DITA是一个开放的基于XML的标准，这意味着有一组不同的工具可用于编写，编辑，格式化和存储DITA文件。随着业务需求的变化，使用最适合你需求的工具。

XML看起来很像HTML。它们都使用尖括号（<和>）来分辨标记标签（例如<title>）。在这两种语言中，正斜杠标识了一个结束标签（</title>）

<title>This is a title</title>

在HTML和XML中， 标签也有属性（标签中的 attribute=”value”） 提供了关于标签的进一步信息：

<note type="warning">Keep your arms and legs inside the vehicle at all times.</note>

HTML和XML之间有两个主要区别：

HTML可以允许你忘记关闭标签或在属性值周围加上引号；但 XML的格式则不允许这样。

HTML使用一组预定义的标签（<body>，<p>，<span>等）。在XML中，标签在单独的文件中定义，并且可以由信息架构师来更改和添加。

DITA构建于：

1. 基于topic的写作
2. 将内容与格式分离
3. 极简主义
4. 结构化写作概念

## DITA重要概念

**Topic作为基本单元。**

在DITA中，topic是创作和重用的基本单元，传统上，DITA主题是一个有标题的信息单元，可以单独理解并在多个上下文中使用。它应该足够短，能够处理一个主题或回答一个问题，但是足够长，能够自己理解，并作为一个独立的单元进行编写。但是，DITA主题也可以是不太独立的信息单元，例如仅包含标题和简短描述的主题，主要用于组织为信息管理、创作方便或交换而设计——比如嵌套的副标题或链接或主题。topic基本结构：标题和（可选）内容。topic可以是通用的或更专业的；专用topic表示更具体的信息类型或语义角色，例如，<concept>、<task>、<reference>或<learningcontent>。

**DITA Map(映射**)

是将topic和其他资源组织成结构化信息集合的文档，文件扩展名应该是.ditamap。DITA映射指定topic之间的层次结构和关系；它们还提供定义和解析键的上下文。

DITA映射用于组织出版物的逻辑顺序和层次结构，搭建交付出版物的结构框架。映射中包含指向DITA主题的链接，这些链接按顺序或层级结构将分散的主题黏合起来成为集合，并按一定的逻辑结构组织成为具备有机结构的交付出版物。在形式上，DITA映射文件同样以XML为载体，D1TA通过映射来连接上下文内容。映射文件在内部使用<topicref>标签组织一个或多个DITA主题，并赋予主题上下文的顺序和层级关系。<topicref>标签引用主题的组装顺序表示内容上下文的先后关系标签的主题嵌套层级表示内容上下文的包含关系。映射文件以．ditamap为后缀，在编译处理时DITA通过映射这样的单一逻辑结构，来管理主题组装后的导航顺序，并且适用于各种类型交付出版物的聚合生成。如PDF文档的导航目录，即可通过DITA映射创建的目录(TableofContents：TOC)生成。DITA映射也可有多个映射文件级联，形成复杂的混合层级映射结构。

DITA 映射能够将一个主题集合组织成为不同类型的出版物。如一本百科全书的主题集合对应各个词条的内容释义。如果将所有具备人名标签的词条抽取出来，可以借助 DITA 映射重组为一本人名志；而将所有具备地名标签的词条抽取出来，则又可映射重组为一本地名志。依此类推可以借助 DITA 映射重组为其他同类属性主题集合的出版物，即一个主题经由多种映射关系组织成不同出版物，不同映射也能够将相同的主题集合组织成不同类别的出版物。DITA 映射为内容的重组和重用提供了途径。一方面，松散的内容模块由 DITA 映射文件赋予逻辑关联关系，使其重组成为有机的整体。另一方面，以主题为单元的模块化内容可以在DITA映射的组织下实现灵活重用。对于不同出版物中重复出现的相同主题模块，可以借助 DITA 映射直接将指定主题引入到出版物中，而无需对相同的内容重复地排版编辑，这在一定程度上减少了内容管理上的冗余。[¶](https://techcomm-book.readthedocs.io/zh_CN/latest/5-info-develop/dita_intro.html" \l "dita-dita-dita-dita-dita-dita-dita" \o "永久链接至标题)

**信息分类（Information typing）**

通过识别不同的topic（例如<concept>概念、<task>任务、<reference>参考）来清楚地区分不同信息类型。（基本信息类型举例——技术内容、机器行业、学习和培训）

