

# RDB-to-RDF 的技术方法和工具综述\*

刘 振<sup>1,2,3</sup> 张智雄<sup>2</sup>

<sup>1</sup>(徐州工程学院管理学院 徐州 221008)

<sup>2</sup>(中国科学院文献情报中心 北京 100190)

<sup>3</sup>(中国科学院大学 北京 100049)

**摘要:**【目的】对 RDB-to-RDF 的技术方法和工具进行梳理和综述,提炼其关键技术。【文献范围】使用 Elsevier、Springer、中国知网数据库进行追溯检索 RDB 到 RDF 转换的技术和工具等相关中英文文献。【方法】使用文献调研法,并按研究主题进行综述。【结果】从映射思路、技术方法和实现方式等角度进行归纳和分析,比较常用的映射工具和语言的重要特性以及适用场合,列举典型的应用场景【局限】映射工具比较时缺少具体的定量测评。【结论】本研究有助于全面了解 RDB 转换 RDF 中相关关键技术、工具以及主要应用场景。

**关键词:** RDB RDF RDB-to-RDF

**分类号:** TP311

## 1 引言

当今社会,正在从文档网络向数据网络加速发展,大量数据存储在关系数据库里,关系数据库在效率等方面依旧具有一定的优势,但欠缺语义。如何采用相互关联的 RDF 三元组集发布原先孤立的数据库,从而形成一个全球范围的数据空间,这是数据网实施的重点内容,这个过程通常称为 RDB-to-RDF 或 RDB2RDF,即把传统关系型数据库存储的数据转换成 RDF 三元组<sup>[1]</sup>。为了确保以前开发的应用的可持续性以及利用 RDB 系统几十年来的属性(可扩展性、ACID 属性、安全和性能优化等)<sup>[2]</sup>,目前最好的方式还是数据在传统的 RDBs 存储和传递,需要使用 RDB 转 RDF 技术,访问关系型数据,转换成 RDF 三元组。人们迫切需要在关系模型和 RDF 之间架起概念级别的桥梁。把这些大量存储在 RDB 里的数据映射到 RDF 中,已经越来越受到相关学者和机构的关注,研发出了各种通用或者特定领域的映射工具或者应用软件,但面对众多的技术和工具,如何选择适合自己的技术和工具,人们会

产生一定的困惑,因此有必要对这些技术和工具进行详细的分析和比较,梳理出 RDB-to-RDF 技术的解决方案思路,指导实际应用。

## 2 映射的主要思路和方法

### 2.1 主要思路

近年来出现了很多 RDB-to-RDF 技术方法和实现策略,不同的技术涉及到不同的解决思路,经过归纳和梳理,笔者认为主要涉及到本体学习、映射语言设计、查询引擎设计三个方面:

(1) 本体工程方法:从关系模式和数据中抽取出自本体的概念和关系,例如简单直接映射方法建立一个临时的本体<sup>[3]</sup>,或者寻找特定数据库设计模式,例如多对多关系和概念包容关系。使用数据挖掘的方法从关系模式中推测出新的本体概念和属性。一些方法创建本体,在数据库和已有本体之间进行可能的匹配,或者从数据库模式得到额外的概念和属性扩展已有的本体。一般情况下,这种方法要结合映射语言和查询引擎使用。

收稿日期:2014-06-30

收修改稿日期:2014-07-27

\*本文系国家“十二五”科技支撑计划项目“科技知识组织体系共享服务平台建设”(项目编号:2011BAH10B03)的研究成果之一。

(2) 通用映射语言的定义: 该思路关注的是描述到已有本体的映射的能力, 确保描述语言的表达能力能够覆盖很多的情况, 既能满足简单直接映射, 又适用于数据库和目标本体相似度很低的情况。

(3) 转换引擎的实现: 主要是依赖于能够响应 SPARQL 查询的处理引擎<sup>[4]</sup>, 或者能立刻把整个关系数据转换成 RDF 表示, 在保证性能的同时, 能够处理大量的并发查询, 在响应时间内处理复杂查询。

