

IBM DB2 Information Integrator



# 数据源配置指南

版本 8.2



IBM DB2 Information Integrator



# 数据源配置指南

版本 8.2

在使用本资料及其支持的产品之前，请阅读第 595 页的『声明』中的一般信息。

本文档包含 IBM 的专利信息。它在许可协议下提供，并受版权法保护。本出版物包含的信息不包括任何产品保证，且本手册提供的任何声明不应作如此解释。

可以在线方式或通过您当地的 IBM 代表订购 IBM 出版物。

- 要以在线方式订购出版物，可访问“IBM 出版物中心”（IBM Publications Center），网址为 [www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)
- 要查找您当地的 IBM 代表，可访问“IBM 全球联系人目录”（IBM Directory of Worldwide Contacts），网址为 [www.ibm.com/planetwide](http://www.ibm.com/planetwide)

当您发送信息给 IBM 后，即授予 IBM 非专有权，IBM 可以它认为合适的任何方式使用或分发此信息，而无须对您承担任何责任。

© Copyright International Business Machines Corporation 1998, 2004. All rights reserved.

# 目录

关于本书 . . . . .	<b>xi</b>
谁应阅读本书 . . . . .	xi
本书中使用的约定和术语 . . . . .	xi
如何阅读语法图 . . . . .	xii

## 第 1 部分 概念和规划 . . . . . 1

### 第 1 章 联合系统概述 . . . . . 3

联合系统 . . . . .	3
联合服务器 . . . . .	4
什么是数据源? . . . . .	4
受支持的数据源 . . . . .	5
联合数据库 . . . . .	7
联合数据库系统目录 . . . . .	7
SQL 编译器 . . . . .	8
查询优化器 . . . . .	8
补偿 . . . . .	9
传递 (Pass-Through) 会话 . . . . .	9
包装器和包装器模块 . . . . .	10
缺省包装器名称 . . . . .	11
服务器定义和服务器选项 . . . . .	12
用户映射 . . . . .	12
昵称和数据源对象 . . . . .	13
有效数据源对象 . . . . .	14
昵称列选项 . . . . .	15
数据类型映射 . . . . .	15
函数映射 . . . . .	16
索引规范 . . . . .	16
如何与联合系统交互作用 . . . . .	17
DB2 命令行处理器 (CLP) . . . . .	17
DB2 命令中心 . . . . .	17
DB2 控制中心 . . . . .	17
应用程序 . . . . .	18
DB2 系列工具 . . . . .	18
Web 服务提供程序 . . . . .	18

### 第 2 章 计划联合数据源配置 . . . . . 19

联合对象命名规则 . . . . .	19
保留联合系统中区分大小写的值 . . . . .	20
更新数据源统计信息 . . . . .	21
选择正确的包装器 . . . . .	22
用来存取 Excel 数据的方法 . . . . .	22
规划用户映射 . . . . .	25
规划数据类型映射 . . . . .	26
规划函数映射 . . . . .	27
用于规划联合系统配置的核对表 . . . . .	27
核对表: 联合对象命名规则 . . . . .	27
核对表: 保留区分大小写的值 . . . . .	27
核对表: 数据源统计信息 . . . . .	28
核对表: 数据类型映射 . . . . .	28

核对表: 用户映射 . . . . .	29
核对表: 包装器 . . . . .	29

## 第 2 部分 联合服务器和数据库 . . . . . 31

### 第 3 章 检查联合服务器的设置 . . . . . 33

检查联合服务器的设置 . . . . .	33
确认包装器库文件的链接编辑 (UNIX) . . . . .	33
确认包装器库文件的链接编辑 (UNIX) . . . . .	33
检查包装器库文件 (UNIX) . . . . .	34
包装器库文件 . . . . .	35
检查链接编辑消息文件 (UNIX) . . . . .	35
将包装器库手工链接至数据源客户机软件 . . . . .	36
检查 FEDERATED 参数 . . . . .	38

### 第 4 章 创建联合数据库 . . . . . 39

联合数据库本地语言注意事项 . . . . .	39
联合系统中的整理顺序 . . . . .	39
设置联合数据库整理顺序 . . . . .	41
联合系统的 Unicode 支持 . . . . .	42
创建联合数据库 . . . . .	44

## 第 3 部分 数据源 . . . . . 47

### 第 5 章 配置对数据源的存取概述 . . . . . 49

配置数据源的快速方法 . . . . .	49
使用 DB2 UDB 控制中心将数据源添加至联合服务器 . . . . .	50
配置多个联合服务器来存取数据源 . . . . .	51
设置数据源环境变量 . . . . .	52
设置数据源环境变量 . . . . .	52
对于 db2dj.ini 文件的限制 . . . . .	53
在多分区实例配置中应用环境变量 . . . . .	53
为数据源注册包装器 . . . . .	54
为数据源注册服务器定义 . . . . .	55
为数据源注册用户映射 . . . . .	56
为数据源注册昵称 . . . . .	56
为数据源注册昵称 . . . . .	56
对昵称创建昵称 . . . . .	58
为非关系数据源指定昵称列 . . . . .	58
可选的配置步骤 . . . . .	59

### 第 6 章 配置对 BioRS 数据源的存取 . . . . . 61

什么是 BioRS? . . . . .	61
将 BioRS 添加至联合服务器 . . . . .	62
将 BioRS 数据源添加至联合服务器 . . . . .	62
为 BioRS 包装器注册定制函数 . . . . .	63
定制函数表 - BioRS 包装器 . . . . .	64
注册 BioRS 包装器 . . . . .	65
BioRS 包装器库文件 . . . . .	65
为 BioRS 数据源注册服务器定义 . . . . .	66

为 BioRS 数据源注册用户映射 . . . . .	66
为 BioRS 数据源注册昵称 . . . . .	67
CREATE NICKNAME 语句 - BioRS 包装器的示 例 . . . . .	69
BioRS 数据源的查询和定制函数 . . . . .	70
定制函数和 BioRS 查询 . . . . .	70
BioRS 包装器的等值连接谓词 . . . . .	72
BioRS AllText 元素 . . . . .	74
BioRS 包装器 - 示例查询 . . . . .	74
优化 BioRS 包装器性能 . . . . .	79
有关优化 BioRS 包装器性能的准则 . . . . .	79
BioRS 统计信息 . . . . .	80
确定 BioRS 数据银行基数统计信息 . . . . .	80
更新 BioRS 昵称基数统计信息 . . . . .	81
更新 BioRS 列基数统计信息 . . . . .	81
更新 BioRS _ID_ 列基数 . . . . .	83
BioRS 包装器的消息 . . . . .	83

## 第 7 章 配置对 BLAST 数据源的存取 . . 87

什么是 BLAST? . . . . .	87
将 BLAST 添加至联合服务器 . . . . .	90
将 BLAST 数据源添加至联合服务器 . . . . .	90
验证是否安装了正确版本的 blastall 可执行文件和 矩阵文件 . . . . .	91
配置 BLAST 守护进程 . . . . .	91
BLAST 守护进程配置文件 - 示例 . . . . .	94
启动 BLAST 守护进程 . . . . .	95
db2blast_daemon 命令 - 语法和示例 . . . . .	96
注册 BLAST 包装器 . . . . .	97
BLAST 包装器库文件 . . . . .	97
为 BLAST 数据源注册服务器 . . . . .	98
为 BLAST 数据源注册昵称 . . . . .	98
设置 TurboBlast 以使用 BLAST 包装器 . . . . .	104
构造 BLAST SQL 查询 . . . . .	105
BLAST 数据源 - 示例查询 . . . . .	105
BLAST 包装器的优化提示 . . . . .	107
BLAST 包装器的消息 . . . . .	107

## 第 8 章 配置对商业应用程序数据源的存 取 . . . . . 109

WebSphere Business Integration 包装器 . . . . .	109
业务对象定义 . . . . .	111
配置 WebSphere Business Integration 适配器 . . . . .	112
将商业应用程序添加至联合服务器 . . . . .	115
将商业应用程序数据源添加至联合系统 . . . . .	115
注册 WebSphere Business Integration 包装器 . . . . .	116
WebSphere Business Integration 包装器库文件 . . . . .	116
为商业应用程序数据源注册服务器定义 . . . . .	117
为商业应用程序数据源注册昵称 . . . . .	118
对于商业应用程序和 Web 服务的包装器的查询限制 . . . . .	139
商业应用程序数据源 - 示例查询 . . . . .	142

## 第 9 章 配置对 DB2 系列数据源的存取 145

将 DB2 系列数据源添加至联合服务器 . . . . .	145
将 DB2 系列数据源添加至联合服务器 . . . . .	145
在联合节点目录中编目节点条目 . . . . .	146

在联合系统数据库目录中编目远程数据库 . . . . .	147
注册 DB2 包装器 . . . . .	147
DB2 包装器库文件 . . . . .	148
为 DB2 数据源注册服务器定义 . . . . .	149
CREATE SERVER 语句 - DB2 包装器的示例 . . . . .	150
为 DB2 数据源创建用户映射 . . . . .	151
CREATE USER MAPPING 语句 - DB2 包装器 的示例 . . . . .	151
测试与 DB2 数据源服务器的连接 . . . . .	152
为 DB2 表和视图注册昵称 . . . . .	153
CREATE NICKNAME 语句 - DB2 包装器的示 例 . . . . .	154

## 第 10 章 配置对 Documentum 数据源 的存取 . . . . . 155

什么是 Documentum? . . . . .	155
将 Documentum 添加至联合服务器 . . . . .	157
将 Documentum 数据源添加至联合服务器 . . . . .	157
使 Documentum 客户机库可供包装器使用 . . . . .	158
设置 Documentum 环境变量 . . . . .	158
注册 Documentum 包装器 . . . . .	160
Documentum 包装器库文件 . . . . .	160
为 Documentum 数据源注册服务器 . . . . .	161
为 Documentum 数据源注册用户映射 . . . . .	161
为 Documentum 数据源注册昵称 . . . . .	162
Documentum 数据源的查询和定制函数 . . . . .	167
为 Documentum 包装器注册定制函数 . . . . .	167
Documentum 数据源 - 示例查询 . . . . .	172
Documentum 包装器的访问控制 . . . . .	173
Documentum 包装器的消息 . . . . .	173

## 第 11 章 配置对 Entrez 数据源的存取 179

什么是 Entrez? . . . . .	179
将 Entrez 添加至联合服务器 . . . . .	180
将 Entrez 数据源添加至联合服务器 . . . . .	180
为 Entrez 包装器注册定制函数 . . . . .	181
定制函数表 - Entrez 包装器 . . . . .	181
注册 Entrez 包装器 . . . . .	182
Entrez 包装器库文件 . . . . .	183
为 Entrez 数据源注册服务器 . . . . .	183
为 Entrez 数据源注册昵称 . . . . .	185
Entrez 数据源的查询和定制函数 . . . . .	186
定制函数和 Entrez 查询 . . . . .	186
Entrez 包装器的关系谓词 . . . . .	188
Entrez 包装器的无效 WHERE 子句 . . . . .	188
模式数据元素规范 . . . . .	188
Entrez 数据源 - 示例查询 . . . . .	189
PubMed 模式表 . . . . .	190
Nucleotide 模式表 . . . . .	194
Entrez 包装器的消息 . . . . .	197

## 第 12 章 配置对 Excel 数据源的存取 201

什么是 Excel? . . . . .	201
将 Excel 添加至联合服务器 . . . . .	202
将 Excel 数据源添加至联合服务器 . . . . .	202
注册 Excel 包装器 . . . . .	203

	Excel 包装器库文件 . . . . .	204
	为 Excel 数据源注册服务器 . . . . .	204
	为 Excel 数据源注册昵称 . . . . .	205
	Excel 数据源 - 示例查询 . . . . .	205
	Excel 数据源 - 样本方案 . . . . .	206
	Excel 包装器的文件访问控制模型 . . . . .	207
	Excel 包装器的消息 . . . . .	207

### 第 13 章 配置对 Extended Search 数据源的存取 . . . . . 213

	什么是 Extended Search? . . . . .	213
	Extended Search 数据源 . . . . .	214
	Extended Search 包装器的工作方式 . . . . .	214
	Extended Search 昵称 . . . . .	215
	Extended Search 垂直表 . . . . .	216
	将 Extended Search 添加至联合服务器 . . . . .	218
	将 Extended Search 数据源添加至联合服务器 . . . . .	218
	注册 Extended Search 包装器 . . . . .	219
	Extended Search 包装器库文件 . . . . .	220
	为 Extended Search 数据源注册服务器 . . . . .	220
	为 Extended Search 数据源注册用户映射 . . . . .	221
	为 Extended Search 数据源注册昵称 . . . . .	221
	为 Extended Search 包装器注册定制函数 . . . . .	222
	查询 Extended Search 数据源 . . . . .	223
	Extended Search 包装器 - 查询指南 . . . . .	223
	Extended Search 包装器 - 示例查询 . . . . .	225
	Extended Search 包装器 - 通用查询语言 . . . . .	227
	Extended Search 包装器的消息 . . . . .	228

### 第 14 章 配置对 HMMER 数据源的存取 233

	什么是 HMMER? . . . . .	233
	将 HMMER 添加至联合服务器 . . . . .	236
	将 HMMER 数据源添加至联合服务器 . . . . .	236
	验证 HMMER 程序可执行文件的版本 . . . . .	237
	配置 HMMER 守护进程 . . . . .	237
	HMMER 守护进程配置文件 - 示例 . . . . .	240
	启动 HMMER 守护进程 . . . . .	241
	db2hmmr_daemon 命令 - 语法和示例 . . . . .	242
	注册 HMMER 包装器 . . . . .	243
	HMMER 包装器库文件 . . . . .	244
	为 HMMER 数据源注册服务器定义 . . . . .	244
	CREATE SERVER 语句 - HMMER 包装器的示 例 . . . . .	245
	为 HMMER 数据源注册昵称 . . . . .	246
	HMMER 昵称的固定列 . . . . .	247
	CREATE NICKNAME 语句 - HMMER 包装器 的示例 . . . . .	250
	HMMER 数据源 - 完整示例 . . . . .	250
	使用样本来构造新的 HMMER 查询 . . . . .	252
	HMMER 包装器的消息 . . . . .	253

### 第 15 章 配置对 Informix 数据源的存取 . . . . . 255

	将 Informix 添加至联合服务器 . . . . .	255
	将 Informix 数据源添加至联合服务器 . . . . .	255
	设置和测试 Informix 客户机配置文件 . . . . .	256

	设置 Informix 环境变量 . . . . .	257
	注册 Informix 包装器 . . . . .	260
	Informix 包装器库文件 . . . . .	261
	为 Informix 数据源注册服务器定义 . . . . .	261
	CREATE SERVER 语句 - Informix 包装器的示 例 . . . . .	262
	为 Informix 数据源创建用户映射 . . . . .	263
	CREATE USER MAPPING 语句 - Informix 包 装器的示例 . . . . .	264
	测试与 Informix 服务器的连接 . . . . .	265
	为 Informix 表、视图和同义词注册昵称 . . . . .	266
	CREATE NICKNAME 语句 - Informix 包装器 的示例 . . . . .	266
	对 Informix 数据源的配置进行调整和故障诊断 . . . . .	267
	通过设置 FOLD_ID 和 FOLD_PW 服务器选项来 提高性能 . . . . .	267

### 第 16 章 配置对 Microsoft SQL Server 数据源的存取 . . . . . 269

	将 Microsoft SQL Server 添加至联合服务器 . . . . .	269
	将 Microsoft SQL Server 数据源添加至联合服 务器 . . . . .	269
	准备联合服务器以存取 Microsoft SQL Server 数 据源 . . . . .	270
	设置 Microsoft SQL Server 环境变量 . . . . .	271
	注册 Microsoft SQL Server 包装器 . . . . .	273
	Microsoft SQL Server 包装器库文件 . . . . .	274
	为 Microsoft SQL Server 数据源注册服务器定义 CREATE SERVER 语句 - Microsoft SQL Server 包装器的示例 . . . . .	274
	为 Microsoft SQL Server 数据源创建用户映射 CREATE USER MAPPING 语句 - Microsoft SQL Server 包装器的示例 . . . . .	277
	测试与 Microsoft SQL Server 远程服务器的连接 为 Microsoft SQL Server 表和视图注册昵称 . . . . .	278
	CREATE NICKNAME 语句 - Microsoft SQL Server 包装器的示例 . . . . .	279
	对 Microsoft SQL Server 数据源的配置进行调整和 故障诊断 . . . . .	280
	获取 ODBC 跟踪 . . . . .	281

### 第 17 章 配置对 ODBC 数据源的存取 283

	将 ODBC 添加至联合系统 . . . . .	283
	将 ODBC 数据源添加至联合服务器 . . . . .	283
	准备联合服务器以通过 ODBC 来存取数据源 . . . . .	284
	注册 ODBC 包装器 . . . . .	285
	ODBC 包装器库文件 . . . . .	286
	CREATE WRAPPER 语句 - ODBC 包装器的示 例 . . . . .	286
	为 ODBC 数据源注册服务器定义 . . . . .	287
	CREATE SERVER 语句 - ODBC 包装器的示例 为 ODBC 数据源创建用户映射 . . . . .	288
	CREATE USER MAPPING 语句 - ODBC 包装 器的示例 . . . . .	289
	测试与 ODBC 数据源服务器的连接 . . . . .	290
	为 ODBC 数据源表和视图注册昵称 . . . . .	291

CREATE NICKNAME 语句 - ODBC 包装器的 示例 . . . . .	292
使用 ODBC 包装器存取 Excel 数据 . . . . .	293
改变缺省数据类型映射 . . . . .	295
ALTER SERVER 语句 - ODBC 包装器存取 Excel 数据的示例 . . . . .	295
对 ODBC 数据源的配置进行调整和故障诊断 . . . . .	296
获取 ODBC 跟踪 . . . . .	296

**第 18 章 配置对 OLE DB 数据源的存取 . . . . . 297**

将 OLE DB 数据源添加至联合服务器 . . . . .	297
注册 OLE DB 包装器 . . . . .	298
OLE DB 包装器库文件 . . . . .	298
为 OLE DB 数据源注册服务器定义 . . . . .	299
CREATE SERVER 语句 - OLE DB 包装器的示例 . . . . .	299
为 OLE DB 数据源创建用户映射 . . . . .	300
CREATE USER MAPPING 语句 - OLE DB 包装 器的示例 . . . . .	301

