

Modeling and Application of a General Questionnaire Solution Based on DSL

Yuzeng Chang, Guangyan Lin, Hongyu Fang, Tantan Man

College of Software, Beihang University, Beijing
Email: cyz@buaa.edu.cn

Received: Nov. 2nd, 2016; accepted: Nov. 22nd, 2016; published: Nov. 25th, 2016

Copyright © 2016 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Based on business area analysis and concept extraction to the survey questionnaires, using XML and Backus-Naur Form, the paper defines a domain specific language which is suitable for questionnaire, and then with template engine and XSLT generates the ultimate code, resulting in a relatively complete survey questionnaire generation solution. At last, a practical demonstration is given out for the verification and retrospection of the solution.

Keywords

Questionnaire, Domain Specific Language, Domain Analysis, Concept Extraction, Object Code Generation

基于DSL的通用调查问卷建模和应用

常玉增, 林广艳, 方弘宇, 满坦坦

北京航空航天大学软件学院, 北京
Email: cyz@buaa.edu.cn

收稿日期: 2016年11月2日; 录用日期: 2016年11月22日; 发布日期: 2016年11月25日

摘要

本文对调查问卷业务进行领域分析和概念抽取, 基于XML、巴科斯范式定义了一种适用于调查问卷的领域特定语言, 并结合模板引擎和XSLT技术生成了最终目标代码, 由此形成一套较完整的调查问卷生成方案。最后通过案例对该方案进行了验证。

关键词

调查问卷, 领域特定语言, 领域分析, 概念抽取, 目标代码生成

1. 引言

由于无地域限制、低成本等因素, 在线调查问卷已经成为目前最普遍的信息采集手段之一。然而设计实现调查问卷系统并非易事。倪浪[1]提出基于模板的动态表单技术来实现问卷制作功能, 但模板的元数据是保存在数据库中, 并需用 PHP 解析后才生成页面, 这种方式对于问卷的多变性支持较差, 且每次都需耗费时间来生成问卷页面。陆茜茜[2]把问卷定制分为出题、组卷两个步骤, 虽然逻辑很清晰, 但留给业务人员的可定制空间较小。由此可见, 若使用通用型语言(如 C++、Java)描述调查问卷领域, 当业务逻辑复杂时, 会造成代码结构庞大、判断逻辑复杂, 这会降低开发人员效率, 并且客户难以通过代码了解开发者思路, 为客户和开发者的交流造成障碍。

针对上述问题, 本文提出一种面向问卷的领域特定语言, 并以该语言为辅助生成了目标代码, 从而弥补通用语言的不足。领域特定语言(Domain Specific Language, DSL), 是针对某一特定领域的、具有受限表达性的一种计算机程序设计语言, 也被称为应用型语言[3] [4]。使用针对调查问卷的 DSL, 可以在开发者和领域专家之间建立起一座桥梁, 使开发者能迅速的构建出满足客户需求的软件。本文旨在设计并实现一种调查问卷生成方案, 主要是通过构建调查问卷领域的 DSL, 借助该 DSL 进行编程, 并最终生成目标代码, 发挥了提高软件开发效率和产品质量的作用。

2. 方案提出

本文的调查问卷生成方案主要包括三个环节: 概念抽取、DSL 定制以及模板填充和目标代码生成, 如图 1 所示。其中概念抽取环节的主要工作是领域界定与分析; DSL 定制环节需经过 DSL 元模型构建, 再使用巴科斯范式进行语法描述; 模板填充环节中, 需针对需求使用该 DSL 进行描述, 并结合模板进行填充以生成目标代码。

3. 概念抽取

分析领域是设计 DSL 的前提, 首先需要认识问题领域, 其次对领域进行概念抽取并划分问题空间, 最终针对问题空间设计 DSL, 但 DSL 的设计并不是一个简单的串行过程, 需要不断回溯验证[3]。DSL 涉及内容包含语法、语义、案例、方法、工具等[5], 如图 2 所示。

3.1. 领域对象的概念抽取

在领域分析过程中, 通常由领域专家提取特定领域的关键概念, 创建特定领域的元模型。本研究涉及的领域为问卷调查领域, DSL 设计的目的是为了生成在线调查问卷的题目, DSL 需要描述不同类型和选项数量的题目。