## 2.2 技术方法的种类和特点

### (1) 直接映射方法

直接映射方法即自动映射, 当不存在关系模式可以映射的领域本体时, 或者目标就是快速将数据资源转化为机读格式, 而不关心语义互操作性, 则采用此方法。直接映射方法能够适应多变的环境, 特别是在数据库快速生成和消失的情况下, 没有时间进行人工的调整, 该方法遵循 Berners-Lee<sup>[5]</sup>定义的简单规则, 基于关系型数据库模式和数据自动生成 URIs, 如表 1 所示。如果需要语义互操作, 稍后可以使用本体校正的方法, 使得本地本体和已经存在的领域本体相一致。W3C 提出了“Direct Mapping”规范<sup>[6]</sup>, 定义了把关系数据库映射为一个 RDF 图的规则, 只关注于映射的自动生成, 而不是映射的表示。SquirrelRDF 的映射也是简单直接的<sup>[7]</sup>, RDF 词表的类和属性都是直接从 RDB 模式的名称中生成, 而映射到领域本体则延迟到后续的阶段, 通过基于 RDF 工具来完成。

表 1 RDB 和 RDF 的简单映射规则

RDB	RDF
表	类 (URI:namespace/database/table)
列	属性 (URI:namespace/database/table/column)
行	资源 (URI:namespace/database/table/primaryKey)

该方法提供专门定制的 RDB-to-RDF 映射方案, 尽量降低 Web 应用开发者的准入门槛, 不需要专门定义一个新的映射语言, 直接利用和扩展 SQL 概念, 把数据库查询结果转换成 RDF 和关联数据。使用 Multiple-Table-to-Class 的方法, 关系型数据转换的结果需要一些结构, 就可以表示成 RDF。每一个视图的第一列包含用于产生 URIs 实例的识别符, 这个识别符号可以是数据库表的主键。结果视图的字段名字将用于产生属性的 URIs。

### (2) 领域语义驱动映射

领域语义驱动映射依靠已有的、定义良好的、具体领域的语义, 使用现有的本体概念和属性转换数据库, 实现复杂的映射<sup>[8]</sup>。

领域语义驱动映射的实现策略主要有以下两种:

映射描述依赖 SQL 查询给出数据。该策略的表达能力受限于 SQL 自身能力, 不能解决复杂的情况。从 SPARQL 查询到 SQL 的重写是直接的, 开发者可以调整查询, 但要依赖于数据库系统的性能。该策略的优势是 SQL 被人们熟知, 有利于数据提供者将其采用为映射语言, 而不需学习一门新的映射语言。

专门的复杂映射语言。由于数据库和本体都是分别设计的, 两者之间的相似度比较低, 映射语言能够高效表达和描述 RDB 和 RDF 的映射, 架起两者的桥梁。这种方式不会受限于 SQL 的表达力, 能够满足特定的复杂需要, 例如关键字查找、正则表达式匹配、自然语言处理和数据挖掘等。当然很多的映射语言, 例如 R2RML<sup>[9]</sup>和 D2RQ<sup>[10]</sup>, 可以同时利用这两个策略: SQL 语句查询和特定的映射描述语言。

### (3) 两种方法的比较分析

直接映射方法把关系数据库的内容转换成 RDF, 简单方便, 映射速度快, 但它不能捕捉数据的真正语义。理想情况下, 关系模式元数据隐藏着语义, 语义包含在利用这些数据库的应用程序编码的具体领域规则中。数据库是性能最优化的结果, 但数据库模式不能完全描述领域。直接映射方法有时候会出现一些问题, 从而导致转换失败, 主要原因有:

对公开数据和私有数据的识别问题。网络应用经常包含不便公开的数据, 例如密码、邮箱地址、技术参数和配置, 自动分辨哪些是严格保密的, 哪些是重要的, 哪些是不相关的信息, 非常困难。

已有词汇的重用。由于缺乏数据库模式中领域语义的机读描述, 因此, 数据库实体(表、字段名)和已有 RDF 词汇集之间的映射有时会失败。

缺乏数据库模式描述。网络应用中使用的很多数据库模式对外键或约束条件缺乏全面详细的定义。数据库模式经常改变, 而又不经常遵守命名习惯, 因此自动生成首次映射后, 需要一定程度的人工调整, 例如解决没有主键的表和字段主键组合, 利用启发式方法发现 RDB 模式下的隐藏子类关系。

领域语义驱动映射方法在转换过程中较为完整地保留了 RDB 数据蕴含的语义, 可以实现较为复杂的映射。但要依赖于专门的领域本体和映射语言, 转化过程较为繁琐, 耗时间较长。

