

OFFICE OPEN XML 概述



ECMA TC45
TOM NGO (NextPage), 编辑

1 简介

Office Open XML (OpenXML) 是一项针对字处理文档、演示文稿和电子表格的建议开放标准，可由多个应用程序在多个平台上自由地实施。不管是要实施支持该格式的应用程序的组织、采购此类软件的商业实体和政府实体还是从事该格式教学的教育者或作家，都会因这种标准的发布而获益。最终，所有用户都可以享受“所有文档，一种 XML 标准”所带来的好处，包括稳定性、持久保存性、互操作性和正在进行的其他改进。

OpenXML 的标准化工作是由 Ecma International 通过其 Technical Committee 45(技术委员会 45, TC45) 执行的，来自 Apple、Barclays Capital、BP、英国国家图书馆 (The British Library)、Essilor、Intel、Microsoft、NextPage、Novell、Statoil、Toshiba 和美国国会图书馆 (United States Library of Congress) 的代表参与了该项工作 (1)。

本白皮书对 OpenXML 进行了总结。阅读本白皮书可以：

- 了解 OpenXML 的目的及其规范的结构
- 了解其特性：它处理向后兼容性、持久保存性、可扩展性、自定义架构、子设置、多平台、国际化和辅助功能的方式
- 了解如何了解任何 OpenXML 文件的高级结构，快速导航到您需要的中获取更详细信息的规范部分

2 标准的目的

OpenXML 最初的设计目的是为了能够如实地再现以 Microsoft Corporation 定义的二进制格式进行编码的字处理文档、演示文稿和电子表格的预先存在的主体。标准化过程包括：在 XML 中镜像表示现有主体所需的功能、对其进行扩展、提供详细文档和支持互操作性。编写时，4 亿多的用户以二进制格式生成文档，估计文档数量会超过 400 亿，而每年正以数十亿的速度增加。

这些文件的原始二进制格式是在空间宝贵、分析时间严重影响用户体验的时代创建的。它们基于 Microsoft® Office® 应用程序使用的内存中数据结构的直接序列化。当今的硬件、网络 and 标准基础架构（尤其是 XML）容许通过多个平台上的多个供应商支持实现并考虑到发展的新设计。

在这些技术发展的同时，市场已经多样化，包含了一系列在简单的文档编辑程序中最初没有考虑的新应用程序。这些新的应用程序包括：

- 根据业务数据自动生成文档的应用程序；
- 从文档中提取业务数据并将这些数据输入到业务应用程序中的应用程序；
- 只对文档的一小部分执行受限任务，但保留文档可编辑性的应用程序；
- 为具有特殊需要的用户群（如盲人）提供辅助功能的应用程序；或
- 在各种硬件（包括移动设备）上运行的应用程序。

也许，最深奥的问题是长期保存的问题。我们已经学会了创建快速增长的大量信息，但是我们使用数字表示形式对这些信息进行了编码，这些表示形式与创建它们的程序结合得如此紧密，以致于一、二十年后，读取它们时通常很难保证不会有明显的损失。保存这些文档（现有文档和新文档）中的金融投资和智力投资已经成为头等大事。

二进制格式的广泛采用、技术发展、需要多种应用程序的市场力量以及长期保存的难度增加这四种影响力的浮现形成了一种迫切的需要，即定义一种开放的 XML 格式并以尽可能少的损失将数十亿文档迁移到该格式。此外，对该开放 XML 格式实现标准化并随着时间推移对其进行维护会创建一个环境，在该环境中，任何组织都可以安全地信任规范的持续稳定性，确信进一步的发展将受益于标准流程提供的制衡。

目前已经存在着各种文档标准和规范，包括 HTML、XHTML、PDF 及其子集、ODF、DocBook、DITA 和 RTF。就像各种表示位图图像的标准（包括 TIFF/IT、TIFF/EP、JPEG 2000 和 PNG）一样，每种都是针对一组不同的用途创建的。OpenXML 满足了人们的需要，提供了一种能涵盖现有文档主体中所表示功能的标准。据我们所知，它是唯一支持每种二进制格式功能的 XML 文档格式。

3 该标准结构

OpenXML 定义了字处理、演示文稿和电子表格文档的格式。每种文档都是通过以下主标记语言指定的：WordprocessingML、PresentationML 或 SpreadsheetML。嵌入机制允许这三种文档中任何一种文档包含其他主标记语言格式的材料以及大量支持标记语言格式的材料。

规范中既包含标准性材料（用于定义 OpenXML 的材料），也包含信息性材料（帮助读者了解的材料，但不是说明性材料）。它的结构按部分进行组织，以满足不同读者的需要。

- 第 1 部分 - 基础知识
165 页
 - 定义词汇表、符号约定和缩写。
 - 总结了三种主标记语言和支持标记语言。
 - 建立一致性条件，提供互操作性指导。
 - 描述开放数据包约定中针对每种文档的约束。

- 第 2 部分 - 开放数据包约定
125 页
 - 定义开放数据包约定 (OPC)。每一 OpenXML 文件包含一组称为 part (部分) 的字节流，这些部分合并到一个称为数据包的容器中。数据包格式由 OPC 定义。
 - 描述使用 Zip 文件格式的 OPC 的推荐物理实现。
 - 在仅以电子形式提供的附件中，将 OPC 的 XML 架构声明为 XML 架构定义 (XSD) (2)。该附件还包含使用 RELAX NG (ISO/IEC 19757-2) (3) 的架构非标准表示形式。

