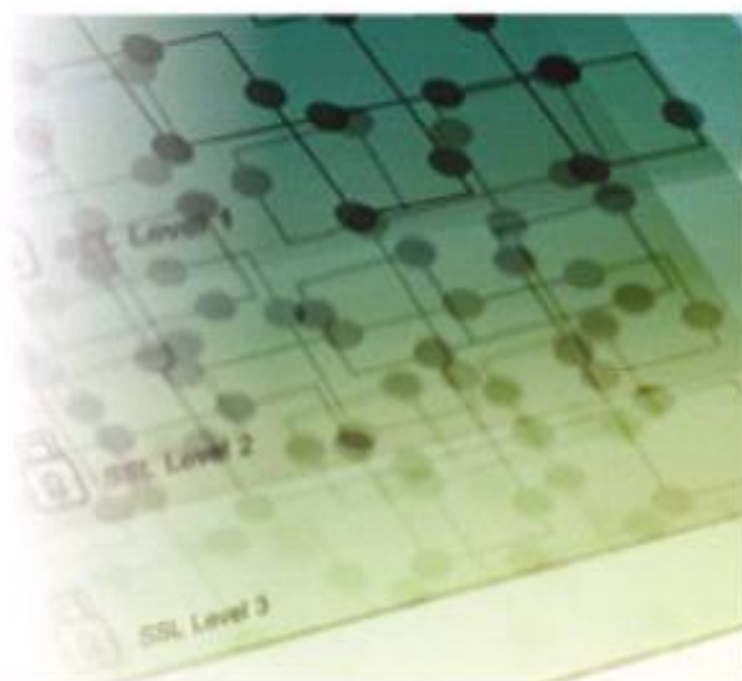


Getting Started With

InfoSphere Data Architect 快速入门

包含
手动练习

适合程序开发人员和数据库管理人员



ERIN WILSON
SAGAR VIBHUTE
CHETAN BHATIA
RAHUL JAIN
LIVIU PERNIU
SHILPA RAVEENDRAMURTHY
ROBERT SAMUEL

合著

李涛
白云峰
仇璐

合译

DB2 ON CAMPUS BOOK SERIES

InfoSphere Data Architect

快速入门

ERIN WILSON, SAGAR VIBHUTE, CHETAN BHATIA,
RAHUL JAIN, LIVIU PERNIU,
SHILPA RAVEENDRAMURTHY, ROBERT SAMUEL

第一版

第一版（2011年6月）

© Copyright IBM Corporation 2011. All rights reserved.

IBM Canada
8200 Warden Avenue
Markham, ON
L6G 1C7
Canada

声明

本信息是为在美国提供的产品和服务编写的。

IBM 可能在其他国家或地区不提供本文中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

有关双字节字符集（DBCS）信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711

本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区： International Business Machines Corporation “按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许除明示或暗含的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和/或程序进行改进和/或更改，而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是此 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的，实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 **IBM** 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。**IBM** 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 **IBM** 产品的声明。有关非 **IBM** 产品性能的问题应当向这些产品的供应商提出。

所有关于 **IBM** 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

本信息包含在日常业务操作中使用的数据和报告示例。为尽可能表述完整，这些示例将包括个人、公司、品牌和产品的名称。所有这些名字都是虚构的，若现实生活中实际业务企业使用的名字和地址与此相似，纯属巧合。

版权许可：

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 **IBM** 付费。这些示例并未在所有条件下作全面测试。因此，**IBM** 不能担保或暗示这些程序的可靠性、可维护性或功能。样本程序“按现状”提供，不附有任何种类的保证。对于因使用样本程序而产生的任何损坏，**IBM** 将不承担任何责任。

本出版物中所提到的 **IBM** 产品或服务并不暗示 **IBM** 将在所有 **IBM** 开展业务的国家或地区中提供它们。

如果您正在查看本信息的软拷贝，图片和彩色图例可能无法显示。

标商

IBM、IBM徽标和ibm.com均为International Business Machines Corp.在全球范围许多管辖区域内的商标或注册商标。其他产品和服务名称可能是IBM或其他公司的商标。Web 站点 www.ibm.com/legal/copytrade.shtml上的“版权和商标信息”中提供了IBM商标的最新列表。

Java 和所有基于 Java 的商标是 Sun Microsystems, Inc.在美国和/或其他国家或地区的商标。

Microsoft 和 Windows 是 Microsoft Corporation 在美国和/或其他国家或地区的商标。

Linux 是 Linux Torvalds 在美国和/或其他国家或地区的注册商标。

UNIX 是 Open Group 在美国和其他国家或地区的注册商标。

其他公司、产品或服务名称可能是其他公司的商标或服务标记。

目录

序.....	11
本书的目标读者.....	11
本书的架构.....	11
一本属于社区的书.....	12
说明.....	12
下一步?.....	12
关于作者.....	15
本书的贡献者.....	16
致谢.....	17
翻译者.....	17
第一部分 – 概述.....	19
第一章 – IBM InfoSphere Data Architect 简介.....	21
1.1 什么是 IBM InfoSphere Data Architect?.....	21
1.2 系统配置要求.....	22
1.3 下载 DB2 Express-C.....	23
1.4 下载 InfoSphere Data Architect.....	23
1.5 安装 InfoSphere Data Architect.....	23
1.6 为 IBM InfoSphere Data Architect 应用许可证.....	27
1.7 运行 IBM InfoSphere Data Architect.....	29
1.7.1 工作台简介.....	31
1.7.2 数据透视图和相关视图介绍.....	34
1.7.3 视图支持的操作.....	35
1.7.4 复位透视图的默认设置.....	36
1.8 练习.....	36
1.9 总结.....	37
1.10 本章回顾.....	37
第二章 – 数据建模概述.....	39
2.1 数据模型设计的生命周期.....	39
2.2 组织数据模型.....	41
2.3 创建学生信息管理系统.....	41
2.4 总结.....	41
2.5 下节展望.....	42
第二部分 – 数据建模.....	43
第三章 – 逻辑数据建模.....	45
3.1 逻辑数据建模：概述.....	46
3.2 创建逻辑数据模型.....	46
3.2.1 使用工作台来创建逻辑数据模型.....	47
3.2.2 使用图来创建实体.....	48
3.2.3 添加关系.....	56
3.3 创建词汇表模型.....	59

3.3.1 命名标准和词汇表模型的最佳实践.....	60
3.3.2 创建词汇表模型.....	61
3.4 使用命名标准	63
3.4.1 使用命名标准分析检查合规性	64
3.5 约束	64
3.6 练习	65
3.7 总结	66
3.8 本章回顾.....	66
第四章 – 域模型.....	69
4.1 域模型	69
4.1.1 创建一个空白的域模型.....	70
4.1.2 原子域	71
4.1.3 列表域和并集域.....	72
4.2 将域模型元素关联到逻辑数据模型元素.....	73
4.3 练习	74
4.4 总结.....	75
4.5 本章回顾.....	75
第五章 – 物理数据建模.....	77
5.1 物理数据建模：概述.....	78
5.2 从头开始创建一个物理数据模型	78
5.3 转换逻辑数据模型到物理数据模型	79
5.4 完善物理数据模型	81
5.4.2 对 DB2 进行存储建模	83
5.5 提炼物理数据模型	85
5.5.1 在物理数据模型中重排表列	85
5.5.2 在物理数据模型中创建角色	86
5.5.3 在物理数据模型中增加一个用户 ID.....	87
5.5.4 验证物理数据模型	88
5.6 生成 DDL.....	89
5.6.1 从数据库对象中生成 DDL 脚本	89
5.7 练习	92
5.8 总结.....	93
5.9 本章回顾.....	93
5.10 下节展望.....	93
第三部分 – 迭代设计：报告、反向设计和模型分析	95
第六章 – 报告的创建，模型的导入和导出.....	97
6.1 报告的创建，模型的导入和导出：概述.....	98
6.2 洞察报告功能	99
6.3 创建 BIRT 报告.....	99
6.3.1 创建基本物理数据模型报告	99
6.3.2 配置报告设计环境	100
6.3.3 为报告添加数据对象	102
6.3.4 报告数据分组	105

6.3.5 为报告添加动态文本	108
6.3.6 从模板创建报告配置	110
6.4 创建 XSLT 报告	111
6.5 模型的导入/导出	112
6.5.1 模型的导出	112
6.5.2 模型的导入	113
6.6 练习	114
6.7 总结	115
6.8 本章回顾	115
第七章 – 反向设计	117
7.1 反向设计：概述	118
7.2 使用工作台进行反向设计	118
7.2.1 从 DDL 脚本进行反向设计	119
7.2.2 从数据库进行反向设计	121
7.3 修改新建的物理数据模型	122
7.4 通过比较和合并来同步变更	123
7.4.1 对数据库进行比较和合并变更	123
7.4.2 比较和合并功能的优点	127
7.5 练习	127
7.6 总结	128
7.7 本章回顾	128
第八章 – 模型映射和发现	129
8.1 映射模型：概述	129
8.1.1 使用映射模型管理元数据	130
8.1.2 使用映射模型进一步管理命名标准	130
8.2 在工作台构建映射	131
8.2.1 创建一个空白的映射模型	131
8.2.2 添加映射到映射模型	133
8.3 映射类型	138
8.4 为映射模型添加表达式和过滤器	138
8.5 生成可部署的脚本	140
8.6 将映射模型导出为 CSV 格式	141
8.7 练习	141
8.8 总结	141
8.9 本章回顾	142
第九章 – 分析数据模型	143
9.1 分析数据模型：概述	143
9.2 使用工作台分析数据模型	143
9.2.1 使用工作台分析逻辑数据模型	143
9.2.2 使用工作台分析物理数据模型	144
9.2.3 在问题视图中修复错误和警告	145
9.3 修改模型分析的首选项	145
9.4 总结	145

10 InfoSphere Data Architect 快速入门

9.5 练习	145
9.6 本章回顾	146
第十章 – 数据管理的生命周期	147
10.1 管理您的数据	147
10.1.1 数据管理的生命周期	148
10.1.2 IBM InfoSphere Data Architect 与其它产品集成	149
10.1.3 同其它基于 Eclipse 的产品进行 shell-sharing	150
参考文献	153
资源	153
网站	153
书籍	155
联系邮箱	155

序

在知识爆炸，新兴技术不断涌现的时代，如何在有限的时间内站在新兴技术的前沿，越来越成为一项挑战。DB2® on Campus 系列丛书将帮助您在最短的时间里学习到这些新兴技术。

本书的目标读者

本书面向于那些将使用 IBM InfoSphere® Data Architect 学习数据建模基础知识的读者，IBM InfoSphere® Data Architect 是一个基于 Eclipse 的工具，它支持多种不同的数据库服务器。通过 IBM InfoSphere® Data Architect，您可以为不同的数据环境设计模型并进行部署，并且还可以将它与其它基于 Eclipse 的产品进行集成。

本书的架构

本书的架构如下：

- 第 1 章介绍 IBM InfoSphere Data Architect 及如何运行 InfoSphere Data Architect 工作台。
- 第 2 章介绍数据建模的基本概念及在本书的练习中所要完成的项目。
- 第 3, 4, 5 章，引导您完成数据建模过程：
 - 第 3 章教您逻辑数据建模，并向您展示如何开始创建数据模型。在本章中，您将学习实体、属性、关系、词汇表模型及命名规范。
 - 第 4 章帮助您熟悉域模型。在本章中，您将学习如何创建特殊数据类型，该类型能够帮助您伪装数据以保护个人隐私信息。
 - 第 5 章向您阐述物理数据建模。在本章中，您将学习如何将现有的逻辑数据模型转换为新的物理数据模型，并基于此物理数据模型生成 DDL 脚本进行部署。
- 第 6, 7, 8, 9 章，帮助您了解迭代设计过程：
 - 第 6 章让您学习如何使用 IBM InfoSphere Data Architect 创建报告。您将学习如何构建 BIRT 与 XSLT 报告，方便数据模型在更大的数据建模团队中进行分享。
 - 第 7 章阐述如何使用工作台进行反向设计。您将学习如何利用 DDL 脚本及现有的数据库连接创建物理数据模型，这样您就可以修改模型并将其部署到数据服务器上。
 - 第 8 章介绍映射模型及如何利用映射模型集成不同的数据模型和数据源。
 - 第 9 章包括如何分析数据模型以确保它们的有效性，以及是否满足通用的最佳实践和设计标准或者确保数据模型部署到服务器上不会引起错误。
- 第 10 章阐述了如何使 IBM InfoSphere Data Architect 在数据管理方面发挥更大的价值，展示了如何将它与其它 IBM 产品进行集成，使其在整个数据生命周期中，进一步的设计、开发和管理数据模型。

本书的大部分章节都包含练习。同时每章的复习题将帮助您更好地学习本书的内容。

一本属于社区的书

本书由大学教授、学生及专业人员（包括 IBM 员工）所组成的社区所创作，并免费向社区提供。本书由来自世界各地的社区成员翻译成不同的语言版本。如果您想提供反馈信息、贡献新的材料、改进现有材料、或者帮助翻译本书到其它语言，请将您的计划发送邮件到 db2univ@ca.ibm.com 并署标题为 IBM InfoSphere Data Architect book feedback。

说明

本书中包含许多示例，包括命令、SQL 语句和代码。关键字以粗体大写字母表示，如：**NULL** 表示未知状态。命令以粗体小写字母表示，如：**dir** 命令列出 Windows 下的所有文件及子目录。SQL 语句以粗体大写字母表示，如：**SELECT** 语句用于获取表中包含的信息。

示例中所包含的对象名称以粗斜体表示。如：表 *flights* 包含 5 列。

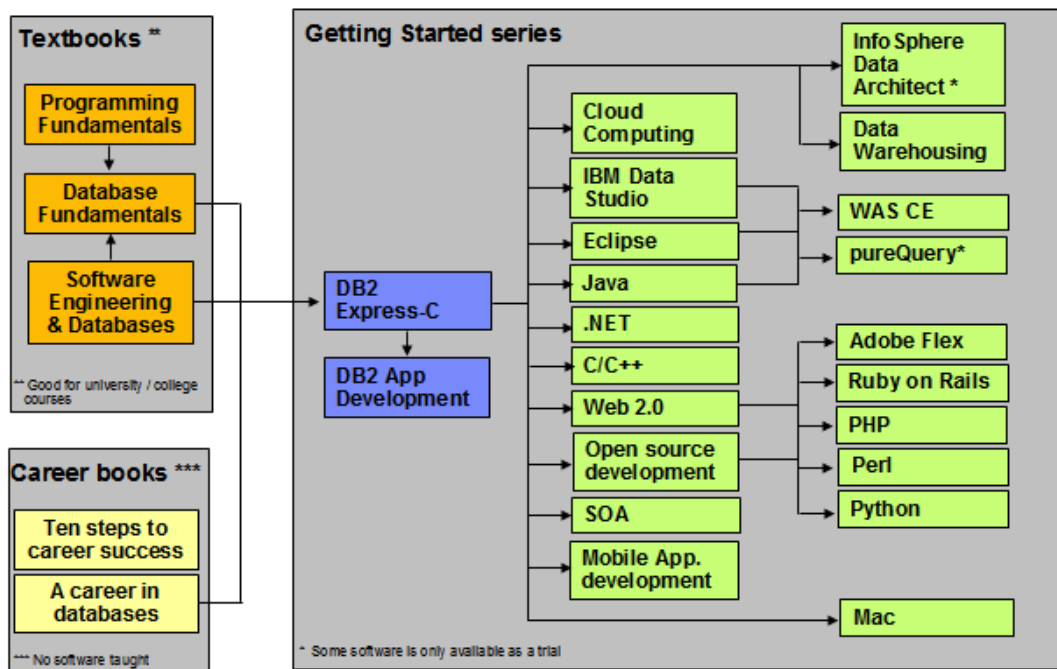
命令及语句中的变量名也用斜体表示。如果变量名中包含多个词语，词语间由下划线连接。如：**CREATE TABLE *table_name***。

下一步？

我们推荐您参阅如下系列丛书，以获得更加详细的信息：

1. *Eclipse 快速入门*
2. *DB2 Express-C 快速入门*
3. *IBM Data Studio for DB2 快速入门*

下图展示了 DB2 on Campus 系列丛书所包含的所有书目，它们可以从 ibm.com/db2/books 免费下载。



DB2 on Campus 系列丛书

关于作者

Erin Wilson 是 IBM 硅谷实验室 (Silicon Valley Laboratory) 的一名信息开发人员。作为 InfoSphere Data Architect 信息开发团队的领导者，她为数据架构师编写文档。她曾经为 InfoSphere Optim™ 数据生命周期产品线中多个基于 Eclipse 的产品工作，特别是数据建模和数据仓库方面。她有着丰富的 DB2 数据库知识。除了编写产品文档，她还致力于为数据生命周期产品线的演示示例制作说明。在加入 IBM 之前，Erin 毕业于普度大学 (Purdue University)，获得专业写作学位。因为由衷地热爱电脑与技术，她在业余时间学习了大量的编程及 web 开发知识。

Sagar Vibhute 于 2008 年加入 IBM。目前他是班加罗尔印度软件实验室 (ISL) JCC 开发团队中的一员。之前他在 InforSphere Data Architect 的 Continuing Engineering 团队工作。他毕业于班加罗尔信息技术国际研究所 (IIIT-Bangalore)，获得信息技术硕士学位。在业余时间，他喜欢弹吉他与骑车旅游。

Chetan Bhatia 从 2008 起进入 IBM 工作。目前他是 InfoSphere Data Architect 的一名开发人员。他拥有计算机应用硕士学位并且在软件开发领域有着 11 年的经验。他拥有 Web 开发的相关经验，现在从事于基于 Eclipse 平台产品的开发工作。他热衷于新技术与新的小应用程序，同时他也喜欢游泳与读书。

Rahul Jain 于 2008 年加入 IBM。他是一名软件开发人员，现在是班加罗尔 ISL InfoSphere Data Architect 开发团队中的一员。在此之前，他在 InfoSphere Data Architect 的 Continuing Engineering 团队中工作。他拥有化学学士学位并在班加罗尔信息技术国技研究所 (IIIT-Bangalore) 取得信息技术硕士学位。他最喜欢听着音乐开车行驶在高速公路上。

Liviu Perniu 是罗马尼亚布拉索夫的 Transilvania 大学自动化系的一名副教授，他教授的课程包括数据需求、分析与建模。他是 Eclipse Innovation Awards 项目的 IBM 2006 Faculty Award 的获得者，同是也是 DB2 on campus 系列丛书中 *Database Fundamentals* 的作者之一。

Shilpa Shree R. 毕业于电子与通信专业。她在 Java 与 J2EE 方面有着 6 年的 IT 经验。她目前是 iGate Global Solution 公司的一名系统分析员。

Pauljayam Sp Robertsamuel 在 IBM 工作了两年多。他拥有康复医学理疗学位，但他改变了兴趣，如今成为了一名 InfoSphere Data Architect Level 3 产品支持工程师。他拥有地理信息系统领域的相关经验。他的爱好包括读书和游泳。

本书的贡献者

以下人员为本书的编辑，审阅，内容的提供做出了重要贡献。

贡献者	所在公司/学校	职位/职业	贡献
白云峰	IBM 中国开发中心	软件工程师	技术审阅
Raul F. Chong	IBM 多伦多实验室	DB2 项目高级经理	DB2 on Campus 丛书的整体项目协调、编辑、格式校订及图书的审阅
Don Clare	IBM 硅谷实验室	InfoSphere Data Architect 软件工程师	技术审阅
Steve Corcoran	Aviva UK Health – Eastleigh, United Kingdom	数据建模工程师	技术及内容审阅
Joe Forristal	IBM – 爱尔兰	商业智能分析员	技术审阅
Leon Katsnelson	IBM 多伦多实验室	IBM Data Servers 项目主管	技术审阅
Hemant Kowalkar	IBM 硅谷实验室	InfoSphere Data Architect 软件开发工程师	技术审阅
李涛	IBM 中国开发中心	InfoSphere Data Architect 软件开发工程师	技术审阅
Wei Liu	IBM 硅谷实验室	Data Tools 软件开发工程师	技术审阅
Diem Mai	IBM 硅谷实验室	Data Studio tools 软件开发工程师 (installation)	技术审阅
仇璐	IBM 中国开发中心	InfoSphere Data Architect 软件工程师	技术审阅
Robin Raddatz	IBM 奥斯汀	SQL 与 XQuery tools 软件开发工程师	技术审阅
Neil Wang	IBM	InfoSphere Data Architect 软件开发工程师	技术审阅
Minghua Xu	IBM 硅谷实验室	InfoSphere Data Architect 软件开发工程师	技术审阅
Joseph Yeh	IBM 硅谷实验室	InfoSphere Data Architect 软件开发工程师	技术审阅

致谢

我们真诚地感谢下列的每一个人，他们帮助和开发本书所引用的材料：

感谢来自 IBM 硅谷实验室的 **Jin Leem** 为第三章阐明逻辑数据建模制作图表。

感谢 **Natasha Tolub** 为本书设计了封面。

感谢 **Susan Visser** 对本书发布的帮助。

感谢 **Kathryn Zeidenstein** 及编写 *IBM Data Studio for DB2 快速入门* 的其他团队成员，他们制作的材料成为本书第 1 章的框架。

翻译者

感谢来自 IBM 的以下人员为本书的翻译、审阅做出了重要的贡献。

贡献者	所在公司/学校	职位/职业	贡献
李涛	IBM 中国开发中心	软件工程师	翻译，内容审阅
白云峰	IBM 中国开发中心	软件工程师	翻译，内容审阅
仇璐	IBM 中国开发中心	软件工程师	翻译，内容审阅
党丰容	IBM 中国开发中心	软件工程师	内容审阅，格式审阅
陶佰明	IBM 中国开发中心	软件工程师	翻译，截图
马骏	IBM 中国开发中心	软件工程师，经理	翻译
刘麒赞	IBM 中国开发中心	软件工程师	翻译
黄飞	IBM 中国开发中心	软件工程师	翻译
曾妍	IBM 中国开发中心	软件工程师	翻译
袁波	IBM 中国开发中心	软件工程师	翻译

第一部分 – 概述

1

第一章 – IBM InfoSphere Data Architect 简介

IBM InfoSphere Data Architect 是一个企业级的数据建模工具，它拥有一套完整的开发环境能够帮助您对异构分布的数据资产进行建模，并且发现和创建这些数据资产之间的关系。

通过本章，您可以了解到：

- 什么是 IBM InfoSphere Data Architect
- 如何使用 IBM InfoSphere Data Architect
- 系统配置要求
- 如何安装 IBM InfoSphere Data Architect

1.1 什么是 IBM InfoSphere Data Architect?

当前企业拥有数量庞大的数据，并且许多决策的制定都依赖于这些数据。IT 业面临非常明确的挑战：了解数据、提高数据的质量和确保数据的一致性、确保数据的设计满足商业需求。IBM InfoSphere Data Architect 是一个协作的数据建模和信息集成工具，用于发现、建模、关联和标准化各种数据资产。它的主要功能和优点包括：

- InfoSphere Data Architect 支持对多种数据源进行分析并发现其数据结构。通过 JDBC 和数据源建立连接后，用户使用 InfoSphere Data Architect 可以方便地浏览数据库中的数据结构，它还提供了易于操作和理解的界面，对数据库中的对象按照层次结构分类，便于用户对数据库中的每一个对象的详细信息进行查看和理解。
- InfoSphere Data Architect 支持多种数据库，如 DB2®、Informix®、Oracle、Sybase、Microsoft SQL Server、MySQL 和 Teradata 等。用户可以为这些数据库创建逻辑数据模型、物理数据模型和域模型。逻辑数据模型和物理数据模型中的元素还可以用图进行可视化展现。图的展现采用信息工程（IE）表示法显示，同时，在物理数据模型中的图还可以通过 UML 表示法显示。对于物理数据模型的创建，InfoSphere Data Architect 不但支持从头开始设计创建物理数据模型，还支持从逻辑数据模型转换为物理数据模型和通过反向设计创建物理数据模型。

IBM InfoSphere Data Architect 还支持创建和设计多维逻辑数据模型和多维物理数据模型。同样地，您可以从无到有地创建新的多维模型，也可以通过反向设计从已有的数据源中创建。

注意：

本书不包含对多维数据模型的介绍。如果需要了解更多关于多维数据模型的概念，可以参考 IBM InfoSphere Data Architect 的信息中心：

http://publib.boulder.ibm.com/infocenter/rdahelp/v7r5/topic/com.ibm.datatools.dimensionai.ui.doc/topics/c_ida_dm_container.html

- 大多数数据开发项目都采用迭代开发的方式。因此，模型设计中的增量变化控制管理和减少增量变化带来的影响非常重要。InfoSphere Data Architect 可以帮助解决这个问题。分析影响功能会帮您列出所有与选中数据对象相关的数据对象。比较和同步功能则可以帮您对两个模型，或者模型和数据库，或者两个数据库进行对比分析，并且支持您在模型或目标数据库中进行设计的改变。您还可以使用 Optim™ Database Administrator 来进行复杂的 DB2 数据库对象的改变和数据迁移。
- InfoSphere Data Architect 支持用户设计和实现统一的标准，这些统一的标准可以用来提升企业数据的质量、规范数据在命名、含义、取值、关联、权限、隐私以及可跟踪性方面的一致性。这些标准一旦定义好，可以应用于不同种类的模型和数据库。同时，InfoSphere Data Architect 还提供了内置的、可扩展的、基于规则的分析验证功能，用来对数据模型或者数据库进行命名、语法、范式化以及是否满足最佳实践等方面的验证。
- 不论是对于小的工作团队中每个人承担多种职能的情况，或者是对于一个大的团队里每个人有明确地工作划分的情况，您都可以将 InfoSphere Data Architect 与基于 Eclipse 的版本控制软件集成起来使用，如 Rational® Clear Case 或者并发版本控制系统（CVS）。

1.2 系统配置要求

在安装 IBM InfoSphere Data Architect V7.5.3 版本前，您需要确认您的电脑满足系统配置的要求，关于系统配置要求的信息可以参考以下链接：

<http://www-01.ibm.com/support/docview.wss?uid=swg27019867>

链接中包含对硬件、软件和数据源的要求。

注意：

如果您选择通过扩展已有的 Eclipse 环境进行安装，需要确保已有的 Eclipse 的版本为 3.4.2 并且 JDK 的版本为 IBM JDK 1.6 SR7。

1.3 下载 DB2 Express-C

本书中多数练习涉及到的数据库使用的是免费的 DB2 Express-C，可以参考下面的步骤来下载和配置数据库：

1. 从 ibm.com/db2/express 下载数据库 DB2 Express-C
2. 参考 db2university.com 中的视频安装 DB2 Express-C
3. 配置 DB2 Express-C 中的 SAMPLE 数据库，在随后的练习中我们将会把数据模型部署到这个数据库中。

注意：

关于 DB2 Express-C 的更多信息，请参阅本系列丛书之一 [Getting started with DB2 Express-C](#)。

1.4 下载 InfoSphere Data Architect

如果您是学生、讲师或者研究员，您可以从 IBM Academic Initiative 处索取到免费版本的 IBM InfoSphere Data Architect。您必须隶属于某一个大学并且满足下面的条件之一：

1. 数据建模研究方向的学生
2. 数据建模或数据架构方向的讲师
3. 使用本产品来辅助学术研究

关于 IBM Academic Initiative 的更多信息可以访问下面链接：

<https://www.ibm.com/developerworks/university/academicinitiative/>

如果您需要购买 IBM InfoSphere Data Architect，可以访问下面的产品页面：

<http://www-01.ibm.com/software/data/optim/data-architect/>

您可以从下面的文档中，通过选择不同的操作系统找到您需要的 Passport Advantage[®] 号：

<http://www-01.ibm.com/support/docview.wss?uid=swg24028103>

如果您想免费试用 30 天，可以下载试用版的 IBM InfoSphere Data Architect，试用版下载链接如下：

<http://www.ibm.com/developerworks/downloads/r/rda/index.html>

1.5 安装 InfoSphere Data Architect

您可以使用 Launchpad 界面方式手动安装或者使用配置文件方式自动安装 IBM InfoSphere Data Architect。使用配置文件自动安装指的是可以将安装需要的选项写在配置文件中，然后运行配置文件进行安装，安装过程不需要用户干涉，这种方式在需要在多台机器上进行安装时非常有用。

在这一章里我们将重点介绍使用 Launchpad 界面的安装方式。IBM InfoSphere Data Architect 的安装需要在 IBM Installation Manager 中进行，下面我们假设您还没有安装 IBM Installation Manager，如果您已经安装了相同版本的 IBM Installation Manager，就不需要再次安装。

注意：

正如信息中心中对于 Data Security, Privacy, 和 Lifecycle Management 描述的一样（http://publib.boulder.ibm.com/infocenter/idm/docv3/index.jsp?topic=/com.ibm.datatools.rda.install.doc/topics/c_plan_imover.html），IBM Installation Manager 是一个用于安装，更新和修改安装包的程序。它可以帮助您管理计算机上已安装的 IBM 应用程序或者安装包。除此之外，IBM Installation Manager 还可以帮助检测已经安装了哪些应用，判断哪些应用目前可以被安装和对安装目录进行组织管理等。

安装之前您需要拥有目标机器操作系统的管理员权限。

请按照以下步骤进行 IBM InfoSphere Data Architect 产品的安装：

1. 解压文件，然后运行 launchpad：
 - Windows 操作系统：在解压后的文件夹中运行 **setup.exe** 文件。
 - Linux 或 UNIX 操作系统：在解压后的根目录中运行 **setup** 命令。

运行后会打开 launchpad。

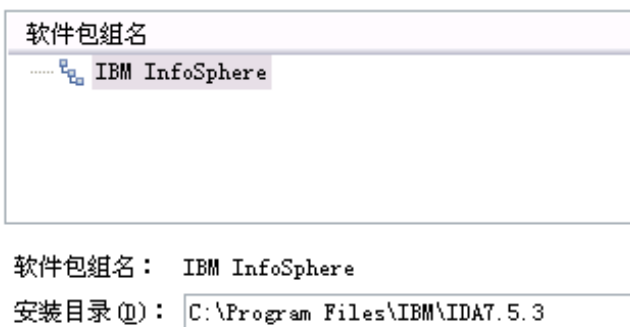
2. 如图 1.1 所示，选择左侧列表中的“安装产品”。
在安装产品页面，您可以选择需要的安装方式。



图 1.1 – 点击安装产品运行 Installation Manager

3. 选择“管理安装”，IBM Installation Manager 将会被打开。
4. 选择您需要安装的软件包，对于本书中的练习，我们选择 *IBM InfoSphere Data Architect* 软件包，点击“下一步”。
5. 仔细阅读并接受许可证协议中的条款，点击“下一步”。
6. 创建一个新的软件包组并选择安装目录：
 - a. 由于我们之前没有安装任何软件包组，我们选择“创建新的软件包组”，如 图 1.2 所示。

- 使用现有软件包组 (U)
- 创建新的软件包组 (C)



软件包组名

..... IBM InfoSphere

软件包组名: IBM InfoSphere

安装目录 (U): C:\Program Files\IBM\IDA7.5.3

图 1.2 – 为 IBM InfoSphere Data Architect 创建新的软件包组

- b. 为共享资源和 Installation Manager 选择安装目录。在本书练习中，您可以使用默认的路径，但是需要注意确保目标磁盘有足够的空间以备之后可以和其它基于 Eclipse 的产品进行 shell-share。

注意：

如果之后使用 Installation Manager 安装产品，您将不能选择共享资源的目录。

- c. 点击“下一步”。
7. 如果在目标机器上已经有安装好的 Eclipse 并且版本号为 3.4.2，您可以选择扩展已有的 Eclipse 开发环境，使用这种方式将只安装产品特有的功能，不会改变您的已有的 Eclipse 开发环境的设置。对于本书练习，我们使用默认设置，点击“下一步”。
 8. 选择需要的额外语言包，并点击“下一步”。
 9. 选择需要安装的功能部件。本例中，我们选择安装所有的功能部件，点击“下一步”。

10. 选择适合的访问帮助系统的方式。默认情况下，所有的帮助文档都存放在 IBM InfoSphere Data Architect 信息中心，用户可以通过 web 方式远程访问。您也可以选择将所需的帮助下载到本地或者在内部网络中某一台机器集中访问。本练习中，我们选择“从 Web 访问帮助方式”，点击“下一步”。
11. 在摘要页面，您可以确认一遍之前的选择。确认之后点击“安装”。
现在 IBM Installation Manager 开始安装产品。安装进度条有可能会偶然停止，请在产品成功安装完成之前不要打断安装或者退出安装。当产品成功安装完成后，可以看到如下界面，如图 1.3 所示。

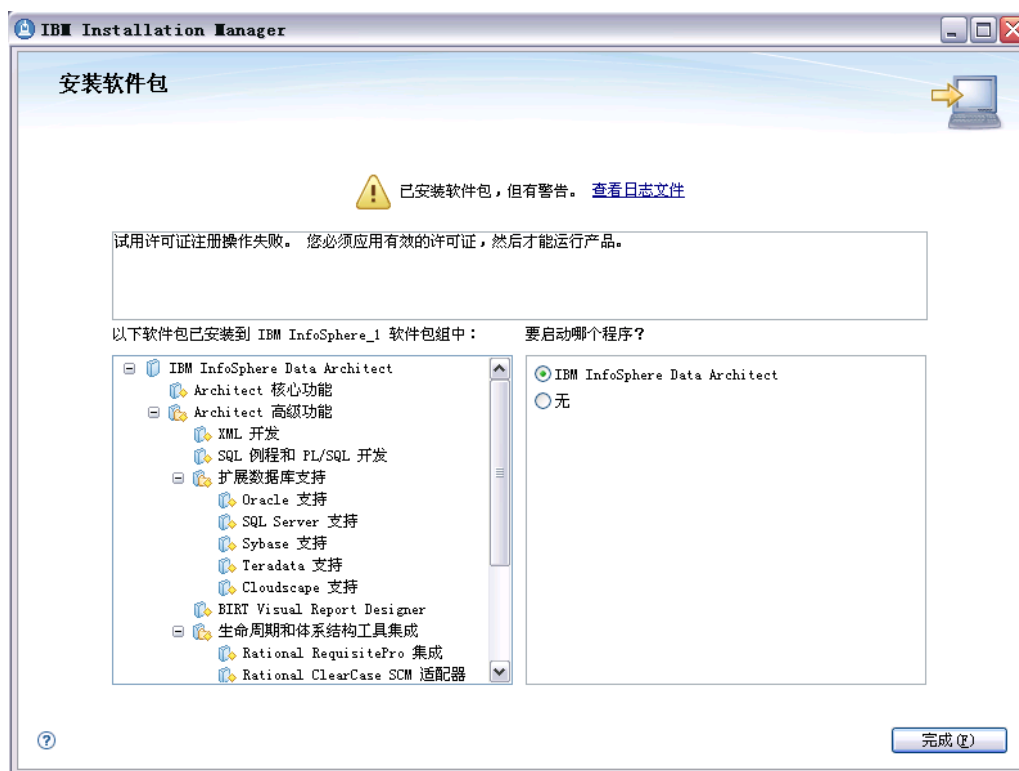


图 1.3 – 产品安装成功界面

12. 在上面的页面中，选择“无”选项，先不启动 InfoSphere Data Architect。如果您购买了产品的许可证，在启动 InfoSphere Data Architect 之前，您需要首先使用 Installation Manager 为产品应用许可证。点击“完成”退出安装界面。
13. 在 launchpad 左侧点击“退出”链接，退出 launchpad。

到此我们已经成功地安装了 IBM InfoSphere Data Architect。如果您拥有产品的许可证，您需要在试用版 30 天有效期过期之前应用许可证激活产品。

1.6 为 IBM InfoSphere Data Architect 应用许可证

InfoSphere Data Architect 支持下面三种类型的许可证：

1. **试用许可证：** 下载产品后，您可以在 30 天有效期内使用产品的所有功能。您可以使用这个许可证对产品进行评估，用以判断是否满足您的项目需求。使用试用版许可证，您不需要做任何许可证的安装操作，产品已经安装好了试用版的许可证。
2. **浮动许可证：** 您可以创建一个 Rational License Server 用来存储和管理浮动许可证。更多关于如何针对 InfoSphere Data Architect 使用浮动许可证的信息，请参考下面的文章：<http://www.ibm.com/support/docview.wss?uid=swg21468395>
3. **单一许可证：** 这个许可证指的是您可以使用产品激活工具箱为每一次的产品安装使用。下面是详细地介绍： .

