论 文

提升生物信息学知识库可重用性的经验教训：Bgee 的实践

Tarcisio Mendes de Farias[1,2,*], Julien Wollbrett[1,2], Marc Robinson-Rechavi[1,2]和 FredericBastian[1,2]

[1]瑞士生物信息学研究所和 [2]洛桑大学生态与进化系

*tarcisio.mendes@sib.swiss

摘要

背景，提升生物信息学知识库的互操作性，是最大限度提高数据可重用性、进而提升其效用（例如提高生物医学研究的投资回报）的首要要求。知识库可以为生命科学家和其他知识库提供有用信息，但只有在知识库得到（重新）使用后，才能产生交换价值；缺乏互操作性，其效用便无从发挥。**结果**，本文讨论了若干提升互操作性的方法，具体取决于需要实现互操作的部分。研究结论以多个真实场景案例为依据，其中大多数由成熟的基因表达数据库 Bgee 实现。为进一步证明这些发现具有可迁移性，我们还针对 Bgee 的每项互操作性实践，列举了主要生物信息学知识库中的类似实现。此外，我们总结了十条普遍适用的主要经验。这些经验可用于任何生物信息学知识库，以促进数据可重用性。**结论**，本文通过聚焦互操作性，提出了切实可行的方法和可迁移的技能，以促进生物信息学知识库的可重用性。

关键词：互操作 性；数据库；数据可重用性；本体；FAIR 原 则

1 引言

生物信息学知识库（KB）通常是为满足特定兴趣社群的需求而构建的。通过提供软件工具、方法、服务和数据，这些知识库旨在为用户的工作（例如科学研究）提供便利和所需手段。因此，可重用性是任何生物信息学知识库都应考虑的重要方面。可重用性是指某种资源能够被不同主体多次使用的能力。这一表述是对 [1] 中数据可重用性定义的概括。

在应用计算领域，自 20 世纪末以来，计算机程序对数据可重用性的需求就已受到关注 [1]。近来，自 2016 [2] 年起，可发现、可访问、可互操作和可重用（FAIR）原则倡导通过机器和人类更高效地发现并重用数字研究资源。根据 Jacobsen 等人 [3] 的说法，可发现性、可访问性和互操作性共同促成最终目标：可信、有效且持续地重用研究资源。Mons 等人在 [4] 中也强调了这一目标，他们表明 FAIR 原则侧重于确保研究对象可重用。因此，将可重用性作为最终目标，凸显了 FAIR 原则内在相关，且理想情况下不应彼此割裂或被视为独立原则。将 FAIR 指导原则的结构和抽象概括为四个主要且彼此不同的原则（即 F.A.I.R.），一直是研究人员和从业者试图使其数据和元数据“符合 FAIR 原则”时产生困惑和误解的根源。这一点在原始作者需要于后续出版物中解释其对 FAIR 原则的初衷和解读时显而易见 [3, 4]；其他研究人员尝试解读这些原则时也是如此 [5, 6]。作者在 [7] 中也承认了这一问题。例如，在 [8] 中，作者指出，这些原则在不同研究领域中的操作化尚未达成共识。尽管存在这些误解，但不可否认，FAIR 原则极大地提高了不同利益相关者（例如政治人物、研究资助机构和研究人员，见 [9]）对让不同主体（即人类和机器）能够重用研究成果这一重要性的认识。

编译日期： 六月 21, 2023。作者准备的手稿
草稿。

本文从 [10] 报告的若干相似且互补的互操作性定义中，采用以下 IEEE 标准定义：“两个或多个系统或元素交换信息，并使用已交换信息的能力”[11]。基于此定义，并将生物信息学知识库视为系统，我们将介绍如何通过增强互操作性来提高可重用性。需要指出的是，本文并不打算讨论 FAIR 原则的所有方面，尽管我们的目标是遵循 FAIR 原则并在此基础上进一步实现最佳可重用性。我们着重讨论互操作性，一个原因是缺乏互操作性会导致资源难以甚至无法被发现和访问。例如，为数字资源分配持久标识符或描述性元数据，分别对应 F1 和 F2 原则 [2]，只有具备互操作性才有用。这是因为要解析持久标识符，就需要与其他系统（例如数字对象标识符（DOI）解析器）互操作。这也是为什么在谈到 DOI 时，我们强调它是符合 ISO 26324:2012 标准 [12] 的持久、可互操作的 标识符。因此，“互操作性”一词（包括其变体和定义）在 DOI手册 [13] 中出现超过 50 次，并非偶然。同样，所生成的描述性元数据只有在能够被使用、也就是具备互操作性的情况下，才会对搜索引擎等发挥作用。

此外，已有多位研究人员认识到，实现互操作性是一项复杂的任务 [14, 15, 16, 17]。由于这项任务难度很大，它也阻碍了彼此独立的生物信息学知识库之间的信息交换。因此，为缓解这一问题，本文聚焦于提升生物信息学知识库互操作性的务实方法。我们主要结合将 Bgee 开发为更具可复用性的知识库这一经历，阐述这些方法。

Bgee 是一个成熟的知识库，可用于检索和比较多种动物物种的基因表达模式 [18]。它整合并协调了多个基于不同技术的数据源，包括单细胞RNA测序（scRNA-Seq）、批量RNA测序、Affymetrix、原位杂交和表达序列标签（EST）。它仅基于经过整理的健康野生型表达数据（例如，不涉及基因敲除、治疗或疾病），以提供可供比较的正常基因表达参考。此外，多位研究人员已认可 Bgee 互操作性实践的实用性，例如 [19]。

尽管本文聚焦于 Bgee 的应用案例，但此处讨论的实践、方法和经验教训也适用于其他知识库，例如《核酸研究》分子生物学数据库合集 [20] 中报道的知识库。为进一步说明这些经验具有可迁移性，我们还将针对 Bgee 的每项互操作性实践，在适用时介绍广泛使用的生物信息学知识库所采用的类似实现，例如 GeneCards（以人类基因为中心的知识库）[21]、UniProtKB（以蛋白质为中心的知识库）[22] 和 Orthologous MAtrix（OMA，一种直系同源关系资源）[23]。

此外，这些经验教训也可部分应用于其他领域，例如 [8] 中提到的低碳能源数据库（DB）。该文指出，能源数据库之间存在显著异质性，采用通用的数据交换格式和语义表示有助于提升这些数据库之间的互操作性。

2 提升互操作性的广泛层面

增强数据/元数据互操作性是解决不同部分（例如系统）之间数据/元数据异质性的办法，而这些部分正是我们希望在其间交换信息的对象 [24, 25]。根据 [26]，我们可以将这些异质性分为结构异质性（模式）、语法异质性（格式）和语义异质性（含义）。正如Alon Y. Halevy在 [27]中所指出的，只要对一组数据的组织方式不止一种，就会出现语义异质性。

为了在不同系统之间正确交换信息，我们必须解决信息生产者与消费者之间可能存在的语法和语义异质性。所谓“正确”，是指消费者感知到的信息与生产者的意图完全一致；反过来，生产者构思或编写的信息也应与消费者的预期完全相符。这样，消费者或生产者便无需依靠猜测、启发式方法或机器学习来正确处理交换的信息。为说明这一点，我们来看看一种用于数据交换的半结构化格式，例如逗号分隔值（CSV）文件（即一种表格数据格式）。2005 技术标准 RFC 4180 [28] 对 CSV 文件格式作出了规范，但定义 CSV 文件的语法方式仍有多种。例如，标题行通常位于文件首行，但它是可选的，而且没有明确的方法来判断它是否存在，因此消费者导入数据时必须格外谨慎。因此，生产者和消费者必须就如何解决这种语法异质性达成一致，才能实现互操作性，Google Ads 系统 [29] 和美国国家生物技术信息中心（NCBI）的 LinkOut 服务 [30]便是如此。

灵活性越高，异质性也越强。选择更灵活的信息交换解决方案，往往意味着需要解决更多异质性 [27]。尽管高度受约束、形式化且准确的互操作性解决方案能够显著降低异质性，但由于实施难度、适应性以及对所交换信息的适用性等问题，其采用可能受到影响。例如，将某个模型（如 Bgee 原生数据模型）中的数据描述到另一个模型（如维基数据 [31] 数据模型）中，可能会因语义异质性而导致数据丢失（即部分互操作性）。只要不同领域的专家采用不同的建模实践和约束（如应用范围、实时能力、安全性），就可能构建不同的概念模型来表示同一批数据。实际上，即便本体被定义为“共享概念化的形式化、明确规范”[32]，不同的本体工程师也可能针对同一知识领域构建不同的本体。例如，Ontology Lookup Service [33] 中有十多个本体报告了不同的 基因 概念。为解决后一问题，本体匹配和本体对齐等方法已被认为是值得关注的途径 [34, 35]。通常，不同可互操作元素之间的数据、元数据和数据模式映射有助于实现匹配／对齐。

然而，即便是为了定义映射和对齐，也会选择并应用一种具有特定词汇表、语法和语义的语言。例如，我们可以使用普通英语（意味着内容无法供机器直接读取）、编程语言（特定用途的适配器/转换器）、OWL——Web 本体语言（例如，owl:sameAs、owl:equivalentClass）[36]、SKOS——简单知识组织系统词汇表（例如，skos:closeMatch）[37]、SWRL——语义网规则语言（例如，swrlb:matches、swrl:Imp）[24]、VoIDext——互联数据集扩展词汇表（例如，voidext:resourceMapping）[38] 等来定义对齐。因此，异质性问题及由此产生的互操作性问题依然存在，只是发生在另一个层面。在选择互操作性解决方案时，通常需要考虑数据模型、用于表示元数据和数据的语言及标准；与定义映射和对齐一样，我们必须考虑需要解决的不同异质性程度。此外，根据异质性的类型、我们希望实现互操作的元素性质以及期望达到的互操作性层级，某种语言可能比其他语言更适合声明映射。例如，在进行本体匹配时，OWL 语言的表达能力不足以定义复杂的数据模式对齐。比如，假设存在 genus、species 和 scientific name 这三个属性，那么 genus（例如 Homo）与 species（例如 sapiens）的串联即可推出该物种的学名（例如 Homo Sapiens）。那么，如果我们假设存在三个属——

属性：属、物种和学名，那么属（例如 Homo）与物种（例如 sapiens）的串联即可推出该物种的学名（例如 Homo sapiens）。因此，要定义这些复杂映射，SWRL 等其他语言更为合适。SWRL 和 OWL 都是基于逻辑的形式化语言。将二者结合起来定义复杂映射，还能借助支持这些语言的推理引擎自动推导出新的对齐方式 [24]。在某些情境下，为了实现不同精度层级的互操作性，定义映射的性质可能至关重要，如 SKOS 词汇表中所述：`skos:closeMatch`、`skos:exactMatch`、`skos:broadMatch`、`skos:narrowMatch` 和 `skos:relatedMatch`。

根据我们希望实现的互操作性目标，可以采用语义放宽方法。语义放宽是指为了实现互操作性而忽略语义和数据异质性的能力 [38]。例如，在不同的直系同源数据库（即包含不同物种对应基因信息的数据库）之间实现互操作时，可以互换使用基因和蛋白质的概念。这是因为有些算法会利用蛋白质序列推断直系同源基因。因此，只要能够接受一定的信息或精度损失，我们就可以提高互操作性。

3 知识库互操作方法与实践

我们将知识库的互操作方法分为以下三种类型：

定义 1。单边互操作性：一方必须严格遵循另一方的流程才能实现互操作。可互操作的各方之间几乎没有协商空间，甚至完全无法协商。如果另一方的互操作流程基于独立的互操作流程，则将其归类为多边互操作性，定义见定义 3。

定义 2。双边互操作性：双方必须相互协调，即达成共同协议以实现互操作。

定义 3。多边互操作性：希望实现互操作的两个或多个方面遵循独立的互操作性流程。可根据请求改进或更改此流程，而维护该流程的第三方组织或社区可以接受，也可以不接受这些改进或更改。此流程通常由 *Schema.org* 等互操作性标准组成。当且仅当双方的协调基于可供他人重复使用的独立互操作解决方案时，双边互操作性才被视为多边互操作性。