回答不同读者问题的topic（比如如何…？——对应<task>任务，什么是…？——对应<concept>概念）可以根据不同的信息类型分门别类。

**条件处理**

条件处理，是根据处理时间标准过滤或标记信息。@audience、@deliveryTarget、@otherProps、@platform和@props属性（以及@props的专门化）为元素分配条件，以便在处理过程中包含、排除或标记元素。@rev标记属性允许将值分配给元素，以便在处理过程中对这些元素应用特殊格式。条件处理配置文件为DITAVAL文档类型。

**DITA寻址（DITA addressing）**

DITA提供了两种寻址机制。DITA地址要么是直接基于URI的地址，要么是间接基于key值的地址。在DITA文档中，单个元素是由@id属性上指定的唯一标识符寻址。DITA定义了两种片段标识符语法；一种是完整的片段标识符语法，另一种是缩写的片段标识符语法，当从同一topic中寻址非topic元素时可以使用缩写的片段标识符语法。

**内容重用**

DITA@conref、@conkeyref、@conrefend和@conaction属性提供了在DITAtopic或地图中重用内容的机制。这些机制可用于拉式和推式内容。

**专业化**

DITA的专门化特性，允许从现有类型创建新的元素类型和属性。这有助于交换一致的DITA内容，并确保对所有DITA内容进行最低级别的通用处理。它还允许支持专门化的处理器添加专门化特定的处理。

面向专业领域的编辑出版，其基本思想是“求专不求全”，较之通用出版物涵盖各类编辑要素的要求有很大区别。专业领域的出版物针对领域特定的出版要求，定义对应的 DITA 主题要素，将通用的主题标签细化为领域专用的术语标签。在 DITA 专门化的过程中，首先要由领域专家分析并建立面向领域出版的要素模型，之后根据模型定义主题标签并形成 DTD 模板。经过领域专门化的DITA 主题，继承原有 DTD 标记的行为和属性，并赋予新标签更高的可读性，能明确地表义主题的内容模块和逻辑层次。与面向对象的编程语言思想类似，出版物的标记定义根据需求进行变更时，仅改变专门化标签的基类即可实现对标记的重定义，而无需分别维护每一个已实例化的主题。DITA 领域专门化使得新的要素定义建立在已有要素定义之上，并且新定义的要素可以使用已有处理规则进行处理。面向领域的 DITA 专门化是对 DITA 主题更高层次的抽象与复用，面向报纸、期刊、工具书、教科书等特定领域的出版从业者定义行业出版物标记，细化主题包含的概念、任务和引用，应用在各领域 DITA主题的生成中。

除 DITA 主题专门化之外，DITA 还支持映射文件专门化。DITA 映射可以面向映射领域实现专门化，通常是将引用主题的<topicref>标签专门化为映射领域，并可以在多种不同映射类型中实现设计模式重用。专门化的<topicref>标签限定了对特定类型主题的引用，如<conceptref>标签引用概念主题，<stepref>标签引用描述操作步骤主题，<summaryref>标签标识提供集合总结的主题。专门化的映射类型保证主题集合符合目标出版物的组织结构，在帮助出版物减少信息预处理、明确信息类型、通过专门化引用保持信息相容性、设计和处理流程的重用等方面具有很高的实用价值。

**从单一来源产生不同的可交付成果**

DITA旨在从一组DITA内容中生成多个可交付格式。这意味着许多格式副本详细信息既没有在DITA规范中指定，也没有在DITA内容中指定；**格式副本详细信息**由**处理器**定义和控制。

与许多用于人类可读文档的基于XML的应用程序一样，DITA支持将内容与表示分离。当内容在不同的上下文中使用时，这是必要的，因为作者无法预测他们所创作的材料将如何使用或在何处使用。以下功能和机制使用户能够从单个来源生成不同的可交付结果格式：

DITA地图

不同的DITA映射可以针对不同的交付格式进行优化。例如，可能有一个用于打印输出的图书映射和另一个用于生成联机帮助的DITA映射；每个映射使用相同的内容集。

专业化

DITA专用工具允许用户创建XML元素，这些元素可以提供适当的格式副本区别。由于专业化的使用不会妨碍交换或互操作性，DITA用户可以安全地创建其本地交付和呈现要求所需的专业化，对依赖或使用内容的系统和业务流程的额外影响最小。虽然一般的XML实践建议元素类型应该是语义的，但是专门化可以用来定义本质上纯粹表示的元素类型。突出显示的域就是这样一个专门化的例子。