**第 19 章 配置对 Oracle 数据源的存取 . . . . . 303**

将 Oracle 添加至联合系统 . . . . .	303
将 Oracle 数据源添加至联合服务器 . . . . .	303
设置 Oracle 环境变量 . . . . .	304
设置和测试 Oracle 客户机配置文件 . . . . .	307
注册 Oracle 包装器 . . . . .	308
Oracle 包装器库文件 . . . . .	309
为 Oracle 数据源注册服务器定义 . . . . .	309
CREATE SERVER 语句 - Oracle 包装器的示例 . . . . .	310
为 Oracle 数据源创建用户映射 . . . . .	311
CREATE USER MAPPING 语句 - Oracle 包装 器的示例 . . . . .	312
测试与 Oracle 服务器的连接 . . . . .	312
为 Oracle 表和视图注册昵称 . . . . .	313
CREATE NICKNAME 语句 - Oracle 包装器的 示例 . . . . .	314
对 Oracle 数据源的配置进行调整和故障诊断 . . . . .	314
连接问题 . . . . .	315

**第 20 章 配置对 Sybase 数据源的存取 . . . . . 317**

将 Sybase 添加至联合系统 . . . . .	317
将 Sybase 数据源添加至联合服务器 . . . . .	317
设置 Sybase 环境变量 . . . . .	318
设置和测试 Sybase 客户机配置文件 . . . . .	320
注册 Sybase 包装器 . . . . .	321
Sybase 包装器库文件 . . . . .	322
为 Sybase 数据源注册服务器定义 . . . . .	322
CREATE SERVER 语句 - Sybase 包装器的示 例 . . . . .	323
为 Sybase 数据源创建用户映射 . . . . .	325
CREATE USER MAPPING 语句 - Sybase 包装 器的示例 . . . . .	325
测试与 Sybase 服务器的连接 . . . . .	326
为 Sybase 表和视图注册昵称 . . . . .	327
CREATE NICKNAME 语句 - Sybase 包装器的 示例 . . . . .	328

对 Sybase 数据源的配置进行调整和故障诊断 . . . . .	328
解决 sp_helpindex 错误 . . . . .	328

**第 21 章 配置对表结构文件数据源的存取 . . . . . 329**

什么是表结构文件? . . . . .	329
表结构文件的属性 . . . . .	329
已排序的文件 . . . . .	329
未排序的文件 . . . . .	330
DB2 Information Integrator 如何使用表结构文件 . . . . .	330
将表结构文件添加至联合系统 . . . . .	331
将表结构文件数据源添加至联合服务器 . . . . .	331
注册表结构文件包装器 . . . . .	332
表结构文件包装器库文件 . . . . .	332
为表结构文件注册服务器 . . . . .	333
为表结构文件注册昵称 . . . . .	333
CREATE NICKNAME 语句 - 表结构文件包装 器的示例 . . . . .	334
表结构文件包装器的文件访问控制模型 . . . . .	335
表结构文件包装器的优化提示和注意事项 . . . . .	335
表结构文件包装器的消息 . . . . .	335

**第 22 章 配置对 Teradata 数据源的存取 . . . . . 339**

将 Teradata 添加至联合系统 . . . . .	339
将 Teradata 数据源添加至联合服务器 . . . . .	339
测试与 Teradata 服务器的连接 . . . . .	340
验证是否为运行时链接启用了 Teradata 库 (AIX) . . . . .	341
设置 Teradata 环境变量 . . . . .	341
注册 Teradata 包装器 . . . . .	344
Teradata 包装器库文件 . . . . .	344
为 Teradata 数据源注册服务器定义 . . . . .	345
CREATE SERVER 语句 - Teradata 包装器的示 例 . . . . .	346
为 Teradata 数据源创建用户映射 . . . . .	347
CREATE USER MAPPING 语句 - Teradata 包 装器的示例 . . . . .	348
测试从联合服务器至 Teradata 服务器的连接 . . . . .	348
联合服务器上的 Teradata 昵称 . . . . .	349
为 Teradata 表和视图注册昵称 . . . . .	350
CREATE NICKNAME 语句 - Teradata 包装器 的示例 . . . . .	350
对 Teradata 数据源的配置进行调整和故障诊断 . . . . .	351
对昵称的 UPDATE 或 DELETE 操作错误 . . . . .	351
调整和禁用 Teradata 存取记录 . . . . .	351
对 libcliv2.so 启用运行时链接 (AIX) . . . . .	351

**第 23 章 配置对 Web 服务数据源的存取 . . . . . 353**

Web 服务包装器和 Web 服务描述语言文档 . . . . .	353
将 Web 服务添加至联合系统 . . . . .	358
将 Web 服务数据源添加至联合服务器 . . . . .	358
注册 Web 服务包装器 . . . . .	359
Web 服务包装器库文件 . . . . .	360



为 Web 服务数据源注册服务器定义 . . . . .	360
为 Web 服务数据源注册昵称 . . . . .	361
对于商业应用程序和 Web 服务的包装器的查询限制	375
Web 服务数据源 - 示例查询 . . . . .	378
Web 服务包装器安全性 . . . . .	381
Web 服务包装器的消息 . . . . .	381

## 第 24 章 配置对 XML 数据源的存取 385

什么是 XML? . . . . .	385
将 XML 添加至联合系统 . . . . .	388
将 XML 添加至联合系统 . . . . .	388
注册 XML 包装器 . . . . .	389
XML 包装器库文件 . . . . .	389
为 XML 数据源注册服务器 . . . . .	390
为 XML 数据源注册昵称 . . . . .	391
为非根昵称 (XML 包装器) 创建联合视图 . . . . .	399
XML 成本模型设施的优化技巧 . . . . .	399
XML 数据源 - 示例查询 . . . . .	400
XML 包装器的消息 . . . . .	401

## 第 4 部分 用户定义的函数 . . . . . 409

### 第 25 章 生命科学用户定义的函数 . . . . . 411

生命科学用户定义的函数 - 概述 . . . . .	411
生命科学用户定义的函数 - 概述 . . . . .	411
生命科学用户定义的函数库文件 . . . . .	411
按函数类别排列的生命科学用户定义的函数 . . . . .	412
注册生命科学用户定义的函数 . . . . .	413
禁用生命科学用户定义的函数 . . . . .	414
向后转换用户定义的函数 . . . . .	415
向后转换用户定义的函数 - 概述 . . . . .	415
LSPep2AmbNuc 用户定义的函数 . . . . .	415
LSPep2AmbNuc 用户定义的函数 - 示例 . . . . .	416
LSPep2AmbNuc 用户定义的函数 - 错误消息 . . . . .	418
LSPep2ProbNuc 用户定义的函数 . . . . .	418
LSPep2ProbNuc 用户定义的函数 - 示例 . . . . .	419
LSPep2ProbNuc 用户定义的函数 - 错误消息 . . . . .	420
定义行语法分析用户定义的函数 . . . . .	420
定义行语法分析用户定义的函数 - 概述 . . . . .	420
LSDeflineParse 用户定义的函数 . . . . .	421
LSDeflineParse 用户定义的函数 - 示例 . . . . .	423
一般化模式匹配用户定义的函数 . . . . .	426
LSPatternMatch 用户定义的函数 . . . . .	426
LSPatternMatch 用户定义的函数 - 示例 . . . . .	427
LSPrositePattern 用户定义的函数 . . . . .	428
LSPrositePattern 用户定义的函数 - 示例 . . . . .	429
正则表达式支持 . . . . .	429
GeneWise 用户定义的函数 . . . . .	430
链接到 GeneWise . . . . .	430
LSGeneWise 用户定义的函数 . . . . .	430
LSGeneWise 用户定义的函数 - 示例 . . . . .	431
图谱用户定义的函数 . . . . .	432
LSBarcode 用户定义的函数 . . . . .	432
LSBarcode 用户定义的函数 - 示例 . . . . .	433
LSMultiMatch 用户定义的函数 . . . . .	434
LSMultiMatch 用户定义的函数 - 示例 . . . . .	434

LSMultiMatch3 用户定义的函数 . . . . .	435
LSMultiMatch3 用户定义的函数 - 示例 . . . . .	435
逆向用户定义的函数 . . . . .	437
LSRevComp 用户定义的函数 . . . . .	437
LSRevComp 用户定义的函数 - 示例 . . . . .	437
LSRevNuc 用户定义的函数 . . . . .	438
LSRevNuc 用户定义的函数 - 示例 . . . . .	439
LSRevPep 用户定义的函数 . . . . .	439
LSRevPep 用户定义的函数 - 示例 . . . . .	440
转换 . . . . .	440
LSNuc2Pep 用户定义的函数 . . . . .	440
LSNuc2Pep 用户定义的函数 - 示例 . . . . .	441
LSTransAllFrames 用户定义的函数 . . . . .	442
LSTransAllFrames 用户定义的函数 - 示例 . . . . .	443
基码频率表格式 . . . . .	444
基码频率表 - 示例 . . . . .	445
转换表格式 . . . . .	445
转换表 - 示例 . . . . .	446

### 第 26 章 KEGG 用户定义的函数 . . . . . 447

KEGG 用户定义的函数 - 概述 . . . . .	447
按函数类别排列的 KEGG 用户定义的函数 . . . . .	447
KEGG 用户定义的函数的函数自变量 . . . . .	449
注册 KEGG 用户定义的函数 . . . . .	450
途径数据库函数 . . . . .	451
GenesByPathwyS 用户定义的函数 . . . . .	451
GenesByPathwyT 用户定义的函数 . . . . .	452
CompoundsByPathwyS 用户定义的函数 . . . . .	453
CompoundsByPathwyT 用户定义的函数 . . . . .	453
EnzymesByPathwyS 用户定义的函数 . . . . .	454
EnzymesByPathwyT 用户定义的函数 . . . . .	455
PathwysByGenesS 用户定义的函数 . . . . .	455
PathwysByGenesT 用户定义的函数 . . . . .	456
PathwysByCompndsS 用户定义的函数 . . . . .	457
PathwysByCompndsT 用户定义的函数 . . . . .	457
PathwysByEnzymesS 用户定义的函数 . . . . .	458
PathwysByEnzymesT 用户定义的函数 . . . . .	459
序列相似性数据库函数 . . . . .	459
从 SSDB 数据库查询返回的列 (表函数) . . . . .	459
AllNbrsByGeneS 用户定义的函数 . . . . .	460
AllNbrsByGeneT 用户定义的函数 . . . . .	461
BstBstNbrsByGeneS 用户定义的函数 . . . . .	462
BstBstNbrsByGeneT 用户定义的函数 . . . . .	463
BestNbrsByGeneS 用户定义的函数 . . . . .	464
BestNbrsByGeneT 用户定义的函数 . . . . .	465
RevBestNbrsByGeneS 用户定义的函数 . . . . .	466
RevBestNbrsByGeneT 用户定义的函数 . . . . .	467
ParalogsByGeneS 用户定义的函数 . . . . .	468
ParalogsByGeneT 用户定义的函数 . . . . .	469
BestHmlgsByGenesS 用户定义的函数 . . . . .	469
BestHmlgsByGenesT 用户定义的函数 . . . . .	470
BstBstHmlgByGenesS 用户定义的函数 . . . . .	471
BstBstHmlgByGenesT 用户定义的函数 . . . . .	471
ScoreBetweenGenesS 用户定义的函数 . . . . .	472
DefinitionsByGeneS 用户定义的函数 . . . . .	473
GenesByMotifsT 用户定义的函数 . . . . .	473

禁用 KEGG 用户定义的函数 . . . . .	474
---------------------------	-----

## 第 5 部分 参考 . . . . . 477

### 第 27 章 非关系数据源支持的数据类型 479

非关系数据源支持的数据类型 . . . . .	479
BioRS 包装器支持的数据类型 . . . . .	479
BLAST 包装器支持的数据类型 . . . . .	479
Documentum 包装器支持的数据类型 . . . . .	480
Entrez 包装器支持的数据类型 . . . . .	480
Excel 包装器支持的数据类型 . . . . .	480
Extended Search 包装器支持的数据类型 . . . . .	481
HMMER 包装器支持的数据类型 . . . . .	481
表结构文件包装器支持的数据类型 . . . . .	481
Web 服务包装器支持的数据类型 . . . . .	482
WebSphere Business Integration 包装器支持的数 据类型 . . . . .	482
XML 包装器支持的数据类型 . . . . .	483

### 第 28 章 改变昵称 . . . . . 485

改变昵称 . . . . .	485
对改变昵称的限制 . . . . .	486
改变昵称列名 . . . . .	488
改变昵称选项 . . . . .	489
改变昵称列选项 . . . . .	490
改变数据源对象的本地类型 . . . . .	491
为数据源对象改变本地类型 - 示例 . . . . .	492

### 第 29 章 DDL 命令参考 . . . . . 495

BioRS DDL 参考信息 . . . . .	495
CREATE SERVER 语句选项 - BioRS 包装器	495
CREATE USER MAPPING 语句选项 - BioRS 包装器 . . . . .	496
CREATE NICKNAME 语句语法 - BioRS 包装 器 . . . . .	496
BLAST DDL 参考信息 . . . . .	498
CREATE SERVER 语句自变量 - BLAST 包装 器 . . . . .	498
CREATE NICKNAME 语句语法 - BLAST 包装 器 . . . . .	498
Documentum DDL 参考信息 . . . . .	500
CREATE SERVER 语句自变量和选项 - Documentum 包装器 . . . . .	500
CREATE USER MAPPING 语句选项 - Documentum 包装器 . . . . .	500
CREATE NICKNAME 语句语法 - Documentum 包装器 . . . . .	501
Excel DDL 参考信息 . . . . .	503
CREATE SERVER 语句自变量 - Excel 包装器	503
CREATE NICKNAME 语句语法 - Excel 包装 器 . . . . .	504
Extended Search DDL 参考信息 . . . . .	505
CREATE WRAPPER 语句语法 - Extended Search 包装器 . . . . .	505
CREATE SERVER 语句语法 - Extended Search 包装器 . . . . .	506

CREATE USER MAPPING 语句语法 - Extended Search 包装器 . . . . .	507
CREATE NICKNAME 语句语法 - Extended Search 包装器 . . . . .	507
CREATE FUNCTION 语句语法 - Extended Search 包装器 . . . . .	510
Entrez DDL 参考信息 . . . . .	510
CREATE SERVER 语句自变量 - Entrez 包装器	510
CREATE NICKNAME 语句选项 - Entrez 包装 器 . . . . .	511
表结构文件 DDL 参考信息 . . . . .	511
CREATE NICKNAME 语句语法 - 表结构文件 包装器 . . . . .	511
XML DDL 参考信息 . . . . .	515
CREATE NICKNAME 语句语法 - XML 包装器	515

### | 附录 A. 全局目录表中包含联合信息的视 | 图 . . . . . 519

### 附录 B. 联合系统的包装器选项 . . . . . 521

### 附录 C. SQL 语句中的有效服务器类型 523

BioRS 包装器 . . . . .	523
BLAST 包装器 . . . . .	523
CTLIB 包装器 . . . . .	523
Documentum 包装器 . . . . .	524
DRDA 包装器 . . . . .	524
Entrez 包装器 . . . . .	525
Excel 包装器 . . . . .	525
Extended Search 包装器 . . . . .	525
HMMER 包装器 . . . . .	525
Informix 包装器 . . . . .	525
MSSQLODBC3 包装器 . . . . .	526
NET8 包装器 . . . . .	526
ODBC 包装器 . . . . .	526
OLE DB 包装器 . . . . .	526
表结构文件包装器 . . . . .	526
Teradata 包装器 . . . . .	526
Web 服务包装器 . . . . .	527
WebSphere Business Integration 包装器 . . . . .	527
XML 包装器 . . . . .	527

### | 附录 D. 联合系统的服务器选项 . . . . . 529

### 附录 E. 联合系统的用户映射选项 . . . . . 541

### | 附录 F. 联合系统的昵称选项 . . . . . 543

### | 附录 G. 联合系统的昵称列选项 . . . . . 551

### 附录 H. 缺省正向数据类型映射 . . . . . 557

DB2 z/OS 和 OS/390 版数据源 . . . . .	557
DB2 iSeries 版数据源 . . . . .	558
DB2 服务器 VM 和 VSE 版数据源 . . . . .	559
DB2 Linux 版、UNIX 版和 Windows 版数据源 . . . . .	560
Informix 数据源 . . . . .	561

Microsoft SQL Server 数据源 . . . . .	563	关于 z/OS 上的联合功能的文档 . . . . .	590
ODBC 数据源 . . . . .	566	关于 Linux、UNIX 和 Windows 上的联合功能的文档 . . . . .	590
Oracle NET8 数据源 . . . . .	567	关于 Linux、UNIX 和 Windows 上的企业搜索功能的文档 . . . . .	591
Sybase 数据源 . . . . .	568	发行说明和安装需求 . . . . .	591
Teradata 数据源 . . . . .	569		
<b>附录 I. 缺省反向数据类型映射 . . . . .</b>	<b>573</b>	<b>辅助功能 . . . . .</b>	<b>593</b>
DB2 z/OS 和 OS/390 版数据源 . . . . .	573	键盘输入和导航 . . . . .	593
DB2 iSeries 版数据源 . . . . .	575	键盘输入 . . . . .	593
DB2 VM 和 VSE 版数据源 . . . . .	576	键盘导航 . . . . .	593
DB2 Linux 版、UNIX 版和 Windows 版数据源 . . . . .	577	键盘焦点 . . . . .	593
Informix 数据源 . . . . .	578	界面显示的辅助功能 . . . . .	593
Microsoft SQL Server 数据源 . . . . .	579	字体设置 . . . . .	593
Oracle NET8 数据源 . . . . .	580	不依赖于颜色 . . . . .	593
Sybase 数据源 . . . . .	581	与辅助技术的兼容性 . . . . .	594
Teradata 数据源 . . . . .	582	文档的辅助功能 . . . . .	594
<b>附录 J. 联合系统的函数映射选项 . . . . .</b>	<b>583</b>	<b>声明 . . . . .</b>	<b>595</b>
<b>DB2 Information Integrator 文档 . . . . .</b>	<b>585</b>	商标 . . . . .	596
访问 DB2 Information Integrator 文档 . . . . .	585	<b>索引 . . . . .</b>	<b>599</b>
关于 z/OS 上的复制功能的文档 . . . . .	587	<b>与 IBM 联系 . . . . .</b>	<b>609</b>
关于 z/OS 版 DB2 通用数据库的事件发布功能的文档 . . . . .	588	产品信息 . . . . .	609
关于 z/OS 上的 IMS 和 VSAM 的事件发布功能的文档 . . . . .	588	对文档的意见 . . . . .	609
关于 Linux、UNIX 和 Windows 上的事件发布功能和复制功能的文档 . . . . .	589		



