Logical Approach in Software and Data Design
软件和数据设计中的逻辑方法

Evgeny Cherkashin

Matrosov Institute for System Dynamics and Control Theory of Siberian Branch of Russian Academy of Sciences, Irkutsk, Russia
俄罗斯伊尔库茨克，俄罗斯科学院西伯利亚分院马特罗索夫系统动力学与控
eugeneai@icc.ru

2023, December,
Yantai, China

Model–Driven Architecture: Research objectives

Main objective of the research is to construct a MDA technology based on nowadays system modeling visual languages (SysML, UML, BPMN, CMMN) and existing Semantic Web vocabularies and technologies. The following techniques and software are under development:

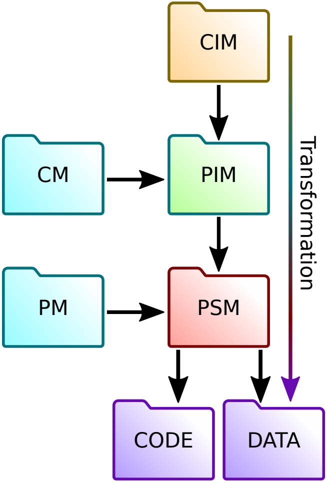
1. CIM representation with SysML, BPMN, CMMN, and results of source code processing,
2. CIM, PIM, PSM representation in UML, RDF with existing vocabularies,
3. transformation implementation with logical language Logtalk,
4. usage of LOD sources in transformations for obtaining additional semantic data,
5. generation of documents and user interfaces with LOD markup.

模型驱动架构： 研究目标

研究的主要目标是基于当今的系统建模可视化语言（SysML、UML、BPMN、MDA 技术。以下技术和软件正在开发中

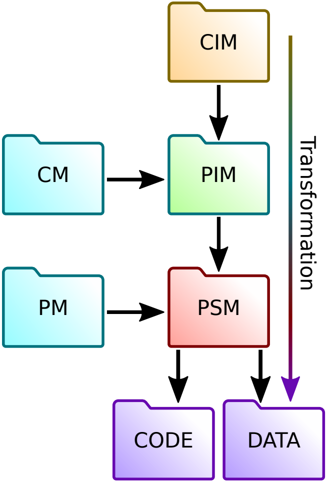
1. 用 SysML、BPMN、CMMN 和源代码处理结果表示 CIM、
2. 用 UML、RDF 和现有词汇表表示 CIM、PIM 和 PSM、
3. 使用逻辑语言 Logtalk 实现转换、
4. 在转换中使用 LOD 源以获取额外的语义数据、
5. 使用 LOD 标记生成文档和用户界面。

Model–Driven Architecture



MDA	Model–Driven Architecture;
CIM	Computationally Independent Model;
CM	Model of Computations;
PIM	Platform Independent Model;
PM	Platform Model;
PSM	Platform–Specific Model;
CODE	Source code of software;
DATA	Initial database state.

模型驱动架构（Model–Driven Architecture）



MDA	模型驱动架构（Model–Driven Architecture）；
CIM	Computationally Independent Model（独立计算模型）；
CM	计算模型;
PIM	平台独立模型;
PM	平台模型;
PSM	特定平台模型;
CODE	软件源代码;
DATA	初始数据库状态。

Logtalk as transformation definition language

We have chosen Logtalk as it

- inherits widely known Prolog language syntax and runtime;
- is implemented as macro package, performance penalties are about 1.5%;
- has flexible semantics: we can define transformations and constraints within the same syntax;
- implement object–oriented knowledge (rules) structuring, encapsulation and replacement;
- compositional way of transformation implementation;
- powerful engine to post constraints on object-to-object messages (events);
- has implementation for various Prolog engines.

The «regular» language allow us to use its libraries not directly related to MDA transformations.

Logtalk 作为转换定义语言

我们选择Logtalk是因为它

- 继承了广为人知的 Prolog 语言的语法和运行时;
- 作为宏包实现，性能损失约为1.5
- 具有灵活的语义：我们可以在相同的语法中定义转换和约束;
- 实现了面向对象的知识（规则）结构化、封装和替换;
- 实现转换的组合方式;
- Logtalk是一个强大的引擎，可以对对象到对象的消息（事件）发布约束;
- 为各种Prolog引擎提供实现。

这种常规语言允许我们使用它与 MDA 转换没有直接关系的库。

Linked Open Data, LOD

1. Information is published in Internet with open access license;
2. It is represented in a machine-readable form, e.g., Excel table instead of a bitmap picture;
3. An open format used, e.g., CSV instead of Excel;
4. The format is based on W3C recommended standards, allowing RDF and SPARQL reference;
5. Published data refer to objects, forming context.

Thus, applications publish data as relations of objects (entities).

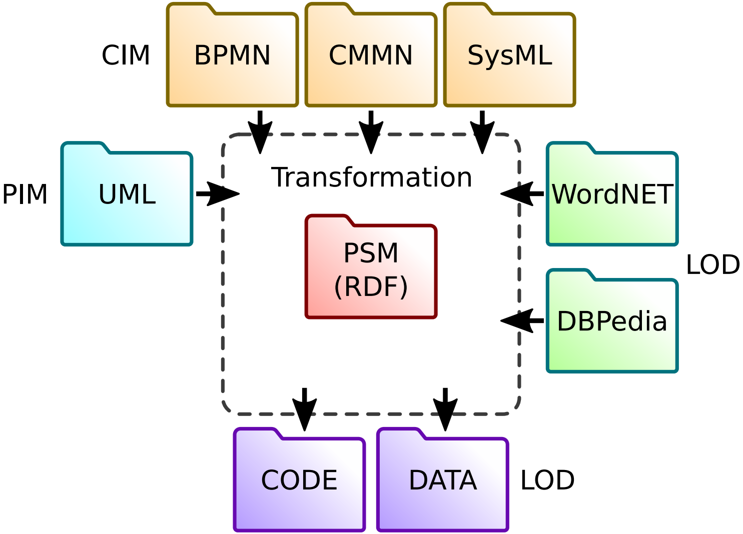
Linked Open Data, 关联开放数据, LOD

1. 信息在互联网上发布，采用开放式获取许可；
2. 它以机器可读的形式表示，例如 Excel 表格，而不是位图图片；
3. 使用的开放格式，如 CSV 而非 Excel；
4. 该格式基于 W3C 推荐的标准，允许引用 RDF 和 SPARQL；
5. 发布的数据指代对象，形成上下文。

