraint\_schema**定义了**explicit\_geometric\_constraint**的一组特定子类型，所有子类型都继承了上面讨论的两个属性。以下段落简要介绍了第一级子类型。在提到“基本形式”的每一种情况下，都存在一个具有修改语义的二级维度子类型。这些进一步的子类型将在后面描述。

3.3.1固定-元素-几何-约束

没有引用元素。受约束元素可以属于任何类型的几何表示项，其中包括高级构造，如实体模型、曲面模型和线框模型。模型传递后，受约束图元集的成员将被冻结，不可编辑。

3.3.2平行-几何-约束

可能存在零个或一个参考元素。受约束图元和参照图元是直线或平面。在无向情况下，约束元素集的成员彼此平行；在定向情况下，它们都与参考元素平行。

3.3.3点-距离-几何-约束

受约束的图元是点。它们与最多四个参考图元（可能是点、曲线或曲面）的距离相同（未指定）。例如，此约束可用于约束一个点，使其与四面体的四个面等距。在此基本形式中，约束只能作为定向约束实例化。

3.3.4倾斜-直线-距离-几何-约束

所涉及的元素是两条不相交的线，其中一条可能是参考元素。线之间的距离是沿着它们的公共法线指定的。

|  |
| --- |
| **第 5 页** |

3.3.5曲线-距离-几何-约束

受约束的图元是曲线。在基本形式中，对单个曲线进行约束，使其与最多三个参考图元的最小距离相等。

3.3.6表面-距离-几何-约束

受约束的图元是曲面。在基本形式中，约束单个曲面，使其与最多三个参考图元的最小距离相等。

3.3.7半径-几何-约束

这是一个无向约束，没有引用元素。它将一组元素的半径限制为相等，这些元素是以单个半径属性为特征的任何几何类型，例如ISO 10303-42中定义的圆、圆柱面或球面。

3.3.8曲线-长度-几何-约束

这是一个无向约束，没有引用元素。它限制一组元素的长度相等，这些元素是ISO 10303-42中定义的几何类型有界曲线。

3.3.9平行-偏移-几何-约束

此约束用于近似计算的偏移曲线或曲面。它可用于约束平面中的一组二维曲线、一组三维曲线或一组曲面。这些图元类型不能在约束的单个实例中混合。约束可以无向形式使用，以要求一组受约束元素的所有对都具有指定的偏移关系。在其定向形式中，约束要求一组受约束图元与单个参照图元具有相同的偏移关系。

3.3.10角度-几何-约束

此类约束定义直线和/或平面之间的关系。在其基本形式中，它是一个定向约束，要求一组此类元素的所有成员与参考元素形成相等的角度。

3.3.11垂直-几何-约束

这里涉及的元素是直线或平面，约束可以是定向的，也可以是无向的。在定向情况下，要求一组受约束图元的构件与一个或两个相同类型的参考图元垂直（仅在三维情况下允许两个参考图元）。在无向情况下，受约束的图元可能有两个或三个（仅在三维情况下允许有三个），它们必须都是直线或平面，但不能是两者的混合体。

3.3.12入射角-几何-约束

这种类型的约束要求一个或多个点、曲线或曲面位于（或关联在）任何这些类型的一个或多个参照图元上。还将介绍反向情况，其中参考图元包含在一个或多个受约束图元中。还有一种情况是无向的，即两个元素中的一个元素必须与另一个元素发生关联。示例：受约束图元可能是一组点，所有点都需要位于指定的参考曲线上。或者，这些点可能是参考图元，并且曲线受约束以对其进行插值。

|  |
| --- |
| **第 6 页** |

3.3.13同轴-几何-约束

此约束要求一组具有轴对称几何图形的图元共享同一轴。在定向情况下，使用单个参考元素来定义该轴。

3.3.14切线-几何-约束

此约束要求两个曲线或曲面图元彼此相切（无向关系），或一个或多个曲线或曲面与一个或多个相同类型的指定参照图元相切（有向关系）。

3.3.15对称-几何-约束

此约束要求两个点、曲线或曲面相对于参考图元对称布置。在2D中，参考元素是线，在3D中，参考元素可以是线或平面。

3.3.16扫掠-点-曲线-几何-约束

此约束与扫描二维轮廓或草图以生成三维形状有关。原始轮廓中的每个顶点扫出三维图形的边曲线。该约束要求，如果修改轮廓，则根据指定的扫掠运动类型，适当地重新计算由其顶点运动扫掠的边下方的曲线。

3.3.17扫掠-曲线-曲面-几何-约束

此约束还与扫描二维轮廓或草图以生成三维形状有关。原始轮廓中的每条边曲线扫掠出三维图形的一个面。该约束要求，如果修改轮廓，则根据指定的扫掠运动类型，适当地重新计算由其顶点的运动扫掠出的面下面的曲面。

3.3.18曲线-平滑度-几何-约束

ISO 10303-42允许对组成复合曲线的各个曲线段之间的每个过渡段的平滑度进行规范。按照增加平滑度的顺序，最简单的可能性包括位置连续性、切线连续性和曲率连续性。ISO 10303-42功能允许在模型传输时指示平滑度，但不限制在接收系统中编辑接收的模型时保持平滑度。曲线–平滑度–几何–约束提供了此附加功能。

3.3.19曲面-平滑度-几何-约束

ISO 10303-42还允许对构成复合表面的各个曲面片之间的每个过渡的平滑度进行规范。同样，ISO 10303-42功能允许在模型传输时指示平滑度，但不限制在接收系统中编辑接收的模型时保持平滑度。曲面–平滑度–几何约束提供了此附加功能。如前所述，上述几种约束类型具有标注子类型。下文给出了这些建议。

