

Design and Verification of the Model of Discipline Achievement Collection Based on Meta Model

Mingyuan Zhu, Huobin Tan, Yiqiong Zhang, Xing Huang

Beihang University, Beijing
Email: zhumingyuan@buaa.edu.cn

Received: Nov. 5th, 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

This paper in view of the unpredictability and dynamic of the collection of discipline achievement index, puts forward a design scheme based on meta model, analyzes the features in the collection business, accesses to the field of basic building elements, defines and describes the discipline achievement data model with its abstractness and stability, so as to implement the management of the data model and discipline achievement. This scheme supports dynamic adjustment of the discipline achievement index flexibly and greatly reduces the development and maintenance costs of the system.

Keywords

Discipline Achievement, Meta Model, Data Model, Building Elements

基于元模型的学科成果采集模型设计与验证

朱明远, 谭火彬, 张祎琼, 黄 星

北京航空航天大学, 北京
Email: zhumingyuan@buaa.edu.cn

收稿日期: 2016年11月5日; 录用日期: 2016年11月22日; 发布日期: 2016年11月25日

摘要

本论文针对学科成果采集指标的未知性和动态性，提出一种基于元模型的设计方案，对采集业务中的特征进行分析，获取该领域的基本构建元素，以其抽象性和稳定性来定义和描述学科成果数据模型，从而实现了对数据模型的维护以及对学科成果数据的管理。该方案灵活的支持了学科成果采集指标的动态调整，大大降低了系统的开发和维护成本。

关键词

学科成果，元模型，数据模型，构建元素

1. 引言

进入信息化时代以来，行业领域的的数据信息成为国家和企业的关注点，业务发展过程中伴随着大量的企业数据，这些信息为企业管理和决策提供支持[1]，为行业的导向提供可靠的分析基础。在教育行业，学科建设被视为促进高校教学和科研事业发展的重要措施[2]，那么学科成果信息在学科评估工作中被教育行业专家视为学科科研实力和教学水平重要依据。与此同时，考虑到学科成果信息的复杂性和多样性，以及对学科考核指标的易变性等特点，如何高效的进行学科成果信息采集和存储成为了关键问题。

2. 问题引入

传统的信息采集系统，由于面向相对稳定的业务需求，所以在需求定义阶段就可以利用一些业务建模技术构建和用户进行交流的业务模型[3]，进而限定业务所涉及的数据范围，可以基本定义出整个系统的底层数据库结构。之后的业务变更，可能对数据库结构进行细微改动，但是不会对整体的结构产生影响。例如：物业管理系统、学校教务系统和网上购物系统等这些系统处理的数据结构比较稳定，不容易变动，可以依照传统的思路进行设计。但是对学科成果信息采集系统而言，学科信息采集的具体内容在需求分析阶段是未知的或者是高频率调整的，其中涉及到数据结构、数据类型的未知性和动态性。因此，传统的数据库设计不足以支持该情况的信息采集，需要有一种更高效机制来应对未知的和动态的数据模型。

为此，我们引入元模型理论，元模型即是描述模型的模型，用于定义模型的结构[4]，并构建特定领域的数据库模型[5]。元模型结构[6]如表 1 所示。

采用元模型建模技术，其核心思想是由元模型自动生成符合特定领域的实例模型[4]，来解决学科成果采集过程中数据对象的未知性和变化性问题。本文对采集业务中的特征进行分析，获取构建该领域模型的构建元素，建立学科采集信息的元模型，来描述未知、易变的采集信息数据结构。

Table 1. Meta model structure

表 1. 元模型结构

层次	描述
信息层	模型的一个实例，定义了详细说明信息领域的语言。
模型层	元模型的一个实例，定义了描述信息领域的语言。
元模型层	元模型的一个实例，定义了详细说明模型的语言。
元元模型层	元模型建模的支撑系统，定义了描述元模型的语言。

3. 学科成果采集系统特征分析

首先对学科成果信息采集的需求进行分析，主要从学科信息数据本身特点和信息采集业务特点两个方面入手，抽象出元模型的构建元素。

3.1. 学科数据分析

学科数据特征决定了数据的存储结构，这就意味着它是元模型组成元素中的重要部分。所以首先我们要对学科数据进一步抽象，构造出元数据模型，利用其稳定性和确定性对数据模型定义和管理。以下是数据信息的逻辑关系：