- 第 3 部分 - 初级知识
466 页
 - 介绍每种标记语言的功能，提供上下文，并通过示例和图表来阐明元素。此部分是信息性材料（非标准性材料）。
 - 描述在数据包中存储自定义 XML 数据以支持与业务数据的集成的功能。

- 第 4 部分 - 标记语言参考
5756 页
 - 定义每个元素和属性、元素父/子关系的层次结构，并根据需要定义其他语义。此部分主要用作需要有关元素或属性的完整详细信息时的参考。
 - 定义用于存储自定义 XML 数据的功能。
 - 在仅以电子形式提供的附件中，将标记语言的 XML 架构声明为 XSD (2)。该附件还使用 RELAX NG (ISO/IEC 19757-2) (3) 以非标准的形式表示了这些架构。

- 第 5 部分 - 标记兼容性和可扩展性
34 页
 - 描述用于扩展 OpenXML 文档的功能。
 - 指定具有不同扩展的应用程序进行互操作所需的元素和属性。
 - 使用 NVDL (ISO/IEC 19757-4) (4) 表示了可扩展性规则。

为了更轻松地读取和浏览这些文档，电子版本包含许多内部活动链接。尤其是，第 4 部分中通篇包含了指向父/子元素的链接。

4 该标准的特性

本节通过描述 OpenXML 的一些高级特性，帮助您了解 OpenXML。每个小节描述了其中的一个特性，并介绍了 OpenXML 中的特定功能。

- “互操作性”描述 OpenXML 如何独立于专有格式、功能和运行时环境，使开发人员可以进行广泛选择。
- “国际化”介绍了 OpenXML 支持每一主语言组的一些代表方式。
- “对开发人员采用的障碍较低”、“精简性”和“模块性”列出了 OpenXML 避免或消除各方实现的实际障碍的具体方式：学习曲线、最小功能集和性能。
- “高保真迁移”描述了 OpenXML 如何满足在现有文档和新文档中保存信息（包括原始创建者的所有意图）的总体目标。
- “与业务数据集成”描述了 OpenXML 如何在自定义架构中并入业务信息，以支持生产力应用程序和信息系统之间的信息集成和重用。
- “创新空间”描述了 OpenXML 如何通过定义进一步的可扩展性机制和提供具有不同功能集的应用程序间的互操作性，来为将来做好准备。

本文档的其余部分（包括本节）是 OpenXML 的主题性指南。对规范的引用都采用“§ 部分:节.小节”的形式；例如，§1:2.5 指的是规范的第 1 部分的第 2.5 节。对本文中其他标题的引用则采用名称形式。

4.1 互操作性

开发人员可以编写在多个平台上使用和生成 OpenXML 的应用程序。

最重要的是，具有不同背景和集团利益的 Ecma TC45 委员会 (1) 成员通过撰写、修改和审阅规范，实现了 OpenXML 的互操作性。代表包括：

- 要将 OpenXML 用于多种用途的多平台（Linux、MacOS 和 Windows）供应商（Apple、Intel、Microsoft、NextPage、Novell 和 Toshiba）
- 在现有内容（包括关键任务事务系统）上具有重大投资的集团（BP、Barclays Capital、Essilor 和 Statoil）
- 英国国家图书馆（The British Library）和美国国会图书馆（United States Library of Congress），它们都对“持久保存”抱有浓厚兴趣

在准备期间，委员会成员提出并解决了数百个与策略、明确性、语义和对环境的可能依赖有关的问题。代表性问题和其他活动包括：

- 支持独立于平台机制的、原始二进制格式的专有功能
- 一致性条件
- 架构内容
- 使用 RELAX NG (ISO/IEC 19757-2) 和 NVDL (ISO/IEC 19757-4) (4) 对架构和可扩展性机制的其他表示
- 对用于自动分析和可视化架构的工具进行的开发
- 国际化
- 在整个规范中对说明进行完善、更正和明确，在很多情况下是尝试实现各个规范部分的结果

本小节的其余部分重点介绍 OpenXML 在互操作性方面与原始二进制格式的具体不同之处。

对互操作性的一个核心要求是独立于任何特定类型的源内容。

- OpenXML 不对图像、音频或视频类型进行限制。例如，图像可以是 GIF、PNG、TIFF、PICT、JPEG 或任何其他图像类型 (§1:14.2.12)。
- 嵌入的控件可以是任何类型，例如 Java 或 ActiveX (§1:15.2.8)。
- WordprocessingML 字体规范可以包括字体指标和 PANOSE 信息，以便在原始字体不可用时帮助查找替代字体 (§3:2.10.5)。

另外，OpenXML 避免依赖生成文档的应用程序的运行时环境。

- 例如、显示外观的一部分图像由外部控件或应用程序生成。为了防止出现控件或应用程序不可用或者在给定的运行时环境中无法运行的情况，文档文件可以包含图像表示。该机制在早期的二进制格式中也存在。
- OpenXML 引入了一个更通用的称为“替代内容块” (§3:2.18.4) 的机制，该机制可在使用应用程序无法解释生产应用程序所编写的内容的各种情况下使用。它通常在可扩展性上下文中使用。“创新空间”小节中将进一步描述该机制。
- OpenXML 避免依赖任何在文档生成者环境中有意义、但在使用者环境中无意义的参数。例如，参数 CT_SYSCOLOR 在生产环境中是颜色表的索引。为了支持向另一使用环境的可移植性，PresentationML 允许生成者缓存在创建文档时使用的系统颜色。