3.3.20平行-几何-约束-带尺寸

此平行-几何-约束子类型用于指定平行线和/或平面之间的尺寸。

3.3.21点-距离-几何-约束-带尺寸

点-距离-几何-约束的子类型用于指定相对于其他几何元素定位一个或多个点的标注。

3.3.22曲线-距离-几何-约束-带尺寸

此“曲线-距离-几何-约束”子类型用于指定相对于其他几何元素定位一条或多条曲线的标注。

3.3.23表面-距离-几何-约束-带尺寸

此曲面-距离-几何-约束的子类型用于指定相对于其他几何元素定位一个或多个曲面的标注。

3.3.24半径-几何-约束-带尺寸

此“半径-几何-约束”子类型约束一组圆形、圆柱形或球形图元的成员具有相同的指定半径。

3.3.25曲线-长度-几何-约束-带尺寸

此“曲线-长度-几何-约束”子类型将约束一组有界曲线的所有成员具有相同的指定长度。

3.3.26平行-偏移-几何-约束-带尺寸

此平行-偏移-几何-约束的子类型要求偏移曲线或曲面位于距其偏移的图元的指定距离处。

3.3.27角度-几何-约束-带尺寸

此“角度-几何-约束”子类型约束直线或平面图元之间的角度值。上面简要描述的所有约束都具有维度属性，该属性要求与约束中涉及的所有几何元素的维度一致。这些约束的主要应用领域是

•二维剖面或草图在这种情况下，所有几何元素以及它们之间的任何约束将具有尺寸2。

•特征间关系这包括一个特征相对于另一个特征的位置和/或方向受到约束的情况。在这种情况下，约束将是三维的，通常应用于与相关特征相关的基准或参考元素之间。

•装配建模在这种情况下，用于在装配模型中定位和定向不同零件模型几何元素之间的约束通常具有尺寸3。但是，某些图解表示法（例如齿轮系）可能使用2D表示法。

ISO 10303-108显式几何约束模式的结构如图3所示，使用EXPRESS-G表示法。为清楚起见，省略了显式-几何-约束的各种子类型属性的详细信息，但在上述第3.3小节中进行了说明。1–3.3.27.应注意，ISO 10303-108中实际规定的尺寸子类型的名称包含如图所示的缩写，尽管这些名称在上述说明中已完整拼写。

Bettig和Shah[10]根据ISO 10303-42[6]中提供的几何定义发布了一套非常全面的几何约束规范。然而，ISO 10303-108中采用的方法仅提供最常用的几何约束类别。如果需要其他功能，可采用ISO 10303-108中更通用的功能。例如，没有提供约束环形表面小半径的直接方法。这种效果可以通过将模型参数与环形曲面实例（定义见ISO 10303-42）的小半径属性相关联，然后以所需的任何方式约束参数值来实现。

|  |
| --- |
| **第 7 页** |

**3.4变分表示模式**

变分模型的特点是存在明确表示的变量参数和约束，如前面所述。这种模型可被视为代表一系列相关的非变异模型。ISO 10303-108将“当前结果”与变分模型相关联，即与所有参数的当前值相对应的代表族成员。根据任何强制约束，通过改变相关参数值，可以从中导出族的不同成员。

当前结果可能是一个以其组成元素表示的显式模型，也可能是一个以构造操作序列定义的程序模型（见第4节）。对于程序模型，规定的操作通常需要提供支持要素；如果这些文件显式包含在exchange文件中，则可以对它们施加显式约束。

通过一个例子，考虑两个圆柱体的布尔联合。基本操作是创建两个圆柱体和联合程序。但创建每个圆柱体需要指定一条线（轴）和一个点（定义一个平面的位置），以及定义圆柱体半径和长度的两个数值参数。轴线通常在交换文件中作为标准ISO 10303-42线实例提供，在这种情况下，可以在轴线之间施加（例如）垂直度约束。

“表示”一词在ISO 10303文件中用作“模型”的同义词ISO 10303表示的属性之一是表示项实例列表，这些是表示或模型所包含的顶级元素。这些实体数据类型在ISO 10303-43“表示结构”[5]中定义。顶级表示项通常引用较低级别的其他支持项，因此仅由表示本身间接引用，但构成其规范中的基本元素。