- 1) 每一条数据信息属于某一特定的场景或领域。
- 2) 每一条数据信息以一条记录的形式独立存在，其中记录之间可能存在某一种关系。
- 3) 每一条数据包含至少一个属性，属性的之间也可能存在某些联系；
- 4) 属性的类型包括：数字、日期、字符串、富文本等类型。

3.2. 采集业务分析

在学科数据信息采集的过程中，存在基于学科数据特征下的用户交互方式、采集数据限制，数据的后续处理等，这也是元模型的必要组成部分。以下是采集数据的业务背景：

不同业务含义的记录，需要提供特定的用户交互界面进行数据录入，例如：“学科师资人数”采集内容，我们提供表单页面，支持数据的录入，因为每一个学科有且只包含一条记录；但是“学科科研论文”采集内容，我们需要提供表格，支持用户进行数据录入，因为一个学科可能包含多篇科研论文。

属性的数据类型不同，需要提供特定的输入框，方便用户录入。例如：日期类型的数据需要提供日期控件；富文本类型属性需要提供在线富文本编辑框；

属性具有特殊的业务含义，需要对数据源进行限制，同时提供相应的输入框，提示用户操作。例如：“获奖级别”属性，限制其输入的内容为“一等奖”、“二等奖”、“三等奖”，并提供下拉框的输入方式，提供用户选择；

在用户输入的过程中，需要对数据格式进行检查，在采集完成之后，需要对某一些数据进行相关的数据加工处理，例如：检查是否符合整数、分数、日期格式；检查是否满足属性之间的逻辑关系；对某一些属性的数据进行规范化等；表 2、表 3 给出了部分在全国第四轮学科评估中的逻辑检查规则。

4. 元模型建立

根据对学科成果信息采集的业务的分析，我们将元模型主要分为 8 个元素信息，分别为：领域信息元素、分类元素、实体风格元素、属性风格元素、校验规则元素和数据字典元素，通过元模型的 8 个元素来描述和构建信息采集系统中的数据模型。元模型组成图如图 1 所示。

Table 2. String type checking rule

表 2. 字符串类型检查规则

规则 ID	规则类型	规则名称	说明
S1	String	字符串非空	字符串非空
S2	String	字符串长度	字符串长度限制
S3	String	邮箱	字符串满足邮箱的格式
S4	String	ISBN 编号	字符串满足 ISBN 编号的格式

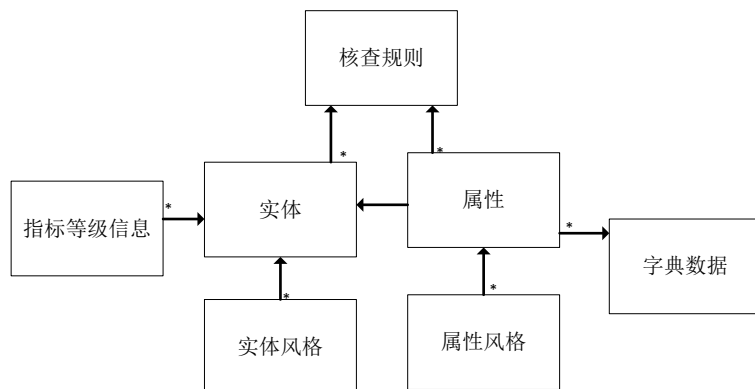


Figure 1. Meta model composition diagram
图 1. 元模型组成图

Table 3. Logical relation checking rule

表 3. 逻辑关系检查规则表

规则 ID	规则类型	规则名称	说明
L1	Logic	参评单位数	参与单位数数据格式为: a(b)。 a 表示总共参与项目的单位数; b 表示本单位在其中的排名。 规则如下: $a \geq 1, b \geq 1$ 且 $b \leq a$;
L2	Logic	该字段数值小于指定字段的数值	该字段的数值小于另一字段的数值, 例如: 学生的来华时间一定要小于该学生毕业时间;
L3	Logic	该字段数值等于其他各个字段之和	该字段等于其他字段数值之和, 例如: 专家总数字段 应等于 30~39 岁专家数相加 40~49 岁专家数相加 50 岁以上专家数;