最后也是最根本的，Office OpenXML 符合开放 W3C 标准（例如 XML (5) 和 XML 命名空间) (6)。这一事实本身即支持了所有符合这些开放标准的平台和操作系统中的基本级别的互操作性。

4.2 国际化

OpenXML 支持阿拉伯语、中文（三种变体）、希伯来语、印地语、日语、朝鲜语、俄语和土耳其语等多种语言所需的国际化功能。

OpenXML 本身支持 Unicode，因为它是 XML。另外，OpenXML 具有一组经过多年细化的丰富国际化功能。下面的列表具有代表性：

文本方向： OpenXML 支持从左向右 (LTR) 和从右向左 (RTL) 语言。它还支持双向 (“BiDi”) 语言，例如阿拉伯语、波斯语、乌尔都语、希伯来语和意第绪语，这些语言从右向左读写，但是可以包含从左向右读写的嵌入文本段。在 WordprocessingML 中，既可以在段落级别 (§4:2.3.1.6) 也可以段落中的片段级别 (§4:2.3.2.28) 控制文本方向。同样，在 DrawingML 文本中，可以在正文级别 (§4:5.1.5.1.1)、段落级别 (§4:5.1.5.2.2) 和项目符号 (§4:5.1.5.4) 中控制文本方向。

文本流： 在 WordprocessingML 中，可以在节或表的级别 (§4:2.3.1.41) 或在段落级别 (§4:2.3.2.28) 控制文本流的方向。在节和表级别，可以在垂直和水平方向上控制文本流。这使得 OpenXML 可以支持所有可能的文本布局（例如支持蒙古语所需的自上而下并

从左向右堆叠的垂直线条）。这会影响到列表、表和其他演示文稿元素的布局。DrawingML 还在段落级别和片级别使用 Kumimoji 设置，以便水平书写文本，垂直书写数字 (§4:5.1.5.2.3 和 §4:5.1.5.3.9)。在 WordprocessingML (§4:2.3.1.16) 和 PresentationML (§4:4.3.1.15) 中，还可以使用首尾字符设置来指定字符流，以指定允许使用哪些字符来开始和结束一行文本。

编号表示： 对于 WordprocessingML 中的字段格式 (§4:2.16.4.3)、WordprocessingML 中的段落/列表编号 (§4:2.9) 和 DrawingML (§4:5.1.5.4、§4:5.1.12.61) 中的编号，可以使用数十种编号格式中的任何一种来设置编号格式，包括平假名、阿拉伯语、Abjad、泰语、序号文本（例如“一百二十三”）、中文、朝鲜语 (Chosung 或 Ganada)、希伯来语、印地语、日语、罗马语或越南语。这些功能还支持任意小数点值（例如“1.00”与“1,00”）和列表分隔符。在 SpreadsheetML 中，国际化编号格式的功能非常强大，在单元格格式 (§4:3.8.30) 中和对外部数据的引用 (§4.3.13.12) 中支持所有这些功能。

日期表示： 在 WordprocessingML (§4:2.18.7) 和 SpreadsheetML (§4:3.18.5) 中，可以使用格里高利历（三种变体）、希伯来历、回历、日本日历（和历）、朝鲜日历（檀纪）、萨卡日历、台湾日历和泰国日历格式来编写日历日期。

公式： SpreadsheetML 中的公式规范提供了一些与国际化相关的转换函数，例如 BAHTTEXT (§4:3.17.7.22)、JIS (§4:3.17.7.185) 和 ASC (§4:3.17.7.11)。

语言标识符： 在 WordprocessingML (§4:2.3.2.18) 和 DrawingML (§4:5.1.5.3) 中，每个段落和片都可以用语言标识符来标记，使应用程序可以选择相应的校对工具和其他特定于语言的功能。除了每种语言的标识符，OpenXML 还支持字符集、字体系列和 PANOSE 值的命名，以便在本地支持不可用时帮助应用程序选择相应的替换字符集。

4.3 对开发人员采用造成的障碍较少

有经验的开发人员阅读规范几个小时后可以编写简单的 OpenXML 应用程序。

虽然规范描述了大量特性,但是 OpenXML 构造的应用程序不必支持规范中的所有特性。一致性声明 (§1:2) 只要求一致性使用者“不要拒绝它期望使用的文档类型的任何一致性文档”,且一致性生成者“能够生成一致性文档”(§1:2.5)。它还提供了对元素语义的角色进行说明的互操作性指南 (§1:2.6)。

一致性应用程序可以有非常集中的功能。例如,它可以是批处理程序(只更新字处理文档集中的版权声明),也可以是文本到语音阅读器(完全理解幻灯片演示文稿内容,可以在用户导航幻灯片时以音频方式呈现幻灯片文本内容)。借助文件格式的结构,只需了解 OpenXML 的最基本知识即可编写此类程序。特别之处:

- 该文件格式符合已建立好的、存在成熟工具的标准(尤其是 XML 和 ZIP)。
- 该文件格式使用开放数据包约定,该约定将 XML 和 ZIP 与标准机制组合起来,以表达文件中的关系。因此,无需了解 OpenXML 中任何主标记语言或支持的标记语言的标记语义,即可导航文件的内容。
- 无需打开其余结构,即可访问和修改 XML 树中深层次的元素。

通过提供缓存值,文件格式的微小细节(其中一些未在二进制格式中提供)能够以最小的功能来支持应用程序。例如:

- 无需实现分页程序,盲人阅读器之类的应用程序即可使用最后计算的分页符提供页面导航 (§4:2.3.3.13)。
- 无需实现公式和与外部数据源集成,电子表格程序即可通过缓存的计算 (§3:3.2.9) 和缓存的外部数据 (§4:3.14 和 §4:3.10.1.76) 运行。

最小一致性文档非常简单;请参见“最小 WordprocessingML 文档”小节。

4.4 精简性

OpenXML 文件格式支持创建高性能应用程序。在本小节中,将描述一些设计点,这些设计点可以生成精简的文件,因而提高了处理和分析速度。在下一小节中,将演示模块文件结构如何支持应用程序通过只分析或修改文档的一小部分来完成多项任务。

OpenXML 文件通常存储在 ZIP 存档中,按照开放数据包约定中的推荐实现来进行打包和压缩。您可能会感到惊奇,OpenXML 文件大小与同等的二进制文件相比,平均要小 25%,有时最多可小 75%。例如,本白皮书的大小比二进制格式文件大 85%! 精简的另一个简单来源(特别是需要未压缩的表示形式的场合)是 XML 中的标识符长度。常用的标记名称是简称。通常鼓励实现者也使用简短命名空间前缀;例如,WordprocessingML 命名空间的常规前缀是“w”。

更进一步的精简是通过在文件格式上避免重复实现的。例如删除大对象的冗余存储。

- 在 SpreadsheetML 中，重复的字符串存储在工作簿的字符串表中，按索引进行引用 (§3:3.3)。
- 在 SpreadsheetML 中，向下填充或跨若干单元格填充的公式作为单一“主”公式存储在左上单元格中，填充范围中的其他单元格通过分组索引来引用它 (§3:3.2.9.2)。
- 在 DrawingML 中，形状名称 (§4:5.1.12.56)、文本几何 (§4:5.1.12.76) 和其他预设（可以通过 §3:5.8、§3:5.9 和 §4:5.1.12 找到一些）是按名称或编号（而不是显式）表示的。在这些情况下，名称和编号的含义位于规范中，而不是在文件中。在这里，所选的表示是标准过程中明确权衡决策的结果。它是简洁的，且允许在正确的抽象级别进行编辑：例如，通过更改一个属性，可以将长方形更改为椭圆 (§4:5.1.11.18)。

在另一类示例中，层次结构用于提供继承语义。由此带来的作用是正面的：通过降低文件大小提高了性能。

- 在 WordprocessingML 中，样式是分层次的 (§3:2.8.9)。
- 在 DrawingML 中，形状按层次结构分组 (§4:5.1.2.1.20)。
- 在 PresentationML 中，默认层次结构与幻灯片母版、幻灯片布局和动画片相关 (§3:4.2)。

OpenXML 的其他方面的设计也是为了支持高效实现。例如，在 SpreadsheetML 中，单元格表只存储非空单元格，并且能够将合并的单元格表示为一个单元。对于稀疏电子表格，此技术带来的经济性是显著的。

4.5 模块性

应用程序可以通过分析或修改文档的一小部分来完成多项任务。

OpenXML 格式的三个功能一同来提供此模块性。

- 文档不是一个整块，它是由多个部分组成。
- 各部分之间的关系由各部分自己存储。
- 通常用于支持 OpenXML 文档的 ZIP 存档格式支持对每个部分的随机访问。

例如：

- 应用程序可以只将一张幻灯片从一个演示文稿移到另一个演示文稿，也可以与图像、布局等资源一起移动到另一个演示文稿，还可以在不分析幻灯片内容的情况下整体移动到另一个演示文稿 (§3:13.3.8)。该示例使用称为显式关系的数据来查找幻灯片及其资源。显式关系是由开放数据包约定定义的，无需了解任何 PresentationML 标记语义即可对该关系进行分析 (§1:9.2 和 §2:8.3)。
- 应用程序可以在不分析 WordprocessingML 文档的任何内容的情况下从该文档中提取所有注释 (§1:11.3.2)。该示例使用称为隐式关系的数据来查找注释。隐式关系特定于 OpenXML，因此需要了解一些相关标记语言的知识 (§1:9.2)。

4.6 高保真迁移

OpenXML 的设计目的是支持 Microsoft Office 97-2003 二进制格式中的所有功能。

不宜夸大完成此目标的难度以及 OpenXML 这样做的独特性。某些格式（例如 PDF）的设计目的是向最终用户提供已完成文档的可视描述。比较而言，OpenXML 的目的是允许将来在原始创建者的抽象级别进行编辑或操作；例如，将矢量图形缩减为位图会将样式层次结构折叠为独立样式，因此不符合此意图。进一步而言，文档可以包含原始创建者希望保存的计算语义，例如依赖中间计算结果的公式逻辑，包括错误代码或产生动态行为的动画规则。