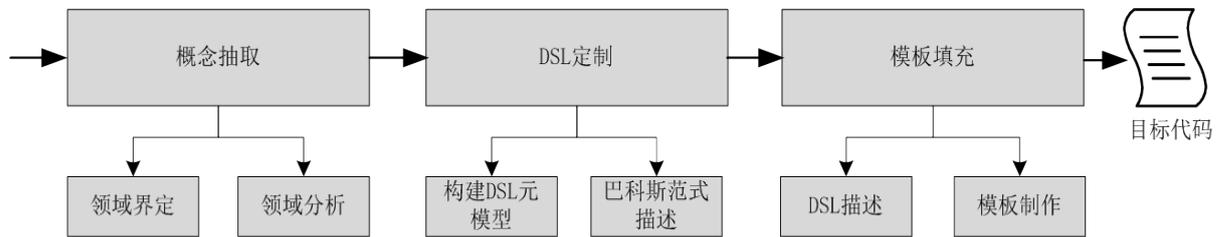


Figure 1. Solution of questionnaire generation

图 1. 调查问卷生成方案概览

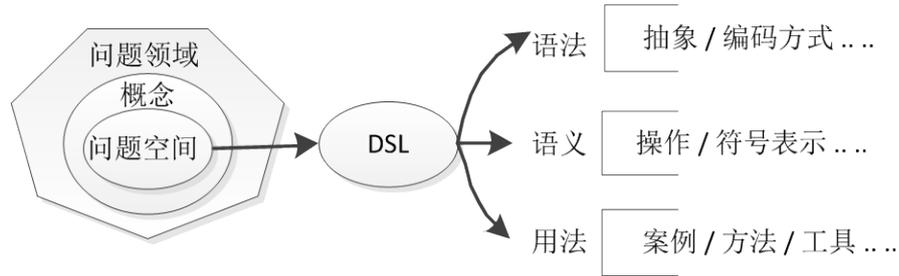


Figure 2. Flow of analysis of the domain and customization of DSL

图 2. 从分析问题领域到定制 DSL

每个问卷下包含若干个、数量不等的题目。题目包括单项选择题、多项选择题、填空题、提示语句、选择填空题、矩阵题等 6 种类型，图 3 是一道矩阵题目的样例。

3.1.1. 题目的基本结构

虽然调查问卷的题目类型不同，但题目间具有一些共同的基本结构：

- 1) 包含一个题干，用于对问题的总体描述；
- 2) 包含一个题目类型属性，用于标记该题目所属的类型；
- 3) 包含一个至多个选项个数，用于用户作答；
- 4) 包含一个必填选型属性，用于标记该题目的作答是必须还是非必须。

对题目共同基本结构的总结有助于抽象对题目的描述。

3.1.2. 题目的其他特性

题目根据用户的要求会具有一些特殊的性质，主要表现为：

- 1) 逻辑关联性。题目之间会有逻辑关系，题目的不同答案会关联其他题目。
- 2) 问题嵌套。一道问题里可能包含多道其他问题，以矩阵题目为例，整个矩阵可视为一道题目，而每一行可视为被包含的小题。

3.2. 界面展示的概念抽取

包含各类题目的问卷的目标代码是 HTML，展示在浏览器上，题目按题号顺序排列；对于有选择项的题目，提供选择框控件；对于有填空项的题目，提供输入框控件。

用户在终端界面作答时，系统代码会对其输入进行判断，并根据判断结果采取对应的交互操作：

- 1) 题目的输入答案具有校验规则，如只能选择一个答案；
- 2) 题目具有必须填写的属性，会要求用户在填写完毕后才能提交；
- 3) 题目和其他题目具有关联关系，会跳转到关联题目。

您对北京市禁烟活动的看法

	优秀	良好	合格
执行力度	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
执行效果	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
群众反映	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure 3. An example of matrix type of questionnaire

图 3. 调查问卷的矩阵型题目样例

3.3. 数据访问的概念抽取

在数据输入方面, 问卷内容根据题目模板和用户定制的具体内容共同生成。题目模板由业务员编写; 题目具体内容由问卷编辑人员通过浏览器在网页上填写, 数据被收集后封装成指定数据结构, 通过模板引擎与题目模板结合。