1) 实体元素

实体元素是对记录集合的描述, 定义此集合中的所有记录的存储空间、物理表名称、主键、外键等信息; 以采集师生资源信息为例, 所有师生资源的记录构成一个集合, 实体元素信息定义此集合的存储空间、物理表名、设定教师编号作为主键, 教师所在的单位作为外键。

2) 属性元素

每一条记录中至少包含一个属性, 属性元素即为对数据记录中属性的描述。定义其存储的物理字段名称, 字段长度、字段类型、字段的取值范围, 同时还维护一个属性之间的关系。以采集科研项目信息为例, 一条科研项目记录中包含: 项目编号、项目名称、项目类型、项目负责人、开始时间、结束时间等。属性元素建立采集数据属性和物理字段之间的映射关系, 定义各个数据的类型和长度。此外, 对于项目类别字段, 属性元素和字典元素结合使用, 定义字段的取值范围; 对于项目的开始、结束时间两个字段, 属性元素维护一个它们之间的逻辑关系。

3) 指标等级元素

学科成果采集指标项分为多个级别, 其中末级指标对应实际的采集内容, 也就是上文所说对应实体元素, 对于非末级指标由指标等级元素进行描述, 维护采集指标的级别关系。

4) 实体样式元素和属性样式元素

数据的采集存在多种方式, 包括: 网络爬虫、数据库导入、用户录入等, 为了支持与用户的交互, 元模型包含实体样式元素和属性样式元素。其作用是描述采集交互界面的样式以及每一个输入字段的显示形式。其中采集界面包含表单、表格和富文本编辑器; 字段的显示样式包含: 单选框、多选框、下拉框等。用户在选择采集项后, 系统会根据实体样式和属性样式元素自动生成相关的交互页面。

5) 数据核查规则元素

在数据人工录入过程中和数据采集完成之后,为了确保采集数据的质量和正确性,会在后续的过程中对数据的格式、合法性、记录和记录之间、属性和属性之间的逻辑关系等进行相关的核查工作。数据核查规则元素从记录和属性两个维度对数据核查方法进行定义,例如:对于一个学生信息采集项,数据核查规则元素定义了该采集指标项限填记录数和出生日期、电话号码、邮箱等字段的规范格式,以及学生入学时间和毕业时间两个字段之间的逻辑核查规则。

6) 字典数据元素

数据字典元数据可以细分为:字典数据和字典条目数据。字典数据包括字典的名称、类型、描述信息等;字典条目数据描述的是一个字典中所包含的条目信息。例如专家职称是一个字典的名称,这个字典中包含了“中国学科院院士”、“中国工程院院士”、“千人计划”、“长江学者”等相关条目信息。

5. 元模型应用

元模型应用主要包含以下 3 个方面:元数据访问、学科成果存储和采集交互页面生成。

• 元数据访问

元数据访问部分主要是对元数据的操作,包括增加、删除、修改、查询。主要以元数据的查询为主。在查询过程中,对元数据进行组合和封装形成一个完整的采集项的描述信息,涉及到采集成果数据存储结构的描述信息(数据表名称、主键、外键、字段名称、类型等)、采集项页面样式的描述信息(采集项显示类型、控件类型、下拉框字典数据等)、采集项数据检查规则等。元数据核心类图如图 2 所示。其中 `MetaEntityDomain` 是对实体信息、实体样式信息、实体检查规则信息的封装; `MetaAttributeDomain` 是对属性信息、属性样式信息、属性的检查规则和属性字典数据集合的封装; `MetaEntityService` 是对外统一提供的元数据访问接口, `MetaEntityDao` 是对 service 层提供访问元数据的接口, `MetaEntityServiceImpl` 和 `MetaEntityDaoImpl` 是这两个接口的具体实现。

