



CirroData Graph

技术白皮书

目录

图数据库概述	4
什么是图数据库	4
属性图模型	4
RDF 模型	5
图数据库的技术优势	5
高性能	6
弹性数据模型	6
敏捷开发	6
与关系数据库相比的主要优势	6
更快的查询和分析	6
更敏捷的数据建模	6
更强大的问题解决能力	7
深度关联分析	7
动态模式更改	7
高级聚合及分析	7
增强的机器学习和人工智能	7
图数据库应用场景	8
知识图谱	8
社交网络应用	8
地理空间管理	9
实时推荐	10
主数据管理 (Master Data Management)	10
网络和数据中心管理	11
授权和访问控制	11
其它行业应用	12
为什么需要图数据库	13
CirroData Graph	14
简介	14
技术特点	15
总体结构	16
Tinkerpop 框架	16
层次结构	16
组件结构	18
分布式存储结构	18
Loader 模式	21
性能指标	22
主流的图查询语言	22
选择 CirroData Graph	24
CirroData Graph 特点	24
图数据库的遍历与分析能力	24

CirroData Graph 能解决什么 25

运行规格..... 26

 硬件要求 26

 运行环境 26

 硬件配置 26

 方案一【ARM 服务器配置】 26

 方案二【X86 服务器配置】 27

图数据库概述

什么是图数据库

图数据库源起欧拉的图理论，基本含义是以“图”这种数据结构存储和查询数据，而不是存储图片的数据库。

图是基于事物关联关系的模型表达，通过将实体与关系点边化的方式将知识结构化地保存，因此具有天然的可解释性，从而备受学术界和工业界的推崇。图数据是指采用图结构存储的数据。

图数据库以图论为理论基础，使用图模型，将关联数据的实体作为顶点（vertex）存储，关系作为边（edge）存储，解决了数据复杂关联带来的严重随机访问问题。在分类上，图数据库可归类为 NoSQL 数据库。

图模型主要是以节点和关系（边）来体现，善于处理大量的、复杂的、互联的、多变的网状数据。

图数据库是图数据库管理系统的简称，使用图形化的模型进行增删改查的数据库。图数据库一般用于 OLTP 系统中，提供在线事务处理能力。与图数据库对应的是图计算引擎，一般用于 OLAP 系统中，提供基于图的大数据分析能力。

图数据库使用图模型来操作数据。目前使用的图模型有 3 种，分别是属性图（Property Graph）、资源描述框架（RDF）三元组和超图（HyperGraph）。现在较为流行的图数据库主要是基于属性图，更确切得说是带标签的属性图（Labeled-Property Graph），当然标签不是必须的。

图是一组对象的集合，由顶点和边构成，顶点表示实体或实例，边是将连接顶点的线，表示顶点之间的关联关系。边是图数据库中的关键概念，在关系型数据库或文档存储数据库中，没有对边的抽象概念进行直接实现。

图模型主要包含属性图、RDF 图两种。

属性图模型

属性图模型由顶点、边及其属性构成。图数据库通常是指基于属性图模型的图数据库。模型由以下 4 个部分组成：

顶点。顶点是事物（Object）的抽象，可以是“人”、“书”等实体，也可以是“端口”、“IP”等抽象；

顶点的属性。顶点的属性是对事物的具体描述，根据顶点代表的类型不同，每个顶点可以有不同的属性，比如以“人”作为顶点，属性可以是“姓名”“身高”、“生肖”等；

边。边是两个顶点间的关联关系，有向或无向。比如两个顶点是“人”和“职务”，那关系

“任职”是连接它们的边；

边的属性。和顶点的属性类似，每条边上也有属性。比如连接顶点“人”和“职务”的“当选”边，属性就是具体的当选时间。

RDF 模型

RDF (resource description framework) 模型原本是描述 Web 资源的标记语言，由 W3C 提出。RDF 主要针对文本语义的场景，用三元组 (subject-predicate-object) 来表达。因此基于 RDF 图模型的图数据库又称三元组数据库。通常意义的 RDF 模型在顶点和边上没有属性，只有一个唯一的资源描述符，这是与属性图模型最根本的区别。顶点属性可以通过顶点与属性值 (literal) 之间的边来表示，边的属性则比较复杂，需要将每条边表示为顶点，然后再将属性与之相连。

属性图和 RDF 图的主要区别有：

- 1) RDF 图可以更好地支持多值属性；
- 2) RDF 图不支持两顶点间多个相同类型的边；
- 3) RDF 图不支持边属性。

图数据库的技术优势

大数据时代带来了新的业务场景和技术挑战，关系型数据库无法解决所有问题，因此留给了包括图数据库在内的 NoSQL 数据库广大的生存和发展空间。

关系型数据库的优点在于实现了快速的逐行访问和保持数据一致性 (ACID 事务)，为此，关系型数据库对数据模式和数据规范化有着严格的要求，这种要求对数据关系的查询产生了束缚。当数据规模变大，且数据间关系随之变得复杂时，使用关系模型检索时需要多个表的属性执行连接操作，数据写入时也需考虑外键约束，从而导致较大的额外开销，产生严重的性能问题，因此传统关系型数据库更适合实体联系为一对一或一对多的应用场景。而图数据库则在处理复杂关系上有着天然优势，在海量数据的多对多的复杂实体联系场景中有着异常突出的性能表现。

图数据库与关系型数据库对比

分类	图数据库	关系型数据库
数据模型	图	表
存储对象	半结构化数据	结构化数据
2-3 度关联查询	高效	低效

4-10 度关联查询	高效	低效/不支持
事务性	支持	支持

在处理关联数据时，图数据库有三个非常突出的技术优势：高性能、灵活、敏捷。

高性能

图数据库相较于关系型数据库和其它非关系型数据库，在处理深度关联数据时，具有绝对的性能提升。

弹性数据模型

图数据库提供了极其弹性的数据模型，可以根据业务变化实时对数据模型进行修改，数据库的设计者无需计划数据库未来用例的详细信息。

敏捷开发

图数据库的数据建模非常直观，而且支持测试驱动开发模式，每次构建时可进行功能测试和性能测试，符合当今最流行的敏捷开发需求，极大地提高了生产和交付效率。

与关系数据库相比的主要优势

更快速的查询和分析

无论数据大小，图数据库均可出色查询相关数据。图模型提供了内置的索引数据结构，它无需针对给定查询而加载或接触无关数据。图模式的高效率使其成为出色的解决方案，能够更好更快地进行实时大数据分析查询。

更敏捷的数据建模

任何学习过关系型数据库建模的人都了解，满足数据库标准化和参照完整性需要遵循严格规则。在图数据库中，您可以定义所需的任意点类型，并定义任意边类型来表示点之间的关系。图模型则是按需定义，没有冗余的标准化和浪费。

更强大的问题解决能力

图数据库能够解决对于关系查询不切实际和切合实际的问题。例如迭代算法（如 PageRank、梯度下降）以及其他数据挖掘和机器学习算法。某些图查询语言是图灵完备的，这意味着您可以使用它编写任何算法。