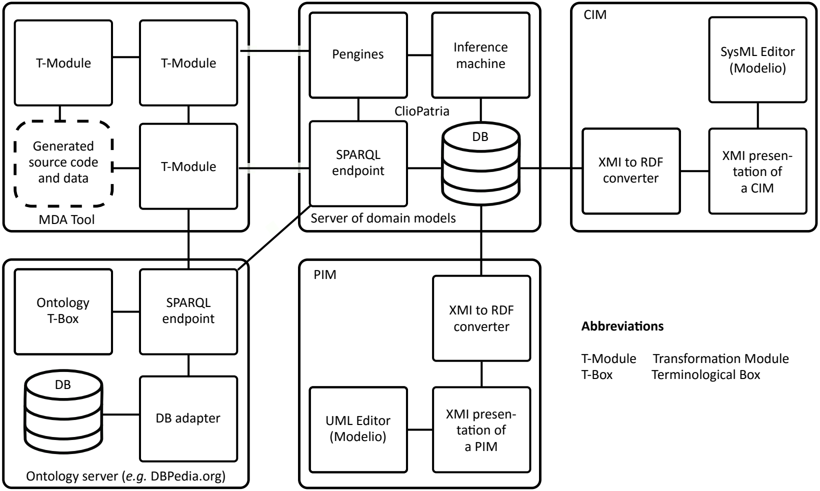
因此，应用程序以对象（实体）关系的形式发布数据。

Model Driven Architecture and Linked Open Data

模型驱动架构和关联开放数据



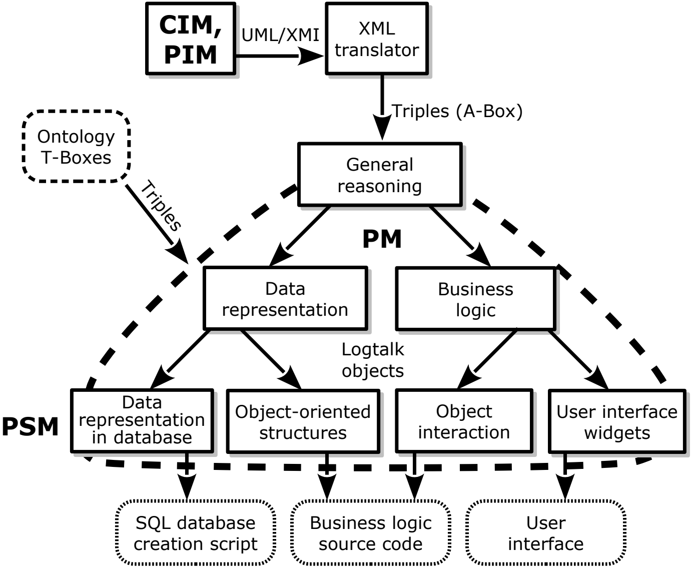
MDA infrastructure, 基础设施



Abbreviations

T-Module	Transformation Module
T-Box	Terminological Box

Architecture of transformation modules, 转换模块的结构



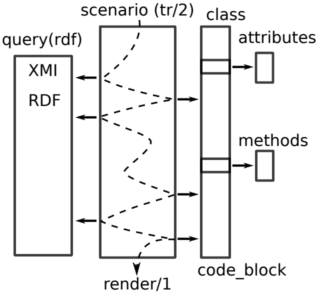
r

PSM: Scenario of a Class synthesis, PSM: 课堂综合情景

```
:- object(direct(_Package,_LocalProf,_CodeProf)).   % 转换驱动程序对象
:- public([tr/4,tr/3]).                   % 类综合方案的公共接口
% . . . . . . . . . .
tr(class, Class, ClassID):- ::package(Package),    % 合成一个类
   query(Package)::class(Name, ClassID),        % XMI 中的查询包结构
   create_object(Class,    % . . . . .        % 创建类对象
   create_object(Attributes, % . . . . .        % 创建属性对象
   create_object(Methods,  % . . . . .        % ...方法
   Class::name(Name),                 % 为班级命名。
   % Generate attributes of the class,
   % organizing them in a local database.
   % ...methods...
   Class::attributes(Attributes),            % 为类设置属性。
   Class::methods(Methods).                 % ...方法。

tr(attribute, Attribute, ClassID, AttributeID):-    % 属性转换
   ::package(Package),
   query(Package)::attribute(Name,ClassID,AttrID),
   create_object(Attribute,  % . . . . .
   Attribute::name(Name).               % 为属性命名。

tr(method, Method, ClassID, MethodID):-         % 方法的转变
   ::package(Package),
   query(Package)::method(Name,ClassID,MethodID),
   create_object(Method,   % . . . . .
   Method::name(Name).                % 方法名称
:- end_object.
```

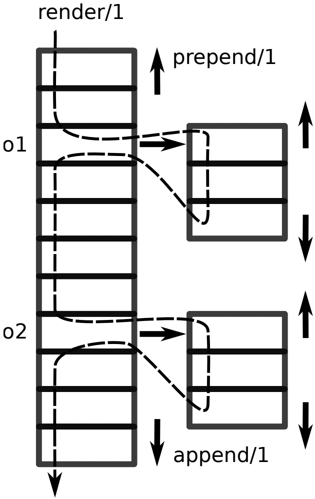


Implementation of Query object, 实现查询对象

```
:- object(query(_XMI)).
:- protected(xmi/1).
:- public([class/2, attribute/3, method/3]).
xmi(XMI) :- parameter(1, XMI).
class(Name, ID):-                    % 识别 RDF 中的类别
   ::xmi(XMI),
   XMI::rdf(ID,rdf:type,uml:'Class'),
   XMI::rdf(ID,rdfs:label, literal(Name)).
attribute(Name, ClassID, ID):-          % ...属性...
   ::xmi(XMI),
   XMI::rdf(ClassID, xmi:ownedAttribute, ID),
   XMI::rdf(ID, rdfs:label, literal(Name)).
method(Name, ClassID, ID):-             % ...方法...
   ::xmi(XMI),
   XMI::rdf(ClassID, xmi:ownedOperation, ID),
   XMI::rdf(ID, rdfs:label, literal(Name)).
% . . . . . . . . . . .
:- end_object.
```

Code Block (idea is taken from llvmlite*, 代码块，创意来自 ” llvmlite ” 图书馆)

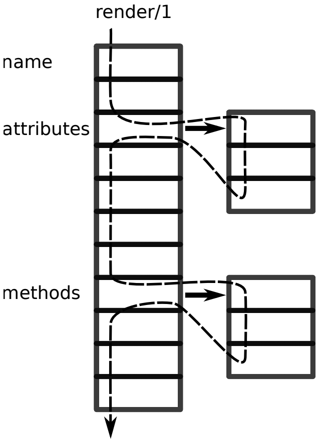
```
:- object(code_block, specializes(root)).
% Public interface of the object
:- public([append/1, prepend/1, clear/0,
  render/1, render_to/1, remove/1,
  item/1, items/1]).
% Code block items
:- dynamic([item_/1]).
:- private([item_/1]).
% Methods specialized during inheritance
:- protected([renderitem/2, render_to/2]).
% . . . . . . . . . . . .
% Delegate rendering to object itself
renderitem(Object, String):-
   current_object(Object), !,
   Object::render(String).
% Convert a literal to its string
% representation
renderitem(literal(Item), String):-!,
   atom_string(Item, String).
% Just print the item (debugging).
renderitem(Item, String):-
   root::iswritef(String, '%q', [Item]).
:- end_object
```



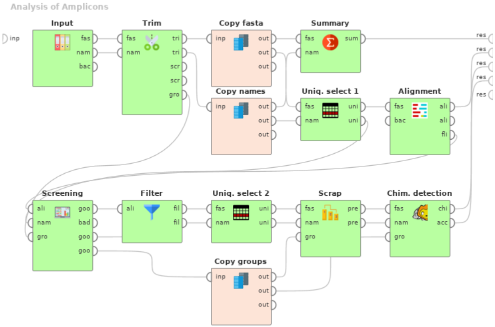
*) https://github.com/numba/llvmlite

PSM of a Python Class as a specialization of Code Block, 作为代码块特化的 Python 类

```
:- object(class, specializes(code_block),
   imports([named])). % Category of named entities
:- public([classlist/1, methods/1, attributes/1]).
% . . . . . . . . . . . . . .
renderitem(Item, Result):-     % proceed with default
   ^^renderitem(Item, Result). % rendering
render(Result):-       % Source generator
   ^^render(Name),     % implemented in a category
   ( ::item(classlist(List)) ->
    % . . . . . . . . . . .
      [Name]) ),
   ( ::item(attributes(Attributes))->
    % . . . . . . . . . . .
      [DefAttrList]),
     Attributes::items(InstanceAttrs),
     findall(S, ( % initialize attributes
       % . . . . . . . . .
       ), AttrAssigns),
      root::unindent,
      AttrList=[ConstructorDef|AttrAssigns];
       % . . . . . . . . .
      AttrList=[ConstructorDef, Pass] ),
   ( ::item(methods(Methods))-> % If any ...
     Methods::render(MethodList);
     MethodList=[] ),
   lists::append(AttrList,MethodList,StringList),
   root::unindent, Result=[Signature|StringList].
```



Applications: Dataflow representation of NGS analysis of amplicons, 应用： 扩增子 NGS 分析的数据流表示法



Term 术语	Description 说明
NGS	新一代测序
Amplicon	复制多次的 DNA 或 RNA 部分
Mothur	用于 NGS 研究的软件工具集
Rapidminer	视觉分析工具

Green blocks are Mothur modules. Others are Rapidminer modules. 绿色块为 Mothur 模块。其他为 Rapidminer 模块。