• 学科成果存储

学科成果存储是通过元数据进行数据结构的表示,获取学科成果真实的存储物理表结构,然后完成学科成果的存储工作。图 3 给出了学科成果存储类图。图中 `CollectService` 接口是对外提供学科成果数据操作的统一接口,包括新增、删除、编辑、查询等功能。`CollectServiceImpl` 是该接口的实现类,主要处理学科成果操作过程中的一些业务逻辑,例如:查询条件参数封装,学校、学科操作范围判定,评估状态检查,日志记录等;学科成果真正的数据库操作在父类 `MetaOperService` 中完成。数据操作过程如下:通过元数据对外接口 `MetaEntityService`,获取元数据,通过 `SqlBuilder` 构造 SQL 语句,然后通过 `Executor` 执行 SQL 语句完成对数据库的访问。在 SQL 构造过程中需要对字段类型进行转化,例如字典类型的在元数据的表示为 DIC,在 SQL 构造需要转为 VARCHAR2 的数据类型。此处使用简单工厂模式完成对应字段类型的转化。

• 采集页面生成

采集页面主要包含采集指标树和采集录入表两部分组成,均基于元数据动态生成,生成过程如下所示:

- ✓ 元数据并封装成 JSON 数据格式;
- ✓ 选择特定的采集页面模板;
- ✓ 元数据解析生成页面。

页面生成类图如图 4 所示。学科成果采集页面包含两个主要部分:采集指标树和采集项录入表。共有采集树模板、采集表单模板、采集表格模板和采集在线编辑模板等多个模板。各个模板各自拥有初