深度关联分析

图数据模型天然具有关系管理的能力。能够在超大图上实时处理遍历多步（4 步以上）的查询，可以进行深度关联关系的挖掘和分析。可将许多用例从不可能变为可能。

动态模式更改

图模型天然没有模式。原则上，图模型允许您通过定义新的点类型和边类型来描述新的数据类型和新的关系类型。或者您可能需要添加或减少属性。您可以关联多个数据集，只需加载数据集并添加一些新边即可关联。因此可动态处理模式更改，即在图处于使用状态时进行处理。

高级聚合及分析

除了传统的按组划分聚合之外，图数据库还可以执行更复杂的聚合，这些聚合在关系型数据库中是不可想象或不切实际的。由于采用表模型，关系型数据库上的聚合查询受到数据分组方式的极大限制。与此相反，图的多维度性质和新型图数据库的并行处理功能可让用户高效地分割、切分、汇总、转换甚至更新数据，而无需预处理数据或使数据进入强模式。

增强的机器学习和人工智能

图可以用做知识图，使人能进一步推理间接事实和知识。任何图都是优化机器学习的强大武器。对于无监督学习，图模型可出色检测集群和异常，因为您只需关注关联。监督学习始终需要更多、更好的训练数据，而图能够提供先前被忽视的特性，是出色的提供源。如果机器学习 / 人工智能应用需要实时决策，我们需要快速的图数据库。图数据库成功用作实时人工智能数据基础架构的关键在于：

- 支持在最新数据流入时进行实时更新
- 表达能力高且用户友好的声明性查询语言为数据科学家提供完全的数据控制能力
- 支持实时（亚秒）深度关联遍历（> 3 步），就像人脑神经元通过神经网络发送

信息一样

- 横向扩展和纵向扩展以管理大型图，它可管理传统关系型数据库无法处理的大数据。

图数据库应用场景

图数据库在如今的大数据时代因其在海量数据挖掘、复杂关联分析、实时查询等方面具有较大的优势而得到快速发展，为大数据领域创新提供了新的思路。图数据库的应用场景多样并不断丰富。如果说**知识图谱**是图数据库最为基础的底层应用场景，充分利用了其图模型在存储和查询的优势为多行业提供知识服务，那么金融风控、社交应用等则是更具有行业特点的高阶应用场景。

知识图谱

知识图谱本质上是一种基于图的数据结构语义网络知识图谱将实体表示为顶点的集合，将实体之间的关系表示为边的集合，通过结构化的形式对知识进行建模和描述，利用可视化的图谱对知识进行展示。知识图谱使用三元组形式组织数据，关系复杂且更新、查询频繁，传统数据库无法适应存储此类模型的需求，图数据库也就因此而产生并发展。

对于行业来说，知识图谱是面向多行业的新一代知识库技术；对于图数据库而言，知识图谱是图数据库关联最为紧密、应用范围最广的应用场景。知识图谱对海量信息进行智能化处理，形成大规模的知识库并进而支撑业务应用。知识图谱中图数据库具有存储和查询两方面的技术优势：

从存储来看，图数据提供了灵活的设计模式。知识图谱中存储了多样、海量的数据。在关系型数据库中所有的数据库模式都需要提前定义，后续改动代价高昂。而图模型中，只需要重新增加模式定义，再局部调整图数据，便可完成在原有的数据源上增加标签或添加属性。

从查询来看，图数据库提供了高效的关联查询。在用图数据库存储的知识图谱中，可以通过查询实体的边和其边上的标签（即联系）来快速地获取与其相联系的另一实体，而不用再进行各种表的关联操作，关系查询的效率显著提高。

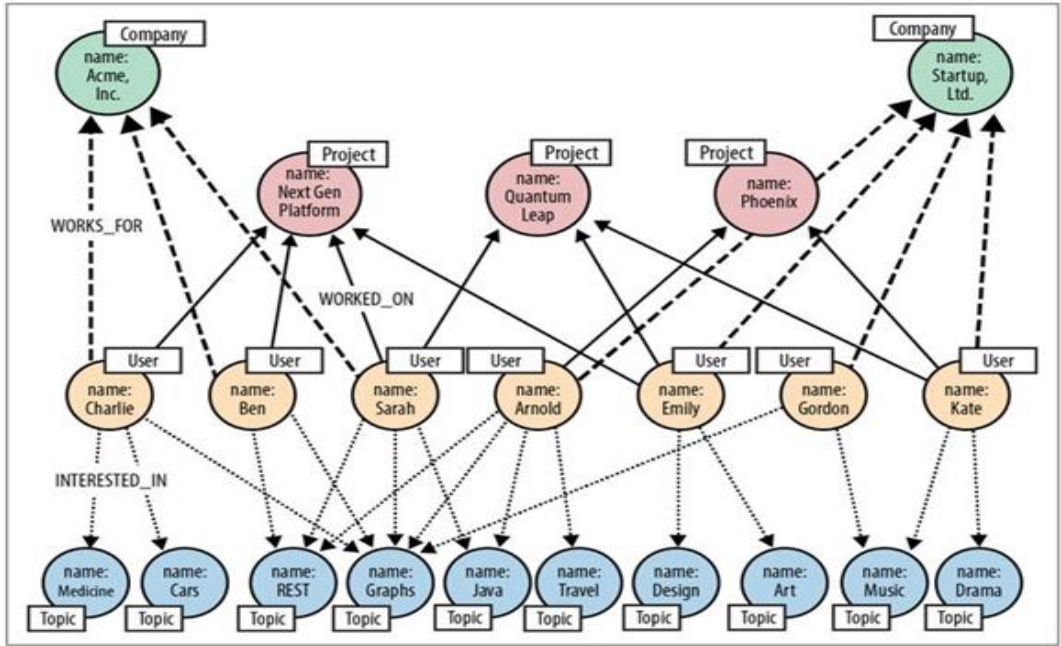
作为图数据库的底层应用，知识图谱可为多种行业提供服务，具体应用场景例如电商、金融、法律、医疗、智能家居等多个领域的决策系统、推荐系统、智能问答等。

社交网络应用

社交是人与人之间的连接，以图数据模型为内在的图数据库天生适用于明显的以联

系为中心的领域。在社交网络中使用图数据库可以方便得识别人/群组和他们交流的事物之间的直接或间接的联系，使用户能够高效地对其他人或事物进行打分、评论、发现彼此存在的关系和共同关系的事情。可以更加直观得了解社交网络中人与人之间如何互动、如何关联、如何以群组的形式来做事情或选择。