Rapidminer module synthesis, 模块合成

```
vector<string> AlignCommand::setParameters(){ // 模块源的一部分
try {
 CommandParameter ptemplate(” reference” ,” InputTypes” ,” ” ,” ” ,” none” ,” none” ,” none” ,” ” ,false,true,true); parameters.pu
 CommandParameter pcandidate(” fasta” ,” InputTypes” ,” ” ,” ” ,” none” ,” none” ,” none” ,” fasta-alignreport-accnos” ,false,true
 CommandParameter psearch(” search” ,” Multiple” ,” kmer-blast-suffix” ,” kmer” ,” ” ,” ” ,” ” ,” ” ,false,false,true); parameters.pu
 CommandParameter pksize(” ksize” ,” Number” ,” ” ,” 8” ,” ” ,” ” ,” ” ,” ” ,false,false); parameters.push_back(pksize);
 CommandParameter pmatch(” match” ,” Number” ,” ” ,” 1.0” ,” ” ,” ” ,” ” ,” ” ,false,false); parameters.push_back(pmatch);
// . . . . . . .
package com.rapidminer.ngs.operator; // 生成的 Java 模块
// imports

class MothurChimeraCcodeOperator extends MothurGeneratedOperator {
 private InputPort fastaInPort = getInputPorts().createPort(” fasta” );
 private InputPort referenceInPort = getInputPorts().createPort(” reference” );
 private OutputPort chimeraOutPort = getOutputPorts().createPort(” chimera” );
 private OutputPort mapinfoOutPort = getOutputPorts().createPort(” mapinfo” );
 private OutputPort accnosOutPort = getOutputPorts().createPort(” accnos” );

 public MothurChimeraCcodeOperator (OperatorDescription description) {
  super(description);
 }
 @Override
 public void doWork() throws OperatorException {
  super();
  // . . . . . .
 }
 @Override
 public List<ParameterType> getParameterTypes() {
  super();
    // . . . . . .
 }
 @Override
 public String getOutputPattern(String type) {
  if (type==” chimera” ) return ” [filename],[tag],ccode.chimeras-[filename],ccode.chimeras” ;
  if (type==” mapinfo” ) return ” [filename],mapinfo” ;
  if (type==” accnos” ) return ” [filename],[tag],ccode.accnos-[filename],ccode.accnos” ;
  return super.getOutputPattern(type);
 }
}
```

RDF (TTL) representation and ad its query object, RDF 表示法并添加其查询对象

```
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
ngsp:spec a ngsp:Specification ;
  ngsp:module mothur:NoCommand,
    mothur:align-check,
    mothur:align-seqs,
# . . . . .
mothur:align-check a ngsp:Module ;
  ngsp:outputPattern [ a cnt:Chars ;
      ngsp:parameterName ” type” ;
      ngsp:pattern [ ngsp:patternString
          ” [filename],align.check” ;
          dc:identifier ” aligncheck” ] ;
      cnt:chars # . . . .
# . . . . .
mothur:align-check-idir-parameter a ngsp:Parameter ;
  ngsp:important false ;
  ngsp:multipleSelectionAllowed false ;
  ngsp:optionsDefault ” ” ;
  ngsp:required false ;
  ngsp:type mothur:String ;
  dc:title ” inputdir” .

mothur:align-check-map-parameter a ngsp:Parameter ;
  ngsp:important true ;
  ngsp:multipleSelectionAllowed false ;
  ngsp:optionsDefault ” ” ;
  ngsp:required true ;
  ngsp:type mothur:InputTypes ;
  dc:title ” map” .

mothur:align-check-name-parameter a ngsp:Parameter ;
  ngsp:chooseOnlyOneGroup ” namecount” ;
  ngsp:important false ;
  ngsp:multipleSelectionAllowed false ;
```

```
    extends(ngsquerybase)).

:- public(type/1). % 立面对象
type(Type) :-
  ::attr(type, Type).
:- public(name/1).
name(Name) :- ::attr(dc:title, literal(Name)).
:- public(options/1).
options(Value):- ::attr(options, Value).
:- public(options_default/1).
options_default(Value):-
  ::attr(optionsDefault, Value).
% . . . . . . . .
:- public(multiple_selection_allowed/0).
multiple_selection_allowed:-
  ::bool_attr(multipleSelectionAllowed).
:- public(required/0).
required:-
  ::bool_attr(required).
:- public(important/0).
important:-
  ::bool_attr(important).
:- protected(attr/2).
attr(NS:Name, Value):-
  ::ngs(RDF),
  ::second(Parameter),
  rdf_db::rdf_global_object(Value, V),
  RDF::rdf(Parameter, NS:Name, V).
attr(Name, Value):-
  \+ Name=_:_,!,
  ::ngs(RDF),
  ::second(Parameter),
  rdf_db::rdf_global_id(Value, V),
  RDF::rdf(Parameter, ngsp:Name, V).
% . . . . .
```

Discussion

Interesting positive impressions obtained:

- Logtalk and RDF are flexible, sufficiently universal and convenient implementation infrastructures for MDA;
- The best implemenation means is Prolog predicate wrapping and Logtalk object encapsulation of rules;
- Not all Logtalk properties are investigated: there might be more sophisticated programming techniques developed, e.g., on the base of message watchers.

Technical problems making the approach somewhat problematic:

- Very simple tasks take too much efforts, e.g., text processing: convert an identifier into the CamelCase;
- It takes too long to surf Internet in order to find a vocabulary for a domain, but it is more productive than development;
- Prolog is not a popular language in MDA, neither Logtalk.

Discussion, 讨论

有趣的正面印象:

- Logtalk 和 RDF 是灵活、足够通用和方便的 MDA 实现基础架构;
- 最好的实现手段是对规则进行Prolog谓词封装和Logtalk对象封装;
- 并非所有的Logtalk属性都得到了研究：在消息监视器的基础上，可能会开发出更复杂的编程技术。

技术问题使得这种方法有些困难:

- 非常简单的任务就需要花费太多精力，比如文本处理：将标识符转换成 CamelCase;
- 为了找到一个领域的词汇而上网太花时间了，但这比开发更有成效;
- Prolog在MDA中并不流行，Logtalk也不是。

Document authoring and storage

In most cases documents are created as a result of

- creative activity of a person with a text processors (authoring);
- printing a digital copy or a data record in a database;
- aggregation operation over database records (report).

Then it is stored either as a physical paper and/or a digital document (PDF, DOCX, HTML).
Since 2000-th, Semantic Web and Linked Open Data (LOD) is being developed, allowing

- structural storage of data within published documents;
- processing stored data computationally;
- integration of data structures and data objects globally.

The aim of this research is to develop technologies, software and services allowing construction of digital archives supporting document data inclusion and inference from existing documents.

Document authoring and storage, 文件编写和存储

在大多数情况下，创建文档的结果是

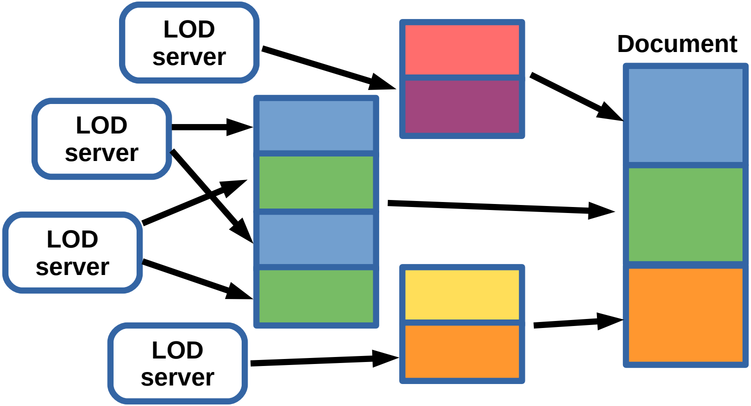
- 文本处理人员的创造性活动（创作）；
- 打印数字副本或数据库中的数据记录;
- 对数据库记录进行汇总操作（报告）。

然后
以实体纸张和/或数字文档（PDF、DOCX、HTML）的形式存储。
自2000年以来，语义网（Semantic Web）和关联开放数据（Linked Open Data，LOD）得到了发展，从而可以

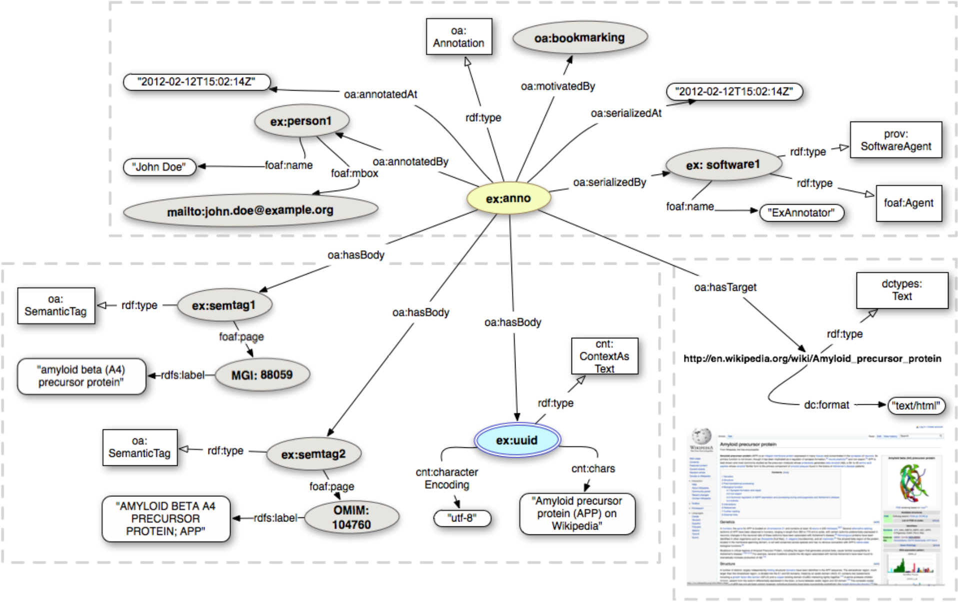
- 在已发布的文档中以结构化方式存储数据;
- 计算处理存储的数据;
- 在全球范围内整合数据结构和数据对象。