### 3 映射的实现方式

#### 3.1 数据实体化

数据实体化是把源数据库转换成图表示的静态方法,也称为抽取转换装载过程(Extraction-Transformation-Loading, ETL)<sup>[11]</sup>,即对数据库的整个内容应用映射规则,创建一个等效的 RDF 图。当实体化过程完成以后, RDF 图能装载到三元组仓储,通过 SPARQL 查询引擎访问实例,在三元组仓储和查询引擎能力范围下能够尽量地集成大量的数据资源,方便进一步进行 RDF 数据处理、分析、推理以及执行推理规则。这个过程中 RDF 数据立即有效,第三方推理工具能够应用复杂的蕴含机制。因为复杂推理的任务在较早的过程被执行,复杂的查询能够被响应,不需要考虑运行性能。但这种方法的主要局限有:

(1) 不支持超大数据集,因为产生的图的大小会超过内存容量。

(2) 欠缺处理失效数据的能力。如果一个应用频繁更新关系型数据, RDF 图很快就失效了,解决问题的方法是定期运行抽取过程,这就要权衡装载的代价和应用程序容忍失效数据的程度。

#### 3.2 按需映射

这种方式下数据保存在传统数据库中,不管使用何种方式访问数据,对目标 RDF 数据的查询必须重新写入到 SQL,再查询评估时间。这种策略适用于超大数据集,保证返回的数据是最新的,因为没有数据的复制。但是如果实现访问控制策略受到隐私或保密限制,查询性能会受到影响,实施蕴含机制使很多数据源需要整合在一起。

### 4 常用工具和映射语言分析

#### 4.1 常用映射工具

##### (1) Virtuoso RDF 视图

Virtuoso RDF 视图<sup>[12]</sup>把 SQL SELECT 查询的结果集转换成三元组集合,用虚拟的 RDF 图进行表示,而不需要建立物理的 RDF 数据集, Virtuoso RDF 视图是由四图模式组成,定义了从一系列的关系数据库字段到三元组的映射。四图模式用 Virtuoso 元模式语言表示,从语法的角度,元模式语言与 SQL DDL 很相似,也支持 SPARQL 形式的表示。

##### (2) Triplify

Triplify<sup>[13]</sup>是从关系数据库里发布关联数据的一种轻量级方法,它基于把 HTTP-URI 请求映射到 RDB 查询,并且把结果集转换成 RDF 陈述。主要目的是把互联网上大量信息以结构化形式存放在关系数据库里,但被网络应用程序以 HTML 方式进行发布(例如内容管理系统、博客等)。映射这些主流的网络应用程序的 RDB 模式促进了语义网的应用,因为这些应用被广泛部署, Triplify 映射是用 PHP 脚本实现的。

##### (3) RDBToOnto

RDBToOnto<sup>[14]</sup>是一个具有高可配置性的、面向用户的工具,简化了从关系型数据库中生成本体的设计和实现,用户可以自行定义规则,整个过程是交互和迭代的,支持从访问输入数据库到填充本体的生成的转换过程。学习参数的设置和过程控制都是通过成熟复杂的界面完成,简化开发。除了使用自动映射方法自动生成本地本体,还有基于关系型模式和数据的半自动化方法,在内容中发现子类,细化从关系模式中得到的类,例如在字段名称中发现词汇线索(匹配预先定义好的关键词“类型”),使用数据冗余发现分类模式。使用第三方工具 LATINO 消除冗余,进行数据库优化。RDBToOnto 不仅支持完整的转换过程,还可以利用连接器和转换器实现新的学习方法。

##### (4) Datalift

Datalift<sup>[15]</sup>包含一系列工具集用于不同格式(关系型数据库、CSV、XML 等)的原始结构化数据的发布过程,帮助用户发布和连接网络上的数据集,这些操作都是基于 Web 界面完成的,用户选择一个数据源,采用直接映射的方法进行转换,虽然开始使用直接映射的方法,但不同的模块为进行后续的调整提供了足够的灵活性,完全可以与领域语义驱动映射方法媲美。

##### (5) R2O

R2O<sup>[16]</sup>是一个可扩展的、基于 XML 语法的声明语言,用来描述 RDB 模式和 RDFS 或 OWL 里的本体的映射。它假设前提是 RDB 和本体模式是预先存在的。它适用于本体和 RDB 之间的相似度很低的情况,具有足够的表达能力能够处理复杂的映射,例如两个模型之间在丰富性、通用性和结构上差距很大。在映射定义中可以用来检测不一致性和歧义性。ODEMapster 引擎可以使用 R2O 文档根据查询执行转换或者在批处理的模式下生成 RDF 转储。