条件处理

条件处理使DITA主题或包含特定于传递的内容的映射成为可能。

内容引用

conref机制能组合通用组件和特定于交付上下文的组件，从而构造特定交付的映射或主题。

键引用

keyref机制使关键字可以在不同的可交付结果中以不同的方式显示。它还允许单个链接解析到不同可交付结果中的不同目标。

@outputclass属性

@outputclass属性提供了一种机制，作者可以在必要时指示特定的格式副本意图。注意，DITA规范没有为@outputclass属性定义任何值；使用@outputclass属性是特定于处理器的。

DITA的设计初衷是构建一种基于主题的技术文档创作模式，并以主题为基础来创作、组织和链接信息内容。所谓主题，在技术信息中有时称为一个条目 (article)，它具有标题和内容。主题是独立的信息单元，一个高质量的主题仅包括一个讨论对象。每个主题既要足够长以保证有其独立存在的价值，又要保持精简(“简约”理念)以确保仅集中于讨论一个对象，而不延伸到别的问题。在多数与产品、服务或技术一起提供的技术文档中，主题应该从属于一组经过有序组织的主题集。这一主题集能够以HTML网页、在线帮助或者PDF手册等多种形式打包输出。

**Dita的文件格式**

**DITA topics**

* \*.dita (preferred)
* \*.xml

**DITA maps**

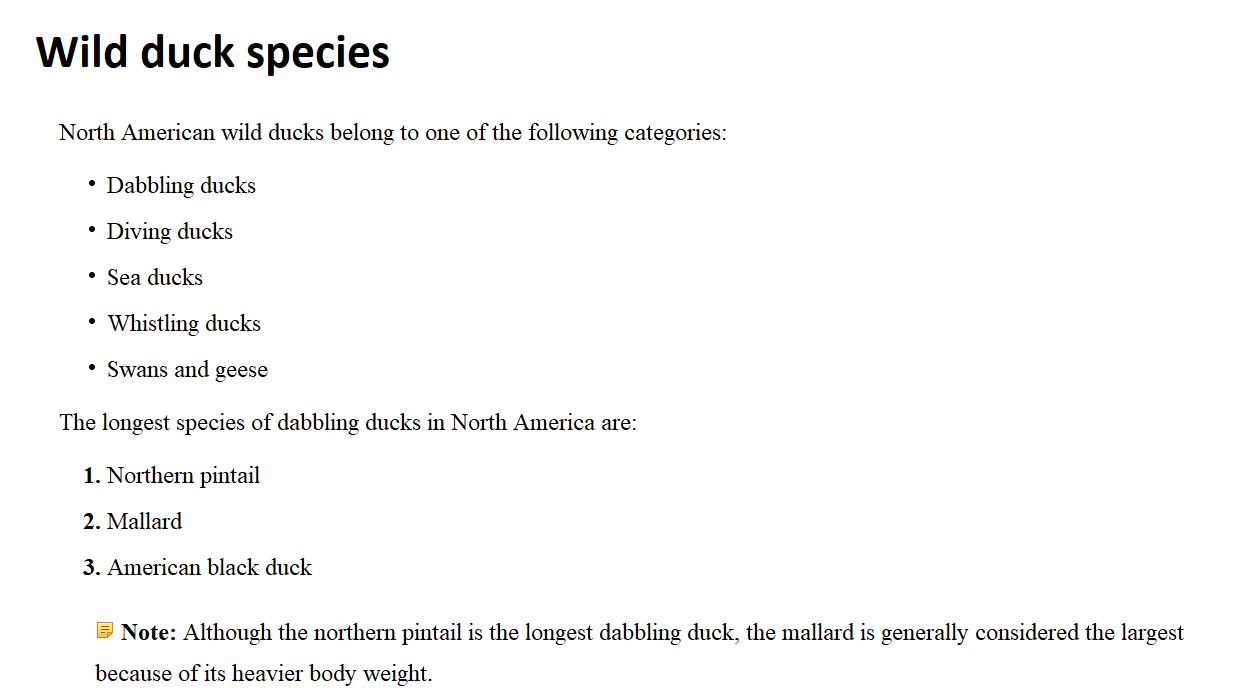
\*.ditamap

**Conditional processing profiles**

\*.ditaval

* **DITA concept 学习**

<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE concept PUBLIC "-//OASIS//DTD DITA Concept//EN" "concept.dtd">  
<concept id="my\_first\_concept">  
 <title>Wild duck species</title>  
 <conbody>  
  