The aim of this research is to develop technologies, software and services allowing construction of digital archives supporting document data inclusion and inference from existing documents.
这项研究的目的是开发技术、软件和服务，以便构建数字档案，
支持文件数据的包含和现有文件的推理。

Structure of a document, 文件结构



Open Annotation ontology (oa), 开放注释本体论

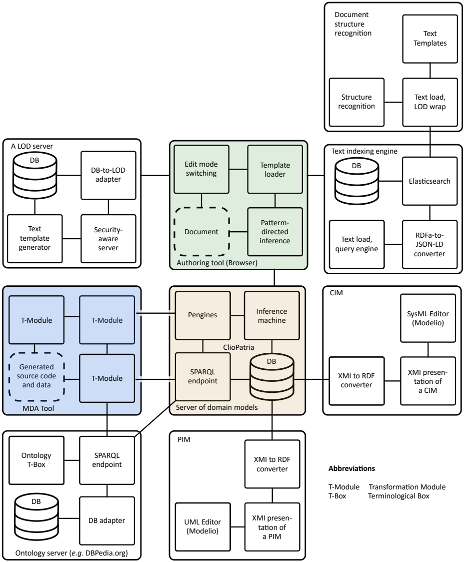


Representation, 代表性

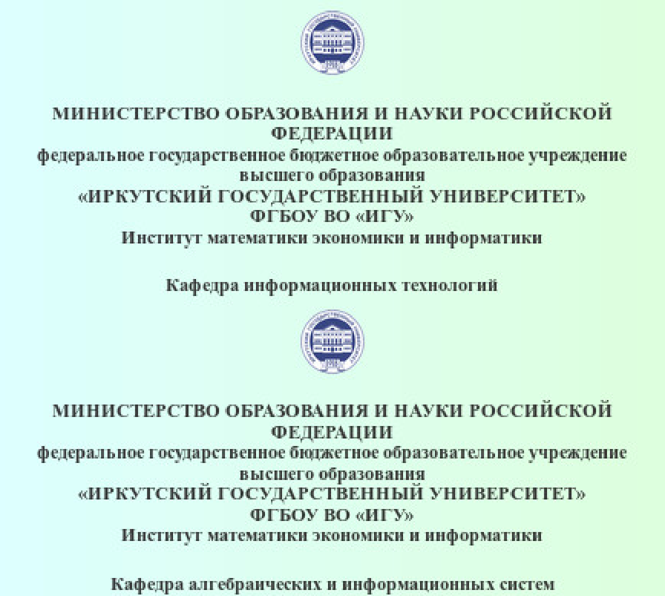
```
<html lang=” ru” xmlns=http://www.w3.org/1999/xhtml
xmlns:taa =http://irnok.net/engine/rdfa-manipulation
xml:lang=” ru” metal:define-macro=” page” >
<head> . . . . </head>
<body prefix=” rdf: http://www.w3.org/1999/...-ns# foaf: http://xmlns.com/foaf/...
imei: imei.html# course: https://irnok.net/college/plan/01..16-...\
%D0\%BA_PB-SM.plm.xml.xlsx-....2.3.1.html#” resource=” #post”
typeof=” schema:CreativeWork sioc:Post prov:Entity” >
<!-- The application control panel -->

<main lang=” ru” resource=” #annotation” typeof=” oa:Annotation” id=” main-doc-cnt” >
<div property=” oa:hasTarget” resource=” #course-work-prog” ></div>
<article property=” oa:hasBody” typeof=” foaf:Document curr:WorkingProgram”
     resource=” #course-work-program” id=” main-document” >
 <div taa:content =” imei:title-page” ></div>
 <div taa:content =” imei:neg-UMK” ></div>
 <section id=” TOC” class=” break-after” > <h2>Table of Contents</h2>
  <div id=” tableOfContents” ></div>
 </section>
 <section id=” course-description” resource=” #description”
      property=” schema:hasPart” typeof=” schema:CreativeWork” >
  <div property=” schema:hasPart” resource=” #purpose”
     typeof=” dc:Text cnt:ContentAsText” >
   <div property=” cnt:chars” datatype=” xsd:string” >
    <h2 property=” dc:title” datatype=” xsd:string” >
      Aims and objectives of the discipline (module)</h2>
    <p>The aim of teaching the discipline ...</p>
   </div>
```

Architecture, 建筑学



Generated list of title page preambles, 生成扉页序言列表



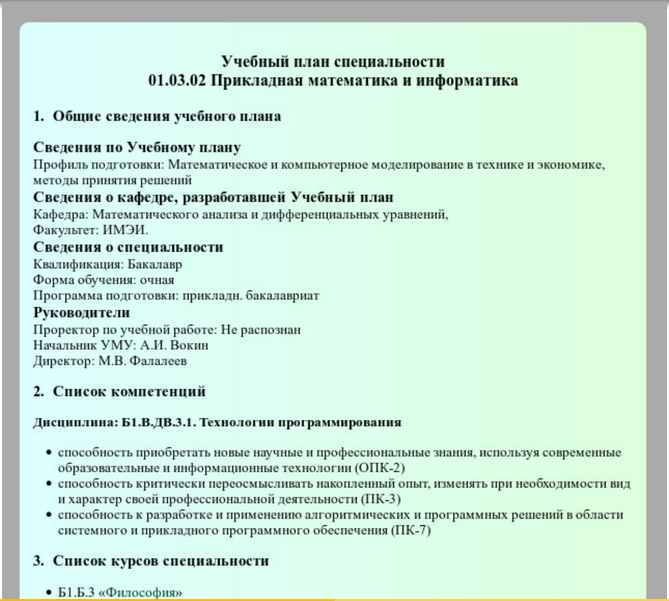
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФГБОУ ВО «ИГУ»
Институт математики экономики и информатики

Кафедра информационных технологий

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФГБОУ ВО «ИГУ»
Институт математики экономики и информатики

Кафедра алгебраических и информационных систем

Generated part of a course description, 生成课程说明的一部分



Учебный план специальности
01.03.02 Прикладная математика и информатика

1. Общие сведения учебного плана

Сведения по Учебному плану
Профиль подготовки: Математическое и компьютерное моделирование в технике и экономике, методы принятия решений
Сведения о кафедре, разработавшей Учебный план
Кафедра: Математического анализа и дифференциальных уравнений,
Факультет: ИМЭИ.
Сведения о специальности
Квалификация: Бакалавр
Форма обучения: очная
Программа подготовки: прикладн. бакалавриат
Руководители
Проректор по учебной работе: Не распознан
Начальник УМУ: А.И. Вокин
Директор: М.В. Фалалеев

2. Список компетенций

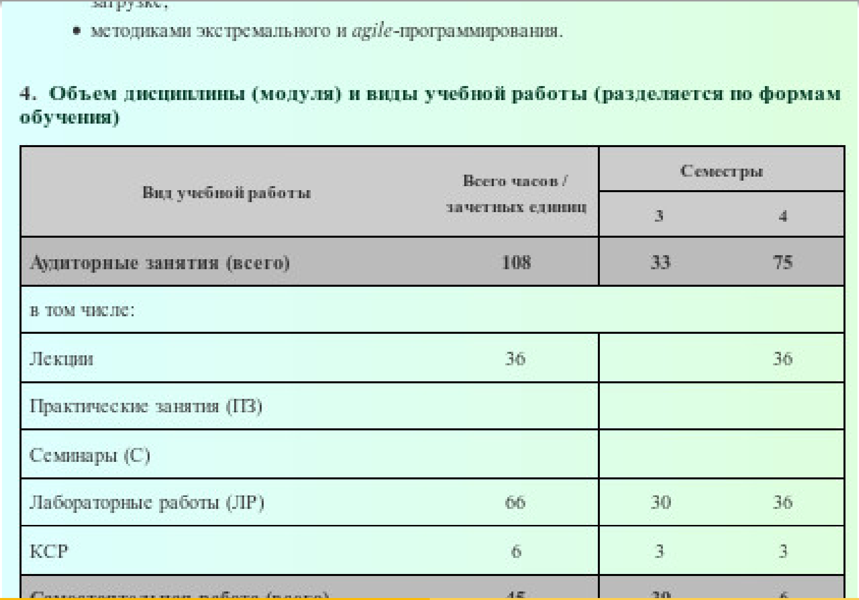
Дисциплина: Б1.В.ДВ.3.1. Технологии программирования