如果您使用的是浮动许可证或者单一许可证，可以按照下面的步骤应用许可证：

1. 打开 IBM Installation Manager:
 - Windows 操作系统： 点击“开始” -> “所有程序” -> *IBM Installation Manager* -> *IBM Installation Manager*。
 - Linux 或 UNIX 操作系统： 打开一个命令行窗口并获取 root 权限，然后将当前目录改变到 IBM Installation Manager 的安装目录。默认情况下，这个目录是 `/opt/IBM/InstallationManager/eclipse`，运行 **IBMIM** 程序。
2. 点击“管理许可证”按钮，如图 1.4 所示：



图 1.4 – 在 IBM Installation Manager 中选择管理许可证

管理许可证页面将会打开。

3. 导入 IBM InfoSphere Data Architect 产品激活工具箱：
 - a. 从已安装的软件包中选择 *IBM InfoSphere Data Architect 7.5.3.0* 。
 - b. 选择“导入产品激活工具箱”，然后点击“下一步”。
 - c. 从本地硬盘上选择保存的扩展名为 **.JAR** 的产品激活工具箱。如图 1.5 所示：

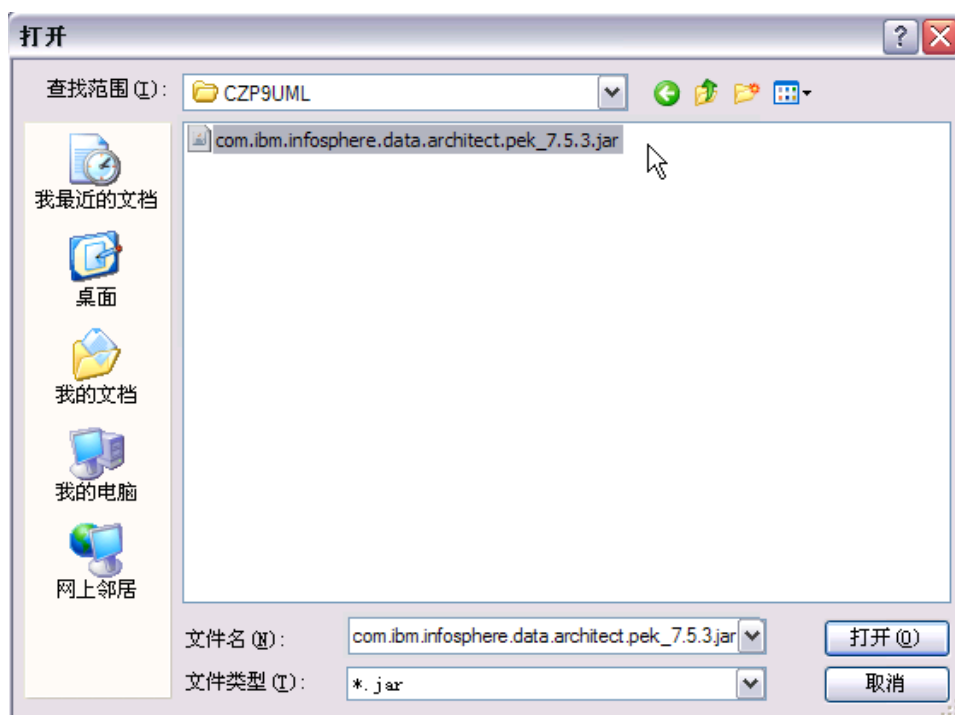


图 1.5 – 选择产品激活工具箱

- d. 点击“下一步”。
- e. 接受许可证协议，并点击“下一步”。
- f. 最后到摘要页面，点击“完成”。

到此，许可证已经成功地导入，如图 1.6 所示：

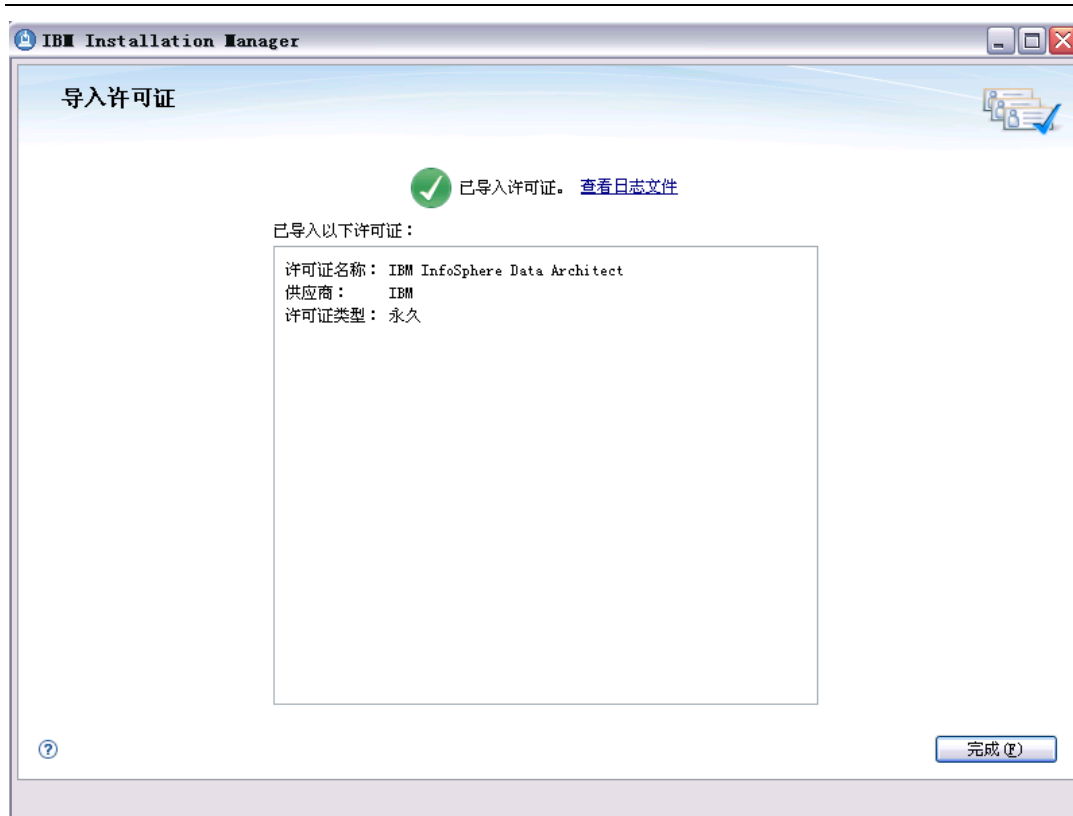


图 1.6 – 许可证安装完毕

现在您可以关闭 IBM Installation Manager 并首次启动 IBM InfoSphere Data Architect。

1.7 运行 IBM InfoSphere Data Architect

可以通过下面的方法之一运行 IBM InfoSphere Data Architect:

- Windows 操作系统： 点击“开始” -> “所有程序” -> *IBM InfoSphere* -> *IBM InfoSphere Data Architect 7.5.3.0*。
- Linux 或 UNIX 操作系统： 打开一个终端命令行，输入如下命令：
`.product_install_directory/eclipse`

命令中的 *product_install_directory* 指的是 InfoSphere Data Architect 的安装目录。

当产品启动时，您可以指定产品的工作空间。这里我们使用 **IDA_GettingStarted** 作为工作空间的名称，如图 1.7 所示：

注意：

工作空间是用来保存您的项目、个性化定制和首选项的位置。您在一个工作空间中的更改在另一个工作空间中不可见。工作空间这个概念来自于 **Eclipse**。

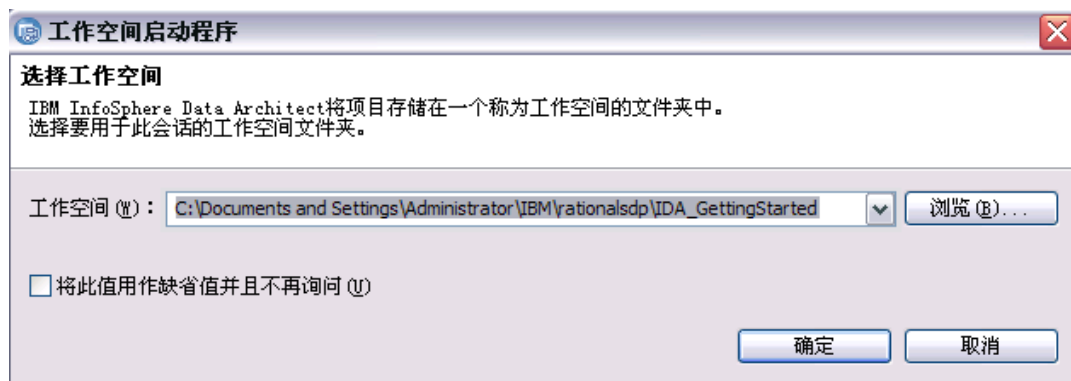


图 1.7 – 选择工作空间

第一次打开 IBM InfoSphere Data Architect 时会自动为您切换到数据透视图，它包含一个任务启动程序。

注意：

透视图的概念来自于 **Eclipse**。一个透视图包含与特定任务相关的多个视图和多个操作。视图用来方便显示和查看资源，它和编辑器是相互关联的。IBM InfoSphere Data Architect 默认的透视图是数据透视图，如 [图 1.8](#) 所示。您还可以看到它所包含的多个视图，如数据项目资源管理器 and 大纲视图。

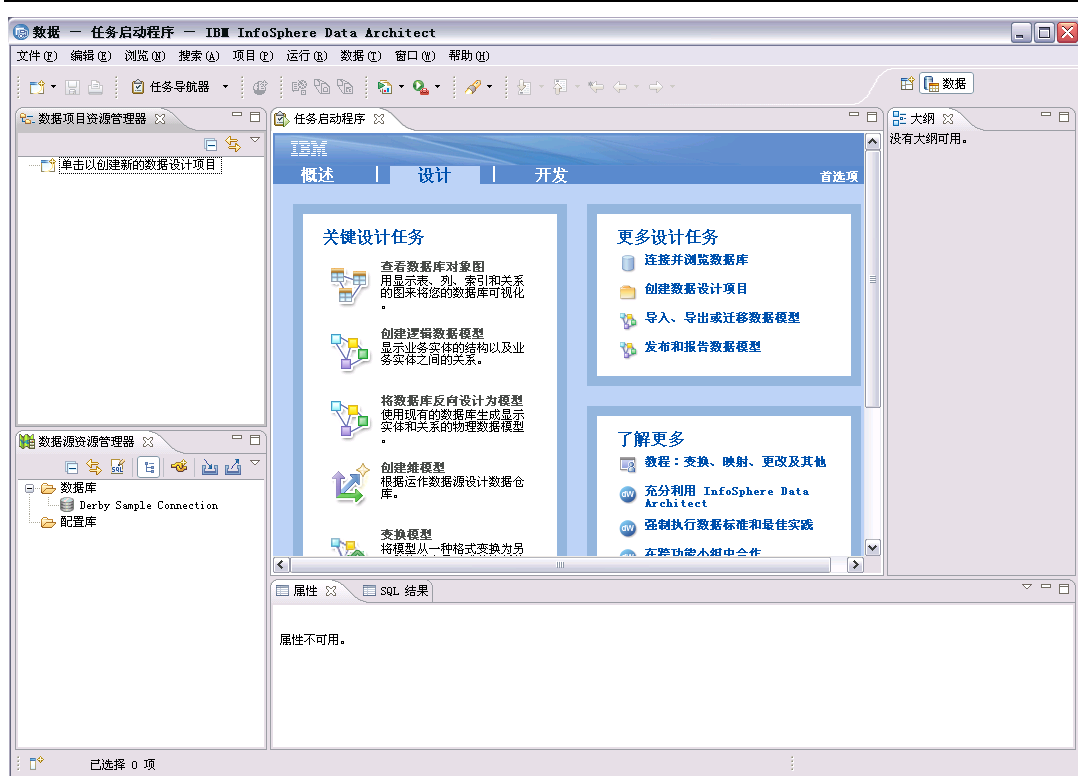


图 1.8 – IBM InfoSphere Data Architect 默认的数据透视图

1.7.1 工作台简介

工作台 的概念来自于 **Eclipse**，它指的是您的桌面开发环境。它为所有功能提供统一的入口，并且提供用来创建、管理和浏览工作空间中的资源的统一方式。

1.7.1.1 任务启动程序

任务启动程序是当您在一个工作空间中第一次启动 **InfoSphere Data Architect** 时，在编辑器中显示的内容。

任务启动程序用来帮助您进行一些基本的操作，例如您可以使用任务启动程序创建一个物理数据模型。

您可以通过任务启动程序顶端的选项卡进行不同的任务页面之间的切换，**IBM InfoSphere Data Architect** 的任务启动程序包含下面三个选项卡：

- “概述”：这个选项卡包含产品的一些概要介绍信息和初始任务，如连接并浏览数据库。
- “设计”：这个选项卡包含创建和设计数据模型需要的一些通用任务。您可以通过 **developerWorks** 上的教程学习和了解更多关于这些通用任务的内容。

- “开发”：这个选项卡包含进行 SQL、XQuery 或者存储过程开发的一些通用任务，从中您可以学习到如何对这些语句进行调试。

任务启动程序的界面如图 1.9 所示：



图 1.9 – 了解任务启动程序

在任务启动程序中启动一个任务的时候，与此任务相关的透视图将会自动被打开，并且相关的帮助信息也会显示在帮助视图，您可以按照帮助信息中的提示完成任务。

让我们练习一下任务启动程序中的连接并浏览数据库任务。

在任务启动程序视图中打开“概述”选项卡。

1. 在“任务”一栏，点击“连接并浏览数据库”链接。如果当前不是在数据透视图，将会自动切换到数据透视图，并且会自动打开帮助视图对这个任务进行描述。如图 1.10 所示：

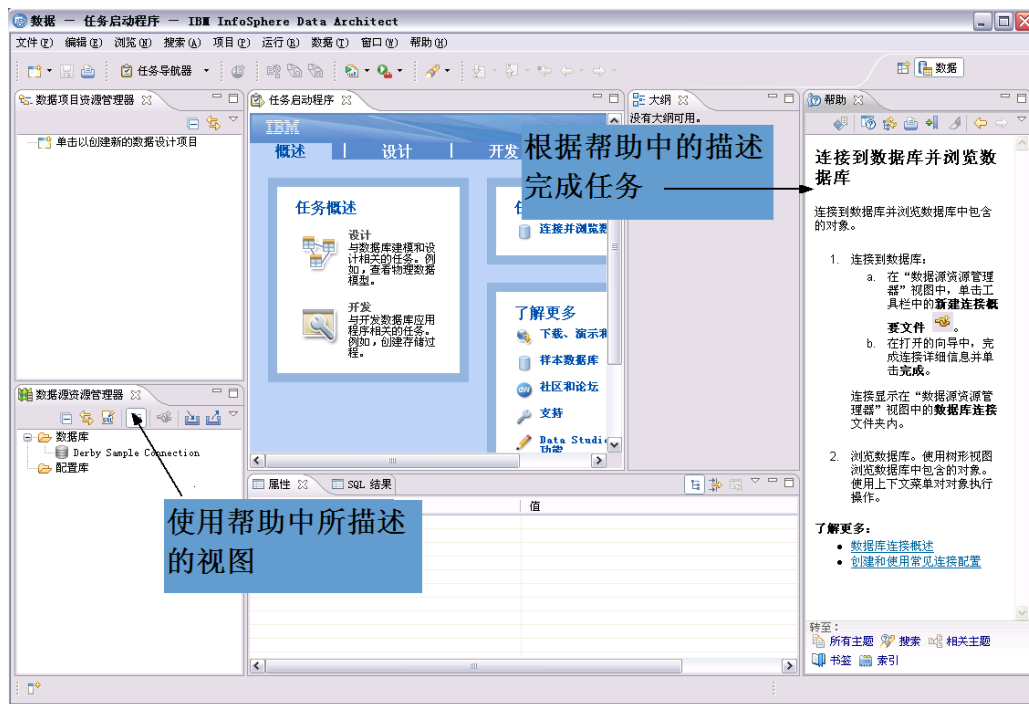


图 1.10 – 在任务启动程序中完成一个任务

2. 参考帮助视图对任务的描述来完成并连接到您之前创建的 SAMPLE 数据。SAMPLE 数据库是安装 DB2 Express-C 时创建的。

当您完成这个任务后，数据库连接将会在数据源资源管理器中显示。

在所有的基于 Eclipse 开发，并且和 Data Security, Privacy, 和 Lifecycle Management 相关的系列软件中，都会有任务启动程序。如果您的产品和这类产品进行 shell-share，任务启动程序中的任务或者选项卡将会相应的改变。我们会在第 10 章对 shell-sharing 以及 Data Security, Privacy, 和 Lifecycle Management 的系列软件进行更详细地讲解。

注意：

如果在产品启动后不想看到任务启动程序，您可以将它关闭。如果想再次打开任务启动程序，可以通过菜单：“帮助” -> “Optim 任务启动程序” 将其打开。

1.7.1.2 透视图

每一个工作台窗口都包含一到多个透视图。透视图包含视图和编辑器以及相应的工具栏和菜单。所以，在 Java 的调试透视图和数据透视图您会看到不同的视图和任务。

让我们看看 Java 的透视图。

可以通过如 图 1.11 所示的方法快速进行透视图切换，也可以通过调用菜单：“窗口” -> “打开透视图” 来进行透视图切换。

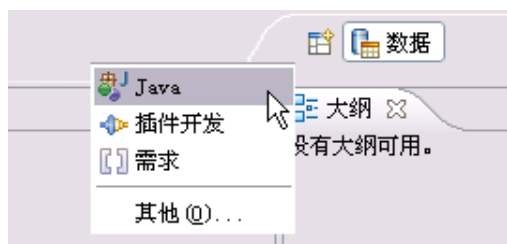


图 1.11 – 打开 Java 透视图

通过对比图 1.8 和图 1.12 我们可以看到，Java 透视图和数据透视图是为不同的任务提供服务的，Java 透视图主要关注 Java 的开发。例如，Java 透视图中的大纲视图是用来展现编辑器中 Java 类的大纲，资源管理器视图展现的是 Java 中的包而不是数据库对象。

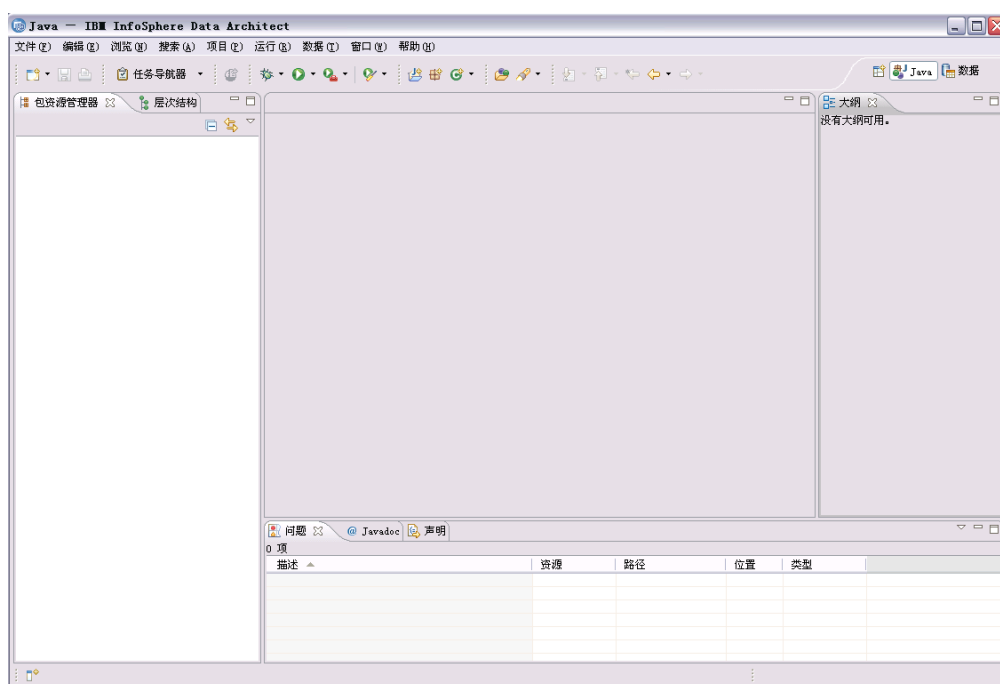


图 1.12 – Java 透视图

我们切换回数据透视图了解关于它的更多的内容。

注意：

关于透视图和视图的更多信息可以参考电子书 *Getting Started with Eclipse*。

1.7.2 数据透视图和相关视图介绍

请确保将工作空间切换到数据透视图，本书中大多数练习将会在这个透视图完成。

如同我们之前介绍的，视图就是我们在工作台看到的窗口，例如数据源资源管理器和属性视图。视图可以被用来浏览信息的层次结构、打开一个编辑器或者显示当前编辑器对应信息的属性等。对视图的改变（如视图的位置和大小）和对资源的改变都将会保存在您的工作空间中。

对数据透视图中的包含视图的更多介绍可以参考以下信息，如表 1.4 所示。

视图	描述
数据项目资源管理器	数据架构师和数据建模师在创建模型时会用到这个视图，它用来显示数据设计项目（用于存储物理数据模型和逻辑数据模型）和数据开发项目。
数据源资源管理器	这个视图允许您连接到数据库并浏览其内容。它不但支持自动发现已经连接的数据库，还支持连接到一个新的数据库。在这个视图中，您可以通过反向设计将数据库中的对象拖拽到数据项目资源管理器中，从而创建一个物理数据模型。
属性	这个视图用来显示在当前工作空间中被选中对象的属性。对于一些对象，您还可以通过这个视图对其属性进行更改，例如可以对数据项目资源管理器中被选中数据对象进行属性的更改。当在数据项目资源管理器或者图中选中某一对象，属性视图将会显示该对象的属性。
SQL 结果	这个视图用来显示 SQL 或者 XQuery 语句执行的结果。您也可以通过这个视图获取并查看数据库的某一个表内的数据。
大纲	这个视图用来结构化地显示在当前编辑器中打开的资源。例如，编辑一个数据图时，这个视图将会显示图的概述。当您对该数据图进行放大和缩小操作时，对应的大纲视图会对当前图在编辑器中的可见范围进行阴影化着重显示，从而方便确认当前操作部分在整个图中的位置。

表 1.4 – 数据透视图中的视图

1.7.3 视图支持的操作

视图支持的基本操作如图 1.13 所示。



图 1.13 – 视图的操作

可以通过点击视图右上角的 **X** 来关闭一个视图，如 [图 1.13](#) 所示。如果操作失误关闭了一个视图，您可以通过菜单“窗口”->“显示视图”并选择您希望打开的视图来重新打开它。如果在列表中找到需要的视图，可以点击“其它”来进行进一步的选择。

1.7.4 复位透视图的默认设置

您可以在工作台中对视图和透视图进行操作，如果您不熟悉 **Eclipse**，可能会对打开和关闭视图比较陌生，这时候您可以通过点击菜单：“窗口”->“复位透视图”来还原透视图的默认设置。

注意：

复位透视图只对当前的透视图起作用。如果您想为其它的透视图恢复其默认设置，您可以通过点击菜单：“窗口”->“首选项”，然后选择“常规”->“透视图”，在界面中选择您需要复位的透视图，然后点击“复位”按钮。这样，当被复位的透视图被再次打开的时候，将会使用它默认的设置和布局。

1.8 练习

在本章的练习中，我们将会练习 **IBM InfoSphere Data Architect** 的安装以及熟悉工作台和 **Eclipse** 工具的基本功能。

1. 如果您还没有安装 **DB2 Express-C**，请先安装。
2. 如果您还没有安装 **InfoSphere Data Architect**，请参考本章的介绍进行安装。
3. 您可以通过下面练习熟悉工作台的使用：
 - 切换到插件开发透视图。
 - 切换回数据透视图。
 - 关闭大纲视图。
 - 最小化和最大化一些视图窗口。
4. 浏览产品文档，关于 **IBM InfoSphere Data Architect** 的产品文档可以在如下链接中找到：
<http://publib.boulder.ibm.com/infocenter/rdahelp/v7r5/index.jsp>
5. 阅读[产品概览](#)。
6. 观看并学习[了解工作台环境](#)教程。
7. 在 **developerWorks** 上观看[Introduction to IBM InfoSphere Data Architect](#) 视频介绍。

1.9 总结

在数据建模的过程中，IBM InfoSphere Data Architect 可以帮助您进行数据的发现、可视化展示、关联以及标准化等工作。您可以为多种数据库如 DB2、Informix、Oracle、Sybase、Microsoft SQL Server、MySQL 和 Teradata 等创建逻辑数据模型、物理数据模型、多维模型和域模型。它支持迭代设计流程，您可以根据业务的需要更新和提炼数据模型，而同时对生产环境的影响降到最低。

下面是 IBM InfoSphere Data Architect 的论坛地址，这个论坛相当活跃，里面有很多针对产品使用的讨论：<http://www.ibm.com/developerworks/forums/forum.jspa?forumID=1796>

在论坛中你可以找到很多非官方的支持。

在这一章中我们还介绍了产品的安装和 Eclipse 工作台的使用。从中您了解到了如何在不同的透视图之间进行切换和如何在一个透视图操作中操作视图。

1.10 本章回顾

1. IBM InfoSphere Data Architect 是基于哪一个开源的产品构建的？
2. 在基于 Eclipse 的产品中，透视图是什么？
3. IBM InfoSphere Data Architect 默认的透视图是什么？
4. SQL 操作的执行结果会在哪一个视图中显示？
5. 在一个透视图，如何打开一个视图？
6. 正误判断：可以在数据项目资源管理器视图中查看数据库连接。
7. 正误判断：每一个工作空间的默认透视图，项目和数据对象都是唯一的。
8. 正误判断：在产品安装的时候，你不能指定哪些功能部件随产品一起安装。
9. 正误判断：在产品安装的时候，如果使用扩展已有的 Eclipse 的方式，需要确保已有 Eclipse 和 JDK 的版本与产品的一致。

2

第二章 – 数据建模概述

在创建数据模型时，我们通常需要遵循一个特定的流程。本书介绍了数据建模的概念和数据建模过程中的最佳实践，保证您在工作台中创建数据模型时能够遵循良好的数据设计原则。

在本章中，您将会了解到如下内容：

- 数据模型设计的生命周期
- 如何组织数据模型
- 结合本书实例，创建一个信息管理系统

注意：

关于数据建模的更多信息，请参阅本系列丛书之一 [Database Fundamentals](#)

2.1 数据模型设计的生命周期

通过本书，您将学到如何创建数据模型并基于数据模型创建数据库。数据模型的设计是一个反复迭代的过程，数据模型设计生命周期将帮您在迭代设计流程中创建概念模型并进行数据模型的开发设计。

数据建模有两种方式：

- **正向设计：**不依赖任何已有的模型，从无到有创建数据模型和数据库。这种建模方法适用于以下场景：
 1. 对新的系统进行建模；
 2. 在当前的系统中，对新的业务流程进行建模；
- **反向设计：**基于已有的数据模型和数据库进行建模。通过 **IBM InfoSphere Data Architect**，您可以根据已有的数据库和数据库模式来创建物理数据模型，然后将该物理数据模型转换为逻辑数据模型，以进行进一步的优化。

注意：

本书侧重于介绍正向设计。但是为了帮助您对数据建模和 InfoSphere Data Architect 的产品功能有一个全面的认识，本书同样花了些篇幅来介绍反向设计。

关于反向设计的更多信息请参阅本书第 7 章。

在 IBM InfoSphere Data Architect 工作台上开发和部署数据库，通常需要遵循以下步骤：

1. **收集项目需求。**和业务流程所有者讨论并决定哪些信息在业务层面上是至关重要的。在设计实体或表，属性或列时通常会使用这些信息。
2. **使用 IBM InfoSphere Data Architect 创建逻辑数据模型。**创建逻辑数据模型是数据建模的第一步，建模人员基于收集到的业务需求来创建实体和属性。然后基于逻辑数据模型来创建物理数据模型。
在设计逻辑数据模型的过程中，您可以使用数据图作为辅助手段，对模型进行可视化展示，使得业务流程所有者能够清楚地了解到数据是如何连接组织到一起的。在与业务流程所有者的进一步讨论过程中，您可以继续对该数据模型进行优化，对数据模型的设计进行改进，直到所有的业务信息都清晰完整地包含在模型中。
3. **使用 IBM InfoSphere Data Architect 将逻辑数据模型转换成物理数据模型。**在设计物理数据模型时，建模人员需要定义物理存储，表空间，索引或者视图。另外，建模人员可以通过在物理数据模型中添加角色和用户，来定义数据库的安全策略。在对物理数据模型进行优化时，建模人员可以在已有的数据图上直接进行修改，也可以创建一个新的数据图来对存储需求进行可视化建模。
4. **创建映射模型，保存数据模型和目标数据库之间的映射关系。**在映射模型中您可以定义源模式和目标模式之间的关系，包含了变换，连接条件，过滤，排序条件，以及注释。映射模型能发现和记录源与目标之间的关系，并且基于映射模型您可以生成脚本来查询或者操作数据源。

注意：

最佳实践：最好在数据模型已经基本稳定后，再创建映射模型。因为在映射模型创建完成后，如果数据对象名称改变或者数据对象之间的关系发生变化，那么之前创建的映射就被破坏，导致出错。

5. **生成 DDL 并部署数据库。**当数据库部署成功后，您就可以创建应用程序来连接和读取数据库，或者使用其他软件来进一步优化数据库。
6. **根据随时变化的业务需求，调整数据模型。**业务需求经常变化，您必须对数据模型进行相应的调整以确保相关的信息被正确获取或记录。

使用 IBM InfoSphere Data Architect 还可以帮助您完成以下任务：

- 创建报表，帮助业务流程所有者更好地了解数据库里的数据。

- 导入导出数据模型。
- 比较和同步数据模型。在应用变更之前，分析变更对目标数据模型的影响，以帮助建模人员更好地管理变更。
- 从已有数据源反向设计创建物理数据模型。
- 分析数据模型，以确保模型满足通用的数据建模规则。

2.2 组织数据模型

数据建模包括对数据的构造和组织。数据建模可以定义和组织数据，并且在数据上显式或隐式地定义一些约束和限制，以确保数据的完整性。

2.3 创建学生信息管理系统

在本书中，您将学习到如何使用 IBM InfoSphere Data Architect 为某大学创建一个学生信息管理系统，该系统存储了学生 ID，成绩以及地址等信息。

我们将会遵循数据设计的生命周期来设计和实现这个系统。首先创建逻辑数据模型，然后基于逻辑数据模型创建物理数据模型。物理数据模型生成后，经过对该模型进行进一步修改、优化，最后生成 DDL 来部署数据模型。

本书的下一个部分，我们将主要关注数据设计生命周期中的数据建模。首先您将会根据业务流程所有者描述的业务需求来创建逻辑数据模型。当逻辑数据模型创建完成后，您可以将逻辑数据模型转换成物理数据模型。物理数据模型创建完成，并且表、列和其他物理数据对象设计完成后，您就可以基于该物理数据模型来生成 DDL 脚本，并创建和部署相应的数据库。最后一步，您将会创建一个映射模型，将您的数据模型和其他数据模型进行有效集成，创建一个高内聚的数据库系统，完备地记录学生各个方面的信息。

在本书的第三部分中，您将了解到如何使用 IBM InfoSphere Data Architect 生成报表。业务流程所有者可以根据这些报表来对设计完成的数据模型进行分析。最后，您将了解到如何对已有的模式或者数据库进行反向设计，对已有的数据对象进行修改，并部署这些修改。在最后一章中，您将学习如何分析数据模型以确保部署到生产环境中的模型是正确有效的。

2.4 总结

通过本章，您已经对数据建模以及如何使用 IBM InfoSphere Data Architect 来设计和部署数据模型有了基本的了解。

数据建模是一个反复迭代的过程，IBM InfoSphere Data Architect 能帮助您非常容易地在已有的数据模型或数据源上进行设计、修改。反向设计功能帮助您对已有的数据源进行建模，然后您可以对模型进行修改和扩展，比如扩展新的功能，提高性能或针对需求变更进行相应地修改。

2.5 下节展望

现在您已经了解了数据建模的基本知识，是时候去设计、创建一个数据模型了。在接下来的章节中，您将会了解到工作台的各种功能，学习如何从头开始创建一个完整的数据模型，如何优化和部署数据模型。

第二部分 – 数据建模

3

第三章 – 逻辑数据建模

逻辑数据建模是数据设计生命周期的第一步。逻辑数据模型可以帮助您清晰完整地表达所有的业务需求，还可以让您在物理存储设计和模型部署前进一步进行模式和数据库的设计。

本章包含如下内容：

- 如何设计逻辑数据模型
- 如何用图来可视化展示模型设计
- 如何为您的组织设计和定义词汇表模型和命名规则
- 如何将定义好的词汇表模型和命名标准应用到您的数据模型

注意：

关于逻辑数据建模方面的更多信息，请参阅本系列丛书之一 [Database Fundamentals](#)。

3.1 逻辑数据建模：概述

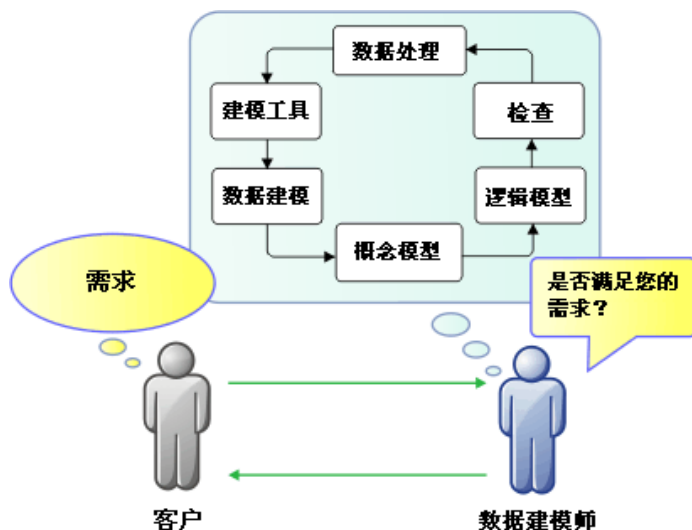


图 3.1 – 使用 InfoSphere Data Architect 作为数据建模工具

创建逻辑数据模型是设计开发数据库或者模式以满足业务需求的第一步。*逻辑数据模型*与具体的数据库厂家无关。它描述了一个业务机构希望如何收集业务数据，并且描述这些数据之间的关系。逻辑数据模型一般都是按层次结构来组织，包含诸如包，实体，属性以及其他关系对象。

在开始创建逻辑数据模型之前，您需要先完成业务需求的收集。业务用户对他们需求的描述很可能是不完整的。通常情况下，业务用户可能只是描述了所有需求的一部分。也就是说您很有可能得到的是一个对业务过程很模糊的描述，比如：“业务要求必须跟踪记录产品销售的情况”。作为数据建模专家，必须完整地理解项目所有的业务需求，这样才能构建一个完整的模型。

在您完成初始的业务需求收集以后，您就可以将设计成果展示给客户，图是一个比较合适和推荐的展示方法。然后您就可以和客户一起对现有的设计进行讨论修改，如图 3-1 所示。在讨论后，您可以对逻辑数据模型进行进一步修改，然后把修订完善后的模型展示给客户。通过这样迭代的讨论开发过程，最终就可以完成一个完整的逻辑数据模型的设计开发了。

在下面的章节，将详细介绍如何设计开发一个逻辑数据模型。

3.2 创建逻辑数据模型

在创建逻辑数据模型时，您将会遇到的一个最棘手的问题就是如何完整地把业务过程模型化。您应该和客户一起讨论决定业务流程中所有的业务需求。创建逻辑数据模型，您应该遵循下面的流程：

1. 全面分析业务各方面的需求，从最重要的业务流程开始。将分析结果记录归档，这样您和其他数据建模师都可以理解这个项目的目标和后面的计划。
2. 与项目所有者一起确定他们想要识别的数据以及数据的重要性。在此基础上，识别通用的初始数据对象，这些数据对象可能跨多个流程、数据源或者每个数据对象的不同粒度。
3. 与项目所有者协商将模糊不清的需求进一步清晰化。
4. 制定一个能够满足需求的初步计划。创建一个概念模型来起草逻辑数据模型的初始设计，然后用这个计划来设计您的数据模型。

3.2.1 使用工作台来创建逻辑数据模型

该大学的业务流程负责人让您帮助他们创建一个用来收集学生信息的学生信息管理系统。在进一步的商谈之后，您决定从小模型开始，创建一个逻辑数据模型来跟踪记录一些基本的学生信息，例如学号，成绩和班级。

接下来我们将通过创建一个小的逻辑数据模型 **STUDENT_INFO_SYSTEM** 来学习如何使用工作台创建逻辑数据模型。

3.2.1.1 创建数据设计项目来存储模型

在开始创建一个逻辑数据模型之前，您需要创建一个新的数据设计项目来存放文件。这个数据设计项目将会被用来存放与这个项目相关的所有数据模型。在一个工作空间中，您可以创建多个数据设计项目，并且一个数据设计项目可以包括多个数据模型。

创建一个数据设计项目：

1. 点击“文件”->“新建”->“数据设计项目”，新建数据设计项目向导将打开。

注意：

因为工作空间是空的，您可以通过点击链接“点击创建新的数据设计项目”。一旦您创建了第一个数据设计项目，这个链接将不可见。

2. 在“项目”字段输入名称，这里我们输入 **University Info System**。
3. 点击“完成”。

这样，数据设计项目 **University Info System** 就成功地在数据项目资源管理器中被创建好了。数据设计项目中有很多文件夹可以存储项目文件。

3.2.1.2 创建您的第一个逻辑数据模型

现在，这个项目还是空的，您需要创建一个新的逻辑数据模型。然后您将用这个逻辑数据模型来创建学生信息管理系统。

创建一个新的逻辑数据模型：

1. 打开新建逻辑数据模型的对话框：
 - a. 在数据项目资源管理器视图中，展开之前创建的数据设计项目 **University Info System**。

- b. 右键单击“数据模型”文件夹，然后选择“新建”->“逻辑数据模型”，打开“新建逻辑数据模型”对话框。
 2. 在新建逻辑数据模型对话框中，完成以下设置：
 - a. 在文件名字段中，输入 **STUDENT_INFO_SYSTEM**。
 - b. 点击“完成”。

这样逻辑数据模型就创建完成了，数据模型处于打开状态，并且包括一个空白图。通过图编辑器，我们可以可视化地创建逻辑数据模型。图 3.2 展示的就是您新创建的数据设计项目和新创建的空的逻辑数据模型。

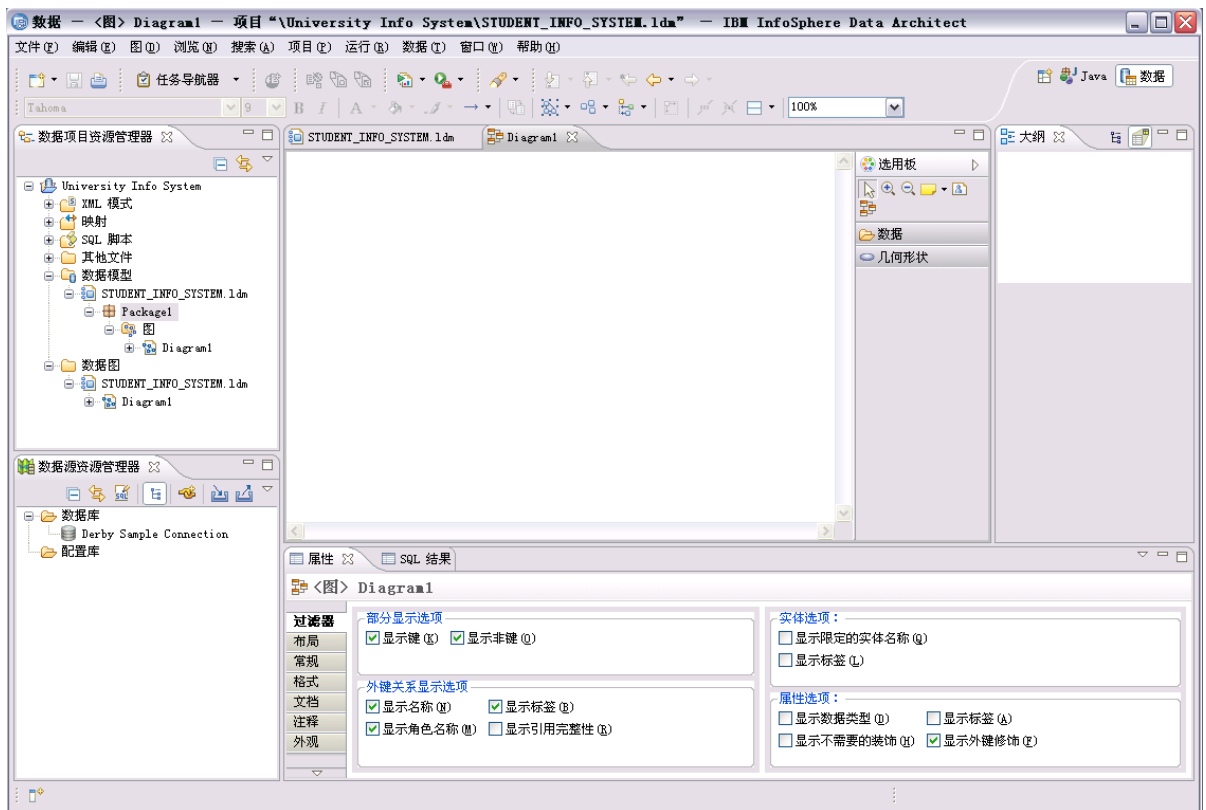


图 3.2 – 创建空白逻辑数据模型

3.2.2 使用图来创建实体

要对一个逻辑数据模型进行操作，您可以通过对模型的可视化展示也就是图来进行操作。一个逻辑数据模型可以包括多个图，每个图都是逻辑数据模型的某一部分的可视化展示。

3.2.2.1 创建图

您可以使用图来可视化展示和编辑数据设计项目中的对象。

您在逻辑数据模型所包含的图中创建对象所做的更改将会自动地传播到逻辑数据模型本身，这样不但让数据对象和数据对象之间关系的可视化展示变得更加容易，而且节省手动创建这些关系的时间。

数据图是数据模型的可视化展示。您可以创建数据图，而图中只包含您感兴趣的模型对象。

3.2.2.2 选用板的数据区

选用板的数据区域包含了数据模型对象。当您通过图选用板或者图来增加或者修改数据模型对象的时候，底层的数据模型也会被相应的修改。

3.2.2.3 使用属性视图来对数据对象重命名

默认情况下，*Diagram1* 是图默认的名字，但是您可以把它修改成您觉得更容易记住或者可以区别于数据设计项目中其他数据模型的名字。您可以参照以下步骤对图和顶级包进行重命名：

1. 在图编辑器中打开图 *Diagram1*。当您打开或者选中任意对象时，这个对象的属性就会在属性视图中展示出来。现在您就可以对图 *Diagram1* 的属性进行编辑了。
2. 打开属性视图中的“常规”选项卡，然后在图名称字段中输入 **STUDENT_INFO**。
3. 在数据项目资源管理器中选中包 *Package1*，您可以在数据模型文件 **STUDENT_INFO_SYSTEM.ldm** 下找到这个包。
4. 打开属性视图中的“常规”选项卡，然后在名称字段中输入 **STUDENT INFO SYSTEM**。

现在数据对象被自动重命名。当你点击图中的空白区域时，图编辑器的名字会变成 **STUDENT_INFO**，数据项目资源管理器中该逻辑数据模型也会被相应地更新。数据对象被修改后，在数据项目资源管理器和对应的数据对象编辑器旁边都将有一个星号 (*) 字符。星号字符在您保存文件或者项目后会消失。

注意：

您可以在属性视图中修改图的显示特征。例如您可以修改以下选项：

- 使用格式选项卡来修改实体或外键的前景色或背景色，修改隐含外键关系线的颜色。
- 使用文档选项卡来记录图的目的是用途。
- 使用外观选项卡来定义字体和文本颜色。

这些选项只会修改图本身的属性，而不会修改底层数据模型。您也可以把图保存为图像文件以用于存档或者演示。

3.2.2.4 给图添加注释和文本框

给图添加注释或者文本框可以帮助您记录数据模型的有关信息。注释和文本框在提取信息或者充分解释复杂图时非常有用。

首先，我们先为图添加一个注释来标注该图。

1. 在图的选用板上，点击“注释”图标，如图 3.3 所示。

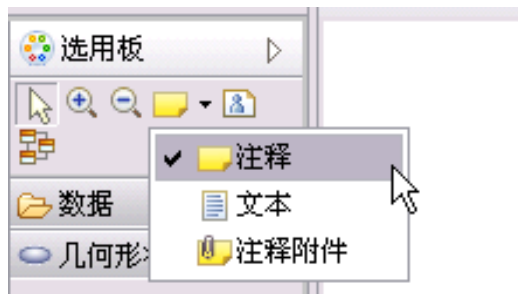


图 3.3 – 点击注释图标

2. 点击图编辑器的空白区域，注释对象就会出现在图中。
3. 在注释对象中，添加以下注释：

STUDENT_INFO: All information relating to student life.