根据这些定义对互操作方法进行分类，很大程度上取决于具体情境和时间线。例如，单边或双边互操作性可以演变为多边互操作性，前提是它成为一项可供他人重复使用的标准，或成为此类标准的一部分。此外，如果双方希望改进并增加信息交换，而现有的单边方法不支持这一点，那么单边互操作性也可能转变为双边互操作性。另一种可能的情形是采用混合方法，即实施不止一种互操作方法。例如，多边互操作性可能不足以或无法及时实现知识库之间所需的互操作程度。在这种情况下，双边互操作方法可以作为多边方法的补充。

表 1 展示了定义 1、定义 2 和定义 3中所述的互操作方法。此外，该表总结了 Bgee 作为数据生产者所涉及的用例，这些用例将在下一节中详细介绍。

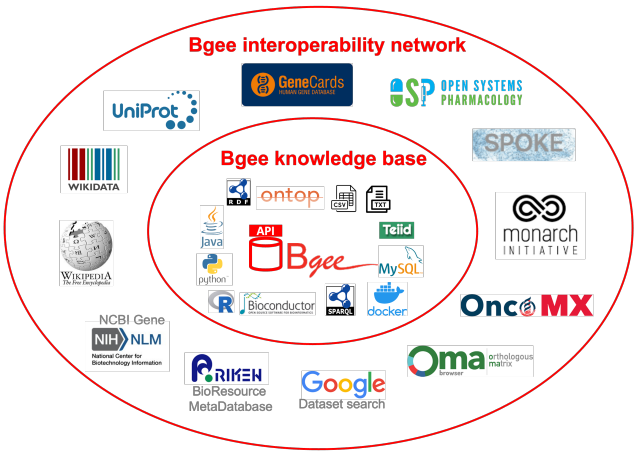


图 1。本文提及的 Bgee 互操作网络增强方案的简化示意图。内圈中的元素是用于实现 Bgee 与各类数据库和系统互操作的一系列技术示例；外圈则以相应的徽标展示这些数据库和系统。

4 增强互操作性：Bgee 与其他知识库互操作的经验

Bgee 知识库通过协调来自异构数据源的数据并采用数据仓库方法，对这些数据进行整合与汇总，最终形成一个大型关系数据库（撰写本文时为 8 太字节）。数据整理和质量控制是 Bgee 使命的核心。此外，Bgee 获得了公有领域数据库许可证，因此成为提升互操作性的有趣案例研究，因为重用其数据不受任何所有权限制。图 1展示了 Bgee 互操作网络的简化视图以及其中涉及的技术；图 2 概述了已实现的 Bgee 数据互操作架构，下一小节将进一步详述。最后，对于 Bgee 的每项实践，我们都会提及其他知识库（即 UniProtKB、GeneCards 或 OMA）是否有类似的实现；在适用的情况下，还会说明它们如何也能从我们的经验中获益。例如，如果它们尚未与感兴趣的目标知识库实现互操作。

4.1 基于文件的数据交换

自计算机文件系统问世以来，人们便一直使用计算机文件交换数据。这种方法在生物信息学知识库之间交换和重用数据的优势包括易于部署且能够实现自主性。例如，数据生产者可以指定数据格式，而数据使用者无需事先与其达成协议即可使用该格式。在这个例子中，使用者必须调整自己的工具（例如，实现文件读取器），才能与数据生产者进行互操作。同样，数据使用者也可以指定数据生产者应采用的数据交换格式。然而，这种互操作模式往往会导致误解，主要原因是缺乏数据互操作标准和数据结构（即数据未结构化或仅半结构化）；此外，使用者与生产者之间存在语法和语义异质性，也使互操作更加复杂。而且，由于静态文件的导出和导入数据操作彼此异步且独立，这种方式不一定能提供对最新数据的访问。

目前，Bgee 通过基于文件的数据交换与以下知识库实现互操作：NCBI Gene 数据库、GeneCards、UniProtKB、RIKEN 元数据库和 OncoMX。此外，Bgee 还通过采用关系型和图数据模型的两种数据转储，为高级用户提供高度结构化的数据。在接下来的小节中，我们将讨论 Bgee 基于文件的每种互操作案例，以及我们如何缓解前述问题，例如误解。

目标 知识-库（知识库）	互操作方法	描述
NCBI Gene［39］	单向	Bgee 必须遵循 NCBI LinkOut 系统的交换文件格式，即 CSV 或 XML 文件。
UniProtKB［22］	单向	Bgee 必须遵循 UniProtKB 交换文件格式，即一种基于其自有格式的文本文件。
GeneCards［21］	双向	Bgee 和 GeneCards 从头定义了一种类 TSV 交换文件格式，便于快速、轻松地由 GeneCards 使用，并由 Bgee 生成。
OncoMX［40］	双边	起初，Bgee 和 OncoMX 从头定义了一种类 TSV 交换文件格式，使其易于快速被 OncoMX 使用并由 Bgee 生成。
RIKEN 元数据库［41］	单向	RIKEN 元数据库将可下载的 Bgee RDF 转储文件直接导入其三元组存储，作为命名图。
Monarch 计划倡议［42］	单向	Monarch Initative 项目直接使用现有的 Bgee下载文件。
SPOKE［43］	单侧	Bgee下载文件未经修改，直接用于构建精准医学知识图谱。
Open 系统药理学［44］	单向	Bgee下载文件被直接用于构建药物开发知识库。
维基数据［31］	单向	Bgee 开发了一个机器人，使用维基数据 Python API 自动提取、转换和加载其数据导入维基数据［45］。
维基百科［46］	单向	Bgee 在现有的基因信息框模块中实现并集成了一个软件组件［47］。这样一来，维基百科便可从维基数据中动态获取 Bgee 数据。
Google 数据集搜索［48］	多方	Bgee 在其网页中嵌入了基于 Schema.org 的元数据。这些元数据会自动——被支持 Schema.org 的其他系统（例如 Google 数据集搜索）检索和使用工具。
OncoMX 联邦tion	单侧	OncoMX 直接使用现有的 Bgee MySQL 数据库（称为 EasyBgee），以实现单侧互操作能力），从而联合两个知识库。为此，联邦数据模式由 OncoMX 原生关系数据模式以及为 EasyBgee 数据模式定义的视图和映射组成。

表 1。此 该表总结了 Bgee 作为数据生产者所涉及的用例，以及已实施的互操作方法。

Bgee 在美国国家生物技术信息中心（***NCBI***）的 ***NCBI Gene*** 数据库中

NCBI Gene 提供以基因为中心的信息，例如序列、表达、结构、功能、引文和同源性数据。为了通过 NCBI LinkOut 服务［49］与 NCBI Gene 实现互操作，我们必须严格遵循 NCBI 自身的规范，编写所需的数据交换文件，文件可以是基于 XML 的文件，也可以是基于 CSV 的文件。因此，文件读取器部署到文件传输位置后，便可自动读取所提供的数据。LinkOut 系统已成功帮助包括 Bgee、GeneCards 和 OMA 在内的 30 多个数据提供者［50］，将其资源链接到不同的 NCBI 数据库，例如 NCBI Gene。我们提供的数据包括基因符号，以及通过 NCBI LinkOut 部分链接到与某个 NCBI 基因页面相对应的 Bgee 基因页面的链接。尽管针对 CSV 和 XML 文件定义都提供了详尽的文档［51］，但由于缺少更完善的语义表示，数据生产者生成的文件可能不符合 NCBI LinkOut 文件读取器的要求。此外，尽管创建 LinkOut XML 文件时有文档类型定义（DTD）［52］，但它无法像 XML 架构定义（XSD）［53］那样充分约束 XML 结构。例如，使用 DTD 无法定义数据类型。LinkOut XML 文件定义中相关 XML 元素的数据类型（例如 *<LinkId>* 数据类型）并未明确，因此我们不知道 *<LinkId>* 可以是任意字符，还是只能为大于零的整数。即使文档中明确规定了数据类型或完整的数据模式，该模式也无法被机器读取，因此会使原本可以自动化的任务变得复杂，例如编写数据交换文件。

为了与 NCBI LinkOut 系统互操作，在 NCBI Gene 基因页面上发布 Bgee 基因链接，我们决定遵循其 CSV 格式。图 3 展示了生成的 CSV 文件的一部分。尽管 CSV 文件的结构不如 XML 文件丰富，但这一决定主要是因为我们可以通过对 Bgee 关系数据库执行一条结构化查询语言（SQL）查询，轻松生成 Bgee 表格数据输出。此外，在该查询中，我们还投影了 NCBI LinkOut 工具所要求的 CSV 标题行。不过，这种文件格式带来的灵活性代价高昂，那就是，缺乏语义和数据结构会造成异质性。首先，在语法层面，LinkOut 工具所要求的 CSV 文件并未完全遵循标准 RFC 4180中记录的大多数实现方式。该标准对 CSV 格式进行了规范，特别指出“每个字段可以用双引号括起来，也可以不括起来”。然而，在 LinkOut CSV 文件中，我们不能用引号括起字段，否则会导致文件无效。此外，LinkOut 工具还规定，同一个数据提供方针对一个 NCBI Gene 条目最多只能提供三条记录。其次，在语义层面，图 3 中所示的必需第三个字段可以有多种解释：它既可以是唯一标识符（UID），也可以是基于自定义语法的查询。该字段至关重要，因为它能够将 Bgee 数据与 NCBI Gene 数据进行交集匹配，换言之，确保 Bgee 链接和基因符号能够正确发布在 NCBI Gene 页面上。对于我们的应用场景，UID 指的是 NCBI Gene 标识符。Bgee 关系数据库中没有 NCBI Gene 标识符，因此，我们最初尝试将类似 NCBI 的查询而非 UID 定义为第三个 CSV 字段，并提供 Bgee 与 NCBI Gene 之间的共同 ID，例如 Ensembl ID。然而，这可能会导致 Bgee 与 NCBI 基因条目之间出现不一致，例如某个 Ensembl ID 在 NCBI 中不存在，或无法检索到相同的基因条目。后一种情况可能是因为查询会匹配 NCBI Gene 页面上任意一个已索引词的关键词。LinkOut 服务要求 Bgee 团队仅考虑 NCBI Gene 中存在的 Ensembl ID，这意味着如果某个查询（即 Ensembl ID 关键词）未检索到任何结果，LinkOut 工具可能无法正常工作。因此，为了避免歧义并解决这种语义异质性，我们决定使用 NCBI Gene ID，并将其纳入 Bgee 数据库。我们得以实现这一点，得益于导入数据库的 NCBI Gene ID 映射文件［54］。值得注意的是，LinkOut 文档既未提及这些问题及其解决方案，也未在任何相关数据模式中对其作出正式定义。我们还想强调，由于 CSV 是一种半结构化数据格式，因此不存在明确、完整且形式化的数据模式。因此，为了解决这些问题，我们不得不通过电子邮件直接联系 LinkOut 服务提供方，以准确确认 LinkOut 系统所要求的数据交换格式。这一过程耗时较长；值得一提的是，此案例中 NCBI 的响应非常及时，但相较之下