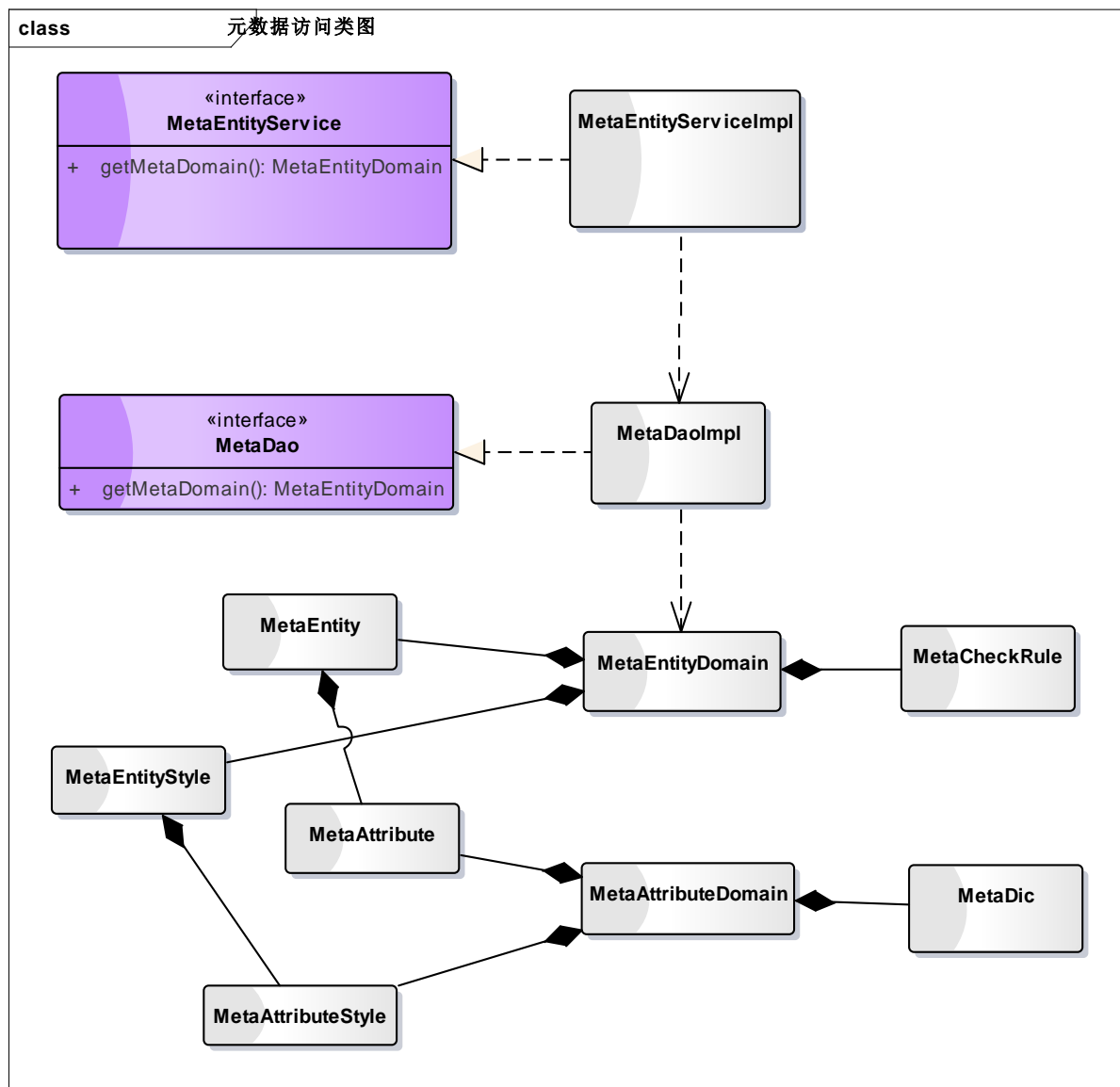


Figure 2. Core structure class diagram of meta model

图 2. 元数据核心类图

始化函数，提供采集页面调用，最终生成采集用户交互页面。采集页面负责请求采集指标树配置信息和采集录入表配置信息，并根据元数据调用对应的模板完成页面生成。**ViewConfig** 是对元数据的封装，利用工厂模式根据不同使用场景，封装成具体的子类，**TreeConfig** 对应采集指标树的配置信息；**FormConfig** 对应采集表单的配置信息；**TableConfig** 对应采集表格的配置新；**OnlineEditConfig** 对应在线编辑配置信息。

6. 实施与验证

本文提出的学科成果采集元模型，分别在教育部学科自检平台和学科评估平台的成果采集系统中得到了应用。学科自检平台的评估目标是希望高校通过系统进行自我检测，其任务是学科自身的考察提供服务，采集的成果信息涉及到更多的细节，同时考虑到高校之间的重点专业以及研究领域的差异性，