这些对规范的引用将 OpenXML 的功能示例化，可以表现出该二进制格式的细小方面。

- SpreadsheetML 说明包含全面的公式规范 (§4:3.17.7)。
- WordprocessingML 规范介绍了使用直接格式编写段落、字符、编号和表属性时的规则 (§3:2.8，尤其是 §3:2.8.10)。
- PresentationML 规范说明了动画功能 (§3:4.4)。

OpenXML 支持多个实现一次保持一致，不必匹配每个不合理的细节。这在涉及数字计算（例如布局、效果呈现和公式求值）时尤其重要。如果要求的一致性比实际更高，会对开发人员达到一致性产生不必要的巨大阻碍。以下陈述重点描述了委员会在这一方面的决策示例。

- OpenXML 定义诸如表面外观的效果 (§5.1.12.50)，不要求开发人员逐像素地匹配这些效果。
- OpenXML 定义诸如页边距 (§4:2.6.11)、字体 (§4:2.8) 和对齐 (§4:2.3.1.13) 的参数。它允许开发人员在使用这些参数时实现不同的流算法。
- 由于删除浮点计算中的变体通常要求一致性应用程序实现较慢的仿真（而不是依赖本机硬件），因此 SpreadsheetML 公式规范 (§4:3.17.7) 不尝试删除浮点计算中的变体。相反，它会指定数字计算的最小精度位数 (§4:3.17.5)。
- SpreadsheetML 公式规范还让一些条件性决策由实现来定义，以便于将来的创新。例如，它不限制诸如 NORMINV (§4:3.17.7.227) 的计算应迭代的次数。（NORMINV 函数通过执行迭代搜索来进行反正态分布。）

包含大量早期功能（例如 VML (§3:6)）主要是为了向后兼容。OpenXML 中使用了较新的标准，例如 DrawingML (§3:5)，在编写新文档时建议使用这些标准。

4.7 与业务数据集成

通过 OpenXML，组织可以将生产率应用程序与通过启用 OpenXML 文档中的自定义架构管理业务流程的信息系统相集成。组织采用此方法的目的是

为了重新使用业务信息并自动对其进行处理，否则，这些信息将被埋葬在文档中而失去价值，业务应用程序不能在这些文档中读取和写入业务信息。

应用包括：

- *搜索*：最终用户可以搜索一组电子表格来查找利润超过 20% 的公司。
- *元数据标记*：公司可以对管理部门已批准的表示形式进行标记。
- *文档程序集*：建议小组可以通过自动进行基础数据的准备来简化建议的生成。
- *数据重用*：销售管理人员可以生成一个报表，其中包含给定日期范围内的所有销售合同，并列出客户、交易规模和所有已修改的条款和条件。
- *行业应用*：行业顶尖专业人员可以在熟悉的创作环境中准备可交付结果，让他们的工作成果自动流入业务系统中。

完成这些目标需要定义一类文档可包含的数据结构和类型，使信息可以在每个文档流中自然出现该信息的位置出现。假设一个简单的简历示例。某人定义了一个包括姓名、电话号码、地址、求职目标和资历等字段的数据结构。然后，他（她）安排这些字段在人力资源工作人员将它们放入文档时显示。在另一个业务设置（例如财务小组或医疗中心）中，结构和数据字段会有所不同。

OpenXML 让此流程以标准形式发生。

首先，先使用自定义 XML 架构表示业务数据的结构。这使组织可以用对于业务前景有意义的标记来表示数据。组织可以创建其自己的架构，也可以使用行业标准架构（例如用于财务报告的 XBRL (7) 和用于医疗信息的 HL7 (8)）。架构是在公司内部的公共事业部创建的，作为行业标准，其应用范围可以涉及出生证到保险信息等各个领域。可以使用任何自定义架构，只要它是用 XSD 形式 (2) 表示的。

其次，自定义架构嵌入在自定义 XML 部分 (§3.7.3) 的任何 OpenXML 文档中，可以使用自定义 XML 数据属性部分 (§4:7.5) 进行描述。通过将这些自定义数据与表示形式分隔开来，OpenXML 可以启用干净的数据集成，使最终用户可以在各种上下文（包括文档、表单、幻灯片和电子表格）中进行演示和操作。因此，可以在更基础和语义上更准确的级别达到互操作性。

4.8 创新空间

OpenXML 旨在鼓励开发人员创建在定义二进制格式甚至是在定义 OpenXML 时没有考虑到的新应用程序。

首先，讨论可扩展性机制，这些机制一同协作以允许具有不同功能集的应用程序间的互操作性。现在考虑一个 *上级* 应用程序（此应用程序包含 OpenXML 中未说明的新功能）和一个 *下级* 应用程序（此应用程序不理解该功能）。可扩展性的三个主要目标是：

- *视觉保真度*：下级应用程序显示上级应用程序应显示的内容的能力。这在本质上要求文件存储同一数据的多种表示。
- *可编辑性*：编译一种或多种表示的能力。
- *隐私*：确保在编辑一种表示之后不会保留另一种表示的旧版本的能力（否则会意外保留用户确信已删除或已修改的信息）。应用程序可以通过消除或同步表示来达到此目的。