在数据存储方面, DSL 模板、模板引擎生成的 DSL 文件以及 HTML 目标代码均以文本文件的方式存储于系统磁盘上, 对于其中频繁访问的文件可以使用数据库和缓存存储以优化性能。

4. DSL 定制

本论文设计的 DSL 基于 XML, 因为使用 XML 具有两个优势: 第一, XML 可通过 XML Schema 验证其内容合法性, 故 XML Schema 可以作为 DSL 的元模型来规范语法, 减少编写编译器检查语法的环节 [6]; 第二, XML 拥有相关规范能将 XML 文档直接转化为目标代码——HTML 文档, 减少编写解析器转化代码的环节。

4.1. 使用 XML Schema 制作 DSL 元模型

本文以题目为粒度划分调查问卷, 以题目为描述对象设计 DSL 的元模型 [7] [8]。DSL 的使用者可将多个描述对象组合成一个调查问卷。

4.1.1. 定义 question 类型描述题目

通过领域分析抽取 question 类型的属性如表 1 所示。

其中, qItems 表示对题目选项的描述, 是复杂类型, 将在 4.1.2 小节给出其数据结构的描述。subQs 表示子题目, 因为题目可能具有嵌套性而包含其他题目, 所以设计时采用自引用结构, 子题目的类型也是 question 类型。fQRef 标记了包含本题目的父类题目的 ID, 该属性的引入实现 question 类型自引用的解决方案, 通过类似数据库外键的机制关联 question 类型对象。ifNecessary 字段标记了题目是否必须作答, 它是枚举类型, 将在 4.1.3 小节给出其数据结构的描述。

question 类型的属性字段和描述以及其关联关系如图 4 所示。

4.1.2. 定义 item 类型描述题目选项

通过领域分析抽取 item 类型的属性如表 2 所示。

其中, value 属性限定了题目选项取值范围, 是枚举类型, 将在 4.1.3 小节给出其描述。logicQRef 记录了与该选项有逻辑关联的题目, 用户作答该选项后, 会根据题目 ID 跳转到关联题目。parentQRef 表示选项所属的题目。

item 类型的属性字段和描述以及其关联关系如图 5 所示。

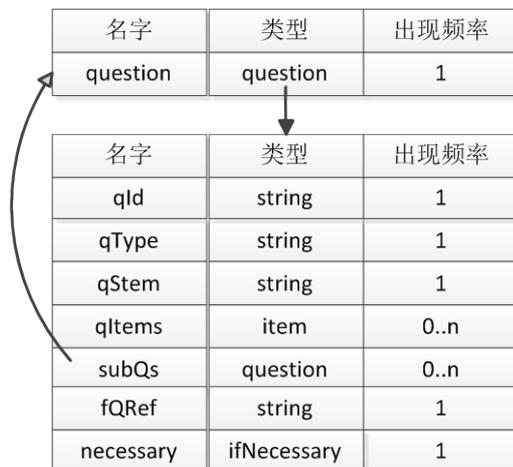


Figure 4. The data structure of *question* type

图 4. Question 类型的数据结构

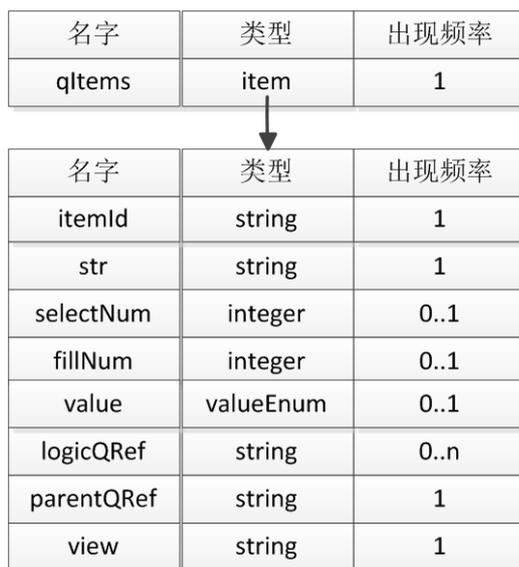


Figure 5. The data structure of *item* type

图 5. Item 类型的数据结构

Table 1. The fields of *question* type

表 1. Question 类型的字段名称及释义

字段名称	释义
qId	题目的唯一性标识
qType	标记题目属于哪种题型
qStem	题干, 即题目的描述部分
qItems	对题目选项的描述
subQs	本题目包含的子题目
fQRef	父类题目 ID 的引用
ifNecessary	标记题目是否是必答题