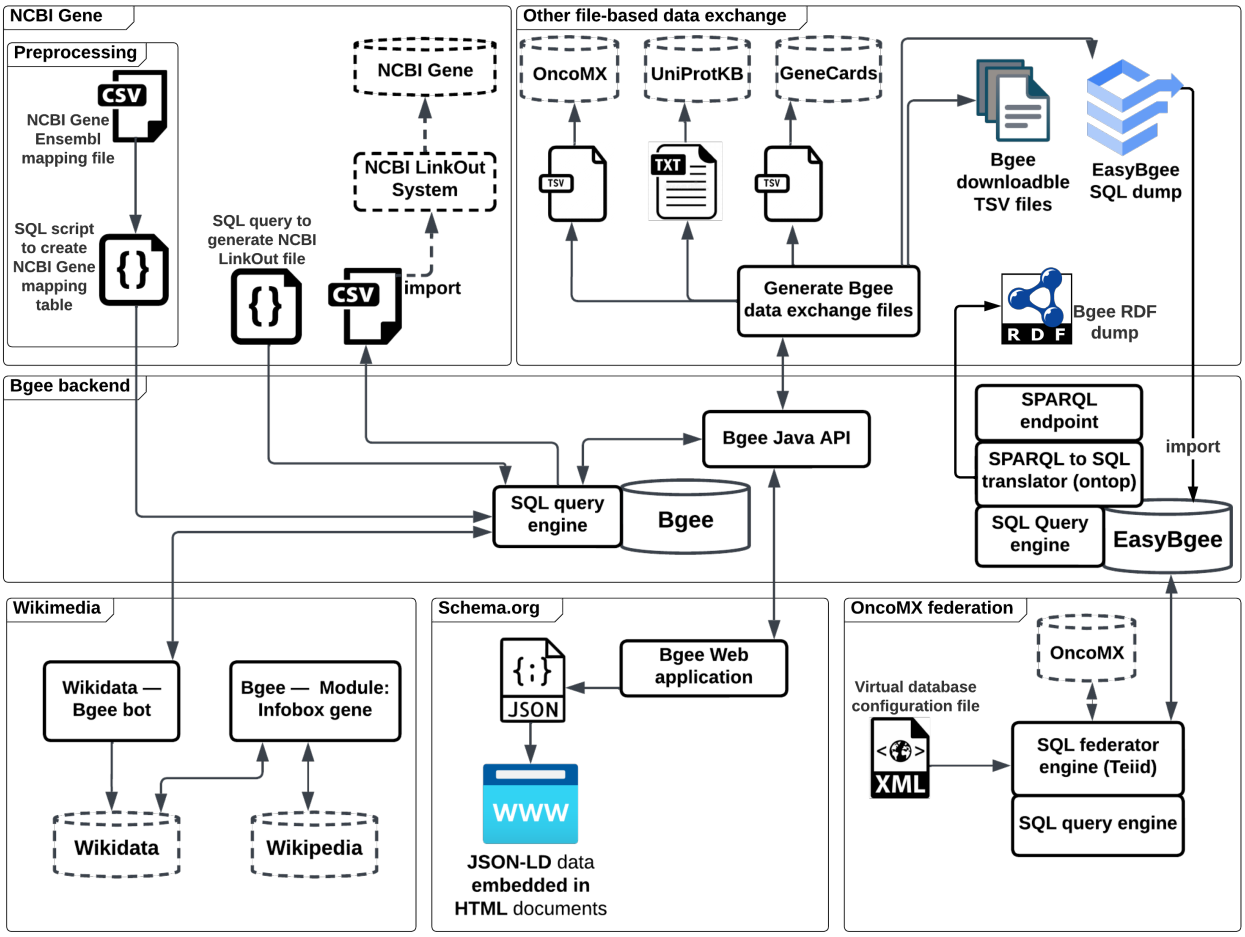


图 2。Bgee 与其他主要知识库实现互操作的架构示意图。虚线圆柱体表示外部知识库（KB）。所有矩形均为 Bgee 实现的软件组件，虚线矩形除外，它表示第三方系统组件。通常，从“SQL 查询引擎”和“Bgee Java API”发出的箭头表示输出数据（例如检索结果），指向它们的箭头则分别表示接收的 SQL 语句或 API 调用。所有从“生成 Bgee 数据交换文件”发出的箭头均表示文件导出。“NCBI Gene”容器展示了为与 NCBI Gene 知识库实现互操作而开发和使用的组件。“其他基于文件的数据交换”容器汇总了所有用于生成 Bgee 交换文件的软件组件，这些文件专用于其他知识库；此外还包括包含 Bgee 数据视图的非知识库专用制表符分隔值文件、Bgee 数据的 MySQL 数据库转储以及资源描述框架转储。为简化示意图，并未展示所有通过文件与 Bgee 互操作的知识库；其中包括导入生成的 Bgee RDF 转储的 RIKEN 元数据库，以及直接使用可下载 Bgee TSV 文件的 SPOKE、Open Systems Pharmacology 和 Monarch Initiative 项目。“Bgee 后端”容器展示了 Bgee 知识库的数据存储层和数据访问方式（即 Bgee Java API、SPARQL 和 SQL 查询语言）。“维基媒体”容器描绘了为与维基数据和维基百科交换信息而实现的软件组件。在“维基媒体”容器中，指向维基数据的机器人箭头表示从 Bgee 向维基数据插入数据；指向“模块：基因信息框”以及从该模块发出的箭头则分别表示查询和检索数据。“Schema.org”容器展示了嵌入 Bgee 网页、采用 Schema.org 词汇表的元数据；支持该词汇表的系统（例如 Google 数据集搜索）会使用这些元数据。最后，“OncoMX 联邦”容器描绘了两个独立知识库（即 Bgee 和 OncoMX）之间的动态互操作方法；这种方法可以取代基于文件的方式（即通过静态交换文件实现的互操作），如“其他基于文件的数据交换”容器所示。

```
1  PrId,DB,UID,URL,IconUrl,UrlName,SubjectType,Attribute
2  10418,Gene,103476274,https://bgee.org/bgee15_0/gene/ENSPREG00000013759,,rbm41 gene expression,,
3  10418,Gene,103465305,https://bgee.org/bgee15_0/gene/ENSPREG00000013760,,kdm5ba gene expression,,
4  10418,Gene,103474996,https://bgee.org/bgee15_0/gene/ENSPREG00000013761,,BACH1 gene expression,,
```

图 3. NCBI LinkOut 系统使用的数据交换文件的一部分。

比现有文档更可靠。

最后，通过这个用例我们可以看到，尽管交换的信息很简单（3），但要实现 Bgee 与 NCBI Gene 等独立资源之间的互操作并非易事。如果交换的信息更复杂，例如包含 LinkOut 系统并未预期的基因表达水平、解剖结构和发育阶段，那么上述问题会严重得多。然而，这些相关信息可以丰富现有 NCBI Gene 基因页面中的某些“表达”部分，并使目前尚无任何表达信息的 NCBI 基因页面也能纳入该部分，例如黑猩猩血红蛋白亚基β基因页面 [55]。

其他知识库的经验。GeneCards 和 OMA 采用单边互操作方法，开发了与 NCBI LinkOut 系统互操作的类似流程。此外，UniProtKB 并未与 LinkOut 系统交换信息。不过，如 [56] 所述，通过采用双边互操作方法，UniProtKB 登录号已整合并纳入 NCBI Gene 知识库。例如，“人类 HBB” NCBI Gene 页面在“NCBI 参考序列（RefSeq）”和“相关序列”部分引用了 UniProtKB 登录号 [57]。

Bgee 在 *UniProt* 知识库中

为了与 UniProt 互操作，我们必须采用其单向互操作方法，该方法依赖特定语法和文本文件格式。

这种文件格式可以理解为类似 CSV 的文件，分隔符是分号后跟一个空格（; ），且没有标题行。不过，第一个单元格包含按特定语法定义的值，即“[UniProt 标识符] [内部

代码] [外部资源名称]”，其中空白字符实际上是三个空格，字符集编码为 us-ascii，内部代码由两个字母组成。下面展示了 UniProt 信息交换格式中的一行（条目）示例。在此示例中，使用的内部代码是“DR”，这是 UniProt 用于交叉引用的双字母代码。

```
A0A3B1E4W9   DR   Bgee; WBGene00304181; Expressed
     in pharyngeal muscle cell (C elegans) and 1
     other tissue.
```

主要得益于第一列和第二列中的标识符，我们能够实现 Bgee 与 UniProt 之间的互操作。因此，对于每个 UniProt 蛋白质条目，都会分配一个对应的 Bgee 基因条目，并附上指向 Bgee 基因页面的链接。该链接由 Bgee 基因页面网址 `https://bgee.org/gene/`与 Bgee 相关标识符（例如 `WBGene00304181`）拼接而成。此外，第三列包含我们为每个 Bgee 条目定义的描述。该描述由该基因表达量最高的单元格或组织，以及 Bgee 中该基因具有表达数据的组织数量组成。

为了生成 UniProt 信息交换文件，我们开发了一个文件写入器，它是 Bgee 软件和工作流的一部分。每个 Bgee 版本都会生成一个新文件。为确保 UniProt 能够获取最新的 Bgee 数据，我们提供了一个永久 URL [58]。

其他知识库的经验。 OMA 和 GeneCards 也通过采用与 Bgee 相同的方法与 UniProtKB 实现互操作，但与 Bgee 不同的是，它们没有使用描述字段提供额外信息。

GeneCards 知识库中的 *Bgee*

GeneCards 是一个知识库，会自动整合来自约 150 个网络来源、以人类基因为中心的数据，包括基因组、转录组、蛋白质组、遗传、临床和功能信息。与 NCBI Gene 不同，GeneCards 不会提供信息交换指南或特定的文件格式。由于缺少预定义的互操作数据格式，我们得以自由定义格式，并更灵活地决定交换哪些信息。在联系 GeneCards 之前，我们首先起草了一个制表符分隔值（TSV）文件，其中包含 Bgee 数据库的基本信息，例如可用于匹配 Bgee 与 GeneCards 的值、Bgee 基因页面链接，以及每个基因的简短表达摘要。此外，为了提升互操作性，我们复用了 Bgee 与其他知识库之间现有的数据交换工作流。例如，为 GeneCards 生成的 TSV 文件包含了与和 UniProtKB 交换的文件类似的信息。我们采取的策略是，先提供一个简单的文件，其中仅包含有关 Bgee 条目的最少信息，以便 GeneCards 轻松理解并纳入其基因表达部分。通过这一策略，我们希望促进讨论，并说服对方复用我们的数据。为了进一步争取对方认同，我们还表明了自身的投入和意愿：在 Bgee 网站上发布指向对方相应基因页面的链接（或在对方感兴趣时发布其他数据）。随后，我们联系了 GeneCards（即潜在的数据使用者），并向其介绍我们的互操作性解决方案。由于该方案简单易用且具有诸多优势，GeneCards 的维护人员很快便表示同意，同时提出了一些修改建议。之后，他们实现了一个用于读取该 Bgee TSV 文件的读取器，从而整合了来自 Bgee 的基因表达信息，如图 4 所示。作为一项良好实践，我们还商定提供一个持久链接，指向包含最新 Bgee 数据的 TSV 文件 [59]。因此，GeneCards 的每个新发布版本都能获取最新的 Bgee 数据。

尽管我们必须从头定义 Bgee 与 GeneCards 之间的信息交换方式，但要让这两个知识库实现互操作，并不需要投入太多精力。例如，我们只需提出一个互操作性协议方案，总共交换五封电子邮件即可。

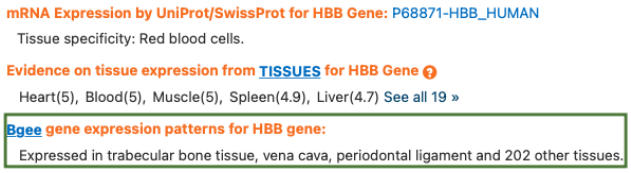


图 4。 GeneCards“表达”标签页中的 Bgee 链接及基因表达摘要示例。

所交换的信息极少且简单，这一点对于从头快速、轻松地实现互操作至关重要。预先建立数据交换格式（例如 NCBI LinkOut 系统），有助于推动知识库之间的互操作性，且每当有新资源加入互操作时，无需再实现新的数据文件读取器。因此，只要数据生产者遵循数据使用者的流程和要求，数据使用者便可轻松实现互操作。因此，互操作性的实现负担主要落在数据生产者一方，即单边互操作性。单边互操作性的主要缺点是缺乏添加新信息的灵活性。例如，使用 LinkOut 系统时，我们无法交换诸如基因表达摘要之类的描述；而与 GeneCards 合作时，我们建立了双边互操作性，因此可以交换此类信息。