这样，注释就添加到图中了，如图 3.4 所示。

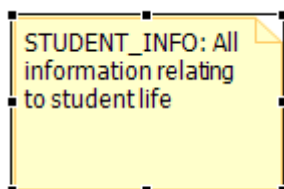


图 3.4 – 在图中创建注释

为了给数据图添加更多的附加说明，您可以添加文本框。让我们添加一个文本框来提醒用户这个逻辑数据模型还处于起草状态。

1. 右键单击图上的一个空白区域，然后选择“添加”->“文本”，一个文本框就会出现在图上。
2. 在文本框中输入：
NOTE: This logical data model is still being drafted.
3. 让文本显示为红色粗体：
 - a. 打开属性视图的“高级”选项卡。
 - b. 展开“样式”属性。
 - c. 将“粗体”属性设置为 *true*。

d. 将字体颜色设置为红色。

4. 点击“文件”->“全部保存”，保存您刚才的工作。

3.2.2.5 设计数据图和逻辑数据模型

实体是一个逻辑数据模型对象，用来存储有意义的业务信息。每一个实体都包含属性，属性用来存储实体的详细信息。例如，您可以创建一个 **Sales** 实体，包括 **ProductKey**，**Quantity**，**UnitPrice** 和 **OrderKey** 属性，这些属性用来描述不同产品的销售情况。当把逻辑数据模型转换为物理数据模型的时候，实体会转换为表，而属性会转换为列。

在创建实体和属性的时候，实体和属性越具体越好。添加所有需要的属性，所有的域约束以及您所能考虑到的其他方面，这些能帮助您创建一个和现实世界最接近的模型。记住逻辑数据模型将会被转换为物理数据模型。

接下来，我们将使用数据图来为逻辑数据模型添加一个新的实体：

1. 在“选用板”上，展开“数据”选项卡，然后拖拽一个实体到图编辑器的空白区域。这样一个新的实体就成功地在模型和图中被创建出来，您可以立即给这个实体一个新的名字。
2. 将实体的名字修改为 **STUDENT**。

这个实体在图中被创建成功，同时相应的数据对象也在逻辑数据模型中被成功创建。当你通过选用板或者图来增加或者修改逻辑数据模型对象时，底层的逻辑数据模型也会被相应地修改。

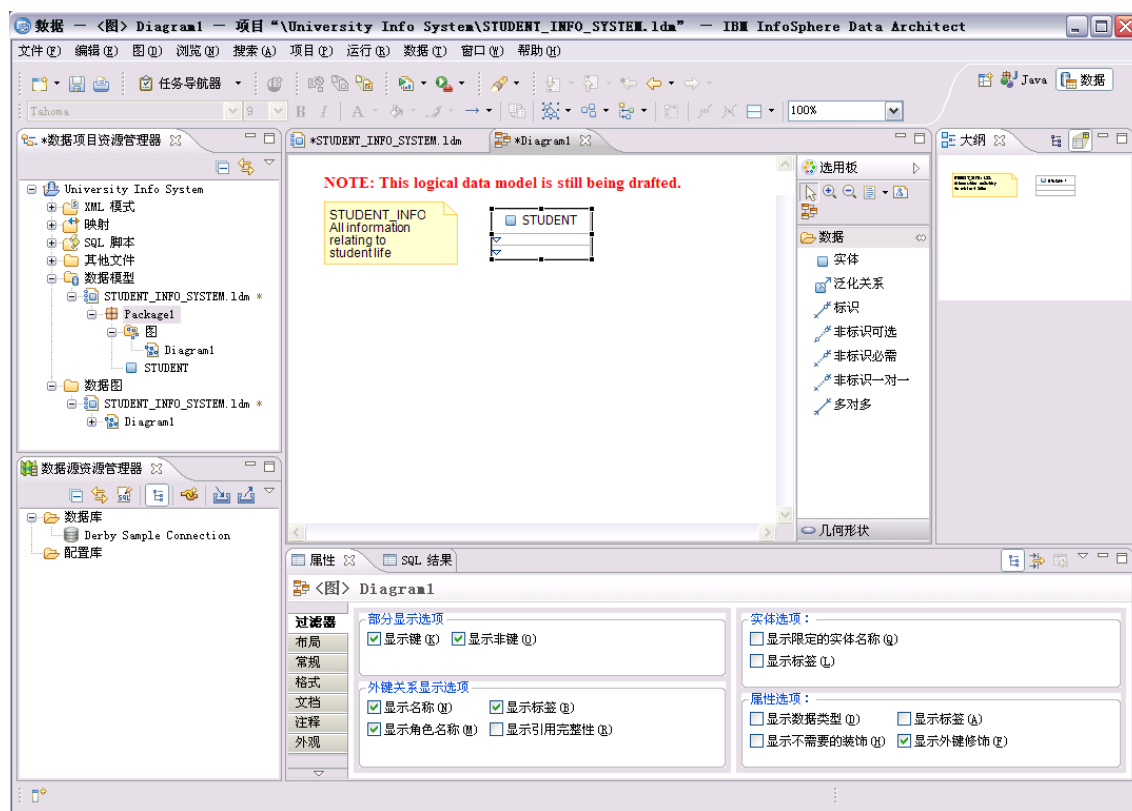


图3.5 – 创建STUDENT实体

您也可以通过以下两种方式来向图中添加实体：

- 使用上下文菜单：您可以通过右键单击图编辑器来向图中添加数据对象。要添加一个新的实体，您可以右键单击编辑器的空白位置，然后选择“添加数据对象”->“实体”。

您也可以通过这种方法为已有的数据对象添加其他数据对象；例如，您可以用这种方法为实体添加属性。

- 使用弹出菜单创建实体：您可以使用图编辑器的空白位置来向图中添加数据对象。当鼠标悬浮在某一个数据对象或者图中的空白领域时，一个弹出菜单条就会出现，如图 3.6 所示：



图3.6 – 使用弹出菜单条在图中添加数据对象

您也可以通过这种方法为已有的数据对象添加其他的数据对象；例如，您可以用这种方法为实体添加主键。

3.2.2.6 为实体 STUDENT 添加属性

这个实体目前还是空的，我们可以为它添加一些属性。实体的属性可以包括主键和非键属性。所有这些属性将会被用来存储大学里每个学生的信息。

让我们从实体 STUDENT 开始。在大学里，每个学生都有一个十个字符长的唯一 ID。您可以创建一个属性来存储学生 ID 信息。要给 STUDENT 实体添加属性，您可以执行以下步骤：

1. 在图 STUDENT_INFO 中选中 STUDENT 实体，属性视图打开并显示 STUDENT 实体的属性。
2. 打开属性视图中的“属性选项卡”。
3. 添加学生 ID 属性(STUDENT ID)到 STUDENT 实体。
 - a. 点击“新建”按钮 (◆)。一个新的属性就会在属性视图和图中被创建。
 - b. 设置这个新属性的属性值如下：
 - “名字”： STUDENT ID
 - “主键”：选中该复选框
 - “类型”： CHAR
 - “长度/精度”： 10

这样您就已经成功创建好了属性 STUDENT ID。STUDENT ID 的数据类型是 CHAR(10)，这样它就有足够的长度来存储这所大学里学生的 ID 信息。当您选中“主键”列的复选框来标示 STUDENT ID 是主键的时候，“必需”属性也会被自动选中，因为主键不能是空值。

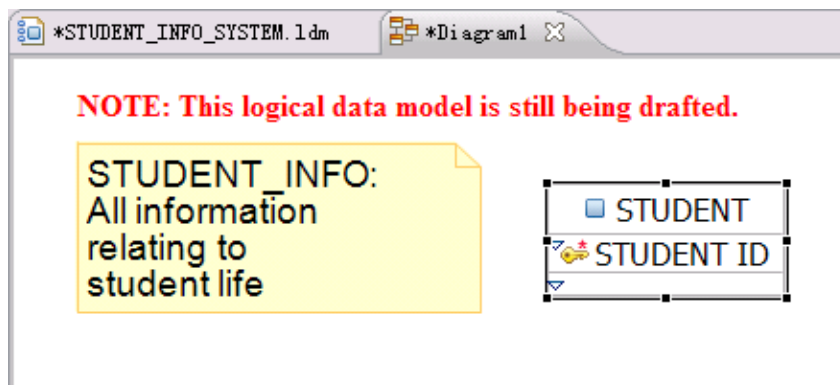


图3.7 – 创建属性STUDENT ID

注意：

如果您不知道应该选择哪种数据类型，可以参考以下原则：

- 如果一个属性或者列是为数学计算而设计的，则可以使用一个数字数据类型。
- 考虑如下因素，对于其他属性可以使用文本数据类型：

- 文本数据类型更加灵活，比如您可以使用约束来减少数据的存储空间。
- 您可以更好地控制文本数据类型的输入，因为 DB2 为字符串数据类型提供了更多的函数。

然而，有些时候您希望主键同时也是数字的，这个时候您应该使用一个数字数据类型。比如，代理键也是数字的。

注意当您往实体添加属性的时候，图也同时被更新了。在图中，属性 **STUDENT ID** 现在被显示在 **STUDENT** 实体的主键区域，而 **STUDENT ID** 的旁边有一个图标来标示它是这个实体的主键。

按照上面的步骤，我们来给 **STUDENT** 实体添加其他属性。将下面属性添加到 **STUDENT** 实体中：

4. 添加 *FIRST NAME* 属性，数据类型为 *VARCHAR(20)*，而且是必需的。
5. 添加 *LAST NAME* 属性，数据类型为 *VARCHAR(50)*，也是必需的。
6. 添加 *START DATE* 属性，数据类型为 *DATE*。
7. 添加 *GRADUATION DATE* 属性，数据类型为 *DATE*。
8. 保存。

至此，您已经学会了如何使用图编辑器来创建一个完整的实体，如 *图 3.8* 所示。接下来我们开始学习如何使用数据项目资源管理器来创建另一个实体。

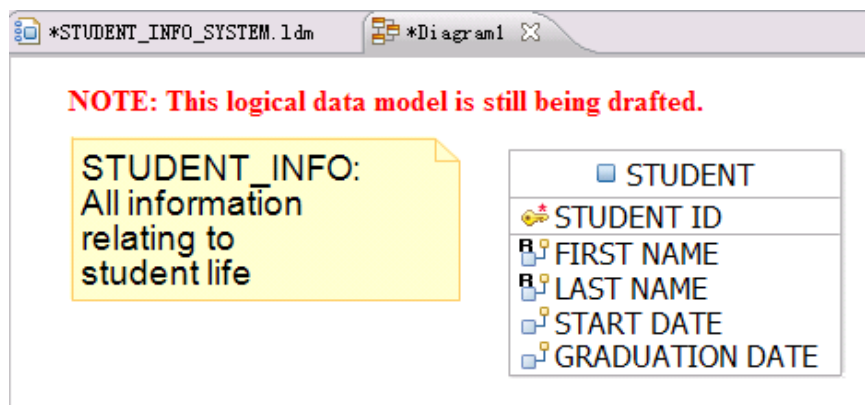


图3.8 – 完成实体STUDENT

3.2.2.7 在数据项目资源管理器中创建实体

接下来我们将使用数据项目资源管理器视图来创建一个新的实体 **GRADE** 用来存储学生的成绩信息。

在数据项目资源管理器视图中创建实体：

1. 在数据项目资源管理器中依次展开文件夹 *University Info System* -> “数据模型” -> *STUDENT_INFO_SYSTEM.Idm* 找到模型 *STUDENT_INFO_SYSTEM.Idm*
2. 创建 **GRADE** 实体：
 - a. 右键单击逻辑数据模型下的 *STUDENT INFO SYSTEM* 包，然后选择“添加数据对象” -> “实体”。
 - b. 在数据项目资源管理器中选中新建的实体 **GRADE**，然后在属性视图中打开“常规”选项卡，在名称和标签字段里输入 **GRADE**。

这样，这个实体就在包 *STUDENT INFO SYSTEM* 下成功地创建了。注意这个实体并没有显示在图编辑器中。这是因为您没有在图中创建它。只有在图中创建或者拖拽到图中的那些对象才会显示在图编辑器中。为了更好地可视化展示数据对象之间的关系，可以把这个实体添加到图中。

1. 把新创建的实体添加到图中：
 - a. 在数据项目资源管理器中选中 **GRADE** 实体。
 - b. 将这个实体从数据项目资源管理器中拖到图编辑器的一个空白区域。当您释放鼠标时，这个实体就会被添加到图中。
2. 选中 **GRADE** 实体。
3. 为 **GRADE** 实体添加以下属性：
 - **STUDENT ID**：设置和之前 *STUDENT* 实体的 *STUDENT ID* 同样的属性值。
 - **COURSE ID**：设置数据类型为 *CHAR(8)*，这个属性同样也是主键的一部分。
 - **SCORE PASSFAIL**：设置数据类型为 *BOOLEAN*，并且设置为必需的。
 - **SCORE TOTAL**：设置数据类型为 *DECIMAL*，长度/精度值为 *0*，小数位为 *3*，并且设置为必需属性。

这样您就成功地创建 **GRADE** 实体。在一些情况下，我们需要使用组合键。组合键就是由一个以上的属性组成的主键。组合键能保证您的主键是唯一的，不会有重复的值。例如，为了保证 **GRADE** 实体唯一，如图 3.9 所示，我们使用了一个组合主键：**STUDENT ID** 和 **COURSE ID**。



图3.9 – 在属性视图中创建组合键

注意:

当您从图中删除数据对象时，它们并没有从数据模型中删除。您可以使用数据项目资源管理器来把这些数据对象从您的数据模型中删除。

3.2.3 添加关系

现在已经成功地在图中创建了两个实体，接下来您需要指定这两个实体之间的关系。因为这个成绩信息是关于学生的，**STUDENT** 实体就是父实体。

前面我们介绍了如何在图中创建实体，同样地您也可以使用图编辑器来指定数据对象之间的关系。

关系包括标识的和非标识的两种。在标识的关系中，其中子实体是依赖实体。依赖实体是指这个实体的存在依赖于另一个实体。在非标识的关系中，实体之间是相互独立的（也被称为强关联集）。

在实体 **STUDENT** 和 **GRADE** 之间创建一个非标识可选的关系：

1. 显示每个实体的键和非键属性：
 - a. 点击图中的空白区域，图的属性显示在属性视图中。
 - b. 选中“过滤器”选项卡，然后选中复选框“显示键”和“显示非键”。
2. 通过图创建一个从实体 **STUDENT** 到实体 **GRADE** 的非标识可选关系：
 - a. 在选用板的“数据”选项卡中，选中“非标识可选”数据对象。
 - b. 在图编辑器中，选中实体 **STUDENT**，然后将鼠标拖到 **GRADE** 实体并放开鼠标。
 - c. 因为属性 **STUDENT ID** 在两个实体中都存在，因此您会被询问将如何处理这个属性。在这个例子中，您需要选择选项“替换为迁移的 FK 属性/列”来确保两个属性是完全一样的。

这样就在模型和图中成功地创建了这个关系。需要记住的是，图仅仅是模型元数据的可视化展示，因此这个关系在逻辑数据模型中已被创建。接下来我们需要描述这个关系。

当您在“描述”这个关系时，您可以用动词短语来描述这些实体在现实世界中是如何互相交互的。在我们的例子中，**STUDENT** 可能拥有多个 **GRADE**，因为他们可能选修了多门课程。同样的，一个 **GRADE** 只能给一个班的一个 **STUDENT**。因此，**STUDENT** 和 **GRADE** 之间关系的基数是一个或者多个。

3. 使用属性视图来描述关系：
 - a. 如图 3.10 所示，在数据图中选中关系对象：

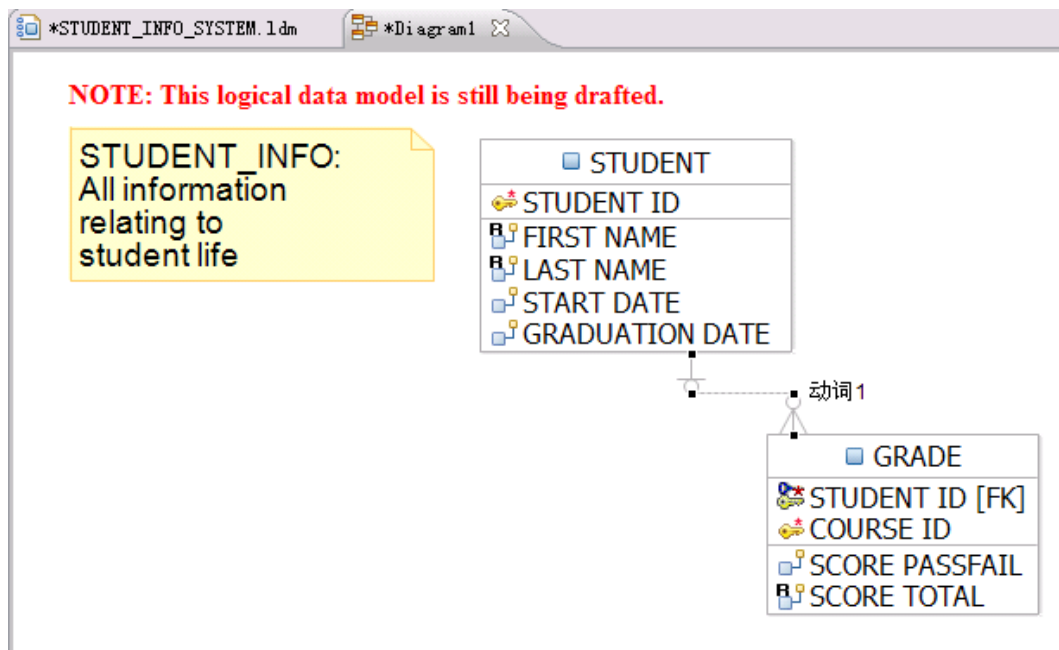


图3.10 – 选择关系对象

- b. 打开“详细信息”选项卡。
- c. 输入动词短语来描述父实体和子实体之间的关系：
 - GRADE: **is given**
 - STUDENT: **receives**

如图 3.11 所示，在属性视图中这个关系被定义好了。

注意：
 所有的外键属性都是自动创建的。如果子实体中不存在一个同名的对应属性，IDA 将不提示进行键属性迁移。

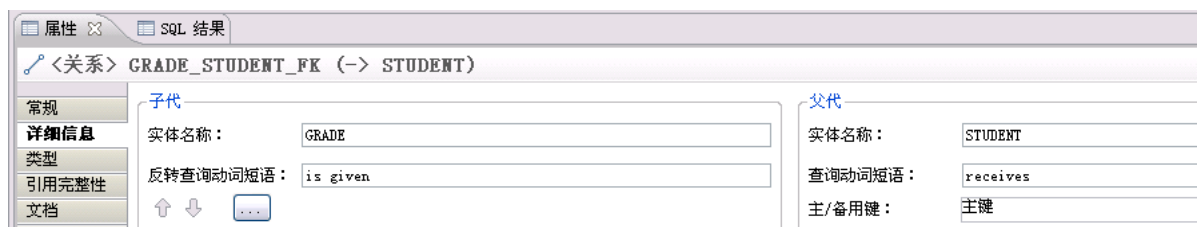


图3.11 – 在属性视图中定义关系

在属性视图中，用户可以选择关系的类型，键属性以及基数。

- 4. 浏览属性视图的其他选项卡，看看还可以定义那些其他信息：

- a. 打开“类型”选项卡。定义如下选项：
 - “关系类型”：标识非
 - “存在”：选可
 - “基数”：个多或个一
- b. 打开“引用完整性”选项卡，可以指定更新该关系的父实体的主键时，子实体将如何变化。在这里，我们将“父操作”组中的“删除”选项设置为 **CASCADE**。

图 3.12 展示了完成后的“引用完整性”选项卡：



图3.12 – 指定子实体GRADE的CASCADE操作

表 3.1 详细列举了指定一个动作后，子实体将会发生什么变化。您在逻辑数据模型中设计好这些需求，然后部署转换后的物理数据模型，当更新被引用的数据时，这些动作就会发生。

动作	定义
NONE	什么都不执行，删除或者更新操作将会被回滚。
RESTRICT	父实体不会被删除，并且相关的键列也不会被更新。因为子实体依赖于它。
CASCADE	如果被修改的父实体拥有一个到其他实体对象的外键关系，那么这个外键引用将会被更新。如果这个实体被删除，而且删除动作被设为 CASCADE ，那么依赖实体将从所有关系中删除。
SET_NULL	如果被修改的父实体拥有一个到其他实体对象的外键关系，那么在所有依赖实体中，外键属性如果可以设置为空，则它们将会被设置为 NULL 。本选项只有当所有的外键属性都允许为空值时才能被使用。
SET_DEFAULT	当父实体被修改时，所有可以为空的外键属性将会被设置为默认值。这个属性只有当所有外键属性都拥有默认值的时候才能被使用。
DO_NOT_ENFORCE	这个将会生成 DB2 系统中的一个信息约束。生成的代码不被强制。 信息参考约束是 DB2 的一种约束，这种约束在通常的操作中不添加任何强制。只要当参照的完整性通过其他途径强制执行时才使用

	<p>这种约束，例如从其他源中提取数据。</p> <p>这些约束允许系统自动重写查询来提高性能。</p>
--	--

表3.1 – 父和子实体操作定义

5. 保存。

3.3 型模表汇词建创

有时候，当您在和其他人合作完成一个项目时，大家对于数据对象的命名通常是不同的。例如，一位数据架构师可能把和销售相关的实体命名为 **SALES_x**，但是另一位数据建模师可能更喜欢把它们命名为 **SLS_x**。随着时间的推移，系统会变得越来越难以理解，而且这些不一致的命名会在系统间传播，最后导致理解上的混乱，使应用开发的时间拉长，甚至于会导致两个不同系统中一样含义的字段无法对接。

为了避免这些问题，您应该在进行数据设计的初期设计开发一些命名标准。*命名标准*可以帮助您和组织里的其他人用一致的方式去查看数据，同时得到具有相同含义的数据。使用 **IBM InfoSphere Data Architect**，您可以创建一系列的规则，而这些规则需要所有的数据建模人员遵守，而且工作台可以依据这些规则自动地来为一些数据对象命名，例如关系。

*词汇表模型*描述组织允许数据对象使用的名称和缩写。因为在整个组织中使用了相同的命名约定，数据对象命名标准可以提升整个组织对于数据的理解。另外使用词汇表模型可以促进跨组织的数据共享和通过合并同义和重叠的数据元素来减少数据冗余。

为数据对象命名时需要考虑两个方面：语义规则和格式。从语义上考虑，词汇表模型对象包括：基本单词，类单词和修饰词：

- *基本单词*：表示收集的数据的业务概念。基本单词是一个描述数据主题领域的名词或名词词组。例如：**LOAN, EMPLOYEE**。
- *类单词*：标识数据的类别或分类，例如：**RATE, NAME**。
- *修饰词*：对基本单词和类单词进行进一步限定或区分。修饰词确保数据对象名称的清晰性和唯一性，并对类单词和基本单词的含义进行限制。例如：**NEXT, FIRST**。

IBM InfoSphere Data Architect可以帮助您来定义语义规则或者命名规则。在 3.4 节，*使用命名标准*，我们将进一步介绍在数据设计项目中如何定义和使用命名规则：

为了保证模型符合词汇表模型中定义的命名约定，包含模型的项目必须引用该词汇表模型。执行如下步骤在项目中引用一个已有的词汇表模型：

注意：

下面的章节将创建一个词汇表模型并被数据设计项目所引用，所以您不需要执行这些步骤。只有当创建词汇表模型时没有添加项目引用，或者从另一个产品中导入词汇表模型的时候，才需要执行这些步骤。

1. 在数据项目资源管理器中选中一个数据设计项目，该项目的属性显示在属性视图中。

2. 打开“命名标准”选项卡并点击“添加”按钮打开选择文件窗口。
3. 展开数据设计项目并选择一个词汇表模型，如图 3.13 所示。

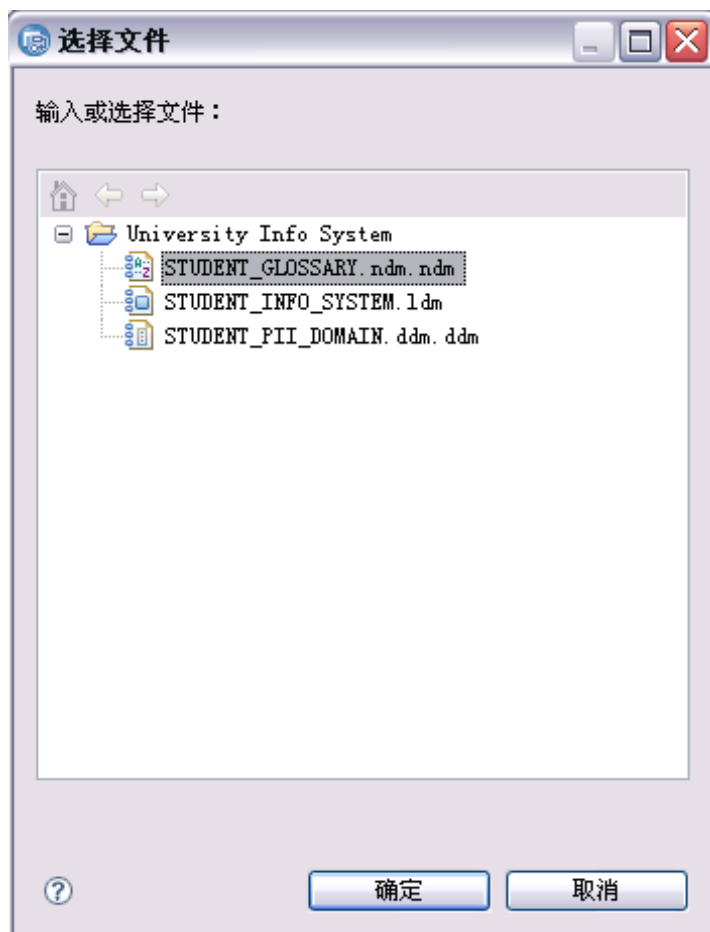


图3.13 – 选择词汇表模型

这样，该数据设计项目就引用了这个词汇表模型。

3.3.1 命名标准和词汇表模型的最佳实践

下面是设计命名标准的一些最佳实践：

- 为实体命名时选择单数。
- 不要使用数字作为实体名字的首字母。
- 尽量用一到两个词来命名，保持名字简短，清楚。

下面是创建词汇表模型的一些最佳实践：

- 创建一个单独的企业标准项目来存放需要在组织中共享的词汇表模型或者域模型。第 4 章将会对域模型进行详细的介绍。
- 用一个特定的数据设计项目来为遗留模型创建词汇表模型，因为这些模型在企业级的词汇表创建之前被创建，并可能使用了不同于企业词汇表的定义。
- 用有意义的名字来给词汇表模型命名。
- 创建多个词汇表对术语进行逻辑分组，将同一业务的相关术语放在相同的术语表中。
- 在一个特定的业务领域里，创建嵌套的词汇表来分类存放术语。
- 创建一个子词汇表来存放所有的类单词，这样可以避免您需要通过浏览整个词汇表来理解这些术语。
- 如果可能的话，您可以用电子制表软件来输入单词，这样通常会更有效率。您可以把词汇表复制到电子制表软件如 Lotus® Symphony 或者 Microsoft Excel，更新后再复制到 InfoSphere Data Architect。

3.3.2 型模表汇词建创

接下来，我们会为学生信息管理系统创建一个词汇表模型和词汇表。

创建一个词汇表模型：

1. 选择“文件”->“新建”->“词汇表模型”。
2. 选择数据设计项目 University Info System 作为目标文件夹。
3. 在文件名中输入 *STUDENT_GLOSSARY*。
4. 确保选项“作为项目命名标准添加到项目属性”是选中的，这样新创建的词汇表模型将会添加到项目的命名标准定义中。点击“完成”创建词汇表模型。

词汇表模型 *STUDENT_GLOSSARY.ndm* 现在已经被打开，您可以在数据设计项目的“数据模型”文件夹下找到该词汇表模型。

5. 为词汇表模型的顶层包指定名字：
 - a. 如图 3.14 所示，在词汇表模型 *STUDENT_GLOSSARY* 下选中数据对象 *NamingStandard*：

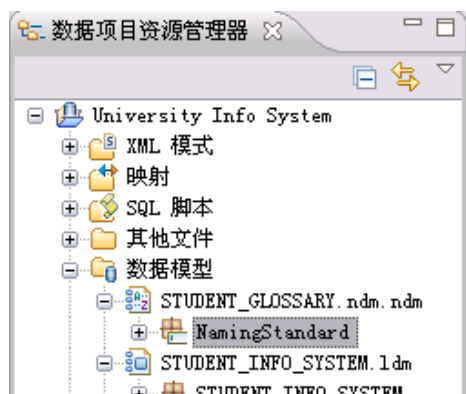



图3.14 – 选择NamingStandard包

- b. 在属性视图的“常规”选项卡中，将包名更新为 **STUDENT_GLOSSARY_ROOT**。
6. 为空的词汇表指定名字。选中词汇表 *Glossary1*，然后将词汇表的名字更新为 **STUDENT_GLOSSARY**。
7. 创建词汇表项 ID Number:
 - a. 在“包含的单词”选项卡中，点击按钮“创建新单词”（）。这样新单词 *Word1* 就成功地添加到词汇表中了。
 - b. 选中新添加的单词，将它的名字更改为 *ID Number*。
 - c. 在“缩写”字段中，输入 ID。
 - d. 在“类型”字段中，指定为 *PRIME*。
8. 继续为这个词汇表模型添加如下单词:
 - a. 创建如下基本单词
 - Address (为写缩 ADDR)
 - Date (DATE)
 - Department (DEPT)
 - Name (NAME 或 NM)
 - Pass/fail (PASSFAIL 或 PF)
 - Social security number (SSN)
 - Total number (TOTAL 或 TTL)
 - b. 创建以下类单词
 - Course (COURSE 或 CRS)
 - Email (EMAIL 或 EML)
 - First (FIRST or FT)，同时也是一个修饰词。

- Graduation date (GRADUATION 或 GRDN)
- Instructor (INSTRUCTOR 或 INSTR)
- Join date (JOIN)
- Last (LAST 或 LT) ，同时也是一个修饰词
- Score (SCORE 或 SCR)
- Start date (START)
- Student (STUDENT 或 SDT)
- Student activity (ACTIVITY 或 ACT)

9. 存保

在接下来 3.4 使用命名标准小节中，您将了解如何修改命名标准来保证在对逻辑数据模型进行验证时，所有的实体和属性名称都符合词汇表模型的定义。

注意：

您不必修改模型验证过程中出现的警告。警告只是提醒您，数据对象的名字不符合词汇表模型，您可以选择修改命名规则选项，或者修改数据对象的名字，使数据对象的名字符合词汇表模型的定义。

3.4 使用命名标准

现在我们已经创建了一个词汇表模型，让我们再为工作空间中的数据设计项目建立需要遵循的命名标准。在与大学官员沟通之后，您决定使用在基本词之前添加描述性类词来命名属性和列，然后您到工作台的首选项中设置该命名标准。

为了指定物理和逻辑数据模型或是词汇表模型的命名标准，打开首选项窗口：

1. 点击“窗口”->“首选项”打开首选项窗口。
2. 展开“数据管理”节点选择“命名标准”。
3. 确保“逻辑”选项卡被打开。
4. 在实体部分，指定如下顺序：
 - a. 修饰符（可选）
 - b. 类字（可选）
 - c. 主字（可选）
 - d. 修饰符（可选）
 - e. 修饰符（可选）
 - f. 修饰符（可选）
5. 在属性部分，指定如下的顺序：

- a. 修饰符（可选）
 - b. 类字（必需）
 - c. 主字（必需）
 - d. 修饰符（可选）
 - e. 修饰符（可选）
 - f. 修饰符（可选）
6. 在“物理-表/列”选项卡中，重复步骤 4 和 5 指定相同的选项。
 7. 点击“确定”来保存首选项。

3.4.1 使用命名标准分析检查合规性

当设置命名标准后，您应该检查哪些数据对象是符合指定的命名标准和词汇表模型项的。使用分析模型向导来指定需要检查的规则，然后再利用问题视图，以期查找和纠正任何不符合命名标准的数据对象。

分析逻辑数据模型：

1. 定位到逻辑数据模型中的包 *STUDENT INFO SYSTEM: University Info System* -> “数据模型” -> *STUDENT_INFO_SYSTEM.Idm* -> *STUDENT INFO SYSTEM*。
2. 在此包上右键单击，然后选择“分析模型”打开分析模型向导。

由于我们分析的是逻辑数据模型，逻辑数据模型规则类别下的每一条规则都被默认选中。

1. 单击规则类别“逻辑数据模型”的复选框以取消选择，然后选择规则“命名标准”。选择这条规则可以缩小模型分析的范围，以确保数据对象服从命名标准规则。您可以在问题视图中看到对应的错误和警告。
2. 点击“完成”来分析此模型。

注意：

如果在问题视图中显示的是警告，您并不需要修复警告来创建一个有效的、可用的逻辑数据模型。警告只是让您知道特定的分析项可能没有遵守或不符合标准的数据建模规则，但它不会影响该模型的功能。

3.5 约束

现在已经创建了几个实体，您还需要创建一些约束。*约束*被用来针对数据强制执行业务规则。

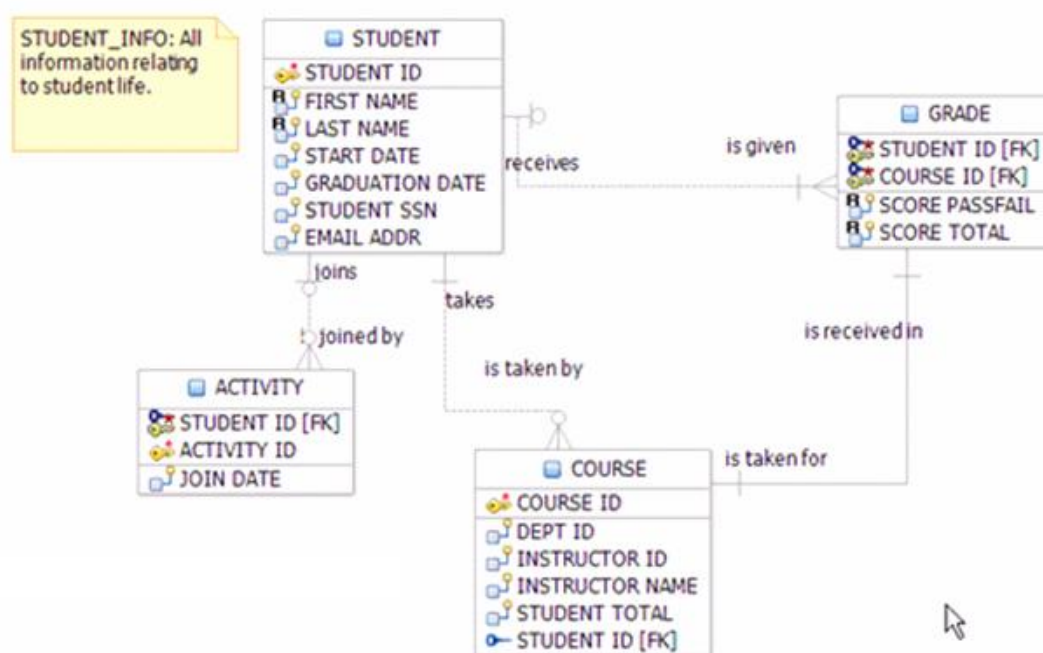
首先，让我们创建一个约束来检查并确保学生的毕业日期晚于学生的开始日期。这个检查约束将确保输入到数据库中的数据是有效的。

为 **STUDENT** 实体添加一个约束：

1. 择选中器理管源资目项据数在STUDENT: 体实 *University Info System* -> “数据模型” -> *STUDENT_INFO_SYSTEM.Idm* -> *STUDENT INFO SYSTEM* -> *STUDENT*。
2. 点击 **STUDENT** 实体, 然后选择 “添加数据对象” -> “约束”。
3. 命名该约束为 **DATE_CK**。
4. 在属性视图中的 “常规” 选项卡中完成以下信息来创建这个约束:
 - a. 指定语言为 **SQL**。
 - b. 在 “表达式” 字段输入: **GRADUATION_DATE>START_DATE**。
 - c. 在 “变换为” 域中, 指定为 “检查约束”。这个检查约束只是检查这个表达式是否为真。
5. 保存。

3.6 习练

1. 添加另一个实体 **ACTIVITY**, 来描述每一个学生所属的俱乐部和组织。本书的其它章节中您将用到这个实体。这个实体应该包括如下属性:
 1. **STUDENT ID**: 存储学生的 ID 号, 这是组合主键的一部分。
 2. **ACTIVITY ID**: 存储 5 个字符。
 3. **JOIN DATE**: 存储学生加入组织的时间。
2. 用 *非标识可选* 关系关联 **STUDENT** 和 **ACTIVITY** 实体, 设置该关系的基数为 *零个或多个*。用 “详细信息” 选项卡来定义父子实体之间的关系, 并为这个关系添加描述性动词。
3. 现在为逻辑数据模型创建另一个实体 **COURSE**。创建如下属性:
 - **COURSE ID**: 存储 5 个字节的课程 ID 号。这是实体的主键。
 - **DEPT ID**: 存储 4 个字节的课程的部门 ID 号。例如, 英语部的课程使用 **ENGL** 前缀。
 - **INSTRUCTOR ID**: 存储课程指导教师 ID, 大小为 5 个字节。
 - **INSTRUCTOR NAME**: 存储课程指导教师姓名。
 - **STUDENT TOTAL**: 存储注册该课程的学生人数。
4. 用 *非标识必需* 关系关联 **STUDENT** 和 **COURSE** 实体, 设置基数为 *一个或多个*。再通过 “详细信息” 选项卡来描述两个实体间的动词关系。用 *标识* 关系关联 **COURSE** 和 **GRADE** 实体, 设置基数为 *只有一个*。为每一个关系添加描述性动词。
5. 保存。
6. 确定该图正确地展示了创建的逻辑数据模型。当完成后, 您的逻辑数据模型应该如图 3.15 所示。



NOTE: This logical data model is still being drafted.