### (6) D2RQ

D2RQ<sup>[17]</sup>提供了访问关系型数据的多种方式的集成环境,包括 RDF 转储,基于 Jena、Sesame API 访问和 D2SQ 服务器的 SPARQL 终端。D2RQ 把关系型数据作为 RDF 图发布的映射平台,只提供关系型数据库上只读的 RDF 视图,根据关联数据的原则,将关系型数据库暴露在网络上,允许应用程序在语义网上读、写、更新数据。它是在每一条三元组基础上进行操作,当一条更新破坏了一些约束,被底层的关系数据库管理系统拒绝,D2RQ++ 就把被拒绝的三元组添加到辅助 RDF 库里,一旦满足数据库的约束条件,RDF 库会定期和数据库进行合并,但这种方法会造成工作在 RDF 视图的 RDF 应用程序和工作在原始数据的应用程序之间的不一致。在辅助 RDF 库的三元组和关系型数据库合并之前,SPARQL 查询不等同于 SQL 的映射。这种定期合并的解决方案一定程度解决了由于外键约束引起的不一致,通过首先在关系型数据库中插入被参照的数据,然后插入参照数据,但是这种方案不能解决非空的约束。因为在第一条插入语句,非空的约束必须立即满足。作用在一个属性上的非空约束的三元组不能插入到数据库里,D2RQ/Update 把三元组按照一个主题进行分组,然后批量更新。从被映射的数据库的模式中读取约束条件,在更新时为了保护外键约束,根据参照完整性约束条件进行拓扑排序。使用 D2RQ 将关系数据库发布为 RDF 有两个步骤:

生成一个映射文件,该映射文件说明了数据库的表和列具体是如何与输出本体中的类和属性进行映射的。该工具本身并不生成输出本体,但是它隐式生成了一个本体,因为它在一个一致的命名空间上定义了一个类和属性的集合,而 D2RQ 处理程序在将 RDB 转换为 RDF 时会用到该命名空间。

建立一个由第一阶段生成的映射文件设定的 D2RQ 实例,并且将该实例包装为一个 Jena 模型。一旦这些工作完成之后,就可以像使用一般 RDF 模型那样使用该模型。

### (7) R2RML

R2RML 是 W3C RDB2RDR 工作组制定的,用来描述把数据从关系型数据转换成 RDF 表示的映射、独立于厂商的通用语言,适用于只读数据访问,定义了相关的词汇表、映射规则和语法、映射框架和工作机制,提供了跨表格映射到更高级的 RDF 模型功能。它的中心思想是尽可能实现直接的、简单的映射。例如,

这种形式的映射允许用户使用 SPARQL 查询关系数据库,过去只能通过使用 SQL 实现。它还允许跨关联数据网在行级别链接关系数据库,消除了手动创建的中间层,并且动态地将关系数据公开为 RDF。R2RML 可以轻松地将所有封闭的关系数据库转换为 Web 可访问的模式。

### (8) Relational OWL

Relational OWL<sup>[18]</sup>是基于本体的关系型数据的模式组件的表示格式,用来改善对等数据库的数据交换能力。由于对等数据库具有分布性和易逝性,数据和模式会频繁变化,因此需要交换格式能够迅速被解读,不需要本体调整或内容协调。它定义了 OWL-Full 本体表示 RDB 数据和模式。这个本体为表、字段、主键和外键、数据类型定义了类和类间的关系。利用一个本地本体和关系型数据本身表示关系型模式组件。

除此以外,还有 eD2R<sup>[19]</sup>、R3M<sup>[20]</sup>等映射语言。

## 4.2 映射工具和语言的重要特性分析

在对上述 RDB-to-RDF 各种工具和映射语言的分析基础上,总结出以下重要特性:

(1) 词表重用:将关系实体映射到已有的词表和本体的实例上,这也是直接映射方法和领域语义驱动映射方法的主要区别。

(2) 多对多关系:RDBS 需要一个特殊的连接表来表示概念之间的多对多关系。这个表的字段都是其他表的外键。将这个连接表转换成简单的三元组,而不能像直接映射方法那样,转换成一个类。但 RDF 不需要这类的辅助表格。连接表需要映射成 RDF 属性而不是类。