|  |
| --- |
| **第 8 页** |

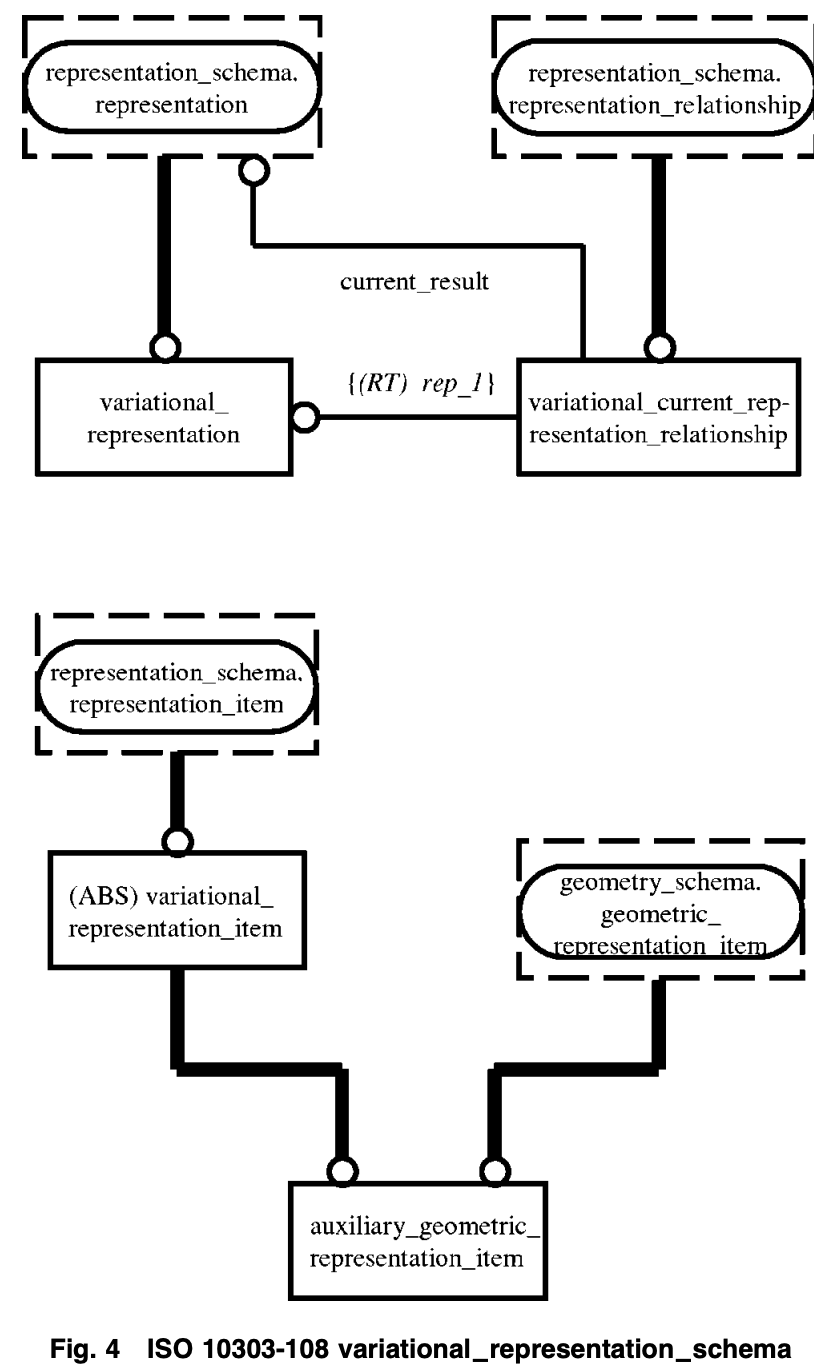
变分表示模式定义了当前结果模型与其控制的变分信息之间的关系。变分表示和当前结果之间的关系实际上非常简单，第二个结果完全包含在第一个结果中。或者，关于显式定义的参数和约束的信息可以被视为构成围绕当前结果的包装器。此架构仅包含四种实体数据类型的定义：

•辅助-几何-表示-项：定义几何-表示-项的子类型，该子类型不打算用作形状模型的元素，但可用于影响该模型元素的约束中。插槽特征的中间平面提供了一个示例。这不是槽本身几何图形的一部分，但可能参与零件模型中定位和定向槽的约束。

•变分–表示–项：表示–项的子类型被声明为模型–参数、固定–实例–属性–集合和显式–约束的超类型。这些实体只能在变分表示的包装中使用。它们提供传输后有效编辑模型所需的控制。

•变分-表示：这种表示的子类型定义了整个模型，包括当前结果（其本身就是一种表示）和包含变分信息的包装。当前结果不允许包含任何变分表示的实例–项。包含关系只需通过要求表示的所有实例（属于当前结果的项）也属于变分表示来建模。

•变分–当前–表示–关系：这捕获了包含变分–表示和嵌入式当前结果表示之间关系的性质。它是ISO 10303-43实体数据类型表示-关系的一个子类型，它指定了几个限制，旨在确保整体表示是自包含的，并且内部是自一致的。图4使用EXPRESS-G表示法描述了变分表示模式中实体之间的关系。



**3.5草图-模式**

sketch–schema为二维纵断面或草图提供表示，这些纵断面或草图可以通过多种方式定义。

首先，该模式定义了一个实体中立的草图表示法，一种形状表示法的子类型。这可能被认为是一个独立的几何模型，存在于模型空间以外的一些二维坐标系中，可能存储在库中。此类草图的几何图形是显式的，由二维曲线段和点的集合组成。中性-草图-表示的实例可能会应用变换，以在三维模型空间中定位和定向该实例，此时该实例可以用作扫掠操作的基础或某些曲面构造操作的横截面。中性-草图-表示的实例可能具有与之相关的变化数据，在这种情况下，还需要属于类型变化-表示，以及所有暗示（见第3.4节）。

其次，模式指定了定位的实体–草图。这是在三维模型空间中定义的，因为它通常会参与零件或部件模型，所以它被定义为几何表示的子类型–项目，而不是其本身的表示。因此，它不具有变分和非变分形式，但如果它参与变分表示的实例，则可能具有与之相关的参数和约束。Posited–sketch的一个实例以曲线段和点的形式指定草图的显式几何图形，所有曲线段和点都必须位于草图平面中。

第三，提供了重新定位的实体-中性-草图。这将通过参考中性-草图-表示和变换的实例，给出模型空间中草图的表示。虽然原始中性草图的几何图形是显式的，但在这里，它仅通过提供的转换隐式定义。此实体是的子类型

几何-表示-项目。

第四，定义了另一个实体隐式-显式-定位-草图-关系，该关系将定位-草图实例与其相应的重新定位-中性-草图实例相关联。

这四种实体类型适用于以下常见情况：

•草图最初在一些中性二维坐标系中创建，然后在三维模型空间中定位和定向。如果这样做，CAD系统通常会显式地创建三维草图的几何图形，并且该几何图形可以作为定位草图的实例捕获。但是，如果还捕获了重新定位的–中性–草图的相应实例，则这将传达应用的变换的细节。三维草图的隐式和显式表示将通过隐式-显式-定位-草图关系的实例进行关联。

|  |
| --- |
| **第 9 页** |

•相反，草图最初可在3D中定义，具有明确的几何图形，并由定位草图实例捕获。然后，可能需要将该草图转换为2D，以存储在库中，以便以后在正在构建的CAD模型中的其他位置重用。在这种情况下，还将捕获重新定位的中性草图的相应实例，该变换表示实际应用的实例的倒数，因为映射方向与前一种情况相反。如前所述，还将记录三维草图的隐式和显式形式之间的关系。