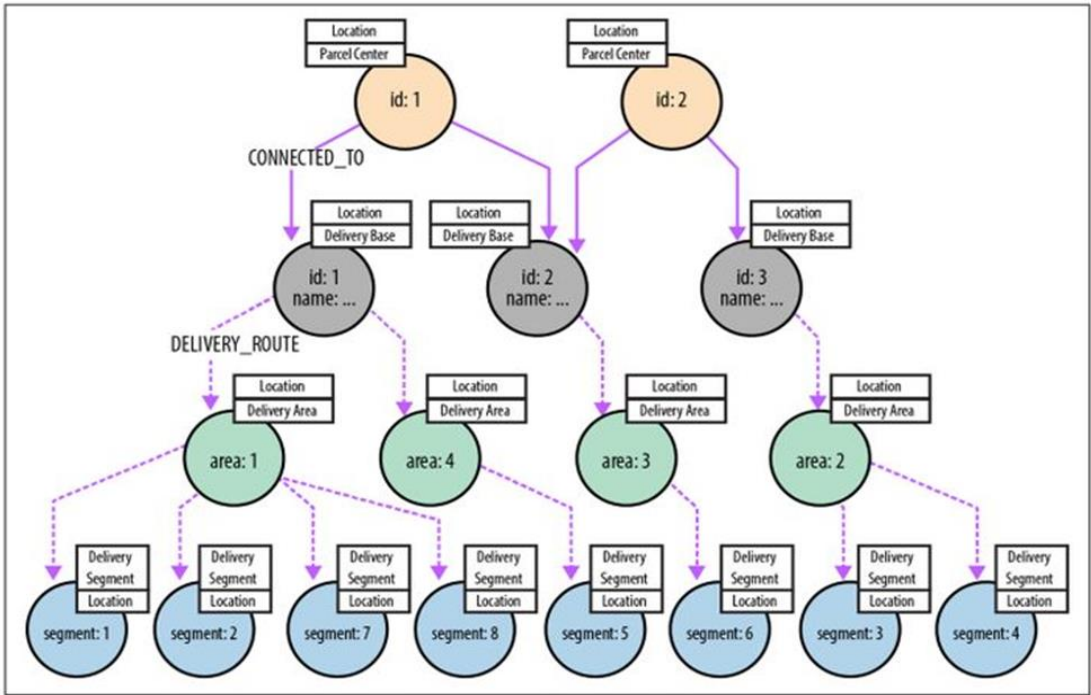
社交网络是最基础的图模型，在此基础上可以叠加更多的内容，比如个人的喜好、购买过的物品、日常的生活方式等，从而演化出更高级的图数据库应用模式，比如实时推荐系统。



地理空间管理

地理空间类的应用程序包括公路网、铁路网等，地理空间操作依赖于特定的数据结构，简单的加权带方向的联系，复杂到空间索引如 R 树。和索引一样，这些数据结构天生就以图的形式呈现，尤其是层级结构，非常适合图数据库。

总的来说，通信、物流、旅游已经路由计算相关领域的地理空间应用经常会使用图数据库。



实时推荐

在零售、招聘、情绪分析、搜索和知识管理领域，社交网络和推荐引擎可以提供关键的差异化能力，有很多种办法可以实现推荐，但使用图数据库在实时性和效率上有其特有的优势。推荐算法在人和事物之间建立联系，而联系建立的基础是用户的行为，比如购买、生产、消费、打分或评论有关资源等行为。推荐引擎可以识别出某些资源会吸引特定个人或群体，或者某些个人或群体可能对特定资源感兴趣。

一个有效的推荐依赖于对事物之间关联的理解，同时也依赖于这些关联的质量和强度，而属性图是所有这些关系密切、关联紧密的数据结构的最佳表达方式。用图数据库存储和查询这些数据使得应用程序可以为最终用户呈现实时结果，反映数据最新的变化，而不是返回给用户那些预计算的状态结果。

主数据管理（Master Data Management）

在企业或组织中，主数据管理(MDM)包括的数据涉及用户、客户、产品、供应商、部门、区域、站点、成本中心和业务单元等。这些数据来源可能是多种多样的，MDM用来识别、清洗、存储和管理这些数据。其关键问题包括谁组织结构的变化、企业合并和业务规则的变化来管理这些变化；融合新的数据源，用外部源数据补充已有的数据；解决报告需求、鉴定需求和商业智能客户的需求；当数据的值和模式变化时对数据进行版

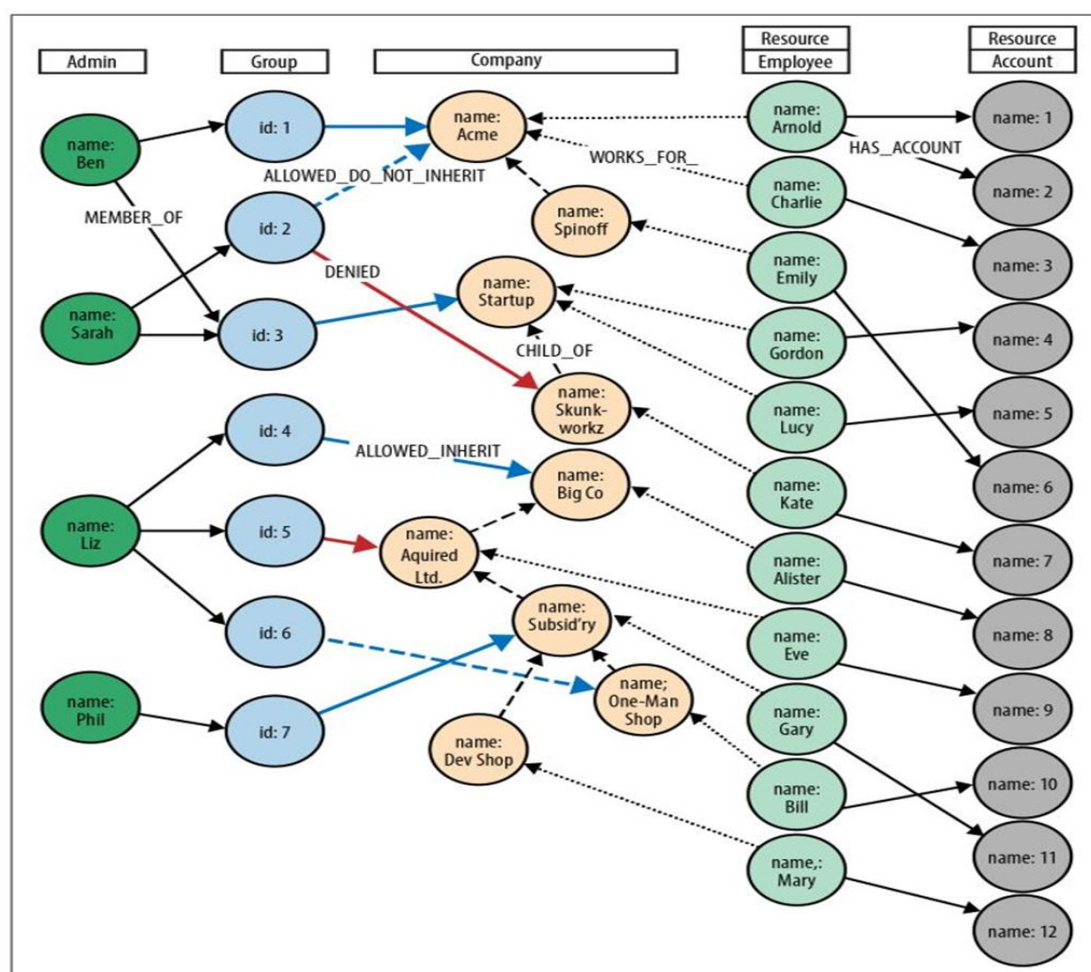
本管理。图数据库的数据模型高效匹配 MDM 的快速演变和不断变化的业务需求。

网络和数据中心管理

图数据库已经成功地使用在了电信、网络管理和分析、云平台管理、数据中心和 IT 资产管理以及网络影响分析等领域。在这些领域里，他们将影响分析和问题解决的时间从数天数小时减少到了分钟级甚至秒级。面对不断变化的网络模式，图数据库的性能和灵活性都是它适合这些领域应用的重要因素。