希望扩展 OpenXML 功能集的开发人员有两个选项：

- *备用内容块*：备用内容块 (§3:2.18.4 和 §5:9.2) 存储同一内容的多种表示，将每一种表示存储在其自己的选择块中。下级应用程序读取能够读取的一个选择块。在编辑时，它能够写入多少选择块，就写入多少选择块。
- *扩展列表*：扩展列表 (§3:2.6) 存储没有可视表示形式的任意自定义 XML。

开发人员拥有在这些可扩展性机制外部进行创新的空间。

- *备用交互范例*。OpenXML 指定的文档语法较多，但是指定的应用程序行为较少。如一致性声明中所述，它的重点放在语义上 (§1:2.2 和 §1:2.3)。因此，一致性应用程序可以随意通过各种方法与最终用户通信，也可以根本不与最终用户进行通信 - 只要它符合指定的语义。
- *Novel 计算环境*。一致性声明允许低容量的应用程序以便它们可以在小型设备上运行，并允许只实现 OpenXML 子集 (§1:2.6) 的应用程序。附加特征机制允许生产应用程序与其容量限制进行通信 (§3:8.1)。

如上面的小节所述，呈现供用户直接交互的文档不会提供一些最重要的创新机会。而使用 XML 消息格式（例如通过 XML Web 服务 (9)）进行计算机到计算机的处理却能提供大量的创新机会。虽然这样的应用程序除了对 OpenXML 文档中包含的数据进行的操作之外，没有用户可以看到的其他行为，但是它们受文档一致性 (§1:2.4)、和应用程序一致性 (§1:2.5)（纯语义）和互操作性指南 (§1:2.6)（集成了语义）的约束。

虽然无法枚举自定义 XML 处理的所有可能使用情况，但是可以预期以 XML 为中心的服务（处理用于自动提取和插入自定义数据的 OpenXML 文档）、自定义的安全服务（例如 XML 数字签名 (10) 或 XML 加密 (11)），甚至任意 XSLT 转换 (12)（转为其他 XML 格式或从其他 XML 格式转换出来）。OpenXML 对这样的处理没有任何禁止或限制。

本白皮书的主要目的是使读者遵循 OpenXML 文件的高级结构。为此，我们提供了关于开放数据包约定 (OPC) 的比较详细的信息，以及关于各个标记语言的简明信息。

5.1 开放数据包约定

开放数据包约定 (OPC) 提供了在一个容器（例如 ZIP 存档）中存储多种内容（例如 XML、图像和元数据）以便完整表示文档的方法。它们介绍了一种用于表示内容和关系的逻辑模型。

OPC 的建议实现使用的是 ZIP 存档格式。用户可以使用任何 ZIP 查看器来检查任何 OpenXML 文件的结构。在阅读此说明时，最好以这种方法检查一个小型 OpenXML 文件的内容。在 Windows 操作系统上，用户只需向文件名添加“.zip”扩展名，并进行双击。

逻辑上而言，OpenXML 文档是 OPC 数据包 (§5:8)。数据包是纯粹部件的集合 (§5:8.1)。每个部件有一个不区分大小写的部件名，该名称由斜杠分隔的段名序列组成，例如“/pres/slides/slide 1.xml” (§5:8.1.1)。每个部件有一个内容类型 (§5:8.1.2)。物理上而言，该 ZIP 存档是一个数据包，该存档中的每个 ZIP 项是一个部件，该 ZIP 存档中的路径名直接对应于部件名。

在 ZIP 实现中，“/[Content_Types].xml”允许用户确定数据包中每个部件的内容类型 (§2:9.2.6)。媒体类型的语法和定义遵循 RFC 2616 的 3.7 节 (13)。

数据包和部件可以包含与数据包中其他部件以及外部资源的显式关系 (§1:9.2)。每个显式关系有一个关系 ID（以允许部件的内容引用该关系）和一个类型（以允许应用程序决定处理该关系的方式）。关系类型是使用 URI 命名的，这样，非对方也可以安全地创建新类型，而不会发生冲突。

给定源数据包或部件的显式关系集合存储在关系部件中。数据包的关系部件总体上称为“/_rels/.rels”；“/a/b/c.xml”部件的关系部件称为“/a/b/_rels/c.xml.rels”。关系部件（在 ZIP 实现中为内容类型部件）是数据包中唯一特别命名的部件。要打开数据包，应用程序必须分析数据包关系部件，并遵循相应类型的关系。

OpenXML 文档中的所有其他部件保存 OpenXML、自定义 XML 或任意类型的内容（例如多媒体对象）。对于嵌入的业务数据和元数据而言，部件保存自定义 XML 的能力是非常有力的机制。

5.2 WORDPROCESSINGML

WordprocessingML 文档由文章集合组成 (§3:2.1)。每个文章是以下类型之一：主文档 (§3:2.2)、词汇表文档 (§3:2.13)、子文档 (§3:2.18.2)、页眉 (§3:2.11.1)、页脚 (§3:2.11.2)、注释 (§3:2.14.5)、框架、文本框 (§3:2.18.1)、脚注 (§3:2.12.1) 或尾注 (§3:2.12.2)。