当然，中性-草图-表示的实例可用于生成多个定位草图，每个草图在整个模型中的位置和方向不同。

实体数据类型positioned–sketch具有属性sketch–basis，可在三种可能性之间进行选择。该草图的基础是（i）平面曲线，（ii）第42部分实体曲线的实例-有界曲面或现有模型的平面。ISO 10303-42中定义了扫描操作的三种几何元素。在每种情况下，基础都需要有维度3。

草图模式还允许将所谓的导入几何体与定位草图实例关联。这是在草图平面中定义的附加几何图元，但不是草图本身的一部分。导入几何图形的典型用途是定义封装应用程序的外壳形状。然后，在导入几何图元的图元和草图的图元之间应用几何约束，可以通过使用参数变化优化草图，同时不违反封闭空间的边界。

导入的几何图元由点和曲线组成，这些点和曲线可以根据草图平面外的几何图元以多种方式定义。草图模式为导入的点和曲线提供了以下可能性，所有这些点和曲线都是辅助-几何-表示-项目的子类型，位于辅助元素的一般标题下：

•隐式–平面–交点–由草图平面与模型空间中曲线的交点定义的点；

•隐式–平面–投影–点–由外部点在草图平面上以指定方向投影定义的点；

•隐式–平面–相交–曲线–由草图平面与模型空间中的曲面相交定义的曲线；

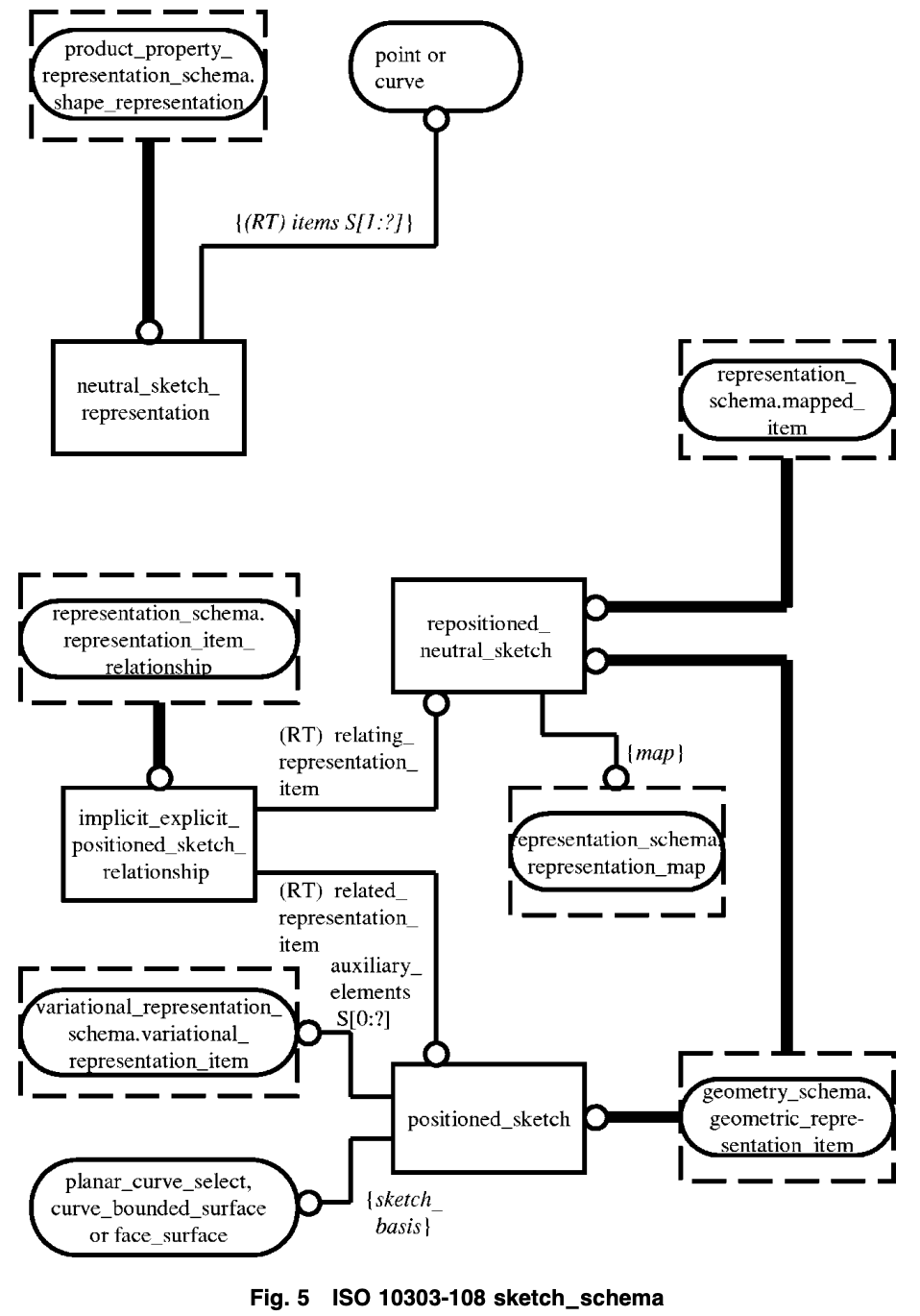
•隐式-模型-交点-曲线-由草图平面与曲面模型或边界表示实体模型的交点定义的（通常为复合）曲线；