(3) 完整性约束:提供了一个基本的语义机制。键(主键、外键)约束和其他约束(非空、唯一和检测)等。

(4) 投影操作:在 RDB 表中一些属性不能作为 RDF 表示的内容,例如一些无关属性或敏感属性密码。只选择需要的属性,也就是一个表的字段的子集进行转换,这是任何 RDB 转 RDF 工具中最基本的属性和必备条件。

(5) 数据类型:能够明显表示数据类型信息,处理在 SQL-XSD 映射中的关系型数据类型和 RDF 数据类型的对应关系。

(6) 数据转换:在产生 RDF 三元组之前对数值进行转换,可以执行复杂类型转换,使用多个列计算一

个值。例如字符串操作函数、小数类型转换。文本值在 RDF 中需要不同的表示方式。在 RDB 和 RDF 表示中实现了值的转换。

(7) 逻辑表: SQL 视图或者一个临时 SQL 查询的结果,这不是物理存在的表。

(8) 1 个表转 N 个类: 把一个字段的值作为分类模式,基于这个属性的值把表的元组转换成不同的本体类的实例。这个特性适用于一个 RDB 模式不规范的时候或对表的一个属性的概念具体化的时候。一般会出现两种情况: 一个表被映射多次,每次都是属性的子集,或者根据一个特定的属性值,把表的记录映射成不同的类。

(9) 命名图: RDF 数据集包含多个命名图,把 RDB 的一部分内容指派给特定的命名图。

(10) 空结点: 在转换过程中能够产生空结点,在产生的图中表示出来。实例使用空结点转换没有主键的表。RDF 中的空结点一般表示的实例是没有 RDF URI 参考标识符,但在一个 RDF 图上又是不同的。

(11) 静态元数据: 给生成的图或者 RDF 实体附加静态的元数据,例如许可或来源信息。

(12) 用户定义的实例 URIs: RDB 中的记录转换成用一个 URI 标识的 RDF 实例。可以基于 RDB 模式和数据自动生成这些实例 URIs,或者用户定义产生的 URIs 形式。不仅仅是对主键值的简单重用和组合。

表 2 列出了常见的映射工具和语言具有的特性,其中 DM 表示 Direct Mapping, R-OWL 表示 Relational-OWL。“是”表示支持该特性,“否”表示不支持该特性,“部分”表示部分支持该特性。

表 2 映射工具和语言的特性

特性	DM	R2RML	R3M	eD2R	D2RQ	R2O	Virtuoso	Triplify	R-OWL
词表重用	否	是	是	是	是	是	是	是	否
多对多关系	否	是	是	是	是	否	是	是	否
完整性约束	否	部分	是	部分	部分	部分	部分	部分	部分
投影操作	否	是	是	是	是	是	是	是	是
数据类型	否	是	是	是	是	是	是	是	是
数据转换	否	是	是	是	是	是	是	是	否
逻辑表	部分	是	部分	是	是	是	是	是	部分
1 个表转 N 个类	否	是	是	是	是	部分	是	是	否
命名图	否	是	否	否	否	否	是	否	否
空结点	否	是	否	是	是	是	是	否	是
静态元数据	否	是	否	否	是	否	否	否	否
用户定义的实例 URIs	否	是	是	是	是	是	是	是	否

## 5 具体的领域应用

RDB-to-RDF 的工具和方法在很多领域都得到了广泛的应用。Byrne<sup>[21]</sup>在文化遗产领域内把关系数据库里苏格兰国家古迹记录转换成 RDF,使用一个简单的知识组织系统框架,把整个包含 150 万多条记录的数据库转换成 2 100 万 RDF 三元组。Sahoo 等<sup>[22]</sup>在生命科学领域内,从多个集成的本体内综合领域语义建立从 RDB 到 RDF 的映射,用 XSLT 样式表的 XPath 规则表示出来,使用批处理方法,存储在 Oracle 11 中,建立 RDF 转储。Green 等<sup>[23]</sup>为了建立水污染扩散的预测模型,使用多级 OWL-DL 本体,描述 RDB 空间数据

的集成。第一级本体称为数据本体,用来映射每一个数据源到下一级本体里的概念。数据本体用 D2RQ 映射语言表示。第二级是领域本体,第三级是应用本体,用来连接领域本体,又添加了应用相关的信息。D2RQ 引擎包括空间操作符作为数据源和数据本体的接口。SPARQL 语言用来查询从数据源产生的虚拟 RDF 图。