唯一需要的文章是主文档。它是数据包关系的目标，其类型为：

<http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument>

从根目录到 XML 树中叶目录的典型路径包含以下这些 XML 元素 (§3:2.2)：

- document - 主文档的根元素 (§3:2.3)。
- body - 主体 (§3:2.7.1)。可以包含多个段落。还可以包含 sectPr 元素中指定的节属性。
- p — 段落 (§3:2.4.1)。可以包含一个或多个片。还可以包含 pPr 元素中指定的段落属性，这些段落属性又可以包含 rPr 元素中指定的默认片属性（也称为字符属性） (§3:2.4.4)。
- r — 片 (§3:2.4.2)。可以包含多种片内容，尤其是文本范围。还可以包含片属性 (rPr)。片是 OpenXML 中的基本概念。片是具有相同属性的一小块连续文本，片不包含任何其他文本标记。例如，如果句子包含“这是三片”，则该句可至少由三片表示：“这是”、“三”和“片”。在这一方面，OpenXML 与允许任意嵌套属性的格式（例如 HTML）大不相同。
- t - 文本范围 (§3:2.4.3.1)。包含任意数量的文本，这些文本不含格式、换行符、表、图形或其他非文本材料。文本的格式从片属性和段落属性继承。此元素通常使用 `xml:space="preserve"` 属性。

在本小节中，我们通过指定段落属性和片属性了解了文本的直接格式。直接格式位于一组应用的结尾，这组应用还包含字符、段落、编号、表样式以及文档默认设置 (§3:2.8.10)。这些样式自己组织到继承层次结构中 (§3:2.8.9)。

下面的小节“最小 WordprocessingML 文档”列出了整个 WordprocessingML 文档。

5.3 PRESENTATIONML

PresentationML 文档由演示文稿部件描述。演示文稿部件是数据包关系的目标，其类型为：

<http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument>

演示文稿引用以下这些主构造 (§3:4.2)，它们在默认层次结构中自上而下列出：

- 幻灯片母版、备注母版和讲义母版 (§3:4.2.2)，它们都从幻灯片继承属性；
- 幻灯片布局 (§3:4.2.5)，它从幻灯片母版继承属性；
- 幻灯片 (§3:4.2.3) 和备注页 (§3:4.2.4)，它们分别从幻灯片布局和备注母版继承属性。

每个母版、布局和幻灯片存储在其自己的部件中。每个部件的名称是在演示文稿部件的关系部件中指定的。除演示文稿部件之外，六个部件中每个部件的构造方式本质上都相同。从根目录到 XML 树中叶目录的典型路径包含以下这些 XML 元素 (§3:2.2):

- sld、sldLayout、sldMaster、notes、notesMaster 或 handoutMaster - 根元素。
- csld - 幻灯片 (§4:4.4.1.15)。可以包含 DrawingML 元素(如下面两个项目符号内容中所述)和其他结构元素(如下所述)。
- spTree - 形状树 (§4:4.4.1.42)。可以包含 grpSpPr 元素中的组形状属性 (§4:4.4.1.20) 和 nvGrpSpPr 元素中的非可视组形状属性 (§4:4.4.1.28)。此节点及其从属节点都是 DrawingML 元素。在这里列出了一些 DrawingML 元素，因为它们在 PresentationML 中具有关键作用。
- sp - 形状 (§4:4.4.1.40)。可以包含 spPr 元素中的形状属性 (§4:4.4.1.41) 和 nvSpPr 元素中的非可视形状属性 (§4:4.4.1.31)。

除了 DrawingML 形状内容，csld 可以包含其他结构元素(这取决于它驻留的根元素)，下表进行了汇总:

	幻灯片	幻灯片布局	幻灯片母版	讲义母版	备注母版	备注页母版
普通数据	X	X	X	X	X	X
切换	X	X	X			
计时	X	X	X			
页眉和页脚		X	X	X	X	
匹配的名称		X				
布局类型		X				
保存		X	X			
布局列表			X			
文本样式			X			

默认层次结构中较低级别对象所指定的属性(幻灯片母版、幻灯片布局、幻灯片)覆盖该层次结构中较高级别对象所指定的对应属性。例如，如果没有为幻灯片指定切换，则它接受幻灯片布局中的切换；如果幻灯片布局中未指定切换，则接受幻灯片母版中的切换。

5.4 SPREADSHEETML

SpreadsheetML 文档由工作簿部件在顶层描述。工作簿部件是数据包关系的目标，其类型为：

<http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument>

工作簿部件存储有关工作簿及其结构的信息，例如文件版本、创建应用程序和要修改的密码。逻辑上而言，工作簿包含一个或多个表 (§3:3.2)；物理上而言，每个表存储在其自己的部件中，按常规方式从工作簿部件引用。每个表可以是一个工作表、图表或对话框表。我们将只讨论工作表，它是最常见的类型。在工作表对象中，从根目录到 XML 树中叶目录的典型路径包含以下这些 XML 元素：

- `worksheet` - 工作表中的根元素 (§3:3.2)。
- `sheet Data` - 单元格表，代表工作表中每个非空单元格 (§3:3.2.4)。
- `row` - 单元格表中的一行单元格 (§3:2.8)。
- `c` - 一个单元格 (§3:3.2.9)。 `r` 属性使用 A1 样式坐标表示单元格的位置。单元格还有一个样式标识符（属性 `s`）和一个数据类型（属性 `t`）。
- `v` 和 `f` - 单元格的值 (§3:3.2.9.1) 和可选公式 (§3:3.2.9.2)。如果单元格包含公式，则该值是最近计算的结果。