•隐式-投影-曲线-由外部曲线在草图平面上以指定方向投影定义的曲线；

•隐式–轮廓–曲线-由曲面模型或边界表示实体模型在指定方向上照亮投影到草图平面上的阴影或轮廓定义的（通常为复合）曲线。

最后，sketch-schema提供了子草图的定义，顾名思义，子草图是草图的一部分。它是根据其所属草图的几何图形子集指定的。这将允许应用程序在此模式的专门化中为草图的不同部分定义不同的处理方式。模式本身只指定了一种这样的处理方式；实体刚体–子草图定义了草图平面内变换下的刚体所需的草图部分，即子草图的所有元素对之间的距离和角度关系需要在变换下保持不变。

图5显示了EXPRESS-G中建模的ISO 10303-108草图-模式的主要结构。未包括上述子草图和辅助几何元素的表示。在两个地方，不同属性类型之间选择的可能性已通过包含可用选项的圆角矩形显示，所有这些选项都在ISO 10303-42（ISO 2000a）中定义。这不是标准的EXPRESS-G符号，但为了清晰和简洁起见，这里采用了这种符号。



|  |
| --- |
| **第 10 页** |

**4 ISO 10303-108及其配套资源在标准化过程中的地位**

在撰写本文时，ISO 10303-108已在ISO TC184/SC4的20个成员国中通过了国际标准草案（DIS）投票。没有一个国家投了反对票，尽管从其中三个国家总共收到了15条评论。其中大多数是次要的编辑评论，但为了更好地与其他相关标准协调，还要求对第108部分草图-模式进行一些详细的技术更改。该文件的最终版本现已送交日内瓦的国际标准化组织出版，应在2004年底之前出版。

在撰写本文时，一份配套文件ISO 10303-55（“程序和混合表示”）[9]也通过了DIS投票，并已发送发布。因此，在结束之前，对这一新增的ISO 10303资源进行简要概述是合适的。

大多数CAD系统使用程序表示作为模型的主要内部描述。这捕获了用于构建模型的操作，而不是构建时模型的组成元素。CAD系统还生成显式边界表示类型的二次模型，主要用于屏幕显示、几何计算和用户交互（例如，用户从屏幕显示中拾取元素）。ISO 10303目前传输的是这个二次显式模型。

过程模型通常是混合的，包含一些明确定义的元素。这些可能包括明确的二维草图，有时这些草图本身就是模型，可以在CAD设计中重复使用多次。如前所述，此类草图可能具有变化性质，ISO 10303-108的主要目的之一是捕获与当前在ISO 10303传输中丢失的草图相关的显式参数和约束信息。

虽然ISO 10303-108可以代表“设计意图”信息的一个重要部分，但程序模型的捕获和转移将提供进一步的重大进展，因为这样的模型本质上很容易修改，所以只需更改构造操作的输入参数值并重新运行构造过程即可。事实上，这是此类表示在CAD系统中得到广泛应用的主要原因。ISO 10303-55旨在为此类模型的交换提供基本机制。这是一个相对较短的文档，通过使用ISO 10303实体数据类型规范作为构造器，即作为接收CAD系统中用于实际创建模型元素的内部操作的调用，它的简洁性得以实现。ISO 10303-11（“快速语言参考手册”）[4]允许这种用法，尽管提供构造函数最初是为了其他目的而设计的。

概述的方法允许立即规范涉及ISO 10303各部分中已有明确规范的任何构件的施工程序。所需要的是一些方法，用于对它们进行排序，用于处理显式元素（如草图）导入构造序列，以及用于处理用户在发送系统中从屏幕上拾取的元素。ISO 10303-55正好提供了这些功能。在最近的两篇论文[11,12]中给出了更多的细节，当这个新资源的标准化过程完成时，将发布ISO 10303-55的完整详细说明。

尽管ISO 10303-55可用于构建由现有ISO 10303实体组成的模型，但仍然存在一个问题，即现代CAD实践中广泛使用的高级设计功能尚未提供明确的表示。目前正在进行一个项目，以制定必要的规格。零件号ISO 10303-111已分配给设计特征资源，该资源的开发是一项涉及韩国、英国和美国的国际努力。该资源最近通过了委员会草案（CD）投票，这是标准化道路上的第一阶段。