- способность приобретать новые научные и профессиональные знания, используя современные образовательные и информационные технологии (ОПК-2)
- способность критически переосмысливать накопленный опыт, изменять при необходимости вид и характер своей профессиональной деятельности (ПК-3)
- способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения (ПК-7)

3. Список курсов специальности

- Б1.Б.3 «Философия»

Imported time distribution for lecture, seminars, …, 导入讲座和研讨会的时间分配、



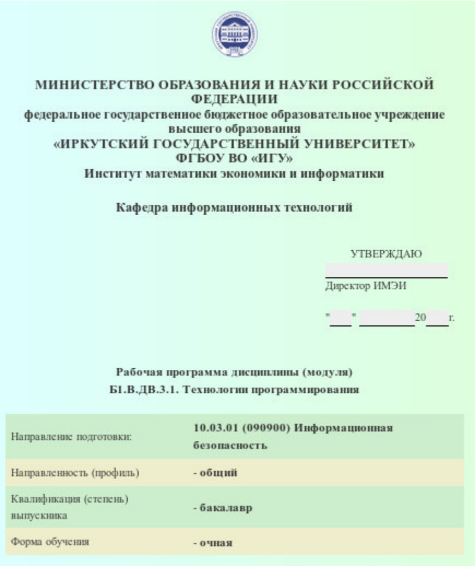
загрузке;

- методиками экстремального и *agile*-программирования.

4. Объем дисциплины (модуля) и виды учебной работы (разделяется по формам обучения)

Вид учебной работы	Всего часов / зачетных единиц	Семестры	
		3	4
Аудиторные занятия (всего)	**108**	**33**	**75**
в том числе:			
Лекции	36		36
Практические занятия (ПЗ)			
Семинары (С)			
Лабораторные работы (ЛР)	66	30	36
КСР	6	3	3
Самостоятельная работа (всего)	45	39	6

Complete document, 完整文件



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФГБОУ ВО «ИГУ»
Институт математики экономики и информатики

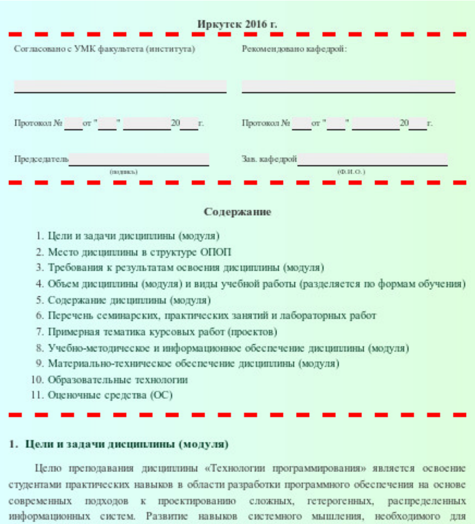
Кафедра информационных технологий

УТВЕРЖДАЮ
____
Директор ИМЭИ

"___" __________ 20___ г.

Рабочая программа дисциплины (модуля)
Б1.В.ДВ.3.1. Технологии программирования

Направление подготовки:	**10.03.01 (090900) Информационная безопасность**
Направленность (профиль)	**- общий**
Квалификация (степень) выпускника	**- бакалавр**
Форма обучения	**- очная**



Иркутск 2016 г.

Согласовано с УМК факультета (института)

____

Протокол № ___ от "___" __________ 20___ г.

Председатель ____________
(подпись)

Рекомендовано кафедрой:

____

Протокол № ___ от "___" __________ 20___ г.

Зав. кафедрой ____________
(Ф.И.О.)

Содержание

1. Цели и задачи дисциплины (модуля)

Целю преподавания дисциплины «Технологии программирования» является освоение студентами практических навыков в области разработки программного обеспечения на основе современных подходов к проектированию сложных, гетерогенных, распределенных информационных систем. Развитие навыков системного мышления, необходимого для

Used ontologies

- Friend-of-a-friend (foaf) - agent information: individuals, legal entities, program agents.
- Provenance (prov) - references between documents.
- Dublin Core (dc) - edited annotation mark up.
- DBPedia resource (dbr) – references to instant objects and classes.
- Schema.org (schema) - Google, Yandex, Yahoo, etc. searchable objects, structural elements.
- The Bibliographic Ontology (bibo) - literature reference mark up.

Conclusion

A tools (components) for digital archive implementation, which allows to device information systems and document processing services with the following features:

- load LOD marked up document, extract, store in a graph and index RDF data;
- retrieve RDF data as triples or as a result of full-text search query;
- combine existing LOD data and its content in new documents dynamically with browser based context inference machine;
- use server-site inference machine (Prolog) to process RDF data upon request from browser’ s part of the system;
- convert created RDFa marked up HTML5 documents into Excel and Word formats.

Applications

- Document authoring automation;
- Context-depended editing;
- Self-organizing global document flows;
- Documents as data sources for information systems.

TabbyXL

Software Platform for Rule-Based Spreadsheet Data Extraction and Transformation

Alexey Shigarov, Vasiliy Khristyuk, et al

shigarov@icc.ru

Motivation

- About arbitrary spreadsheet tables
    - A large volume of valuable data for science and business applications
    - A big variety of layout, style, and content features
    - Human-centeredness (incorrect structure and messy content)
    - No explicit semantics for interpretation by computers

- Challenges
    - How to extract tables from worksheets
    - How to recognize and correct cell structure anomalies
    - How to recover semantics needed for the automatic interpretation
    - How to conceptualize extracted data by using external vocabularies

Background

Table understanding includes the following tasks

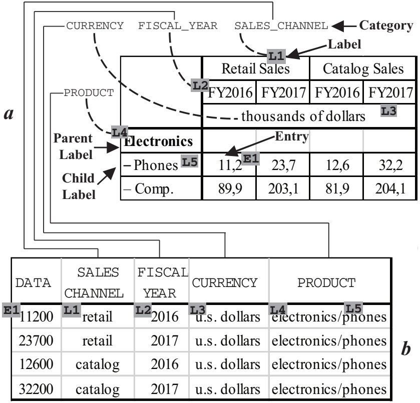
1. Extraction — detecting a table and recognizing the physical structure of its cells
2. Role analysis — extracting functional data items from cell content
3. Structural analysis — recovering internal relationships between extracted functional data items
4. Interpretation — linking extracted functional data items with external vocabularies (general-purpose or domain-specific ontologies)

Contribution

TabbyXL is a software platform aiming at the development and execution of rule-based programs for spreadsheet data extraction and transformation from arbitrary (a) to relational tables (b)

Novelty

- Table object model assigning roles to data items, not cell
- CRL, domain-specific language to express user-defined rules for table analysis and interpretation
- CRL-to-Java translator to synthesize executable programs for spreadsheet data transformation



	Retail Sales		Catalog Sales	
	FY2016	FY2017	FY2016	FY2017
	thousands of dollars			
Electronics				
– Phones	11,2	23,7	12,6	32,2
– Comp.	89,9	203,1	81,9	204,1

DATA	SALES CHANNEL	FISCAL YEAR	CURRENCY	PRODUCT
E1 11200	L1 retail	L2 2016	L3 u.s. dollars	L4 electronics/phones L5
23700	retail	2017	u.s. dollars	electronics/phones
12600	catalog	2016	u.s. dollars	electronics/phones
32200	catalog	2017	u.s. dollars	electronics/phones

Table Object Model

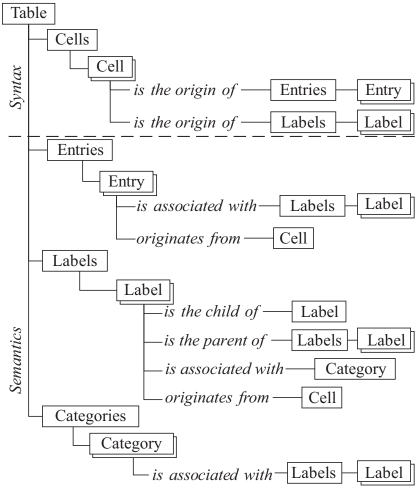
Physical Layer

Cells characterized by layout, style, and content features

Logical Layer

Functional data items and their relationships:

- entries (values)
- labels (keys)
- categories (concepts)
- entry-label pairs
- label-label pairs
- label-category pairs



CRL Grammar

```
rule          = 'rule' <a Java integer literal> 'when' condition
                'then' action 'end' <EOL> {rule} <EOF>
condition     = query identifier [':' constraint {',' constraint}
                [',' assignment {',' assignment}]] <EOL> {condition}
constraint    = <a Java boolean expr>
assignment    = identifier ':' <a valid Java expr>
query         = 'cell' | 'entry' | 'label' | 'category' | 'no cells' |
                'no entries' | 'no labels' | 'no categories'
action        = merge | split | set text | set indent | set mark |
                new entry | new label | add label | set parent |
                set category | group <EOL> {action}
merge         = 'merge' identifier 'with' identifier
split         = 'split' identifier
set text      = 'set text' <a Java string expr> 'to' identifier
set indent    = 'set indent' <a Java integer expr> 'to' identifier
set mark      = 'set mark' <a Java string expr> 'to' identifier
new entry     = 'new entry' identifier ['as' <a Java string expr>]
new label     = 'new label' identifier ['as' <a Java string expr>]
add label     = 'add label' identifier | (<a Java string expr>
                'of' identifier | <a Java string expr>)
                'to' identifier
set parent     = 'set parent' identifier 'to' identifier
set category   = 'set category' identifier | <a Java string expr>
                 'to' identifier
group          = 'group' identifier 'with' identifier
identifier     = <a Java identifier>
```

Cell Cleansing

The actions correct an inaccurate layout and content of a hand-coded table

- <merge> combines two adjacent cells when they share one border
- <split> divides a merged cell that spans n-tiles (row-column intersections) into n-cells
- <set text> modifies a textual content of a cell
- <set indent> modifies a text indentation of a cell

Example

```
when
  cell corner: cl == 1, rt == 1, blank
  cell c: cl > corner.cr, rt > corner.rb
then
  split c
```

Role Analysis

The actions recover entries and labels as functional data items presented in a table

- <set mark> annotates a cell with a user-defined tag that can be used in subsequent table analysis
- <new entry> (<new label>) creates an entry (label) from a cell content with the use of an optional string processing

Example

```
when
 cell corner: cl == 1, rt == 1, blank
 cell c: cl > corner.cr, rt > corner.rb
then
 new entry c
```

Structural Analysis

The actions recover pairs of two kinds: entry-label and label-label

- <add label> associates an entry with a label
- <set parent> binds two labels as a parent and its child

Example

```
when
  cell c1: cl == 1
  cell c2: cl == 1, rt > c1.rt, indent == c1.indent + 2
  no cells: cl == 1, rt > $c1.rt, rt < $c2.rt, indent == $c1.indent
then
  set parent c1.label to c2.label
```

Interpretation

The actions serve to recover label-category pairs

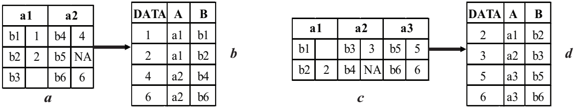
- <set category> associates a label with a category
- <group> places two labels to one group that can be considered as an undefined category

Example

```
when
 label l1: cell.mark == ” stub”
 label l2: cell.mark == ” stub” , cell.rt == l1.cell.rt
then
 group l1 with l2
```

Illustrative Example

The transformation of arbitrary tables with the same layout features (a and c) to their canonicalized versions (b and d)



a1		a2	
b1	1	b4	4
b2	2	b5	NA
b3		b6	6

a

DATA	A	B
1	a1	b1
2	a1	b2
4	a2	b4
6	a2	b6

b

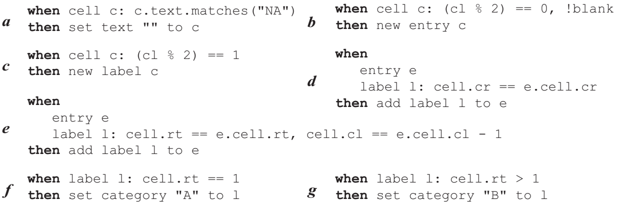
a1		a2		a3	
b1		b3	3	b5	5
b2	2	b4	NA	b6	6

c

DATA	A	B
2	a1	b2
3	a2	b3
5	a3	b5
6	a3	b6

d

The ruleset for the cell cleansing (a), role analysis (b, c), structural analysis (d, e), and interpretation (f, g)



a

```
when cell c: c.text.matches("NA")
then set text "" to c
```

b

```
when cell c: (cl % 2) == 0, !blank
then new entry c
```

c

```
when cell c: (cl % 2) == 1
then new label c
```

d

```
when
   entry e
   label l: cell.cr == e.cell.cr
then add label l to e
```

e

```
when
   entry e
   label l: cell.rt == e.cell.rt, cell.cl == e.cell.cl - 1
then add label l to e
```

f

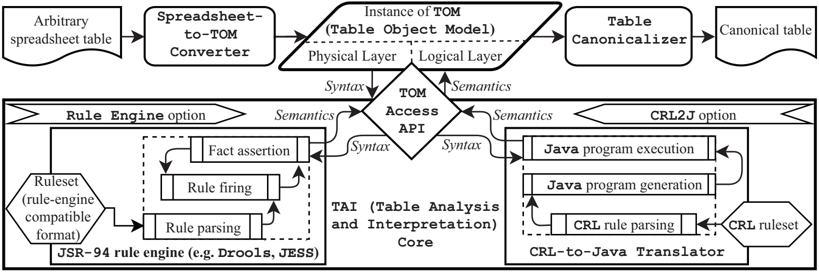
```
when label l: cell.rt == 1
then set category "A" to l
```

g

```
when label l: cell.rt > 1
then set category "B" to l
```

This example is reproducible at https://codeocean.com/capsule/5326436

Architecture



Two options are provided

Rule Engine option

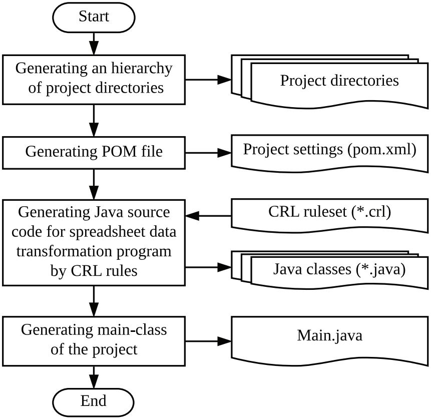
Executing a ruleset in an appropriate format with a JSR-94 compatible rule engine (e.g. Drools, Jess)

CRL2J option

Translating a ruleset expressed in CRL to an executable Java program

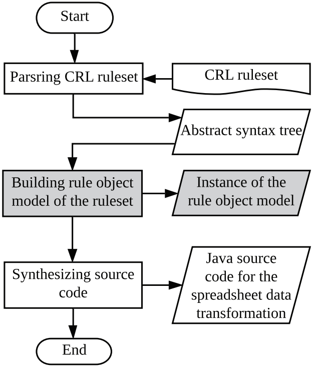
CRL2J Translation

Workflow for generating a Maven-project of a spreadsheet data transformation program



Start
Generating an hierarchy of project directories
Project directories
Generating POM file
Project settings (pom.xml)
Generating Java source code for spreadsheet data transformation program by CRL rules
CRL ruleset (*.crl)
Java classes (*.java)
Generating main-class of the project
Main.java
End

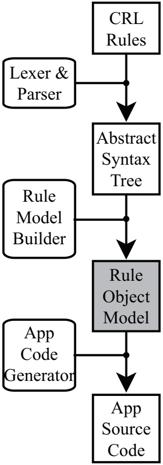
Workflow for translating a CRL ruleset to Java source code



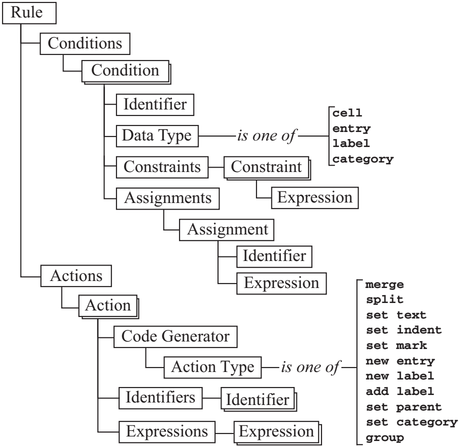
Start
Parsring CRL ruleset
CRL ruleset
Abstract syntax tree
Building rule object model of the ruleset
Instance of the rule object model
Synthesizing source code
Java source code for the spreadsheet data transformation
End