字符串和公式都存储在共享表中 (§3:3.3 和 §3:3.2.9.2.1)，以避免冗余存储，提高加载和保存的速度。

5.5 支持的标记语言

一些支持的标记语言也可以用于描述 OpenXML 文档的内容。

- DrawingML (§3:5) - 用于表示文档中的图形和其他以图形方式呈现的对象。
- VML (§3:6) - 矢量图形的格式，包括它是为了保证向后兼容性，它将逐渐被 DrawingML 取代。
- 共享 ML：数学 (§3:7.1)、元数据 (§3:7.2)、自定义 XML (§3:7.3) 和书目 (§3:7.4)。

5.6 最小 WORDPROCESSINGML 文档

本小节介绍最小 WordprocessingML 文档，该文档包含三个部件。

内容类型部件 “/[Content_Types].xml” 描述两个其他必需部件的内容类型。

```
<Types xmlns="http://schemas.openxmlformats.org/package/2006/content-types">
  <Default Extension="rels"
    ContentType="application/vnd.openxmlformats-package.relationships+xml"/>
  <Default Extension="xml"
    ContentType="application/vnd.openxmlformats-officedocument.wordprocessingml.document.main+xml"/>
</Types>
```

数据包关系部件 “/_rels/.rels” 描述数据包与主文档部件之间的关系。

```
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
  <Relationship Id="rId1"
    Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument"
    Target="document.xml"/>
</Relationships>
```

本例中的文档部件 “/document.xml” 包含文档内容。

```
<w:document xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main">
  <w:body>
    <w:p>
      <w:r>
        <w:t>Hello, world.</w:t>
      </w:r>
    </w:p>
  </w:body>
</w:document>
```

规范为 WordprocessingML (§1:11.2)、PresentationML (§1:13.2) 和 SpreadsheetML (§1:12.2) 提供了最小文档和其他详细信息。

6 总结

OpenXML 是具有不同背景和组织利益的多个行业和公共研究机构的代表经过艰苦努力的成果。它包含了现有文档主体中使用的完整功能集，以及全世界所有主要语言组中固有的国际化需求。作为 Ecma TC45 (1) 标准化工作和公共评论贡献的结果，OpenXML 已经达到互操作性和平台独立性的高级级别；其文档既完整（借助广泛的参考资料）又可访问（借助非标准性说明）。它还包含足够的信息，可供辅助技术产品正确地处理文档。OpenXML 实现既可以非常小并提供集中的功能，也可以囊括整个功能集。格式中内置的可扩展性机制保证了创新空间。

将该格式规范标准化并随时对其维护，可以确保各方可以安全地依赖它，以及确信进一步的发展将受益于开放标准流程提供的制衡。人们对能够保存用已存在的二进制格式创建的数十亿文档和每年要继续创建的数十亿文档的开放文档格式标准的需要越来越迫切。硬件、网络和基于标准的软件基础结构的领先技术使这一需要成为可能。市场需求涌现的变化（包括关键任务事务系统中的现有巨大投资）使其成为根本的需要。

7 引用

1. **Ecma International**。TC45 - Office Open XML 格式。 *Ecma International*。 [在线]
<http://www.ecma-international.org/memento/TC45.htm>。
2. **W3C**。XML 架构。 *万维网联合会*。 [在线] <http://www.w3.org/XML/Schema>。
3. **ISO**。ISO/IEC 19757-2:2003。 *国际标准化组织*。 [在线]
<http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=37605&ICS1=35&ICS2=240&ICS3=30>。
4. **ISO**。ISO/IEC 19757-4:2006。 *国际标准化组织*。 [在线]
<http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=38615&ICS1=35&ICS2=240&ICS3=30>。
5. **W3C**。可扩展标记语言 (XML) 1.0 (第四版)。 *万维网联合会*。 [在线] 2006。
<http://www.w3.org/TR/2006/REC-xml-20060816/>。
6. **W3C**。XML 1.0 (第二版) 中的命名空间。 *万维网联合会*。 [在线] 2006。
<http://www.w3.org/TR/2006/REC-xml-names-20060816/>。
7. **XBRL International**。XBRL 规范。 *可扩展业务报告语言*。 [在线]
<http://www.xbrl.org/Specifications/>。
8. **Health Level Seven**。HL7 ANSI 批准的标准。 *Health Level Seven*。 [在线]
http://www.hl7.org/about/directories.cfm?framepage=/documentcenter/public/faq/ansi_approved.htm。
9. **W3C**。W3C Web 服务体系结构。 *万维网联合会*。 [在线] 2002。
<http://www.w3.org/2002/ws/>。
10. **W3C**。W3C XML 签名。 *万维网联合会*。 [在线]
<http://www.w3.org/Signature/>。
11. **W3C**。W3C XML 加密。 *万维网联合会*。 [在线] 2001。
<http://www.w3.org/Encryption/2001/>。
12. **W3C**。XSL 和 XSLT。 *万维网联合会*。 [在线]
<http://www.w3.org/Style/XSL/>。
13. **W3C**。超文本传输协议 - HTTP/1.1。 *万维网联合会*。 [在线]
<http://www.w3.org/Protocols/rfc2616/rfc2616.html>。