 <p>North American wild ducks belong to one of the following categories:</p>  
 <ul>  
 <li>Dabbling ducks</li>  
 <li>Diving ducks</li>  
 <li>Sea ducks</li>  
 <li>Whistling ducks</li>  
 <li>Swans and geese</li>  
 </ul>  
 <p>The longest species of dabbling ducks in North America are:</p>  
 <ol>  
 <li>Northern pintail</li>  
 <li>Mallard</li>  
 <li>American black duck</li>  
 </ol>  
 <note>Although the northern pintail is the longest dabbling duck, the mallard is generally  
 considered the largest because of its heavier body weight.</note>  
 </conbody>  
</concept>



* **DITA task 学习**

DITA定义了两种任务主题类型：严格任务（strict task）和常规任务（general task）。

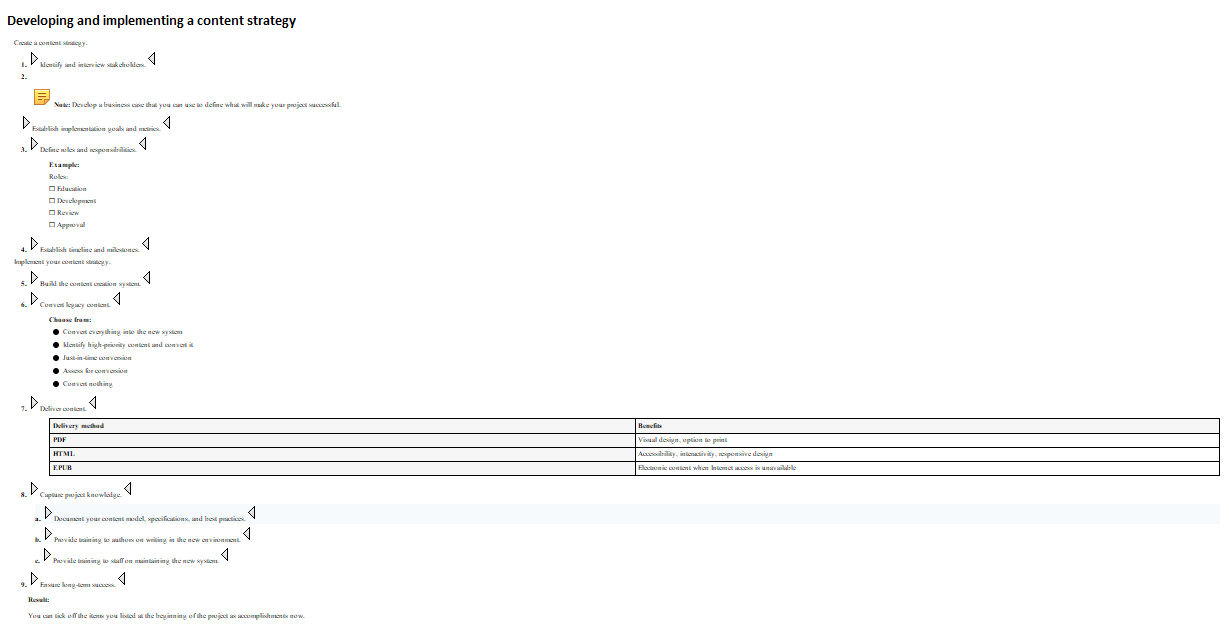
strict task主体中的主要元素必须按明确的顺序排列，且只能包含一组步骤。另外，用户操作的步骤必须使用明确的元素（<steps>和<step>）。

就元素排序而言，general task不如strict task要求严格。general task中允许有一组或多组步骤。步骤中使用的元素可以是DITA <section>元素所允许包含的任何元素。