其他知识库的经验。 GeneCards 中包含 UniProtKB 的信息，因为 GeneCards 使用 UniProtKB 的单边互操作方法（例如 Web API）[60]，单方面从 UniProtKB 提取数据。另一方面，GeneCards 中没有 OMA。不过，OMA 可以像 Bgee 一样建立双边互操作方法，并纳入每个 GeneCards 条目的“直系同源基因”和“旁系同源基因”部分。

OncoMX 知识库中的 *Bgee*

OncoMX 是一个知识库，它整合相关数据集，以支持癌症生物标志物研究 [40]。Bgee 通过双边互操作方法向 OncoMX 提供健康的基因表达数据。因此，Bgee 数据集可在 OncoMX Web 门户中访问 [61]。我们 Bgee 团队定义了制表符分隔值数据文件，其中包含人类和小鼠在每种实验条件（即 OncoMX 中涉及的解剖结构和发育阶段）下的基因表达存在与缺失判定，以及表达评分。此外，我们复用 UBERON 等本体 [62] 来表示解剖结构，以避免歧义并改善语义互操作性。这有助于与 OncoMX 整合的其他癌症生物标志物相关数据（例如差异表达数据集）进行整合。此外，在智能开放数据探索（IN-ODE）项目 [63] 的背景下，Bgee 与 OncoMX 之间的联邦式自动互操作工作已经开展。通过应用这一联邦式方法，我们解决了第 4 小节第一段中提到的大多数问题。1

其他知识库的经验。 在 OncoMX 中，每个生物标志物或基因条目也会分配 UniProtKB 登录号（即标识符）。这是 OncoMX 开发者在整合不同数据集并使用基因名称与 UniProtKB 标识符之间的映射时完成的。因此，尽管这种互操作性有限，UniProtKB 与 OncoMX 仍采用单边互操作方法实现了互操作。若同时查询 UniProtKB，以获取 OncoMX 所需的相关信息（例如与某个生物标志物相关的疾病），则可以进一步改进这一方法。此外，OncoMX 也可以获取 GeneCards 基因列表，并像处理 UniProtKB 那样引用 GeneCards 条目（即添加 GeneCards 交叉引用）。另一种做法是，UniProtKB 和 GeneCards 可以像 Bgee 一样，采用双边互操作方法构建可供 OncoMX 直接使用的数据集。这样提供的信息会比知识库之间简单的交叉引用更有助于 OncoMX 用户。最后，OMA

可以提供人鼠直系同源基因，将 Bgee 整合的人类和小鼠基因表达数据关联起来，并在 OncoMX 门户中直接为生物标志物研究人员提供更多见解。

Bgee数据库转储与 ***RIKEN*** 元数据库用例
包含主要处理后信息的 Bgee 数据转储，是整个 Bgee 关系数据库的简化版本。因此，我们通过提供明确且经过处理的基因表达信息，呈现一个简化视图，以排除集成原始数据所带来的复杂性。若没有这一视图，最终用户（包括第三方计算机工具）将难以理解和处理海量数据，也难以编写复杂查询来提取所需信息。这些数据转储包含高度结构化的数据，分别基于关系数据模型和资源描述框架（RDF）[64] 数据模型。关系数据库转储称为 EasyBgee，图 5 展示了其部分数据模式。我们定义了声明式映射，并将 Ontop 工具 [65] 应用于 EasyBgee 数据库，以生成 Bgee RDF转储 [65, 66]。因此，EasyBgee 数据也可表示为 RDF三元组模式，具体而言，是使用 Turtle（简洁RDF三元组语言）这一 RDF 的具体语法 [67]。

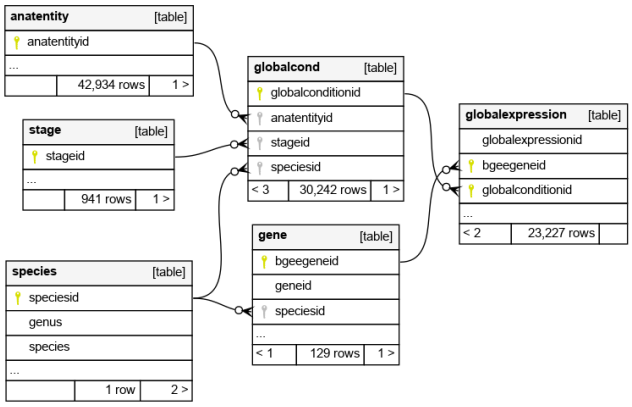


图 5. EasyBgee 关系数据模式的一部分。

我们同时为 EasyBgee 提供两种数据模型，这是一种良好实践，有助于触达更多用户、促进互操作性，进而提升 Bgee 数据的可复用性。例如，提供 RDF 转储使日本理化学研究所（RIKEN）能够将 Bgee知识图谱直接导入 RIKEN 生物资源元数据库，并将其作为命名图。RIKEN 元数据库整合了多个生命科学数据集，帮助研究人员充分利用 RIKEN 的研究成果。得益于这种互操作，RIKEN 现在可以复用 Bgee 基因表达数据，在研究人员搜索生物资源时为其提供支持，例如 [68] 中描述的阿尔茨海默病研究用例。

其他知识库的经验。与 Bgee 用例类似，RIKEN 元数据库会直接导入可用的 OMA RDF转储，其三元组数量少于 Bgee RDF 转储。GeneCards 则无法采用这种方式，因为它不提供任何 RDF 数据转储。此外，尽管 UniProtKB RDF转储规模庞大（>100 十亿个三元组），将其导入 RIKEN 元数据库可能难以实现，但 UniProtKB 提供了其 RDF 数据的视图，RIKEN 可以加以复用，例如人类疾病数据集。

Bgee下载文件：一种单边互操作方法
作为一种单边互操作方法，Bgee以按物种划分的制表符分隔值（TSV）文件形式提供数据视图。这些文件包含基因表达判定，标示表达的存在或缺失，以及经过处理的表达值；目前，在不同用例中，这些文件被用作与Bgee互操作的方法。例如，Monarch Initiative项目旨在将表型（如疾病）与基因型（如导致疾病的基因）关联起来 [42]，并使用 Bgee下载文件获取基因与其表达所在解剖实体之间的关联信息。随后，这些信息会显示在其网站每个基因页面条目的“解剖结构”部分。类似地，Bgee下载文件也用于构建一个名为SPOKE的系统所用的精准医学开放知识图谱 [43]：通过获取基因与其上调或下调表达所在解剖实体之间的关联，在图谱中生成新的边。在开放系统药理学套件（Open Systems Pharmacology suite）的框架下，这些下载文件还用于构建与药物开发相关的基因表达信息知识库 [44]；Bgee为包括人类在内的多个物种提供健康条件下正常基因表达的参考数据。

其他知识库的经验。作为一种单边互操作性方法，OMA、UniProtKB 和 GeneCards 分别提供下载文件，用于交换信息，包括其数据视图，在 [69]、[70] 和 [71]中。

4.2 编程接口

提供多种编程方式来互操作并使用知识库中包含的数据和信息，有助于提高其可重用性。这是因为根据用户的技能和用例，一种互操作方式可能比另一种更合适。为此，Bgee 提供了三种不同的编程接口，用于查询和操作其数据：SPARQL 端点、R 软件包和 Web API。对于后者，尽管 Web API 已可用 [72]，我们仍在编写文档，并致力于使其完全符合 OpenAPI [73] 规范和标准，以提高互操作性。

SPARQL 是一种用于基于资源描述框架的数据的结构化查询语言和协议。基于4.1小节中展示的 Bgee 资源描述框架数据转储，可通过 [74] 访问 SPARQL 端点。查询结果可采用多种格式获取，例如 JSON 或 CSV。也可以通过多种编程语言的应用程序编程接口（API）执行查询（例如通过 SPARQLWrapper [75]使用 Python）。此外，SPARQL 还支持执行联邦连接查询，以组合来自同样提供 SPARQL 1.1端点的多个知识库的数据，例如维基数据和 UniProtKB。[66]中充分展示了使用 Bgee 执行联邦查询的这一能力。

BgeeCall R 包可通过根据未表达的基因组特征（即基因间序列）估算背景转录噪声，使用户无需使用任意截断值（例如1 每百万转录本），即可生成基因表达存在/缺失判定。我们还提供 BgeeDB 包，用于从 Bgee 数据库下载注释和基因表达数据（即实现互操作），并进行 TopAnat 分析：这是一种 GO 类富集分析，可根据表达模式将解剖学术语映射到基因。两个包及其文档均可在 Bgee 网站[76] 和 Bioconductor [77]上获取。因此，我们也触达了有意重用基因表达相关数据的 Bioconductor 用户。为方便重用这些包，我们提供了 Docker 容器 [78]，这是一种轻量、独立且可执行的软件包，包含所有依赖项。此外，BioContainers [79]等其他软件包仓库和系统也与 Bioconductor 实现了互操作。因此，我们得以触达更广泛的 R 用户社区。例如，截至目前，BgeeDB 包已从 BioContainers [80]下载约 325,000 次，显著多于过去 7 年间从 Bioconductor 下载的 14,000 次。

其他知识库的经验。UniProtKB 和 OMA 也提供 SPARQL 端点来访问其数据。另一方面，GeneCards 没有提供 SPARQL 端点。拥有 SPARQL 端点将有助于促进 GeneCards 与 UniProtKB、OMA

和 Bgee 之间的互操作，正如 [66] 中所充分展示的那样。此外，OMA 在 Bioconductor 中提供了一个 R 包以及 REST API。此外，过去 5 年间，OMA R 包（OmaDB）已通过 BioContainers 下载超过 67,000 次。另一方面，UniProtKB 和 GeneCards 没有提供由其编写和维护的 R 包，但 UniProtKB 提供了 REST API。提供一个可直接使用的 R 包，能让这两个知识库惠及生物信息学家。我们强调，R 语言是生物信息学中使用最广泛的语言之一。为了说明这样做对可重用性的重要性，Bioconductor 提供了一个并非由 UniProtKB 维护的第三方 R 包，名为“UniProt.ws：UniProtWeb 服务的 R 接口” [81]，该包已从 Biocontainers 下载超过 280,000 次。

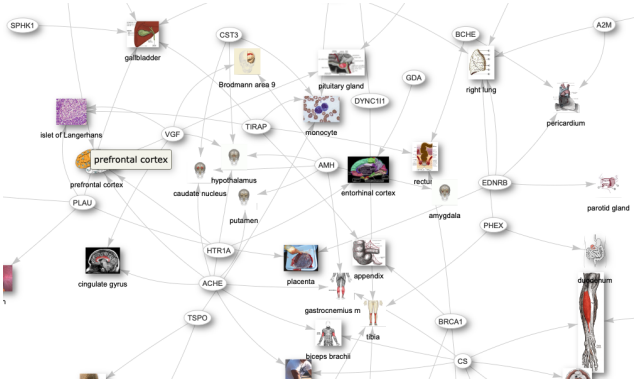


图 6。 整合到维基数据知识图谱中的部分 Bgee 数据。图中展示了基因表达调用，其中边表示“表达于”陈述。

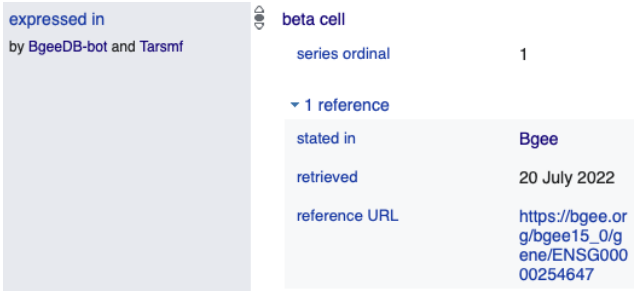


图 7。 Bgee 在维基数据中添加的一条“表达于”陈述条目，其中通过 Bgee 版本化 URL 标明了数据来源。此图截取自以下维基数据页面 https://www.wikidata.org/wiki/Q21163221。