**4.1 ISO 10303-108的实验验证**

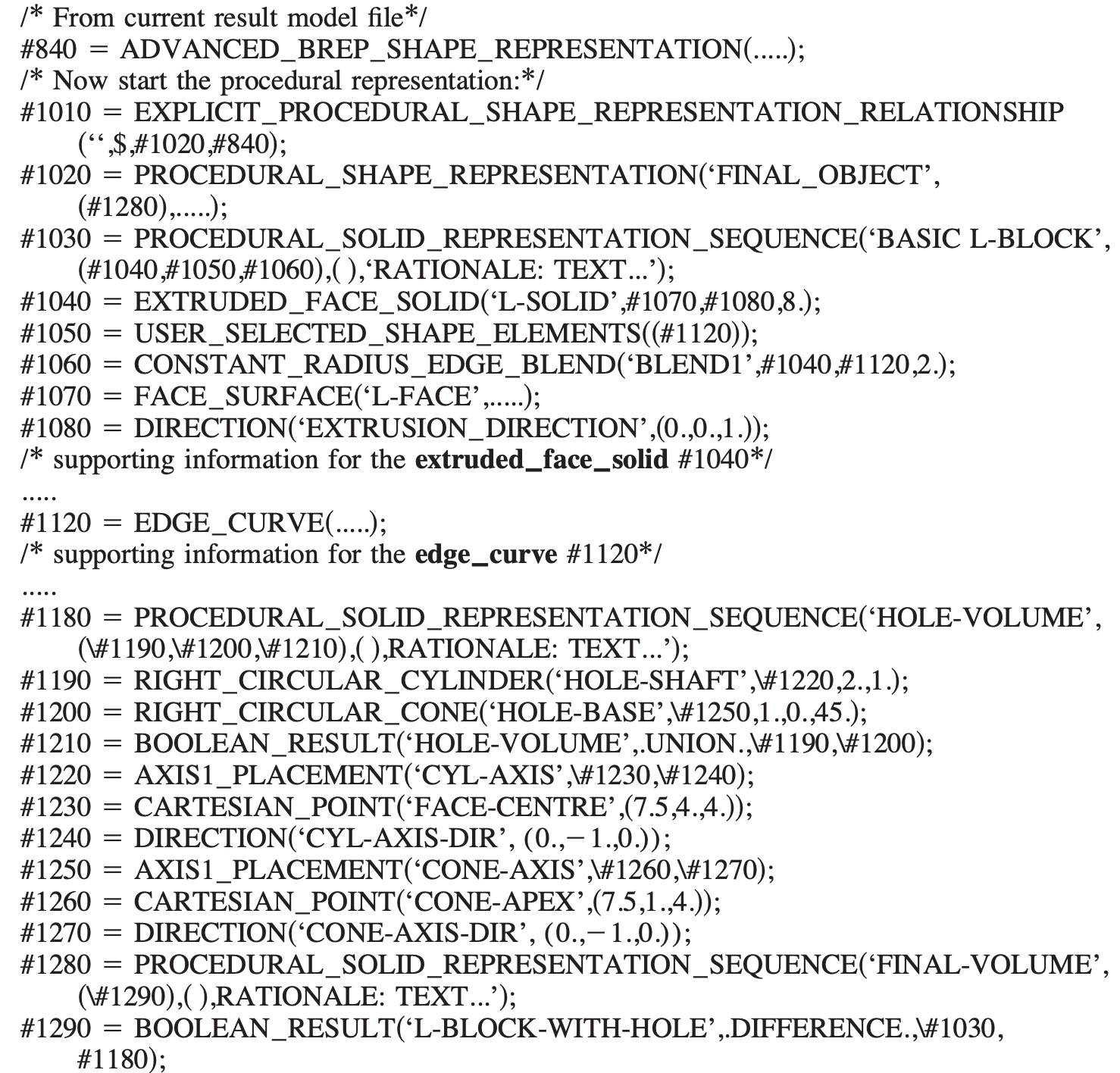
各种组织通过编写和测试实验翻译人员来验证ISO 10303-108的能力。其中包括美国国家标准与技术研究所（NIST）和美国工业/政府财团PDES Inc。到目前为止，尚未遇到任何根本性问题。已知有几份关于这些实验的报告正在准备中，其中一份已经发布[13]。根据ISO 10303-55、ISO 10303-108和ISO 10303-111草案版本的使用，这恰好涉及到最全面的测试。

该报告涵盖了一个为期14个月的项目，该项目由美国海军研究办公室（Office of Navy Research）资助，该项目旨在提高供应链的效率。该项目由PDES Inc.管理。涉及六家工程公司、两家主要航空航天公司及其四家供应商，这些公司建议将一系列实际工程零件用作测试模型。使用的CAD系统是达索系统公司的CATIA、参数化技术公司的ProEngineer和PLM系统公司的Unigraphics。通过ISO 10303新翻译人员测试的先前标准，获得的结果是良好的。遇到的最严重困难是由于ISO 10303-111的范围有限，它仍处于开发过程中。该项目表明，现在已经为使用STEP标准完全传输参数化模型奠定了主要基础[14]。

|  |
| --- |
| **第 11 页** |

**5 ISO 10303-108与ISO 10303-55结合使用的示例**

本节提供了一个示例，说明如何使用ISO 10303-108中定义的功能对产品形状进行建模。还显示了ISO 10303-55的基本用途，尽管未详细描述此资源，但如上文所述，它将作为未来论文的主题。模型形状是一个L形块，块上有一个孔。当前结果模型是ISO 10303-203（参见[2]）模型，位于ISO 10303-21[7]交换文件中的过程表示之前。ISO 10303其他部分中定义的实体数据类型用作构造函数。



假设模型的显式当前结果表示由advanced–brep–shape–representation实例#840总结。实例#1010定义了L块的显式表示和过程表示之间的双重模型关系。

在文件片段实例中#1030出现了ISO 10303-55中定义的程序-实体-表示-序列。该序列包含一个有序的实体列表，接收系统将这些实体解释为构造函数。该列表仅指定创建的主要元素；为了在本示例中清楚起见，其定义中涉及的次要元素紧随操作顺序列出。例如，实例#1040表示创建ISO 10303-42中定义的拉伸面实体实例的操作，以生成L形块的基础形状。该实例的属性之一是要拉伸的面。相关面列在操作序列之后，如#1070所示，后面是其定义中涉及的所有较低级别元素。

实例#1040创建形成L形块基本形状的拉伸后，将选择生成体积的凹边并对其执行圆角操作。请注意，选择操作由ISO 10303-55中定义的用户选择形状元素实例在操作序列中指示。在这种情况下，选择单个元素，其作为选定元素的区别决定了后续操作对其进行的适当处理。

实例#1180表示块中孔的形状，该孔由圆柱形部分和锥形尖端组成；单独创建各个卷，并通过布尔并集操作将其融合在一起。实例#1280定义了最终模型，该模型通过从L块形状中布尔减去孔形状获得。注意，前面的两个操作序列被重用为第三个序列的输入，这说明了在结构化设计的表示中嵌入不同级别的操作序列的可能性。