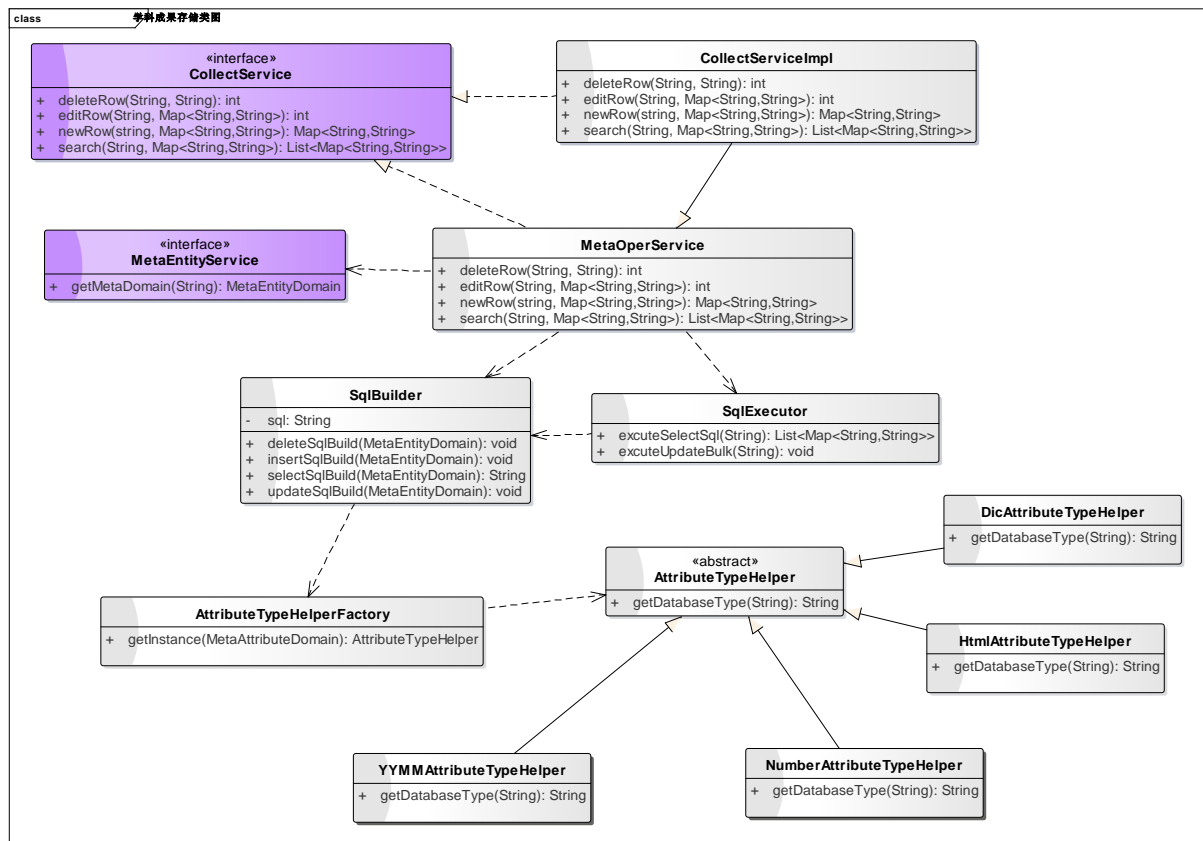


Figure 3. Academic achievement storage class diagram

图 3. 学科成果存储类图

采集成果中也会包含很多学校的特色科研信息；学科评估平台是学位中心提出的面向全国高校的评估任务，指标的制定涉及教学、科研、学生质量、社会声誉等多个方面，采集的成果内容更具有普遍性和可比性，为最终能够计算出学科在全国范围内的学科排名提供客观和公正的数据依据。这两个平台分别是面向不同性质的评估任务，所以学科成果采集指标的差异性很大。其次面临的问题就是为了推进学科的科研和教学适应时代的发展，在同一学科评估任务的不同轮次中，采集指标会进行较大调整，例如：第四轮学科评估中，新添加了海外留学经历和国际交换生的采集内容。此外，采集信息的内容涉及范围很广，包括：管理类、计算机类、建筑类、理工农医类、人文社科类、体育学科类和艺术类7大学科门类，导致录入的数据类型繁多，可能包括数字类型、时间类型、名次/百分比等，针对不同的数据类型还需要有相应的约束规则进行数据检验。针对以上问题，建立数据采集元模型来描述学科成果采集的数据模型，在每一次评估任务开始之前在数据库中做好元模型数据的配置工作，并建立采集数据模型物理表结构。之后，就可以通过元模型的接口来完成所有的采集工作，为后续的学科评估提供数据支撑。目前系统可以高效地应对各学校的海量学科成果采集工作，并灵活支持指标体系在每一轮次评估中的动态调整。

7. 结论

基于元模型理论对未知的、动态的采集数据模型进行抽象，提取出基本的构建元素，建立元模型框架，以相对稳定的元数据结构去描述和管理学科成果数据模型。虽然在数据采集初期阶段，元模型数据的配置工作较为繁重，但是其以这种配置的方式减少了对代码修改，大大降低了对系统的开发和维护成本，使得采集系统更具有灵活性和扩展性。