4.3 自动化互操作性

知识库之间理想的互操作性，是能够实现无缝信息交换，让它们看起来如同一个统一的系统。为了实现这种顺畅、连续的信息交换，我们致力于自动化互操作性。例如，我们采用这种方法，使 Bgee 能够与维基数据、维基百科、Orthologous MAtrix（OMA）[23]、Google 数据集搜索引擎 [82]和 OncoMX [83]实现互操作。这种方法解决了第 4.1小节中提到的基于文件的互操作性所存在的若干问题，例如数据导出和导入操作是异步且彼此独立的。

为了实现知识库之间的自动化互操作性，不同解决方案共有的步骤之一是提供结构化数据，通常借助互操作性标准来解决语法和语义异质性。为此，Bgee 会根据具体用例采用单边、双边或多边互操作性，正如下文所述。

维基数据知识库中的 ***Bgee***

维基数据是一个开放、自由的知识库，任何主体（即人类和机器）都可以读取和编辑。它包含与其他维基媒体项目（包括维基百科）相关的结构化数据，并充当其中央存储库。维基数据的内容采用自由许可（即 CC0 [84]），可使用数据标准格式导出，也可与关联数据网络上的其他数据集互相关联。其内容包括高度相关的生命科学数据。由于 Bgee 数据也采用 CC0 许可，因此在维基数据中复用这些数据不受限制。图 6 展示了维基数据图谱的一部分，其中包含 Bgee 数据。为了与维基数据实现互操作，我们开发了一个机器人，可自动从 Bgee 关系数据库提取数据，按照维基数据数据模型 [85]对其进行结构化，并将其载入维基数据知识库。该机器人使用 Python 和 WikidataIntegrator 库 [86]编写，代码托管在我们的 GitHub 仓库中 [45]。该机器人会向维基数据插入基因条目及其“表达于”陈述。例如，请参阅 INS 基因维基数据页面 [87]中的“表达于”陈述，以及图7 中展示的一项插入操作。请注意，我们将带版本号且持久的链接定义为“表达于”陈述的引用，这是追踪信息溯源的良好做法。目前，仅考虑维基数据中已存在的 Ensembl 基因条目，以及标注了相应 UBERON 本体术语的维基数据解剖实体（例如胃），其中也包括细胞本体。因此，Bgee 中的数据并非全部都插入了维基数据。对于与维基数据共有的物种，Bgee 基因条目通过 Ensembl 基因 ID 进行标识。这样做是为了避免歧义，并确保将基因表达判定准确地纳入维基数据。因此，UBERON 本体和 Ensembl 基因标识符使我们能够处理语义异质性。

维基数据定义了特定的数据模型来组织数据，并提供了维基数据中的属性与资源描述框架中的属性之间的关系 [85]。为了与维基数据实现互操作性，我们必须复用现有的维基数据模式，或基于维基数据模型提出新模式。更多说明见 [88]。此外，通过维基数据机器人（如 Bgee 机器人）大规模插入数据，需要获得维基数据社群的授权 [89]。获得授权后，我们便能够自动将 Bgee 数据插入维基数据条目并进行更新。授权通常基于维基数据贡献者的支持而授予；目前，Bgee 机器人得到了三位不同维基数据用户的支持 [90]。因此，我们采用单边互操作性，因为我们必须严格遵循维基数据互操作流程。此外，运行 Bgee 维基数据机器人是 Bgee 流水线在每次发布版本更新时最后几个步骤的一部分。

其他知识库的经验。 GeneCards 是商业化知识库，UniProtKB 和 OMA 并非采用 CC0 许可证的知识库，因此不符合维基数据的版权要求。原则上，这些知识库无法与维基数据互操作。尽管如此，未采用 CC0 许可证的知识库可以在 CC0 许可证下捐赠部分数据。因此，它们可以与维基数据互操作，从而促进开放知识再利用，并为这些知识库带来更多流量。例如，目前 UniProtKB 在维基数据中的内容仅限于维基数据蛋白质条目中的标识符（即交叉引用）。这些交叉引用由第三方机器人提供给维基数据，*ProteinBoxBot* [91]，因此并非由 UniProtKB 维护。

维基百科上的 ***Bgee***

Bgee 与维基百科之间的互操作是完全自动化的。从最终用户的角度来看，每篇英文维基百科基因条目的信息框中，都会列出基因表达所在的解剖结构（例如胰腺），并附上指向相应 Bgee 基因页面的链接，如图 8 所示。为此，我们编写了一个 Lua [92] 脚本，该脚本定义在维基百科基因信息框模块 [47]中，用于从维基数据检索结构化数据。因此，该脚本会查询维基数据，获取 Bgee 基因表达信息并将其显示在信息框中。由于 Bgee 数据是由上文所述的维基数据机器人添加的，

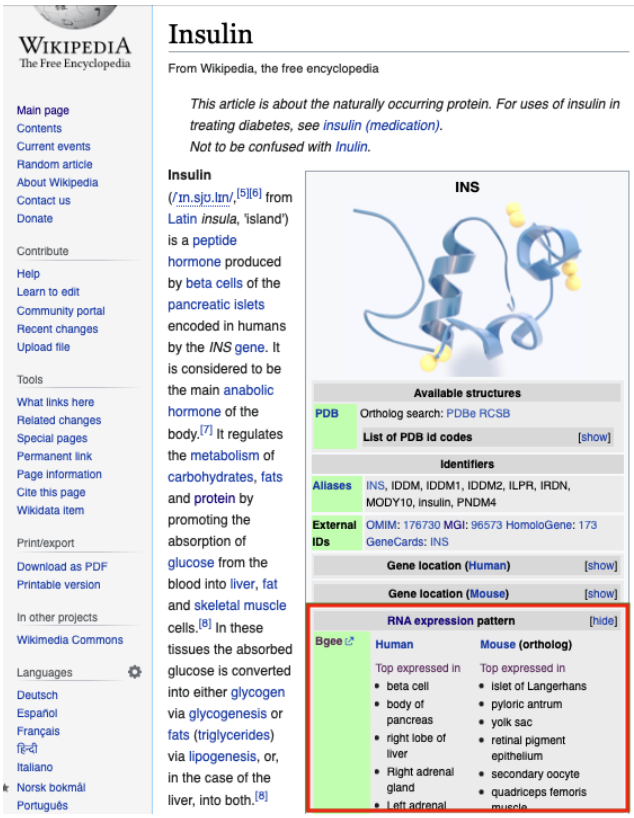


图 8。 一篇维基百科基因条目包含了来自 Bgee 的基因表达信息。这些信息提取自 https://en.wikipedia.org/wiki/Insulin。

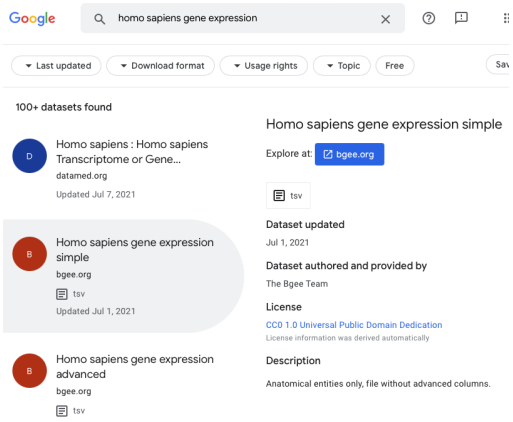


图 9。 通过 Google 数据集搜索引擎搜索人类基因表达数据集并检索 Bgee 数据集。

与 Bgee 的互操作是通过维基数据间接实现的。这样做的一项重要优势是，维基数据中的更改会迅速显示在维基百科基因条目中。与维基数据的使用场景类似，修改维基百科信息框模块的代码需要获得维基百科社区授予的权限，并完全遵循其互操作性流程，因此这是一种单向互操作方法。尽管如此，每个信息框模块都提供了一个称为沙盒的测试环境，原则上任何人都可以在其中进行编辑 [93]。

其他知识库的经验。 OMA 未收录于维基数据，因此，维基百科基因信息框模块无法像在 Bgee 用例中那样访问其数据。由于 UniProtKB 标识符可在维基数据中获取，因此维基百科基因页面的信息框中引用了这些标识符。GeneCards 也被提及，并根据从维基数据检索到的基因名称创建链接。然而，这些知识库（即 OMA、GeneCards 和 UniProtKB）均未提供对维基百科用户有实际意义、可直接使用的信息，所提供的内容不止是简单的外部知识库链接。此外，维基百科基因信息框中的 UniProtKB 和 GeneCards 链接由第三方贡献者（即非权威来源）维护。因此，这些知识库以及其他生物信息学知识库可以借鉴 Bgee 的经验，改进其知识重用。

Bgee 与 *Google* 数据集搜索引擎

此用例是多边互操作性方法的一个示例。Google 数据集搜索引擎会全自动地从包含 Schema.org 结构化数据的网页中索引并检索元数据。图 9 展示了在这一 Google 工具中搜索“智人基因表达”数据集的结果。请注意，Schema.org 并非由 Google 或 Bgee 独家管辖。因此，原则上，生成和使用 Schema.org 结构化数据与互操作的各方彼此独立。这也使其他数据使用者能够重复利用这些数据，只要它们遵循 Schema.org 方法。遵循全局数据模式、数据模型（例如 RDF 图）和语法（例如关联数据 JSON——JSON-LD [94]），本质上就能解决互操作各方之间的语义和语法异构性。Schema.org 词汇表由 Google、Microsoft、Yahoo 和 Yandex 创立，并通过邮件列表[95] 和 GitHub，采用开放社区流程进行开发。这种方法的缺点包括缺乏灵活性，而且难以就变更达成一致。因此，它极大地限制了我们能够交换的信息。例如，截至 2023 年 3 月 2 日，Schema.org 的 GitHub 上有超过 700 个未解决的问题，其中一些自 2014 起便一直悬而未决，另有约 1300 个已关闭的问题 [96]。

实现细节。 Google Dataset 仅考虑 Schema.org [97]中的 Dataset、Datacatalog 和 Download 概念及其属性。因此，与 Bgee 的互操作仅限于这些概念。尽管 Google 数据集搜索不考虑 Taxon、Gene 和“解剖结构”这些 Schema.org 概念，我们仍通过嵌入每个 Bgee 基因页面的 JSON-LD 脚本提供这些概念，任何实现这种多方互操作方法的工具都可以使用它们。图 10 展示了采用 Schema.org 结构化的“表达于”陈述，这些陈述作为脚本包含在人胰岛素 Bgee 基因页面中。

其他知识库的经验。 GeneCards 和 UniProKB 并未使用 Schema.org 来描述其数据集，也未在网页中嵌入元数据。因此，[70] 中的数据集无法通过 Google 数据集搜索直接获取。与 Bgee 类似，OMA 为数据集采用了 Schema.org，但 [69] 中列出的数据集只有少数经过描述。此外，OMA 也可以像 Bgee 一样提供更丰富的元数据，例如，为可下载的文件形式赋予标识，以便用户和软件工具直接访问和下载这些文件。

Bgee 和 *OncoMX*：知识库联邦

数据源联邦化是指能够以统一方式访问不同且可能存在异构的数据源，而无需将数据从这些数据源中实际迁移出来（即数据虚拟化）。因此，查询会直接在原始数据源上实时执行。这里所说的统一方式，是指用户看到的数据仿佛都来自同一个数据源。

为了解决第 4.1 小节中讨论的 Bgee 与 OncoMX 之间基于文件的互操作性所存在的缺陷，我们在 INODE 项目 [63] 的背景下，通过采用数据虚拟化和单边互操作方法，将两个知识库联邦起来。为实现这一联邦，我们使用 Teiid [98]，这是一种支持高查询量和事务的实时集成引擎。与 OncoMX 交换的信息由 EasyBgee 提供；如文中所述，EasyBgee 是 Bgee 关系数据库的物化视图