授权和访问控制

图数据库可以存储那些复杂的、高度关联的、跨越数十亿参与者和资源的访问控制结构。尤其适用于内容管理、联合授权服务、社交网络偏好已经软件服务化提供。将这些系统从关系型数据库切换到图数据库后，性能从分钟级提升到毫秒级。



上面仅列举了部分例子，除此之外，图数据库产品还广泛用在金融和保险行业反欺诈、风控，电商和社交类产品防机器人作弊等领域。

其它行业应用

过去几年，机器学习获得了空前的关注，使用通用模型从海量数据中获得结论，降低了专业知识的需求。但机器学习的模型是黑盒模型，结果不具有可解释性。图是基于事物关联关系的模型表达，具有天然解释性，因此图数据库与图处理引擎集成的图系统带来的强大的图存储和分析能力，推动了图数据库在金融风控等典型应用场景的落地，也带来了物联网等新的行业应用发展方向。

金融风控

互联网金融的兴起下，金融业务拓宽服务能力显著提升，但与此同时产生了大量的非结构化数据，数据量的剧增和关系的复杂多变性给金融风控带来了新的挑战。

传统关系型数据库在海量非结构化数据存储和复杂关系挖掘方面

具有天然的缺陷。相比之下，图数据库能够融合来自金融机构销售、审批、风控、电销、交易、催收等系统的多源数据以及第三方数据，打破数据孤岛，构建全方位用户关联网络。图数据库通过利用多维交叉关联信息深度刻画申请和交易行为，可以有效识别多种复杂的、规模化、隐蔽性的欺诈网络和洗钱网络；结合机器学习、聚类分析、连通图挖掘、风险传播等相关算法，可以实时计算用户的风险评分，在风险行为发生前预先识别，有效帮助金融机构提升效率、节省人工时间、控制成本、降低风险。

应用图数据库的金融风控应用场景例如个人信贷业务，失联客户管理，金融反欺诈，洗钱路径追踪，企业图谱，个人/企业征信等。

智能物联网

物联网（Internet of Things, IoT）近年来发展迅猛，“万物互联”已是大势所趋。然而，物联网要实现智能化，仍面临许多挑战：网络中传感器产生的数据量大、数据变化迅速，而且数据之间往往相互关联、查询频繁。

物联网领域中，图数据库在结合图处理引擎后还可以提供其所需的关联数据的高效复杂查询与计算能力。例如传统电力管理系统，使用关系型数据库储存传感器数据，电力供需错配、电流检查和全网状态评估等作业，涉及复杂、庞大、耗时的矩阵运算，很难在一个监视控制与数据采集周期内完成。基于图数据库的智能电力管理系统中，电力供需错配和电网状态检测作业所需的核心计算全部在图上完成，解出的值作为点和边的属性来存储，替代了开销巨大的矩阵运算，提高电网效率，同时能够检测并发现电网网络瓶颈，及时发出告警，减少电网网络故障和中断。

应用图数据库的智能物联网场景例如智能交通、车联网、智能家居、网络流量控制、智慧农业、电网智能电力管理等。

除了金融风控和智能物联网以外，图数据库还可以为公安、社交、地理等多个行业

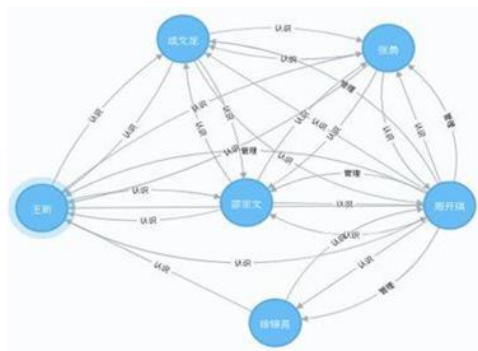
提供服务。

为什么需要图数据库

兼具直观性的图数据库

一冬胜过千言万语

比起传统的信息存储和组织模式，图数据库能够很清晰揭示复杂的模式，尤其在错综复杂的社交、物流、金融风控行业效果更为明显。



关系型数据库不擅长处理数据之间的关联关系。尤其是数据量比较大时。与传统的关系型数据库相比，图数据库的优势有：

- 1.可以很自然的表达现实世界中的实体及其关联关系;
- 2.灵活的数据模型可以适应不断变化的业务需求;
- 3.灵活的图查询语言, 轻松实现复杂关系网络的分析;
4. 图数据库在遍历关系网络并抽取信息的能力方面拥有天然优势;
- 5.关系型数据库在规模庞大时很难做多层关联关系分析, 图数据库则天然把关联数据连接在一起, 无需耗时耗内存的 join 操作, 可以保持常数级的时间复杂度。
- 6.大数据时代的新型应用需要 (知识图谱、风险控制、实时推荐、物流路网、物联网等)

CirroData Graph

简介

CirroData Graph 是东方国信分布式数据库研发中心基于开源单机版 HugeGraph 研发的一套分布式图数据库，使用 CirroData Graph 可以帮助企业快速的构建图模型数据，同时在图相关的应用场景中帮助企业提升了开发效率。CirroData Graph 基于 **Apache TinkerPop3** 框架搭建，支持 **Gremlin** 查询语言，支持**百亿级的节点和关系的快速导入**，并提供**毫秒级的关联关系查询能力**。

CirroData Graph 是一款易用、高效、通用的**开源**图数据库系统（Graph Database），实现了 Apache TinkerPop3 框架及完全兼容 Gremlin 查询语言，具备完善的工具链组件，助力用户轻松构建基于图数据库之上的应用和产品。CirroData Graph 支持百亿以上的顶点和边快速导入，并提供毫秒级的关联关系查询能力（OLTP），并可与 Hadoop、Spark 等大数据平台集成以进行离线分析（OLAP）。

CirroData Graph 典型应用场景包括深度关系探索、关联分析、路径搜索、特征抽取、数据聚类、社区检测、知识图谱等，适用业务领域有如网络安全、电信诈骗、金融风控、广告推荐、社交网络 and 智能机器人等。

CirroData Graph 支持在线及离线环境下的图操作，支持批量导入数据，支持高效的复杂关联关系分析，并且能够与大数据平台无缝集成。

CirroData Graph 支持多用户并行操作，用户可输入 Gremlin 查询语句，并及时得到图查询结果，也可在用户程序中调用 CirroData Graph API 进行图分析或查询。

CirroData Graph 具备如下特点：

- 易用：CirroData Graph 支持 Gremlin 图查询语言与 RESTful API，同时提供图检索常用接口，具备功能齐全的周边工具，轻松实现基于图的各种查询分析运算。
- 高效：CirroData Graph 在图存储和图计算方面做了深度优化，提供多种批量导入工具，轻松完成百亿级数据快速导入，通过优化过的查询达到图检索的毫秒级响应。支持数千用户并发的在线实时操作。
- 通用：CirroData Graph 支持 Apache Gremlin 标准图查询语言和 Property Graph 标准图建模方法，支持基于图的 OLTP 和 OLAP 方案。集成 Apache Hadoop 及 Apache Spark 大数据平台。
- 可扩展：支持分布式存储、数据多副本及横向扩容，内置多种后端存储引擎，也可插件式轻松扩展后端存储引擎。
- 开放：CirroData Graph 代码开源（Apache 2 License），客户可自主修改定制。