上述示例中的以下实例是ISO 10303-42中定义的实体数据类型，其名称大多是自解释的：

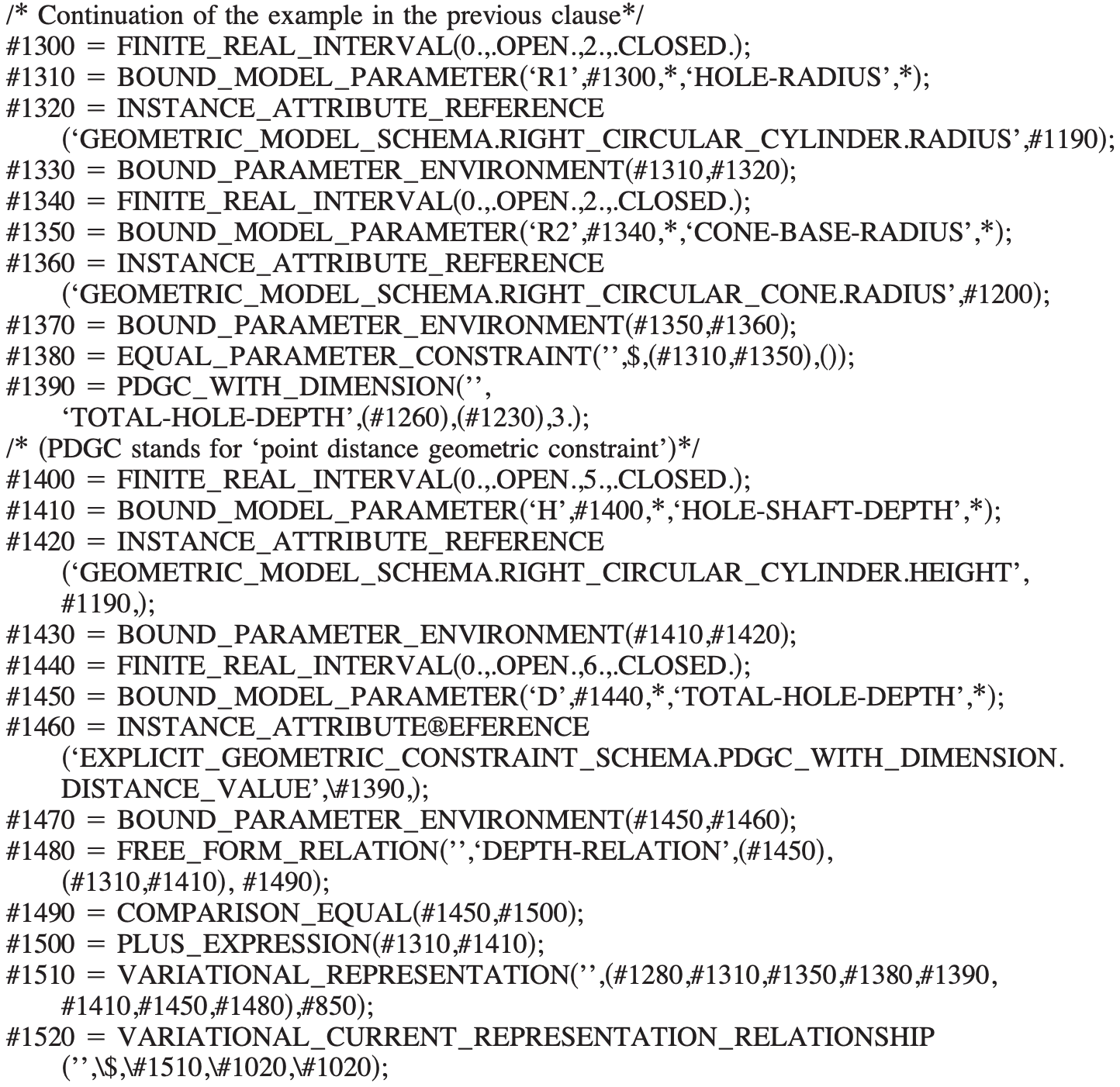
* 轴1–放置（一种局部轴系统）；
* 布尔-结果；
* 笛卡尔点；
* 方向
* 边-曲线；
* 挤压-面-实体；
* 面-面；
* 右-圆锥体；
* 右–圆形–圆柱体。

本示例程序部分中的所有其他实例（一个除外）均为ISO 10303-108中定义的实体数据类型，并在本文前面进行了描述。例外情况是虚拟实体数据类型constant–radius–edge–blend的实例#1060，它在一条边上定义了一个constant radius blend特征。其属性包括名称、对要定义混合的实体的引用、对要混合的特定边的引用以及混合半径。新兴的ISO 10303资源ISO 10303-111将包含表示此和许多其他高级基于特征的建模元素的规范。

如前所述，ISO 10303-108定义了变分表示的概念，一种包含明确表示的模型参数和约束的表示类型。任何变分表示的实例都需要有一个相关的当前结果，即所表示的参数产品族的一个成员，特别是具有所有相关参数的当前值的一个。使用ISO 10303-108实体数据类型-当前-表示-关系提供了必要的关联。

|  |
| --- |
| **第 12 页** |

ISO 10303-21[7]传输文件的以下片段显示了如何通过包含变化信息来扩展前一条款的示例：



文件片段提供了L块对象中孔的参数化。它分别在文件段#1300–#1330、#1340–#1370、#1400–#1430和#1440#1470中创建绑定-模型-参数的四个实例。这些段中的每一段都定义了

* 参数值的有效域（在所有情况下，非零实的有限范围）；
* 模型参数本身，具有名称、对其域的引用和描述。显示为派生的最后一个属性是参数的值，该值派生自参数所绑定的属性；
* 模型参数绑定到的文件中实例的特定属性，它表示该实例的值；
* 绑定–参数–环境实例，提供模型参数与其绑定到的属性之间的基本链接。
* 具有与其关联的参数的属性是
* 形成孔轴的圆柱体的半径；
* 形成孔底部的圆锥体的底面半径；
* 形成孔轴的圆柱体的高度；
* 从孔轴顶部中心到锥形底面顶点的距离。