图3.15 – 完整的图 STUDENT_INFO和逻辑数据模型

现在，您已经创建了一个小的逻辑数据模型。定义一个域模型可供该逻辑数据模型和项目中的其它数据模型所用，域模型可以让您为组织定义自定义数据类型。在下一章中，您将学习域模型以及如何将它应用到逻辑数据模型中。

3.7 总结

本章讨论了如何使用 IBM InfoSphere Data Architect 来进行逻辑数据建模。您学习了如何用数据图和数据项目资源管理器来创建数据模型。

本章还介绍了如何使用命名标准和词汇表模型。命名标准帮助组织里的每一个人，以同样地方式看待相同的词汇和数据对象，从而得到相同的含义。这些词汇在整个 IT 系统中有一致的表示。命名标准促进了数据理解的一致性。

3.8 本章回顾

1. 为了确保一个属性值仅包含 ‘B’ 或 ‘G’，您应选择哪种数据对象？
 - A. 约束
 - B. 主键
 - C. 词汇表模型

- D. 关系
2. 下面哪个选项不是实体的定义？
- A. 一个实体是一个真实世界的对象或概念的集合，它包含许多共同的属性。
 - B. 一个实体是一个真实世界对象的实例。
 - C. 一个实体是相同类型实体的集合，它可以分享相同的属性。
 - D. 一个实体是一个存在的并且不同于其它的项。
3. 定义一个人的电话号码属性，下面哪种数据类型最合适？
- A. NUMERIC
 - B. INTEGER
 - C. CHAR
 - D. LONG
4. 画出下面的实体和关系集。
- A. 每个车库可容纳一辆或多辆车。每辆车可停在一个且只有一个车库。
 - B. 每个学生可以有一位或多位老师。每位老师可能指导一个或多个学生。
 - C. 一个员工可能管理一个或多个员工。一个员工必须被一个员工管理。
 - D. 一个员工可能属于一个公司的分支机构。一个公司的分支机构必须有一个或多个员工。
5. 为了解决学生和教师之间的多对多的关系。考虑为这个新实体设置额外的属性。
6. 一个逻辑数据模型中有如下实体，现在用图形化的展现方式来展示如下实体和关系集：
- 一个公司的分支机构；
 - 公司分支机构的员工和他们的亲属；
- 画一个数据模型图。
7. 在一个大学逻辑数据模型中，选择 **STUDENT** 和 **GRADE** 实体之间最适合的关系：
- A. 每一个 **STUDENT** 必须是一个或多个 **GRADES** 的接受者。
 - B. 每一个 **STUDENT** 必须是一个或多个 **GRADES** 的发送者。
 - C. 每一个 **STUDENT** 可能是一个或多个 **GRADES** 的接受者。
 - D. 每一个 **STUDENT** 可能是一个或多个 **GRADES** 的发送者。

4

第四章 – 域模型

在这一章，您将要学习如何创建一个域模型。域模型是一种描述组织中的域数据类型和其约束的模型。您可以定义域模型作为逻辑数据模型的一部分，或者在项目中创建一个单独的域模型应用在所有逻辑和物理数据模型中。

接下来我们将创建一个域模型，该域模型将对测试环境中需要保护的信息进行建模。在我们的例子中，我们将对每一位学生的社会保险号、姓名和电子邮件地址进行加密。由于该测试系统用真实数据填充，我们需要利用隐私规则确保敏感信息受到保护。这样您将可以使域模型与逻辑数据模型中的数据对象进行关联。

注意：

您在 InfoSphere Data Architect 中创建的域模型可以和 Optim Data Privacy Solution 的一系列产品进行兼容。更多关于 Optim Data Privacy Solution 的信息，请访问如下链接：

<http://www-01.ibm.com/software/data/optim/protect-data-privacy/>

4.1 域模型

域模型是一种描述组织中的域数据类型和其约束的模型。您可以在域模型内创建组织特有的数据类型（换句话说，就是描述某一领域不同侧面的数据类型）。使用域数据类型代替基本的数据类型确保了整个组织的一致性，而通过重用域模型类型的定义可以确保团队更加高效。

域数据类型可以存储在域模型或者作为逻辑数据模型的一部分。您可以将一个域数据模型关联到数据设计项目，这样域模型就可以应用于该数据设计项目中的所有逻辑和物理数据模型。利用原子域代替基本数据类型可以确保团队成员之间的一致性。

一个域模型包含一组域数据类型。一个域数据类型是由添加了约束限制的基本数据类型组成的。逻辑或者物理数据模型中的属性或列可以使用域模型中定义的域数据类型。

域数据类型基于基本的数据类型。例如，您可以针对公共的描述符定义域数据类型，例如性别或者成绩。我们可以使用 IBM InfoSphere Data Architect 来创建域模型和原子域。所有从事建模工作的团队成员可以共享这些域模型。IBM InfoSphere Data Architect 提供了创建和管理域模型的工具。您可以在向导中基于已有的模板或者从头创建域模型。在数据项目资源管理器中添加域对象，然后在属性视图中对其进行修改。

下面列出了创建域模型的一些最佳实践：

- 把所有的域放到一个或者多个域模型，这样今后的项目就可以重用这些域。
- 创建一个单独的企业级标准项目，其中包含词汇表模型和域模型，这些模型在整个组织中可以共享。
- 如果遗留的模型中包含任何企业域创建之前创建的域，而这些域不符合新的企业标准，那么我们需要为这个遗留的模型在专有的项目中创建一个域模型。
- 用有意义的名字为域模型命名。您可以在这个模型命名中加入“域”进行标识，从而使域模型在数据项目资源管理器视图中可以轻松识别。
- 创建子包（在根域包下创建单独的域包）为特定的业务主题领域提供一个逻辑分组的域。

现在，我们已经创建了基本的逻辑数据模型，还需要把一些敏感的信息存储到这个逻辑数据模型中，包括社会保险号码和电子邮件地址。在展开包含这些敏感信息的逻辑数据模型之前，您决定创建一个域模型以确保这些敏感信息是受保护的。然后，您可以在 **STUDENT** 实体上创建新的属性并用域模型数据类型定义这些属性。

首先，让我们定义一个域模型来保护个人信息。这个域模型将帮助您确保学生的个人信息是受保护的。

4.1.1 型模域的白空个一建创

创建域模型 **STUDENT_PII_DOMAIN**：

1. 右键单击数据项目资源管理器中的“数据模型”文件夹，然后选择“新建”->“域模型”。新建的域模型窗口被打开。
2. 为这个域模型指定文件名：**STUDENT_PII_DOMAIN**。
3. 确保“作为项目域模型添加到项目属性”的选项是选中的，单击“完成”。名为 **STUDENT_PII_DOMAIN.ddm** 的域模型被创建在数据设计项目的“数据模型”文件夹中。
4. 为这个域模型的根包指定名称：
 - a. 选择在 **STUDENT_PII_DOMAIN** 域模型下的包，如图 4.1 所示：

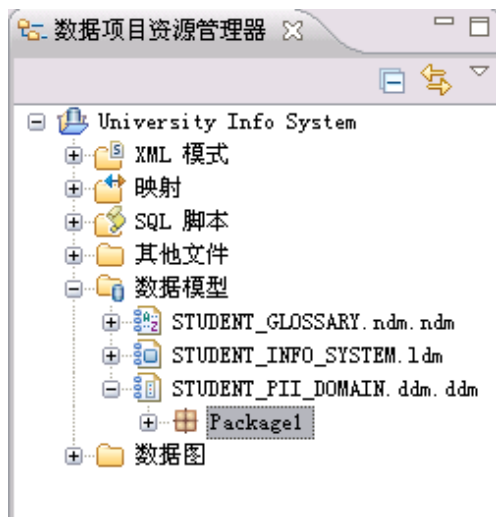


图 4.1 – 择选 STUDENT_PII_DOMAIN 型模域

属性视图中显示了这个包的属性。

b. 在“名称”字段中指定如下名字：**STUDENGT_PII_ROOT**。

5. 保存。

域模型和包被创建并显示在数据项目资源管理器中。现在，您已经创建了一个空的域模型，接下来您可以添加一个原子域、列表域或并集域。在本练习中，您将添加一个原子域以确保每一个学生的社会保险号是安全的。

4.1.2 原子域

原子域是您为了保证整个组织的一致性，创建的自定义数据类型。创建特定的域规则，然后用这些规则来定义数据模型内的其他对象。

例如，您可以定义一个成绩的域模型以确保输入的值在适当的范围以内。如果是限制为 10 的范围，那您只能输入 1 到 10 之间的值。

您也可以创建一个域模型来指定需要伪装的隐私数据，如学生或教师标识符。

默认情况下这些屏蔽的数据是不强制数据伪装的，但是为了确保敏感数据的隐私性，您应该强制执行数据隐私。

使用原子域，您可以指定四个方面的隐私规则：

- 分类：在测试环境中，这个数据应该如何分类？它是个人可标识信息，还是机密信息？
- 强制执行：在测试环境中，原子域的隐私性是必需的、不必需的还是只是最佳实践？
- 隐私策略类型：在测试环境中，什么类型的信息需要进行数据伪装？

- 隐私策略：在测试系统中采用什么方式对数据进行伪装？

4.1.2.1 创建一个原子域

您决定首先确保学生的社会保险号的安全性。要做到这一点，您需要创建一个新的原子域 **SSN**。通过该原子域，一个随机的社会保险号会替代真实的社会保险号，这样就可以对原有的敏感信息进行伪装。

数据设计项目中的这个原子域创建了一个特有的数据类型，在您项目的各种模型中都可以使用这个特有的数据类型。在整个设计流程中，这个域模型也可用来定义学生信息存储策略。这将帮助确保所有数据模型的一致性。

下面让我们创建原子域 **SSN** 和 **STUDENT_ID**：

1. 在包 **STUDENT_PII_ROOT** 上单击右键，然后选择“添加数据对象”->“原子域”。在数据项目资源管理器下的根域包下，一个新的原子域对象被创建。
2. 命名这个原子域对象为 **SSN**。
3. 在属性视图的“常规”选项卡中定义 **SSN** 原子域对象的基本数据类型。由于社会保险号码包含数字和破折号，您应该指定基本数据类型为 **CHAR(11)**。
4. 为这个原子域的数据类型定义数据隐私属性：
 1. 打开属性视图中的“数据隐私”选项卡。
 2. 指定如下设置来对学生的社会保险号码进行伪装：
 - “分类”：个人可标识信息
 - “强制执行”：必需
 - “隐私策略类型”：社会保险号 (SSN)
 - “隐私策略”：具有有效源区域号的随机 **SSN**
5. 创建原子域 **STUDENT_ID** 并设置如下：
 - “基本类型”：**CHAR(10)**
 - “分类”：个人可标识信息
 - “强制执行”：必需
 - “隐私策略类型”：混排
 - “隐私策略”：随机混排

通过指定如上选项，任何使用这个域模型的属性或列在测试环境中都需要进行伪装。这将有助于减少大学中任何可以识别学生的数据的泄露。

4.1.3 域集并和域表列

列表域的值是由空格分隔的，如 **AvailableSizes (10 12 14)**。这是原子域的一种类型。

并集域的值可以是原子值、列表值或并集值的集合。例如，您可以创建一个并集域 **DressSize**，该并集域允许的值是一个 2 到 18 的整数或 **small**、**medium**、**large** 的字符串类型值。

注意：

本书将不详细介绍如何创建列表域或原子域。这里提供的定义仅供参考。

4.2 将域模型元素关联到逻辑数据模型元素

一旦在域模型中创建元素，您应该将它们关联到逻辑数据模型中的元素。通过关联域模型与逻辑数据模型（或者是后面章节提到的物理数据模型），您迈出了信息私有化过程中必需的第一步，即识别业务需求。这样就可以与其他的团队和系统共享您的策略。然后，您可以将这个域模型导出到 **Optim Designer**（**Optim Data Privacy** 解决方案的一部分），这样就可以将与该域模型相关联的实体和属性中的数据替换为伪造的（或安全的）数据。一个常见的应用是创建一个用真实数据填充的测试数据库，然后测试数据对象和域模型是如何关联以确保安全的。

注意：

伪装数据只应用于测试系统上。

本书中，您应该将 **SSN** 原子域与 **STUDENT** 实体的相应属性进行关联。为了确保学生的社会保险号码的安全，您将指定 **SSN** 原子域的特定数据类型。另外还应该创建一个原子域来确保每一个学生的学生 ID 号的安全。

对原子域数据类型指定如下属性：

1. 首先，为实体 **STUDENT** 创建一个新的属性，并将它命名为 **STUDENT SSN**。
2. 在数据项目资源管理器中选择属性 **STUDENT SSN**。属性视图将显示相关属性。
3. 改变属性 **STUDENT SSN** 的数据类型：
 - a. 打开属性视图中的“类型”选项卡。
 - b. 单击“数据类型”的字段旁边的省略号 (...)，打开选择数据类型窗口。
 - c. 选择 **STUDENT_PII_ROOT** 包中的 **SSN** 数据类型，如 [图 4.2](#) 所示。

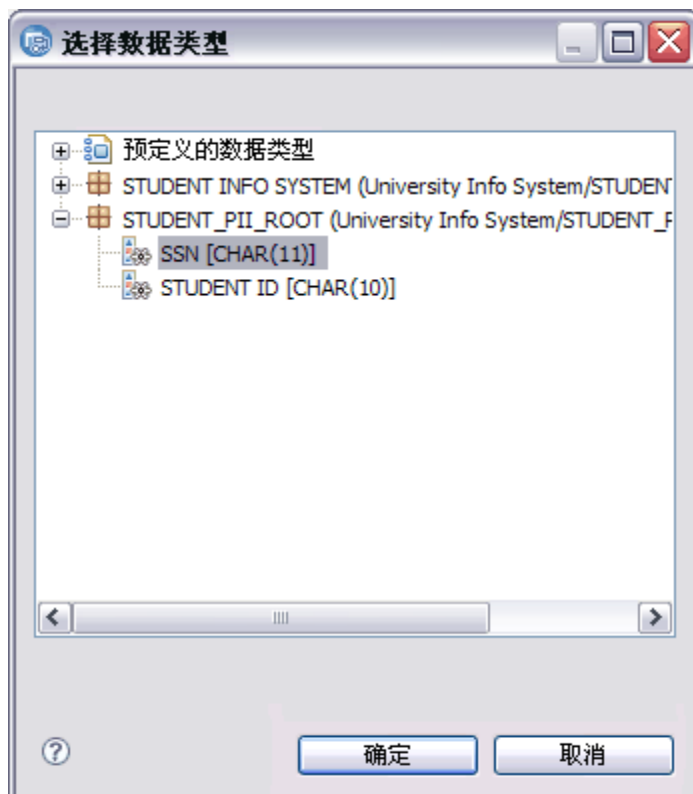


图 4.2 – 选择原子域数据类型

这个域数据类型被指定到 *SSN* 属性的数据类型字段中。

4. 改变属性 *STUDENT ID* 的数据类型：
 - a. 打开属性视图中的“数据类型”选项卡。
 - b. 单击“数据类型”的字段旁边的省略号 (...) 打开选择数据类型窗口。
 - c. 从 *STUDENT_PII_ROOT* 包中选择 *STUDENT_ID* 数据类型。
5. 保存。

4.3 习练

完成域模型 *STUDENT_PII_DOMAIN*。

1. 创建以下的原子域元素：
 - *EMAIL*: 基本类型是 *VARCHAR (128)*。设置分类为个人可标识信息，执行类型为必需。当进行数据伪装时，隐私策略为随机电子邮件地址。
 - *LAST_NAME*: 基本类型是 *VARCHAR (128)*。设置分类为个人可标识信息，执行类型为最佳实践。当进行数据伪装时，隐私策略为随机混排。

2. 在 **STUDENT** 实体中创建 **EMAIL ADDR** 属性，并把该属性关联到原子域元素 **EMAIL** 的属性中。
3. 把实体中的 **LAST NAME** 属性与原子域元素 **LAST NAME** 进行关联。
4. 保存。

4.4 结总

本章概述了如何在 **IBM InfoSphere Data Architect** 中创建域模型。您现在应该已经了解了如何将域模型关联到一个已有的逻辑数据模型，以及当测试模型和系统时，如何识别私有信息。

4.5 回顾章本

1. 域模型的定义是什么？
2. 域模型中存储的信息是什么？
3. 正误判断：域模型可以帮助您为企业识别数据隐私政策。
4. 正误判断：您可以用域模型定义自己的数据类型。
5. 什么是嵌套域？

5

第五章 – 物理数据建模

在本章中，您将要学习 *物理数据模型* 以及它们与逻辑数据模型的不同。您也将了解到如何把一个逻辑数据模型转换成一个物理数据模型。在本章的结尾，您还将学习如何通过物理数据模型创建数据库特有的 DDL 语句，这可以把练习中创建的数据建模转变为数据库中的模式。

5.1 物理数据建模：概述

物理数据模型是特定于某一种数据库的数据模型，它们是基于数据库规范的。通过物理数据模型，您可以对存储进行建模，包括分区、表空间、索引和其他的存储对象。

您可以在一个特定的数据库和数据库平台上用特定的属性进行数据建模。例如，物理数据模型在 DB2 系统、Oracle 系统或 SQL Server 系统上建模是大不相同的。IBM InfoSphere Data Architect 支持多种不同的数据库系统类型和版本，您可以利用一个物理数据模型为这些系统创建数据库特有的数据对象。

注意：

InfoSphere Data Architect 仅仅支持 DB2 for Linux、UNIX、Windows、DB2 for z/OS®和 DB2 for i5/OS®数据库的存储建模。对于其他数据库供应商，您可以在工作台上创建表、列和其它相关对象，但不支持存储建模，例如表空间和缓冲池。

在这本书中，我们将介绍如何为 DB2 系统建模。

通常情况下，当您从事物理数据模型的工作时，可以使用下面的工作流程：

1. 创建一个物理数据模型，该模型可能是从头开始创建、利用模板进行创建、从逻辑数据模型转换而来或者从其他系统中导入。
2. 通过增加存储对象、索引和视图来完善物理数据模型。
3. 生成可用于部署到数据库或模式的 DDL。
4. 运行 DDL 脚本在服务器上创建数据对象。

您可能会想，“逻辑和物理数据模型之间到底有什么区别？”表 5.1 总结了它们的区别。

逻辑数据模型	物理数据模型
与任何数据库供应商无关	总是与某一数据库供应商和数据库版本相关
基于一个更高层次的抽象，更通用	更具体；接近实际的数据库设计
不考虑物理存储	用来对所支持的数据库供应商进行物理存储建模
不需要数据库供应商构建知识；只需要知道模型如何被使用	需要了解数据库供应商的知识和在每个平台上模型是如何构建；需要了解数据是如何被存储的

表 5.1 – 逻辑和物理数据模型的不同

5.2 从头开始创建一个物理数据模型

理想情况下，您应该在创建物理数据模型之前先创建逻辑数据模型。这有助于保持逻辑和物理设计的独立性，允许您在设计一个数据库或是模式的物理存储之前，在逻辑层次修改数据模型。

然而，您也可以在工作台内创建一个空白的物理数据模型。

注意：

下面的步骤仅供参考。您不必执行这些步骤也可以继续设计学生信息系统。在下一节，您将学

习如何将逻辑数据模型转换为物理数据模型。

创建一个空白的物理数据模型：

1. 点击“文件”->“新建”->“物理数据模型”。新建物理数据模型向导被打开。
2. 完成新建物理数据模型向导。您可以创建一个空白的物理数据模型，也可以利用一个存在的数据库、模式或 DDL 脚本通过反向设计创建。反向设计将在后面的章节介绍。
3. 使用工作台添加数据对象，如表、模式或列。右键单击在数据模型中的各种对象，您可以从上下文菜单中添加数据对象或创建一个图来可视化展示这些数据对象以及它们之间的关系。

5.3 转换逻辑数据模型到物理数据模型

为了不用从头开始创建一个物理数据模型，我们将使用之前章节中创建的逻辑数据模型（STUDENT_INFO_SYSTEM.ldm）并转换为一个物理数据模型。转换过程中创建了数据库或模式的物理设计，从而节省了您手动创建表、列和其他数据对象。

从逻辑数据模型转换之后，您可以在物理数据模型中添加更多特定的属性以成为一个实用的、可部署的数据模型。

注意：

数据类型从一个数据模型到另一个数据模型的映射依赖于首选项窗口中的设置。为了确保数据类型在逻辑数据模型和新的物理数据模型之间映射的正确性：

1. 从主菜单中单击“窗口”->“首选项”打开首选项窗口。
2. 展开数据管理节点，找到“变换”->“数据类型映射”页面。
3. 展开在首选项窗口中的数据类型映射节点，指定将逻辑数据模型转换为各种数据库供应商对应的物理数据模型时数据类型的映射选项。

将逻辑数据模型转换为物理数据模型：

1. 在数据设计项目中选择 **STUDENT_INFO_SYSTEM.ldm** 文件。
2. 从主菜单中点击“数据”->“变换”->“物理数据模型”打开变换为物理数据模型向导。
3. 在目标物理数据模型页面，选择“新建模型”单选按钮，然后点击“下一步”。
4. 完成物理数据模型文件页面：
 - a. 在目标文件夹中指定数据设计项目为 *University Info System*。
 - b. 指定数据库类型为“DB2 Linux 版、UNIX 版和 Windows 版”。
 - c. 选择数据库版本为 V9.7。
 - d. 点击“下一步”。
5. 在选项页面中，在模式名字段中输入 **STUDENT_LIFE**，然后点击“下一步”。如图 5.1 所示。

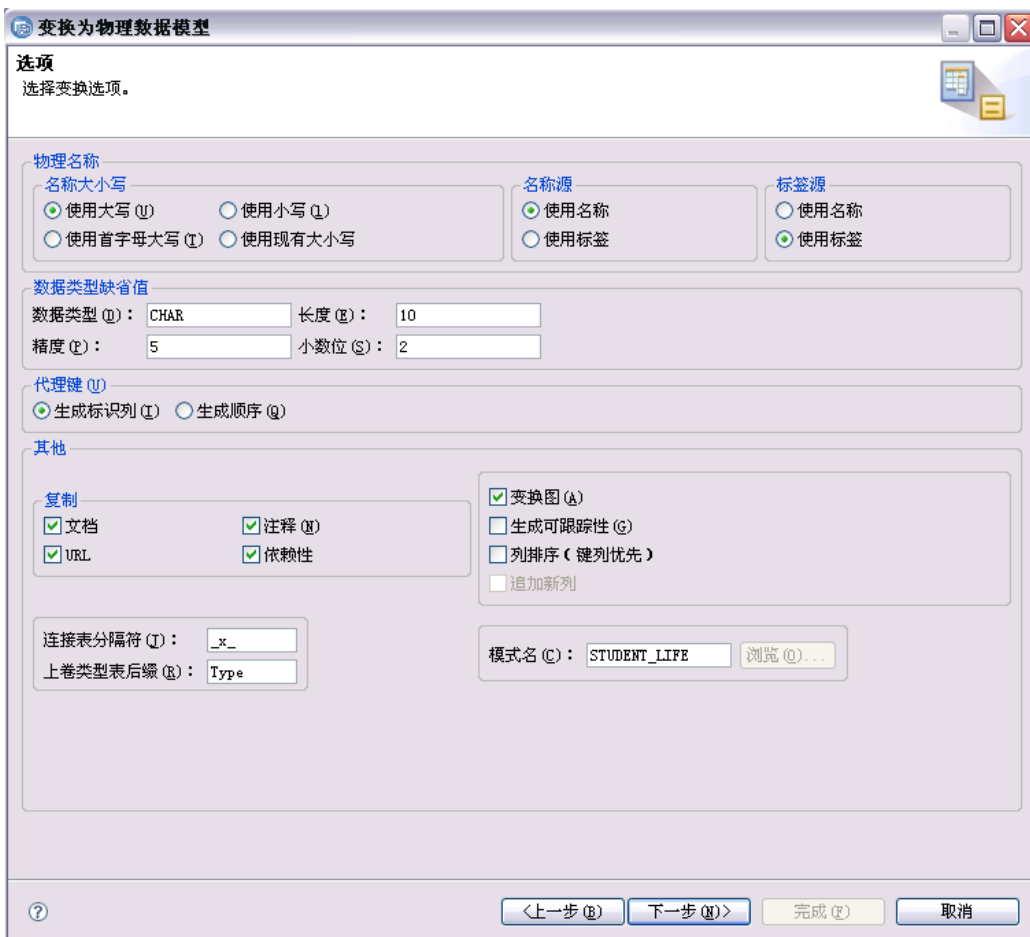


图 5.1 – 完成选项页

注意：

作为最佳实践，你应该选择“生成可追踪性”选项。此选项允许您识别出逻辑和物理数据模型之间的依赖关系，这将有助于在提炼数据模型时进行影响分析。

6. 在输出页面点击“完成”。物理数据模型在数据设计项目中的数据模型文件夹中被创建，并在编辑器视图中处于打开状态。
7. 为新数据库定义一个更具描述性的名字。
 - a. 在物理数据模型中选择数据库对象。
 - b. 在属性视图的“常规”选项卡上，指定名称为 **SAMPLE**。
8. 保存。

图 5.2 展示了转换后生成的数据对象和文件夹。

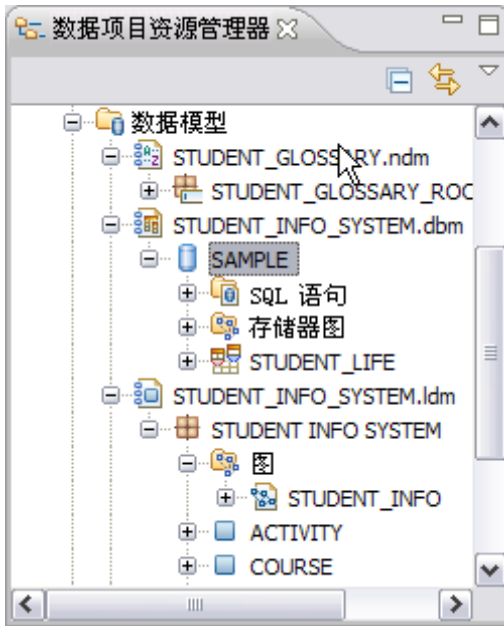


图 5.2 – 新的物理数据模型和数据对象

5.4 完善物理数据模型

5.4.1 模型解剖

图 5.3 展示了一个物理数据模型。

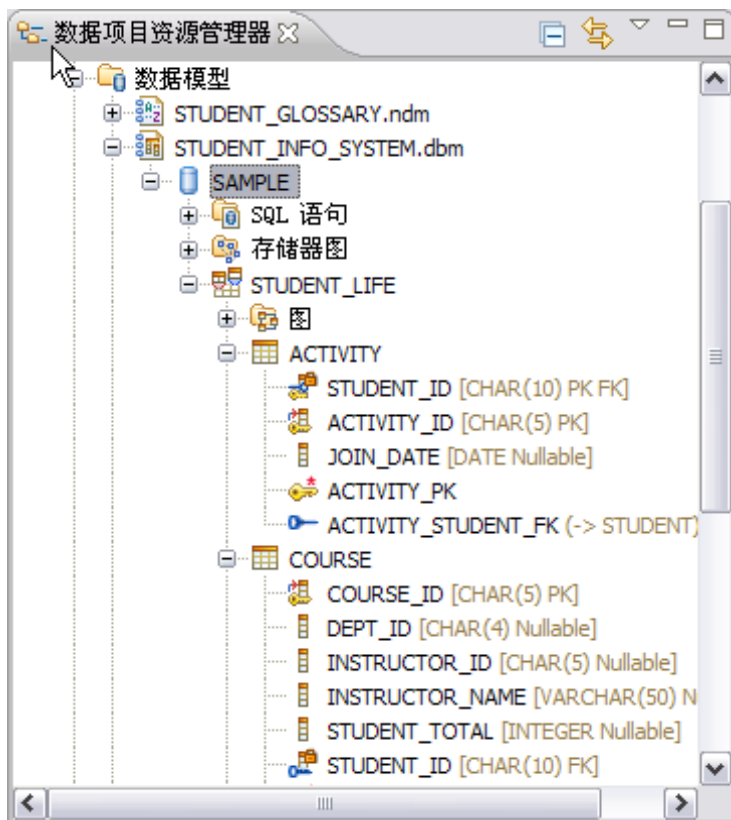


图 5.3 – 工作台中的物理数据模型

每一个物理数据模型有如下的特征：

- 文件后缀名：物理数据模型的文件后缀名是 *.dbm*。
- 模式和数据库设计：
 - 结构大体与一个逻辑数据模型相似。
 - 正如您以前学到的，逻辑数据模型中包含一个单一的包，可以在其中存储子包、实体和图。而物理数据模型是用来存储数据库信息。每个数据库可以有多个模式，而每个模式可以包含多个表。
- 图：
 - 当您转换一个逻辑数据模型的图时，您的设计包括注释、格式和关系不会被影响。唯一受到影响的是用来描述新的表、列和其他数据对象的图标。
 - 跟逻辑数据模型一样，您可以创建物理数据模型图来帮助您可视化展示和理解复杂的数据模型的设计。
- 存储器图：您可以通过图帮助您对 DB2 数据库进行存储建模。这包括分区、表空间、缓冲池和其他存储对象。
- SQL 语句：“SQL 语句”文件夹用来存储物理数据模型中创建的 SQL 脚本。

如果对物理数据模型进行探索，您会发现您指定的原子域元素也被转换到新的物理数据模型中。

5.4.2 对 DB2 进行存储建模

DB2 存储建模涉及到创建表空间、分区和数据库分组。

注意：

通过 IBM InfoSphere Data Architect，您只能为 DB2 for Linux，UNIX 和 Windows 数据库或者 DB2 for z/OS 数据库进行存储建模。

更多关于 DB2 存储模型的信息，请参阅[Getting started with DB2 Express-C](#)，这是本系列丛书的一部分。

本章没有详细地介绍如何进行存储建模，因为我们要创建的模式非常小。但是，本节介绍了存储建模背后的基本概念，当为复杂的数据模型设计存储时您应该记住这些信息。

5.4.2.1 表空间

数据库中的数据存储在表空间中。表空间和特定的数据库供应商相关。

注意：

更多关于 DB2 for Linux，UNIX 和 Windows 的表空间的信息，请访问如下地址：

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.dboobj.doc/doc/c0004935.html>

更多的关于 DB2 for z/OS 的表空间的信息，请访问如下地址：

http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp?topic=/com.ibm.db2z.doc.gloss/src/gloss/db2z_gloss.htm

表空间通常采用以下两种方式之一进行管理：

- **系统管理的表空间：**由操作系统管理的表空间。容器是操作系统文件，对容器的访问也是通过操作系统统一管理的。这意味着操作系统将负责以下操作：
 - 缓冲输入和输出
 - 空间分配
 - 自动扩展表空间
- **数据库管理的表空间：**由数据库管理的表空间。容器作为文件或者设备被定义，当容器是文件时，表空间的大小在空间创建时指定并分配。当您使用 DB2 来管理表空间，DB2 会处理如下操作：
 - 缓冲的输入和输出（缓冲大小由分配方法和操作系统的限制决定）
 - 使用 ALTER TABLESPACE 扩充容器
 - 释放数据库管理容器的未使用部分

每个表空间至少有一个容器。每个容器只属于一个表空间（但是一个表空间可以有多个容器）。如果数据库是分区的，在表空间还未被分配容器时，您只能为系统管理的表空间添加一个容器。当您为表空间添加一个容器时，数据才可以分发到所有的容器中。

当您指定一个表空间时，需要考虑如下设置（在表空间属性视图的“常规”选项卡中可以看到）：

- **页大小：**定义表空间使用的页面的大小。DB2 支持的大小为 4K、8K、16K 和 32K。页面大小限制了表空间中存放的表的行宽和列数。表空间上限是 16,777,216 页。选择较大的页大小可以增加表空间的容量。
- **扩展数据块大小：**指定在下一个容器被启用之前当前容器支持的写入页面数量。在数据存储的过程中，数据库管理程序循环重复地在容器间按照扩展数据块大小设置进行进行数据分发存储。本设置只有当一个表空间有多个容器时才起作用。
- **预取大小：**指定数据预取时，从表空间读取的页面数量。预取在一个语句实际使用数据之前对数据进行预先读取准备。这样就保证了当一个语句实际需要数据的时候立刻能获取到数据。预取是由数据库管理器控制，并当需要对数据进行顺序的输入或输出的时候选择执行的。无可厚非，预取数据可以提高性能。

最佳实践：

预取大小的设置可以通过扩展数据块大小和表空间的容器个数的乘积来确定。例如，如果你的扩展数据块大小是 32 而容器个数为 4，那么预取大小应该是 128，256 或是 128 的倍数。如果一个表经常被使用，为该表创建一个单独的表空间以改善性能。

- **开销和传送速率：**在查询优化过程中确定输入和输出的成本。它以毫秒为单位，是所有容器的平均值。开销是与输入输出控制器活动、磁盘寻道和旋转延迟的时间相关。传送速率是一个页面读取到内存的时间。这些值可以根据硬件规格来计算。

5.4.2.2 缓冲池

缓冲池是与单个数据库关联，并可用于多个表空间。当您为一个表空间创建一个缓冲池时，必须确保表空间大小和缓冲池页面大小都是一样的，并且缓冲池页面大小要与这个缓冲池所服务的所有表空间大小一致。一个表空间只能使用一个缓冲池。

合适的缓冲池大小对数据库性能很关键，因为它减少了磁盘的输入和输出，而输入输出是数据库最耗时的操作。大型缓冲池对查询优化的也有影响，因为这样更多的工作就可以在内存中完成。

5.4.2.3 存储建模影响性能的方面

当设计一个模式或数据库的存储时，您应该主要关注缓冲的利用率以及输入和输出的最大化并行数量。使用如下工作流以确保存储模型是高效的：

1. 考虑表设计中的约束。您可能需要使用多个常规表空间。
2. 考虑是否可以在大型表空间中通过不同的设置以提高性能。
3. 设计一个试验性的表空间。
4. 使用缓冲池，这可能会帮助改善表空间设计。
5. 为表空间分配容器。
6. 提炼模型，并且通过全面的测试和模型基准来验证你的设计是稳定的。

作为一般的规则，从最简单的设计开始。只有当性能会受到影响时才允许复杂设计。

5.5 提炼物理数据模型

现在我们已经将一个逻辑数据模型转换为一个物理数据模型，在我们生成 DDL 部署到数据库之前应该做一些最终的提炼。首先，我们将对表 `STUDENT` 中的列进行重新排序以使它们更符合逻辑顺序。然后我们创建一个角色并将一个用户添加到数据库中。

5.5.1 在物理数据模型中重排表列

当创建逻辑数据模型时，您的关注点是哪些属性和数据特征必须包含在实体定义中。让我们重新排列 `STUDENT` 表中的列，使对象更容易被最终用户所理解。

重新排列这些列，请按如下步骤操作：

1. 选择模式 `STUDENT_LIFE` 下的 `STUDENT` 表。表的属性显示在属性视图中。
2. 打开属性视图中的“列”选项卡。
3. 重新排列这些列，确保重要的学生信息放在前面：
 - a. 将列 `FIRST_NAME` 排在表的首位。选择 `FIRST_NAME` 列，然后通过列上移箭头(↑)来移动此列到表的首位。如图 5.4 所示。



图 5.4 – 重排表中的列

- b. 把 `LAST_NAME` 列移动到 `FIRST_NAME` 下，设置为表的第二列。
- c. 把 `STUDENT_SSN` 列设置为表的第四列。
- d. 把 `EMAIL_ADDR` 列设置为表的第五列。

这些列的顺序如图 5.5 所示。

名称	主键	域
FIRST_NAME	<input type="checkbox"/>	
LAST_NAME	<input type="checkbox"/>	LAST_NAME [VARCHAR]
STUDENT_ID	<input checked="" type="checkbox"/>	STUDENT_ID [VARCHAR]
STUDENT_SSN	<input type="checkbox"/>	SSN [CHAR(11)]
EMAIL_ADDR	<input type="checkbox"/>	EMAIL [VARCHAR(12)]
START_DATE	<input type="checkbox"/>	
GRADUATION_D...	<input type="checkbox"/>	

图 5.5 – 列在表中的适当位置

4. 保存。

5.5.2 在物理数据模型中创建角色

角色可以简化权限的管理。角色是一种数据库对象，它把一个或多个权限组合在一起并分配给用户、组、公共或其他角色。角色提供的优点使它更容易在数据库系统中管理权限：

- 安全管理员（SECADM）可以控制对数据库的访问，在某种程度上反映了其组织结构。他们可以在数据库中创建角色使其直接映射到组织中的工作职能。
- 用户被赋予不同的角色，而角色反映他们的工作职责。当工作职责改变时，他们的角色可以很容易地被授予和撤销。
- 权限的分配是简单的。对于某一特定的工作职责，取代将一组相似的权限直接赋予一组最终用户，管理员可以事先将这一组权限赋予执行某一特定工作的角色，然后将角色赋予需要参与工作职责的最终用户。
- 角色的权限可以被更新而所有被授予该角色的用户可以获得此更新。管理员并不需要单独为每一个用户更新权限。
- 当您创建视图、触发器、物化查询表（MQT）、静态 SQL 和 SQL 例程时，授予给角色的特权和权限将被使用。然而被授予组（直接或间接的）的特权和权限是不可用的。因为角色是在数据库内管理的，DB2 数据库系统可以决定授权的变化，并采取相应的行动。
- 当用户建立一个连接时，分配给用户的所有角色将可用，而授予角色的所有特权和权限也将可用。角色不能被显式地启用或禁用。
- 安全管理员可以把某一角色的管理委托他人。

在数据库中的所有 DB2 特权和权限都可以授予给角色。例如，一个角色可以被授予以下权限和特权：

- 数据库权限 DBADM, SECADM, LOAD 和 IMPLICIT_SCHEMA
- 数据库权限 CONNECT, CREATETAB, CREATE_NOT_FENCED, BINDADD, CREATE_EXTERNAL_ROUTINE 或者 QUIESCE_CONNECT
- 任何数据对象的特权（包括 CONTROL）

当用户连接到数据库时，一个用户的角色将自动启动并被赋予权限；您不需要使用 **SET ROLE** 语句来激活角色。例如，当您创建视图、物化查询表（MQT）、触发器、包或 SQL 例程时，您通过角色获得的权限起了作用。

角色没有拥有者。

与你的信息系统团队讨论后，您决定为应用程序开发人员创建一个角色，这样他们就可以连接到您的系统。您可以为这个角色赋予适当的权限以使应用开发人员能够部署存储过程、例程和其他数据对象，从而更方便地访问模式和数据库。

创建 **APP_DEV** 角色：

1. 在 **SAMPLE** 数据库对象上右键单击，然后选择“添加数据对象”->“角色”。一个新的角色被创建在数据项目资源管理器中。
2. 将该角色命名为 **APP_DEV**。
3. 给这个角色加特权：
 - a. 打开属性视图的“特权”选项卡。选定“数据库”标签。
 - b. 点击授予新特权按钮(🔑)。授予新特权的窗口被打开。
 - c. 选择 **BINDADD**, **CONNECT**, **CREATE_EXTERNAL_ROUTINE** 和 **CREATE_NOT_FENCED_ROUTINE** 选项，点击“确定”。数据库的这些特权被赋予给这个角色。
4. 保存。