CirroData Graph 系统的功能包括但不限于：

- 支持从多数据源批量导入数据(包括本地文件、HDFS 文件、MySQL 数据库等数据源)，支持多种文件格式导入(包括 TXT、CSV、JSON 等格式)
- 具备可视化操作界面，可用于操作、分析及展示图，降低用户使用门槛
- 优化的图接口：最短路径(Shortest Path)、K 步连通子图(K-neighbor)、K 步到达邻接点(K-out)、个性化推荐算法 PersonalRank 等
- 基于 Apache TinkerPop3 框架实现，支持 Gremlin 图查询语言
- 支持属性图，顶点和边均可添加属性，支持丰富的属性类型
- 具备独立的 Schema 元数据信息，拥有强大的图建模能力，方便第三方系统集成
- 支持多顶点 ID 策略：支持主键 ID、支持自动生成 ID、支持用户自定义字符串 ID、支持用户自定义数字 ID
- 可以对边和顶点的属性建立索引，支持精确查询、范围查询、全文检索
- 存储系统采用插件方式，支持 RocksDB、Cassandra、ScyllaDB、HBase、MySQL、PostgreSQL、Palo 以及 InMemory 等
- 与 Hadoop、Spark GraphX 等大数据系统集成，支持 Bulk Load 操作
- 支持高可用 HA、数据多副本、备份恢复、监控等

技术特点

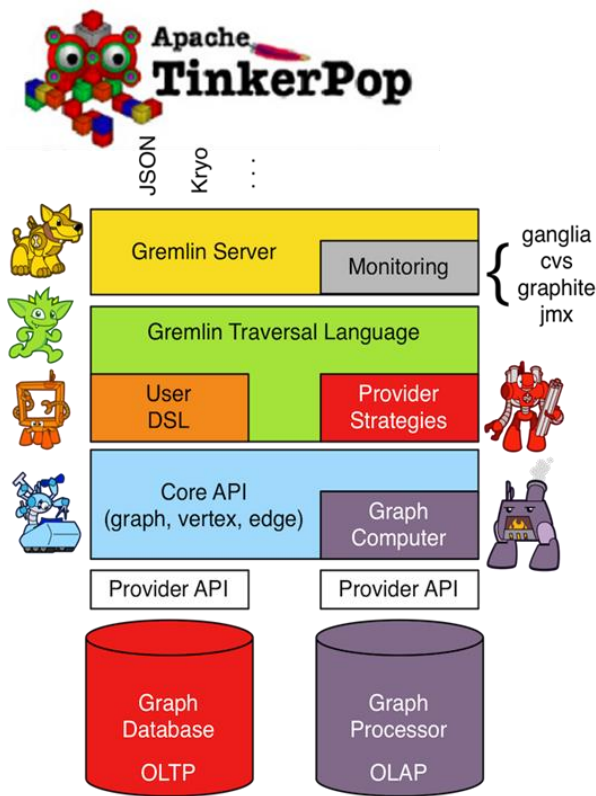
CirroData Graph 是一款面向业务操作型，支持批量操作的图数据库系统，它能够与大数据平台无缝集成，有效解决海量图数据的存储、查询和关联分析需求。CirroData Graph 基于 hugegraph 开发，支持 HBase 和 Cassandra 等常见的分布式系统作为其存储引擎来实现水平扩展。CirroData Graph 可以与 Spark GraphX 进行链接，借助 Spark GraphX 图分析算法（如 PageRank、Connected Components、Triangle Count 等）对 CirroData Graph 的数据进行分析挖掘。

CirroData Graph 的主要特点包括：

- 高可用分布式存储支撑，计算与存储分离模式
- 独立的 loading 模式，快速导入海量数据
- **基于 TinkerPop 3 API 实现，支持 Gremlin 图查询语言；**
- 拥有完善的周边工具链，提供易用高效的使用体验；
- **支持丰富的 schema 校验，支持 schema 动态修改；**
- 支持数据的备份和还原，可以在不同的后端存储之间转换；
- 多种 ID 生成策略应对不同业务场景，拥有完善的索引管理机制，支持多种索引查询操作；
- 可以实现与 Hadoop、Spark、HBase 等大数据系统集成，支持多种 Bulk Load 操作，实现海量数据快速插入；
- **支持二级索引、范围索引、全文索引、联合索引**
- CirroData Graph 还针对图数据库的高频应用（例如：ShortestPath、k-out、k-neighbor 等）做了特定性能优化，并且为用户提供更为高效的使用体验。