尽管听起来编写一个general task更容易些，但在不少情况下，使用strict task却能带来一系列的好处。这些包括：

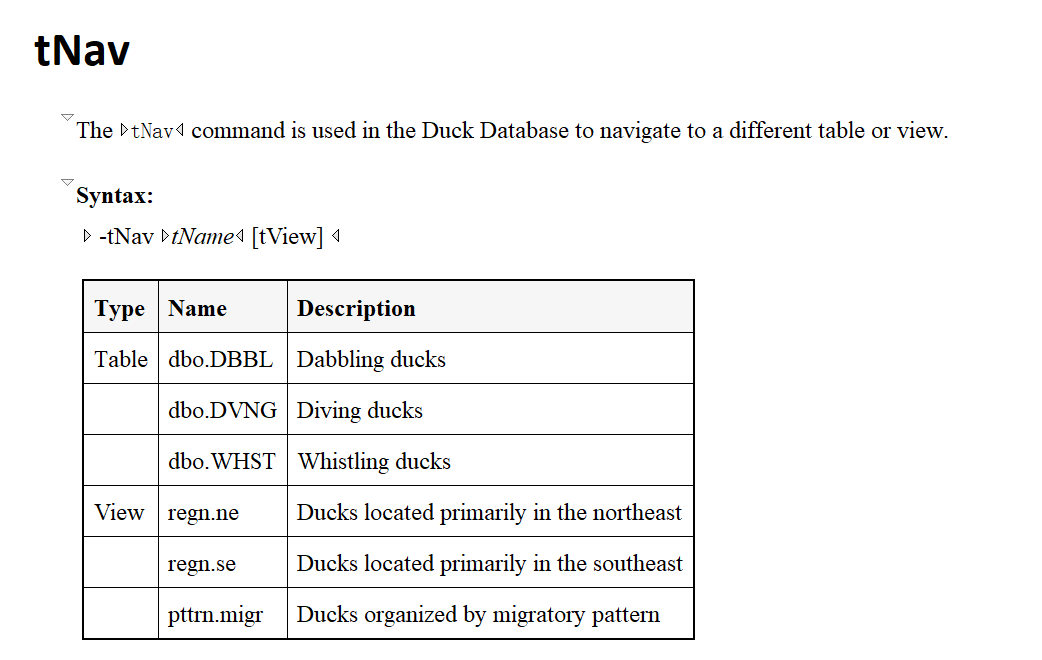
1. 一致性。你创作的单个文档或多个文档中所有的任务主题使用相同的组织形式。
2. 可靠性。如果组织形式保持一致，用户可放心地在相同的位置查找相似的信息。
3. 易于创作。DITA编辑器能引导作者到下一个需要添加的元素。
4. 简洁性。多数任务仅要求有限的步骤就可完成。
5. 语义性。在strict task中使用<steps>和<step>元素能高质量地传达信息，体现其语义价值。

<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE task PUBLIC "-//OASIS//DTD DITA Task//EN" "task.dtd">  
<task id="task\_tlf\_3h2\_jt">  
 <title>Developing and implementing a content strategy</title>  
 <taskbody>  
 <steps id="steps\_xmc\_nh2\_jt">  
 <stepsection>Create a content strategy.</stepsection>  
 <step>  
 <cmd>Identify and interview stakeholders.</cmd>  
 </step>  
 <step>  
 <note>Develop a business case that you can use to define what will make your project  
 successful.</note>  
 <cmd>Establish implementation goals and metrics.</cmd>  
 </step>  
 <step>  
 <cmd>Define roles and responsibilities.</cmd>  
 <stepxmp>  
 <lines>Roles:  
☐ Education  
☐ Development  
☐ Review  
☐ Approval</lines>  
 </stepxmp>  
 </step>  
 <step>  
 <cmd>Establish timeline and milestones.</cmd>  
 </step>  
 <stepsection>Implement your content strategy.</stepsection>  
 <step>  
 <cmd>Build the content creation system.</cmd>  
 </step>  
 <step>  
 <cmd>Convert legacy content.</cmd>  
 <choices id="choices\_zwl\_4h2\_jt">  
 <choice>Convert everything into the new system</choice>  
 <choice>Identify high-priority content and convert it</choice>  
 <choice>Just-in-time conversion</choice>  
 <choice>Assess for conversion</choice>  
 <choice>Convert nothing</choice>  
 </choices>  
 </step>  
 <step>  
 <cmd>Deliver content.</cmd>  
 <choicetable frame="all" relcolwidth="1.0\* 1.0\*" id="choicetable\_g43\_ph2\_jt">  
 <chhead>  
 <choptionhd>Delivery method</choptionhd>  
 <chdeschd>Benefits</chdeschd>  
 </chhead>  
 <chrow>  
 <choption>PDF</choption>  
 <chdesc>Visual design, option to print</chdesc>  
 </chrow>  
 <chrow>  
 <choption>HTML</choption>  
 <chdesc>Accessibility, interactivity, responsive design</chdesc>  
 </chrow>  
 <chrow>  
 <choption>EPUB</choption>  
 <chdesc>Electronic content when Internet access is unavailable</chdesc>  
 </chrow>  
 </choicetable>  
 </step>  
 <step>  
 <cmd>Capture project knowledge.</cmd>  
 <substeps id="substeps\_fhj\_qh2\_jt">  
 <substep>  
 <cmd>Document your content model, specifications, and best practices.</cmd>  
 </substep>  
 <substep>  
 <cmd>Provide training to authors on writing in the new environment.</cmd>  
 </substep>  
 <substep>  
 <cmd>Provide training to staff on maintaining the new system.</cmd>  
 </substep>  
 </substeps>  
 </step>  
 <step>  
 <cmd>Ensure long-term success.</cmd>  
 <stepresult>  
 <p>You can tick off the items you listed at the beginning of the project as dita  
 accomplishments now.</p>  
 </stepresult>  
 </step>  
 </steps>  
 </taskbody>  
</task>