Table 2. The fields of *item* type
表 2. Item 类型的字段名称及释义

字段名称	释义
itemId	选项的唯一性标识
str	选项的文字描述
selectNum	选择框组件个数
fillNum	填空框组件个数
value	选项的取值范围
logicQRef	选项逻辑关联的题目的 ID
parentQRef	本选项所属的题目的 ID
view	选项用户界面层次的显示

4.1.3. 定义枚举类型约束 DSL 语法

枚举类型主要针对 question 类型中的 necessary 以及 item 类型中的 value 属性，XML Schema 定义的枚举类型如图 6 所示。

综上所述，借助 XML Schema，本文建立了包含 question 类型、item 类型和枚举类型的 DSL 的元模型，该元模型充当 DSL 的规范。至此使用 XML Schema 定制调查问卷 DSL 的过程完毕。客户和开发人员可以使用该 DSL 进行问卷描述和开发。DSL 的完整元模型如图 7 所示。

4.2. 调查问卷 DSL 的巴科斯范式描述

巴科斯范式(Backus-Naur Form, BNF)，是用于描述计算机语言语法的符号集，本文使用巴科斯范式描述调查问卷 DSL，声明 DSL 语法，巴科斯范式符号声明如表 3 所示。

调查问卷 DSL 的巴科斯范式如图 8，图 9 及图 10 所示。

5. 模板引擎使用与目标代码生成

DSL 定制完成后，相关人员可以使用该 DSL 进行业务需求描述，但只是产生了中间结果，还需进一步转换为最终结果，即目标代码。生成目标代码的主要逻辑是：编写 DSL 代码模板，使用相应的引擎填充 DSL 代码模板，并把生成的中间文件最终转化为目标代码[9]。

本节以 DSL 的设计和 DSL 代码模板制作为基础，以模板引擎和相关规范为途径[10]，阐述目标代码生成方案。

5.1. 使用引擎填充 DSL 模板

本文使用基于标签替换原理的模板引擎 FreeMarker 制作代码模板，标签替换是根据标记的名称在 DSL 中查找相应的值进行替换而生成具体代码。FreeMarker 通过\${}与#指令标记代码模板的特定部分。\${}引用动态数据，#指令建立模板语言的控制结构，如选择、循环等。FreeMarker 标签替换示意如图 11 所示。

模板制作人员将 DSL 编写的文件存储成 ftl 格式就可以使用 FreeMarker 对其进行生成，其工作流程如图 12 所示。这个过程主要借助于 Java 原生的 SAX 接口和开源 XML 解析库 DOM4J。其中，读入环节是通过创建 SAX 读取器，解析相应路径上的 DSL 模板，从而得到一个 Document 对象；验证环节，创建 SAX 解析器，指定 Schema、XSD、异常处理器等属性，然后对读入环节的 Document 对象进行校验，若