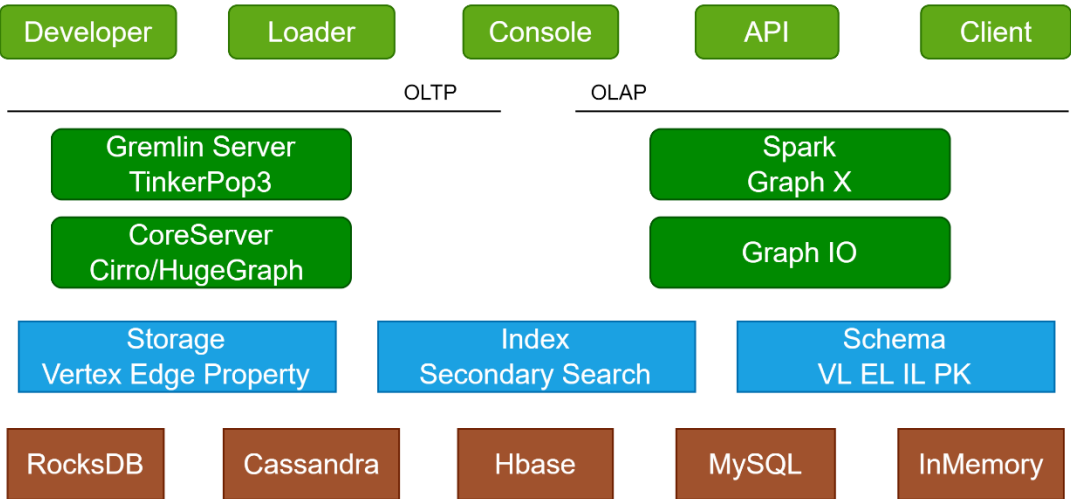
总体结构

Tinkerpop 框架



层次结构

系统架构主要包括存储层、计算层和用户接口层三个功能层次。

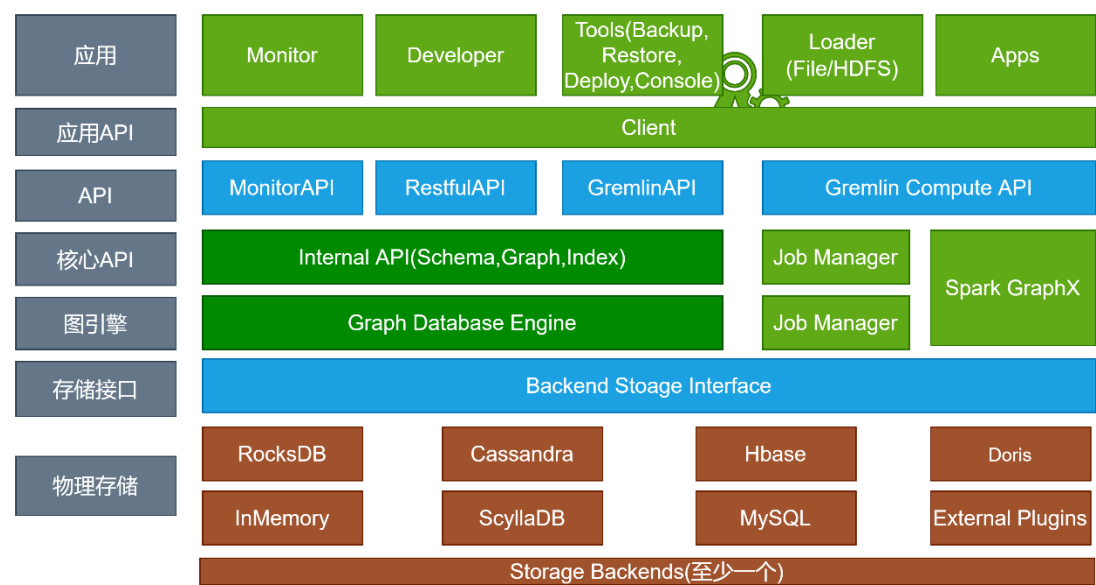


HugeGraph 的存储层包括图数据（顶点、边和属性等）存储、索引数据存储和 Schema 元数据存储。HugeGraph 后端存储会采用插件化方案，目前已经支持 RocksDB、Cassandra、ScyllaDB、HBase、Doris（原 Baidu Palo）和 MySQL 等，后续会适配更多的后端存储系统。HugeGraph 的计算层包括 OLTP 和 OLAP 两种类型。其中 Cirro/HugeGraph 重点实现了 OLTP 核心功能，而 OLAP 部分功能需要和 Spark GraphX 相结合完成。

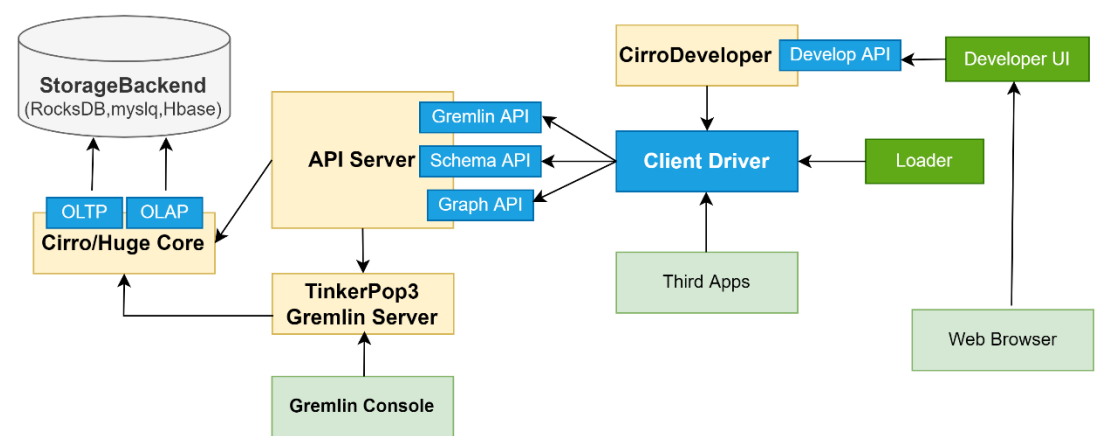
HugeGraph 提供了多种接口方式，用户可以根据自己的业务需求和喜好，选择相应的接口来操作 CirroGraph，具体包括：

- Developer/Studio（基于 Web 的 IDE 可视化编辑环境）
- Loader（数据导入工具）
- Console（基于命令行的 Gremlin 接口）
- Api（基于 Rest 的 Api 接口）
- Client（基于 Java 的客户端驱动程序）