CRL2J Translation

In the Workflow



Rule Object Model



CRL2J Translation

Example (Source Rule)

```
when
 cell corner: cl == 1, rt == 1, blank
 cell c: cl > corner.cr, rt > corner.rb, ! marked
then
 set mark ” @entry” to c
 new entry c
```

Example (Fragment of the Generated Java Code)

```
...
Iterator<CCell> iterator1 = getTable().getCells();
while (iterator1.hasNext()) {
 corner = iterator1.next();
 if ((corner.getCl() == 1) && (corner.getRt() == 1) && ...
  Iterator<CCell> iterator2 = getTable().getCells();
  while (iterator2.hasNext()) {
...
```

Performance Evaluation

The results of the transformation of 200 tables of Troy200 dataset

	Role analysis		Structural analysis	
	Type of instances			
Metrics	entries	labels	entry-label pairs	label-label pairs
Recall	0.9813 $\frac{16602}{16918}$	0.9965 $\frac{4842}{4859}$	0.9773 $\frac{34270}{35066}$	0.9389 $\frac{1951}{2078}$
Precision	0.9996 $\frac{16602}{16609}$	0.9364 $\frac{4842}{5171}$	0.9965 $\frac{34270}{34389}$	0.9784 $\frac{1951}{1994}$
F-score	0.9904	0.9655	0.9868	0.9582

Metrics

$$\text{recall} = \frac{|R \cap S|}{|S|} \qquad \text{precision} = \frac{|R \cap S|}{|R|}$$

S is a set of instances in a source table, R is a set of instances in its canonical form

All data and steps to reproduce the results are available at http://dx.doi.org/10.17632/ydcr7mcrtp.5

Performance Evaluation

The comparison of the running time by using TabbyXL with three different options for transforming 200 tables of Troy200 dataset

Running time of	CRL2J	Drools	Jess
Ruleset preparation (t_1)	2108* ms	$1711^{\dagger}$ ms	$432^{\dagger}$ ms
Ruleset execution (t_2)	367** ms	$1974^{\ddagger}$ ms	$4149^{\ddagger}$ ms

* t_1 — a time of parsing and compiling the original ruleset into a Java program
** t_2 — a time of executing the generated Java program

† t_1 — a time of parsing the original ruleset and adding the result into a rule engine session
‡ t_2 — a time of asserting facts into the working memory and matching rules against the facts

For testing, we used 3.2 GHz 4-core CPU

Comparison with Others

Role Analysis

- Contest task: The segmentation of a table into typical functional cell regions
- Testing dataset: Troy200
- Contestant: MIPS (TANGO)
- Accuracy: MIPS (TANGO) — 0.9899 vs. TabbyXL — 0.9950

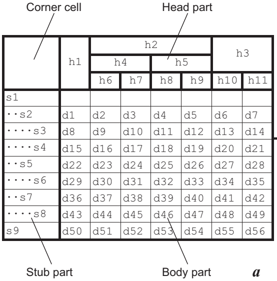
Structural Analysis

- Contest task: The extraction of header hierarchies from tables
- Testing dataset: A random subset of SAUS[a]
- Contestant: Senbazuru
- F-score: Senbazuru — 0.8860 vs. TabbyXL — 0.8657

[a] http://dbgroup.eecs.umich.edu/project/sheets/datasets.html

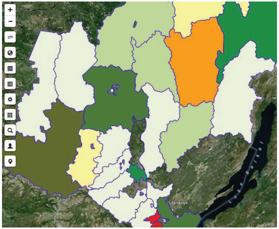
Application Experience

Populating a web-based statistical atlas of the Irkutsk region — (b) via extracting data from government statistical reports — (a)


Corner cell
Head part
Stub part
Body part
a

	h1	h2				h3	
		h4		h5			
		h6	h7	h8	h9	h10	h11
s1							
··s2	d1	d2	d3	d4	d5	d6	d7
····s3	d8	d9	d10	d11	d12	d13	d14
····s4	d15	d16	d17	d18	d19	d20	d21
··s5	d22	d23	d24	d25	d26	d27	d28
····s6	d29	d30	d31	d32	d33	d34	d35
··s7	d36	d37	d38	d39	d40	d41	d42
····s8	d43	d44	d45	d46	d47	d48	d49
s9	d50	d51	d52	d53	d54	d55	d56

DATA	HEAD	STUB
d1	h1	s1|s2
d2	h2|h4|h6	s1|s2
d3	h2|h4|h7	s1|s2
d4	h2|h5|h8	s1|s2
d5	h2|h5|h9	s1|s2
d6	h3|h10	s1|s2
d7	h3|h11	s1|s2
d8	h1	s1|s2|s3
d9	h2|h4|h6	s1|s2|s3
d10	h2|h4|h7	s1|s2|s3
d11	h2|h5|h8	s1|s2|s3
d12	h2|h5|h9	s1|s2|s3
d13	h3|h10	s1|s2|s3
d14	h3|h11	s1|s2|s3
...	...	...
d56	h3|h11	s9

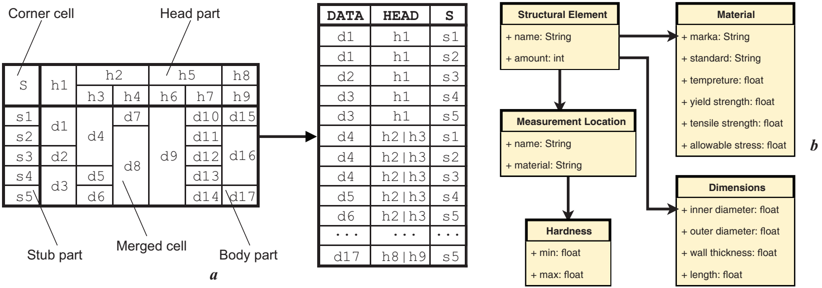


b

The more detail can be found at https://github.com/tabbydoc/tabbyxl/wiki/statistical-atlas

Application Experience

Generating conceptual models — (b) from arbitrary tables presented in industrial safety inspection reports — (a)



DATA	HEAD	S
d1	h1	s1
d1	h1	s2
d2	h1	s3
d3	h1	s4
d3	h1	s5
d4	h2|h3	s1
d4	h2|h3	s2
d4	h2|h3	s3
d5	h2|h3	s4
d6	h2|h3	s5
. . .	. . .	. . .
d17	h8|h9	s5

The more detail can be found at https://github.com/tabbydoc/tabbyxl/wiki/industrial-safety-inspection

Conclusions & Further Work

- Impact on software development for spreadsheet data management
  - Table object model associating functional roles with data items
  - Table analysis and interpretation driven by user-defined rules
  - Formulated actions to recover missing semantics of arbitrary tables
  - Translation of rules to executable spreadsheet transformation programs
- Limitations
  - The inaccurate cell structure prevents the table analysis
  - The very limited interpretation (without external vocabularies)
- Further work
  - Rearrangement of cell structure by using visual (human-readable) cells
  - Detecting derived data by spreadsheet formulas
  - Enriching the table analysis by named entity recognition
  - Linking extracted data items with LOD cloud

Thanks

Read more about the project at
http://td.icc.ru

The project source code is available at
https://github.com/tabbydoc/tabbyxl

Domain Knowledge Graphs Induction from Tables

Tables are the most available sources of information. They are valuable data sources for Knowledge Bases (KB)

Knowledge Base Construction Populating with document and structured table extracted data

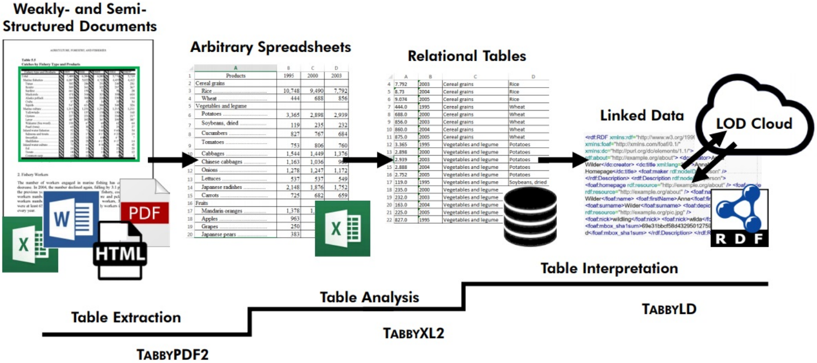
Knowledge Base Population Populating with recognized new facts on entities from big text corpses

Knowledge base Augmentation Populating with relations with table data.