这个角色被创建在物理数据模型中。如果您想要部署在 5.6 节生成的 DDL，您可以将用户分配到这个角色。

5.5.3 在物理数据模型中增加一个用户 ID

为了确保特定的用户在数据库部署后可以访问数据库，您应该将用户的 ID 添加到物理数据模型中。您还可以添加其他用户，不过让我们先从大多数 DB2 数据库创建的默认用户 **user1** 开始：

1. 在 **SAMPLE** 数据库上右键单击，选择“添加数据对象”->“用户”。一个新用户对象被创建在物理数据模型中。
2. 指定 **user1** 作为用户名。如图 5.6 所示。

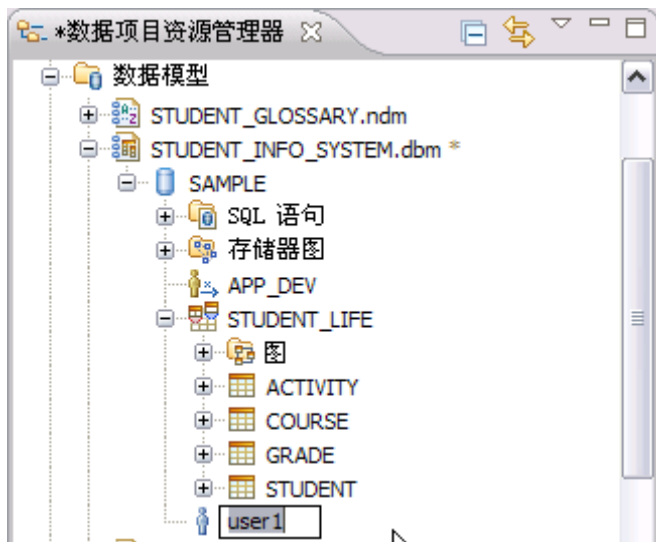


图 5.6 – 指定一个用户

3. 在属性视图的“特权”选项卡中，点击“数据库”区域。
4. 点击新建 (📁) 按钮打开授予新特权窗口。
5. 选择所有的特权，点击“确定”。
6. 保存。

特权被授予用户 `user1`，现在该用户可以完全访问该数据库。

5.5.4 验证物理数据模型

在部署物理数据模型之前，应该先验证它。当您验证该模型时，工作台需要检查该模型以确保数据模型可以生成有效的 DDL。验证过程不仅确保模型符合您创建的设计标准，还将确保它遵守通用的数据建模标准。这些“最佳实践”规则内置在产品中，使您更轻松地了解模型在部署之前是被很好地设计的。

验证物理数据模型：

1. 在数据项目资源管理器中右击数据库 `SAMPLE`，选择“分析模型”打开分析模型向导。默认情况下，所有的“物理数据模型”规则都被选中。
2. 取消“维模型”和“命名标准”的选项，点击“完成”。

注意：

维模型在本书中未被涵盖。关于维模型的更多信息和它的好处，请访问 IBM InfoSphere Data Architect 信息中心：

<http://publib.boulder.ibm.com/infocenter/rdahelp/v7r5/index.jsp>

模型被分析后，您可以在问题视图中看到一些警告和信息列表，如图 5.7 所示。

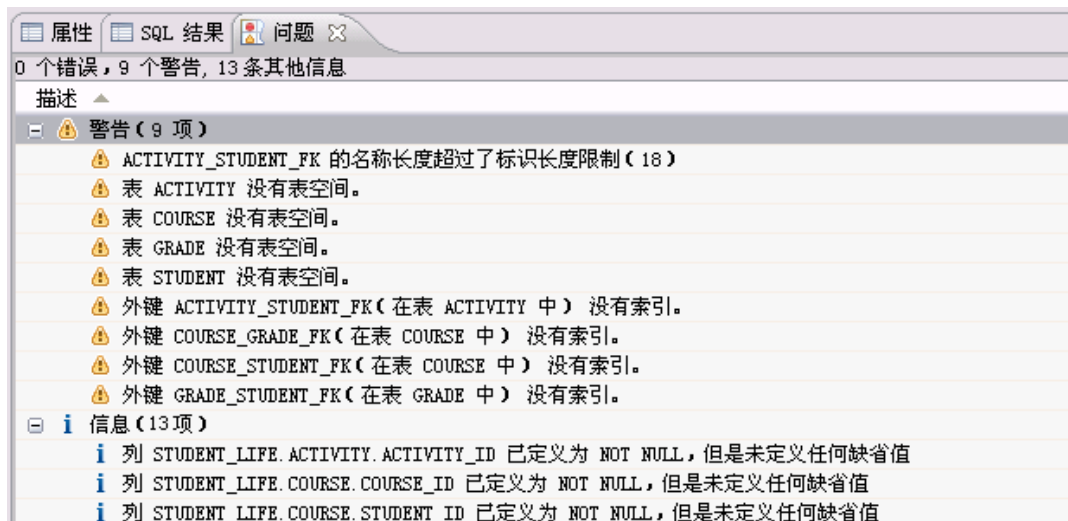


图 5.7 – 属性视图中的警告列表

对于本书中的练习，您不需要采取任何措施来纠正这些警告和消息；这些消息只是让您知道存在的潜在问题，它可能会影响性能或使得 SQL 语句失败。例如，当插入信息到图 5.7 中列出的数据对象时，更新操作可能会失败，因为这些列被设置为 NOT NULL，但是没有定义任何缺省值。由于您的数据模型相对来说较小，并且不会进入真实的生产环境，所以不需要创建表空间和索引来纠正这些警告。

然而，作为一个最佳实践，推荐您对每一个警告进行评估，以确保模式和数据库能拥有最佳的性能。

5.6 生成 DDL

现在，您已经生成了一个完整的物理数据模型，也已经知道了数据模型的创建、编辑和更新的过程。但是如何将这个数据库部署到一个测试或生产环境呢？

IBM InfoSphere Data Architect 能够从您所创建的模型中创建特定数据库的 *数据定义语言* (DDL) 脚本。在您生成 DDL 脚本后，可以将该脚本在服务器上运行，用户可以访问数据库中的这部分模型。

5.6.1 从数据库对象中生成 DDL 脚本

您从物理数据模型的数据库对象中生成 DDL 脚本。生成 DDL 向导将创建 SQL 语句，并部署数据库到服务器上。

让我们创建一个 DDL 脚本来部署您的新数据库：

1. 在数据源资源管理器中选中 SAMPLE 数据库右键单击，并选择“连接”。

注意：

如果您不打算在 DDL 生成过程中部署 DDL，您将不必连接到数据库。

2. 在数据项目资源管理器选中 SAMPLE 数据库右键单击，选择“生成 DDL”。生成 DDL 向导被打开。
3. 在向导的选项页中，确保所有的选项是被选中的，点击“下一步”。

表 5.1 解释了选项页中选项的含义：

选项	解释
检查模型	在脚本被创建之前模型需要被验证。
标准名称	任何对象用它的全名引用。例如， STUDENT_LIFE 模式中的 STUDENT 表使用 STUDENT_LIFE.STUDENT 。
加引号的标识	指定带引号的对象。如果“Table A”和“TABLE A”是不同的对象那么就可以选择本选项。
DROP 语句	在 DDL 脚本的开始包含 DROP 语句来删除已经存在的数据对象。
CREATE 语句	在 DDL 语句中包含 CREATE 语句用来创建或者重新创建数据对象。
COMMENT ON 语句	在数据对象属性视图的“文档”选项卡中包含的语句。
IN TABLESPACE 子句	从存储模型或是默认的表空间获取信息，并把它添加到 DDL 脚本中。

表 5.1 – 可以加入到 DDL 脚本中的模型元素

- 在对象页保留默认设置，点击“下一步”。
- 在保存并运行 DDL 页面，指定如下的文件名：**DEPLOY_SYS.sql**。

注意：

DDL 脚本显示在预览 DDL 区域中。使用该区域来验证您的脚本中包含的所有 SQL，您部署信息系统到服务器时需要这些 SQL，SQL 的第一部分确保了服务器上不存在这些数据对象。然后，SQL 将重新创建这些数据对象，确保在服务器上创建您所指定的角色。

- 选择“在服务器上运行 DDL”选项，点击“下一步”。
- 选择 **SAMPLE** 数据库完成选择连接页面。
- 在摘要页检查您的选择。在这个页面不能改变您的选择。如果您准备生成脚本，点击“完成”。

DDL 脚本生成被保存到数据设计项目中的“SQL 脚本”文件夹中，如图 5.8 所示。

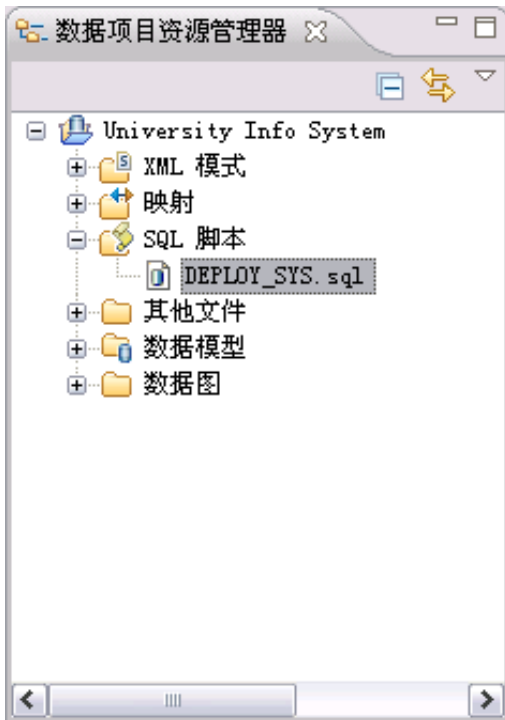


图 5.8 – 在 SQL 脚本文件夹中保存的 DDL 脚本

DDL 脚本也同时在服务器上运行。这就将您的新数据库和模式部署到 DB2 系统中，它可以用数据填充，并且应用程序开发人员可以基于数据创建应用程序来进行查询和报告。要验证模式已经部署，需要打开数据源资源管理器视图，展开 **SAMPLE** 节点和数据库。找到并展开“模式”文件夹，可以看到模式 **STUDENT_LIFE** 已经在数据库中创建，如图 5.9 所示。

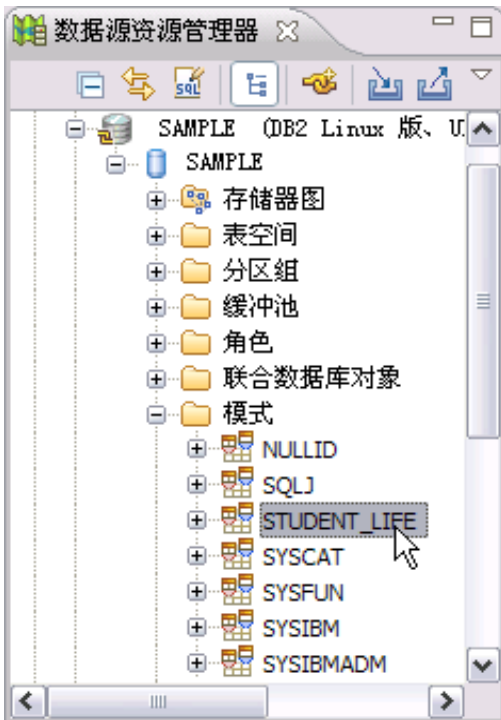


图 5.9 – 新部署的模式

5.7 练习

为物理数据模型生成一个新的 DDL 脚本：

1. 在生成 DDL 向导的选项页上，确保不选择如下选项：
 - DROP 语句
 - COMMENT ON 语句
 - 使用域（如果存在）
2. 在生成 DDL 向导的对象页上，仅仅生成如下对象的语句：
 - 表
 - 特权
 - 主键约束
 - 角色
3. 完成保存并运行 DDL 向导页。
 - a. 指定 EXERCISE_5_7.sql 作为文件名。
 - b. 选择“打开 DDL 文件以编辑”选项。
4. 完成向导。

5. 选择两个 SQL 文件（DEPLOY_SYS.sql 和 EXERCISE_5_7.sql），然后右键单击并选择“比较对象”->“互相”打开比较编辑器。在这两个文件中有什么区别？

5.8 总结

在本章中，您学习了物理数据模型。物理数据模型是数据设计过程的下一步，它们采用了逻辑设计，并帮助您为数据库和服务器起草了物理存储（表、列、索引和其它数据对象）。

您还学到了数据定义语言（DDL）文件，以及如何用它们来对 IBM InfoSphere Data Architect 中创建的数据模型进行部署。DDL 脚本包含了删除、创建或修改数据对象的操作中必须的查询。

5.9 本章回顾

1. 物理数据模型是什么？
2. 物理数据模型可以对下面的一个或者所有的对象进行建模：
 - a. 表
 - b. 角色
 - c. 模式
 - d. 以上所有
3. 物理数据模型中的什么对象可以生成 DDL 脚本？
4. 正误判断：在工作区创建 DDL 脚本之前您不能预览它。
5. 存储建模的目的是什么？
6. 在物理数据模型中可以创建什么样的存储模型对象？
7. 正误判断：在您创建一个物理数据模型之前必须创建一个逻辑数据模型。您不能从头开始创建一个物理数据模型。

5.10 下节展望

恭喜您，您已经设计并部署了一个完整的数据模型。

在下一章，您将要学到如何生成关于您的数据模型信息的报告。

第三部分 – 迭代设计：报告、反向设计和模型分析

6

第六章 – 报告的创建，模型的导入和导出

创建报告是 IBM InfoSphere Data Architect 的一个重要功能。报告中包含的信息可以被复制、打印，并作为物理文件进行存储及分发。报告在许多组织中被用于提供合规性信息。

IBM InfoSphere Data Architect 中的模型导入/导出功能能够帮助您确保数据模型和来源于其它建模、设计或报告工具的模型相互兼容。导入功能帮助您从其它的数据建模工具中导入现有的可兼容数据模型，而导出功能则能够帮助您将现有的数据模型导出为可被其它建模工具所兼容的模型。

6.1 报告的创建，模型的导入和导出：概述

当您创建报告时，报告的内容既可以是完整的模型，也可以是部分模型，比如模型中的对象或关系等。IBM InfoSphere Data Architect 为各种模型提供了一系列丰富的内置报告模板，如逻辑数据模型、物理数据模型、域模型和词汇表模型等。图 6.1 描绘了创建报告和模型的导入、导出的基本情况。

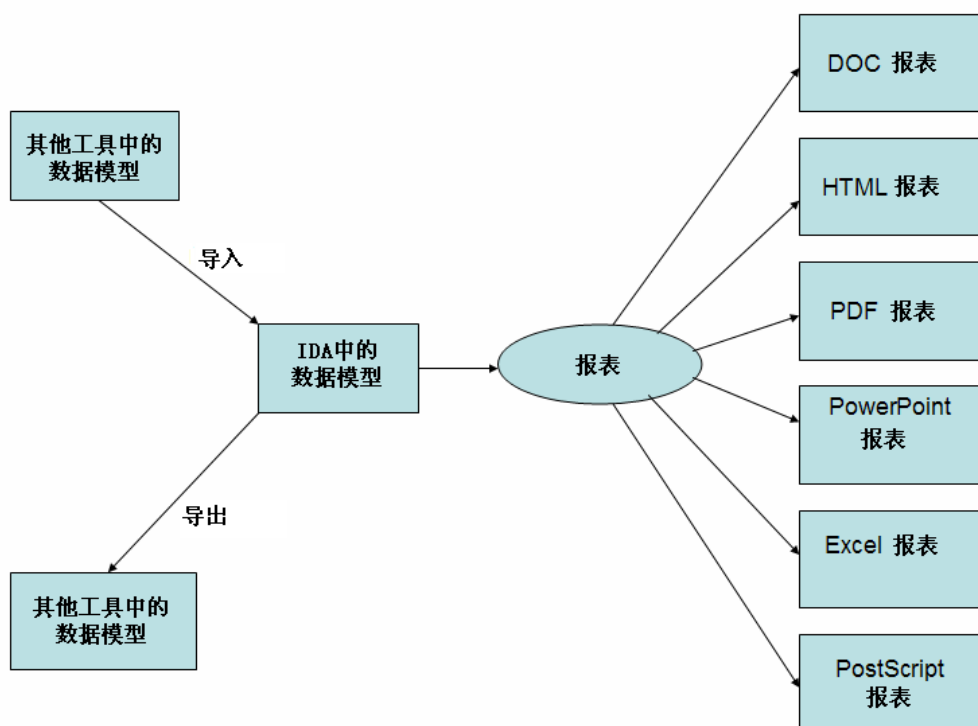


图6.1 – 创建报告，模型的导入、导出

图6.1的左半部分描述了 IBM InfoSphere Data Architect 的模型导入、导出功能。通过导入功能，您可以将由其他建模工具所构建的模型转换成 InfoSphere Data Architect 中的物理数据模型或逻辑数据模型；而通过导出功能，您则可以将 InfoSphere Data Architect 中的物理数据模型或逻辑数据模型转换成被其它工具所支持的模型。

图 6.1 的右半部分描述了 IBM InfoSphere Data Architect 的报告创建功能。通过该功能，您可以为逻辑数据模型、物理数据模型、域模型和词汇表模型创建 6 种不同格式的报告。这些格式包括 HTML、PDF、PowerPoint、Excel、Word (.doc)和 PostScript。

注意：

只有物理数据模型、逻辑数据模型和域模型才允许被导入、导出。

6.2 洞察报告功能

IBM InfoSphere Data Architect 的报告创建功能能够创建那些可以帮助您分析模型和包的报告。同时，它提供了丰富的内置报告模板，可以满足您对各种报告样式的需求。在下一章节，您将了解到这些丰富的内置报告模板的相关信息。

IBM InfoSphere Data Architect 支持的报告类型有两种：

- Business Intelligence Reporting Tool (BIRT) 报告
- XSLT 报告

在 IBM InfoSphere Data Architect 中，*BIRT 报告*被广泛使用。它提供了更加灵活、丰富的配置选项供您选择，并且支持多种输出格式。

*XSLT 报告*仅支持 PDF 作为其输出格式。

IBM InfoSphere Data Architect 为逻辑数据模型、物理数据模型、词汇表模型和域模型提供多种内置的报告模板。BIRT 报告模板的文件后缀为 **.rptdesign**，而 XSLT 报告的文件后缀为 **.xsl**。

注意：

您还可以创建自己的报告模板。在本章后面的练习部分，您将创建一个自定义的报告模板。

6.3 创建 BIRT 报告

现在，让我们开始学习如何为已有的模型创建 BIRT 报告。我们将利用 *Physical Data Model* 报告类型为物理数据模型 **STUDENT_INFO_SYSTEM.dbm** 创建一个 HTML 格式的报告。

首先，您将学习如何为物理数据模型创建报告。在此之后，您还将学习如何利用报告设计透视图指定报告所需的数据对象来定制您的报告。

6.3.1 创建基本物理数据模型报告

在创建了一个报告模板之后，您可以重用该模板来创建具有相同信息和基本格式的报告。这样，您可以为不同的业务领域的用户提供具有统一用户体验的报告。

利用报告配置窗口为已有的物理数据模型创建报告：

1. 点击“运行”->“报告”->“报告配置”，打开报告配置窗口。
2. 在左侧面板，通过双击“BIRT 报告”选项，创建一个新的报告配置。
3. 设置报告配置选项。
 - a. 在“名称”字段中，输入 **SDT_Info_Sys**。
 - b. 选择“内置”单选按钮，然后单击“浏览”，打开“选择内置报告”对话框。
 - c. 展开“物理数据模型报告”结点，并选择“物理数据模型报告”，然后点击“确定”来将报告加入到配置中。
 - d. 在“报告数据”的“数据源”字段中，选中“Data Source”项，然后点击“添加”按钮，打开“装入资源”窗口。

- e. 点击“浏览工作空间”，然后展开 *University Info System* 节点并选择 **STUDENT_INFO_SYSTEM.dbm**，点击“确定”。
 - f. 点击“确定”来将物理数据模型加入到配置中。
 - g. 点击“工作空间”按钮来指定报告存储位置。
 - h. 在“文件名”字段中输入 **SDT_Info_Sys_Rpt**，点击“确定”。
4. 点击“报告”按钮来生成报告。

您将在浏览器中看到生成的报告，它展示了物理模型中所包含的信息。

同时，您还可以对报告设计进行定制，让它能够更好的展示数据模型。现在，让我们来创建一个新的报告配置来展示物理数据模型中有关列的重要信息。

在创建报告配置之前，您需要将工作台切换到“报告设计”透视图上。下面的章节中所要创建的报告模板能展示物理模型相关的信息，并可以将它与您的团队进行分享。这个报告能够很好地分析数据模型，并将其以更加清晰易懂的格式展现出来。

6.3.2 配置报告设计环境

首先，让我们配置报告设计环境：

1. 打开“报告浏览器”视图：
 - a. 点击“窗口”->“显示视图”->“其它”，打开“显示视图”对话框。
 - b. 展开“报告”结点，选择“报告浏览器”，在“报告浏览器”视图中，您将看到报告模板列表。
2. 将“空白物理模型报告 (BlankPhysicalModel.rptdesign)”复制到数据设计项目的“其他文件”文件夹下。
 - a. 在“报告浏览器”视图中，展开“物理数据模型”结点，右键单击“空白物理模型报告 (BlankPhysicalModel.rptdesign)”，然后选择“复制”。
 - b. 在“数据项目资源管理器”视图中，展开 *University Info System* 文件夹，右键单击“其他文件”文件夹，然后选择“粘贴”。
 - c. 在“数据项目资源管理器”视图中，右键单击 *University Info System* -> “其他文件” -> **BlankPhysicalModel.rptdesign**，然后选择“重命名”。
 - d. 在“重命名资源”对话框中，输入新的资源名称 **SDT_Info_Sys_custom**，然后点击“确定”。
3. 打开“报告设计”透视图：
 - a. 在主菜单中，点击“窗口”->“打开透视图”->“其它”，弹出“打开透视图”对话框。
 - b. 在“打开透视图”对话框中，选择“报告设计”。此时，工作台切换到“报告设计”透视图。在一个报告设计在报告编辑器中被打开前，“选用板”、“数据资源管理器”、“大纲”及“属性编辑器”视图都是空的。并非所有的视图都可见，例如“导航器”和“大纲”视图同时仅一个可见。您可以通过点击选项卡，在不同的视图间作切换。
4. 在“报告编辑器”中打开 **SDT_Info_Sys_custom.rptdesign**：

- a. 打开“导航器”视图。
- b. 展开“University Info System”文件夹，双击 **SDT_Info_Sys_custom.rptdesign**，报告模板将在“报告编辑器”中显示，如图 6.2 所示。

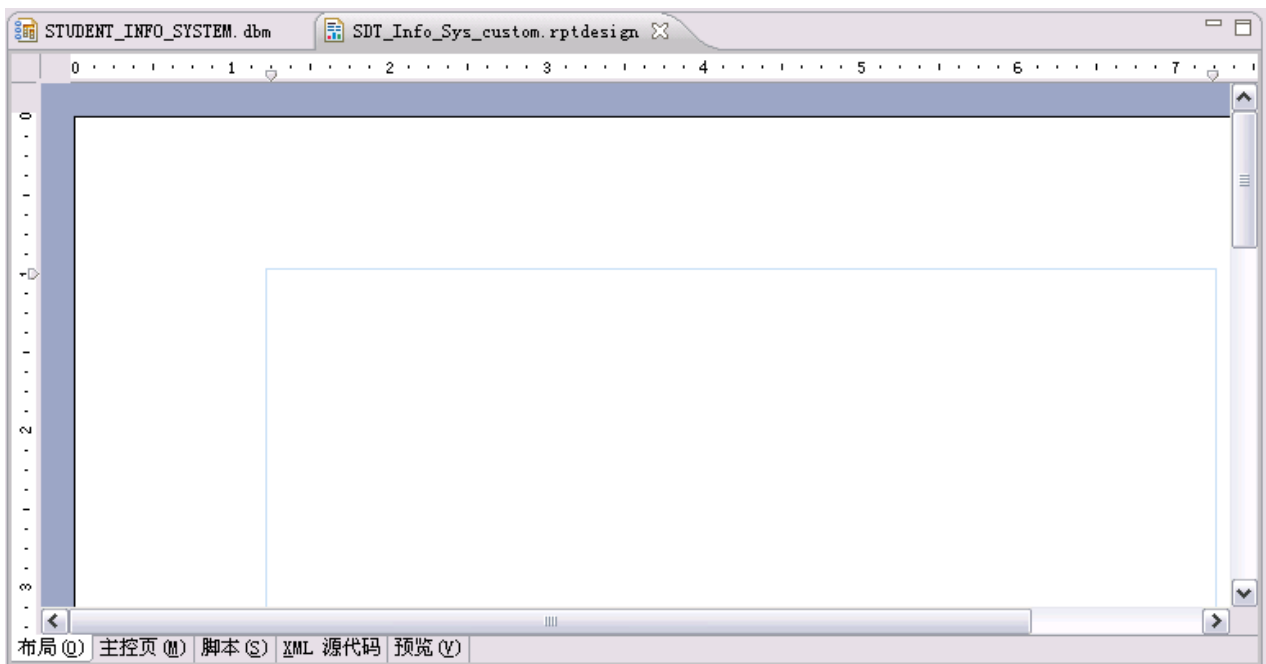


图6.2 – 在报告编辑器中打开报告

5. 将 **STUDENT_INFO_SYSTEM.dbm** 设置为 BIRT 报告的数据源。
 - a. 选择“数据资源管理器”视图。
 - b. 在“数据资源管理器”视图中，展开“数据源”文件夹并用右键单击“数据源”对象，然后选择“编辑”。打开“编辑数据源 – Data Source”对话框。
 - c. 在“模型”页面的“实例模型”区域，点击“添加”，打开“装入资源”对话框。
 - d. 在“装入资源”对话框中，点击“浏览工作空间”，打开“选择文件”对话框。
 - e. 选择 University Info System 下的 **STUDENT_INFO_SYSTEM.dbm**，点击“确定”，如图 6.3 所示。

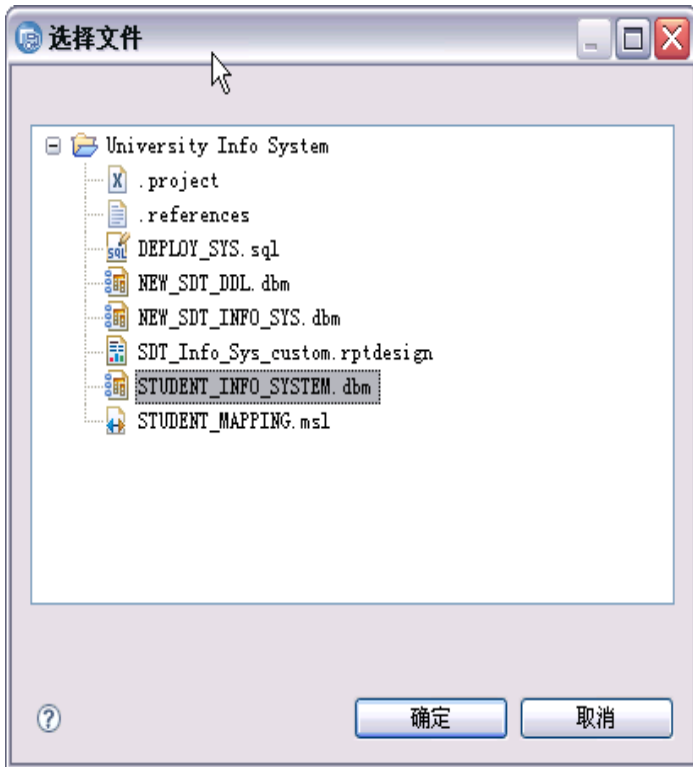


图6.3 – 选择物理数据模型

- f. 点击“确定”来加载数据模型资源。
 - g. 点击“确定”来保存数据源设置。
6. 保存。

此时，您已完成报告设计环境的配置，并将物理数据模型 **STUDENT_INFO_SYSTEM** 设置为 BIRT 数据源。

6.3.3 象对据数加添告报为

为了在报告中展示数据模型的信息，您需要将所要展示的数据对象与报告相关联。首先，让我们从展示数据表的列相关信息开始吧。

1. 首先，为报告添加标题：
 - a. 在工作台左上方面板中，打开“选用板”视图。
 - b. 在“选用板”视图选中“标签”元素，将其拖拽到“报告编辑器”的“布局”页面。
 - c. 双击“报告编辑器”的标签，然后输入：**Table and column report**。
 - d. 在“属性编辑器”视图中，选择“常规”选项卡。
 - e. 将字体设置为“粗体”。
 - f. 将文本设置为“居中对齐”。“常规”选项卡如图 6.4 所示。

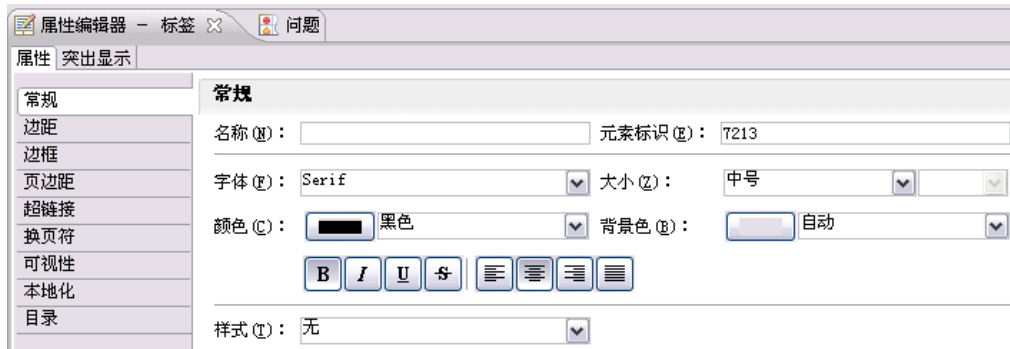


图6.4 – 设置标签元素的文本属性

2. 添加 BIRT 报告表。

- a. 在“选用板”视图选中表元素，并将其拖拽到“布局”页面中的报告标题元素下方。打开“插入表”对话框。
- b. 将“列数”设置为2。
- c. 在“数据集”的下拉列表中，选择“Column”。这样，一个包含两列的表将被插入到报告中，包含一个表头行、一个详细信息行及一个表脚行。并且，该报表与列数据集相关联。

注意：

列数据集是报告设计中预先定义的数据集。在“数据资源管理器”视图中，您将看到预定义的数据集。

3. 为报告表指定名称，并更新表样式：

- a. 打开“大纲”视图。
- b. 点击 *SDT_Info_Sys_custom.rptdesign* -> “主体” -> “表”。

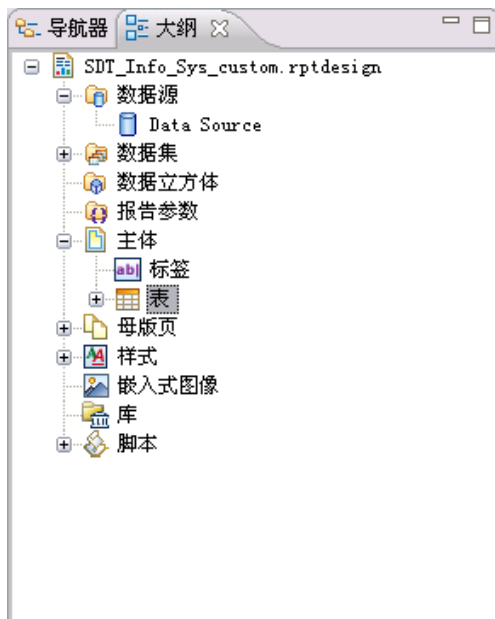


图6.5 – 在大纲视图中选择报告对象

- c. 在“属性编辑器”视图中，选择“常规”选项卡。
- d. 在“名称”字段中，输入 **Column Report**。

注意:

报表的样式可以通过修改报表中的元素进行修改，您可以选择报表中的元素，并在“属性编辑器”视图中进行属性修改。

4. 将“Column”数据集中“Name”和“Full Data Type Information”元素添加到报表中。
 - a. 打开“数据资源管理器”视图。
 - b. 展开“数据集”->“Column”，并将“Name”元素拖拽到报表中详细信息行的左边列上。
 - c. 将“Full Data Type Infomation”元素拖拽到报表中详细信息行的右边列上。

此时报告如图6.6所示。

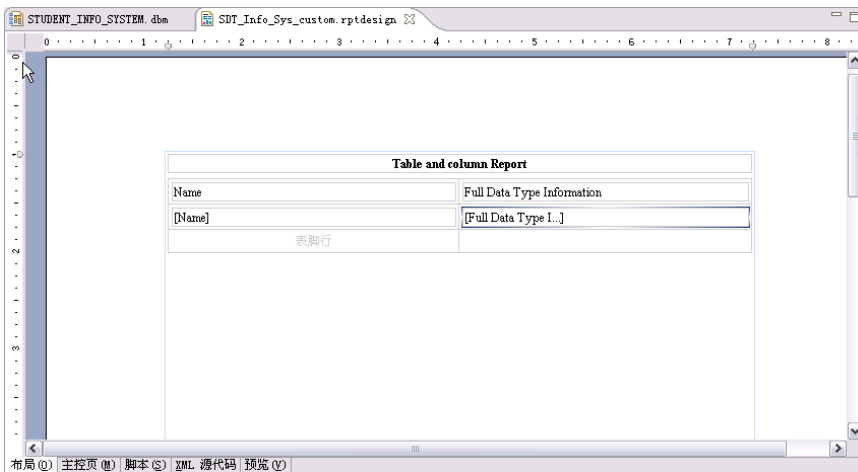
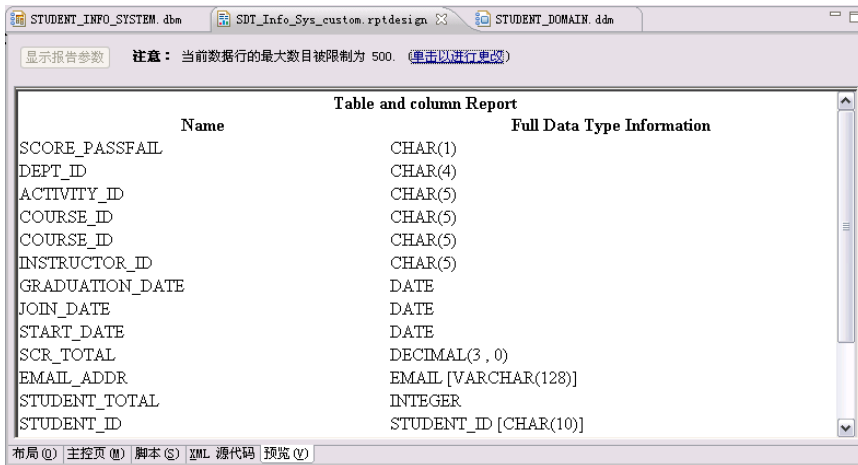


图 6.6 – 添加报告元素

5. 设置表属性，使数据能够按数据类型及列名排序。
 - a. 打开“大纲”视图。
 - b. 展开 *SDT_Info_Sys_custom.rptdesign* -> “主体”，选择“表 – Column Report”，并激活“属性编辑器”视图。
 - c. 在“属性编辑器”的顶部，选中“排序”选项卡，点击“Add”，打开“新建排序键”对话框。
 - d. 在“键”字段的下拉框中选择“Full Data Type Information”，然后点击“确定”。
 - e. 点击“添加”按钮，为 Name 添加排序键。
 - f. 在“键”字段的下拉框中选择 Name，然后点击“确定”。

您已经为报表设置好了排序键，生成的报告中的数据列将根据数据类型及名称进行排序。

6. 点击“报告编辑器”下方的“预览”选项卡, 生成的报告预览如图 6.7 所示。



显示报告参数 注意: 当前数据行的最大数目被限制为 500. (单击以进行更改)

Table and column Report	
Name	Full Data Type Information
SCORE_PASSFAIL	CHAR(1)
DEPT_ID	CHAR(4)
ACTIVITY_ID	CHAR(5)
COURSE_ID	CHAR(5)
COURSE_ID	CHAR(5)
INSTRUCTOR_ID	CHAR(5)
GRADUATION_DATE	DATE
JOIN_DATE	DATE
START_DATE	DATE
SCR_TOTAL	DECIMAL(3, 0)
EMAIL_ADDR	EMAIL [VARCHAR(128)]
STUDENT_TOTAL	INTEGER
STUDENT_ID	STUDENT_ID [CHAR(10)]

布局 (U) | 主控页 (M) | 脚本 (S) | XML 源代码 | 预览 (V)

图6.7 – 报告预览

您已经创建了一个基本的 BIRT 报告, 在生成正式报告前您可以在“预览”选项卡中审核您的设计。

现在, 您可以通过进一步修改该报告模板来对数据列进行重新排列, 例如通过在原有的报表中嵌套新的表来对列进行分组。

6.3.4 报告数据分组

让我们根据表以及对数据列按数据类型和名称排序的结果对列进行分组。在报告中将数据进行分组:

1. 点击“报告编辑器”下方的“布局”选项卡。
2. 选择表。当鼠标停在表上时, 表标签将会在编辑器上显示, 如图 6.8 所示。

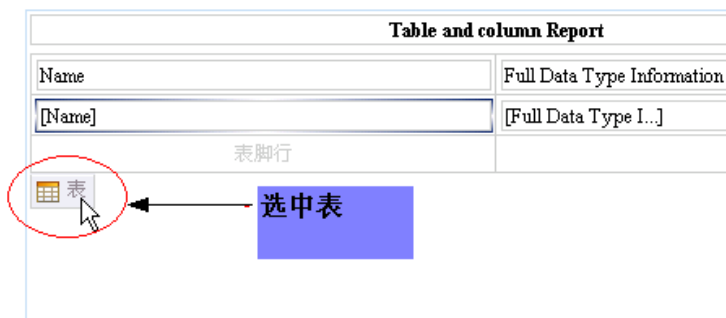


图6.8 – 选择表标签

在报表的左侧及上方, 出现提示单元格。

3. 为详细信息行添加排序组:
 - a. 右键单击“详细信息行”, 在其提示单元格中选择“插入组”。提示单元格位于报表左侧中部, 如图 6.9 所示。



图6.9 – 为详细信息提示单元格添加组

“新建组”对话框弹出。

- b. 在“名称”字段中输入 **GroupByTable**。
- c. 在“分组依据”下拉框中选择“Source Table”。
- d. 点击“确定”。

此时，表中包含一个按照来源表进行归类的组。报告的布局如图 6.10 所示。

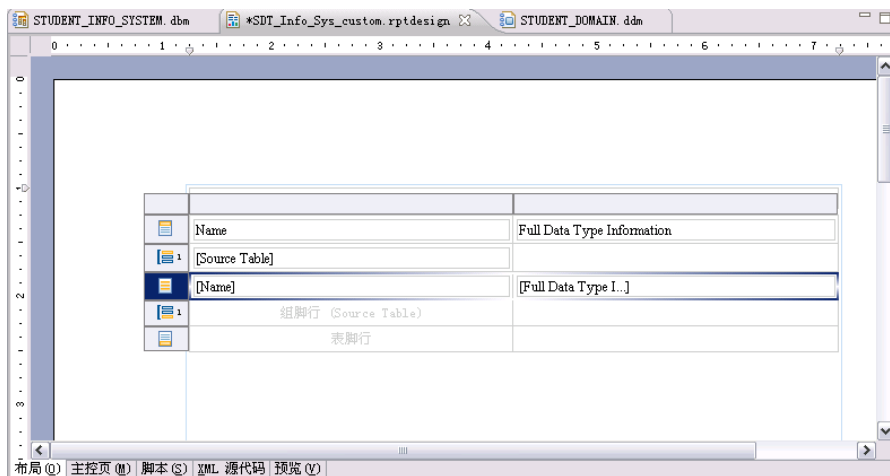


图6.10 – 为报告添加分组

4. 修改[Source Table]元素的样式:
 - a. 在“报告编辑器”中, 选择[Source Table]元素, 然后该元素的属性将展示在属性编辑器视图中。
 - b. 在“属性编辑器”视图的“属性”面板中, 选择“常规”选项卡。
 - c. 设置字体颜色为“栗色”, 样式为“粗体”。
5. 修改[Full Data Type Information]元素样式:
 - a. 在“报告编辑器”中, 选择[Full Data Type Information]元素。
 - b. 在“属性编辑器”视图的“属性”面板中, 选择“页边距”选项卡。
 - c. 在“左”字段中, 输入 20。
6. 修改[Name]元素样式:
 - a. 在“报告编辑器”中, 选择[Name]元素。
 - b. 在“属性编辑器”视图的“属性”面板中, 选择“页边距”选项卡。
 - c. 在“左”字段中, 输入 40。
7. 创建标题行:
 - a. 在“报告编辑器”中, 右键单击[Full Data Type Information]元素所在行的提示单元格。
 - b. 选择“插入”->“行”->“下面”, 一个新的表头行被添加到报表中。
 - c. 在“报告编辑器”中, 将 Name 标签拖拽至新建行的左侧列中, 将 Full Data Type Information 拖拽至新建行的右侧列中。
 - d. 删除新的空白表头行: 右键单击表头行左侧的提示单元格, 然后点击“删除”。
 - e. 修改 Name 标签的样式: 选中 Name 元素, 在“常规”选项卡中, 将字体设为粗体; 在“页边距”选项卡中, 将左边距设为 40。