Gene 0 ERRORS 0 WARNINGS

ID: https://bgee.org/gene/ENSG00000254647/

@type	Gene
@id	https://bgee.org/gene/ENSG00000254647/
description	insulin [Source:HGNC Symbol;Acc:HGNC:6081]
alternateName	iddm2
alternateName	iddm1
identifier	ENSG00000254647
name	INS
expressedIn	
@type	AnatomicalStructure
@id	http://purl.obolibrary.org/obo/UBERON_0002167
identifier	UBERON:0002167
name	right lung
expressedIn	

图 10. 使用 Schema.org 结构化的 Bgee 基因表达数据示例。

第 4.1 小节及其数据模式（见图 5）所示。因此，实际操作中，我们将 OncoMX 与 EasyBgee 进行互操作。这样做是为了优化查询性能，因为 Bgee 原生关系数据库并未明确提供此联邦所需的信息。要使用 Teiid 定义虚拟数据库（VDB）（即 OncoMX 与 EasyBgee 的联邦），我们必须编写一个 XML 文件，其中记录有关 VDB、其集成的数据源以及导入元数据的偏好设置等信息 [99]。XML 还可以嵌入数据定义语言（DDL）语句（即一些 SQL命令）。图 11 展示了用于设置基于 OncoMX 和 EasyBgee 的 VDB 的 XML 文件片段。OncoMX SQL数据库转储和 VDB XML 文件可在 [100] 中获取。

因此，集成这两个数据源不再需要人工操作，也不涉及数据重复。由此可见，通过设置虚拟数据库（VDB）实现联邦化的主要优势在于，无论 Bgee 还是 OncoMX 更新，新的数据都能立即提供，而无需像通过制表符分隔值文件进行人工互操作那样处理。此外，这也有助于维护互操作性，主要是因为只需对相关数据进行修改，或调整数据源的数据模式。通常，只需编辑 VDB 配置即可完成所需更改，例如创建 SQL视图，如图 11 所示。例如，目前仅与 OncoMX 交换人类和小鼠基因表达数据；不过，如果希望使用 Bgee 中提供的其他物种，只需修改 *SELECT* 语句，在列表中添加该物种的分类单元标识符即可（9606，10090）。语义和数据异质性也可通过创建视图并使用 SQL内置函数，在查询执行期间进行数据转换（例如拼接列）来处理。因此，保持互操作正常运行变得更加容易，无需执行大规模数据导出和导入操作来部署更改。此外，如图 11 所示，我们还解决了数据存储异构性问题，因为 EasyBgee 是原生 MySQL 数据库，而 OncoMX 则是 PostgreSQL 数据库。

最后，在 INODE 项目 [63]的背景下，我们使用 Ontop 工具为 OncoMX-Bgee联邦提供基于本体的数据访问。这主要是为了实现两个目标：通过应用本体减少歧义，从而改善数据源的语义；以及提供执行联邦 SPARQL 查询的可能性，进而实现与

```
<vdb name="oncomx_federation"  version="1">
   <model visible="false" name="easybgee_v15">
       <source name="easybgee_v15" translator-name="mysql5"
           connection-jndi-name="java:/easybgee_v15"/>
       <metadata type="NATIVE"></metadata>
    </model>
    <model visible="true" type="VIRTUAL" name="oncomx_bgee">
       <source name="easybgee_v15" translator-name="mysql5"
           connection-jndi-name="java:/easybgee_v15"/>
       <metadata type="DDL"><![CDATA[
           CREATE VIEW species (
            speciesId long NOT NULL,
           genus string(70) NOT NULL,
           species string(70) NOT NULL,
           speciesCommonName string(70) DEFAULT ''  )
           AS SELECT speciesId, genus, species, speciesCommonName
           FROM easybgee_v15.species WHERE speciesId in (9606,10090);
           ...]]>
       </metadata>
    </model>
      <model visible="true" name="oncomx_v1_0_36">
       <property name="importer.tableTypes" value="TABLE,VIEW"/>
       <property name="importer.schemaName" value="oncomx_v1_0_36"/>
       <source name="oncomx_v1_0_36" translator-name="postgresql"
           connection-jndi-name="java:/oncomx_v1_0_36"/>
       <metadata type="NATIVE"></metadata>
    </model>
</vdb>
```

图 11. 用于联合 OncoMX 和 Bgee 知识库的虚拟数据库（VDB）配置文件片段。借助 XML 元素 `<model>`，我们定义要纳入考虑的数据源和元数据（例如数据模式）。还可以指定某些属性/属性值，例如，设置模型是否应在联邦中可见。声明的元数据类型为 `NATIVE`（即 `<metadatatype="NATIVE"></metadata>`），意味着数据库元数据（例如数据模式）将完全按照其在数据源中的原始定义纳入考虑。我们还可以通过 XML 元素`<metadata>` 创建视图，进而创建新的数据模式来组织带下划线的数据。这可以通过数据定义语言（DDL）语句实现，如上文第二个 `<metadata>` 标签定义所示。VDB XML 文件的完整文档见 [99]。

通过其 SPARQL 端点与其他知识图谱互通 [101]。

其他知识库的经验。据我们所知，除了 Bgee 和 OncoMX 之外，本文讨论的知识库均未对由不同组织拥有和管理的关系数据库实现联邦。

Bgee 中的 *Orthologous MAtrix*（*OMA*）

虽然在考虑两个互操作部分时，我们侧重于数据提供方，但在本小节中，我们希望从数据使用者的角度说明互操作，并进一步通过另一个知识库来证明此处讨论的互操作方法也适用。OMA 是一个知识库，包含物种间基因进化关系的信息，例如直系同源基因。直系同源基因是不同物种中由同一个祖先基因通过物种形成演化而来的基因。Bgee 整合 OMA 的进化关系数据，以进一步实现物种间的基因表达比较。

为此，Bgee 通过 OMA SPARQL 端点与 OMA 互操作。我们开发了一款工具，并编写了若干基于 ORTHology 本体的 SPARQL 查询，用于从层级直系同源群中提取成对的直系同源关系和旁系同源关系 [102]。源代码可在 Bgee 流水线的 GitHub 仓库中找到 [103]。目前，该工具已纳入 Bgee 流水线，并用于每次新发布版本，以确保与 Bgee 交换 OMA 信息的流程半自动化。之所以称为半自动化，主要是因为我们将 OMA 数据纳入 Bgee 的单边互操作方法独立于 OMA 知识库的管理，且不支持实时更新。因此，OMA 的重大变更无法由我们的专用工具完全自动处理。不过，大多数相关变更都可以通过编辑 SPARQL 查询或配置文件来完成。例如，OMA 数据模式的修改（这种情况并不常见，上次发生在三年多以前 2）可能需要修改 SPARQL 查询。因此，我们简化了互操作维护工作，并实现了与 OMA 近乎无缝的互操作。

最后，通过这个用例，我们想强调，在单向

互操作方法中，数据生产者或数据使用者都可以决定如何进行互操作。不过，像 OMA 和 Bgee 这样的生物信息学知识库提供了多种信息交换方式，这有助于提升数据可重用性，因为数据使用者可以选择最适合的方法。例如，OMA 提供了多种程序化接口，例如 Python 库、REST API 和 SPARQL，进一步实现了互操作性的自动化。因此，互操作方法虽由一方决定，但仍提供了多种选择。

4.4 最终实现考量。

需要说明的是，当有多种与目标知识库互操作的方案可供选择时，我们选择了对我们而言更容易、更快捷实现，且不会损害我们希望交换的最低限度信息的方案。否则，我们便实现了当时 Bgee 出现在目标知识库中的唯一可行方案。例如，NCBI LinkOut 系统支持两种数据语法：CSV 和 XML。我们选择了 CSV，因为这样更容易对关系数据库执行单个 SQL 查询，并直接以 NCBI 所需的表格格式获取 Bgee 数据。另一个例子与我们在维基百科上的内容有关。最初，我们只纳入了少数几个表达了人类基因的主要器官，而这些信息对于维基百科基因页面而言已经具有意义和价值。这样做是为了简化最初与维基百科版主的讨论。在 Bgee 与维基百科建立最低限度的互操作后，后续改进便更容易实施和部署。目前，维基百科还会复用 Bgee 小鼠基因表达数据，并且除了器官和组织外，也纳入了细胞。

5 提升生物信息学知识库可重用性的十条经验

经验**1**—部分互操作性胜过毫无互操作性。 我们认为，为了提高知识库的（可）重用性，实现一定程度的互操作性总比完全没有互操作性好。由于遗留系统和惯例、资源不足（例如人力资源配置与技能、技术获取）、技术不兼容或协调困难等问题，独立知识库之间往往难以实现完美且自动化的互操作。因此，若短期内无法实现全面互操作，采用部分互操作的目标，至少可以实现最低限度的数据重用，并为今后逐步改进留出空间。

经验 **2**—循序渐进地改进信息交换，胜过试图一步到位实现全面互操作。减少需要交换的信息量，可以大幅简化独立知识库代表之间的讨论及互操作工作。知识库代表更愿意开展简单、易于理解和实施的信息交换，而不是复杂的信息交换。这种简化通常会带来部分互操作性，但其巨大益处在于能够与目标知识库建立联系并实现互操作。目标知识库之后也可能更愿意接受并实施改进。因此，一旦知识库之间建立了最低限度的互操作，后续改进便更容易实施和部署。

经验**3**——可重用性意味着更高的可见度，反之亦然。 通过与其他知识库进行信息交换来传播知识库内容，可以提高其可见度，进而提升可重用性。这是因为数据不仅会被外部知识库重用，也可能被其自身的用户社群和相关软件工具重用。此外，通过提供重用数据的来源信息，用户可以访问并发现原始数据源，从而实现更高的互操作性和数据重用。

经验**4**——互操作性需要维护。 与包含维护环节的软件开发生命周期类似，确保长期互操作性也需要维护。因此，知识库之间的互操作性工作理想情况下应持续进行，只要它们仍在相互重用数据。这样也能提高重用最新数据以及交换质量更高、数量更多的信息的可能性。为减少维护工作，重要的是在实施之前，首先考虑根据互操作伙伴的需求和限制，哪种互操作性方案最为合适。

经验教训**5**—尽可能实现互操作性自动化。 这样做有多种好处，尤其是：(a) 减少维护工作，(b) 提供实时处理，以及 (c) 促进可扩展性。自动化方法能显著减少维护工作，因为知识库中的数据一旦更新，这些变更就会传播到其他具有互操作性的知识库例如，可以通过运行机器人实现，并有可能实时进行。此外，实时处理是一项互操作性功能，数据使用者几乎会立即处理收到的信息。为了进一步解释（c），我们可以强调知识库联邦方法。通过实现知识库联邦化，信息交换可以轻松扩展，因为添加新的知识库或新的信息和数据类型通过编辑配置文件并定义以下对象之间的映射来完成数据源（例如，数据模式对齐、数据转换函数）。

经验教训**6**—选择和提供互操作方法时应保持灵活。 如何建立互操作性，取决于知识库可能采用且能够使用哪些方法。此外，不同的技术和资源限制以及知识库代表的技能可能会使不同的方法更具优势。例如，实现互操作自动化需要知识库代表具备相应的技术技能，但这些技能未必总是具备。文档可以缓解这一问题（经验教训10）。此外，能够实现单边互操作性的知识库应尽可能提供不同的信息交换方式。这样做可以提高知识库的可重用性，因为它最终能够适应用户的技能，并克服第三方知识库限制，从而实现互操作。此外，如果将可重用性作为知识库的主要目标，数据生产者或消费者就应做好让步的准备；而如果需要付出更多工作，目标知识库的代表也不太愿意合作，进而不愿实施互操作。因此，在进行互操作时，大幅减少他们的工作量非常重要。例如，按照目标知识库的惯例提供即用型数据。

经验教训 **7**—关注知识库代表。当无法实现完全自动化的互操作时，只有在知识库代表之间进行沟通后，知识库之间才可能实现互操作。这应促使他们相互协作，实现知识库之间的互操作。因此，重点建立可互操作的知识库代表之间良好的工作关系十分重要。如果缺少人工协调这一环节，任何技术互操作机制都可能难以发挥作用。此外，若没有代表之间的协调，任何双边互操作方法都无法实施。