1. (Ré, 2014) Ré C., et al. Feature engineering for knowledge base construction. IEEE Data Eng. Bull., 37, 26–40, (2014).
2. (Balog, 2018) Balog K. Populating knowledge bases. Entity-Oriented Search. INRE, 39, 189–222, (2018).
3. (Zhang & Balog, 2020) Zhang S. & Balog K. Web table extraction, retrieval, and augmentation: A survey. ACM Trans. Intell. Syst. Technol., 11, (2020).

Automatic Table Interpretation

There three main stages of Automatic table interpretation (Shigarov, 2017)



Arbitrary Spreadsheets

	A	B	C	D
1	Products	1995	2000	2003
2	Cereal grains			
3	Rice	10,748	9,490	7,792
4	Wheat	444	688	856
5	Vegetables and legume			
6	Potatoes	3,365	2,898	2,939
7	Soybeans, dried	119	235	232
8	Cucumbers	827	767	684
9	Tomatoes	753	806	760
10	Cabbages	1,544	1,449	1,376
11	Chinese cabbages	1,163	1,036	[illegible]
12	Onions	1,278	1,247	1,172
13	Lettuces	537	537	549
14	Japanese radishes	2,148	1,876	1,752
15	Carrots	725	682	659
16	Fruits			
17	Mandarin oranges	1,378	[illegible]	
18	Apples	963		
19	Grapes	250		
20	Japanese pears	383		

Relational Tables

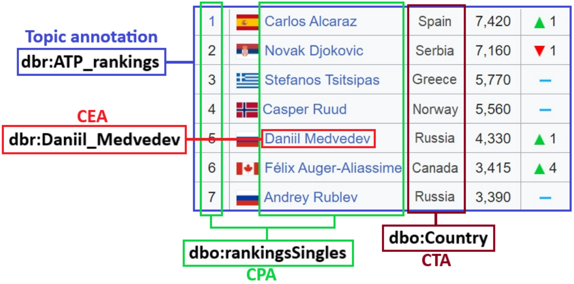
	A	B	C	D
4	7.792	2003	Cereal grains	Rice
5	8.73	2004	Cereal grains	Rice
6	9.074	2005	Cereal grains	Rice
7	444.0	1995	Cereal grains	Wheat
8	688.0	2000	Cereal grains	Wheat
9	856.0	2003	Cereal grains	Wheat
10	860.0	2004	Cereal grains	Wheat
11	875.0	2005	Cereal grains	Wheat
12	3.365	1995	Vegetables and legume	Potatoes
13	2.898	2000	Vegetables and legume	Potatoes
	2.939	2003	Vegetables and legume	Potatoes
15	2.888	2004	Vegetables and legume	Potatoes
16	2.752	2005	Vegetables and legume	Potatoes
17	119.0	1995	Vegetables and legume	Soybeans, dried
18	235.0	2000	Vegetables and legume	
19	232.0	2003	Vegetables and legume	
20	163.0	2004	Vegetables and legume	
21	225.0	2005	Vegetables and legume	
22	827.0	1995	Vegetables and legume	

1. (Shigarov, 2017) Shigarov A., Mikhailov A. Rule-based spreadsheet data transformation from arbitrary to relational tables. Information Systems, 71, 123-136 (2017).

Semantic Table Interpretation

Semantic interpretation (Annotation) of tables (Semantic Table Interpretation, STI) is a recognition of mutual and external relations between elements of table content. Externals relate to an enterprise KG and/or a global KG (e.g. DBPedia.org).

- Cell-Entity Annotation (CEN)
- Column-Type Annotation (CTA)
- Column Property Annotation (CPA)
- Topic Annotation



Topic annotation

dbr:ATP_rankings

CEA

dbr:Daniil_Medvedev

1	Carlos Alcaraz	Spain	7,420	▲ 1
2	Novak Djokovic	Serbia	7,160	▼ 1
3	Stefanos Tsitsipas	Greece	5,770	–
4	Casper Ruud	Norway	5,560	–
5	Daniil Medvedev	Russia	4,330	▲ 1
6	Félix Auger-Aliassime	Canada	3,415	▲ 4
7	Andrey Rublev	Russia	3,390	–

dbo:rankingsSingles

CPA

dbo:Country

CTA

Cell-Entity Annotation

CEA comprises tho sequential steps as follows:

1. Select a candidate entity set from DBPedia.org for each value of a cell via SPARQL endpoint and DBPedia lookup.
2. Disambiguation

A SPARQL-query matching words of a phrase.

```
SELECT DISTINCT (str(?subject) as ?subject)
WHERE {
   ?subject a ?type .
   ?subject rdfs:label ?label .
   ?label <bif:contains> ” .*%value1*.”  AND ” .*%value2*.”  ... .
   FILTER NOT EXISTS { ?subject dbo:wikiPageRedirects ?r2 } .
   FILTER (!strstarts(str(?subject), ” http://dbpedia.org/resource/Category:” )) .
   FILTER (!strstarts(str(?subject), ” http://dbpedia.org/property/” )) .
   FILTER (!strstarts(str(?subject), ” http://dbpedia.org/ontology/” )) .
   FILTER (strstarts(str(?type), ” http://dbpedia.org/ontology/” )) .
   FILTER (lang(?label) = ” en” )
}
ORDER BY ASC(strlen(?label))
LIMIT 100
```

Evaluation on Test Table Sets

A well-known precision measurement (accuracy) is used for assessment

$$\text{Accuracy} = \frac{CC}{NC},$$

where CC is the number of the correctly related columns to a categorical entity, and CN is the total number of columns.

Recognition stage	T2Dv2	Tough_Tables	GitTables
Stage 2, Atomic column classification	0.994	0.956	0.938
Stage 3, Column entity identification	0.924	–	–

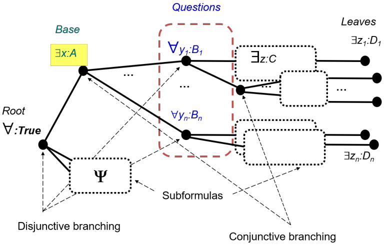
Comparison with analogs

	TAIPAN	TableMiner+	T2Dv2	MantisTable
Column entity identification	0.540	0.871	0.924	0.979

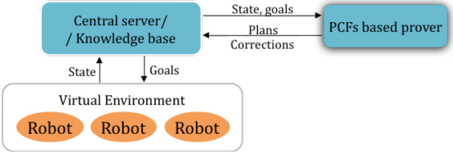
Knowledge Representation and Reasoning: the PCF-Calculus

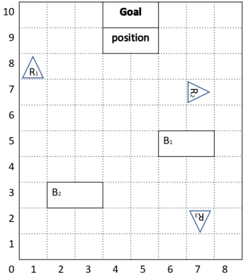
The main properties of the language of positively constructed formulas (PCF) and its calculi:

- PCFs have a large-block structure (tree-like) and consist of only positive quantifiers ∃ and ∀
- the PCF-based calculus have a unique inference rule
  - the proof in the PCF-calculus is organized as a question-answering procedure
  - PCF-calculus is both machine-oriented and human-oriented; it is compatible with heuristics
  - the semantic of the PCF-calculus can be changed without modifying axioms and the inference rule



PCF-Based Method for Problem Solving





- The goal of the team of robot is to transport blocks to the target area
- Each block can be dragged by two or more robots
- The current state of the World and the goal of the group are formalized in PCF
- The PCF-based prover and a selection mechanism produce the optimal joint plan of actions for the team
- The current plan can be easily modified whenever the state of the World is changed

A Master Degree Program. Semantic Technologies and Multiagent Systems

It is a joint effort of Saint-Petersburg Electrotechnical University (LETI), Irkutsk State University, and ISDCT SB RAS. Main subjects.

- Computation Geometry, Digital Signal Processing, Internet of Things,
- Semantic web, Semantic web Information System Development,
- AI Basics, Knowledge representation, Object-oriented Logic Programming,
- Answer Set Programming (SAT), Natural Language Processing,
- Machine Learning, Neural Networks, Deep Learning,
- Multiagent Systems, Optimization with Multiagent Systems.

Started at 2022-09-01.

https://etu.ru/sveden/education/programs/semanticheskie-tehnologii-i-mnogoagentnye-sistemy-01.04.02.html

Conclusion (the final one)

- Classic knowledge-based systems are powerful AI tools for solving wide class of recognition problems and synthesis of various kind: source code, data objects, control
- Contemporary means combine classic and new approaches
- Less dependent on computational resources (as compared to machine learning)
- Allow justification of the produced solutions
- Cover a larger set of tasks
- Natural for math science, and require higher level of AI education

Thank You!