* Dita reference 学习

<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE reference PUBLIC "-//OASIS//DTD DITA Reference//EN" "reference.dtd">  
<reference id="my\_first\_ref">  
 <title>tNav</title>  
 <refbody>  
 <section>  
 <p>The <cmdname>tNav</cmdname> command is used in the Duck Database to navigate to a  
 different table or view.</p>  
 </section>  
 <refsyn>  
 <synph>  
 -tNav <var>tName</var> [tView]  
 </synph>  
 </refsyn>  
 <properties>  
 <prophead>  
 <proptypehd>Type</proptypehd>  
 <propvaluehd>Name</propvaluehd>  
 <propdeschd>Description</propdeschd>  
 </prophead>  
 <property>  
 <proptype>Table</proptype>  
 <propvalue>dbo.DBBL</propvalue>  
 <propdesc>Dabbling ducks</propdesc>  
 </property>  
 <property>  
 <proptype/>  
 <propvalue>dbo.DVNG</propvalue>  
 <propdesc>Diving ducks</propdesc>  
 </property>  
 <property>  
 <proptype/>  
 <propvalue>dbo.WHST</propvalue>  
 <propdesc>Whistling ducks</propdesc>  
 </property>  
 <property>  
 <proptype>View</proptype>  
 <propvalue>regn.ne</propvalue>  
 <propdesc>Ducks located primarily in the northeast</propdesc>  
 </property>  
 <property>  
 <proptype/>  
 <propvalue>regn.se</propvalue>  
 <propdesc>Ducks located primarily in the southeast</propdesc>  
 </property>  
 <property>  
 <proptype/>  
 <propvalue>pttrn.migr</propvalue>  
 <propdesc>Ducks organized by migratory pattern</propdesc>  
 </property>  
 </properties>  
 </refbody>  
</reference>



## DITA出版流程[¶](https://techcomm-book.readthedocs.io/zh_CN/latest/5-info-develop/dita_intro.html" \l "id4" \o "永久链接至标题)

在DITA定义的出版流程中，内容组织的最细粒度单元是以XML格式描述的结构化内容模块。这种内容模块在DITA标准中被称作主题(Topic)。主题是能够自我描述且按照单一逻辑范畴组织的内容信息。根据出版物的结构组织要求，描述相同对象的主题通过对象映射(Mapping)机制进行逻辑顺序组织，形成内容完整的统一体。组织完成的出版内容经样式渲染(Rendering)，形成交付终端展示的数字出版物。

在数字出版的内容流转过程中，备选内容存储在内容仓库中。内容仓库是以XML格式为代表的非结构化数据存储的容器，例如以文档存储和检索为中心的领域专用数据仓库MarkLogicServer。内容仓库专门针对半结构化和非结构化数据进行设计和优化，能够实现TB级非结构化数据资源的全文检索。

在数据模型组织方面，内容仓库采用XML树状结构组织，数据查询和检索使用的DML(Data Manipulation Language)和DDL(DataDefinitionLanguage)语言为XQuery查询。 抽取自内容仓库的信息单元根据DITA标准定义的标签进行格式化，形成描述同一信息主体的主题块。构成同一出版物的不同主题块围绕出版物内容展开描述，相互之间具备一定的顺序或层级关系。不同主题块之间的松耦合关联由DITA映射维系，DITA映射描述不同主题块相互组合的层级关系和先后逻辑顺序。在DITA映射的黏合作用下，分散的主题块被重组成为具备逻辑关系的结构化文档。结构化文档包含形成交付出版物的所有内容元素，内容被重新组合成为具备逻辑层次关系和先后顺序的有机整体。

在形成数字出版物之前，结构化文档经由可扩展样式表语言(Extensible Stylesheet Language，XSL)及扩展样式转换语言(Extensible Stylesheet LanguageTransformations，XSLT)进行样式渲染，成为具备排版格式且样式美观的出版物。在渲染过程中，数字出版物能够根据出版需求，生成各种XML能够转换形成的目标出版格式，如PDF、RTF或HTML等，由展示终端提供给用户阅读。

* **DITA目前**[¶](https://techcomm-book.readthedocs.io/zh_CN/latest/5-info-develop/dita_intro.html" \l "id8" \o "永久链接至标题)**应用分析**