经验教训 **8**——知识库互操作性会产生积极的多米诺效应。与另一个知识库实现互操作，可以逐步构建起更完整的知识库间信息交换网络，这就是互操作性带来的积极“多米诺效应”。这主要归因于三个原因：(i) 互操作性可能具有传递性，即 A 与 B 实现互操作，而 B 又与 C 实现互操作，那么 A 也能与 C 实现互操作；(ii) 树立积极的榜样，说服更多知识库代表开展互操作；(iii) 互操作流程可能得到重用。为进一步说明后者，已有的互操作解决方案（例如用于实现知识库联邦化的数据虚拟化解决方案）可以利用可扩展性，将新知识库纳入知识库互操作网络

此外，积极参与或协作开展多方互操作倡议，也有助于产生这种积极的多米诺效应。总而言之，持续与外部资源保持一致，能够推动数据可重用性。

经验教训 9—采用最合适的许可证。 尽管这一课题已在许多不同的语境中得到讨论，包括 FAIR原则（即“（元）数据以明确且易于获取的数据使用许可证发布”[2]），我们仍希望强调，知识库应尽可能采用限制最少的许可证。这样可以消除潜在的法律互操作问题，从而让我们专注于解决信息交换中的技术和人为因素问题。尽管如此，我们也承认，有些知识库需要采用限制性很强的许可证。这样做或许是为了保护作者的数据所有权，但实际上会阻碍互操作性和开放科学的诸多可能性[104]。

经验教训 10—为互操作性提供文档、培训和教程。 为充分发挥单边互操作方法的作用，知识库应提供文档完备的技术解决方案、实用教程和培训，以促进可重用性。此外，建立能够及时响应的沟通渠道，对于确保用户持续参与至关重要，其中也包括外部知识库代表。

6 结论

总之，我们认为，最佳的互操作方法，是即便存在部分互操作性或知识库代表之间意见不一，仍能得到实施的方法。因此，单边、双边和多边互操作方法都有助于促进知识库数据的（再）利用。不过，提供并实施一种或多种此类方法，不应被视为互操作性的最终解决方案。这是因为知识库、信息交换技术与实践都在不断演进，新的知识库也不断涌现。最后，我们以 Bgee 知识库为例，阐述了多种互操作方法及其实施方式。我们还通过重点介绍 UniProtKB 等主要生物信息学知识库的类似实现，以及在适用情况下可以改进之处，说明了这些方法如何推广至其他知识库。这使我们能够通过务实的示例，为如何与 Wikipedia 等各类生物学及通用知识库实现互操作提供指导。

7 缩略语列表

应用程序编程接口—应用程序编程接口 CSV—逗号分隔值 数据库—数据库 DDL—数据定义语言 数字对象标识符—数字对象标识符 文档类型定义—文档类型定义 FAIR—可发现、可访问、可互操作和可再利用 信息框—信息框 INODE—智能开放数据探索 JSON—JavaScript 对象表示法 JSON-LD—关联数据的 JavaScript 对象表示法 知识库—知识库 NCBI—美国国家生物技术信息中心 OMA—Orthologous MAtrix OWL—Web 本体语言 资源描述框架—资源描述框架 RIKEN—日本理化学研究所 SKOS—简单知识组织系统参考 SPARQL—SPARQL 协议与资源描述框架查询语言 SWRL—语义网规则语言 SQL—结构化查询语言 太字节—太字节 TPM—每百万转录本 TSV—制表符分隔值 Turtle—简明资源描述框架三元组语言 UID—唯一标识符 虚拟数据库—虚拟数据库 VoIDext—互联数据集扩展词汇表 XML—可扩展标记语言 XML 架构定义—XML 架构定义

8 利益冲突

作者声明，他们不存在利益冲突 EST。

9 资金来源

瑞士生物信息学研究所；沃州；瑞士国家科学基金会[173048, 207853, 167149]；美国国立卫生研究院奖项[U01CA 215010 （OncoMX）]；欧盟地平线计划 2020 研究与创新计划[863410]；国家教育、研究与创新秘书处（SERI）通过苏黎世联邦理工学院拨款 BG 02-072020。开放获取费用由瑞士国家科学基金会资助。

9。1 作者贡献

概念构思：TMF；软件：TMF、JW、FB；调查研究：TMF；监督：自组织；撰写初稿：TMF；审阅与编辑：所有作者；经费获取：TMF、FB、MRR。

10 致谢

感谢我们一直保持联系的所有知识库代表持续支持 Bgee 互操作网络。我们也感谢 Bgee 团队历任及现任成员的辛勤工作，正是他们的努力使 Bgee 成为一个成功的知识库。

参考文献

1. Ancona M, Cagetti P, Castagna P, Gruosso L. 可重用的分布式“数据环境”。载于： 1992ACM/SIGAPP 应用计算研讨会论文集：1990's - SAC '92 堪萨斯城，密苏里州，美国：ACM 出版社； 1992。第 1083–1090页。 http://portal.acm.org/citation.cfm?doid=130069.130134.2. Wilkinson MD, Dumontier M, Jan Aalbersberg I, Appleton G, Axton M, Baak A 等。补遗：科学数据管理与 stewardship 的 FAIR 指导原则。《科学数据》 2019 年12月；6(1)：6。 http://www.nature.com/articles/s41597-019-0009-6.3. Jacobsen A, de Miranda Azevedo R, Juty N, Batista D, Coles S, Cornet R 等。FAIR原则：解读与实施考量。《数据智能》 2020年1月；2(1-2)：10–29。 https://direct.mit.edu/dint/article/2/1-2/10-29/10017.4. Mons B, Neylon C, Velterop J, Dumontier M, da Silva Santos LOB, Wilkinson MD。云端化，日益趋向 FAIR；重新审视欧洲开放科学云的 FAIR 数据指导原则。《信息服务与使用》 2017年3月；37(1)：49–

56. https://www.medra.org/servlet/aliasResolver?alias=iospress&doi=10.3233/ISU-170824.5. Dunning A, De Smaele M, Böhmer J。FAIR数据原则公平吗？《国际数字策展期刊》 2017年1月；12(2)：177–195。 http://www.ijdc.net/article/view/567.6. Wood-Charlson EM, Crockett Z, Erdmann C, Arkin AP, Robin-son CB。获取和给予数据认可的十条简单规则。《PLOS计算生物学》 2022 09；18(9)：1–11。 https://doi.org/10.1371/journal.pcbi.1010476.7. Bernasconi A, Simon AG, Guizzardi G, Santos LOBdS, Storey VC。FAIR原则的本体表示：构建更符合FAIR原则的数据源蓝图。载于：Indulska M, Reinhartz-Berger I, Cetina C, Pastor O 编。《高级信息系统工程》，尚佩恩：施普林格·自然瑞士； 2023。第 261–277页。8. Schwanitz VJ, Wierling A, Biresselioglu ME, Celino M, Demir MH, Bałazińska M 等。低碳能源数据的可发现性、可访问性、互操作性和可重用性：现状与行动倡议。《科学报告》2022年12月；12(1)：5208。 https://www.nature.com/articles/s41598-022-08774-0.9. Borycz J, Carroll B。为人类和机器实现FAIR数据：影响与意义——研究数据社区研讨会成果。《信息服务与使用》2020年10月；40(1-2)：71–85。 https://www.medra.org/servlet/aliasResolver?alias=iospress&doi=10.3233/ISU-200083.10. Beránková M, Kvasnička R, Houška M。迈向知识互操作性的定义。载于： 2010 2届国际软件技术与工程会议，第 1卷；2010。第V1–232–V1–236页。11. IEC 62243:2012(E)（IEEE Std 1232-2010）：面向所有测试环境的人工智能交换与服务（AI-ESTATE）。IEC 62243 第二版 2012-06 IEEE Std1232 2012；第 1–172 页。12. ISO 26324:2012 信息与文献——数字对象标识符系统；访问日期： 2023-03-17。 https://www.iso.org/standard/43506.html.13. 国际DOI基金会。《DOI®手册》。DOI.org；2019。 https://doi.org/10.1000/182.14. Edmunds M, Peddicord D, Frisse ME。互操作性难以实现的十个原因。《医疗保健信息管理系统：案例、策略与解决方案》 2016；第 127–137页。15. Benson T, Grieve G, Benson T, Grieve G。互操作性为何如此困难。《健康互操作性原理：FHIR、HL7 和SNOMED CT 2021》；第 21–40页。16. Diallo SY。互操作性的复杂性。载于：智能、自适应与自主系统建模与仿真会议论文集 2016 （MSCIAAS2016）及行星空间探索空间仿真（SPACE 2016），MSCIAAS '16，美国加利福尼亚州圣迭戈：国际计算机仿真学会； 2016。.17. Kadadi A, Agrawal R, Nyamful C, Atiq R。大数据中的数据集成与互操作性挑战。载于： 2014 IEEE国际大数据会议（大数据），IEEE； 2014。第38–40页。18. Bastian FB, Roux J, Niknejad A, Comte A, Fonseca Costa S, de Farias TM 等。Bgee套件：整合的、经过整理的表达图谱及动物比较转录组学。《核酸研究》 2020 10；49(D1)：D831–D847。 https://doi.org/10.1093/nar/gkaa793.19. Oza VH, Whitlock JH, Wilk EJ, Uno-Antonison A, Wilk B, Ga-japathy M 等。使用公共生物学数据开展研究的十条简单规则。《PLOS计算生物学》2023；19(1)：e 1010749。20. Rigden DJ, Fernández XM。 2023《核酸研究》数据库专刊与在线分子生物学数据库合集。《核酸研究》 2023；51(D1)：D1–D8。21. Safran M, Rosen N, Twik M, BarShir R, Stein TI, Dahary D, 等。《GeneCards Suite》。生命科学数据库实用指南 2021;第 27–56 页。22。UniProt： 2021 中的通用蛋白质知识库。《核酸研究》 2021;49(D1):D480–D489.23。Altenhoff AM, Train CM, Gilbert KJ, Mediratta I, Mendes de Farias T, Moi D 等。OMA 在 2021 中的直系同源关系：网站改版、保守异构体、祖先基因顺序等。《核酸研究》 2021;49(D1):D373–D379.24。de Farias TM, Roxin A, Nicolle C。用于本体互操作性的 SWRL 规则选择方法。《数据与知识工程》 2016;105:53–72.25。Farias TM, Roxin A, Nicolle C。FOWLA，一种面向本体的联邦式架构。载于：语义网规则与规则标记语言国际研讨会，Springer； 2015。第97–111页。26。George D。理解数据库模式集成背景下的结构与语义异构性。《UCLAN计算系期刊》 2005;4(1):29–44.27。Halevy A。为什么你的数据无法混合：新工具和技术可帮助减轻协调模式的难题。《Queue》2005年10月；3(8):50–58。 https://doi.org/10.1145/1103822.1103836.28。Shafranovich Y。逗号分隔值（CSV）文件的通用格式和 MIME 类型； 2005.29。在 Google Ads Editor 中导入 CSV 文件；访问日期： 2023-03-17。 https://support.google.com/google-ads/editor/answer/56368.30。文件准备：资源 CSV 文件；访问日期： 2023-03-17。 https://www.ncbi.nlm.nih.gov/books/NBK3812//#ft.File_Preparation_Resource_CSV_File.31。Vrandečić D, Krötzsch M。维基数据：一个免费的协作知识库。《美国计算机协会通讯》 2014;57(10):78–85.32。Guarino N, Oberle D, Staab S。什么是本体？载于：《本体手册》，Springer； 2009。第 1–17 页。33。Ontology Lookup Service (OLS)：Gene 的搜索结果；访问日期：2023-03-17。 https://www.ebi.ac.uk/ols/search?q=Gene&groupField=iri&exact=on&start=0&type=class.34。Otero-Cerdeira L, Rodríguez-Martínez FJ, Gómez-Rodríguez A。本体匹配：文献综述。《专家系统与应用》 2015;42(2):949–971.35。Thiéblin E, Haemmerlé O, Hernandez N, Trojahn C。复杂本体匹配综述。《语义网》 2020;11(4):689–727.36。Hitzler P, Krötzsch M, Parsia B, Patel-Schneider PF, Rudolph S 等。OWL 2 Web 本体语言入门（第二版）。W3C 建议标准 2012;https://www.w3.org/TR/owl2-primer/.37。Miles A, Bechhofer S。SKOS 简单知识组织系统参考文档。W3C 建议标准 2009;.38。Mendes de Farias T, Stockinger K, Dessimoz C。VoIDext：用于通过虚拟链接增强可互操作数据集的词汇表和模式。载于：OTM 联合国际会议《迈向有意义的互联网系统》，Springer； 2019。第 607–625页。39。Brown GR, Hem V, Katz KS, Ovetsky M, Wallin C, Ermolaeva O 等。Gene：NCBI 以基因为中心的信息资源。《核酸研究》 2014 10;43(D1):D36–D42。 https://doi.org/10.1093/nar/gku1055.40。Dingerdissen HM, Bastian F, Vijay-Shanker K, Robinson-Rechavi M, Bell A, Gogate N 等。OncoMX：用于探索癌症相关癌症生物标志物及健康数据的知识库。《JCO 临床癌症信息学》2020;(4):210–220。 https://doi.org/10.1200/CCI.19.00117，pMID： 32142370.41。Kobayashi N, Kume S, Lenz K, Masuya H。RIKEN 元数据库：作为关联开放数据云缩影的医疗保健与生命科学数据库平台。《语义网与信息系统国际期刊》（IJSWIS）