---

## 关于本书

本书描述如何配置联合系统以存取数据源。本书包含：

- 联合系统概念、组件和功能简介。
- 有关规划数据源的配置的建议。
- 有关注册联合服务器和数据库存取数据源所必需的对象指示信息。
- 特定于每种数据源的大量参考信息包括：服务器设置信息、环境变量需求、数据类型映射以及为了定制和调整数据源配置而可以设置的选项。

### 查找技术更改

对文本的技术更改是在更改位置的左边用竖线指示的。

---

## 谁应阅读本书

本书针对的是需要设置和配置联合系统的系统管理员、数据库管理员、安全性管理员和系统操作员。使用本书来管理联合系统以及从关系和非关系数据源存取数据。本书还可供需要了解联合系统的配置和使用的程序员和其他用户使用。本书假定您熟悉 DB2。您应该熟悉标准的数据库术语，并且具有数据库设计和数据库管理方面的经验。本书假定您熟悉自己的应用程序和您想要存取的数据源。

---

## 本书中使用的约定和术语

### 联合术语：

本书中的词汇表定义了讨论联合系统时所使用的术语。

### DB2 命令：

本书假定 DB2 命令是在 DB2 命令行处理器 (CLP) 或 DB2 命令中心 GUI 中输入的，除非另有指定。当在文本中提到 DB2 命令时，只列示该命令及其参数。命令将不由 DB2 处理。

### DB2 控制中心：

该文档指示何时可以通过使用 DB2 控制中心来执行任务，并且本文档中还包括执行这些任务的步骤。

### 突出显示约定：

本书使用以下突出显示约定：

**粗体** 指示命令和图形用户界面控件（例如，字段的名称、按钮的名称和菜单选项）。粗体字用来指定注释、限制、先决条件和建议。

**等宽体** 指示您输入的文本、文件名以及代码示例。等宽字体也用于 SQL 语句或 DB2 命令参数名。

**斜体** 指示可以替换为适当值的 SQL 语句或 DB2 命令参数值。SQL 语句或 DB2 命令示例使用斜体来显示样本参数值。斜体用来强调字、标识新的术语以及指示文档标题。

**大写** 指示 DB2 命令和 SQL 语句的名称以及它们的关键字。大写还用于数据类型名称、选项和字首组合词。

---

## 如何阅读语法图

本书从头到尾，语法都是使用如下定义的结构描述的：

阅读语法图时应从左到右，从上到下，遵循线条所指示的路径。

▶▶—— 符号指示语句的开始。

——▶ 符号指示语句语法将在下一行继续。

▶—— 符号指示语句是上一行的继续。

——▶▶ 符号指示语句结束。

必需的项出现在水平线上（主路径）。

▶▶—STATEMENT—*required item*—▶▶

可选的项出现在主路径下面。

▶▶—STATEMENT—  
                  └*optional item*┘—▶▶

如果可选的项出现在主路径上面，则表示该项对语句的执行没有影响，使用它仅仅是为了可读性。

▶▶—STATEMENT—  
                  ┌*optional item*┐—▶▶

如果可从两项或多项中选择，它们会出现在堆栈中。

如果必须选择其中一项，则堆栈的其中一项会出现在主路径上。

▶▶—STATEMENT—  
                  ┌*required choice1*┐  
                  └*required choice2*┘—▶▶

如果这些项都不是选项，则整个堆栈出现在主路径下面。

▶▶—STATEMENT—  
                  ┌*optional choice1*┐  
                  └*optional choice2*┘—▶▶

如果其中一项是缺省项，则它将出现在主路径上面，而其余选项将显示在主路径下面。

▶▶—STATEMENT—  
                  ┌*default choice*┐  
                  ┌*optional choice*┐  
                  └*optional choice*┘—▶▶

主线上返回至左边的箭头指示某项可重复。在这种情况下重复的项必须用一个或多个空格隔开。



如果重复箭头包含逗号，则必须用逗号将重复的项隔开。



堆栈上面的重复箭头指示可从堆栈项中选择多项或重复单个选项。

关键字以大写形式出现（例如，FROM）。它们必须完全按显示的形式来拼写。变量以小写形式出现（例如，column-name）。它们在语法中表示用户提供的名称或值。

如果显示标点符号、圆括号、算术运算符或其它这样的符号，则必须将其作为语法的一部分输入。

有时单个变量表示几个参数的集合。例如，在下图中，变量 `parameter-block` 可替换为标题为 **parameter-block** 的任何图例：



**parameter-block:**



出现在“大项目符号”（●）之间的相邻段可以任何顺序指定。



上图显示可以按任意顺序指定 `item2` 和 `item3`。下面两种形式都有效：

```
STATEMENT item1 item2 item3 item4  
STATEMENT item1 item3 item2 item4
```





---

# 第 1 部分 概念和规划



# 第 1 章 联合系统概述

下列各节提供联合系统概述。

## 联合系统

DB2® 联合系统是一个特殊类型的分布式数据库管理系统 (DBMS)。联合系统由一个作为联合服务器运作的 DB2 实例、一个充当联合数据库的数据库、一个或多个数据源以及存取数据库和数据源的客户机 (用户和应用程序) 组成。借助于联合系统, 您在一个 SQL 语句中就可以将分布式请求发送至多个数据源。例如, 您可以在单个 SQL 语句中将 DB2 Universal Database™ (DB2 通用数据库) 表、Oracle 表和 XML 标记文件中的数据连接起来。下图显示联合系统的各个组件和您可以存取的数据源样本。

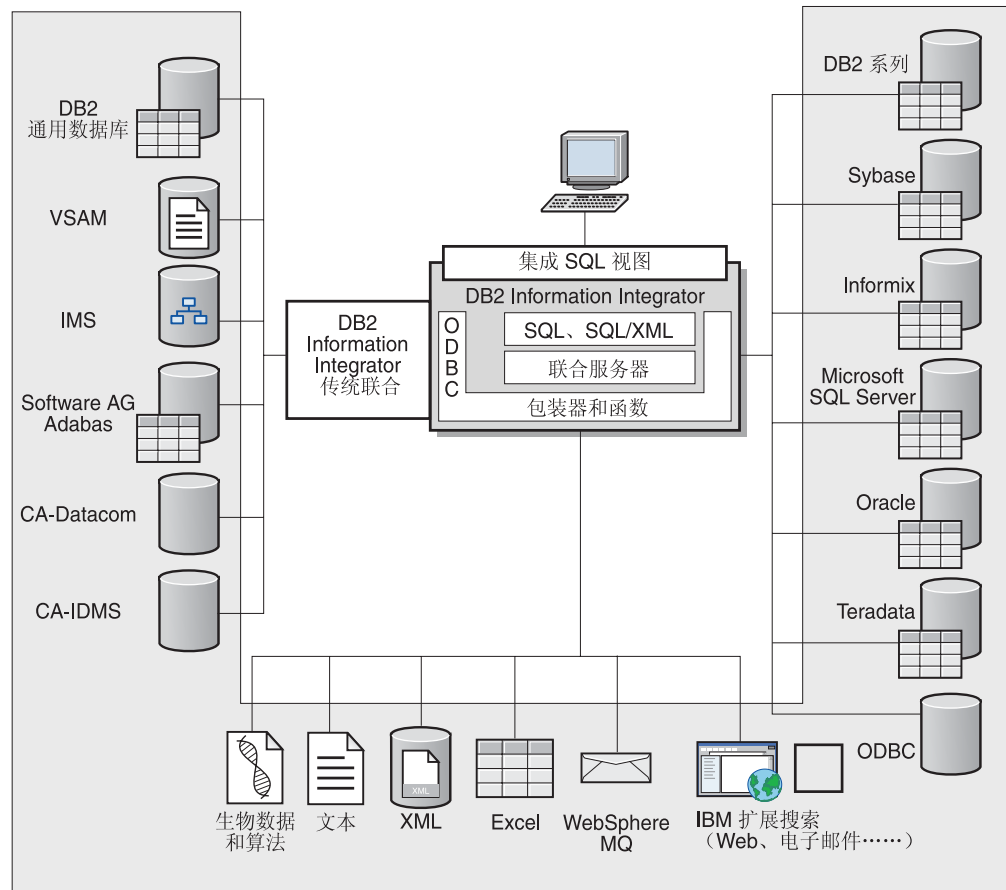


图 1. 联合系统的组件

DB2 联合系统的强大功能表现在它能够:

- 连接本地表中与远程数据源中的数据, 就好象所有数据都本地存储在联合数据库中一样
- 更新关系数据源中的数据, 就好象数据存储在联合数据库中一样
- 将数据复制到关系数据源或从关系数据源复制数据

- 利用数据源处理能力，方法是将请求发送到数据源以便处理
- 通过在联合服务器上处理分布式请求的各个部分来补偿数据源的 SQL 限制

---

## 联合服务器

联合系统中的 DB2<sup>®</sup> 服务器称为**联合服务器**。可以配置任何数目的 DB2 实例来作为联合服务器工作。您可以使用现有 DB2 实例作为联合服务器，也可以专门为联合系统创建新的服务器。

管理联合系统的 DB2 实例称为**服务器**，原因是它响应来自最终用户和客户机应用程序的请求。联合服务器经常将它接收到的请求的各个部分发送至数据源以进行处理。下推操作是以远程方式处理的操作。管理联合系统的 DB2 实例称为**联合服务器**（即使在将请求下推至数据源时充当客户机）。

与任何其它应用程序服务器相似，联合服务器是一个数据库管理器实例。应用程序进程连接请求并将请求提交给联合服务器中的数据库。但是，以下两个主要功能将它与其它应用程序服务器区别开来：

- 配置联合服务器来接收可能是部分或全部针对数据源的请求。联合服务器将这些请求分发给数据源。
- 与其它应用程序服务器相似，联合服务器使用 DRDA<sup>®</sup> 通信协议（基于 TCP/IP）来与 DB2 系列实例进行通信。但是，与其它应用程序服务器不同，联合服务器使用数据源的本机客户机来存取数据源。例如，联合服务器使用 Sybase Open Client 来存取 Sybase 数据源，而使用 Microsoft SQL Server ODBC 驱动程序来存取 Microsoft<sup>®</sup> SQL Server 数据源。

### 相关概念:

- 第 4 页的『什么是数据源?』

---

## 什么是数据源?

在联合系统中，**数据源**可以是关系 DBMS 实例（例如，Oracle 或 Sybase），也可以是非关系数据源（例如，BLAST 搜索算法或 XML 标记文件）。通过某些数据源，您可以存取其它数据源。例如，通过 Extended Search 数据源，可以存取一些数据源（例如，Lotus<sup>®</sup> Notes 数据库、Microsoft<sup>®</sup> Access、Microsoft Index Server、Web 搜索引擎和轻量级目录访问协议（LDAP）目录）。

用来存取数据源的方法或协议取决于数据源的类型。例如，DRDA<sup>®</sup> 用来存取 DB2<sup>®</sup> z/OS<sup>™</sup> 和 OS/390<sup>®</sup> 版数据源，而 Documentum Client API/Library 用来存取 Documentum 数据源。

数据源是半自控的。例如，在 Oracle 应用程序可以存取 Oracle 数据源的同时，联合服务器可以将查询发送至这些数据源。DB2 联合系统不会独占或限制对其它数据源的存取（完整性约束和锁定约束除外）。

### 相关概念:

- 第 7 页的『联合数据库』

### 相关参考:

- 第 5 页的『受支持的数据源』

## 受支持的数据源

可以使用联合系统来存取许多数据源。下表列示了受支持的数据源:

表 1. 受支持的数据源版本和存取方法

数据源	受支持的版本	存取方法
DB2 Universal Database™ (DB2 通用数据库) Linux 版、UNIX 版和 Windows® 版	7.2、8.1 和 8.2	DRDA®
DB2 通用数据库 z/OS™ 和 OS/390® 版	应用了下列 APAR 的 6.1 和 7.1: <ul style="list-style-type: none"> <li>• PQ62695</li> <li>• PQ55393</li> <li>• PQ56616</li> <li>• PQ54605</li> <li>• PQ46183</li> <li>• PQ62139</li> </ul> 8.1	DRDA
DB2 通用数据库 iSeries™ 版	5.1 <ul style="list-style-type: none"> <li>• 应用了以下 APAR:               <ul style="list-style-type: none"> <li>– SE06003</li> <li>– SE06872</li> <li>– II13348</li> </ul> </li> <li>• 应用了以下 PTF:               <ul style="list-style-type: none"> <li>– SI05990</li> <li>SI05991</li> </ul> </li> </ul> 应用了 PTF SI0735 的 5.2。	DRDA
DB2 服务器 VM 和 VSE 版	应用了模式函数 APAR 的修订的 7.1 (或更新版本)	DRDA
Informix™	7.31、8.32、8.4、9.3 和 9.4	Informix Client SDK V2.7 (或更新版本)
ODBC	3.x	用于数据源的 ODBC 驱动程序 (例如, 访问 Redbrick 的 Redbrick ODBC 驱动程序)。
OLE DB	2.7 和 2.8	OLE DB 2.0 (或更新版本)
Oracle	8.0.6、8.1.6、8.1.7、9.0、9.1、9.2、9i 和 10g	Oracle net 客户机或 NET8 客户机软件

表 1. 受支持的数据源版本和存取方法 (续)

数据源	受支持的版本	存取方法
Microsoft SQL Server	7.0、2000 SP3 以及该发行版上更新的 service pack	在 Windows 上, Microsoft SQL Server Client ODBC 3.0 (或更新版本) 驱动程序。  在 UNIX 上, DataDirect Technologies (以前称为 MERANT) Connect ODBC 3.7 (或更新版本) 驱动程序。
Sybase	11.9.2 和 12.x	Sybase Open Client ctlib 接口
Teradata	V2R3、V2R4 和 V2R5	Teradata Call-Level Interface V2 (CLIV2) R04.06 (或更新版本)
BLAST	支持 2.2.3 和更新的 2.2 修订包	BLAST 守护进程 (随包装器一起提供)
BioRS	v5.0.14	无
Documentum	3.x 和 4.x	Documentum Client 库 / APL3.1.7a (或更新版本)
Entrez (PubMed 和 GenBank 数据源)	1.0	无
HMMER	2.2g 和 2.3	HMMER 守护进程 (随包装器一起提供)
IBM Lotus Extended Search	4.0.1 和 4.0.2	Extended Search Client Library (随包装器一起提供)
Microsoft Excel	97、2000、2002 和 2003	安装在联合服务器上的 Excel 97、2000、2002 或 2003
PeopleSoft	8.x	IBM WebSphere Business Integration Adapter for PeopleSoft v2.3.1, 2.4
SAP	3.x 和 4.x	IBM WebSphere Business Integration Adapter for mySAP.com v2.3.1, 2.4
Siebel	7、7.5 和 2000	IBM WebSphere Business Integration Adapter for Siebel eBusiness Applications v2.3.1, 2.4
表结构文件		无
KEGG 的用户定义的函数	受支持	
Life Sciences 的用户定义的函数	受支持	
Web 服务	SOAP 1.0 和 1.1 以及 WSDL 1.0 和 1.1 规范	HTTP
XML	1.0 规范	无

#### 相关概念:

- 第 4 页的『什么是数据源?』

---

## 联合数据库

对于最终用户和客户机应用程序，数据源在 DB2® 中作为单个集合数据库出现。用户和应用程序与联合服务器管理的联合数据库进行交互。联合数据库包含系统目录。联合数据库系统目录包含标识数据源及其特征的条目。联合服务器参考存储在联合数据库系统目录和数据源包装器中的信息来确定处理 SQL 语句的最佳方案。

联合系统处理 SQL 语句时就好像数据源是联合数据库中的普通关系表或视图一样。因此：

- 联合系统可以将关系数据与非关系格式的数据连接起来。即使数据源使用不同的 SQL 语言或根本不支持 SQL，也可如此进行操作。
- 当联合数据库的特征与数据源的特征之间存在差别时，联合数据库的特征优先：
  - 假定联合服务器使用的代码页不同于数据源使用的代码页。当数据源的字符数据返回给联合用户时，该数据是根据联合数据库使用的代码页转换的。
  - 假定联合服务器使用的整理顺序与数据源使用的整理顺序不同。对字符数据的任何排序操作都是在联合服务器上而不是数据源上执行的。

### 相关概念：

- 第 8 页的『SQL 编译器』
- 第 7 页的『联合数据库系统目录』

---

## 联合数据库系统目录

联合数据库系统目录包含有关联合数据库中对象的信息和有关数据源上对象的信息。联合数据库中的目录称为全局目录，原因是它包含有关整个联合系统的信息。DB2® 查询优化器使用全局目录和数据源包装器中的信息来规划处理 SQL 语句的最佳方法。存储在全局目录中的信息包括远程和本地信息（例如，列名、列数据类型、列缺省值和索引信息）。

远程目录信息是数据源使用的信息或名称。本地目录信息是联合数据库使用的信息或名称。例如，假定远程表包括名为 *EMPNO* 的列。全局目录将把远程列名存储为 *EMPNO*。除非您指定另一个名称，否则本地列名将存储为 *EMPNO*。您可以将本地列名更改为 *Employee\_Number*。提交包括此列的查询的用户将在他们的查询中使用 *Employee\_Number* 而不是 *EMPNO*。使用 ALTER NICKNAME 语句来更改数据源列的局部名。

对于关系数据源，存储在全局目录中的信息既包括远程信息又包括本地信息。

对于非关系数据源，存储在全局目录中的信息对于各数据源来说是不同的。

要查看存储在全局目录中的数据源表信息，在联合数据库中查询 SYSCAT.TABLES、SYSCAT.TABOPTIONS、SYSCAT.INDEXES、SYSCAT.COLUMNS 和 SYSCAT.COLOPTIONS 目录视图。

全局目录还包括有关数据源的其他信息。例如，全局目录包括联合服务器用来连接至数据源并将联合用户权限映射至数据源用户权限的信息。全局目录包含有关您显式设置的数据源的属性（例如，服务器选项）。