- f. 修改 *Full Data Type Information* 标签的样式：选中 *Full Data Type Information* 元素，在“常规”选项卡中，将字体设为“粗体”。

修改后的报告布局如图 6.11 所示。

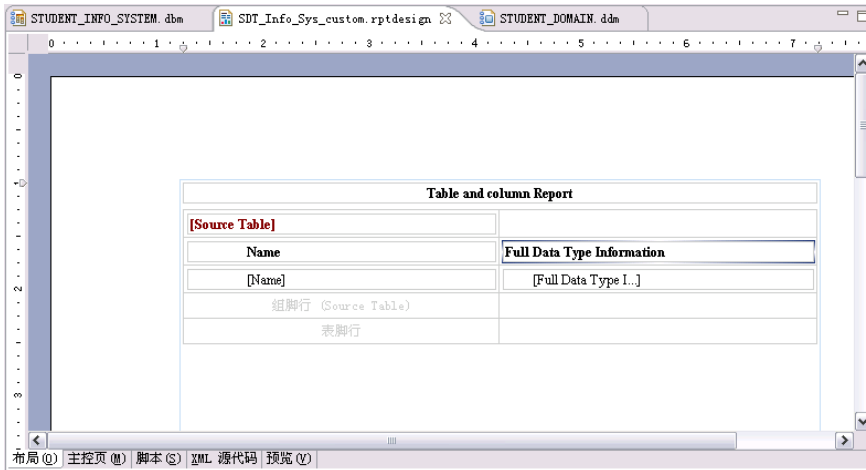


图6.11 – 报告数据分组及样式修改

8. 预览报告：打开“报告编辑器”的“预览”选项卡。报告如图 6.12 所示。

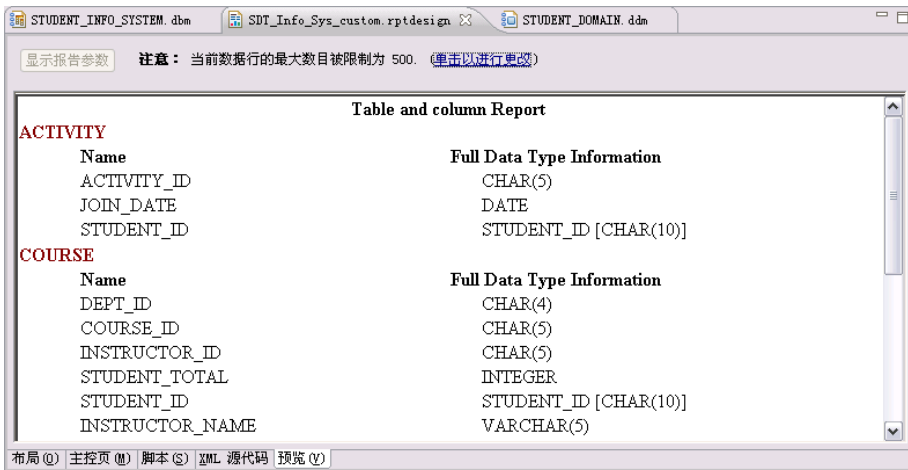


图6.12 – 预览修改后的报告

6.3.5 为报告添加动态文本

目前已经创建了一个基本的报告，您还可以进一步为报告添加动态的文本及样式来增强报告的展示效果。动态文本标签可以更加清晰地展示模型元素，并且其内容随着输入的变化而变化。

在您创建的报告中，您需要为表和列添加清楚的标签。只有这样，当您与团队中其它成员（或不同业务领域的相关人）共享该报告模板时，他们才可以更好地理解物理数据模型的设计。

对报告做如下修改：

1. 打开“报告编辑器”的“布局”选项卡。

2. 为表头行创建动态文本：
 - a. 在组头行中删除[Source Table]元素。
 - b. 从“选用板”视图中，将“动态文本”元素拖拽到您刚删除的[Source Table]元素所在的组头行单元格中。“表达式构建器”对话框弹出。
 - c. 在“表达式构建器”对话框的“表达式”字段中，输入如下表达式，然后点击“确定”：

"Table: " + row["Source Table"]

添加完表达式的动态文本元素如图 6.13 所示。

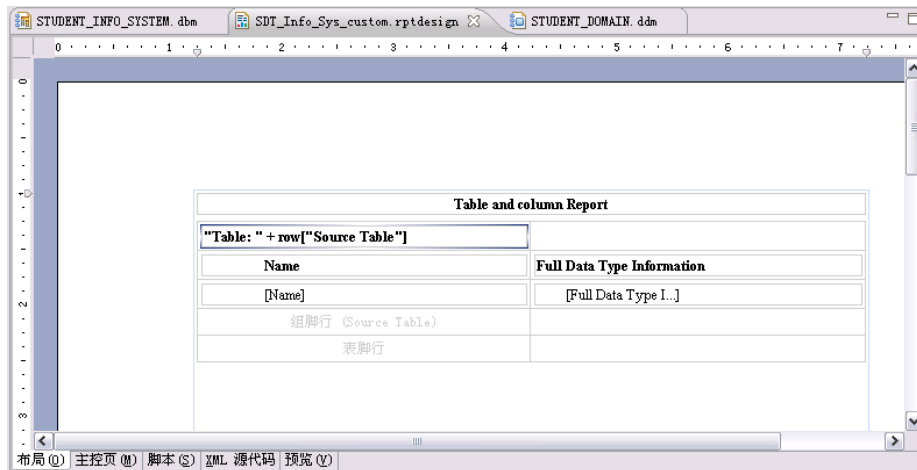


图6.13 – 为 BIRT 报告添加动态文本元素

3. 修改动态文本标签：
 - a. 在“报告编辑器”中选中动态文本标签。
 - b. 在“属性编辑器”视图中，打开“常规”选项卡，将字体设为“粗体”，颜色为“栗色”。
4. 预览报告。

报告如图 6.14 所示。

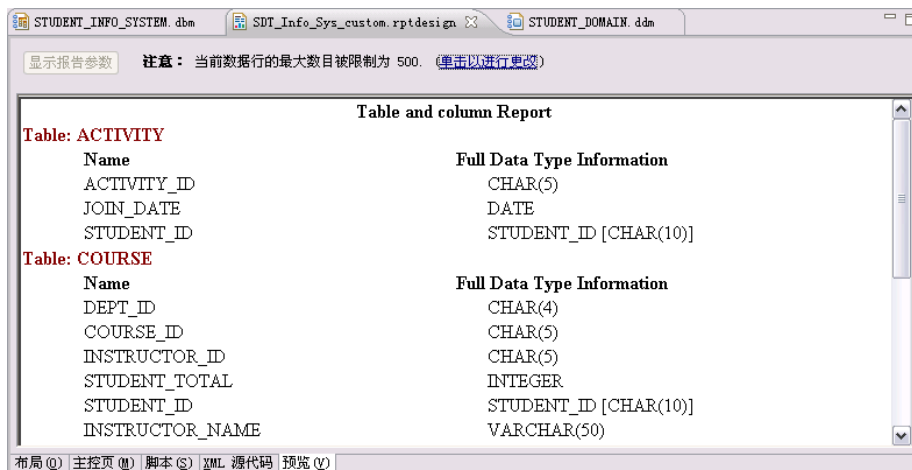


图6.14 – 预览编排过的报告

在报告里，信息被合理地分类，使其能够帮助您的团队更便利地识别模型中各种各样的对象。

6.3.6 从模板创建报告配置

现在，让我们开始创建报告配置，通过该配置您可以与团队一起共享报告。您同样能够重复地使用这个报告配置，只要对生成的模型做一些更改，您就能够以最小的代价创建出符合需求的报告。

利用 **BIRT** 模板生成报告：

1. 打开“导航器”视图。
2. 选择报告及创建配置：
 - a. 右键单击 **SDT_Info_Sys_custom.rptdesign** 模板，选择“报告为”->“报告配置”。
 - b. 通过双击“**BIRT 报告**”结点，创建一个新的报告配置。
 - c. 在“名称”字段中，输入 **SDT_Info_Sys**。
 - d. 选中“位置”单选按钮，然后点击“工作空间”按钮来打开“选择报告”对话框。
 - e. 展开 *University Info System* 结点，选择你刚创建的 **SDT_Info_Sys_custom.rptdesign** 模板，然后点击“确定”把它加入到报告配置中。
 - f. 指定报告输出位置，如 **C:\Temp\IDA_Reports\Student_Info_Sys.html**。
 - g. 点击“应用”保存配置。
2. 点击“报告”按钮，生成报告。

此时，报告配置被执行。由于选择的输出的类型为 **HTML**，报告将在默认的浏览器中打开，同时生成的报告文件保存在所指定的目标目录。

该报告展示了模型中表和列的信息。**HTML** 报告如 [图 6.15](#) 所示。

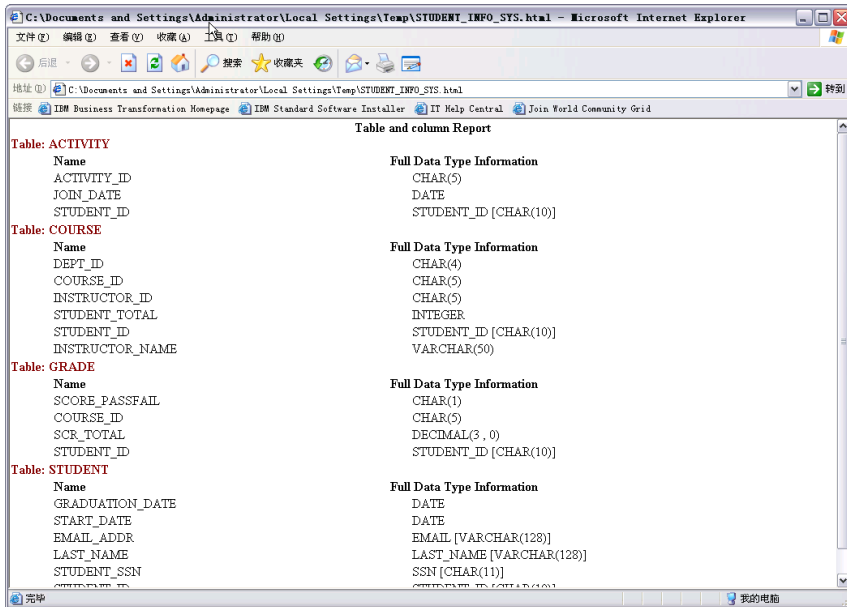


图 6.15 – 报告在默认浏览器中展示

请注意：该物理模型报告并非展示数据库中的数据，它能够帮助团队成员更好地分析和理解数据模型。

注意：

在运行报告配置后，配置将被自动地保存在工作空间中。如果您的模型发生改变，您可以点击“运行”->“报告”->“报告配置”，然后选择一个目标报告配置来运行。

6.4 创建 XSLT 报告

接下来，我们将学习创建另一种报告：XSLT 报告。在本小节中创建的报告，将用于展示逻辑数据模型中所包含的信息。

创建 XSLT 报告：

1. 通过点击“窗口”->“打开透视图”->“其它”，切换至“数据”透视图。
2. 在主菜单中，通过点击“运行”->“报告”->“报告配置”，打开“报告配置”对话框。
3. 通过双击“数据模型标准 XSLT 报告”结点，创建一个新的 XSLT 报告配置。
4. 配置报告选项：
 - a. 在“名称”字段，输入 **SDT_Info_Sys_XSLT**。
 - b. 在“位置”字段，点击“浏览”，然后选择逻辑数据模型 *STUDENT_INFO_SYSTEM.Idm*。
 - c. 在“元素”字段，点击“浏览”，然后选择包 *STUDENT INFO SYSTEM*。
 - d. 选中“内置”单选按钮，点击“浏览”，然后打开“选择内置报告”对话框。
 - e. 展开“逻辑数据模型（标准 XSLT 报告）”结点，选择“标准样本逻辑模型报告”，然后点击“确定”。

- f. 在“报告输出”位置指定报告保存路径及名称，如
C:\Temp\IDA_Reports\SDT_Info_Sys_XSLT.pdf。
5. 点击“应用”保存配置。
6. 点击“报告”来运行该配置。

报告配置被执行后，生成的报告展示了逻辑模型中的数据对象的属性信息，并包含通过 IBM InfoSphere Data Architect 所构建的模型图。这样，团队中的非技术人员就能更好地理解数据对象之间的关系。

6.5 导入的型模/导出

导入/导出功能帮助您将其它工具创建的模型导入到 InfoSphere Data Architect 中，或者将 InfoSphere Data Architect 中创建的模型导出为其它工具所兼容的模型格式。这些工具包括 IBM Cognos Framework Manager、IBM Rational Rose 和 IBM Rational Software Architect 等。

当您导入/导出模型时，模型中的元数据将被转换成目标工具所兼容的格式。

6.5.1 出导的型模

在本节中，您将学习如何通过 IBM InfoSphere Data Architect 将逻辑数据模型导出为 XML 文件。这样，导出的模型可以被 XML 工具所重用并生成报告。

导出数据模型：

1. 点击主菜单：“文件”->“导出”，打开“导出”向导
2. 展开“数据”结点，选择“数据模型导出向导”，然后点击“下一步”。
3. 完成“选择模型”向导页：
 - a. 在“模型类型”下拉框中，选择“W3C XML Schema 1.0 (XSD)”。
 - b. 点击“模型”字段旁的“浏览”按钮，然后选择 *STUDENT_INFO_SYSTEM.Idm*。
 - c. 点击“导出到”字段旁的“浏览”按钮，然后指定目标文件名称为 **SDT_Info_Sys_Exp.xml**，如图 6.16 所示。

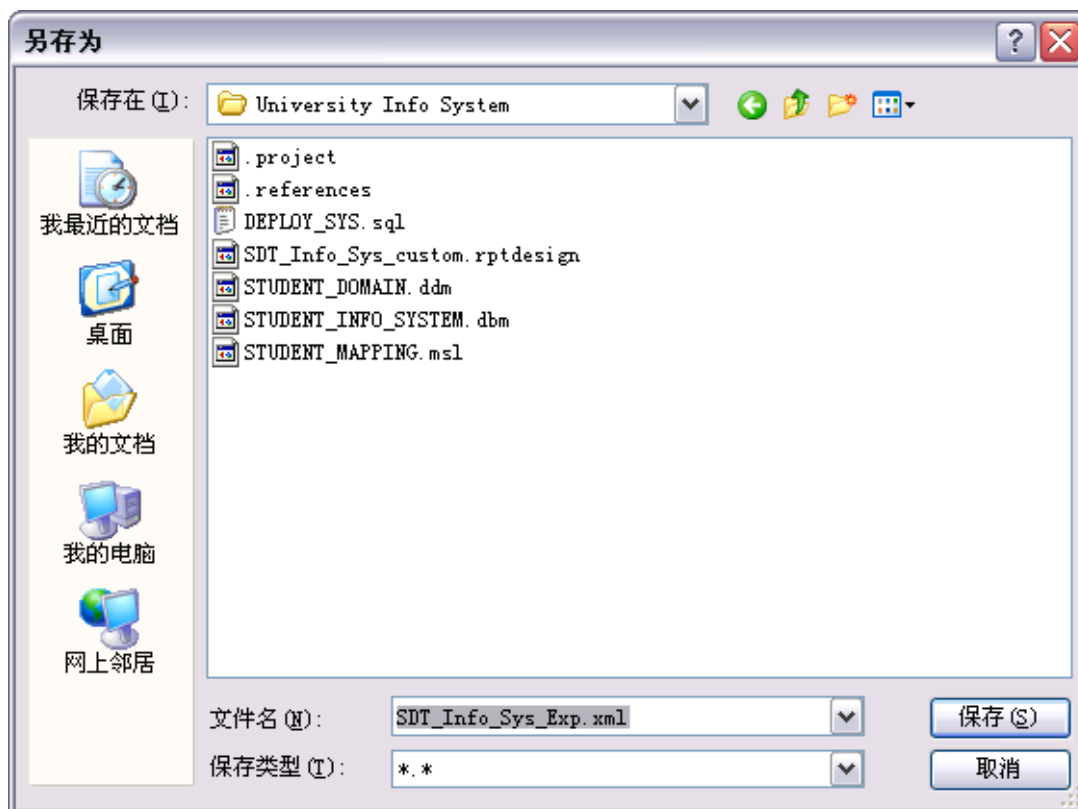


图6.16 – 导出并保存数据模型

- d. 点击“下一步”。
4. 在“选择选项”向导页，通过点击“下一步”来导出数据模型。
5. 点击“完成”。

导出的文件在项目的“XML 模式”文件夹下。

注意：

由于导出的文件格式为 XML，因此不包含模型图。

6.5.2 模型的导入

在上一节中您学习了如何导出模型，现在让我们学习如何导入模型。IBM InfoSphere Data Architect 能够读取许多来源于其它工具的信息，以便您更好地改进和修改模型。

让我们导入模型：

1. 在主菜单中点击“文件”->“导入”，打开“导入”向导。
2. 展开“数据”结点并选择“数据模型导入向导”，然后点击“下一步”。
3. 完成“选择模型”向导页：

- a. 在“模型类型”下拉框中，选择 *W3C XML Schema 1.0 (XSD)*。
- b. 点击“模型”字段旁的“浏览”按钮，然后选择 *SDT_Info_Sys_Exp.xml*。
- c. 点击“目标项目”字段旁的“浏览”按钮，然后选择数据设计项目 *University Info System*。
- d. 在“模型类型”下拉框中选择“仅限逻辑”。生成的模型为逻辑数据模型，而您可以修改该模型并将其转换为物理数据模型。
- e. 在“文件名”字段中输入 *DT_Info_Sys_Imp*。
- f. 点击“下一步”。

4. 在“选择选项”向导页，点击“下一步”。

数据模型将被转换为逻辑数据模型并添加到数据设计项目中。

5. 点击“完成”以退出向导。

该模型位于“数据模型”文件夹。当您展开逻辑数据模型时，能看到数据模型已经完成。

6.6 练习

创建一个定制的报告模板 *SDT_Info_Sys_Diag.rptdesign*，该模板将捕获逻辑数据模型 *STUDENT_INFO_SYSTEM.ldm* 中的图，包括该图中实体的信息。可以通过复制和修改一个空白图报告来创建这个新的报告。

1. 在“报告编辑器”中插入一个包含 1 个数据列和 2 个详细信息行的表元素，并指定一个图作为数据集。
2. 在第一详细信息行中，插入图像，并设置类型为动态图像。通过点击“选择图像数据”按钮，为信息行指定所要绑定的数据集。
3. 点击“添加”，新建一个数据绑定，并设置数据类型为 *Blob*，然后点击“表达式”字段旁的按钮，打开“表达式构建器”对话框。
4. 在“表达式构建器”对话框中，输入如下表达式：**dataSetRow[“Image”]**。
5. 在添加了表达式之后，选中列绑定数据集左侧的复选框，然后点击“确定”。
6. 在第二详细信息行中插入一个新的表元素，设置其 *列数*为 1，*详细信息行数*为 2，*数据集*为无。
7. 在新建的嵌套表的第一详细信息行中，插入一个 1 行 2 列的网格。
8. 在网格元素的左侧列中，插入标签 **Entity**。
9. 将“数据资源管理器”中的 *Entity -> Name* 对象插入到网格的右侧列中。
10. 切换到“数据”透视图，创建一个报告配置。报告的数据源设置为逻辑数据模型 *STUDENT_INFO_SYSTEM.ldm*，并指定输出的格式为 PDF，然后运行报告配置。生成的报告如图 6.17 所示。

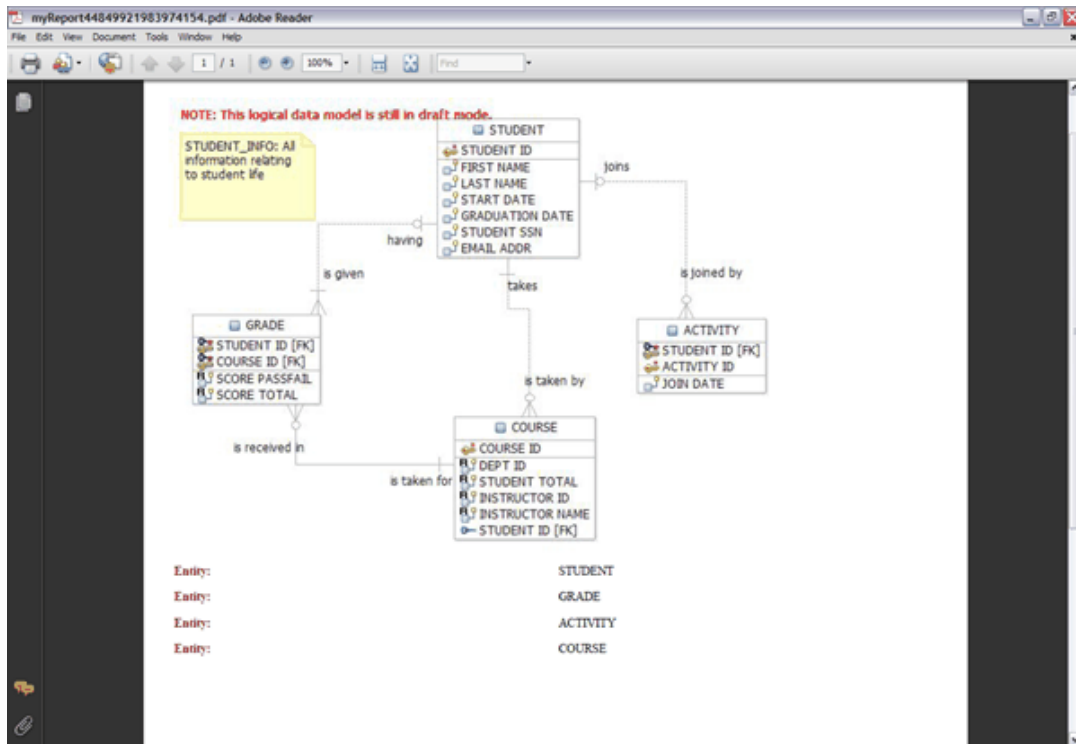


图6.17 – 生成的图报告

6.7 总结

在这个章节中，您学习了创建报告的基本知识，并知道如何通过 IBM InfoSphere Data Architect 提供的内置报告模板来定制满足不同需求的报告。

此外，您还学习了如何利用 IBM InfoSphere Data Architect 导入/导出数据模型。通过该功能，您可以将 InfoSphere Data Architect 所支持的数据模型导出为其它工具所支持的模型，或将其它工具所支持的模型导入为 InfoSphere Data Architect 所支持的模型。

6.8 本章回顾

1. BIRT 报告不支持以下哪种报告格式？
 - A. HTML
 - B. PDF
 - C. 纯文本
 - D. Microsoft Excel
2. 以下的哪些模型被允许导入或导出？
 - A. 物理数据模型
 - B. 词汇表模型

- C. 映射模型
 - D. 上述所有选项都是
3. 在模型导出时，不支持的验证选项有哪些？
- A. 基本的有效性验证
 - B. 普通的有效性验证
 - C. 无有效性验证
 - D. A 和 C
4. BIRT 报告模板的文件后缀名是什么？
- A. .xsl**
 - B. .xsl design**
 - C. .rpt design**
 - D. 上述所有选项都不是
5. 下面的哪个文件格式是 XSLT 报告所支持的？
- A. PDF
 - B. Microsoft Excel
 - C. 纯文本
 - D. HTML

7

第七章 – 反向设计

通常情况下，在对数据结构进行设计之前，企业的数据库上就已经存在了大量的模式，其中包含大量的数据库表。为了保证不影响已有的约束或丢失数据，在修改这些模式的时候，将需要大量的时间和精力。

然而，您可以通过已有的数据库创建物理数据模型。在对已有数据库进行反向设计之后，可以对获得的物理数据模型进行修改，并将修改的结果部署在已有数据库上，这使得对已有数据库的修改过程变得简单。本章将详细介绍如何通过反向设计来修改已有数据结构。

在本章中，您将了解到以下知识：

- 使用工作台对已有数据库进行反向设计
- 通过比较将变更同步到模型中
- 通过生成 DDL 脚本将数据模型变更部署到数据库中

7.1 反向设计：概述

当我们通过物理数据模型生成 DDL 脚本并将其部署在数据库上的时候，模型中的每一个对象将转变为对应数据库中的对象。反向设计则是将这个过程倒置过来：将数据库中的对象转变为物理数据模型中的对象。图 7.1 描述了反向设计的概况。

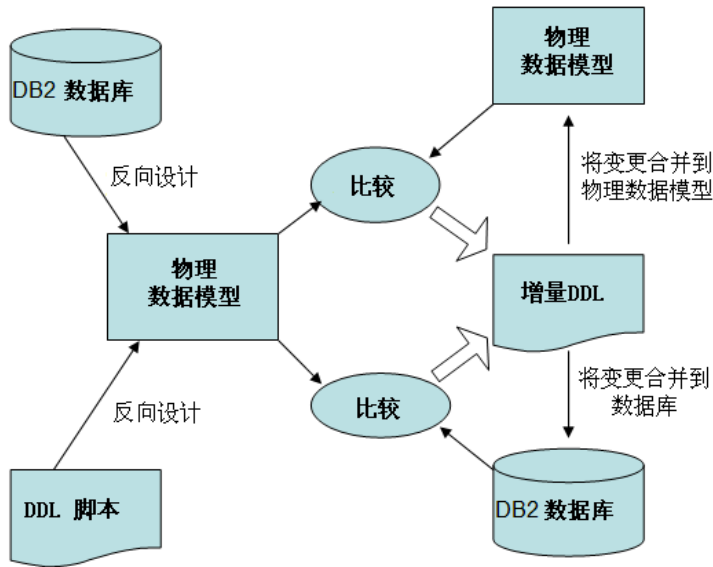


图 7.1 – 反向设计概述

图 7.1 的左侧部分表明，反向设计的源可以为数据库或 DDL 脚本，进而在数据项目资源管理器中创建物理数据模型(.dbm)，该数据模型将包含数据库中数据对象的信息。在通过反向设计生成物理数据模型后，就可以修改数据模型，或是将其转换为逻辑数据模型，以及进行其他模型相关操作。

在对物理数据模型进行修改后，可以将变更部分部署在数据库上。在工作台中使用比较功能，将显示出开发环境和生产环境中数据模型的不同；然后，可以分析模型变更对已有数据模型的影响；最后，通过工作台合并变更部分并生成增量 DDL 脚本文件，该脚本文件可以将模型的变更部分部署在数据库上。在执行增量 DDL 脚本的过程中，已有数据库数据将不会丢失。

注意：

反向设计可以分别从 DDL 脚本或者已有数据库开始。如果是从数据库开始，由于数据库的目录表中保存了大量元数据信息，这使得生成物理数据模型时有更多配置选项，而 IBM InfoSphere Data Architect 则可以很好地利用元数据信息来更好地生成物理数据模型。

7.2 使用工作台进行反向设计

当对数据库进行反向设计时，工作台会读取数据库对象的细节信息，并将其转为对应物理数据模型中的对象。比如，源数据库中如果包含一个模式，该模式包含有一个数据库表和一个索引，那么在通过反向设计生成的物理数据模型中，将同样地包含一个模式、一个表和一个索引。

为了理解如何对 DDL 脚本进行反向设计，我们通过以下例子来说明：使用 DDL 脚本创建了一个模式，其中包含一个表和一个索引。当对该 DDL 进行反向设计时，工作空间会在物理数据模型中创建相应的模式、表和索引。详细内容将在下一节进行介绍。

7.2.1 从 DDL 脚本进行反向设计

7.2.1.1 创建样本 DDL 脚本

首先，我们创建一个样本 DDL 脚本实例，而该实例将被用来进行反向设计。这个 DDL 脚本将在 SAMPLE 数据库连接上创建以下数据库对象：

- 数据库 SAMPLEUNIV。
- 在数据库 SAMPLEUNIV 中创建模式 UNIVERSITY。
- 在模式 UNIVERSITY 中创建两个表：DEPARTMENT 和 STUDENT。

创建样本 DDL 脚本实例步骤：

1. 在数据设计项目中右键单击“SQL 脚本”文件夹，然后选择“新建”->“SQL 或 XQuery 脚本”。新建 SQL 或 XQuery 脚本的窗口将打开。
2. 完成新建 SQL 或 XQuery 脚本窗口：
 - a. 选择 *University Info System* 项目。
 - b. 输入名字 **SAMPLE_DDL.sql**。
 - c. 确保以下选项被选中：“SQL XQuery 编辑器（用于包含一条或多条 SQL 和 XQuery 语句的脚本）”。
 - d. 完成。

“选择连接概要文件窗口”将打开。

3. 选择 SAMPLE 数据库连接概要文件。由于不会将脚本部署到数据库中，所以在这里不需要过多关注所选的数据库连接。

工作台会创建一个空白的 SQL 文件并将其在 SQL 文件编辑视图中打开。

4. 输入以下 SQL 语句来创建 SAMPLEUNIV 数据库，该数据库中将包含模式 UNIVERSITY 以及两张表：表 DEPARTMENT 和表 STUDENT。

```
CREATE DATABASE SAMPLEUNIV;
CREATE SCHEMA "SAMPLEUNIV"."UNIVERSITY";
CREATE TABLE "UNIVERSITY"."STUDENT"
  (STUDENT_ID CHAR(10) NOT NULL,
  STUDENT_FIRSTNAME CHAR(20) NOT NULL,
  STUDENT_LASTNAME CHAR(20) NOT NULL,
  STUDENT_ADDRESS CHAR(100) NOT NULL,
  STUDENT_PHONE CHAR(10) NOT NULL,
  DEPARTMENT_ID INTEGER NOT NULL);
CREATE TABLE "UNIVERSITY"."DEPARTMENT"
  ( DEPARTMENT_ID INTEGER NOT NULL,
```

```
DEPARTMENT_NAME CHAR(20) NOT NULL,  
DEPARTMENT_ESTDATE DATE);  
ALTER TABLE "UNIVERSITY"."STUDENT" ADD CONSTRAINT  
"STUDENT_PK" PRIMARY KEY(STUDENT_ID);  
ALTER TABLE "UNIVERSITY"."DEPARTMENT" ADD CONSTRAINT "DEPARTMENT_PK" PRIMARY  
KEY(DEPARTMENT_ID);  
ALTER TABLE "UNIVERSITY"."STUDENT" ADD CONSTRAINT "STUDENT_DEPARTMENT_FK"  
FOREIGN KEY(DEPARTMENT_ID) REFERENCES  
"UNIVERSITY"."DEPARTMENT"(DEPARTMENT_ID);
```

5. 保存。

SQL 脚本已经创建好并保存在“SQL 脚本”文件夹中。

注意:

下一章中将介绍如何用新建的脚本来进行反向设计，此处的脚本不需要在数据库上执行。

7.2.1.2 从 DDL 脚本进行反向设计

通过新建的 DDL 脚本进行反向设计，进而生成物理数据模型。

步骤:

1. 通过菜单“文件”->“新建”->“物理数据模型”，打开新建物理数据模型向导。
2. 填写模型文件向导页面：
 - a. 选择数据设计项目 *University Info System* 作为目标文件夹。
 - b. 输入“文件名”**NEW_SDT_DDL**。
 - c. 选择数据库类型“DB2 Linux 版、DB2 UNIX 版和 Windows 版”。
 - d. 选择数据库版本 V9.7。
 - e. 选择“从反向设计创建”。
 - f. 点击“下一步”。
3. 进入源页面，选择“DDL 脚本”，然后点击“下一步”。
4. 进入脚本文件页面，点击“浏览”并选择文件 **SAMPLE_DDL.sql**，然后点击“下一步”。
5. 进入选项页面，本页面不需要配置，点击“下一步”。
6. 进入反向设计状态页面，确保反向设计已成功执行，点击“完成”创建物理数据模型。

数据项目资源管理器中将显示新建的物理数据模型 **NEW_SDT_DDL.dbm**，如 [图 7.2](#) 所示。

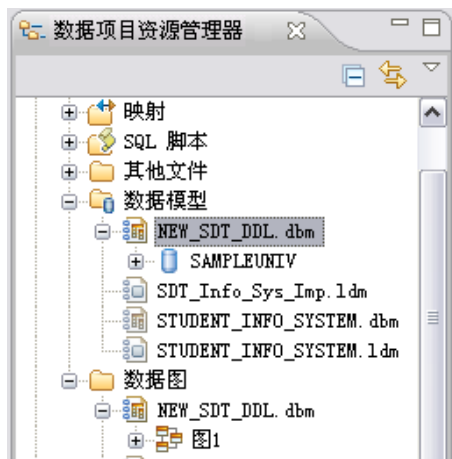


图 7.2 – 从 DDL 脚本创建的新物理数据模型

如果打开新建的物理数据模型，将看到在 DDL 脚本中所声明的模式和表。

7.2.2 从数据库进行反向设计

从数据库进行反向设计的步骤和从 DDL 脚本进行反向设计的步骤非常相似，只是前者额外提供了一个数据库对象选择功能。例如，数据库中包含多个模式，用户可以仅选择部分模式进行反向设计，在生成的物理数据模型中，修改这些模式并将变更部署到数据库中。

尽管在物理数据模型 `STUDENT_INFO_SYSTEM.dbm` 中，已包含了模式 `STUDENT_LIFE`，但是我们仍然选择对已部署到数据库中的模式进行反向设计，并对生成模型进行修改，然后进行比较同步，在生成 DDL 脚本后再将变更部分部署到 `SAMPLE` 数据库中。

对数据库进行反向设计的步骤：

1. 通过菜单“文件”->“新建”->“物理数据模型”，打开新建物理数据模型向导。
2. 填写模型文件向导页面：
 - a. 选择数据设计项目 *University Info System* 作为目标文件夹。
 - b. 输入“文件名”`NEW_SDT_INFO_SYS`。
 - c. 选择数据库类型“DB2 Linux 版、DB2 UNIX 版和 Windows 版”。
 - d. 选择数据库版本 V9.7。
 - e. 选择“从反向设计创建”，点击“下一步”。
3. 进入源页面，选择“数据库”，点击“下一步”。
4. 进入选择连接页面，选择 `SAMPLE` 数据库连接概要文件，点击“下一步”。
5. 进入选择对象页面，选择模式 `STUDENT_LIFE`，点击“下一步”。
6. 进入数据库元素页面，保持默认选项，点击“下一步”。
7. 进入选项页面，本页面不需要配置，点击“完成”。

新建的物理数据模型成功地在数据项目资源管理器中创建出来（如图 7.3），并在物理数据模型编辑器中打开。

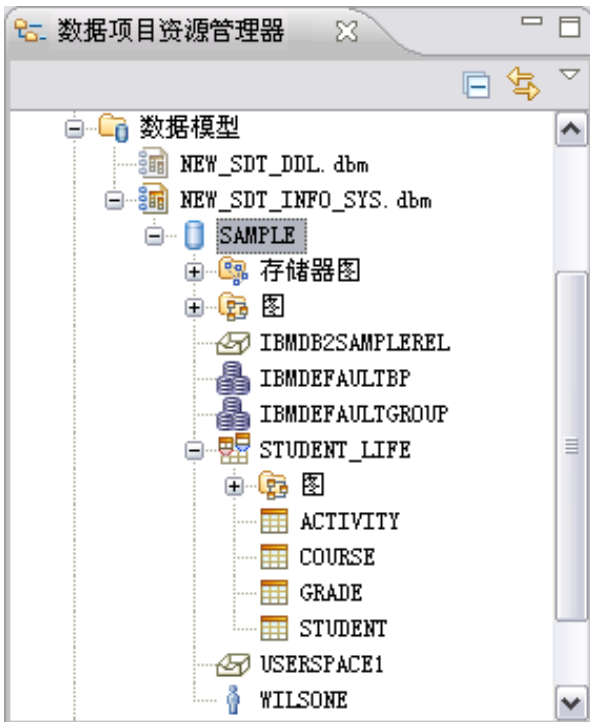


图 7.3 – 从已有的数据库创建的物理数据模型

7.3 修改新建的物理数据模型

通过修改新建的物理数据模型 `NEW_SDT_INFO_SYS.dbm`，创建一些新的列和索引。例如，业务逻辑需要一些列来保存每一门课的教师的电话号码以及学生的电话号码，同时需要添加索引，用来对学生姓名进行排序，使得数据检索更加快速。

做完修改后，您可以使用工作台的比较、合并功能生成 DDL 脚本将所作的修改部署到 `SAMPLE` 数据库中。

修改物理数据模型的步骤：

1. 在数据项目资源管理器中打开表 `COURSE: University Info System` -> “数据模型” -> `NEW_SDT_INFO_SYS.dbm` -> `SAMPLE` -> `STUDENT_LIFE` -> `COURSE`。
2. 在表 `COURSE` 中新建列 `INSTRUCTOR_PHONE`:
 - a. 在表 `COURSE` 的右键菜单中选择“添加数据对象” -> “列”。表 `COURSE` 中创建了一个新的列。
 - b. 将新列命名为 `INSTRUCTOR_PHONE`。
 - c. 设定列 `INSTRUCTOR_PHONE` 的“数据类型”为 `CHAR`。
 - d. 设定列 `INSTRUCTOR_PHONE` 的“长度”为 10。

3. 重复步骤 2，在表 *STUDENT* 中创建列 *STUDENT_PHONE*，其数据类型与步骤 2 中相同。
4. 创建索引，使学生信息根据学生的姓氏排序：
 - a. 在 *STUDENT* 表的右键菜单上，选择“添加数据对象”->“索引”。表 *STUDENT* 中创建了一个新的索引。
 - b. 命名新的索引为 **STUDENT_LAST_NAME_IDX**。
 - c. 打开属性视图的“列”选项卡，点击“键列”右面的 (...)按钮，打开选择列窗口，修改键中包含的列。
 - d. 取消选中列 *FIRST_NAME*，然后选中列 *LAST_NAME*，点击“确定”。
5. 保存。