DITA在国内外IT技术类企业中应用较为广泛,具体应用领域包括技术文档编写、数字出版,企业信息化(知识管理、内容管理、文档管理)等。国际上已经存在一些成熟的应用和相应的解决方案,如PTC公司的Arbortext[6]、Adobe公司的FrameMaker[7]、JustSystems公司的XMetal[8]等。除了商业解决方案之外,也存在一些支持DITA的开源工具,如DITAOpen Toolkit是一款基于JAVA技术的开源引擎,为DITA文档解析与转换提供技术支持[9]。虽然DITA具有面向主题、模块化复用、灵活输出等诸多优点,但结构化写作的技术模式对传统内容创作而言,是一种写作范式上的巨大转变。人们已经习惯于在文字处理软件撰写文档,虽然商业解决方案都力图在用户体验界面上简化和方便用户的使用,但DITA本身仍然具有较高的技术门槛,因此没有被广泛接受。数字内容存储形式多种多样,从图书馆数字内容组织和管理的观点来看,使用DITA对庞大的数字资源进行描述和转化的想法是不现实的,因此界定DI-TA的应用范围非常重要。

企业知识管理实践中从内容生产的源头处介入,应用DITA方案是一条行之有效的内容管理途径。

在数字图书馆知识库建设中,对于是否考虑采用DITA进行数字内容的结构化写作和发布,以下几点可供决策参考:

a.内容交换的需要。XML是目前内容交换的一种优选格式。使用DITA制作的XML内容文件,可以方便地进行共享与交换,并为机器可读可理解的自动化处理机制打下基础。

b.内容结构与重用。考查内容的逻辑结构,像文学体裁之类的创造性写作,本身不具有严谨成型的结构模式,而且文学内容基本没有重用性的考虑,这就不适合采用DITA方法。DITA方法的一个核心理念是主题的重用(Reuse),内容重用意味着主题内容本身具有一定的参考和利用价值,因此DITA处理的内容应该是选择那些高价值的知识或情报内容。

c.大规模内容生产与管理。需要处理的内容数量较大时,可以考虑采用DITA的结构化处理机制。为内容预先设定结构模式,DITA使整个内容生产流程系统工程化和自动化,提高了内容管理的质量和效率。如果数字内容数量较少,则没有必要特意设计内容结构,否则采用DITA方法的投入会超过内容结构化处理所带来的好处,起不到事半功倍的效果。

d.协同写作与内容风格。多位作者同时为一个文档项目编制内容,DITA的结构化预设、内容与表现分离的处理方式,使得作者们在同一套结构化文档规则下,集中精力进行内容的写作和维护,保证了内容风格的一致性。排版和表现方面则由样式模板的XSLT转换机制进行统一化管理,这样能够有效提高内容写作上的协同质量。

e.服务要求复杂与个性化定制需求。对于复杂的用户需求,DITA流程可以实现内容“组装”到各类样式输出的快速响应,提供个性化定制的知识产品。

图书馆资源与服务指南(以下简称服务指南)是介绍图书馆资源和服务的小册子,能够帮助用户了解和更好地利用图书馆提供的各种信息资源和服务。图书馆的规模越大,采购的数据库资源越来越多,服务方式更加多元化的情况下,资源服务指南的编制会随之变得越来越复杂,维护成本也会不断增加。通常服务指南是一份篇幅冗长的PDF电子文稿或HTML网页。对于用户而言,他们一般仅仅想了解对自己有帮助的资源和服务类型,而不是费力翻阅整本服务指南后过滤出自己想知道的,无疑增加了用户的学习负担。对于频繁变动的资源和服务情况,及时进行修订和告知用户,这是当前服务指南编制模式所不能有效支持的。运用DITA方法对服务指南内容进行解构,进而建立起一个服务指南主题仓库。通过单源维护、多元化发布机制来改进服务指南的管理和利用效率。这种思路试图将传统平面线性化的文字写作进化到一个立体多维的知识系统工程,从撰写一本小册子进化到支撑图书馆服务的知识库建设的一部分,服务指南最终变为由知识库驱动的针对不同用户群体需求的动态输出。下面以中科院国家科学图书馆的资源与服务指南为例,讨论DITA应用的可行性。中科院国家科学图书馆截至目前已开通180余个文献数据库,建设了若干服务系统,其资源种类与服务方式较为丰富。服务指南的内容主要包括国家科学图书馆资源与服务简介、如何查找资料、如何利用服务、查询技巧、合理使用资源和术语介绍六个主要部分和三个附录,按照“提问-回答”方式组织整理。