### 相关概念：

- 第 8 页的『SQL 编译器』

**相关参考:**

- 第 519 页的附录 A, 『全局目录表中包含联合信息的视图』

---

## SQL 编译器

要从数据源获取数据, 用户和应用程序使用 DB2<sup>®</sup> SQL 将查询提交给联合数据库。当提交查询时, DB2 SQL 编译器参考全局目录和数据源包装器中的信息来帮助它处理查询。这包括有关连接至数据源、服务器属性、映射、索引信息和处理统计信息的信息。

**相关概念:**

- 第 10 页的『包装器和包装器模块』
- 第 8 页的『查询优化器』

---

## 查询优化器

作为 SQL 编译器进程的一部分, 查询优化器将分析查询。编译器开发了一些备用策略 (称为存取方案) 来处理查询。存取方案可能要求查询:

- 由数据源处理
- 由联合服务器处理
- 部分由数据源处理, 部分由联合服务器处理

DB2<sup>®</sup> UDB 主要根据有关数据源功能和数据的信息来评估存取方案。包装器和全局目录都包含此信息。DB2 UDB 将查询分解成称为查询片段的段。通常, 如果数据源可以处理段, 则将查询段下推至数据源会更有效率。但是, 查询优化器要考虑其它一些因素, 例如:

- 需要处理的数据量
- 数据源的处理速度
- 段将返回的数据量
- 通信带宽
- 联合服务器上是否有可用的具体查询表表示相同的查询结果

查询优化器根据资源成本生成本地和远程存取方案以处理查询段。然后, DB2 UDB 选择它认为将用最小资源成本处理查询的方案。

如果有任何片段要由数据源处理, 则 DB2 UDB 将这些片段提交给数据源。在数据源处理这些片段之后, 将检索结果并将这些结果返回到 DB2 UDB。如果 DB2 UDB 执行了任何处理部分, 则它会将其结果与从数据源检索到的结果组合起来。然后, DB2 UDB 将所有结果返回到客户机。

**相关概念:**

- 第 8 页的『SQL 编译器』
- 第 9 页的『补偿』
- 『调整查询处理』(《联合系统指南》)



---

## 补偿

如果数据源不能处理查询段或如果联合服务器处理查询段的速度可比数据源处理查询段的速度更快，则 DB2® 联合服务器不下推该查询段。例如，假定数据源的 SQL 语言不支持 GROUP BY 子句中的 CUBE 分组。包含 CUBE 分组并引用该数据源中的表的查询被提交给联合服务器。DB2 Information Integrator 不会将 CUBE 分组下推至数据源，而是自己处理 CUBE。DB2 Information Integrator 处理不受数据源支持的 SQL 的能力称为补偿。

联合服务器以下列两种方法补偿数据源的功能不足：

- 它可以要求数据源使用与查询中陈述的 DB2 函数等同的一个或多个运算。假定数据源不支持余切 (COT(x)) 函数，但是支持正切 (TAN(x)) 函数。DB2 Information Integrator 可以要求数据源执行计算 (1/TAN(x))，它等同于余切 (COT(x)) 函数。
- 它可将数据集返回到联合服务器并以本地方式执行函数。

对于关系数据源，每类 RDBMS 均支持国际 SQL 标准的一个子集。另外，RDBMS 的一些类型支持超出此标准的 SQL 构造。SQL 语言是某种 RDBMS 类型支持的 SQL 总体内容。如果 SQL 构造是以 DB2 SQL 语言编写的，但不是以关系数据源语言编写的，则联合服务器可以为数据源实现此构造。

DB2 Information Integrator 可以对 SQL 语言中的差异进行补偿。此功能的一个示例是公共表表达式子句。DB2 SQL 包括公共表表达式子句。在此子句中，可以指定名称，fullselect 中的所有 FROM 子句都可以按该名称引用结果集。联合服务器将为数据源处理公共表表达式，即使数据源使用的 SQL 语言不包括公共表表达式也是如此。

借助于补偿，对于针对数据源的查询，联合服务器可以支持整个 DB2 SQL 语言。甚至具有很弱的 SQL 支持和不具有任何 SQL 支持的数据源都可从“补偿”受益。您必须将 DB2 SQL 语言与联合系统配合使用（在传递 (Pass-Through) 会话中除外）。

相关概念：

- 第 9 页的『传递 (Pass-Through) 会话』

---

## 传递 (Pass-Through) 会话

通过使用一种称为传递 (Pass-Through) 的特殊方式，可以将 SQL 语句直接提交给数据源。可以在数据源使用的 SQL 语言中提交 SQL 语句。当想要执行 DB2® SQL/API 不可能进行的操作时，使用传递 (Pass-Through) 会话。例如，使用传递 (Pass-Through) 会话来创建过程、创建索引或以数据源的本机语言执行查询。

当前，支持传递 (Pass-Through) 的数据源支持使用 SQL 的传递。将来，数据源有可能可以支持使用除了 SQL 之外的数据源语言的传递 (Pass-Through)。

类似地，可以使用传递 (Pass-Through) 会话来执行不受 SQL 支持的操作（例如，某些管理任务）。但是，不能使用传递 (Pass-Through) 会话来执行所有管理任务。例如，可以在数据源表上创建或删除表，但是不能启动或停止远程数据库。

可以在传递 (Pass-Through) 会话中同时使用静态 SQL 和动态 SQL。

联合服务器提供下列 SQL 语句来管理传递 (Pass-Through) 会话：

#### SET PASSTHRU

打开传递 (Pass-Through) 会话。当您发出另一个 SET PASSTHRU 语句来启动新的传递 (Pass-Through) 会话时, 当前传递会话被终止。

#### SET PASSTHRU RESET

终止当前的传递 (Pass-Through) 会话。

#### GRANT (服务器特权)

对用户、组和授权标识列表或 PUBLIC 授予启动与特定数据源的传递 (Pass-Through) 会话的权限。

#### REVOKE (服务器特权)

撤销启动传递 (Pass-Through) 会话的权限。

下列限制适用于传递 (Pass-Through) 会话:

- 您必须使用数据源的 SQL 语言命令, 而不能使用 DB2 SQL 语言。因此, 您不查询昵称, 但直接查询数据源对象。
- 当在传递 (Pass-Through) 会话中执行 UPDATE 或 DELETE 操作时, 不能使用 WHERE CURRENT OF CURSOR 条件。
- LOB 在传递 (Pass-Through) 会话中不受支持。

#### 相关概念:

- 第 10 页的『包装器和包装器模块』
- 『直接使用传递 (Pass-Through) 查询数据源』(《联合系统指南》)

---

## 包装器和包装器模块

包装器是这样的机制 - 联合服务器可通过它们来与数据源交互作用。联合服务器使用存储在称为包装器模块的库中的例程来实现包装器。这些例程让联合服务器能够执行诸如连接至数据源并反复从数据源检索数据之类的操作。通常, DB2® 联合实例所有者使用 CREATE WRAPPER 语句来在联合数据库中注册包装器。可以使用 DB2\_FENCED 包装器选项将包装器注册为受防护和可信的。

您为想要存取的每种数据源类型都创建一个包装器。例如, 假定您想要存取三个 DB2 z/OS™ 版数据库表、一个 DB2 iSeries™ 版表、两个 Informix® 表和一个 Informix 视图。您只需创建两个包装器: 一个用于 DB2 数据源对象, 另一个用于 Informix 数据源对象。一旦在联合数据库中注册了这些包装器, 您就可以使用这些包装器来从那些数据源存取其它对象。例如, 可以将 DRDA® 包装器与所有 DB2 系列数据源对象: DB2 Linux 版、UNIX® 版和 Windows® 版、DB2 z/OS 和 OS/390® 版、DB2 iSeries 版以及 DB2 服务器 VM 和 VSE 版。

您使用服务器定义和昵称来标识每个数据源对象的特定事项 (名称和位置等等)。

包装器执行许多任务。这些任务包括:

- 它连接至数据源。包装器使用数据源的标准连接 API。
- 它将查询提交给数据源。
  - 对于支持 SQL 的数据源, 查询是用 SQL 提交的。
  - 对于不支持 SQL 的数据源, 查询被转换成数据源的本机查询语言或转换成一系列数据源 API 调用。

- 它从数据源接收结果集。包装器将数据源标准 API 用于接收结果集。
- 它响应有关数据源缺省数据类型映射的联合服务器查询。包装器包含为数据源对象创建昵称时使用的缺省类型映射。对于关系型包装器，您创建的数据类型映射将覆盖缺省数据类型映射。用户定义的数据类型映射存储在全局目录中。
- 它响应有关数据源缺省函数映射的联合服务器查询。包装器包含联合服务器确定是否将 DB2 函数映射至数据源的函数以及如何映射这些函数所需要的信息。此信息由 SQL 编译器用来确定数据源是否能够执行查询操作。对于关系型包装器，您创建的函数映射将覆盖缺省函数类型映射。用户定义的函数映射存储在全局目录中。

包装器选项用来配置包装器或定义 DB2 Information Integrator 如何使用包装器。

#### 相关任务:

- 『可信的和受防护的方式处理环境』（《IBM DB2 Information Integrator 包装器开发者指南》）
- 第 54 页的『为数据源注册包装器』

#### 相关参考:

- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）
- 第 521 页的附录 B, 『联合系统的包装器选项』

## 缺省包装器名称

对于每个受支持的数据源都存在一些包装器。某些包装器具有缺省包装器名称。当使用缺省名称来创建包装器时，联合服务器自动检取与该包装器相关联的数据源库。

表 2. 每个数据源的缺省包装器名称。

数据源	缺省包装器名称
DB2 Universal Database™ (DB2 通用数据库) Linux 版、UNIX 版和 Windows® 版	DRDA
DB2 通用数据库 z/OS 和 OS/390® 版	DRDA
DB2 通用数据库 iSeries 版	DRDA
DB2 服务器 VM 和 VSE 版	DRDA
Informix	INFORMIX
Microsoft® SQL Server	MSSQLODBC3
ODBC	ODBC
OLE DB	OLEDB
Oracle	NET8
Sybase	CTLIB
Teradata	TERADATA
BLAST	无
BioRS	无
Documentum	无
Entrez	无
Extended Search	无
HMMER	无

表 2. 每个数据源的缺省包装器名称。(续)

数据源	缺省包装器名称
Microsoft Excel	无
表结构文件	无
Web 服务	无
WebSphere Business Integration	无
XML	无

**相关概念:**

- 第 10 页的『包装器和包装器模块』

---

## 服务器定义和服务选项

在为数据源创建包装器之后，联合实例所有者对联合数据库定义数据源。实例所有者提供了标识数据源的名称以及与数据源有关的其它信息。此信息包括：

- 数据源的类型和版本
- 数据源的数据库名称（仅限于 RDBMS）
- 特定于数据源的元数据

例如，DB2<sup>®</sup> 系列数据源可以具有多个数据库。定义必须指定联合服务器可以连接至哪个数据库。相反，Oracle 数据源具有一个数据库，并且联合服务器可以连接至该数据库，而不必知道其名称。数据库名称不包括在 Oracle 数据源的联合服务器定义中。

实例所有者提供给联合服务器的名称和其它信息总称为服务器定义。数据源响应数据的请求，并且它们本身就作为服务器。

CREATE SERVER 和 ALTER SERVER 语句用来创建和修改服务器定义。

服务器定义内的其中一些信息是作为服务器选项存储的。创建服务器定义时，了解可指定的关于服务器的选项很重要。一些服务器选项配置包装器，一些则影响 DB2 Information Integrator 使用包装器的方式。

服务器选项可设置为在与数据源的连续连接持续期间存在或设置为在单个连接的持续期间存在。

**相关任务:**

- 第 55 页的『为数据源注册服务器定义』

**相关参考:**

- 『ALTER SERVER statement』（*SQL Reference, Volume 2*）
- 『CREATE SERVER statement』（*SQL Reference, Volume 2*）
- 第 529 页的附录 D，『联合系统的服务器选项』

---

## 用户映射

当联合服务器需要将请求下推至数据源时，服务器必须首先建立与数据源的连接。

对于大多数数据源，联合服务器通过使用对该数据源有效的用户标识和密码来完成此任务。当连接至数据源需要用户标识和密码时，可以定义联合服务器授权标识和数据源用户标识及密码之间的关联。必须为将使用联合系统来发送分布式请求的每个用户标识创建此关联。此关联称为用户映射。

在某些情况下，如果用于连接联合数据库的用户标识和密码与用于存取远程数据源的用户标识和密码相同，则不需要创建用户映射。

#### 相关任务:

- 第 56 页的『为数据源注册用户映射』

#### 相关参考:

- 『ALTER USER MAPPING statement』 (*SQL Reference, Volume 2*)
- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)
- 第 541 页的附录 E, 『联合系统的用户映射选项』

---

## 昵称和数据源对象

在创建服务器定义和用户映射之后，联合实例所有者创建昵称。昵称是一个标识，用来引用位于您想要存取的数据源上的对象。昵称标识的对象称为数据源对象。

昵称与别名不是同一形式的数据源对象的备用名称。前者是联合服务器用来引用这些对象的指针。昵称通常是使用 CREATE NICKNAME 语句以及特定昵称列选项和昵称选项定义的。

当最终用户或客户机应用程序将分布式请求提交给联合服务器时，请求不需要指定数据源。相反，请求通过数据源对象的昵称引用这些对象。昵称被映射至数据源上的特定对象。这些映射不再需要用数据源名称来限定昵称。对于最终用户或客户机应用程序，数据源对象的位置是透明的。

假定您定义昵称 DEPT 来表示名为 NFX1.PERSON 的 Informix® 数据库表。联合服务器允许使用语句 SELECT \* FROM DEPT。但是，除非 NFX1.PERSON 联合服务器上存在本地表，否则联合服务器不允许使用语句 SELECT \* FROM NFX1.PERSON (传递 (Pass-Through) 会话中除外)。

当为数据源对象创建昵称时，有关该对象的元数据被添加至全局目录。查询优化器使用此元数据以及包装器中的信息来使存取数据源对象变得容易。例如，如果昵称用于具有索引的表，则全局目录包含有关该索引的信息。包装器包含 DB2® 数据类型与数据源数据类型之间的映射。

当前，不能对昵称执行某些 DB2 UDB 实用程序操作。

不能使用 Cross Loader 实用程序来交叉装入昵称。

#### 相关概念:

- 第 15 页的『昵称列选项』

#### 相关任务:

- 第 56 页的『为数据源注册昵称』

#### 相关参考:

- 『ALTER NICKNAME statement』 (SQL Reference, Volume 2)
- 『CREATE NICKNAME statement』 (SQL Reference, Volume 2)
- 第 551 页的附录 G, 『联合系统的昵称列选项』
- 第 543 页的附录 F, 『联合系统的昵称选项』
- 第 14 页的『有效数据源对象』

## 有效数据源对象

昵称标识数据源上您想要存取的对象。下表列示联合系统中您可以为其创建昵称的对象类型。

表 3. 有效数据源对象

数据源	有效对象
DB2 Linux 版、UNIX 版和 Windows 版	昵称、具体查询表、表和视图
DB2 z/OS 和 OS/390 版	表和视图
DB2 iSeries 版	表和视图
DB2 VM 和 VSE 版	表和视图
Informix	表、视图和同义词
Microsoft SQL Server	表和视图
ODBC	表和视图
Oracle	表、视图和同义词
Sybase	表和视图
Teradata	表和视图
BLAST	对 BLAST 搜索算法建立了索引的 FASTA 文件
BioRS	BioRS 数据银行
Documentum	Documentum 文档库中的对象和已注册表
Entrez	Entrez 数据库
Extended Search	数据源 (例如, Lotus Notes 数据库、Microsoft Access、Microsoft Index Server、Web 搜索引擎和 LDAP 目录) 中的文件。
HMMER	可由 HMMER 的 hmmpfam 或 hmmsearch 程序搜索的 HMM 数据库文件 (“分层马尔可夫模型”的库, 例如, PFAM)。
Microsoft Excel	.xls 文件 (只存取工作簿中的第一个表)
表结构文件	符合特定格式的文本文件。
Websphere Business Integration 适配器	Websphere Business Integration 业务对象, 它们映射到 SAP 中的 BAPI、Siebel 中的业务组件以及 PeopleSoft 中的组件接口
Web 服务	Web 服务描述语言文件中的操作
XML 标记文件	XML 文档中的项组

### 相关概念:

- 第 13 页的『昵称和数据源对象』
- 第 15 页的『昵称列选项』

---

## 昵称列选项

您可以向全局目录提供有关使用具有昵称的对象的更多元数据信息。此元数据描述数据源对象的某些列中的值。将此元数据分配给称为昵称列选项的参数。此昵称列选项告诉包装器以不同于平常的方式处理列中的数据。SQL 编译器和查询优化器使用元数据来开发较好的存取数据方案。

昵称列选项还用来将其它信息提供给包装器。例如，对于 XML 数据源，昵称列选项用来告诉包装器当包装器将列从 XML 文档解析出来时要使用的 XPath 表达式。

借助于联合，DB2® 服务器将昵称引用的数据源对象视为好像是本地 DB2 表一样。因此，您可以为其创建昵称的任何数据源对象设置昵称列选项。一些昵称列选项是针对特定数据源类型的，它们只可应用于这些数据源。

假定数据源具有不同于联合数据库整理顺序的整理顺序。联合服务器通常将不对包含数据源的字符数据的任何列进行排序。它将数据返回到联合数据库并以本地方式执行排序。但是，假定该列为字符数据类型（CHAR 或 VARCHAR）且只包含数字字符（“0”、“1”、... 和“9”）。您可以通过将值 'Y' 指定给 NUMERIC\_STRING 昵称列选项来指示此假定。这为 DB2 查询优化器提供在数据源上执行排序的选项。如果排序以远程方式执行，您可以避免将数据移植到联合服务器并以本地方式执行排序的开销。

可以使用 ALTER NICKNAME 语句为关系昵称定义昵称列选项。可以使用 CREATE NICKNAME 和 ALTER NICKNAME 语句为非关系昵称定义昵称列选项。

### 相关概念:

- 第 15 页的『数据类型映射』

### 相关任务:

- 『使用昵称』（《联合系统指南》）

### 相关参考:

- 第 551 页的附录 G, 『联合系统的昵称列选项』

---

## 数据类型映射

数据源的数据类型必须映射至相应的 DB2® 数据类型，以便联合服务器可以从数据源检索数据。以下是缺省数据类型映射的一些示例:

- Oracle 类型 FLOAT 映射至 DB2 类型 DOUBLE
- Oracle 类型 DATE 映射至 DB2 类型 TIMESTAMP
- DB2 z/OS™ 版类型 DATE 映射至 DB2 类型 DATE