2018;14(1):140–164.42。Shefchek KA, Harris NL, Gargano M, Matentzoglu N, Unni D, Brush M 等。 2019 年的君主计划：一个连接跨物种表型与基因型的集成数据与分析平台。《核酸研究》2020;48 (D1):D704–D715.43。Morris JH, Soman K, Akbas RE, Zhou X, Smith B, Meng EC 等。可扩展精准医学开放知识引擎（SPOKE）：一个包含海量生物医学信息的知识图谱。《生物信息学》 2023;39(2): btad080.44。Cordes H, Rapp H。用于人体和动物物种生理学基础药代动力学建模的基因表达数据库。《CPT: Pharmacometrics & Systems Pharmacology》2023;.45。维基数据 BgeeDB-bot GitHub 仓库； https://github.com/BgeeDB/Wikidata_BgeeDB-bot，备注 = 访问日期：2023-03-17.46。维基百科，自由百科全书；访问日期： 2023-05-17.https://www.wikipedia.org.47。维基百科基因信息框模块；访问日期： 2023-03-17. https://en.wikipedia.org/wiki/Module:Infobox_gene.48。Brickley D, Burgess M, Noy N。Google 数据集搜索：在开放网络生态系统中构建数据集搜索引擎。载于：万维网大会； 2019。第 1365–1375 页。49。NCBI LinkOut 服务；访问日期： 2023-03-17。 https://www.ncbi.nlm.nih.gov/projects/linkout/.50。其他 LinkOut 资源：数据集、数据库等；访问日期： 2023-03-17。 https://www.ncbi.nlm.nih.gov/ projects/linkout/journals/htmllists.cgi?type_id=9.51。LinkOut 帮助；访问日期： 2023-03-17。 https://www.ncbi.nlm.nih.gov/books/NBK3812/.52。DTD 教程；访问日期： 2023-03-17。 https://www.w3schools.com/xml/xml_dtd_intro.asp.53。XML Schema 教程；访问日期： 2023-03-17。 https://www.w3schools.com/xml/schema_intro.asp.54。NCBI Gene ID 到 Ensembl ID 映射文件；访问日期： 2023-03-17。 https://ftp.ncbi.nlm.nih.gov/gene/DATA/gene2ensembl.gz.55。HBB β-珠蛋白亚基 [黑猩猩（黑猩猩）];. https://www.ncbi.nlm.nih.gov/gene/?term=ENSPTRG00000040047，备注 = 访问日期： 2023-03-17.56。UniProtKB 与 NCBI 资源（GeneID、RefSeq）之间的映射：其工作原理是什么？；访问日期： 2023-05-30。 https://www.uniprot.org/help/ncbi_mappings.57。NCBI 人类 HBB 基因页面；访问日期： 2023-05-11。 https://www.ncbi.nlm.nih.gov/gene/3043.58。Bgee-UniProtKB 交叉引用文件；访问日期： 2023-03-17。 https://bgee.org/ftp/current/XRefBgee.txt.59。Bgee-GeneCards 交叉引用文件；访问日期： 2023-03-17。 https://bgee.org/ftp/current/geneCards_XRefBgee.tsv.60。GeneCards 来源与外部链接；访问日期： 2023-05-11。 https://www.genecards.org/Guide/Sources.61。OncoMX 数据库；访问日期： 2023-03-17。 https://oncomx.org.62。Mungall CJ, Torniai C, Gkoutos GV, Lewis SE, Haendel MA。Uberon：一种集成的多物种解剖学本体。《基因组生物学》 2012;13(1):1–20.63。Amer-Yahia S, Koutrika G, Bastian F, Belmpas T, Braschler M, Brunner U 等。INODE：在实践中构建端到端数据探索系统 [扩展愿景]。arXiv 预印本 arXiv:210404194 2021;.64。资源描述框架 1.1 概念与抽象语法；访问日期：2023-03-17。 https://www.w3.org/TR/2014/ REC-rdf11-concepts-20140225/Overview.html.65。Calvanese D, Lanti D, De Farias TM, Mosca A, Xiao G。通过 Ontop 利用知识图谱访问科学数据。《Patterns》 2021;2(10):100346。

66. Sima AC, Mendes de Farias T, Zbinden E, Anisimova M, Gil M, Stockinger H 等。支持跨联邦生物信息学数据库进行语义查询。Database 2019;2019.67。资源描述框架 1.1 Turtle；访问日期： 2023-03-17。 https://www.w3.org/TR/turtle/.68。Mendes deFariasT, Kushida T, Sima AC, Dessimoz C, Chiba H, Bastian F 等。阿尔茨海默病研究中使用的数据：结合基因表达、直系同源关系、生物资源和疾病数据集。载于：即将在 14届国际语义网应用与医疗保健及生命科学工具（SWAT4HCLS）会议论文集中发表，CEUR-WS； 2023. https://github.com/tarcisiotmf/swat4hcls.69。OMA — 下载当前发布版本；访问日期： 2023-05-11. https://omabrowser.org/oma/current/.70。UniProKB — 下载；访问日期： 2023-05-11。 https://www.uniprot.org/help/downloads.71。GeneCards Suite 数据请求；访问日期： 2023-05-11。 https://www.genecards.org/Guide/Datasets.72。Bgee 应用程序编程接口；访问日期： 2023-03-17。 https://bgee.org/api/.73。OpenAPI 规范 v3.1.0；访问日期： 2023-03-17。 https://spec.openapis.org/oas/v3.1.0.74。Bgee SPARQL 1.1 端点；访问日期： 2023-05-16。 https://bgee.org/sparql/.75。SPARQL 端点 Python 接口；访问日期： 2023-03-17. https://sparqlwrapper.readthedocs.io.76。Bgee R 软件包；访问日期： 2023-03-17。 https://bgee.org/resources/r-packages.77。Bioconductor；访问日期： 2023-03-17。 https://bioconductor.org/.78。BgeeDB Docker 容器；访问日期： 2023-03-17。 https://hub.docker.com/r/bgeedb/bgee_r.79。da Veiga Leprevost F, Grüning BA, Alves Aflitos S, Röst HL, Uszkoreit J, Barsnes H 等。BioContainers：一个开源、由社群驱动的软件标准化框架。《生物信息学》 2017 03;33(16):2580–2582。 https://doi.org/10.1093/bioinformatics/btx192.80。BioContainers：bioconductor-bgeedb；访问日期： 2023-03-17. https://biocontainers.pro/tools/bioconductor-bgeedb.81。Carlson M, Ramos M，UniProt.ws：UniProt 网络服务的 R 接口。R 包版本 2.40.0；访问日期： 2023-05-11. https://github.com/Bioconductor/UniProt.ws.82。Google 数据集搜索；访问日期： 2023-03-17。 https://datasetsearch.research.google.com.83。Dingerdissen HM, Bastian F, Vijay-Shanker K, Robinson-Rechavi M, Bell A, Gogate N 等。OncoMX：一个用于在相关癌症和健康数据背景下探索癌症生物标志物的知识库。《JCO 临床癌症信息学》2020;4:210–220.84。CC 0 1.0 通用（CC0 1.0）公有领域贡献声明；访问日期： 2023-03-17。 https://creativecommons.org/publicdomain/zero/1.0/.85。Wikibase 数据模型；访问日期： 2023-03-17。 https://www.mediawiki.org/wiki/Wikibase/DataModel.86。Wikidata Integrator GitHub 存储库；访问日期： 2023-03-17。 https://github.com/SuLab/WikidataIntegrator.87。INS 基因的维基数据条目；访问日期： 2023-03-17。 https://www.wikidata.org/wiki/Q21163221.88。维基数据：模式提案；访问日期： 2023-03-17。 https://www.wikidata.org/wiki/Wikidata:Schema_proposals.89。维基数据：机器人权限申请；访问日期： 2023-03-17。 https://www.wikidata.org/wiki/Wikidata:Requests_ for_permissions/Bot.90。维基数据：机器人权限申请/BgeeDB-bot；访问日期： 2023-03-17。https://www.wikidata.org/wiki/ Wikidata:Requests_for_permissions/Bot/BgeeDB-bot.91。用户：ProteinBoxBot；访问日期： 2023-05-11。 https://www.wikidata.org/wiki/User:ProteinBoxBot。

92。Lua 语言；访问日期： 2023-03-17。 https://www.lua.org.93。维基百科基因信息框模块沙盒；访问日期： 2023-03-17。https://en.wikipedia.org/wiki/Module:Infobox_gene/sandbox.94。用于链接数据的 JSON；访问日期： 2023-03-17。 https://json-ld.org.95。Schema.org 公开电子邮件；访问日期： 2023-03-17.public-schemaorg@w3.org.96。Schema.org GitHub 未解决问题；访问日期： 2023-03-17。 https://github.com/schemaorg/schemaorg/issues?q=is%3Aopen+is%3Aissue+sort%3Acreated-asc.97。Google 数据集文档；访问日期： 2023-03-17. https://developers.google.com/search/docs/appearance/structured-data/dataset.98。Teiid：云原生数据虚拟化；访问日期： 2023-03-17。 https://teiid.io.99。Teiid 文档：XML 虚拟数据库；访问日期： 2023-03-17。http://teiid.github.io/teiid-documents/16.0.x/content/reference/r_xml-deployment-mode.html.100。OncoMX 数据库转储和 Teiid 虚拟数据库配置文件；访问日期： 2023-06-17。 https://purl.org/inode/oncomx-federation.101。OncoMX SPARQL 门户；访问日期： 2023-03-17。 http://testbed.inode.igd.fraunhofer.de:18005.102。de Farias TM, Chiba H, Fernández-Breis JT。利用逻辑规则有效表示大型直系同源数据集。载于：第 10届国际语义网应用与医疗保健及生命科学工具（SWAT4HCLS）会议论文集，第 2042 卷，CEUR-WS； 2010. http://ceur-ws.org/Vol-2042/paper36.pdf.103。OMA-Bgee 同源基因工具；访问日期：2023-05-11. https://github.com/BgeeDB/bgee_pipeline/tree/develop/pipeline/OMA_homologs.104。Lenharo M。GISAID 陷入危机：这个备受争议的新冠病毒基因组数据库能否存续？《自然》 2023年5月；617 (7961):455–457。 https://doi.org/10.1038/d41586-023-01517-9。