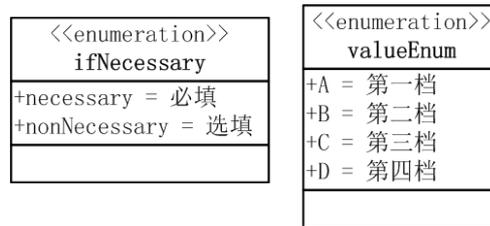


Figure 6. The enumeration types defined by XML Schema
图 6. XML Schema 定义枚举类型

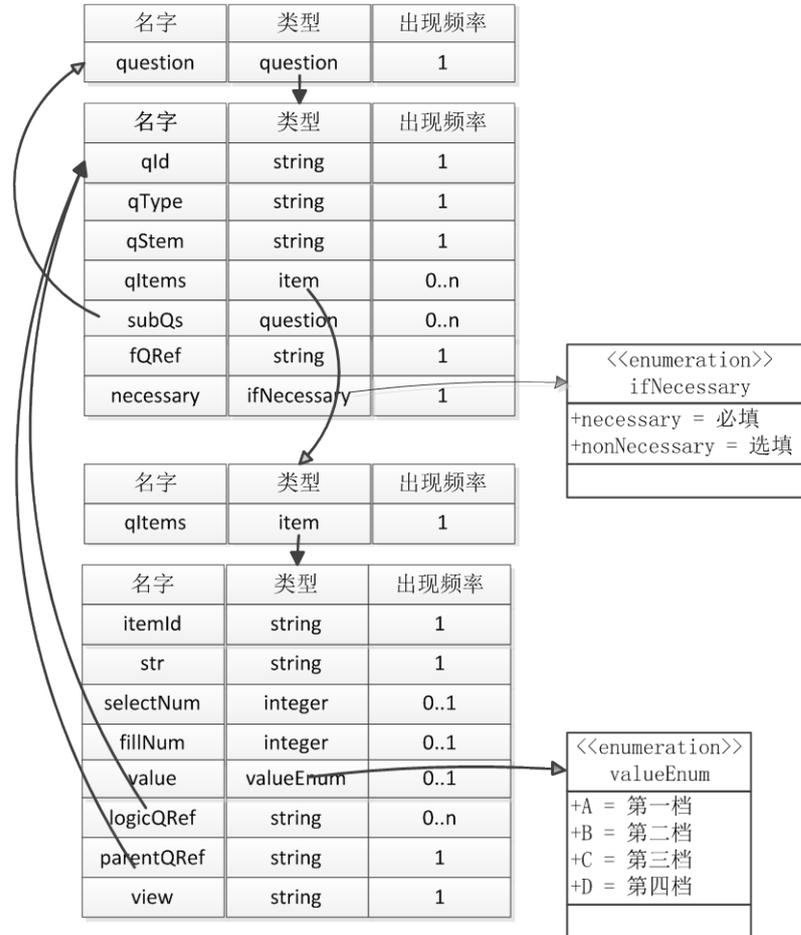


Figure 7. Meta-model of DSL
图 7. DSL 元模型

```

digit          ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
letter        ::= a | b | c | d | e | f | g | h | i | j | k | l | m |
                n | o | p | q | r | s | t | u | v | w | x | y | z |
                A | B | C | D | E | F | G | H | I | J | K | L | M |
                N | O | P | Q | R | S | T | U | V | W | X | Y | Z

unsigned_int  ::= <digit>{digit}
string       ::= {<letter> | <digit>}
    
```

Figure 8. Base elements described by BNF
图 8. 基础数据的巴科斯范式描述

```

targetFile ::= version '<question xmlns:xsi=' string
            'xsi:noNamespaceSchemaLocation='
            string '>' content '</question>'
version    ::= '<?xml version=' string 'encoding=' string '?>'
content    ::= qId qType qStem <question> | <subQ> {subQ}
question   ::= '<question>' qItems necessary fQRef '</question>'
subQ       ::= '<subQ>' qId qType qStem question '</subQ>'
qItems     ::= '<qItems>' {item} '</qItems>'
item       ::= '<item>' itemId str selectNum
            fillNum logicQRef parentQRef '</item>'
    
```

Figure 9. Compound types described by BNF
图 9. 复杂类型的巴科斯范式描述

```

qId        ::= '<qId>' string '</qId>'
qType      ::= '<qType>' string '</qType>'
qStem      ::= '<qStem>' string '</qStem>'
necessary  ::= '<ifNecessary>' string '/</ifNecessary>'
fQRef      ::= '<fQRef>' string '</fQRef>'
itemId     ::= '<itemId>' string '</itemId>'
str        ::= '<str>' string '</str>'
selectNum  ::= '<selectNum>' unsigned_int '</selectNum>'
fillNum    ::= '<fillNum>' unsigned_int '</fillNum>'
logicQRef  ::= '<logicQRef>' string '</logicQRef>'
parentQRef ::= '<parentQRef>' string '</parentQRef>'
    
```