## 6 结 语

通过以上对 RDB-to-RDF 技术和工具的分析,总结了现有技术方法和工具的特点,详细地讨论了它们的共性和区别,对于这些不同的解决方案有了全面的

认识, 有利于帮助人们选择适合的方式, 在网络上以 RDF 形式发布关系型数据。为了使得数据更加可用、可操作、可互联, 有效地打造数据集成和沟通的平台, 该领域的研究还有很多问题值得关注, 例如如何产生具有丰富语义的 RDF 数据、如何有效使用领域本体、数据集成(把来自多个 RDB 的数据表示成一个 RDF 图)等。

### 参考文献:

- [1] Heese R, Znamirowski M. Resource Centered RDF Data Management [C]. In: Proceedings of the 10th International Semantic Web Conference, Bonn, Germany. 2012.
- [2] Stefanova S, Risch T. Scalable Reconstruction of RDF-archived Relational Databases [C]. In: Proceedings of the 5th Workshop on Semantic Web Information Management. New York: ACM, 2013.
- [3] Shvaiko P, Euzenat J. Ontology Matching: State of the Art and Future Challenges [J]. IEEE Transaction on Knowledge and Data Engineering, 2013, 25(1): 158-176.
- [4] Roshdy M H, Fadel K M, El Yamany H F. Developing a RDB-RDF Management Framework for Interoperable Web Environments [C]. In: Proceeding of the 4th IEEE Eurocon Conference, 2013: 307-313.
- [5] Berners-Lee T. Relational Databases on the Semantic Web [EB/OL]. [2014-04-02]. <http://www.w3.org/DesignIssues/RDB-RDF.html>.
- [6] A Direct Mapping of Relational Data to RDF [EB/OL]. [2014-04-02]. <http://www.w3.org/TR/2012/REC-rdb-direct-mapping-20120927/>.
- [7] SquirrelRDF [EB/OL]. [2014-04-02]. <http://jena.sourceforge.net/SquirrelRDF/>.
- [8] Berardi R, Breitman K, Casanova M A, et al. StdTrip+K: Design Rationale in the RDB-to-RDF Process [C]. In: Proceeding of the 24th International Conference on Database and Expert Systems Applications, Prague, Czech Republic, 2013. DOI: 10.1007/978-3-642-40285-2\_26.
- [9] R2RML: RDB to RDF Mapping Language [EB/OL]. [2014-04-02]. <http://www.w3.org/TR/r2rml/>.
- [10] Eisenberg V, Kanza Y. D2RQ/update: Updating Relational Data via Virtual RDF [C]. In: Proceedings of the 21st International Conference Companion on World Wide Web. ACM, 2012: 497-498.
- [11] Bugiotti F, Goasdoué F, Kaoudi Z, et al. RDF Data Management in the Amazon Cloud [C]. In: Proceedings of the 2012 Joint EDBT/ICDT Workshops. ACM, 2012: 61-72.
- [12] Erling O, Mikhailov I. Virtuoso: RDF Support in a Native RDBMS [A] // Semantic Web Information Management [M]. Springer Berlin Heidelberg, 2010: 501-519.
- [13] Auer S, Dietzold S, Lehmann J, et al. Triplify: Light-weight Linked Data Publication from Relational Databases [C]. In: Proceedings of the 18th International Conference on World Wide Web. New York: ACM, 2009: 621-630.
- [14] RDBToOnto [EB/OL]. [2014-04-02]. <http://sourceforge.net/projects/rdbtoonto/>.
- [15] Datalift [EB/OL]. [2014-04-02]. <http://datalift.org/>.
- [16] Rodríguez J B, Gómez-Pérez A. Upgrading Relational Legacy Data to the Semantic Web [C]. In: Proceedings of the 15th International Conference on World Wide Web. 2006: 1069-1070.
- [17] Neto L E T, Vidal V M P, Casanova M A, et al. R2RML by Assertion: A Semi-Automatic Tool for Generating Customized R2RML Mappings [C]. In: Proceedings of ESWC 2013 Satellite Events, Montpellier, France. Springer Berlin Heidelberg, 2013: 248-252.
- [18] De Laborda C P, Conrad S. Relational. OWL: A Data and Schema Representation Format Based on OWL [C]. In: Proceedings of the 2nd Asia-Pacific Conference on Conceptual Modelling. 2005: 89-96.
- [19] Barrasa J, Corcho O, Gómez-Pérez A. Fund Finder: A Case Study of Database-to-Ontology Mapping [C]. In: Proceedings of the Semantic Integration Workshop. 2003.
- [20] Hert M, Reif G, Gall H C. Updating Relational Data via SPARQL/update [C]. In: Proceedings of the 2010 EDBT/ICDT Workshops. ACM, 2010.
- [21] Byrne K. Having Triplets-Holding Cultural Data as RDF [C]. In: Proceedings of the ECDL 2008 Workshop on Information Access to Cultural Heritage, Heritage, Aarhus, Denmark. 2008.
- [22] Sahoo S S, Bodenreider O, Rutter J L, et al. An Ontology-Driven Semantic Mashup of Gene and Biological Pathway Information: Application to the Domain of Nicotine Dependence [J]. Journal of Biomedical Informatics, 2008, 41(5): 752-765.
- [23] Green J, Hart G, Dolbear C, et al. Creating a Semantic Integration System Using Spatial Data [C]. In: Proceedings of the 7th International Semantic Web Conference on Posters & Demonstration Session. 2008.