对于大多数数据源，缺省类型映射在包装器中。DB2 数据源的缺省类型映射在 DRDA® 包装器中。Informix® 的缺省类型映射在 INFORMIX 包装器中，依此类推。

对于某些非关系数据源，您必须在 CREATE NICKNAME 语句中指定数据类型信息。当创建昵称时，必须为数据源对象中的每一列指定相应的 DB2 Linux 版、UNIX® 版和 Windows® 版数据类型。必须将每一列映射至数据源对象中的特定字段或列。

对于关系数据源，可以覆盖缺省的数据类型映射。例如，缺省情况下，Informix INTEGER 数据类型映射至 DB2 INTEGER 数据类型。可以覆盖缺省的映射并将 Informix 的 INTEGER 数据类型映射至 DB2 DECIMAL(10,0) 数据类型。

应该在创建昵称之前创建新的类型映射或修改缺省的类型映射。否则，在类型映射更改之前创建的昵称不会反映新的映射。

**相关概念:**

- 『联合系统中的数据类型映射』（《联合系统指南》）

---

## 函数映射

要使联合服务器能够识别某一数据源函数，必须将该函数映射到 DB2<sup>®</sup> Linux 版、UNIX<sup>®</sup> 版和 Windows<sup>®</sup> 版中的现有相应函数。DB2 Information Integrator 提供了现有内置数据源函数与内置 DB2 相应函数之间的缺省映射。对于大多数数据源，缺省函数映射在包装器中。至 DB2 z/OS<sup>™</sup> 和 OS/390<sup>®</sup> 版函数的缺省函数映射在 DRDA<sup>®</sup> 包装器中。至 Sybase 函数的缺省函数映射位于 CTLIB 包装器中，等等。

对于关系数据源，可以在您想要使用联合服务器不识别的数据源函数时创建函数映射。您创建的映射是数据源函数与联合数据库的 DB2 相应函数之间的映射。函数映射通常在新的内置函数或新的用户定义的函数在数据源上成为可用时使用。函数映射还在 DB2 相应函数不存在时使用。在此情况下，必须还创建一个函数模板。

**相关概念:**

- 『联合系统中的函数映射』（《联合系统指南》）
- 第 16 页的『索引规范』

---

## 索引规范

当为数据源表创建昵称时，有关该数据源表具有的任何索引的信息被添加至全局目录。查询优化器使用此信息来加快分布式请求的处理。有关数据源索引的目录信息是一组元数据，称为索引规范。当您为下列对象创建昵称时，联合服务器不创建索引规范：

- 不具有索引的表
- 视图，它通常没有任何索引信息存储在远程目录中
- 不具有远程目录（联合服务器可从中获取索引信息）的数据源对象

假定表除了创建昵称时所具有的那些索引以外，它还获得了新的索引。因为创建昵称时对全局目录提供了索引信息，所以联合服务器不会察觉到新的索引。类似地，当为视图创建昵称时，联合服务器也不会察觉到从中生成了视图的基础表（及其索引）。在这些情况下，您可以将必要的索引信息提供给全局目录。您可以为不具有任何索引的表创建索引规范。索引规范告知查询优化器要搜索表中的哪一列或哪些列以快速查找数据。

**相关概念:**

- 『联合系统中的索引规范』（《联合系统指南》）



---

## 如何与联合系统交互作用

因为联合数据库是一个 DB2<sup>®</sup> 通用数据库，所以您可以使用下列任一方法来与联合系统交互作用：

- DB2 命令行处理器 (CLP)
- DB2 命令中心 GUI
- DB2 控制中心 GUI
- 应用程序
- DB2 系列工具
- Web 服务提供程序

联合文档中的步骤提供了可在 DB2 命令行处理器或 DB2 命令中心 GUI 中输入的命令和 SQL 语句。该文档指示何时可以通过 DB2 控制中心 GUI 执行任务。因为 DB2 控制中心 GUI 是直观的，所以通过 DB2 控制中心执行这些任务的步骤不包括在此文档中。

### DB2 命令行处理器 (CLP)

您可以通过 DB2 命令行处理器执行设置、配置、调整和维护联合系统所需的大多数任务。在某些情况下，您必须使用 DB2 命令行处理器或 DB2 命令中心。例如：

- 创建、改变或删除用户定义的数据类型映射
- 创建、改变或删除用户定义的函数映射

### DB2 命令中心

通过 DB2 命令中心，您可以创建和运行分布式请求而不必手工输入冗长的 SQL 语句。当您调整联合系统的性能时，使用 DB2 命令中心。DB2 命令中心是一个使用 DB2 Explain 功能来查看分布式请求存取方案的方便方法。DB2 命令中心还可用来与“SQL 助手”工具配合使用。

### DB2 控制中心

DB2 控制中心 GUI 允许您执行设置、配置和修改联合系统所必需的大多数任务。DB2 控制中心使用面板（对话框和向导）来指导您完成任务。当鼠标指针悬浮在某个控件（例如，列表框或命令按钮）上时，这些面板就会包含交互式帮助。另外，每个面板都具有一个帮助按钮，它提供有关面板任务及指向相关概念和参考信息的链接的信息。

可以使用向导来创建联合对象，或者可以分别创建每个对象。

使用 DB2 控制中心来配置对 Web 服务、WebSphere<sup>®</sup> Business Integration 和 XML 数据源的访问。DB2 控制中心内置的功能部件简化了配置联合服务器以访问这些数据源所需的步骤。

DB2 控制中心 GUI 是执行基本数据源配置任务的最容易的方法：

- 创建包装器并设置包装器选项
- 指定数据源的环境变量
- 创建服务器定义并设置服务器选项

- 创建用户映射并设置用户选项
- 创建昵称并设置昵称选项或列选项

在配置联合服务器以存取数据源之后，可以使用 DB2 控制中心来执行以下任务：

- 修改数据源配置
- 监控昵称和服务器的状态
- 维护昵称的当前统计信息
- 创建并修改高速缓存表
- 指定昵称的信息约束
- 通过 DB2 Information Integrator 使用透明 DDL 创建远程表

## 应用程序

应用程序不需要任何特殊编码就可以处理联合数据。应用程序访问系统的方式就好象任何其它 DB2 客户机应用程序一样。应用程序与联合服务器内的联合数据库进行交互。要从数据源获取数据，应用程序使用 DB2 SQL 将查询提交给联合数据库。然后，DB2 Information Integrator 将各查询分发给适当的数据源、收集所请求的数据并将此数据返回到应用程序。但是，由于 DB2 Information Integrator 通过昵称与数据源进行交互作用，所以您需要了解：

- 使用昵称时具有的 SQL 限制
- 如何对具有昵称的对象执行操作

## DB2 系列工具

您还可以使用诸如以下工具之类的主机和中型机工具来与联合数据库进行交互作用：

- DB2 z/OS™ 和 OS/390® 版上的 DB2 SPUFI
- DB2 iSeries™ 版上的 Interactive SQL (STRSQL)

## Web 服务提供程序

还可以使用 Web 服务包装器通过 Web 服务提供程序来与联合数据库进行交互作用。

相关概念：

- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

相关任务：

- 第 358 页的『将 Web 服务数据源添加至联合服务器』

---

## 第 2 章 计划联合数据源配置

下列各项提供可用来帮助您计划联合系统的信息。

---

### 联合对象命名规则

与其它 DB2 对象一样，为联合数据库对象命名也必须遵循一些规则。

联合数据库对象包括：

- 函数映射
- 索引规范
- 昵称
- 服务器
- 类型映射
- 用户映射
- 包装器

联合对象名必须以下列其中一项开头：

- 一个字母，包括有效的重音字母（例如，Ö）
- 多字节字符，不包括多字节空格（适用于多字节环境）

联合对象名不能以数字或下划线字符开头。

联合对象名还可以包括下列字符：

- A 到 Z
- 0 到 9
- @、#、\$ 和 \_（下划线）

联合对象名不能超过 128 个字节。

选项（例如，服务器选项和昵称选项）和选项设置最多只能有 255 个字节。

未用引号引起来的名称将被转换成大写。

**相关概念：**

- 『NLS 环境中的命名规则』（《管理指南：实现》）
- 『Unicode 环境中的命名规则』（《管理指南：实现》）

**相关参考：**

- 第 27 页的『用于规划联合系统配置的核对表』
- 第 20 页的『保留联合系统中区分大小写的值』

---

## 保留联合系统中区分大小写的值

在联合系统中，有时需要指定一些值（例如，用户标识和密码），这些值在数据源中是区分大小写的。为了确保将这些值传送给数据源时它们的大小写是正确的，应遵循下列准则：

- 按要求的大小写指定这些值，并用正确的引号将它们引起来。对于某些对象名（例如，包装器的名称或昵称），双引号是可选的。对于选项值（例如，`REMOTE_AUTHID` 和 `REMOTE_PASSWORD` 用户映射选项），必须使用单引号。
- 对于用户标识和密码，可以设置 `FOLD_ID` 和 `FOLD_PW` 服务器选项，以便将值自动转换成正确的大小写。设置了此选项，就不必记住每个数据源要求的大小写情况。您可以按任意大小写输入值，然后将自动转换这些值。

### 在 UNIX 操作系统命令提示符处：

在联合服务器操作系统命令提示符处，如果用引号将区分大小写的值引起来了，则必须确保对引号进行解析之后它是正确的：

- 如果 SQL 语句中包含双引号，但是不包含单引号，则将整个语句用单引号引起来。例如，如果想发出以下 SQL 语句：

```
CREATE NICKNAME my_nick FOR my_server."owner"."my_table"
```

则在 UNIX 命令提示符处输入以下文本：

```
DB2 'CREATE NICKNAME my_nick FOR my_server."owner"."my_table"'
```

- 如果 SQL 语句中包含单引号，但是不包含双引号，则将整个语句用双引号引起来。例如，如果想发出以下 SQL 语句：

```
CREATE USER MAPPING FOR USER SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

则在 UNIX 命令提示符处输入以下文本：

```
DB2 "CREATE USER MAPPING FOR USER SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

- 如果 SQL 语句既包含单引号又包含双引号：

- 将整个语句用双引号引起来
- 在需要双引号的值前面添加反斜杠

例如，要发出以下 SQL 语句：

```
CREATE USER MAPPING FOR "local_id" SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

则在 UNIX 命令提示符处输入以下文本：

```
DB2 "CREATE USER MAPPING FOR \"local_id\" SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

以上各个示例都假定您正在从 UNIX 命令提示符处输入 SQL 语句，并将语句传送给 DB2 命令，不带 `-f` 选项。如果使用带有 `-f` 选项的 DB2 命令来从一个文件中输入 SQL 语句，则应该象每个示例第一次出现时显示的那样输入语句。

### 从 Windows 操作系统命令提示符处：

在 Windows 上，要保留区分大小写的值，在每个双引号前面应添加反斜杠。例如，假定想要为 Microsoft SQL Server 表 `weekly_salary` 创建昵称 `NICK1`。该表驻留在 `NORBASE` 数据库中。本地模式为 `my_schema`。

则在联合服务器上的 Windows 命令提示符处应输入：

```
DB2 CREATE NICKNAME nick1
      FOR norbase.\"my_schema\".\"weekly_salary\"
```