组件结构

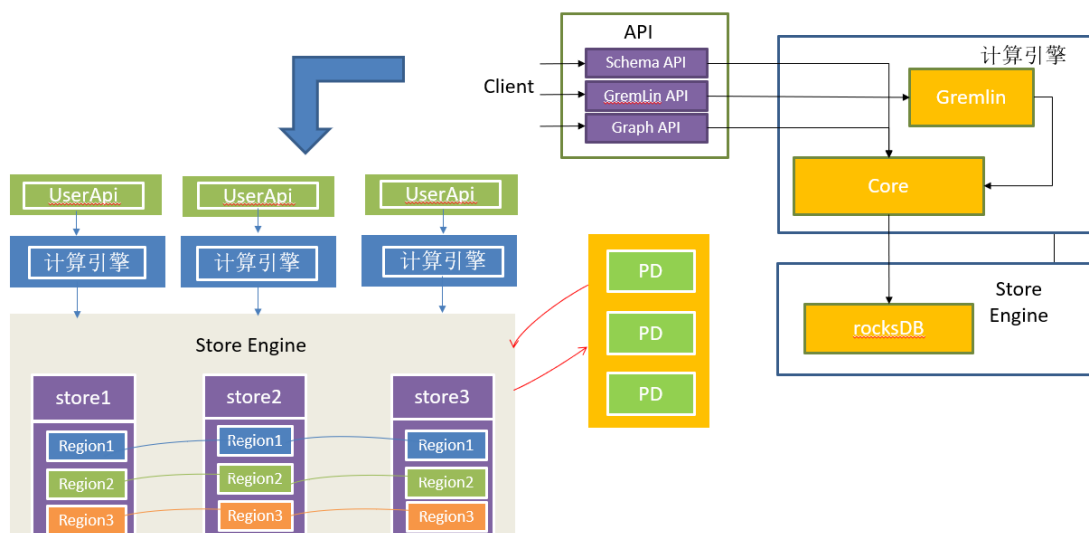


各组件之间的关系



分布式存储结构

CirroData Graph 扩展了 HugeGraph 的存储引擎，实现了高可用。 CirroData Graph 扩展的目标结构如下：



图中 Store1、2、3 中的 region leader 分别为 region1、2、3。

- Store 向 pd 发送心跳
- region leader 向 pd 发送心跳。
- region leader 向 follower 发送日志。

■ Placement Driver

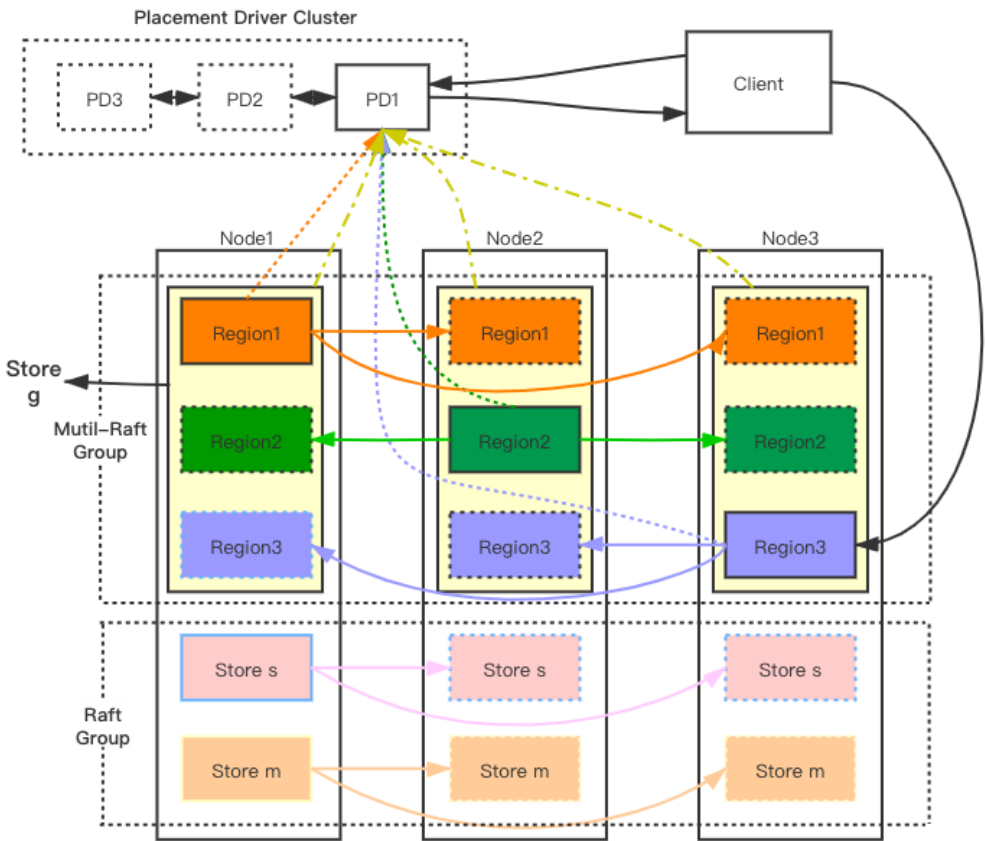
简称 PD，PD 服务需要单独部署，主要负责整个集群的管理调度，Region ID 生成等。

■ Store

一个节点一个 Store，可看作 Region 的容器，储存多个分片的数据。

■ Region

数据存储的最小单元，每个 Region 存在多副本，存在不同的 Store 中，一起组成 Raft Group。



原 HugeGraph 中分为 s、m、g 三个存储端实例，CirroData Graph 将实际存储顶点与关系的实例 g 进行开发，实现 Multi-raft-group 逻辑分区，保证批量插入时，每个节点都会工作。

并且保证高可用，只要服务集群中超过半数节点正常，即可正常对外提供服务。使用 pd 集群来记录各 Store 中的 region 信息和状态，客户端访问时，通过 pd 获取 region 信息，然后向对应 region 的 leader 发起请求。

高可用 Raft

作为一个分布式系统，KVStore 的 replication，scale out 等功能需要 Raft 的支持。CirroData Graph 使用 Raft 来支持了一致性。

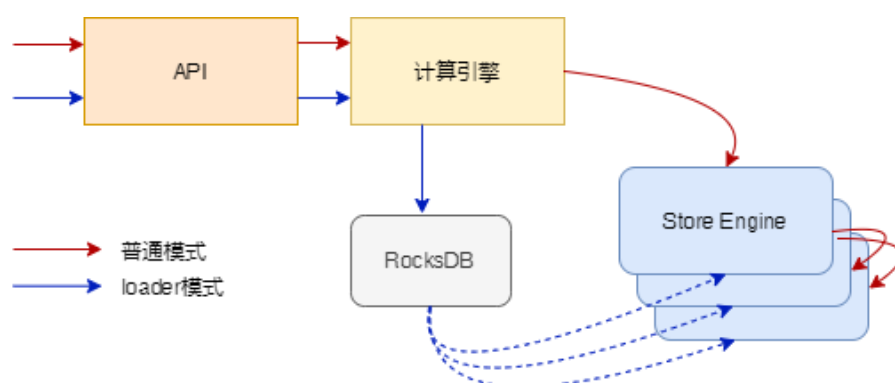
CirroData Graph 的包含三个存储实例：g、m、s。其中 s 实例中用来存储图数据库运行过程中的一些任务信息，m 用来存储 schema 信息，g 用来存储边和点的数据。在实际应用过程中，数据主要是存储到 g 实例中的，根据这一特性，如上图所示：在 m 与 s 中只做了单 raft group 的扩展，在 g 实例中实现了 Multi-raft-group。

PD

Placement Driver（简称 PD）是整个集群的管理模块，其主要工作有三个：一是存储集群的元信息；二是对集群进行调度和负载均衡（如 region 分裂、Raft group leader 的转移等）；三是分配全局唯一的 RegionID。

PD 是一个集群，需要单独部署，一般线上推荐至少部署 3 个节点。

Loader 模式



普通模式下，数据经由 API、计算引擎加载到一个节点的存储引擎中，然后再通过 Jraft 同步到其它两个节点中因为需要保证数据的一致性，所以在数据加载方面会存在很大的性能瓶颈，为适配 CirroData Graph，我们新增了 loader 模式。

在 loader 模式下，数据经由 API、计算引擎先落地到 RocksDB 中，然后再将 RocksDB 的 sst 文件以异步任务的方式通过 JRaft 同步到所有节点中，这样在保证数据一致性的前提下，也让加载性能有了一个质的提高。

在 loader 模式下，我们的加载性能可达单机版 HugeGraph 性能的 90%左右，这对于分布式系统来说已经达到了一个非常不错的性能。后期我们还会继续完善 loader 模式，争取在数据加载性能方面与单机版持平，甚至可以在增加集群机器数量后超越单机版。

性能指标

HugeGraph批量插入的性能

数据库	email-enron (30w edge)	amazon0601 (300w edge)	com-youtube.ungraph (300w edge)	com-lj.ungraph (3000w edge)
HugeGraph	0.629	5.711	5.243	67.033
TitanDB	10.15	108.569	150.266	1217.944
Neo4j	3.884	18.938	24.890	281.537

HugeGraph K-out性能

测试场景	维度	一度	二度	三度	四度	五度
K-out	时间	0.039s	0.045s	0.053s	1.10s	2.92s
	返回顶点数	10	140	2555	50825	1,070,230

HugeGraph最短路径查询性能

数据库	email-enron (30w edge)	amazon0601 (300w edge)	com-youtube.ungraph (300w edge)	com-lj.ungraph (3000w edge)
HugeGraph	0.494	0.103	3.364	8.155
TitanDB	11.818	0.239	377.709	575.678
Neo4j	1.719	1.800	1.956	8.530

主流的图查询语言

与关系型数据库不同，图数据库领域目前没有统一的查询语言标准，大多数查询语言与产品紧密关联。声明式查询语言通常作为常规查询语言，提高图数据的易用性；而命令式查询语言用在对图数据性能有较高要求的场景，同时复杂图分析场景也多使用命令式语言。据了解目前国内一些图数据库产品使用的是自研的声明式查询语言。而常见的声明式图数据库查询语言包括 Cypher, Gremlin 和 SPARQL。