## 作者贡献声明：

刘振：调查、整理及分析文献，起草论文；

张智雄：提出研究思路及论文框架，最终版本修订。

(通讯作者：刘振 E-mail: liuzhen@mail.las.ac.cn)

## Survey of Technical Methods and Tools of RDB-to-RDF

Liu Zhen<sup>1,2,3</sup> Zhang Zhixiong<sup>2</sup><sup>1</sup>(School of Management, Xuzhou Institute of Technology, Xuzhou 221008, China)<sup>2</sup>(National Science Library, Chinese Academy of Sciences, Beijing 100190, China)<sup>3</sup>(University of Chinese Academy of Sciences, Beijing 100049, China)

**Abstract:** [Objective] Summarize the researches on technical methods and tools of RDB-to-RDF and extract the key technologies. [Coverage] Retrieve English and Chinese literatures related to RDB-to-RDF from database of Elsevier, Springer, CNKI. [Methods] Use literature investigation, summarized by research topic. [Results] Summarize and analyze from the perspective of mapping ideas, techniques and implementation methods, comparing important feature and application occasions of mapping tools, and enumerating the typical application. [Limitations] Lack of specific quantitative evaluation when comparing mapping tools. [Conclusions] This study is helpful to understand the related key techniques, tools and main application scenarios in RDB-to-RDF process.

**Keywords:** RDB RDF RDB-to-RDF

## SirsiDynix 与 Total BooX 就电子资源中心达成合作协议

Total BooX 和 SirsiDynix 于近日达成合作，为电子资源中心(eRC)又添一位新的内容供应商。Total BooX 是一家创新企业，提供来自多家出版商(包括 Workman、Sourcebooks、O'Reilly Media 等)的成千上万种高质量电子书。此次电子资源中心的合作将使得用户只需在其图书馆目录中轻轻一点，便可以检索和访问 Total BooX 的所有资源。

“我们非常高兴可以把 Total BooX 的资源整合到我们的电子资源中心，” SirsiDynix 公司 CEO Bill Davison 说：“我们希望读者可以很容易地获取他们需要的所有资源，Total BooX 有着和我们一样的使命。此次合作中，我们将共同努力，减少电子书下载过程中的障碍，使得读者在一个地方就能找到他们需要的所有图书，如此，图书馆不需要增加一个第三方网站，还能免去人工数据输入，同时，读者还能方便访问 Total BooX 的资源。”

Total BooX 允许读者即时下载图书，为他们提供无限制的访问，同时只收取读者所阅读页面的费用。这一点非常符合 SirsiDynix 公司的愿景，即：避免读者使用时走进“死胡同”(包括排队和预约)，同时，帮助图书馆轻松、经济地管理他们的电子资源。

“我们非常荣幸可以和 SirsiDynix 这么高水准的 ILS 供应商合作，此次合作中，我们看到了巨大的潜力，” Total BooX 创始人兼 CEO Yoav Lorch 指出，“这将不仅使得用户可以发现并即刻享有内容，还支持所有的 eRC 图书馆在不预先收取图书费的情况下为其用户提供更多的电子书。”

(编译自: <http://www.sirsidynix.com/press/sirsidynix-to-partner-total-booX-with-eresource-central>)

(本刊讯)