根据DITA的基本主题类型和服务指南的编写体例,大体可将内容划分为四个主题类型:

a.概念主题:介绍图书馆资源与服务,方便用户使用图书馆服务之前了解相关背景,具体包括资源与服务体系简介、合理使用资源等;

b.任务主题:如何利用图书馆完成某个特定信息查询任务的详细步骤与方法,具体包括查找资料、查找技巧等;

c.参考主题:包括常用服务联系方式等;

d.术语主题:包括术语介绍等。主题的编写遵守DITA主题类型结构和相关元素,保证了内容结构的一致性,方便管理与利用。任务主题“查找网络数据库”在< oXygen/> XML Author工具中的编写示例,左侧是大纲设图,显示了文档结构及相关元素,右侧是所见即所得的内容编辑窗口。

利用DITA这四类主题的结构对服务指南内容进行重新组织,将每个主题集中存入主题仓库。之前面向用户的都是同一份完整的服务指南,运用DITA方法在输出完整的服务指南基础上,根据用户的个性化需求输出各种“衍生品”。学科馆员通过DITA映射机制,按照学科服务的六个领域(资源环境、能源工程、信息光电、数理科学、化学化工和生命科学等)从主题仓库中调出相关主题,有针对性地制作符合特定学科用户需求的个性化服务指南产品,主动推送给读者。服务指南的动态性和个性化能够在一定程度上保证学科服务的质量和及时做出响应。由于所有的资源和服务内容都转化成DITA主题后存入DITA库,服务指南的局部修改和维护工作也变得相对轻松和容易。

关于DITA项目中人物角色的管理方面,这里定义三个人物角色:首席学科馆员,学科馆员和模板设计者。首席学科馆员负责服务指南的整体设计与规划;学科馆员根据自己负责的学科或科研院所对应的资源与服务内容按照四种预设主题结构类型进行撰写和负责相关的维护工作;模板设计者属于技术支撑人员,依据服务指南的输出形式要求,设计出样式模板,这里要求掌握XSL等相关技能。这样的处理使得学科馆员不必在内容撰写时考虑排版细节问题,这些都交由样式模板自动处理。

服务指南运用DITA方法进行改造的劣势也比较明显。一开始要求培训学科馆员使用XML编辑工具进行结构化内容创作,而且需要掌握DITA的一系列标记语法,整体使用门槛较高。选择一些商用的类似文字处理软件操作界面的所见即所得的DITA编辑工具可以在一定程度上降低使用门槛,但DITA理念的接受和理解过程是不可避免的。另外DITA在内容数量越多的情况下优势和效益体现得越明显,而对于中小型图书馆而言,自身订购的信息资源种类和提供的服务方式较少,服务指南的撰写与维护工具就相对简单一些,传统的文字编辑工具即可胜任。这类情况没有必要采用DITA方法,否则就把简单问题复杂化了,因此要根据图书馆资源数量和服务手段的具体情况,谨慎做出决策。

* **DITA与DOCBOOK对比**[¶](https://techcomm-book.readthedocs.io/zh_CN/latest/5-info-develop/dita_intro.html" \l "ditadocbook" \o "永久链接至标题)

DITA和DocBook是数字出版领域的两种标准，通过定义规范化的文档描述规则，来解决文档交付过程中遇到的问题。面向不同类型的交付出版物，DITA和DocBook各有见长，但在实际应用中也有自身的限制因素。

DITA解决了出版物的结构化描述和内容重组问题，且支持多语言版本制作，适用于对格式有严格限定的技术手册类出版物。但DITA不能实现很完美的样式渲染，且对于内容与格式一体化的复杂出版物，DITA很难进行主题和界定与划分。所以使用DITA进行书籍出版的成本和难度较高。

相比较来说，DocBook适用于通用出版物，文档易于组织和排版。但DocBook内容以Section段落组织，不具备DITA的内容映射机制，无法做到类似Topic这样粒度的内容划分与重组。且对于内容需要频繁修改的文档排版，Docbook略显力不从心。 DITA和DocBook专注于交付技术信息，但DITA侧重于交付主题，而DocBook侧重于交付书籍。DITA提供基于主题级粒度的信息分类，允许作者组织并描述特定信息领域。在生成多种文档格式的信息重用过程中，能够保持内容的高度一致性。在最终交付物的输出格式方面，DITA能够生成 PDF、CHM、HTML等大部分的出版交付类型。DocBook常用的交付格式为PDF和HTML，其他输出格式需要借助相关的功能插件。