物理数据模型中已经创建并保存了新的列和索引。现在，可以通过比较该物理数据模型与目标数据库，生成增量 DDL 脚本，并将变更同步到 *SAMPLE* 数据库中。

7.4 通过比较和合并来同步变更

比较 功能可以帮助您发现不同版本的模型之间的差异，*合并* 功能则可以将变更同步到已有模型上，同时，还可以帮助分析这些变更对已有模型的影响，这有助于发现数据对象之间的依赖，进而更好的理解已做的变更是如何影响相关数据对象的。

注意:

由于在这个模型上所做的变更非常小，模型上没有其他依赖对象受到影响。但是，在每次生成 DDL 脚本并部署到数据库之前，最好分析所做变更的影响。

如果需要分析变更的影响，首先要对模型进行比较，将需要变更部分从一个模型复制到另一个模型；然后使用 *分析左边影响* 或 *分析右边影响* 对变更进行分析。被影响的对象将在受影响对象视图中显示，可以通过向下钻取数据来查看受影响的数据对象的详细信息。

为了更好地学习在工作台中使用比较和合并功能，我们对物理数据模型 *NEW_SDT_INFO_SYS.dbm* 和 *SAMPLE* 数据库进行比较。由于已经对物理数据模型进行修改，所以在比较过程中，能够产生差异，并可以针对差异生成 DDL 脚本，然后部署到数据库中。

注意:

您也可以对同一个工作空间中的两个模型进行比较，其比较步骤与模型-源数据库比较步骤相似，在本书中将不做介绍。

7.4.1 对数据库进行比较和合并变更

可以将数据模型和源数据库进行比较。比较功能适用于以下对象：

- 源数据库
- 逻辑数据模型
- 物理数据模型

- 词汇表模型
- 域模型

现在我们比较一下新的物理数据模型和源数据库的区别，步骤如下：

1. 找到数据项目资源管理器中的模式 *STUDENT_LIFE: University Info System* -> “数据模型” -> *NEW_SDT_INFO_SYS.dbm* -> *SAMPLE* -> *STUDENT_LIFE*。
2. 右键单击模式 *STUDENT_LIFE*，选择“比较对象” -> “其他数据对象”，将打开与其他数据对象比较向导。
3. 展开 *SAMPLE* 数据库，选择源数据库的模式 *STUDENT_LIFE*，如图 7.4 所示：

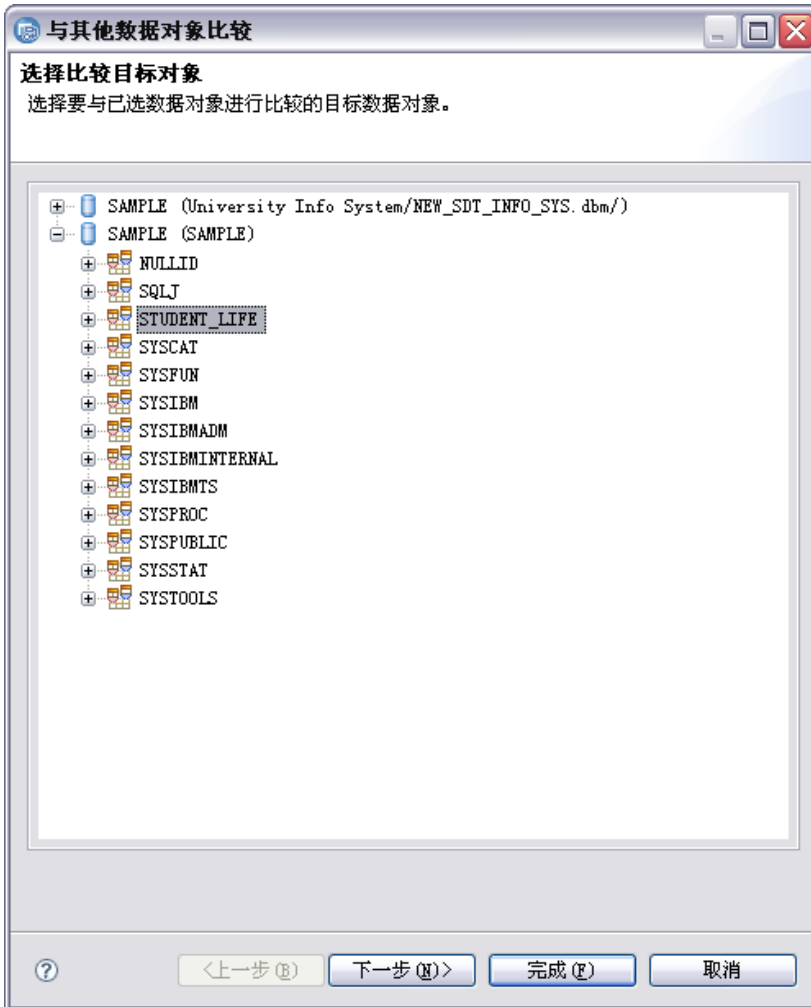


图 7.4 – 选择源数据库

点击“下一步”，进入过滤条件页面。

4. 指定需要比较的数据对象：
 - a. 点击“全部取消选择”按钮，取消选中所有数据对象。

- b. 在“要包含在比较中的对象”区域中，选择“列”，所有与列相关的数据对象将自动选中。
 - c. 选择“索引”。
5. 完成。

在比较编辑器中，将显示物理数据模型与源数据库 **SAMPLE** 中模式 **STUDENT_LIFE** 的比较结果。

在结构比较窗口中，可以通过导航显示两个模式的区别。其中左面是数据设计项目中的模式，右面是源数据库中的模式。如图 7.5 所示。



图 7.5 – 比较模式

6. 使用结构比较窗口中的“下一个差别”或“上一个差别”，来查看两个模式之间的区别。
7. 在结构比较窗口找到列 **COURSE.INSTRUCTOR_PHONE**，如图 7.6 所示。注意到在源数据库中并没有该列。

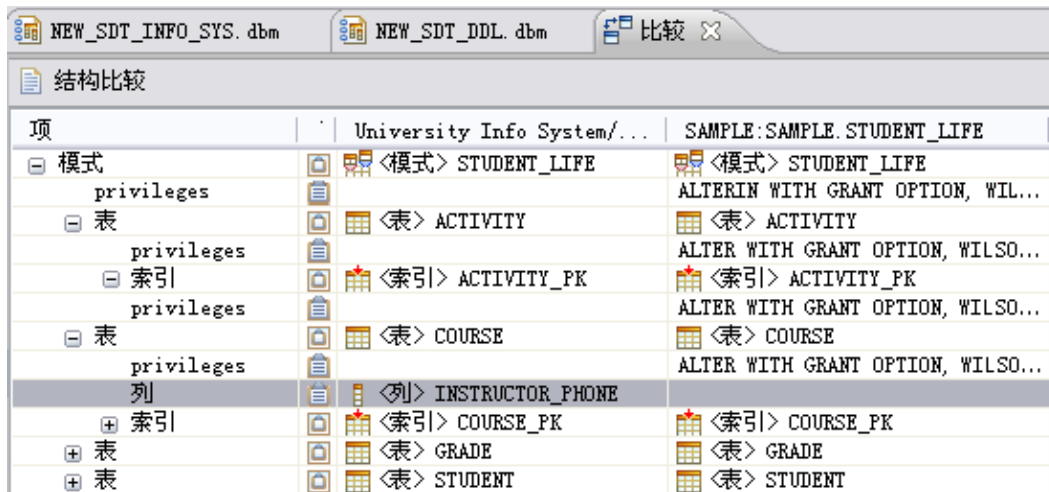





图 7.6 – 定位到新的列

在属性比较窗口中，会显示数据对象之间的区别。左面部分显示数据模型中的数据对象信息，右面显示源数据库的数据对象信息。若无数据对象，则不显示。

8. 右键单击列 **INSTRUCTOR_PHONE**，选择“分析左边影响”。由于没有其他的数据对象引用或依赖该列，在受影响的对象视图中没有受影响的对象。

9. 使用属性比较窗口，确定想要复制的新列。点击按钮“从左边复制到右边” ，将新列复制到源数据库。
10. 在结构比较窗口中，找到列 `STUDENT_PHONE`。
11. 右键单击列 `STUDENT_PHONE`，选择“分析左边影响”。由于没有其他的数据对象引用或依赖该列，在受影响的对象视图中没有受影响的对象。
12. 使用结构比较窗口，确定想要复制的新列 `STUDENT_PHONE`，点击按钮“从左边复制到右边” ，将新列复制到源数据库。
13. 找到新索引 `STUDENT_LAST_NAME_IDX`，分析影响后，复制到源数据库中。
14. 生成 DDL 脚本并部署到数据库上。通过按钮“生成右边 Delta DDL” ，打开生成 DDL 的向导窗口。
15. 完成保存并运行 DDL 页面：
 - a. 确定选择 *University Info System* 数据设计项目作为目标文件夹。
 - b. 输入“文件名” `DELTA_SDT_CHANGES.sql`。
 - c. 选中“在服务器上运行 DDL”选项，将变更部署到服务器上。
 - d. 点击“下一步”。
16. 在选择连接页面，选中 `SAMPLE` 数据库连接，点击“下一步”。
17. 在摘要页面，检查摘要信息，然后点击“完成”。

在服务器端运行 DDL 脚本后，`SAMPLE` 数据库中创建了新的列和索引。检查 SQL 结果视图，确保 SQL 语句成功执行。刷新数据源资源管理器，检查新的列和索引是否存在。如图 7.7 显示新增加的列。

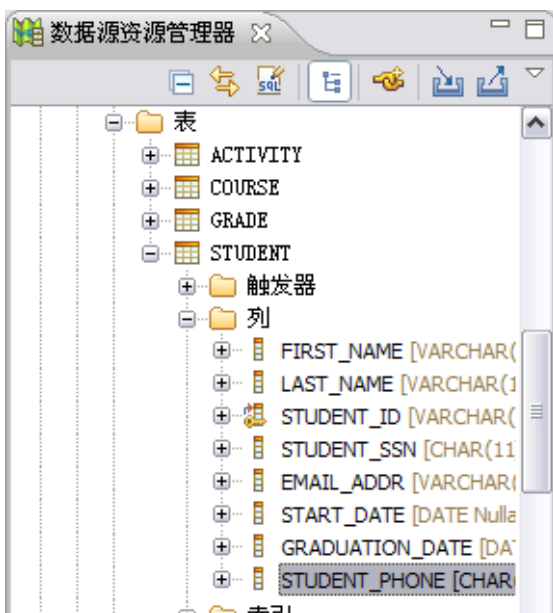


图 7.7 – 在数据源资源管理器中查看变更

7.4.2 比较和合并功能的优点

如果没有 IBM InfoSphere Data Architect，比较和变更模型是一个非常耗时的工作，因为您需要手动设计所有的查询语句，而且手动变更数据库很容易引起很多错误，尤其是数据库包含大量模式、表和约束的时候。

IBM InfoSphere Data Architect 能够在比较和合并的时候，找出已有的约束和依赖，这使得对数据库和模式的修改变得简单。

7.5 练习

1. 创建如下脚本，并命名为 **EXERCISE_7_5.sql**：

```
CREATE SCHEMA "EXERCISE";
CREATE TABLE "EXERCISE"."TABLE1"
(
    "TABLE1_ID" INTEGER,
    "TABLE1_NAME" VARCHAR(20),
    "TABLE2_ID" INTEGER);
CREATE TABLE "EXERCISE"."TABLE2"
(
    "TABLE2_ID" INTEGER,
    "TABLE2_NAME" VARCHAR(20),
    "TABLE3_ID" INTEGER);
CREATE TABLE "EXERCISE"."TABLE3" (
    "TABLE3_ID" INTEGER,
    "TABLE3_NAME" VARCHAR(20));
CREATE UNIQUE INDEX "TABLE1_ID_INDEX" ON "EXERCISE"."TABLE1"
("TABLE1_ID");
CREATE UNIQUE INDEX "TABLE2_ID_INDEX" ON "EXERCISE"."TABLE2"
("TABLE2_ID");
CREATE UNIQUE INDEX "TABLE3_ID_INDEX" ON "EXERCISE"."TABLE3"
("TABLE3_ID");
```

2. 在 DDL 脚本上执行反向设计，生成物理数据模型 **EXERCISE_7_5.dbm**。在新建物理数据模型的向导页面，确保生成“概述图”和“推理隐式关系”。记录结果。
3. 在图 EXERCISE 的属性视图中，显示数据对象的键和非键。在“布局”标签中将图的布局改为“分层”->“自顶向下”。新的物理数据模型概述图将显示如图 7.8。

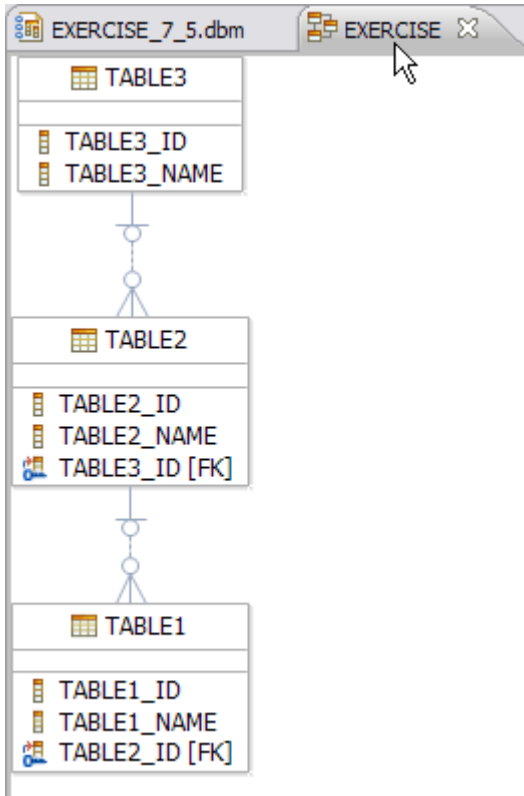


图 7.8 – 查看 EXERCISE 图

7.6 总结

通过本章，我们学习了如何通过使用反向设计，将已有的数据库或 DDL 脚本中的表集、模式集生成为物理数据模型中的数据对象；学习了在使用反向设计时，如何将想要的对象类型加入到即将生成的模型中；还学习了如何使用比较和合并功能，在不丢失数据的情况下，修改已有数据库。

7.7 本章回顾

1. 正误判断：在对 DDL 脚本进行反向设计时，只能使用后缀为.DDL 的文件。
2. 正误判断：在对 DDL 脚本进行反向设计时，工作台允许选择数据对象类型。
3. 在使用反向设计生成物理数据模型时，什么是不同的数据对象类型？
4. 正误判断：只能在物理数据模型以及其源数据库上做比较和合并。
5. 可以比较的数据模型有哪几种？

8

第八章 – 模型映射和发现

信息可以来自不同的数据源，对任何企业来说如何集成这些数据是一个共有需求。当已有的数据源需要与收购的数据源进行集成的时候，理解元数据非常重要。

元数据是用来描述数据的数据，它用来存储数据的数据类型、意义和其它标记信息。理解元数据可以帮助我们更好的理解不同的数据源之间的关联，并据此在数据源之间创建映射和关系。映射模型可以帮助您将元数据映射到不同的数据源。

IBM InfoSphere Data Architect 提供了如下元数据集成工具：

- 在数据源之间定义关系和映射
- 定义转换规则
- 发现关系
- 生成可以用来部署到数据库的脚本

8.1 型模射映：概述

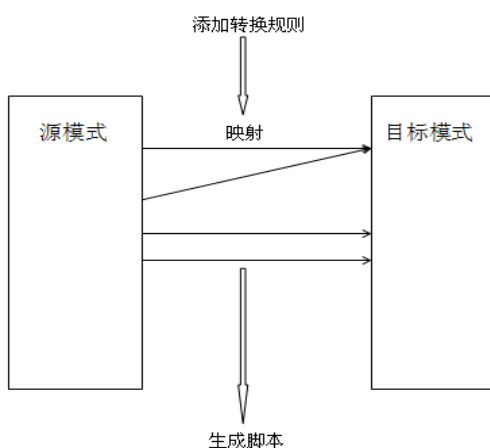


图 8.1 – 映射模型概述

一个**映射模型**定义两个不同的模式之间的关系：一个源模式和一个目标模式。映射模型存储为一个映射规范语言（MSL）文件。源模型可以是逻辑数据模型或者物理数据模型。目标模型可以是一个逻辑数据模型、物理数据模型或者一个 XML 模式文件（XSD）。

映射模型是信息集成建模的一部分。*信息集成建模*提供了一组功能可以帮助您管理元数据。为了让新的数据源与已有的企业级数据进行集成，您需要理解新的数据源中的数据和已有环境中的数据之间的关系。然后您可以构建一个流程用来将新数据源中的数据复制到已有的数据源中。通过定义映射和关系，您可以管理不同元数据的集合。

8.1.1 使用映射模型管理元数据

IBM InfoSphere Data Architect 可以在如下方面帮助管理模型的元数据：

- **发现关系：**工作台可以帮助您发现源数据中一个或者多个模式跟目标数据中的一个模式之间的可能的关联。关联的关系可以基于元数据或者任何可用的样本数据内容。
- **映射关系：**IBM InfoSphere Data Architect 可以通过自动发现或者手动指定的方式将您的源和目标数据模型进行映射并且进行可视化展示。对于模型之间创建的映射，它描述了如何通过合并和变换将一个或者多个源数据模型以一定的格式对应目标数据模型。映射结果将被保存为映射模型。
- **构建表达式：**您可以为映射模型定义诸如函数、连接条件、过滤条件和排序条件等表达式，在部署脚本中可以使用这些表达式。
- **生成脚本：**您可以从映射模型中生成脚本。脚本可以被用来从映射模型兼容的源中转换和过滤数据到映射模型兼容的目标中。

映射模型通过定义关系可以帮助我们从一个或多个源中集成不同的模型。转换规则可以被添加到不同数据模式之间定义的关系和映射中。

*映射*是还未在数据源中实现的两个数据结构之间的依赖。映射模型用于存储这些映射。

试想这个学生信息系统被设计用来处理多个校区的需求。例如，一些大学有一个主校区，同时它在同一地理区域又有一些小的卫星校区。假设物理数据模型 `NEW_SDT_INFO_SYS.dbm` 是对一所大学的一个小的卫星校区的一些数据源进行建模，但是你想无缝的集成这些信息。

因为您在第 6 章中通过反向设计得到了这个物理数据模型，您可以创建一个映射模型来对它的元数据和物理数据模型 `STUDENT_INFO_SYSTEM.dbm` 中的相关对象进行映射。

8.1.2 使用映射模型进一步管理命名标准

您可以使用一个词汇表模型或者命名模型来发现源和目标数据模式之间的关系。您在本书的第 3 章已经学过词汇表模型。而您之前在数据设计流程中定义的命名规则可以被用来识别不同数据模式之间的关系。

注意：

本节中的步骤仅供参考，你可以不用执行这些步骤。

您已经创建了一个词汇表模型用来跟踪数据模型中的实体、表和其它数据对象的命名。您可以使用词汇表模型来确保所有的数据模型符合您为您的组织内部定义的命名标准。如果映射模型的自动发现不能正确地确定哪些对象应该被映射，那么也许这些对象不符合标准。

为了确定模型是否符合映射模型所需的命名标准：

1. 打开企业命名标准窗口，在映射模型中添加一个命名模型：
 - a. 通过一个已有的映射模型打开映射编辑器。
 - b. 右键单击映射编辑器
 - c. 选择“发现关系”->“企业命名标准”打开企业命名标准窗口。
2. 选择您想要应用命名模型的数据模式。
3. 点击“命名模型列表”窗格旁边的“添加”按钮打开选择命名模型文件窗口。
4. 选择一个词汇表模型（文件扩展名为.ndm），然后点击“确定”按钮。
5. 对于任何其他的数据模式重复如上步骤，然后点击“完成”按钮。
6. 基于最佳匹配发现关系：
 - a. 右键单击映射编辑器，选择“发现关系”->“查找最佳匹配”打开发现关系窗口。
 - b. 选择你想要进行关系发现的数据模式，然后点击“完成”。

通过对模型进行分析，工作台可以确定哪些数据对象在映射模型中相关联。

7. 拒绝或者接受映射。
8. 如果必要的话创建新的映射。
9. 保存您的工作结果。

8.2 在工作台构建映射

在数据设计项目中，您可以把逻辑数据模型或者物理数据模型作为源模型，逻辑数据模型、物理数据模型或者 XML 模式作为目标模型创建映射模型。

限制：

如果源是一个逻辑数据模型，您将不能生成脚本。当您在逻辑数据模型和其他源或目标之间创建映射的时候，映射只能用来进行发布，也就是说您只能据此发布映射报告。

8.2.1 创建一个空白的映射模型

让我们创建一个新的映射模型来对物理数据模型 `NEW_SDT_INFO_SYS.dbm` 和 `STUDENT_INFO_SYSTEM.dbm` 进行映射。

1. 点击“文件”->“新建”->“映射模型”打开映射编辑器。
2. 完成创建映射页面：
 - a. 确保“目标文件夹”位置被设置为数据设计项目：*University Info System*。
 - b. 在“文件名”位置指定：*STUDENT_MAPPING.msl*。
 - c. 为了可以创建 SQL 脚本来部署您的映射，请确保选中“为该模型启用脚本生成”选项。
 - d. 点击“下一步”。
3. 完成指定映射源页面：

- a. 点击“添加”按钮打开选择映射源窗口。
 - b. 选择物理数据模型文件 *STUDENT_INFO_SYSTEM.dbm*，点击“确定”，将映射文件添加至“映射源文件”列表中。
 - c. 点击“下一步”。
4. 在指定映射源页面不要更改任何选项，点击“下一步”。
 5. 完成指定映射目标页面：
 - a. 点击“添加”按钮打开选择映射目标窗口。
 - b. 选择物理数据模型文件 *STUDENT_INFO_SYSTEM.dbm* 并点击“确定”，把映射目标文件添加到“映射目标文件”列表中。
 - c. 确保选择“映射目标模式”列表中的所有对象。
 - d. 点击“下一步”。
 6. 在新建模型摘要页面中检查您的选项，然后点击“完成”。

在你的数据设计项目中的“映射”目录下，您可以找到映射模型文件，同时它在编辑器里已经被打开。

如图 8.2 所示，映射编辑器由三个窗格组成：

- “源”：包含源文件和模式。
- “映射”：此窗格将显示映射模型中的映射和映射组。
- “目标”：包含目标文件和模式。

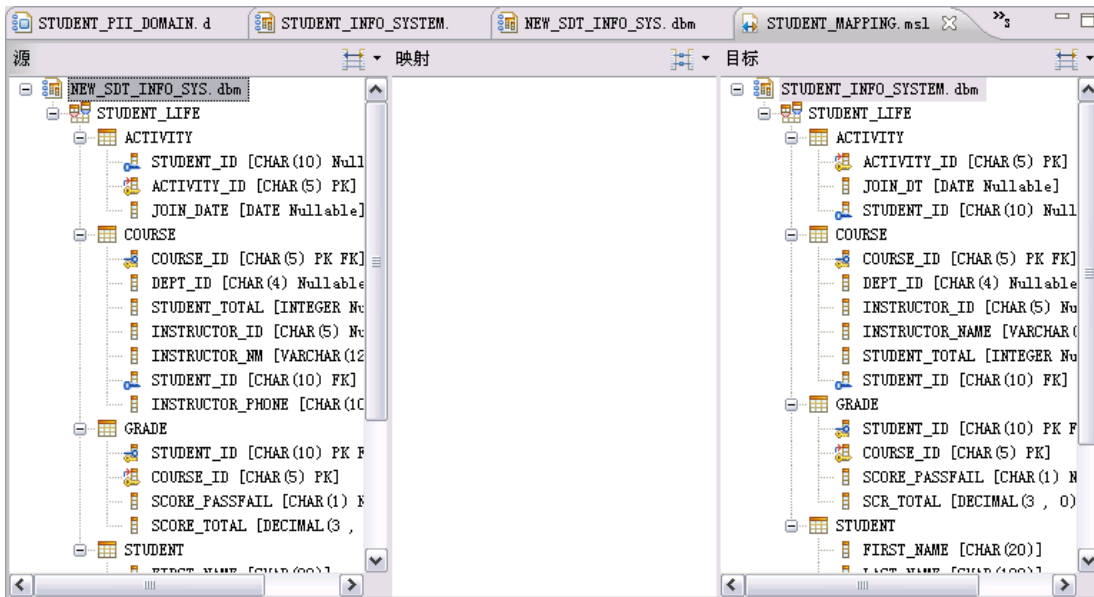


图 8.2 – 映射模型编辑器

8.2.2 添加映射到映射模型

您可以在映射编辑器中创建源和目标模式间的映射。创建映射有如下两种方式：

- **手动添加映射：** 在“源”和“目标”窗格中选择对象并创建映射。
- **工作台自动发现映射：** 运行发现关系向导来确定数据对象间的最佳映射。

8.2.2.1 手动创建映射

当您手动创建一个映射的时候，您需要确保每一个源对象都被映射到目标数据模型中的相应的对象。手动创建映射的过程通常是很耗时的，但对于小的数据模型，有时候手动地创建映射反倒是不错的方法。

接下来我们通过手动创建两个映射来了解相关步骤：

1. 对STUDENT.STUDENT_ID 列进行映射：
 - a. 在“源” 击点键左中格窗列 STUDENT.STUDENT_ID。
 - b. 在“目标” 窗格中左键点击列 STUDENT.STUDENT_ID，如图 8.3 所示。

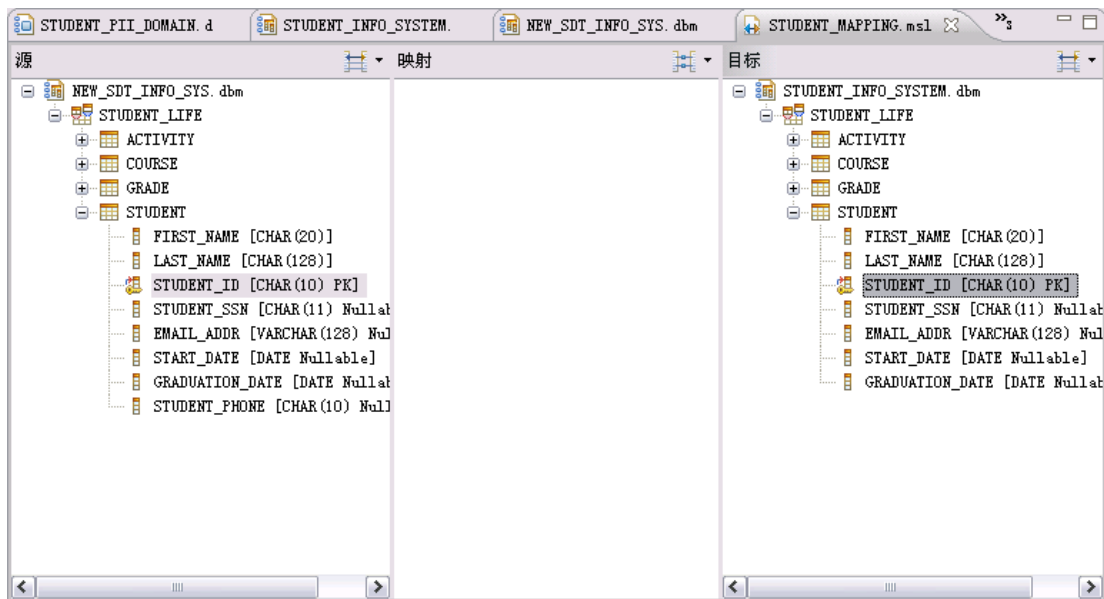


图 8.3 – 选择源和目标对象

- c. 在映射编辑器中右击任意位置并选择“创建映射”。

在映射编辑器中创建的映射如图 8.4 所示。

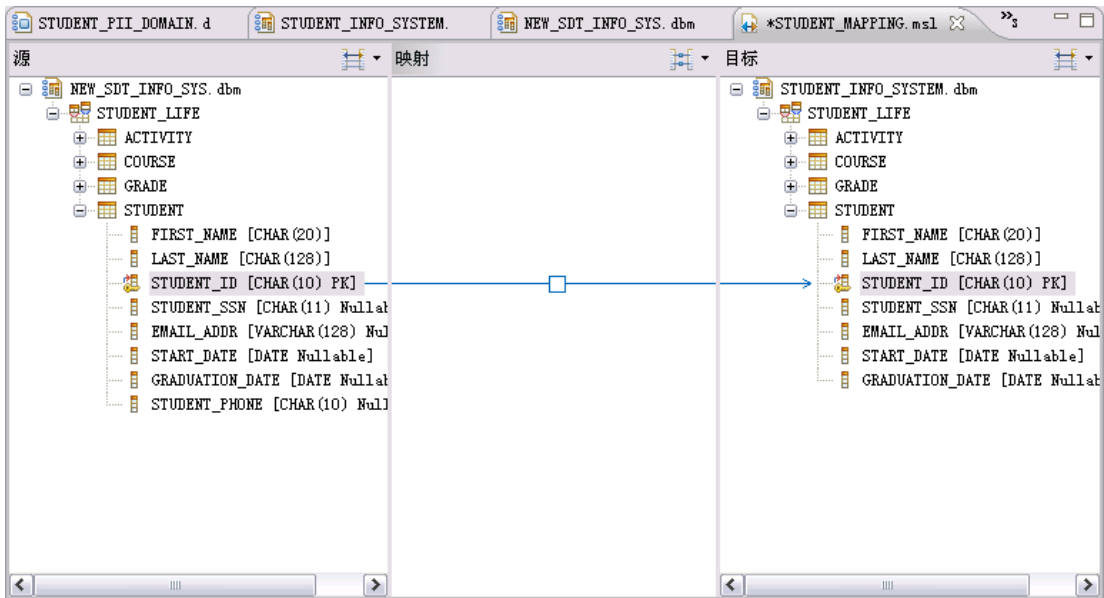


图 8.4 – 映射编辑器中新创建的映射

2. 列对步骤上如复重STUDENT.FIRST_NAME 进行映射，映射编辑器如图 8.5 所示：

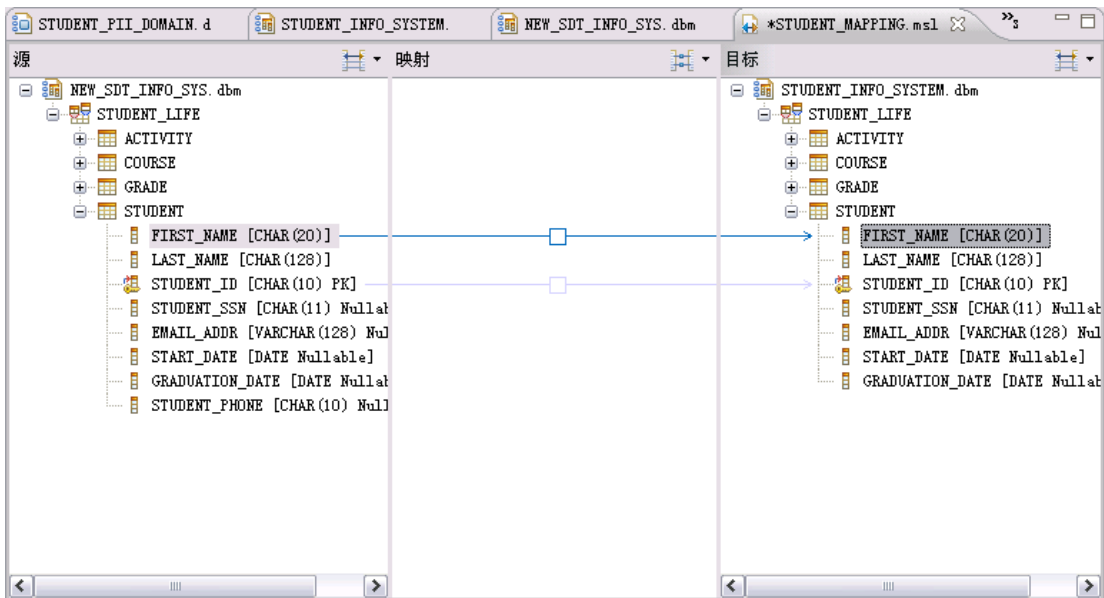


图 8.5 – 创建另外新的映射

3. 保存您的工作。

8.2.2.2 系关现发动自

当您有一个很大的数据模型时，在不同的数据源之间识别关系往往是很乏味和耗时的。此外，有些关系并不容易识别。您可以使用工作台中的发现关系向导基于特定的算法帮您识别关系。

注意:

您可以通过首选项窗口定制关系发现方式。点击“窗口”->“首选项”打开首选项窗口。展开“数据管理”节点选择“映射编辑器”->“发现关系”来定义如何进行关系发现。

使用已有的映射模型来发现两个数据模型中的对象关系:

1. 在映射模型中右击选择“发现关系”->“查找最佳匹配”打开发现关系窗口。
2. 在“源”窗格中，选择 `NEW_SDT_INFO_SYS.dbm` 数据模型旁边的复选框，确保物理数据模型下的所有对象被选中，如 [图 8.6](#) 所示。

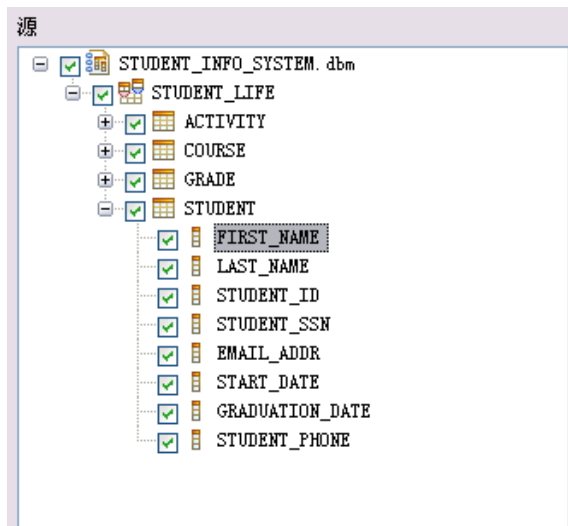


图 8.6 – 在源窗格中选择所有对象

3. 在“目标”窗格中选择 `STUDENT_INFO_SYSTEM.dbm` 数据模型旁边的复选框，确保物理数据模型下的所有对象被选中。
4. 点击“完成”按钮。

发现进程开始运行，您可以看到在两个数据模型的对象之间出现一系列的映射线，而这些映射线标记了可能存在的关系。如 [图 8.7](#) 所示。

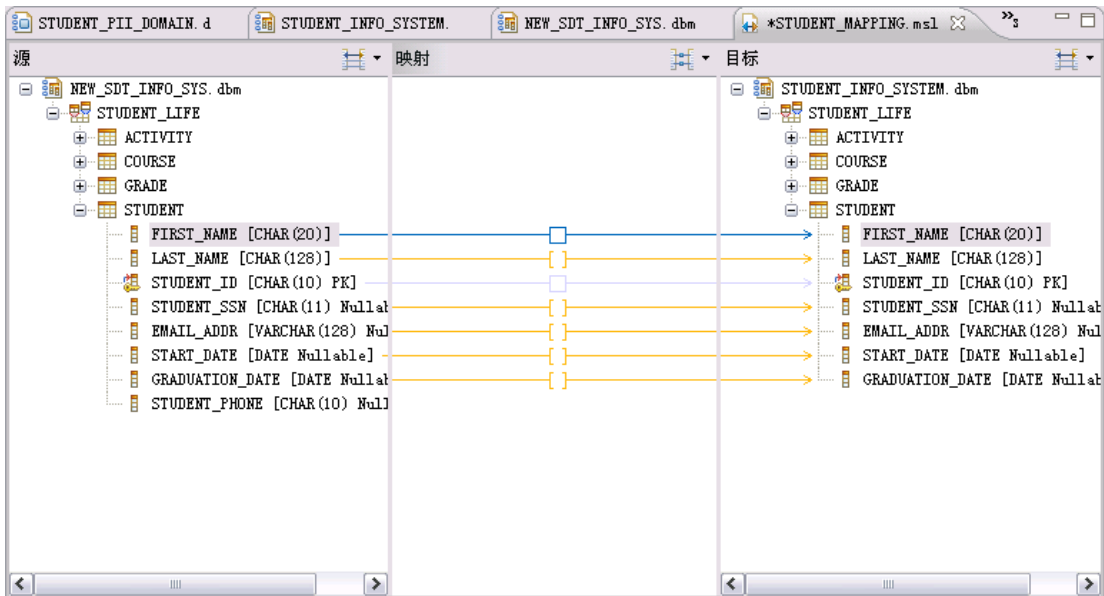


图 8.7 – 模型中对象间的映射线

8.2.2.3 接受和拒绝发现的关系

发现的关系只是潜在的关系，在这些关系变成映射模型的一部分之前，你必须接受或者拒绝这些映射。

映射模型中的关系是用颜色标记的。颜色可以在首选项窗口中进行设置，具体设置可以在“数据管理”节点下的映射编辑器页面中找到。

需要注意的是您创建的关系是蓝色还是淡蓝色取决于该关系是否被选中。在工作台内，这些关系是可以被接受的映射，是映射模型中被接受的部分。被发现的映射在没有被选中的情况下是金色的，而一旦您选中它，它就会变成棕色。

您选中映射并确定接受或者拒绝它们，而一旦您拒绝一个映射，该映射线就会在该映射模型中消失。

首先，我们手动的接受一些映射关系。然后您可以检查剩余的的关系并决定是否接受这些映射。

为了接受或者拒绝已经发现的映射，您可以执行如下步骤：

1. 选中 `STUDENT.LAST_NAME` 的映射，这时映射线的颜色变成了棕色，而发现的映射显示在属性视图中。
2. 右击映射并选择“接受映射”。如图 8.8 所示。

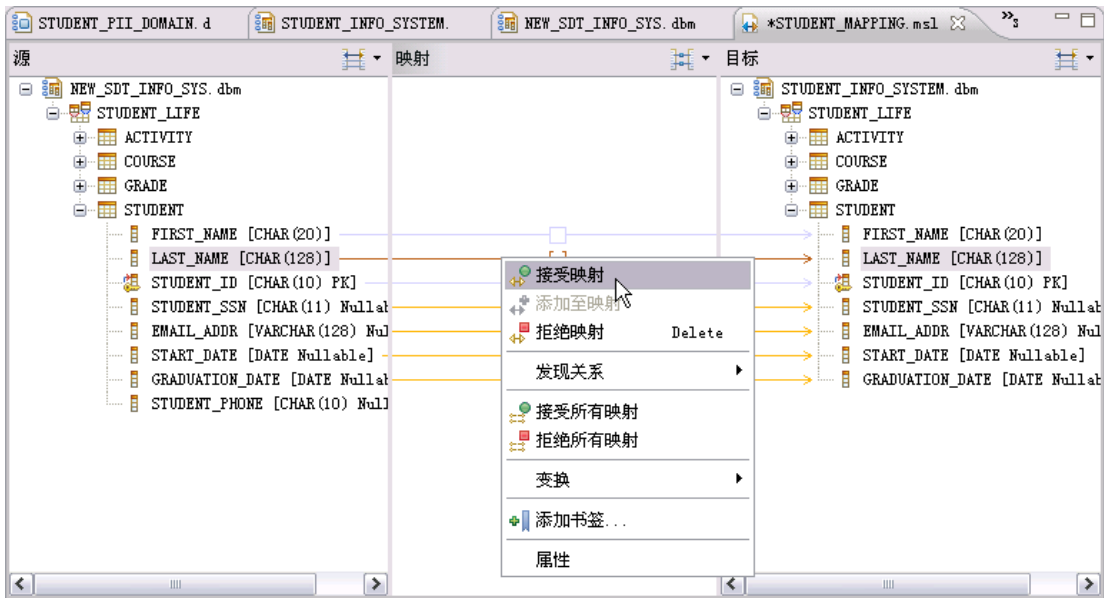


图 8.8 – 手动接受映射关系

映射线颜色变成了蓝色。

3. 重复步骤 1 和步骤 2，接受 *STUDENT.STUDENT_SSN* 和 *STUDENT.EMAIL_ADDR* 的映射。
4. 保存您的工作。

现在，您可能已经意识到如果目标模型是比较大的数据模型，这个过程将非常乏味。既然您已经熟悉如何手动地接受或者拒绝映射，你可以查看剩余的映射并决定是否接受所有建议的映射。