Figure 10. Simple types described by BNF
图 10. 简单类型的巴科斯范式描述

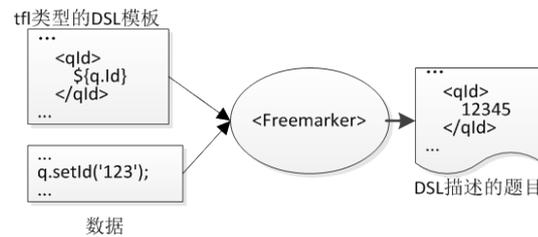


Figure 11. Demonstration of how FreeMarker works
图 11. FreeMarker 标签替换示意图

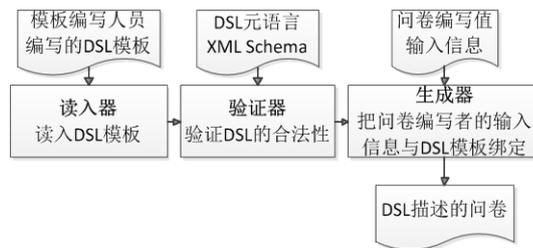


Figure 12. How FreeMarker works with DSL templates
图 12. 使用 FreeMarker 填充 DSL 模板

Table 3. Symbol declaration
表 3. 符号声明

符号	含义
'	单引号中的字代表这些字符本身
<>	其内包含的为必选项
[]	其内包含的为可选项
{}	其内包含的为可重复 0 只多次项
	其左右两边任一项
::=	是“被定义为”的意思

有异常则输出报错日志并警告相关人员，否则正常执行后续环节；在验证 DSL 模板成功后开始生存问卷环节，FreeMarker 模板引擎使用具体问卷信息对 DSL 模板中的占位符进行解析和填充，从而得到最后的问卷描述文件。

FreeMarker 输出结果是题目的 DSL 描述文件，问卷中每一道题目都对应一个 DSL 描述文件。最终这些 DSL 描述文件保存在系统磁盘上。

DSL 描述文件具有两个作用：第一，它作为一种数据结构定义了某个题目，并保存了制作题目的输入信息；第二，充当了开发者和领域专家交流的媒介。

5.2. 使用 XSLT 技术生成目标代码

调查问卷最终展示于浏览器，目标代码为 HTML。本文采用 XSLT 技术将 DSL 描述文件，即 XML 类型的文件直接转化为 HTML 目标代码。

XSLT 技术的工作原理是利用名为 stylesheet 的文件作为源文档和目标文档的转换映射。用户在 stylesheet 文件中定义源文档节点和与之对应的欲转换的预定义模板，XSLT 处理工具会遍历 XML 源文档，一旦在文档中匹配到指定节点，就会把匹配部分用预定义模板填充，从而转换为结果文档[11]。其工作过程如图 13 所示。

6. 案例与验证

6.1. 抽象 DSL 代码模板

DSL 通过 XML 标签描述对象，输出较冗长。业务员若对每道题目手动编写，复杂且易错。所以需要抽象代码模板，通过模板引擎填充用户输入的信息，自动化生成题目的 DSL 描述。模板有两个优点，第一，自动化生成 DSL，提高效率；第二，模板编写一次可以多处运行，提高复用率。

图 14 是矩阵题目的抽象 DSL 代码模板缩略图(模板中嵌套了供引擎解析的相关语法,如\${}占位符以及循环)。

6.2. 用 DSL 编写不同类型题目

本文以较为复杂的业务情况为例阐述调查问卷 DSL 的实际应用，假设有一道矩阵题包含两道小题，分别是多项选择题和填空题，其中多项选择题第一个选项与填空题有逻辑关联，该题目样例如图 15 所示。

首先使用 DSL 描述外层的矩阵题目，因为矩阵题中包含两道小题，所以最终目标文档中包含两个 subQ 节点，目标文档如图 16 所示(subQ 具体内容由后文阐述)。

矩阵题中包含的小题均为 question 类型，挂载于父 question 的 subQ 节点。此外需要设置多项选择题第一个选项与填空题的逻辑关联，具体的 DSL 描述文档树形结构示意图如图 17 所示。

利用 XSLT 技术对 DSL 文档进行转化，可将 XML 树形结构转化为 HTML 树形结构，得到目标代码。经验证，最终生成的目标代码可以在浏览器上正常展现，如图 18 所示。