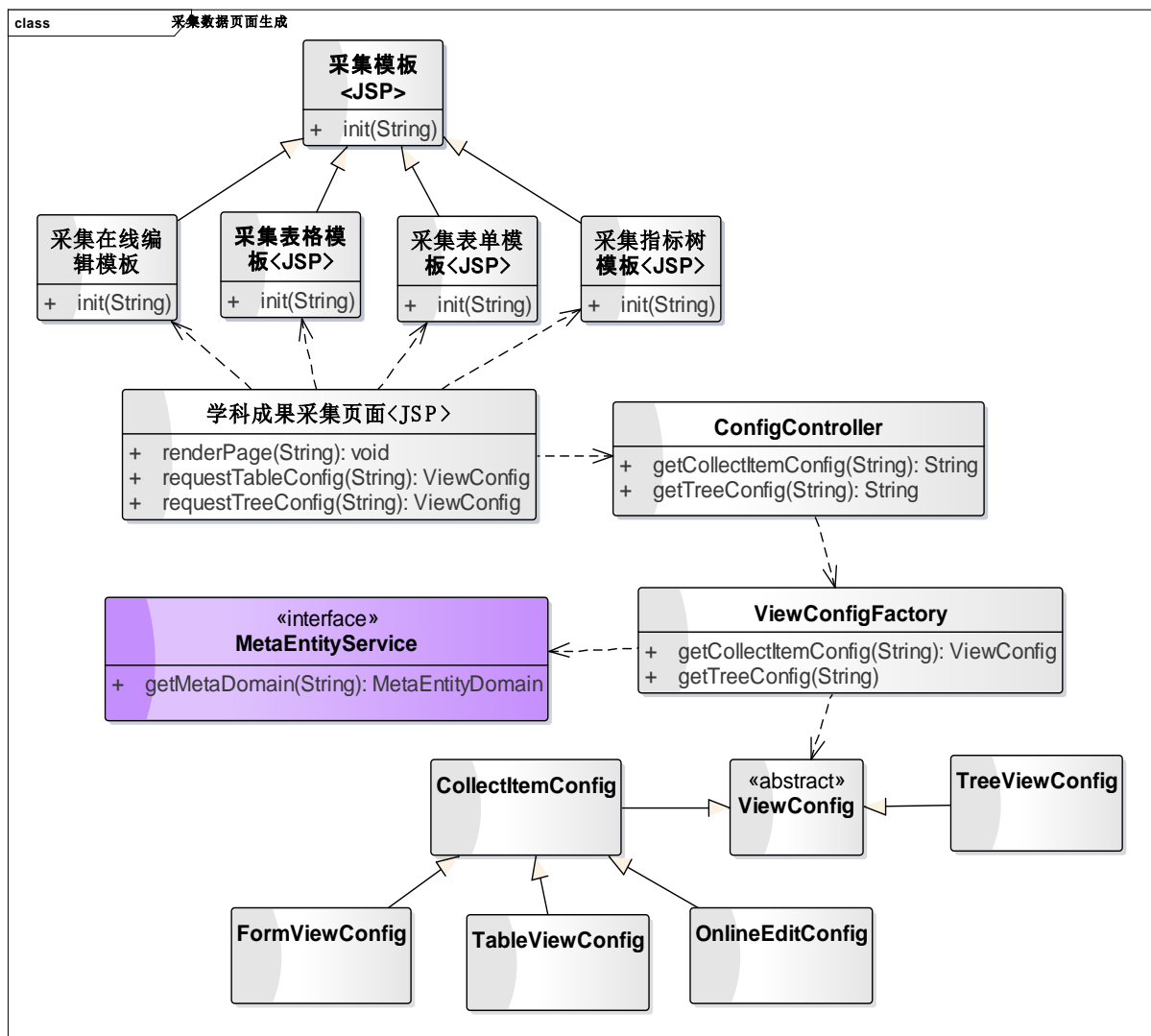


Figure 4. Page generation class diagram

图 4. 页面生成类图

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] 苗虹. 本体支持下的企业数据模型构建[J]. 清华大学学报(自然科学版), 2006, 46(S1): 40-48.
- [2] 王向前. 高校重点学科评估模型的研究与实现[J]. 计算机工程与应用, 2008, 44(8): 232-235.
- [3] 赵占梁. 需求分析过程中规范性业务交流语言研究[J]. 科学技术与工程, 2007, 7(11): 37-41.
- [4] 宛海宁. 元模型理论及其在企业体系结构开发设计中的应用[J]. 系统工程理论与实践, 2012, 32(4): 847-853.
- [5] 刘威. 基于元模型的统一制造资源模型框架[J]. 计算机集成制造系统, 2007, 13(10): 1903.
- [6] 毛媛. 基于元模型的复杂系统建模方法研究[J]. 系统仿真学报, 2002, 14(4): 411.

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