5. 展开“源”和“目标”窗格中的所有表节点。编辑器如图 8.9 所示。

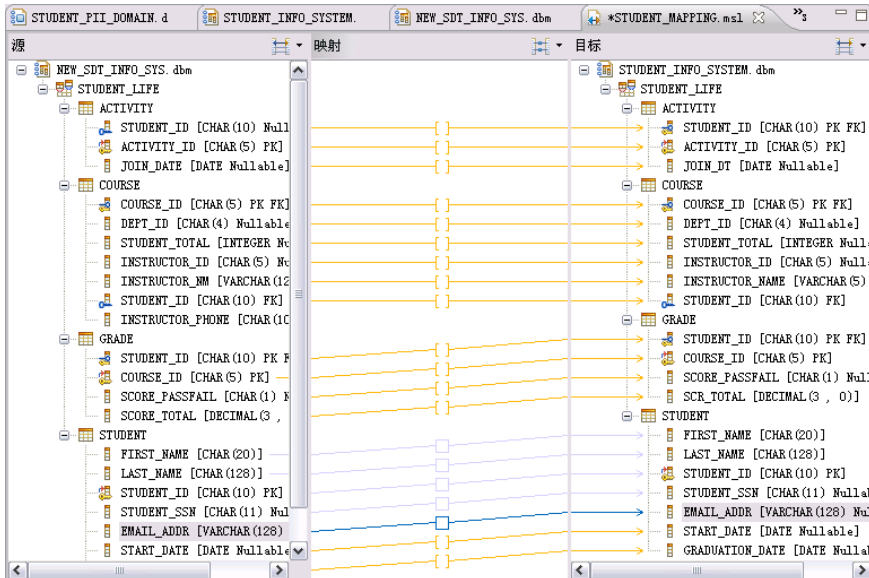


图 8.9 – 查看所有建议的映射

6. 查看映射并验证源和目标是否应该被映射到一起。在这种情况下，您应该跟卫星校区的项目组成员进行沟通以决定每个列的意义，然后确定哪些列应该被映射。而因为物理数据模型 `NEW_SDT_INFO_SYS.dbm` 是您通过反向设计得到的，您应该知道那些被发现的关系映射是否正确。
7. 在映射编辑器中右击并选择“接受所有映射”。

映射线的颜色变为浅蓝色，这些映射为映射模型中的有效部分。

8. 保存您的工作。

8.3 类型射映

根据映射的属性以及映射的创建方式，五种类型的映射可以被识别。映射在编辑器中通过在源节点和目标节点之间的线表示。每个映射线中间有一个集线器。

映射线的颜色取决于映射类型。被发现的映射线包含的集线器显示为括号。所有其他类型的映射线包含的集线器显示为正方形。

- **映射：**一个映射连接一个或者多个源列到单一的目标列，或者一个 XML 模式文档（XSD）中的单一元素或者属性。
- **发现的映射：**这些映射是发现关系向导建议的映射。你可以接受或者拒绝每个关系中的个别映射。

在您接受或者拒绝发现的映射线之前，映射线包含的集线器显示为括号。而一旦您接受一个发现的映射，集线器就会显示为正方形。

- **常量映射：**这些映射有一个目标对象，但是它们没有源对象。通过常量映射，你可以给一个目标对象分配一个变换功能，例如日期或者时间。在映射组详细信息视图中，您可以通过右键单击一个目标对象，然后在菜单中选择“创建常量映射”来创建常量映射。
- **映射组：**映射组为目标表（关系型数据库中）或者元素（XML 模式中）进行排序。当您为映射组生成脚本时，每个映射组生成一条语句。
- **无效映射：**当您改变源或者目标模型中的数据对象的属性时，映射将变为无效映射。当映射编辑器打开时，所有的映射都会重新验证。如果源或者目标对象无效时，对应的映射将无效，而映射线上将出现红色的“X”标记。

8.4 为映射模型添加表达式和过滤器

您可以为每一个映射创建变换表达式。表达式可能是在一个源列上定义的过滤器、连接、排序或者变换功能，进而得到想要的目标列值。表达式可能会通过修改源列的值或者数据类型以与目标列兼容。表达式是 SQL 片段，并且会出现在映射模型生成的 DDL 脚本中。

为了确保物理数据模型 `NEW_SDT_INFO_SYS.dbm` 中列 `STUDENT.FIRST_NAME` 的所有行的数据是大写字母，从而与物理数据模型 `STUDENT_INFO_SYSTEM.dbm` 中的信息统一，让我们来创建一个映射变换表达式。

创建一个变换表达式需要执行如下步骤：

1. 选择两个列 `STUDENT.FIRST_NAME` 之间的映射线，映射的属性出现在属性视图中。
2. 点击“变换”域附近的“表达式构建器”按钮打开表达式构建器窗口。

3. 选择一个“功能”。在我们的例子中，您将添加一个功能以确保数据库中存储的所有姓名中的名是大写，展开“字符串”节点，选择功能 `UCASE(VARCHAR)-VARCHAR`

该功能出现在“变换表达式”窗格中。

4. 把光标放在括号中间，如图 8.10 所示。

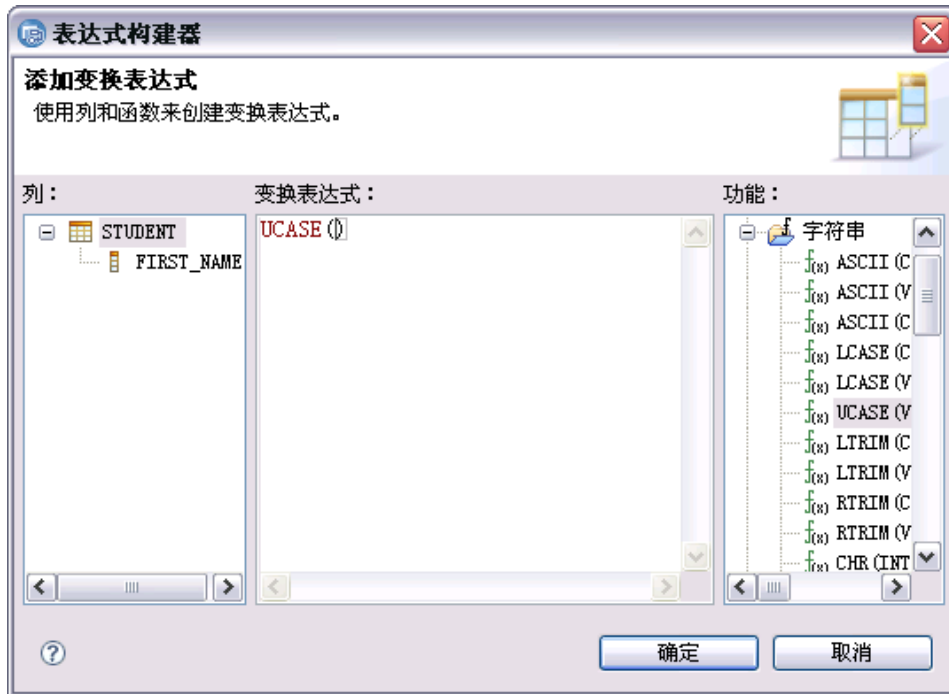


图 8.10 – 在表达式编辑器中创建表达式

5. 双击列 `STUDENT.FIRST_NAME` 把该列添加到表达式中，如图 8.11 所示。

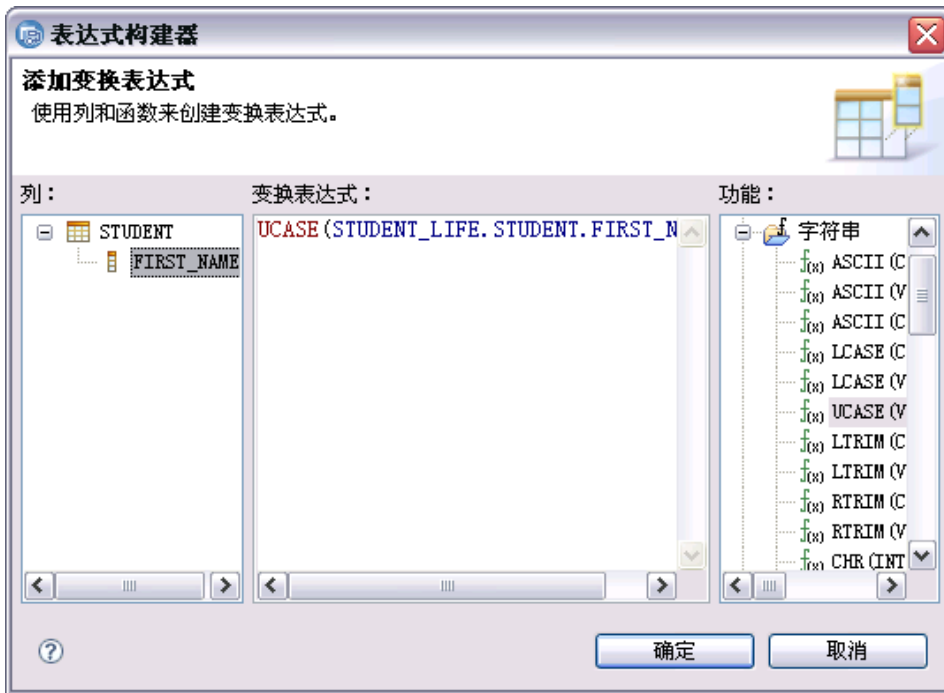


图 8.11 – 创建完整的表达式

6. 点击“确定”按钮把变换表达式添加到映射模型中。

8.5 生成可部署的脚本

您可以从映射模型生成一个 SQL 脚本，该脚本中包含映射模型中创建的所有映射相关的 SQL 脚本并可以用于数据库的部署。

接下来我们将对映射模型生成脚本并进行部署：

1. 在数据项目资源管理器中定位映射模型 **STUDENT_MAPPING.ms1: University Info System** -> “映射” -> **STUDENT_MAPPING.ms1**
2. 右键单击映射模型文件并选择“生成脚本”，打开生成脚本向导。
3. 完成生成脚本选项：
 - a. 确保 SQL 查询语言被选中。
 - b. 确保“创建 Select 语句”查询类型被选中。
 - c. 选择复选框“创建标准名称”。
 - d. 为名称文本脚定指 **STUDENT_MAPPING.sql**。
 - e. 点击“下一步”。
4. 在总结页面验证脚本，然后点击“完成”生成 SQL 脚本。

脚本生成并在编辑器视图中打开。当您在服务器上运行该脚本时，语句将会运行并生成物理数据模型中不同的模式之间的映射。

8.6 将映射模型导出为 CSV 格式

您可以把映射模型导出为一个逗号分割文件（使用.csv 为扩展名）。CSV 格式被许多工具支持（包括电子表格工具），因此您可以通过逗号分隔文件将映射模型导入到其他工具中。

当您将一个映射模型导出为 CSV 格式的文件时，映射模型中的内容将被作如下转换：

- 对于一对一的映射在 CSV 文件中存储为一行。
- 对于多对一的映射在 CSV 文件中存储为多行，每个源和目标对为一行。

导出您的映射文件到一个 CSV 文件中：

1. 在主菜单中，点击“文件”->“导出”打开导出向导。
2. 展开“数据”节点，选择选项“将映射模型导出至 CSV 格式的文件”，然后点击“下一步”。
3. 完成映射模型导出页：
 - a. 点击“要导出的模型”域旁的“浏览”按钮打开导出映射模型窗口。
 - b. 选择映射文件 **STUDENT_MAPPING.ms1** 然后单击“确定”按钮把它添加到向导中。
 - c. 点击“导出到”域旁的“浏览”按钮，打开保存为窗口。
 - d. 在“文件名”区域输入 **STUDENT_MAPPING.csv** 并点击“确定”。
4. 点击“完成”按钮在您的工作空间中创建 CSV 文件。

CSV 文件被创建并保存到您的工作空间路径下的 **University Info System** 文件夹内。例如，在一台 Windows 的机器中，它可能会被保存到如下路径：

```
C:\Documents and Settings\Administrator\IBM\workspaces\IDA_GettingStarted\University Info System\STUDENT_MAPPING.csv
```

您可以将该文件导入到其他支持*.csv 文件格式的工具中。

8.7 习练

1. 创建物理数据模型 **NEW_SDT_DDL.dbm** 和物理数据模型 **STUDENT_INFO_SYSTEM.dbm** 之间的映射模型。
2. 运行发现关系工具并接受或者拒绝映射模型中的映射。
3. 如果必要，创建新的映射。
4. 选择一个映射并添加变换功能。
5. 从映射模型中生成一个 SQL 脚本。
6. 创建一个可供其他工具使用的 CSV 文件。

8.8 总结

在本章中，您学习了映射模型以及映射模型如何被用来管理不同的数据源和数据模型之间的关系。您学习了如何创建映射模型和在您的数据模型间发现潜在的关系，您也学习了如何手动地在对象之间创建关系。

然后您可以生成一个 SQL 文件并通过它可以在服务器上创建关系，也可以生成一个 CSV 文件并通过它在其他工具中使用映射。

8.9 本章回顾

1. 什么是映射模型？
2. 为什么需要创建映射模型？
3. 映射有哪些不同的类型？
4. 下面的变换功能哪个可以被添加到映射组中？
 - a. 过滤
 - b. 排序
 - c. 连接
 - d. 上面所有项
5. 正误判断：您可以将多个源对象映射到单一的目标对象。

9

第九章 – 分析数据模型

IBM InfoSphere Data Architect 是一个灵活的建模工具，这意味着您可以创建任何合适您的业务需求的模型，工作台不会对建模工作有任何的限制。

在模型设计的时候，您可能无意中违反一些逻辑或物理数据建模的规则或约束，这将会导致模型中含有错误或导致模型不符合通常数据建模的最佳实践。

在这一章中，利用模型分析向导可以帮助您识别出模型中的问题，您将学习到以下的概念：

- 如何分析数据模型
- 如何通过设置选项参数来控制模型分析
- 如何在分析之后解决模型中的错误和警告

9.1 分析数据模型：概述

在模型创建之后我们会对模型进行分析，分析向导将会从如下两个角度对模型进行分析：

1. **设计建议：** 设计建议是一些关于建模的最佳实践或者设计约束，用来确保设计的模型符合标准。
2. **语法检查：** 语法检查是用来验证属性的语法是否正确和检查实体或表之间的关联是否有效。

理想状况下，在您转换或者部署模型之前进行模型分析，这将保证您不会操作无效的模型，使您能够在模型进入产品并引发潜在问题之前检查出模型中存在的问题。

9.2 使用工作台分析数据模型

9.2.1 使用工作台分析逻辑数据模型

您只能通过包节点来分析逻辑数据模型，下面我们对逻辑数据模型 `STUDENT_INFO_SYSTEM.ldm` 进行分析以确保它是有效的。

分析逻辑数据模型的步骤为：

1. 在逻辑数据模型中定位到包 `STUDENT INFO SYSTEM : University Info System` -> “数据模型” -> `STUDENT_INFO_SYSTEM.ldm` -> `STUDENT INFO SYSTEM`。
2. 右键单击包 `STUDENT INFO SYSTEM` 并选择“分析模型”，分析模型向导将会打开。

因为我们选择的是逻辑数据模型包，默认情况下，所有关于逻辑数据模型的规则集都被选择，这些规则集包含最佳实践、命名规范和通用设计建议，每一个规则集包含具体的规则内容。当您验证模型时，您可以选择最满足您的需求的规则。命名标准是否重要？是否需要域模型？数据模型是否需要完全的范式化？

3. 由于您之前已经对模型的命名规范进行验证，这里我们不选择“命名标准”规则集。
4. 由于您没有创建一个多维模型，我们不选择“维模型”规则集。
5. 点击“完成”进行模型分析。

模型分析完成后您将在问题视图中看到两条警告：

警告出自于对最佳实践、数据库的限制或组织的需求的违反。警告不意味模型的无效，也不需要再模型部署之前对其进行修改。警告仅仅指出那些潜在的、会带来模型的设计变化的冲突。您应该察看所有的警告并采取相应的操作确保您的模型不存在对数据库局限的违反，并遵循行业组织标准。无论如何，由您自己决定是否去更正那些和数据库最佳实践相关的警告。

在本练习中，问题视图中显示的两条警告指出在实体 **COURSE** 中的 **COURSE ID** 属性和实体 **STUDENT** 中的 **STUDENT ID** 属性不是唯一的。这些属性是在您为两个实体添加关联的时候增加的，因此，您不需要处理这些属性。

错误也显示在问题视图中，它表示数据模型中存在的必须被修改的问题。错误在生产环境中会导致问题，您必须在模型转换或者部署到生产环境之前将其更正。

9.2.2 使用工作台分析物理数据模型

下面让我们对物理数据模型 **STUDENT_INFO_SYSTEM.dbm** 进行分析。您可以从物理数据模型的数据库或者模式节点对物理数据模型进行分析。

分析物理数据模型的步骤为：

1. 在物理数据模型中定位到 **SAMPLE** 数据库节点：*University Info System* -> “数据模型” -> *STUDENT_INFO_SYSTEM.dbm* -> *SAMPLE*。
2. 右键单击 **SAMPLE** 数据库节点然后选择“分析模型”，打开分析模型向导。

由于我们在物理数据模型中选择了一个对象，默认情况下，所有的属于“物理数据模型”分类的规则集都会被选中。与逻辑数据模型分析一样，物理数据模型的分析包含多种规则来检查是否符合最佳实践、命名标准和设计建议。

3. 由于您没有创建多维模型，我们不选择“维模型”的规则集。
4. 因为我们的物理模型中没有表空间，请您确保有关表空间的验证规则不被选上。步骤如下：
 - a. 在“设计建议”节点下，选择“索引和存储器”规则集，索引和存储器的具体规则显示在“规则”面板中。
 - b. 取消“表空间设计检查”和“具有表空间的表检查”规则的选择。
 - c. 在“语法检查”节点下，取消“表空间”规则集的选择。
5. 取消“命名标准”规则集的选择。点击“完成”进行物理数据模型的分析。

分析完成后，您可以看到关于这个物理数据模型的 7 个警告和 13 条信息。如果您愿意，您可以通过问题视图定位和修复这些警告。

9.2.3 在问题视图中修复错误和警告

如果您想修复问题视图中的错误和警告，可以参考如下步骤：

1. 查看工具提供的完整的设计建议信息。
2. 双击信息定位到相关的数据对象。
3. 根据问题视图中提供的设计建议相应的修改数据对象。

注意：

您可以在问题视图中通过右键单击一条信息然后选择“快速修正”来修复一部分问题。

9.3 修改模型分析的首选项

您可以修改默认的模型分析类别和规则。按照如下步骤，在每次使用模型分析向导时，您将不必要花费时间在选中和不选中一些常用的模型分析规则。

注意：

在本节中，您不必要修改这些规则的首选项，但是您会学习到如何在首选项窗口中找到这些设置的位置。

更改分析数据模型向导的首选项步骤如下：

1. 在主菜单中选择：“窗口”->“首选项”，首选项窗口将会被打开。
2. 展开“模型验证”节点并选择“约束”页面。在这个页面中您可以设置那些在模型分析时默认被选中或取消的分类和规则。
例如，在对没有使用命名标准的逻辑数据模型进行分析的时候，您可以展开：“逻辑数据模型”->“设计建议”节点，然后取消“命名标准”分类的选择。
3. 一旦您完成您的选择，点击“应用”按钮，然后点击“确定”按钮。下一次，当您进行模型验证分析的时候，您在首选项中的修改将会生效。

9.4 总结

在本章中，您学习了如何对模型进行分析验证，以及通过分析如何确保模型符合一定的组织标准或者标准的语法规则。您还学习了如何在问题视图中查看并处理错误和警告。

9.5 练习

为模型分析向导修改首选项中的默认设置。

1. 首先，在不改变首选项的情况下，分析您的物理或逻辑数据模型并记录结果。
2. 其次，打开模型分析向导的首选项，为物理数据模型和逻辑数据模型取消命名标准、表空间和维模型的分析验证的选择。
3. 分析您的物理数据模型和逻辑数据模型，并记录结果。不要修复任何警告和错误。
4. 然后，打开首选项，恢复模型分析工具的默认设置。

9.6 本章回顾

1. 在数据模型的什么层次节点上可以调用模型分析向导？
 - A. 模型文件
 - B. 图
 - C. 包，数据库或模式
 - D. 实体或表
2. 当您进行模型分析的时候会进行下面哪种类型的验证？
 - A. 最佳实践
 - B. 模型语法
 - C. 设计的有效性
 - D. 以上所有
3. 警告可以指出下面哪种潜在的问题：
 - A. 数据库局限和最佳实践
 - B. 组织需求
 - C. A 和 B
 - D. 以上都不正确
4. 怎样改变模型分析向导的默认首选项设置？
5. 分析模型的最佳时机是什么时候？
6. 当您分析模型或者决定是否处理警告的时候应该注意什么？
7. 正误判断：您必须修改问题视图中的警告。
8. 正误判断：如果在问题视图中出现错误，那您的模型将被认为是无效的。

10

第十章 – 数据管理的生命周期

10.1 管理您的数据

Data Security, Privacy, 和 Lifecycle Management 产品组合是一个统一的解决方案集合。该解决方案集合能够遍历贯穿于软件开发周期中从需求收集到项目退役的所有阶段所涉及到的数据管理工作。IBM InfoSphere Data Architect 是 Data Security, Privacy, 和 Lifecycle Management 产品组合的一部分。该产品组合中的一些流行的产品有：

- IBM InfoSphere Data Architect
- Optim Development Studio and pureQuery Runtime
- Optim Database Administrator
- Optim Performance Manager Extended Edition
- Optim High Performance Unload
- Optim Query Tuner

下面的列表包含了对以上 Data Security, Privacy, 和 Lifecycle Management 产品家族的产品概括性描述：

- **Optim Development Studio**

Optim Development Studio 允许您开发和管理数据库。借助 Optim Development Studio, 您能创建、更改和删除对象；管理数据库特权；查看数据库分布以及从数据对象搜集统计信息。您也能开发和测试 SQL 和 XQuery 语句、存储过程、Web Services 和 Java 数据访问层。

Optim pureQuery Runtime 是一个能与 Optim Development Studio 共同使用的附加组件。Optim pureQuery Runtime 是一个高性能的数据访问平台，它能使监控、开发、优化、安全以及数据管理工作变得更加容易。通过使用其提供的 APIs 来简化最佳实践以及建立开发者和管理员之间的合作，或者通过监控您的 SQL 热点来获得能够改进反应时间的洞察力。

- **Optim Database Administrator**

Optim Database Administrator 使您能够管理复杂的数据库结构的变化以及管理数据库迁移方案。您能够通过自动化结构的变化来更好地管理您的测试数据库以及加速部署、流水

化管理的任务。通过分析数据库变化带来的影响来最小化您部署环境的风险。您也能够通过使用 **Optim Database Administrator** 更加有效地管理大规模的环境。

- **Optim Performance Manager Extended Edition**

Optim Performance Manager 帮助数据库管理员改进 **DB2** 性能和可用性。**DB2 Performance Expert** 帮助您在性能问题影响到业务前就对其进行识别、诊断、解决以及预防。**Optim Performance Manager** 提供优化和调优的建议，并帮助您分析趋势来为数据库增长制定计划。它监控您的应用程序和数据库、识别潜在的问题并在您的工作负载影响到性能前就对其进行管理。

- **Optim High Performance Unload**

Optim High Performance Unload 帮助数据库管理员以较少的代价但更快的速度来对很大规模的数据进行管理。**Optim High Performance Unload** 与生产系统并行地运行，在维护系统可用性的同时最大限度地降低对生产系统性能的影响。**Optim High Performance Unload** 还能将卸载、备份、迁移、重分区操作变为紧凑或缩紧的批窗口操作，来降低对在线生产系统环境的影响。

- **Optim Query Tuner**

Optim Query Tuner 帮助开发者创建高效的语句，以及培养他们的调优技巧。它通过提供有关如何写高质量的语句和改善数据库设计的专家建议来改善性能。

10.1.1 数据管理的生命周期

在先前章节列出的产品，既可以单独地也可以协同地为软件开发生命周期的每个阶段提供解决方案。图 10.1 阐明了这个能够与 IBM 的 **Data Security, Privacy, 和 Lifecycle Management** 产品组合得以实现的生命周期。

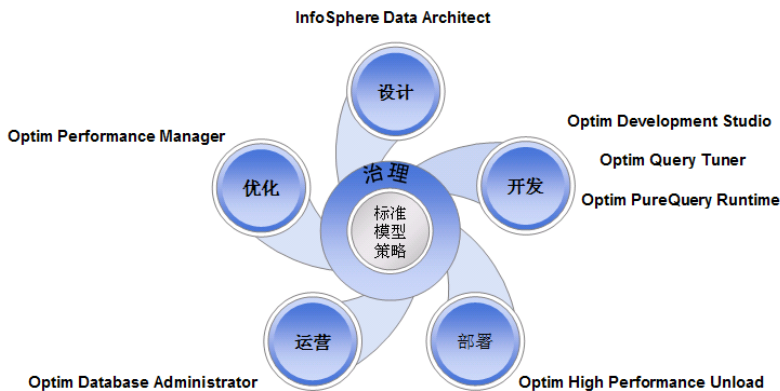


图 10.1 – **Data Security, Privacy, 和 Lifecycle Management** 工具

如之前的讨论，IBM **InfoSphere Data Architect** 主要适用于集成数据管理生命周期中的设计阶段，尽管对设计的更改通常贯穿任何项目的生命周期。您能使用工作台去管理这些改变。

Data Security, Privacy, 和 Lifecycle Management 产品组合中的所有这些产品以一个唯一的解决方案呈现在您面前(参考有关 IBM **InfoSphere Data Architect** 集成能力的段落 10.1.2 和 有关 **shell-sharing** 的段落

10.1.3)。携手 Data Security, Privacy, 和 Lifecycle Management 产品组合，整个产品生命周期能够被管理。由于这些产品组合很好地解决了协同性和兼容性的问题，您能够获得以下的改善：

- 更快地开发可用于企业部署的应用。
- 高效地管理数据增长。
- 优化数据库应用性能。
- 在产品或应用生命周期的所有阶段保护数据隐私。
- 简化迁移和升级。

10.1.2 IBM InfoSphere Data Architect 与其它产品集成

在任意一个软件开发团队里，无论其规模大小，您都需要能够准确无误地分享数据或信息的工具。这使得在产品开发的阶段中重用所有的信息变得简单。

IBM InfoSphere Data Architect 通过和以下产品的集成来支持模型驱动的设计：

- InfoSphere Business Glossary: 贯穿业务的企业词汇表共享。
- Cognos Business Intelligence: 创建、分享和使用来自您的数据库信息的报表。
- Rational Software Architect: 同步应用和信息的设计。
- Rational RequisitePro: 确保模型对于业务需求的完整性。
- WebSphere® Business Modeler Advanced: 将数据模型转换为应用于业务流程的 XML 模式定义。

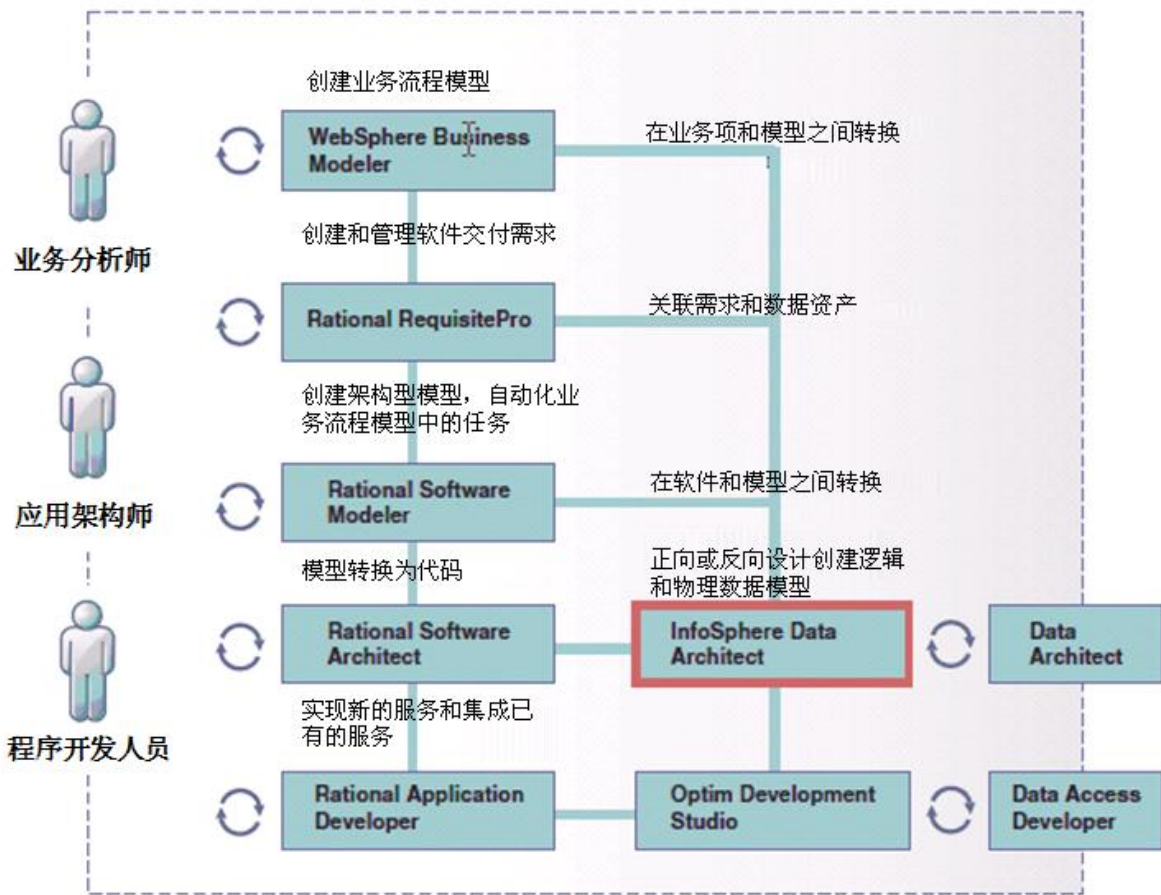


图 10.2 – 将 InfoSphere Data Architect 与其它软件开发产品进行集成

10.1.3 于其它其同Eclipse行进品产的shell-sharing

IBM InfoSphere Data Architect 创建于 Eclipse 平台之上。Eclipse 平台是一个流行的应用开发平台，它允许您将它和其它基于 Eclipse 的产品进行集成。

在基于 Eclipse 的产品之间进行 shell-sharing 意味着您能够通过共享产品间的相同核心组件来降低这些组件在不同产品里的冗余。它使每个产品都无需独属于自己的 Eclipse 平台。可以通过与 Microsoft Windows 的安装方式进行一个简单的类比来理解 shell-sharing: 所有 Windows 程序文件所依赖的共同组件都被存放在文件夹 `C:\Windows` 下。

从基本的层次上来看，shell-sharing 节省了磁盘空间并避免了相同组件的重复安装。Shell-sharing 通过共享通用和兼容的组件来达到这个目标。因此，您能通过一个唯一的用户接口与每一个产品的所有功能进行交互。比如，如果您把 IBM InfoSphere Data Architect 和 Optim Development Studio 彼此进行 shell-sharing，您就能在一个唯一和公共的平台上开发和设计数据库。这个公共的平台启动时不考虑究竟是哪个产品被启动了。

您能够 **shell-share** 任意数量的彼此兼容的产品。例如，为了使用 Rational Software Architect 的功能，您也可以把 Rational Software Architect 与 IBM InfoSphere Data Architect 和 Optim Development Studio 进行集成。

注意：

关于如何将 InfoSphere Data Architect, Optim Development Studio, 和 Optim Database Administrator 进行 **shell-sharing** 的例子，可以参考以下的 developerWorks 文章：

<http://www.ibm.com/developerworks/data/library/techarticle/dm-0811khatri/index.html>

参考文献

- [1] IBM® InfoSphere Data Architect Information Center, October 2010.
<http://publib.boulder.ibm.com/infocenter/rdahelp/v7r5/index.jsp>
- [2] LIVENGOOD, E. *InfoSphere Data Architect Evaluation Guide*, white paper, October 2010.
<http://download.boulder.ibm.com/ibmdl/pub/software/data/sw-library/studio/evaluation-guides/RDA75EvalGuide.pdf>
- [3] TSOUNIS, S., CHOU, D.P. *Best Practices for Rational Data Architect*, best practices paper, October 2008. <http://www.ibm.com/developerworks/data/bestpractices/informationmodeling/>
- [4] MACHADO, J., VOELLINGER, H. *Develop mapping models with IBM InfoSphere Data Architect*, developerWorks article, January 2010. <http://www.ibm.com/developerworks/data/tutorials/dm-1101mappingmodelsida/index.html>
- [5] PATEL, A. *DB2 basics: Table spaces and buffer pools*, developerWorks article, April 2010.
<http://www.ibm.com/developerworks/data/library/techarticle/0212wieser/index.html>
- [6] GOPAL, V., BHAGAVAN, S. *Data modeling with InfoSphere Data Architect and Informix Dynamic Server*, developerWorks article, March 2009. <http://www.ibm.com/developerworks/data/tutorials/dm-0903datamodel/section8.html>
- [7] LIU, W. *Use InfoSphere Data Architect to define and enforce data object naming standards*, developerWorks article, October 2010. <http://www.ibm.com/developerworks/data/library/techarticle/dm-0701liu/index.html>

资源

网站

DB2 Express-C 网站:

www.ibm.com/db2/express

使用该网站下载 DB2 Express - C 服务器、DB2 客户端、DB2 驱动程序和使用手册的介质，访问团队博客，邮件列表注册等。

DB2 Express-C 论坛:

www.ibm.com/developerworks/forums/dw_forum.jsp?forum=805&cat=19

使用该论坛来发布在使用手册中无法找到答案的技术问题。

DB2 信息中心:

<http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp>

信息中心提供的访问 DB2 在线使用手册。它是最新信息发布的来源。

InfoSphere Data Architect 信息中心:

<http://publib.boulder.ibm.com/infocenter/rdahelp/v7r5/index.jsp>

InfoSphere Data Architect 的信息中心提供在线访问使用手册和帮助文档，它包含帮助您产品入门的教程、概念信息和任务。

developerWorks: DB2

<http://www-128.ibm.com/developerworks/db2>

该网站为开发人员和 DBA 免费提供访问当前文章、教程等极好的资源。

developerWorks: InfoSphere Data Architect

<http://www.ibm.com/developerworks/data/products/dataarchitect/index.html>

该网站提供文章、教程和其它资源帮助您了解更多关于如何使用 InfoSphere Data Architect 进行数据建模。

InfoSphere Data Architect developerWorks forum

<http://www.ibm.com/developerworks/forums/forum.jspa?forumID=1796>

该论坛是非官方支持的来源，它是由 IBM InfoSphere Data Architect 的专家和爱好者来管理的。

alphaWorks®

<http://www.alphaworks.ibm.com/>

该网站提供了直接访问 IBM 的新兴技术。它是一个可以找到来自 IBM 研究中心最新技术的地点。

planetDB2

www.planetDB2.com

这是一个由许多贡献者发布的关于 DB2 的博客的聚合。

DB2 Technical Support

http://www.ibm.com/software/data/db2/support/db2_9/

如果您已购买 DB2 Express - C 的 12 个月的订购许可证，您可以从这个网站下载补丁包。

InfoSphere Data Architect technical support

http://www.ibm.com/support/entry/portal/Overview/Software/Information_Management/InfoSphere_Data_Architect

这是对 InfoSphere Data Architect 提供支持门户网站。使用该网站可以看到最新的支持和故障排除信息，也可以看到与产品有关的其他文件或者联系 IBM 支持。

ChannelDB2

<http://www.ChannelDB2.com/>

ChannelDB2 是 DB2 社区的社会网络，它的特色在 Linux, Unix, Windows, z/OS 和 i5/OS 平台上的与 DB2 有关的视频、演示、博客、播客、讨论和资源等。

书籍

1. 免费红宝书: DB2 Express-C: The Developer Handbook for XML, PHP, C/C++, Java, and .NET
Whei-Jen Chen, John Chun, Naomi Ngan, Rakesh Ranjan, Manoj K. Sardana,
August 2006 - SG24-7301-00
<http://www.redbooks.ibm.com/abstracts/sq247301.html?Open>
2. Understanding DB2 – Learning Visually with Examples V9.5
Raul F. Chong, et all. January 2008
ISBN-10: 0131580183
3. DB2 9: pureXML overview and fast start by Cynthia M. Saracco, Don Chamberlin, Rav Ahuja
June 2006 SG24-7298
<http://www.redbooks.ibm.com/abstracts/sq247298.html?Open>
4. DB2® SQL PL: Essential Guide for DB2® UDB on Linux™, UNIX®, Windows™, i5/OS™, and z/OS®, 2nd Edition
Zamil Janmohamed, Clara Liu, Drew Bradstock, Raul Chong, Michael Gao, Fraser McArthur, Paul Yip
ISBN: 0-13-100772-6
5. 免费红宝书: DB2 pureXML Guide
Whei-Jen Chen, Art Sammartino, Dobromir Goutev, Felicity Hendricks, Ippei Komi, Ming-Pang Wei, Rav Ahuja, Matthias Nicola. August 2007
<http://www.redbooks.ibm.com/abstracts/sq247315.html?Open>
6. Information on Demand - Introduction to DB2 9 New Features
Paul Zikopoulos, George Baklarz, Chris Eaton, Leon Katsnelson
ISBN-10: 0071487832
ISBN-13: 978-0071487832

联系邮箱

General DB2 Express-C mailbox: db2x@ca.ibm.com

General DB2 on Campus program mailbox: db2univ@ca.ibm.com

InfoSphere Data Architect快速入门.

本书的用途:

- 了解IDA能为您做什么
- 学习如何创建、部署和更新数据模型
- 管理和分析数据模型变化的影响
- 探索如何对您的数据资源进行正向和反向设计
- 了解更多关于**Data Security, Privacy, 和 Lifecycle Management**的内容
- 实际动手练习

InfoSphere Data Architect是一个协作的数据设计解决方案，您可以用它来发现，建模，可视化展示，关联和标准化多样的分布式数据资产。在整个数据生命周期里，您可以使用InfoSphere Data Architect使数据模型的价值跨越数据的生命周期，从而提供跨生命周期、跨角色和跨组织的项目的协作和确保流程服务、应用以及数据架构的一致性。

本书对InfoSphere Data Architect提供了一个全面介绍并包含使用SQL和DB2 Express - C（免费版本的DB2）的动手练习。

需要了解更多内容或下载试用版的InfoSphere Data Architect，可以访问 ibm.com/developerworks/downloads/r/rda/

需要了解更多内容或下载DB2 Express-C，可以访问 ibm.com/db2/express

需要观看相关的视频信息，可以访问 channelDB2.com

这本书是DB2校园系列丛书之一，电子书对社区免费。想了解更多可以访问 db2university.com

ISBN 978-0-9866283-9-9



Price: 24.99USD