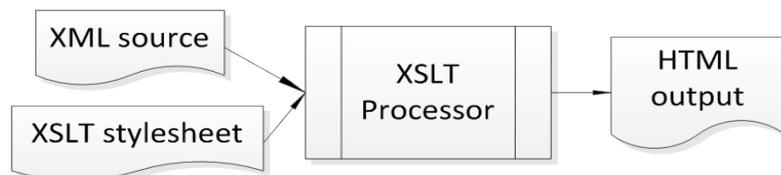


Figure 13. Demonstration of object code generation by XSLT

图 13. 利用 XSLT 生成目标代码过程示意图

```

...
<question ...>
  <!-- 题号 -->
  <qId>${q.qId}</qId>
  ...
  <!-- 遍历子问题列表 -->
  <#list q.subQs as subQ>
    <subQ>
      <qId>${subQ.qId}</qId>
      <qType>${subQ.qType}</qType>
      ...
      <!-- 子问题的具体选项 -->
      <qItems>
        <#list subQ.items as c>
          <item>
            <itemId>${c.itemId}</itemId>
            ...
            <#if 0 == c_index><value>A</value>
            <#elseif 1 == c_index><value>B</value>
            ...
            </#if>
          </item>
        </#list>
      </qItems>
    </subQ>
  </#list>
  <#if Necessary>${q.ifNecessary}</ifNecessary>
  <!-- 父类题目引用 -->
  <fQRef>${q.fQRef}</fQRef>
</question>

```

Figure 14. Matrix questionnaire problem DSL template
图 14. 矩阵题的 DSL 模板缩略图

请回答下列问题

择业因素	<input type="checkbox"/> 工资	<input type="checkbox"/> 环境
毕业时间	_____年__月	

Figure 15. An example of matrix questionnaire problem
图 15. 包含多选题和填空题的矩阵题目样例

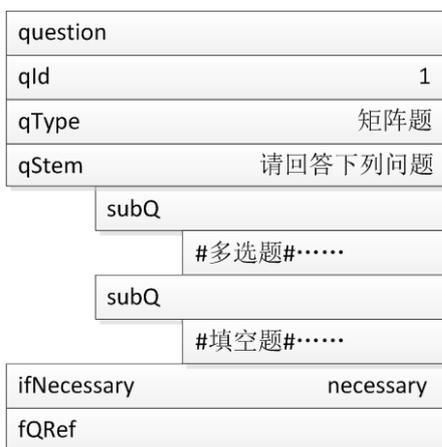


Figure 16. The outline of DSL document structure
图 16. 外层矩阵题的 DSL 文档结构示意图

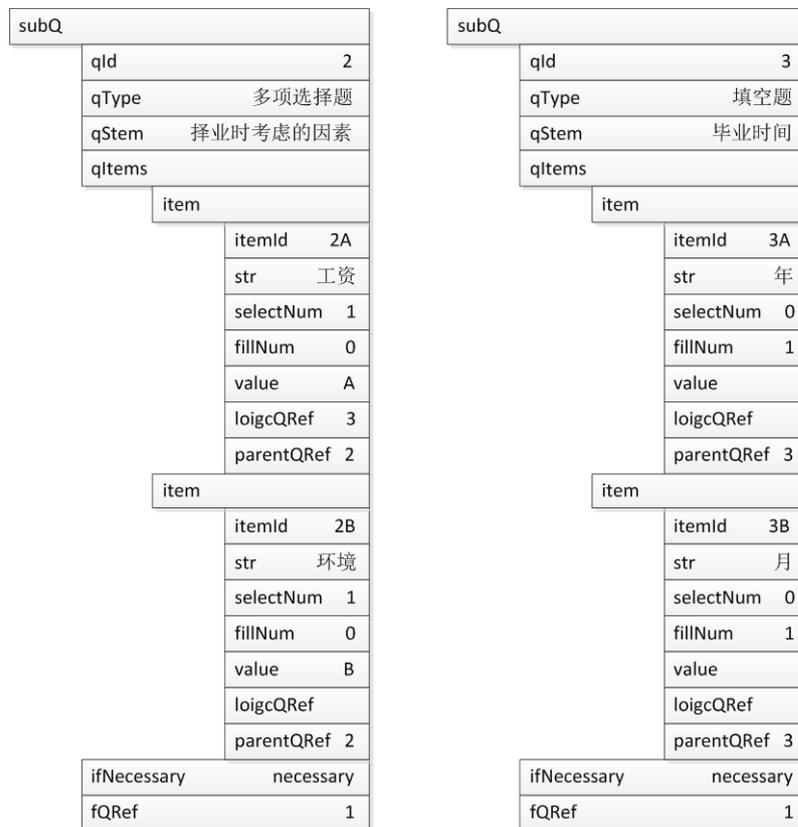


Figure 17. The outline of sub-questions document structure
 图 17. 多选题和填空题的 DSL 文档结构示意图