存在两个显式约束实例。第一个是equal–parameter–constraint的实例，它只要求圆柱体半径和圆锥体的底面半径具有相等的值。第二个约束是孔的顶部和底部点之间的距离，要求该距离等于圆柱体的高度和半径之和。由于锥顶角为45度，且锥底半径等于圆柱体半径，因此该总和给出了孔的总深度。如果使用上述一组参数和约束增强了L块模型，则只需更改其圆柱形组件的半径值即可编辑实体中的孔，在这种情况下，模型中的其他尺寸将自动更改，以便生成的孔形状将保持具有锥形底座的钻孔的特征。

|  |
| --- |
| **第 13 页** |

文件片段中的最后两个实例定义了变分表示的实例及其与相应的当前结果的链接，在本例中是通过等效的过程形式。变分表示是指一组表示实例——包括当前结果（程序等价物）所指的所有实例，以及定义模型变分方面的显式表示参数和约束的所有实例。当前结果的程序形式被变分-当前-表示-关系引用了两次，这是该实体明确定义的一个要求，并且源于应用局部唯一性规则的需要。本例中的所有参数化和约束实例均为ISO 10303-108中定义的实体数据类型，但ISO 13584-20中定义的两种实体数据类型除外。ISO 10303-108使用该资源表示数学表达式和关系。相关实体包括：

* 比较——平等；
* 加-表达。

它们定义了数学表达式或这些表达式的元素之间的关系。

**6 总结**

本文描述了国际标准ISO 10303（STEP）的新部分ISO 10303-108的功能。它允许在显式模型的系统间交换中包含参数化和约束信息，特别是边界表示和密切相关类型的几何模型。它允许通过任何此类交换模型传输其原始创建者施加的显式参数化方案和约束，即迄今为止在ISO 10303信息传输中丢失的信息。CAD模型中此类参数和约束的主要应用领域是二维草图或轮廓、三维装配模型中的特征间关系和零件间关系。

ISO 10303-108设计为与新增的ISO 10303资源ISO 10303-55（“程序和混合表示”）兼容运行，这将提供使用ISO 10303完全传输参数化模型所需的其余基本机制。另一个资源ISO 10303-111将指定一组与主要CAD系统提供的设计特征相对应的设计特征，并将与前面提到的两个文档进行互操作

**致谢**

来自ISO 10303开发社区（ISO TC184/SC4）的许多人都参与了本文描述的工作。并非所有这些都可以提及，但在ISO 10303-108项目的生命周期内，比尔·安德森（美国先进技术研究所）、诺埃尔·克里斯滕森（美国霍尼韦尔）、雷·古尔特（英国LMR系统公司）、汤姆·克莱默（美国国家标准与技术研究所）、菲利普·克劳沙（美国波音公司）、大隆明彦（日本Unisys公司）做出了重大贡献，Guy Pierra（法国LISI/ENSMA）和Chia Hui Shih（美国Pacific STEP）。

作者衷心感谢美国国家标准与技术研究所的支持，使他能够参与本文所述的工作。

**参考**

[1] Owen, J., 1997, STEP: An Introduction, 2nd ed., Information Geometers, Winchester, UK.

[2] Pratt, M. J., 2001, ‘‘Introduction to ISO 10303—The STEP Standard for Product Data Exchange,’’ ASME J. Comput. Inf. Syst. Eng., 1, No. 1, pp. 102–103.

[3] ISO, 2004, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 108: Integrated application resource: Parameterization and constraints for explicit geometric product models,’’ International Standard ISO 10303-108, International Organization for Standardization, Geneva.

[4] ISO, 1994, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 11: Description methods: The EXPRESS language reference manual,’’ International Standard ISO 10303-11, International Organization for Standardization, Geneva.

[5] ISO, 2000, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 43: Integrated generic resource: Representation structures,’’ International Standard 10303-43, International Organization for Standardization, Geneva.

[6] ISO, 2000, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 42: Integrated generic resource: Geometric and topological representation,’’ International Standard 10303-42, International Organization for Standardization, Geneva.

[7] ISO, 1994, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 21: Implementation methods: Clear text encoding of the exchange structure,’’ International Standard 10303-21, International Organization for Standardization, Geneva.

[8] ISO, 2002, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 50: Integrated generic resource: Mathematical constructs,’’ International Standard 10303-50, International Organization for Standardization, Geneva.

[9] ISO, 2004, ‘‘Industrial automation systems and integration—Product data representation and exchange—Part 55: Integrated generic resource: Procedural and hybrid representation,’’ International Standard ISO 10303-55, International Organization for Standardization, Geneva.

[10] Bettig, B., and Shah, J. J., 2000, ‘‘Derivation of a Standard set of Geometric Constraints for Parametric Modeling and Data Exchange,’’ Comput.-Aided Des., 33, No. 1, pp. 17–33.

[11] Pratt, M. J., 2001, ‘‘Extension of the STEP Standard for Parametric CAD Models,’’ ASME J. Comput. Inf. Syst. Eng., 1, No. 3, pp. 269–275.

[12] Pratt, M. J., 2002, ‘‘Parametric STEP for Concurrent Engineering,’’ in Advances in Concurrent Engineering #Proc. 9th ISPE International Conference on Concurrent Engineering, Cranfield, UK, July 2002$ R. Gonc¸alves, R. Roy, and A. Steiger-Garc¸a˜o eds., Balkema Publishers, Lisse, Netherlands, pp. 933–940.

[13] Stiteler, M., 2004, ‘‘C! onstruction H! istory And P! arametricS!: Improving Affordability through Intelligent CAD Data Exchange,’’ CHAPS Program Final Report, Advanced Technology Institute, 5300 International Boulevard, North Charleston, SC 29418, USA.

[14] Pratt, M. J., Anderson, W. D., and Ranger, T., 2004, ‘‘Towards the Standardized Exchange of Parameterized Feature-based CAD Models,’’ in press, Computer Aided Design.