图数据库查询语言

查询语言	提出者	介绍
------	-----	----

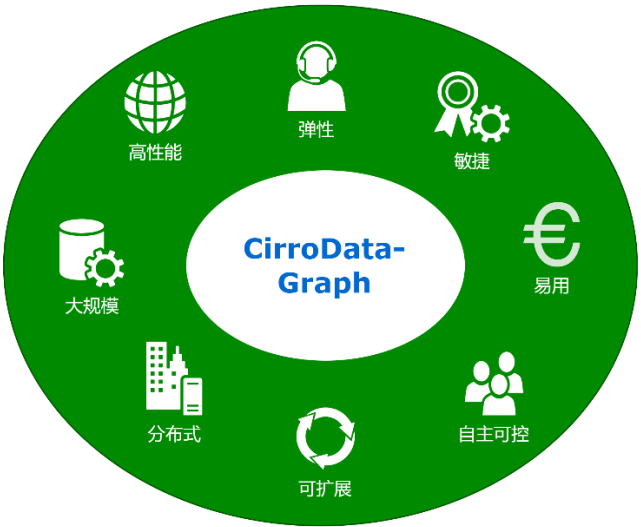
Cypher	Neo4J 提出	采用类 SQL 语法，其开源版本为 OpenCypher
Gremlin	Apache TinkerPop 开源项目的一部分	采用类 Scala 语法
SPARQL	W3C 标准	SPARQL 是一种用于资源描述框架 (RDF) 的查询语言

除了上述查询语言外，目前业界相当一部分产品使用自己开发定义的查询语言。

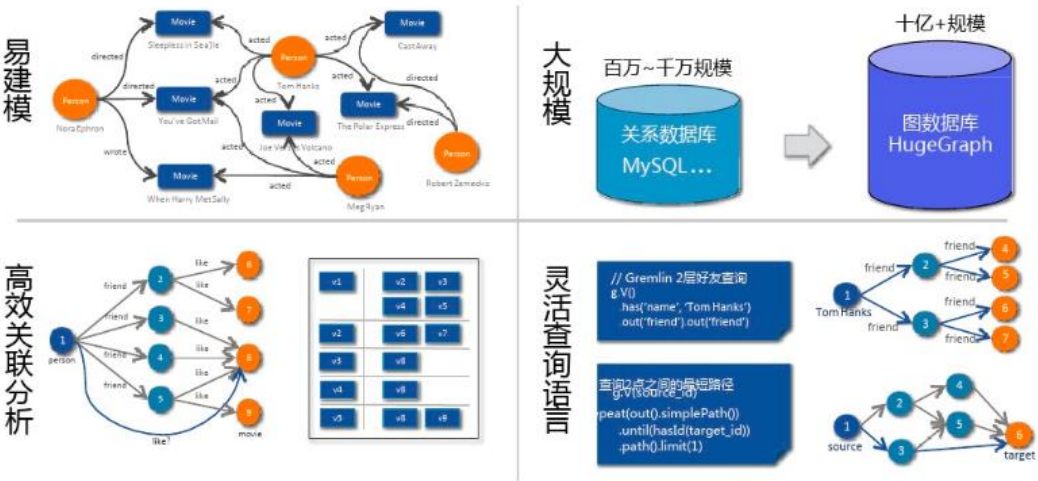
CirroData Graph 支持图灵完备的 Gremlin 查询语言。

选择 CirroData Graph

CirroData Graph 特点



- 易建模、大规模数据、高效关联分析、灵活查询语言（Gremlin）



图数据库的遍历与分析能力

- ◆ 支持常用的图查询算法：包括广度优先、深度优先查询、入跳出跳查询、邻居查询、共同邻居查询、全路径查询、最短路径查询、多节点关联查询等；

- ◆ 支持主流图分析算法，包括中心性分析算法、社群发现算法、图挖掘算法、人群关系算法等（部分算法可以结合 GraphX 分析）；
- ◆ 支持 Gremlin 查询语言接口，不支持类 SQL 查询；
- ◆ 支持可视化图构建、图管理、图分析等。

CirroData Graph 能解决什么

- ✓ CirroData Graph 图数据库的核心能力是为解决复杂关联关系而生的。CirroData Graph 图数据模型和图数据库天生适用于以联系为中心的领域
 - ✓ CirroData Graph 也可以找到更多的关系，包括物与物、物与人、人与人之间的关系，这也是传统数据库无法解决的问题。
 - ✓ CirroData Graph 图天生没有模式。CirroData Graph 数据结构能随时变更 Schema。
-
- **分钟级到毫秒级的性能提升：**将复杂 join 查询转换为图的快速遍历，而图的遍历几乎与数据集的大小关系不大。
 - **开发周期的显著缩短：**领域专家、技术架构师和开发人员可以使用同样的模型来讨论和描述核心领域问题，减少了领域与技术之间的阻抗不匹配，没有了规范化和反规范化之间的困扰。
 - **急速的业务响应能力：**变化的商业环境、技术以及用户行为催生系统的快速应变能力：新业务、新模式必然导致数据转移和转换、数据模型的修改，这个数据模型又与应用程序紧密相关。CirroData Graph 图天生没有模式，意味着可以快速适应业务和环境的变化。
 - **企业可用：**数据是重中之重，一个关键业务中使用的数据技术必须是健壮和可扩展的。大数据环境下的可用性不仅体现在数据上，也体现在性能响应上。
 - **自主研发，安全可控。**CirroData Graph 是基于国人开发的 HugeGraph 二次开发的，开发团队深度参与，掌握自主可控、安全可信的数据库核心技术。CirroData Graph 适配多种国产操作系统以及服务器。

运行规格

硬件要求

- 环境节点数：最少 3 台，推荐 3 台；
- CPU：Intel 或 AMD 4 核及以上主流型号；
- 内存：单台机器的内存至少 64GB ECC RAM，推荐 128GB 及以上 ECC RAM；
- 存储：只需使用本地硬盘，推荐单台机器 6 个硬盘位；硬盘型号不做特别要求，为降低系统构建成本，可以采用 SATA 中端普通服务器硬盘；硬盘空间大小随应用的数据规模而定；
- 网卡：千兆或者万兆网卡，为了达到最佳性能推荐使用多网卡并在交换机上进行链路聚合。

运行环境

- OS：64 位 R CentOS7.0 及以上
- JVM：预装 64 位 JDK 1.8 及以上版本

硬件配置

CirroData Graph 分别提供了基于 ARM 服务器和 X86 服务器的节点硬件的部署方案。方案一均为 ARM 服务器节点配置；方案二均为 X86 服务器节点配置。下面按照硬件具体类型分别提供硬件配置要求。

方案一【ARM 服务器配置】

配置	要求【示例】
CPU	CPU: 鲲鹏 920*2 48Core
内存	不少于 64GB DDR3 ECC 内存
硬盘	硬盘：600GB SAS*2, 1.8T SAS*12
网络	网卡：Hi1822*1
电源	双路电源供应
操作系统	64 位 CentOS7.6

方案二【X86 服务器配置】

硬件要求

配置	要求【示例】
CPU	不少于 2 颗，单颗不少于 6 核 Intel CPU
内存	不少于 64GB DDR3 ECC 内存
硬盘	4*2T 2.5 SAS 硬盘，支持 SAS/SATA/SSD，支持 raid0\1\5\6\10
网络	2*10GbE 的网络连接，两网卡做 bond 以提供失效保护
电源	双路电源供应
操作系统	64 位 CentOS6.5