1. 请回答下列问题:

择业因素 工资 环境

毕业年月 2017 年 6 月

Figure 18. Screen shot of the example on the browser
 图 18. 矩阵题目在浏览器上的展现截图

7. 结论

XML 具有通用的、不被特定目的限制的语法，并且拥有完善的规范和技术支持，可作为快速创建 DSL 的方案。

本文在抽取调查问卷领域概念的基础上，探讨了使用 XML Schema 制作 DSL 的元模型、规范 DSL 文档结构，从而创建自定义 DSL 的过程；使用 FreeMarker 模板引擎和 XSLT 技术将 DSL 文件模板生成目标代码；最后通过实际案例展示了 DSL 编写模板文件及目标代码生成的效果。通过 DSL 编程和目标代码生成技术实现了提高软件开发效率和产品质量的目的。

合理的领域分析与设计方法是提取特定领域关键概念与创建完整的 DSL 的基础，我们希望在不同领域能抽象出普遍的 DSL 创建方法；此外基于 XML 的 DSL 虽可以减少语法检查和语言解析的工作，但 XML 由于是标签式语言，比较冗长，我们希望可以设计出更加简洁 DSL。以上两点将是本文以后研究的方向。

基金项目

中央高校基本科研业务费专项资金资助。

参考文献 (References)

- [1] 倪浪. 基于 B/S 的在线问卷调查系统的设计与实现[D]: [硕士学位论文]. 成都: 电子科技大学, 2014.
- [2] 陆茜茜. 面向精神卫生领域的问卷调查系统的设计与实现[D]: [硕士学位论文]. 苏州: 苏州大学, 2015.
- [3] Mernik, M., Heering, J. and Sloane, A.M. (2005) When and How to Develop Domain-Specific Languages. *ACM Computing Surveys*, **37**, 316-344. <https://doi.org/10.1145/1118890.1118892>
- [4] Fowler, M. 领域特定语言[M]. 北京: 机械工业出版社, 2013.
- [5] 张伟, 梅宏. 一种面向特征的领域模型及其建模过程[J]. 软件学报, 2003, 14(8): 1345-1356.
- [6] 王伟良, 施佺, 曹渠江. 基于 XMLSchema 抽象模型的 XML 模式验证方法[J]. 计算机应用与软件, 2007, 24(3): 41-43.
- [7] 殷丽凤, 郝忠孝. 存在 XML 强多值依赖的 XML Schema 规范化研究[J]. 计算机科学, 2010(1): 192-196.
- [8] Jiang, T. and Wang, X. (2012) Research on Metamodels Consistency Verification Based on Formalization of Domain-Specific Metamodeling Language. *Journal of Shanghai Jiao Tong University (Science)*, **17**, 171-177. <https://doi.org/10.1007/s12204-012-1248-2>
- [9] 杨兴涛, 苏桂平, 王瑞芳, 等. 特定领域建模与代码生成的研究与实现[J]. 计算机系统应用, 2009(4): 100-103.
- [10] 管太阳. 基于模板的自动代码生成技术的研究[D]: [硕士学位论文]. 成都: 电子科技大学, 2007.
- [11] 王海林. 特定领域建模与 Web 应用系统生成[J]. 计算机与信息技术, 2011(7-8): 24-29.

期刊投稿者将享受如下服务:

1. 投稿前咨询服务 (QQ、微信、邮箱皆可)
2. 为您匹配最合适的期刊
3. 24 小时以内解答您的所有疑问
4. 友好的在线投稿界面
5. 专业的同行评审
6. 知网检索
7. 全网络覆盖式推广您的研究

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: csa@hanspub.org