从 **DB2 CLP** 或者从应用程序：

当从 DB2 命令行提示符（CLP）处指定值或者在应用程序中指定值时，可以通过将区分大小写的值用正确的引号引起来的方法来保留这些值。

例如，假定您想为用户标识 `local_id` 创建用户映射。远程用户标识为 `my_id`，远程密码为 `my_password`。您想要所有这三个值都保留为小写。则在 DB2 命令提示符处应输入：

```
| CREATE USER MAPPING FOR "local_id" SERVER my_server
|     OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

相关参考：

- 第 529 页的附录 D，『联合系统的服务器选项』
- 第 541 页的附录 E，『联合系统的用户映射选项』
- 第 27 页的『用于规划联合系统配置的核对表』

---

## 更新数据源统计信息

如果您打算存取关系数据源，则在配置联合服务器以存取该数据源之前应该更新远程数据源的统计信息。通过确保远程数据源具有最新的统计信息，可以提高查询性能。

联合服务器依赖存储在联合数据库中的数据源统计信息来优化查询处理。这些统计信息是当您为数据源对象创建昵称时收集的。联合数据库验证数据源中的对象是否存在，然后尝试收集该数据源的现有统计数据。对于查询优化器很有用的信息是从数据源目录中读取的，并且添加至联合数据库中的系统目录。因为查询优化器可能要使用来自数据源的某些或所有目录信息，所以，建议您在创建昵称之前要更新数据源中的统计信息。在数据源中使用与 `DB2 RUNSTATS` 命令等价的命令来更新数据源统计信息。

当为对象创建昵称时，联合数据库将检索数据源对象的统计信息。如果在您创建昵称之后，数据源更新一个对象的目录统计信息，则不会将统计信息方面所作的更改传播给联合数据库中的系统目录。为了确保联合数据库中的系统目录反映远程数据源对象的最新统计信息，您必须请求联合服务器更新统计信息。

**操作：** 标识您想要存取的数据源对象。这些对象是将其创建昵称的对象。确定哪些数据源（这些对象是这些数据源的一部分）允许您更新统计信息。并在规划核对表的数据源统计信息表中列示这些数据源。

相关概念：

- 『昵称统计信息更新设施 - 概述』（《联合系统指南》）

相关参考：

- 第 27 页的『用于规划联合系统配置的核对表』

---

## 选择正确的包装器

对于大多数数据源，只有一个可以用来存取数据源的包装器。但是，对于某些数据源，可以选择使用哪个包装器来存取数据源中的数据。

可以通过使用为支持 ODBC API 的数据源设计的包装器或者通过使用 ODBC 包装器来存取这些数据源。这些数据源的示例包括 Oracle、Microsoft Excel 和 Microsoft SQL Server。通常，当您使用特别为这些数据源设计的包装器时，查询性能会更好。

使用 ODBC 包装器来存取那些具有 ODBC 驱动程序但 DB2 Information Integrator 中没有特定数据源包装器进行支持的任何数据源。例如，使用 ODBC 包装器来存取 RedBrick 数据源。

### DB2 Linux 版、UNIX 版和 Windows 版数据源

不要使用 ODBC 包装器来存取 DB2 通用数据库 Linux 版、UNIX 版和 Windows 版数据源。不支持使用 ODBC 包装器来存取 DB2 通用数据库 Linux 版、UNIX 版和 Windows 版数据源。应使用 DRDA 包装器来存取 DB2 通用数据库 Linux 版、UNIX 版和 Windows 版数据源。

### Excel 数据源

根据需要，可以使用 ODBC 包装器而不是 Excel 包装器来存取 Excel 数据。

### Informix 数据源

不要使用 ODBC 包装器来存取 Informix 数据源。不支持使用 ODBC 包装器来存取 Informix 数据源。要存取 Informix 数据源，使用 Informix 包装器。

**操作：**在规划核对表的包装器表中标识将为联合系统创建的包装器。

#### 相关概念：

- 第 22 页的『用来存取 Excel 数据的方法』

#### 相关任务：

- 第 202 页的『将 Excel 数据源添加至联合服务器』
- 第 283 页的『将 ODBC 数据源添加至联合服务器』

#### 相关参考：

- 第 27 页的『用于规划联合系统配置的核对表』

---

## 用来存取 Excel 数据的方法

可以使用 Excel 包装器或 ODBC 包装器来存取 Microsoft® Excel 工作表中的数据。

要查询 Excel 数据，以上两个包装器都需要 DB2® 联合服务器，该服务器可打开和读取 Excel 工作簿中的工作表。因此，Excel 工作簿必须处于联合服务器所在的那台计算机上或处于可通过网络访问的驱动器上。

如果使用 Excel 包装器，则必须在联合服务器上安装 Excel 应用程序。

如果使用 ODBC 包装器，则联合服务器上必须有 Excel ODBC 驱动程序。此驱动程序是随 Microsoft Windows<sup>®</sup>一起自动安装的。不需要在联合服务器上安装 Excel 应用程序。

每个包装器对 Excel 工作簿中数据的位置和布局都有一些要求。对于 Excel 包装器而言，只可存取工作簿的第一个工作表中的数据。而对于 ODBC 包装器，您可以存取工作簿的任何工作表中的数据。

下列示例显示了对这两个包装器的工作表布局要求。

#### 包含几行标题和一个公式的工作表的示例:

此示例显示了一个工作表，它包含位于工作表顶部的几行标题，一些空白行以及第 13 行中的一个公式。要存取工作表中的数据，必须标识您想要存取的单元格区域。

	A	B	C	D
1	<b>Compound Analysis</b>			
2				
3	<b>Compound Name</b>	<b>Weight</b>	<b>Molecular Count</b>	<b>Tested?</b>
4	compound_A	1.23	367	tested
5	compound_G		210	
6	compound_F	0.000425536	174	tested
7	compound_Y	1.000256		tested
8	compound_Q		1024	
9	compound_B	33.5362		
10	compound_S	0.96723	67	tested
11	compound_O	1.2		tested
12				
13		<b>Total Compounds Tested</b>		<b>5</b>

图 2. 包含几行标题和一个公式的工作表

#### 如果使用 Excel 包装器

您将通过使用 RANGE 选项在 CREATE NICKNAME 语句中指定单元格区域。在您指定的区域中仅包括数据。在该区域中不包括任何列标题。包含公式（例如，SUM）的单元格将返回该公式的结果而不是返回公式。除非您想要返回公式结果，否则不要在该区域中包括那些包含公式的单元格。在此示例中，您在 RANGE 选项中包括的单元格区域是 A4:D11。

#### 如果使用 ODBC 包装器

您必须为单元格区域创建一个名称，以显式指定工作表内数据的位置。Excel 将此单元格区域称作命名区域。Excel ODBC 驱动程序仅识别一行标题，即区域中的第一行。在标题与数据之间不允许有空白行。命名区域必须只包括一行列标题。您将在 CREATE NICKNAME 语句中指定该“命名区域”。必须在您命名

的区域中包括一行列标题。如果在命名区域中不包括一行列标题，则第一行数据被视作列标题。包含公式（例如，SUM）的单元格将返回该公式的结果而不是返回公式。除非您想要返回公式结果，否则不要在该区域中包括那些包含公式的单元格。在此示例中，您指定的单元格区域是 A3:D11。

#### 包含一行标题的工作表的示例:

此示例显示了一个工作表，它只包含工作表顶部的一行列标题。该布局不包括其它带标题的行、空白行或带公式的单元格。

	A	B	C	D
1	Compound Name	Weight	Molecular Count	Tested?
2	compound_A	1.23	367	tested
3	compound_G		210	
4	compound_F	0.000425536	174	tested
5	compound_Y	1.000256		tested
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested
10				
11				

图 3. 第 1 行是列标题的工作表

#### 如果使用 Excel 包装器

您必须通过使用 RANGE 选项在 CREATE NICKNAME 语句中指定单元格区域。该区域不能包括第 1 行中的列标题。您将指定的单元格区域是 A2:D9。

#### 如果使用 ODBC 包装器

您可以存取此数据而无须创建命名区域。您将在 CREATE NICKNAME 语句中指定工作表名称。包装器读取第一个非空行作为标题，并将该列名形式的信息用作昵称。随后的行则作为数据读取。

#### 只包含数据的工作表的示例:

此示例显示了只包含数据的工作表。没有列标题行，没有空白行，也没有带公式的单元格。



	A	B	C	D
1	compound_A	1.23	367	tested
2	compound_G		210	
3	compound_F	0.000425536	174	tested
4	compound_Y	1.000256		tested
5	compound_Q		1024	
6	compound_B	33.5362		
7	compound_S	0.96723	67	tested
8	compound_O	1.2		tested
9				
10				

图 4. 只包含数据的工作表

#### 如果使用 Excel 包装器

如果数据在工作簿的第一个工作表中，则包装器无须使用 RANGE 选项就可以存取数据。如果数据在工作簿的其它工作表中，则您必须在 CREATE NICKNAME 语句中指定 RANGE 选项。

#### 如果使用 ODBC 包装器

当您使用 ODBC 包装器来存取 Excel 数据时，该包装器受到 Excel ODBC 驱动程序支持的内容的限制。Excel ODBC 驱动程序需要一个特定格式来用于工作表。驱动程序假定第一个非空行包含列标题。如果第一个非空行包含数据，则该行中的数据被视作其余数据的列标题。如果工作表不包含列标题行，则第一行将被用作标题而不是数据。事实上，您丢失了第一行数据。可以通过修改工作表来应付这个需求。在数据前面插入一个新行并对每列数据列添加标题，这样，该示例就与包含一行标题的示例类似了。

#### 相关任务:

- 第 202 页的『将 Excel 数据源添加至联合服务器』
- 第 293 页的『使用 ODBC 包装器存取 Excel 数据』

## 规划用户映射

当联合服务器需要将请求下推给数据源时，服务器必须首先与数据源建立连接。对于某些数据源，联合服务器通过使用对该数据源有效的用户标识和密码来建立连接。对于这些数据源，必须定义联合服务器用户标识和密码与数据源用户标识和密码之间的关联。必须为将使用联合系统来发送分布式请求的每个用户标识创建此关联。此关联被称为用户映射。

可以使用 DB2 控制中心来为将用同一用户标识和密码来存取数据源的一组用户创建用户映射。

**操作:** 标识需要具有联合服务器与数据源之间的用户映射的用户标识。列示规划核对表的用户映射表中的联合服务器用户标识和相应的数据源用户标识。

**相关参考:**

- 第 27 页的『用于规划联合系统配置的核对表』

---

## 规划数据类型映射

数据源数据类型被称为远程数据类型，联合数据库数据类型被称为本地数据类型。

对于某些数据源，包装器中包含数据源数据类型与联合数据库数据类型之间的缺省映射。当为数据源对象创建昵称时，有关各列的信息被存储在联合数据库系统目录中。这些列的数据类型来自于缺省正向数据类型映射。

对于其它数据源，必须在创建昵称时指定列信息和数据类型。某些非关系型包装器会创建存取数据源所需的所有列。这些列被称为*固定列*。对于其它非关系数据源，可以指定列的部分或所有数据类型。

您的应用程序可能需要不同于缺省映射的数据类型映射。对于允许您指定数据类型映射的包装器，可以覆盖缺省映射：

- 更改位于特定服务器上的所有数据源对象的类型映射
- 更改特定数据源对象的类型映射
- 更改特定数据源类型的类型映射
- 更改特定数据源类型和版本的类型映射

使用 `CREATE TYPE MAPPING` 语句来定义新的数据类型映射。创建的映射将存储在联合数据库全局目录 `SYSCAT.TYPEMAPPINGS` 视图中。

在为数据源对象创建昵称之前更改数据类型映射。当为数据源对象创建昵称时，联合服务器将为全局目录填充有关表的信息。此信息包括昵称、数据源表名、列名和为每个表列定义的数据类型。

只有在更改映射之后所创建的昵称才会反映新的类型映射。在更改映射之前所创建的昵称将使用缺省数据类型映射。

如果在创建昵称之后创建数据类型映射，则将必须改变每个昵称以反映新映射，或者先删除昵称然后再次创建昵称。

**注:** 如果数据源表中包含具有单值数据类型或用户定义的数据类型的列，则您具有两种选择：

- 可以在为数据源表创建昵称之前在联合数据库中创建类型映射。通过在创建昵称之前创建类型映射，联合服务器就可以知道要将这些列映射至哪种数据类型。如果这些单值数据类型或用户定义的数据类型的映射不是在发出 `CREATE NICKNAME` 语句之前创建的，则将接收到错误。
- 如果数据源表中的列满足下列任一条件：
  - 列的数据类型是基于系统数据类型或内置数据类型的用户定义的数据类型
  - 列具有数据类型映射不支持的属性

可以在这样的数据源中创建视图：这些数据源中的列与基本内置数据类型相关联，或者这些列被强制类型转型为基本内置数据类型。然后，为视图而不是为表创建昵称。

**操作：** 标识想要为其定义新映射的数据类型映射。在规划核对表的数据类型映射表中列示数据源以及想要创建的类型映射。

**相关概念：**

- 『联合系统中的数据类型映射』（《联合系统指南》）

**相关参考：**

- 第 27 页的『用于规划联合系统配置的核对表』
- 第 479 页的『非关系数据源支持的数据类型』

---

## 规划函数映射

DB2 UNIX 版和 Windows 版提供现有内置数据源函数与内置 DB2 函数之间的缺省函数映射。对于大多数数据源，缺省函数映射存在于包装器中。对于一些非关系数据源，不能改变缺省函数映射。

要使用联合服务器不识别的数据源函数，必须创建函数映射。您创建的映射是数据源函数与联合数据库中的相应函数之间的映射。通常，当新的内置函数或新的用户定义的函数在数据源中变得可用时就要使用函数映射。

当 DB2 相应函数不存在时也要使用函数映射。在此情况下，创建函数映射之前将必须在联合数据库中创建函数模板。

**操作：** 确定是否需要为数据源创建函数映射。列示规划核对表的函数映射表中需要的函数映射。

**相关概念：**

- 第 16 页的『函数映射』

---

## 用于规划联合系统配置的核对表

遵循此规划核对表，可以使联合系统配置更容易。此核对表将指导您如何优化联合系统配置。

### 核对表：联合对象命名规则

您熟悉联合对象的命名规则吗？

请参阅本主题末尾的相关链接以找到有关联合对象命名规则的信息。

### 核对表：保留区分大小写的值

您想要设置 FOLD\_ID 和 FOLD\_PW 服务器选项以保留发送给数据源的用户标识和密码的大小写吗？使用下表来标识将对哪些服务器定义应用这些选项。

表 4. 规划核对表: 要为联合系统设置的 FOLD\_ID 和 FOLD\_PW 服务器选项

数据源	在此数据源的服务器定义中您将服务器指定什么名称?	将为 FOLD_ID 服务器选项指定什么设置?	将为 FOLD_PW 服务器选项指定什么设置?

## 核对表: 数据源统计信息

在下表中, 列示了将为联合系统的一部分的数据源。指示在配置联合服务器以存取数据源之前将更新其统计信息的数据源。DB2 UDB Linux 版、UNIX 版和 Windows 版列示在此表中作为示例。

表 5. 规划核对表: 要为联合系统更新的数据源统计信息

数据源	此数据源要维护目录信息吗? (Y/N)	您将更新此数据源的统计信息吗? (Y/N)	数据源用来更新统计信息的实用程序的名
DB2 Linux 版、UNIX 版和 Windows 版	Y	Y	RUNSTATS

## 核对表: 数据类型映射

在下表中, 标识需要为其创建映射的数据源数据类型和相应的联合服务器数据类型。

表 6. 规划核对表: 要为联合系统创建的数据类型映射

数据源	在此数据源的服务器定义中您将服务器指定什么名称?	数据源数据类型	联合服务器数据类型

## 核对表： 用户映射

在下表中，标识将为联合系统的一部分的每个数据源的联合服务器用户标识和相应的用户标识。

表 7. 规划核对表： 要为联合系统创建的用户映射

		数据源	数据源	数据源
用户名	DB2 Linux 版、 UNIX 版和 Windows 版用户标识	用户标识	用户标识	用户标识

## 核对表： 包装器

在下表中，标识将创建的包装器。

表 8. 规划核对表： 要为联合系统创建的包装器

数据源	缺省包装器名称	将为包装器指定的名称
BioRS	无	
BLAST	无	
商业应用程序 (WebSphere Business Integration 包装器)	无	
DB2 Universal Database™ (DB2 通用数据库) Linux 版、UNIX 版和 Windows® 版 DB2 通用数据库 z/OS 和 OS/390® 版 DB2 通用数据库 iSeries 版 DB2 Server VM 和 VSE 版	DRDA	
Documentum	无	
Entrez	无	
Excel	无	
Extended Search	无	
HMMER	无	
Informix	INFORMIX	
Microsoft® SQL Server	MSSQLODBC3	
Oracle	NET8	
ODBC	无	
OLE DB	OLEDB	
Sybase	CTLIB	

表 8. 规划核对表: 要为联合系统创建的包装器 (续)

数据源	缺省包装器名称	将为包装器指定的名称
表结构文件	无	
Teradata	TERADATA	
Web 服务	无	
XML	无	

**相关概念:**

- 第 49 页的『配置数据源的快速方法』
- 第 22 页的『用来存取 Excel 数据的方法』

**相关任务:**

- 第 33 页的『检查联合服务器的设置』
- 第 44 页的『创建联合数据库』

**相关参考:**

- 第 39 页的『联合数据库本地语言注意事项』
- 第 19 页的『联合对象命名规则』
- 第 20 页的『保留联合系统中区分大小写的值』
- 第 21 页的『更新数据源统计信息』
- 第 26 页的『规划数据类型映射』
- 第 27 页的『规划函数映射』
- 第 25 页的『规划用户映射』
- 第 22 页的『选择正确的包装器』

---

## 第 2 部分 联合服务器和数据库





---

## 第 3 章 检查联合服务器的设置

---

### 检查联合服务器的设置

通过检查联合服务器上的关键设置来避免存在潜在的配置问题。

过程:

要检查联合服务器的设置:

- 确认包装器库文件与数据源客户机软件之间的链接编辑 (UNIX)
- 检查 FEDERATED 参数是否设置为 YES

检查联合服务器的设置之后, 必须创建联合数据库。

相关任务:

- 第 33 页的『确认包装器库文件的链接编辑 (UNIX)』
- 第 38 页的『检查 FEDERATED 参数』
- 第 44 页的『创建联合数据库』

---

### 确认包装器库文件的链接编辑 (UNIX)

#### 确认包装器库文件的链接编辑 (UNIX)

确认包装器库文件的链接编辑是检查联合服务器的设置这一大型任务的一部分。

在 UNIX 联合服务器上, 某些包装器必须与数据源的数据源客户机软件进行链接编辑。将在安装 DB2 Information Integrator 时尝试执行链接编辑步骤。链接编辑步骤将为要与联合服务器进行通信的每个数据源创建包装器库。

此任务只适用于下列数据源:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

在配置联合服务器和数据库以存取数据源之前, 应该确认包装器库文件的链接编辑是否已成功。

先决条件:

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件 (例如, 数据源客户机软件)。

过程:

确定包装器库文件的链接编辑的状态:

- 如果链接编辑已成功, 则包装器库文件将出现在 DB2 Information Integrator 的安装目录中。
- 如果链接编辑失败, 则在 DB2 Information Integrator 的安装目录中检查错误消息文件。
- 如果未执行链接编辑, 则库文件和消息文件都不会出现在 DB2 Information Integrator 的安装目录中。您将必须手工运行链接脚本。

**相关任务:**

- 第 34 页的『检查包装器库文件 (UNIX)』
- 第 35 页的『检查链接编辑消息文件 (UNIX)』
- 第 36 页的『将包装器库手工链接至数据源客户机软件』
- 第 50 页的『使用 DB2 UDB 控制中心将数据源添加至联合服务器』

## 检查包装器库文件 (UNIX)

检查包装器库文件是确认包装器库与数据源客户机软件之间的链接编辑这一大型任务的一部分。

必须具有包装器库文件您才能存取数据源。对于某些数据源, 库文件是当您安装 DB2 Information Integrator 时添加至联合服务器的。对于其它数据源, 必须运行链接编辑脚本才能创建库文件。

此任务只适用于下列数据源:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

**过程:**

要检查联合服务器上是否存在包装器库文件:

1. 检查库文件是否在包装器库的目录路径中。必须确认对于想要存取的每个数据源, 联合服务器上是否存在库文件。
2. 如果库文件不在该目录中, 则必须将包装器库手工链接至数据源客户机软件。

**相关任务:**

- 第 33 页的『检查联合服务器的设置』
- 第 35 页的『检查链接编辑消息文件 (UNIX)』
- 第 36 页的『将包装器库手工链接至数据源客户机软件』

**相关参考:**

- 第 35 页的『包装器库文件』
- 第 411 页的『生命科学用户定义的函数库文件』

## 包装器库文件

必须具有包装器库文件您才能存取数据源。对于某些数据源，库文件是当您安装 DB2 Information Integrator 时添加至联合服务器的。对于其它数据源，必须运行链接编辑脚本才能创建库文件。

当为数据源注册包装器时必须具有包装器库文件。

您应该验证联合服务器上是否存在包装器库文件。对于想要存取的每个数据源都应该具有一组包装器库文件。

如果联合服务器上不存在包装器库文件，则必须手工运行链接编辑脚本来创建库文件。

### 相关任务:

- 第 34 页的『检查包装器库文件 (UNIX)』
- 第 36 页的『将包装器库手工链接至数据源客户机软件』

### 相关参考:

- 第 97 页的『BLAST 包装器库文件』
- 第 65 页的『BioRS 包装器库文件』
- 第 148 页的『DB2 包装器库文件』
- 第 160 页的『Documentum 包装器库文件』
- 第 183 页的『Entrez 包装器库文件』
- 第 204 页的『Excel 包装器库文件』
- 第 220 页的『Extended Search 包装器库文件』
- 第 244 页的『HMMER 包装器库文件』
- 第 261 页的『Informix 包装器库文件』
- 第 274 页的『Microsoft SQL Server 包装器库文件』
- 第 286 页的『ODBC 包装器库文件』
- 第 298 页的『OLE DB 包装器库文件』
- 第 309 页的『Oracle 包装器库文件』
- 第 322 页的『Sybase 包装器库文件』
- 第 344 页的『Teradata 包装器库文件』
- 第 332 页的『表结构文件包装器库文件』
- 第 116 页的『WebSphere Business Integration 包装器库文件』
- 第 360 页的『Web 服务包装器库文件』
- 第 389 页的『XML 包装器库文件』

## 检查链接编辑消息文件 (UNIX)

检查链接编辑消息文件是确认包装器库与数据源客户机软件之间的链接编辑这一大型任务的一部分。

如果链接编辑失败，则会将错误列示在库目录的消息文件中。存在消息文件并不意味着链接编辑失败。即使链接编辑成功，库目录中也将具有消息文件。

必须打开消息文件才能确定链接编辑是否失败。

只有下列数据源才需要执行此任务:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

**过程:**

要确定链接编辑是否失败, 打开链接编辑消息文件。链接编辑消息文件位于 DB2 的安装目录的 lib 或 lib64 子目录中。链接编辑消息文件的名称列示在下表中。

表 9. 按数据源列示的链接编辑消息文件名

数据源	消息文件名
Informix	djxlinkInformix.out
Microsoft SQL Server	djxlinkMssql.out
Oracle	djxlinkOracle.out
Sybase	djxlinkSybase.out
Teradata	djxlinkTeradata.out

当您设置联合服务器时造成链接失败可能有几个原因:

- 如果在尝试进行链接编辑之前未安装数据源客户机软件, 则链接编辑将失败。例如, 如果在安装 DB2 服务器软件之前未安装 Informix 客户机软件, 链接编辑就将失败。同样, 如果在安装 DB2 Information Integrator 之前未安装 Sybase Open Client 软件, 链接编辑就将失败。在这些情况下, 您将需要手工执行链接。
- 验证数据源客户机软件的版本是否受支持。如果您已经安装的数据源客户机软件的版本不受支持, 链接编辑就将失败。您将必须安装受支持的客户机版本然后手工执行链接。

**相关任务:**

- 第 33 页的『确认包装器库文件的链接编辑 (UNIX)』
- 第 34 页的『检查包装器库文件 (UNIX)』
- 第 36 页的『将包装器库手工链接至数据源客户机软件』

## 将包装器库手工链接至数据源客户机软件

将包装器库手工链接至数据源客户机软件是检查联合服务器的设置这一大型任务的一部分。

如果库文件不在目录路径中, 则必须手工链接包装器库。

此任务只适用于下列数据源:

- Informix
- Microsoft SQL Server
- Oracle

- Sybase
- Teradata

**先决条件:**

需要具有 root 用户权限才能运行链接脚本。

要使 djxlinkxxx 脚本使用您的语言发出它们的消息，必须至少具有一个 DB2 实例。如果实例不存在，这些脚本仍然会工作。但是，您将接收到错误消息。每条错误消息都以 db2djxmsg: Error retrieving message number 开头。此错误消息后面接着是一条英文消息。例如:

```
db2djxmsg: Error retrieving message number 9004
(return code -2029059891 from sqllogmsg).
Begin processing for wrapper: INFORMIX
INFORMIXDIR = /wsdb/v82/bldsupp/AIX/informix2.81
db2djxmsg: Error retrieving message number 9015
(return code -2029059891 from sqllogmsg).
Library libdb2informixF.a was built successfully.
db2djxmsg: Error retrieving message number 9006
(return code -2029059891 from sqllogmsg).
End processing for wrapper: INFORMIX
```

**过程:**

要将包装器库快速链接至数据源客户机软件:

1. 在 DB2 联合服务器上安装和配置客户机软件（如果需要的话）。
2. 使用产品 CD 来执行链接:
  - 对于 Informix 数据源，再次运行 DB2 服务器安装并指定“典型”安装选项。
  - 对于 Microsoft SQL Server、Oracle、Sybase 和 Teradata 数据源，再次运行 DB2 Information Integrator 安装。单击启动板中的**安装产品**，并遵循向导中的指示信息。
3. 执行链接之后，检查对包装器库的许可权。确保 DB2 实例所有者可以读取和执行这些库。

或者，可以从 UNIX 命令提示符处运行链接编辑脚本。

1. 打开 UNIX 命令提示符。
2. 对于想要存取的每个数据源运行链接编辑脚本。

下表列示了每个数据源的链接编辑脚本的名称。

表 10. 按数据源列示的链接脚本

数据源	链接脚本名称
Informix	djxlinkInformix
Microsoft SQL Server	djxlinkMssql
Oracle	djxlinkOracle
Sybase	djxlinkSybase
Teradata	djxlinkTeradata

例如，如果您正在设置联合服务器来存取 Informix 数据源，则从 UNIX 命令提示符处运行 djxlinkInformix 脚本:

```
djxlinkInformix
```

3. 在每个 DB2 实例上发出 **db2iupdt** 命令来启用对数据源的联合存取。
4. 执行链接之后，检查对包装器库的许可权。确保 DB2 实例所有者可以读取和执行这些库。

**注意：**还有一个脚本（即，djsxlink 脚本）将尝试为 DB2 Information Integrator 支持的每个数据源创建包装器库。如果您运行 djsxlink 脚本并且只为某些数据源安装了客户机软件，则对于未安装的每个数据源您都将接收到错误消息。

#### 相关任务:

- 『安装 DB2 Information Integrator (Windows) 』 ( 《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》 )
- 『安装 DB2 Information Integrator (UNIX) 』 ( 《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》 )
- 第 33 页的 『检查联合服务器的设置』
- 第 38 页的 『检查 FEDERATED 参数』
- 第 35 页的 『检查链接编辑消息文件 (UNIX) 』

---

## 检查 FEDERATED 参数

检查 FEDERATED 参数是检查联合服务器的设置这一大型任务的一部分。

在将数据源添加至联合服务器和数据库之前，应该检查 FEDERATED 参数设置。

必须将 FEDERATED 参数设置为 YES 才能使联合服务器能够存取数据源。

#### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上

#### 过程:

要检查 FEDERATED 参数设置:

1. 发出以下 DB2 命令以显示所有参数以及它们的当前设置:  

```
GET DATABASE MANAGER CONFIGURATION
```
2. 检查 CONCENTRATOR 参数设置。不能将 CONCENTRATOR 参数和 FEDERATED 参数同时配置为 YES。如果 CONCENTRATOR 参数设置为 YES，则将该设置更改为 NO。发出以下 DB2 命令来更改设置:  

```
UPDATE DATABASE MANAGER CONFIGURATION USING CONCENTRATOR NO
```
3. 检查 FEDERATED 参数设置。如果 FEDERATED 参数设置为 NO，则将该设置更改为 YES。发出以下 DB2 命令来更改设置:  

```
UPDATE DATABASE MANAGER CONFIGURATION USING FEDERATED YES
```

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 ( 《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》 )

#### 相关任务:

- 第 33 页的 『检查联合服务器的设置』

---

## 第 4 章 创建联合数据库

在可以配置联合服务器以存取数据源之前，必须创建用作联合数据库的数据库。

---

### 联合数据库本地语言注意事项

对于许多关系数据源，当包装器连接至数据源时包装器将执行下列任务：

1. 确定联合数据库的代码页和地域。
2. 将代码页和地域映射至数据源客户机语言环境名称。
3. 根据数据源，设置一个环境变量或者调用数据源 API 以将客户机语言环境告诉数据源。
4. 然后，数据源在远程数据库的代码页与联合数据库的代码页之间转换字符数据。

例如，如果联合数据库使用代码页 819，地域为 US。则等价的 Oracle 客户机语言环境为 American\_America.WE8ISO8859P1。包装器将把 NLS\_LANG 变量设置为 Oracle 客户机语言环境值。当将数据从 Oracle 数据库发送至包装器时，Oracle 数据库将把数据从代码集 American\_America.WE8ISO8859P1 转换为代码页 819。当将数据从 Oracle 数据库发送至包装器时，Oracle 服务器或客户机将把数据从 Oracle 数据库的代码页转换为代码页 819。当将数据从包装器发送至 Oracle 数据库时，Oracle 服务器或客户机将把数据从代码页 819 转换为 Oracle 数据库的代码页。

对于不执行代码页转换的关系数据源，某些包装器将执行转换。

将联合数据库定义为与数据源使用相同的代码集、地域和整理顺序可以提高性能。如果将联合数据库定义为与数据源使用相同的代码集、地域和整理顺序，则不需要进行代码页转换。当您传送大量字符数据时，使用相同的本地语言设置可以提高性能。

要指定联合数据库上的代码集和地域，在 CREATE DATABASE 语句上使用 USING CODESET 和 TERRITORY 参数。

#### 相关概念：

- 第 42 页的『联合系统的 Unicode 支持』
- 第 39 页的『联合系统中的整理顺序』

#### 相关任务：

- 第 44 页的『创建联合数据库』
- 第 41 页的『设置联合数据库整理顺序』

---

### 联合系统中的整理顺序

当联合服务器接收到查询时，DB2<sup>®</sup> SQL 编译器就会参阅全局目录和数据源包装器中的信息来帮助联合服务器处理查询。作为 SQL 编译器进程的一部分，查询优化器分析查询。编译器开发了一些备用策略（称为存取方案）来处理查询。存取方案可要求查询：

- 由数据源处理
- 由联合服务器处理
- 部分由数据源处理，部分由联合服务器处理

在数据源中执行字符排序和比较通常可以提高性能。

如果查询要求进行字符排序或比较，则 SQL 编译器将使用整理顺序信息来确定使用哪个存取方案。

缺省情况下，联合数据库整理顺序是区分大小写的。但是，某些联合数据源使用不区分大小写的整理顺序。如果联合数据库与数据源的整理顺序不同，则查询结果也可能不同。如果操作是进行字符排序，则将返回相同的数据，但是结果的顺序将不相同。如果操作是进行字符比较，则返回的结果可能会不相同。

在何处处理排序或比较取决于下面几个因素：

- 如果联合数据库整理顺序与数据源整理顺序相同，则可以在数据源中进行字符排序或比较。查询优化器可以决定哪个整理顺序是完成查询（本地操作或远程操作）的最有效方法。假定按数据源执行的所有类型的字符比较和排序都将产生相同的结果，就象是由联合数据库执行那些操作一样。
- 如果联合数据库整理顺序与数据源整理顺序不相同，但是数据源整理顺序是区分大小写的，则将在联合数据库中执行字符排序或比较。假定对于 WHERE=、DISTINCT 和 GROUP BY 操作，数据源将对字符数据产生相同的结果。但是带有大于或小于谓词的其它操作（例如，ORDER BY 和 WHERE）将对字符数据产生不同的结果。
- 如果联合数据库整理顺序与数据源整理顺序不相同，但是数据源整理顺序是不区分大小写的，则将在联合数据库中执行字符排序或比较。假定无论在请求的操作中是指定了大写还是小写，数据源都将把大写和小写字母当作是相同的，并且将把它们都包括在结果中。WHERE=、带有大于或小于谓词的 WHERE、ORDER BY、DISTINCT 和 GROUP BY 操作将不会下推至数据源。

例如，不区分大小写的数据源将为字符“S”和“s”指定相同的权重。具有“英语”代码页的不区分大小写的数据源将单词 STEWART、SteWArT 和 stewart 认为是相同的。但是，当使用区分大小写的整理顺序时，将为字符指定不同的权重。根据整理顺序是否区分大小写，字符排序或比较的结果集将有所不同。

当联合数据库与数据源的整理顺序不同时，联合服务器将检索数据，并在本地执行数据和字符排序和比较。之所以在本地执行这些任务是因为 DB2 用户期望看到按照为联合服务器定义的整理顺序排序的查询结果。通过在本地对数据进行排序，可以保证 DB2 用户的结果集是一致的。为本地排序和比较检索数据通常会降低性能。

如果需要看到按数据源整理顺序排序的字符数据，则可以在传递（Pass-Through）会话中提交查询。

要确定数据源与联合数据库是否具有相同的整理顺序，考虑下列因素：

**代码页** 联合服务器和数据源使用的代码页方案（例如，ASCII 和 EBCDIC）将影响结果。

#### **本地语言支持（NLS）**

整理顺序与服务器上受支持的语言有关。将您的操作系统的 DB2 NLS 信息与数据源 NLS 信息比较。



## 数据源特征

某些数据源是使用不区分大小写的整理顺序创建的，这可能会在 DB2 中的与顺序有关的操作中产生不同的结果。

**定制** 一些数据源为整理顺序提供多个选项或允许定制整理顺序。

设置整理顺序时有几个选项可供选择：

- 创建联合数据库时设置整理顺序
- 为数据源创建服务器定义时设置 COLLATING\_SEQUENCE 选项。此选项只可用于关系数据源。

## 相关概念：

- 第 42 页的『联合系统的 Unicode 支持』

## 相关任务：

- 第 44 页的『创建联合数据库』
- 第 41 页的『设置联合数据库整理顺序』

## 相关参考：

- 第 39 页的『联合数据库本地语言注意事项』

---

## 设置联合数据库整理顺序

管理员可使用与数据源整理顺序匹配的特定整理顺序创建联合数据库。

您将联合数据库整理顺序设置为 CREATE DATABASE API 的一部分。通过此 API，您可以指定下列顺序之一：

- 标识顺序
- 系统顺序（支持数据库的操作系统使用的顺序）
- 定制的顺序（DB2 UDB 提供或您自己定义的预定义顺序）

## 过程：

要指定联合数据库的整理顺序，在 CREATE DATABASE 语句上使用 COLLATE USING 参数。

对于关系数据源，如果联合数据库与数据源使用相同的整理顺序，则应该将 COLLATING\_SEQUENCE 服务器选项设置为 'Y'。将 COLLATING\_SEQUENCE 服务器选项设置为 'Y' 时将告诉联合服务器，联合数据库与数据源的整理顺序是相匹配的。为关系数据源创建服务器定义时设置 COLLATING\_SEQUENCE 服务器选项。

支持 COLLATING\_SEQUENCE 服务器选项的关系数据源包括：

- DB2 系列
- Informix
- Microsoft SQL Server
- ODBC
- OLE DB
- Oracle
- Sybase

- Teradata

#### 相关概念:

- 第 42 页的『联合系统的 Unicode 支持』
- 第 39 页的『联合系统中的整理顺序』

#### 相关任务:

- 第 44 页的『创建联合数据库』

#### 相关参考:

- 第 39 页的『联合数据库本地语言注意事项』

---

## 联合系统的 Unicode 支持

关系型和非关系型包装器及用户定义的函数可以在 Unicode 数据库（UTF-8 数据库）上运行。Unicode 数据库提供独立于平台的联合服务器环境。Unicode 数据库能够处理以各种代码页存储在不同数据源中的数据。

支持 Unicode 的包装器和用户定义的函数是:

- 关系型包装器
  - DRDA<sup>®</sup>
  - Informix<sup>®</sup>
  - MS SQL Server
  - ODBC
  - OLE DB
  - Oracle
  - Sybase
  - Teradata
- 非关系型包装器和用户定义的函数
  - BioRS 包装器
  - BLAST 包装器
  - Documentum 包装器
  - Entrez 包装器
  - Excel 包装器
  - HMMER 包装器
  - IBM<sup>®</sup> Lotus<sup>®</sup> Extended Search 包装器
  - KEGG 用户定义的函数
  - MQ 用户定义的函数
  - 表结构文件包装器
  - Web 服务用户定义的函数
  - Web 服务包装器
  - WebSphere<sup>®</sup> Business Integration 包装器
  - XML 包装器

在第 43 页的图 5 中，一个公司在不同国家或地区有分支机构。每个分支机构以自己的代码页用其自己的数据库存储客户数据。Microsoft<sup>®</sup> SQL Server 数据库以代码页 A 存储数据。Oracle 数据库以代码页 B 存储数据。代码页 A 和代码页 B 位于不同地域中。

要从不同地域集成数据，公司可以将联合数据库的代码页设置为 Unicode。随后公司可以将表连接起来以查看采购单总数，无论地域如何。

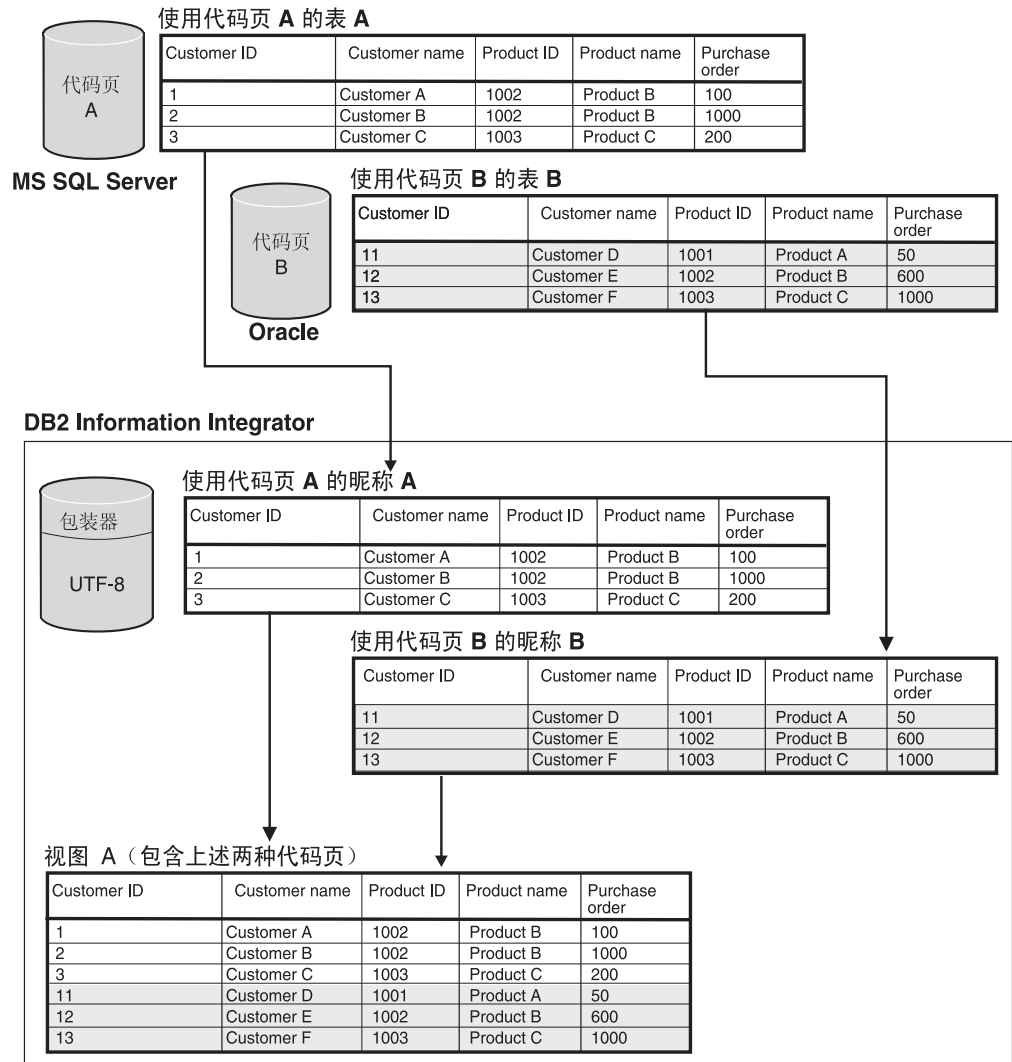


图 5. Unicode 示例

**相关任务:**

- 『为对 Microsoft SQL Server 和 ODBC 数据源的 Unicode 支持指定客户机代码』（《联合系统指南》）
- 『为对表结构文件数据源的 Unicode 支持指定文件代码页』（《联合系统指南》）

**相关参考:**

- 『Unicode 缺省正向数据类型映射 - NET8 包装器』（《联合系统指南》）
- 『MSSQL 和 ODBC 包装器 CODEPAGE 选项的受支持的 Unicode 代码页』（《联合系统指南》）
- 『Unicode 缺省逆向数据类型映射 - NET8 包装器』（《联合系统指南》）
- 『Unicode 缺省正向数据类型映射 - Sybase 包装器』（《联合系统指南》）
- 『Unicode 缺省逆向数据类型映射 - Sybase 包装器』（《联合系统指南》）
- 『Unicode 缺省正向数据类型映射 - ODBC 包装器』（《联合系统指南》）

- 『Unicode 缺省逆向数据类型映射 - ODBC 包装器』（《联合系统指南》）
- 『Unicode 缺省正向数据类型映射 - Microsoft SQL Server 包装器』（《联合系统指南》）
- 『Unicode 缺省逆向数据类型映射 - Microsoft SQL Server 包装器』（《联合系统指南》）
- 『为对表结构文件数据源的 Unicode 支持指定文件代码页 - 示例』（《联合系统指南》）

## 创建联合数据库

在设置联合服务器之后，DB2 实例所有者必须在联合服务器实例上创建将充当联合数据库的 DB2 数据库。

**建议：**如果您需要连接至的远程数据源正在使用另外的代码页或不兼容的代码页，则将联合数据库定义为 Unicode 数据库。要将联合数据库定义为 Unicode 数据库，应在 CREATE DATABASE 语句上指定 USING CODESET UTF-8。

必须在完成此步骤之后才能配置联合服务器来存取数据源。

### 先决条件:

- 具有创建 DB2 数据库的 SYSADM 或 SYSCTRL 权限。
- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 确定当您创建联合数据库时是否想指定整理顺序

### 过程:

可以从 DB2 控制中心或 DB2 命令行创建联合数据库。

要通过 DB2 控制中心执行此任务:

1. 右键单击**数据库**文件夹，并单击**创建 -> 数据库（使用向导）**。“创建数据库”向导打开。
2. 完成向导中的步骤。

要从 DB2 命令行执行此任务，请发出 CREATE DATABASE 命令。例如:

```
CREATE DATABASE federated
```

此命令将:

- 初始化新的数据库
- 创建三个初始表空间
- 创建系统表
- 分配恢复日志

如果 DB2 实例使用多分区配置，则 CREATE DATABASE 命令将影响 db2nodes.cfg 文件中列示的所有分区。发出此命令的数据库分区将成为新数据库的目录分区。

### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）

|                   • 第 49 页的『配置数据源的快速方法』

|                   **相关任务:**

|                   • 第 41 页的『设置联合数据库整理顺序』



---

## 第 3 部分 数据源





## 第 5 章 配置对数据源的存取概述

下列各节提供配置联合服务器和数据库以存取数据源的简明指南:

- 它们包含有关快速执行配置步骤所需的基本步骤的信息。
- 它们概述用来对数据源配置作细微调整的几个可选步骤（如果您需要它们的话）。

每个数据源都有各自的配置章节。

### 配置数据源的快速方法

通过 DB2<sup>®</sup> 控制中心可以完成配置对数据源的存取所需要的大多数步骤。对于需要命令行的步骤使用 DB2 命令中心。在这些图形用户界面之间进行切换可以快速配置对数据源的存取。

在配置对数据源的存取之前，应确保已经正确设置了联合服务器。

无论哪种数据源，配置联合服务器以存取数据源的步骤都是相似的。基本步骤和建议的界面为:

表 11. 建议的界面和配置步骤

配置步骤	建议的界面	注释
1. 为数据源准备联合服务器。	客户机配置助手	仅对于某些数据源是必需的。此步骤可能要求您安装软件、配置文件或检查设置。
2. 设置必需的环境变量。	DB2 控制中心	下列系统必须具有环境变量： <ul style="list-style-type: none"><li>• Documentum</li><li>• Informix<sup>®</sup></li><li>• Microsoft<sup>®</sup> SQL Server</li><li>• Oracle</li><li>• Sybase</li><li>• Teradata</li></ul>
2. 注册包装器。	DB2 控制中心中的“联合对象”向导。	对于想要存取的每个数据源，包装器都是必需的。
3. 注册服务器定义。	DB2 控制中心中的“联合对象”向导。	服务器定义与包装器相关联，并在注册昵称时使用。
4. 创建用户映射。	DB2 控制中心中的“联合对象”向导。	仅对于某些数据源是必需的。  如果您试图从 SYSCAT.USEROPTIONS 目录视图中检索与用户映射相关联的远程密码，则远程密码值将显示为已加密。
5. 测试与数据源服务器的连接。	DB2 命令中心	仅对于某些数据源是必需的。

表 11. 建议的界面和配置步骤 (续)

配置步骤	建议的界面	注释
6. 创建昵称。	DB2 控制中心中的“联合对象”向导。	对于想要存取的每个数据源对象，昵称都是必需的。

**相关概念:**

- 第 59 页的『可选的配置步骤』

**相关任务:**

- 第 50 页的『使用 DB2 UDB 控制中心将数据源添加至联合服务器』
- 第 54 页的『为数据源注册包装器』
- 第 55 页的『为数据源注册服务器定义』
- 第 56 页的『为数据源注册用户映射』
- 第 56 页的『为数据源注册昵称』
- 第 52 页的『设置数据源环境变量』
- 第 33 页的『检查联合服务器的设置』

---

## 使用 DB2 UDB 控制中心将数据源添加至联合服务器

要配置联合服务器以存取数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

**过程:**

DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。要启动该向导，右键单击**联合数据库对象**文件夹并单击**创建联合对象**。

对于每个数据源来说，配置联合服务器所需要的步骤是不同的。

可以使用“操作输出”窗口来配置多个联合服务器以存取数据源。

**相关任务:**

- 第 51 页的『配置多个联合服务器来存取数据源』
- 第 331 页的『将表结构文件数据源添加至联合服务器』
- 第 157 页的『将 Documentum 数据源添加至联合服务器』
- 第 202 页的『将 Excel 数据源添加至联合服务器』
- 第 90 页的『将 BLAST 数据源添加至联合服务器』
- 第 180 页的『将 Entrez 数据源添加至联合服务器』
- 第 339 页的『将 Teradata 数据源添加至联合服务器』
- 第 62 页的『将 BioRS 数据源添加至联合服务器』
- 第 145 页的『将 DB2 系列数据源添加至联合服务器』
- 第 218 页的『将 Extended Search 数据源添加至联合服务器』
- 第 236 页的『将 HMMER 数据源添加至联合服务器』
- 第 255 页的『将 Informix 数据源添加至联合服务器』

- 第 269 页的『将 Microsoft SQL Server 数据源添加至联合服务器』
- 第 283 页的『将 ODBC 数据源添加至联合服务器』
- 第 297 页的『将 OLE DB 数据源添加至联合服务器』
- 第 303 页的『将 Oracle 数据源添加至联合服务器』
- 第 317 页的『将 Sybase 数据源添加至联合服务器』
- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』

---

## 配置多个联合服务器来存取数据源

联合系统可以由多个联合服务器组成。无须单独配置每个联合服务器，可以使用 DB2 控制中心来配置多个联合服务器，从而节省时间。当配置第一个服务器时，**操作输出**窗口将捕获您在创建联合对象时发出的 DDL 语句。可以重用或修改这些语句，并应用这些语句以便快速配置其它联合服务器。

对于当前会话，**操作输出**窗口保持为活动的。如果关闭**操作输出**窗口，则当前会话的 DDL 语句将继续存储在**操作输出**窗口中。但是，如果关闭 DB2 控制中心，则会从**操作输出**窗口中除去当前会话的所有 DDL 语句。

### 先决条件:

- DB2 Information Integrator 安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

### 过程:

要配置多个联合服务器来存取数据源:

1. 使用 DB2 控制中心来为想要存取的数据源配置第一个联合服务器。这将捕获每个 DDL 语句。
2. 在**操作输出**窗口中显示**操作输出**页。

如果关闭了**操作输出**窗口，则右键单击**联合数据库对象**文件夹并单击**显示操作**以打开**操作输出**窗口。

3. 删除不想在其它联合服务器上使用的任何 DDL 语句。要删除某个语句，右键单击该语句并单击**除去**。例如，您可能想删除在**操作输出**页面上的状态列中显示“失败”的任何语句。
4. 将想在其它联合服务器上使用的语句复制到**命令编辑器**页面中:
  - a. 选择想要复制的语句。要选择多个语句，应使用 Ctrl 键。
  - b. 右键单击所选择的语句并单击**复制到命令编辑器**。**命令编辑器**页面打开。
5. 在**命令编辑器**页面中更改您想在其它联合服务器上使用的任何 DDL 语句。例如，您可能想更改指定本地模式的任何语句。

必须更改用户映射以指定密码。当在**操作输出**窗口中捕获到 CREATE USER MAPPING 语句的 DDL 时，就会使用星号将密码屏蔽起来。可以使用正确的密码来替换星号。

6. 在下一个联合服务器上发出 DDL 语句。

### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 第 49 页的『配置数据源的快速方法』

相关任务:

- 第 44 页的『创建联合数据库』

---

## 设置数据源环境变量

### 设置数据源环境变量

当安装 DB2 Information Integrator 时，安装过程将尝试设置某些数据源需要的环境变量。在某些情况下，例如，如果在安装 DB2 Information Integrator 之前未在联合服务器上安装客户机软件，则安装过程可能不能设置环境变量。

当遵循一定步骤将数据源添加至联合服务器时，可以检查环境变量并设置它们（如果需要的话）。

- 如果使用 DB2 控制中心来将数据源添加至联合服务器，则会自动检查对于环境变量的需求。可以在创建或改变包装器时设置环境变量。
- 如果使用 DB2 命令行来将数据源添加至联合服务器，则必须手工设置环境变量。

对于下列数据源必须设置环境变量:

- Documentum
- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

先决条件:

应该由系统管理员执行此任务。

限制:

请参阅主题: db2dj.ini 文件的限制

过程:

对于每个数据源，检查环境变量的步骤是不同的。

相关概念:

- 第 49 页的『配置数据源的快速方法』

相关任务:

- 第 158 页的『设置 Documentum 环境变量』
- 第 54 页的『为数据源注册包装器』
- 第 257 页的『设置 Informix 环境变量』
- 第 271 页的『设置 Microsoft SQL Server 环境变量』
- 第 304 页的『设置 Oracle 环境变量』

- 第 318 页的『设置 Sybase 环境变量』
- 第 341 页的『设置 Teradata 环境变量』

相关参考:

- 第 53 页的『对于 db2dj.ini 文件的限制』

## 对于 db2dj.ini 文件的限制

下列限制适用于 db2dj.ini 文件:

- 各个条目的格式必须为 *evname=value*

*evname* 是环境变量的名称, 而 *value* 是它的值。

- 环境变量名的最大长度为 255 个字节。
- 环境变量值的最大长度为 765 个字节。
- 文件中任何一行的最大长度为 1021 个字节。超过该长度的数据将被忽略。

相关任务:

- 第 158 页的『设置 Documentum 环境变量』
- 第 52 页的『设置数据源环境变量』
- 第 257 页的『设置 Informix 环境变量』
- 第 271 页的『设置 Microsoft SQL Server 环境变量』
- 第 304 页的『设置 Oracle 环境变量』
- 第 318 页的『设置 Sybase 环境变量』
- 第 341 页的『设置 Teradata 环境变量』

## 在多分区实例配置中应用环境变量

如果联合服务器实例具有多分区配置, 则必须将数据源环境变量应用于所有分区。

仅当联合服务器具有多分区实例配置时才需要执行此步骤。

db2dj.ini 文件包含数据源环境变量。此文件是在您安装 DB2 Information Integrator 时添加至联合服务器的。

必须将 db2dj.ini 文件的副本添加至多分区实例配置中的所有分区。该文件的缺省名称为 db2dj.ini 文件。在 UNIX 联合服务器上, db2dj.ini 文件的缺省路径为 INSTHOME/sqllib/cfg, 其中 INSTHOME 是实例所有者的主目录。在 Windows 联合服务器上, 该文件的缺省路径为 x:\SQLLIB\cfg, 其中 x:\SQLLIB 是在 DB2PATH 注册表变量或环境变量中指定的驱动器和目录。

db2dj.ini 文件的缺省路径和文件名可以被 DB2\_DJ\_INI 注册表变量覆盖。

可以将 DB2\_DJ\_INI 注册表变量应用于所有分区或一部分分区。

过程:

要将 DB2\_DJ\_INI 注册表变量应用于联合服务器上的适当分区, 可使用 **db2set** 命令。**db2set** 命令将显示、设置或除去 DB2 概要文件变量。

您使用的 **db2set** 命令的语法取决于数据库系统结构。

- 要将 DB2\_DJ\_INI 注册表变量应用于此实例中的所有数据库分区，可发出以下命令：

```
db2set -g DB2_DJ_INI=$HOME/sql1lib/cfg/my_db2dj.ini
```

- 要将 DB2\_DJ\_INI 注册表变量只应用于当前分区，可发出以下命令：

```
db2set DB2_DJ_INI=$HOME/sql1lib/cfg/my_db2dj.ini
```

- 要将 DB2\_DJ\_INI 注册表变量应用于特定分区，可发出以下命令：

```
db2set -i INSTANCEX 3 DB2_DJ_INI=$HOME/sql1lib/cfg/partition3.ini
```

*INSTANCEX*

实例的名称。

3 db2nodes.cfg 文件中列示的分区号。

*partition3.ini*

db2dj.ini 文件的经过修改和重命名的版本。

**注意：**设置 DB2\_DJ\_INI 注册表变量时，必须将它设置为一个绝对路径。如果 FEDERATED 参数设置为 YES 而 DB2\_DJ\_INI 注册表变量设置为相对路径，则将不会启动 DB2 通用数据库引擎。

**相关任务：**

- 第 52 页的『设置数据源环境变量』

---

## 为数据源注册包装器

为想要存取的每种类型的数据源都注册一个包装器。要存取三个 DB2 z/OS 版数据库表、一个 DB2 iSeries 版表和两种 BLAST 搜索类型，需要创建两个包装器。需要为 DB2 数据库注册一个包装器，为 BLAST 搜索类型注册一个包装器。

在联合数据库中注册了包装器之后，您就可以使用这些包装器来从那些数据源中存取其它对象。例如，可以使用 DRDA 包装器来存取所有 DB2 系列数据源对象中的数据源，这些 DB2 系列数据源对象包括：DB2 Linux 版、UNIX 版和 Windows 版、DB2 z/OS 和 OS/390 版、DB2 iSeries 版以及 DB2 Server VM 和 VSE 版。

**过程：**

可以从 DB2 控制中心或 DB2 命令行注册包装器：

- 要从 DB2 控制中心执行此任务，使用“联合对象”向导或者右键单击**联合对象**文件夹并单击**创建包装器**。
- 要从 DB2 命令行执行此任务，请使用 CREATE WRAPPER 语句。

**相关任务：**

- 第 332 页的『注册表结构文件包装器』
- 第 160 页的『注册 Documentum 包装器』
- 第 203 页的『注册 Excel 包装器』
- 第 97 页的『注册 BLAST 包装器』
- 第 389 页的『注册 XML 包装器』
- 第 182 页的『注册 Entrez 包装器』

- 第 344 页的『注册 Teradata 包装器』
- 第 65 页的『注册 BioRS 包装器』
- 第 147 页的『注册 DB2 包装器』
- 第 219 页的『注册 Extended Search 包装器』
- 第 243 页的『注册 HMMER 包装器』
- 第 260 页的『注册 Informix 包装器』
- 第 273 页的『注册 Microsoft SQL Server 包装器』
- 第 285 页的『注册 ODBC 包装器』
- 第 298 页的『注册 OLE DB 包装器』
- 第 308 页的『注册 Oracle 包装器』
- 第 321 页的『注册 Sybase 包装器』
- 第 116 页的『注册 WebSphere Business Integration 包装器』
- 第 359 页的『注册 Web 服务包装器』

---

## 为数据源注册服务器定义

对于不同的数据源，服务器定义的用途有所不同。

关系数据源的服务器定义通常表示远程数据库、数据库分区或节点。对于非关系数据源，某些服务器定义映射至搜索类型和守护进程，某些映射至 Web 站点，而某些映射至 Web 服务器。对于其它非关系数据源，创建服务器定义只是因为进行联合需要服务器定义。

为其创建昵称的每个数据源对象必须与特定的服务器定义相关联。

对于某些数据源，注册服务器定义时必须指定节点。对于不同的数据源，节点的概念有所不同。对于关系数据源，节点将反映数据源的服务器实例。

### 过程:

可以从 DB2 控制中心或 DB2 命令行注册服务器定义:

- 要从 DB2 控制中心执行此任务，使用“联合对象”向导或者右键单击**服务器定义**文件夹并单击**创建**。
- 要从 DB2 命令行执行此任务，请使用 CREATE SERVER 语句。

### 相关任务:

- 第 333 页的『为表结构文件注册服务器』
- 第 161 页的『为 Documentum 数据源注册服务器』
- 第 204 页的『为 Excel 数据源注册服务器』
- 第 98 页的『为 BLAST 数据源注册服务器』
- 第 390 页的『为 XML 数据源注册服务器』
- 第 183 页的『为 Entrez 数据源注册服务器』
- 第 345 页的『为 Teradata 数据源注册服务器定义』
- 第 66 页的『为 BioRS 数据源注册服务器定义』
- 第 149 页的『为 DB2 数据源注册服务器定义』

- 第 220 页的『为 Extended Search 数据源注册服务器』
- 第 244 页的『为 HMMER 数据源注册服务器定义』
- 第 261 页的『为 Informix 数据源注册服务器定义』
- 第 274 页的『为 Microsoft SQL Server 数据源注册服务器定义』
- 第 287 页的『为 ODBC 数据源注册服务器定义』
- 第 299 页的『为 OLE DB 数据源注册服务器定义』
- 第 309 页的『为 Oracle 数据源注册服务器定义』
- 第 322 页的『为 Sybase 数据源注册服务器定义』
- 第 117 页的『为商业应用程序数据源注册服务器定义』
- 第 360 页的『为 Web 服务数据源注册服务器定义』

---

## 为数据源注册用户映射

对于某些数据源，必须定义联合服务器授权标识与数据源用户标识和密码之间的关联。为使用联合系统来发送分布式请求的每个用户标识创建用户映射。

### 过程:

可以从 DB2 控制中心或 DB2 命令行创建用户映射:

- 要从 DB2 控制中心执行此任务，使用“联合对象”向导或者右键单击**用户映射**文件夹并单击**创建**。
- 要从 DB2 命令行执行此任务，请使用 CREATE USER MAPPING 语句。

### 相关任务:

- 第 161 页的『为 Documentum 数据源注册用户映射』
- 第 347 页的『为 Teradata 数据源创建用户映射』
- 第 66 页的『为 BioRS 数据源注册用户映射』
- 第 151 页的『为 DB2 数据源创建用户映射』
- 第 221 页的『为 Extended Search 数据源注册用户映射』
- 第 263 页的『为 Informix 数据源创建用户映射』
- 第 277 页的『为 Microsoft SQL Server 数据源创建用户映射』
- 第 289 页的『为 ODBC 数据源创建用户映射』
- 第 300 页的『为 OLE DB 数据源创建用户映射』
- 第 311 页的『为 Oracle 数据源创建用户映射』
- 第 325 页的『为 Sybase 数据源创建用户映射』

---

## 为数据源注册昵称

### 为数据源注册昵称

注册昵称的任务通常是配置任务中最复杂的。对于每个数据源，注册昵称的步骤和需求是不同的。



| 建议：使用 DB2 控制中心来注册昵称。对于大多数数据源，可以使用 DB2 控制中心  
| 中的“发现”工具。“发现”工具可以帮助您快速标识您想要为其注册昵称的数据源  
| 对象。

| 必须为想要存取每个数据源对象注册昵称。

| 数据源对象可以是关系数据源对象，也可以是非关系数据源对象：

- 关系数据源对象的示例为：数据库表、视图和同义词（仅适用于 Informix 和 Oracle）
- 非关系数据源对象的示例为：支持 BLAST 的数据库、Documentum Docbase 中的对象和已注册的表、Microsoft Excel 文件（.xls）、表结构文件（.txt）和 XML 标记文件

| 驻留在联合数据库中的表和视图是本地对象。不要为这些对象注册昵称。在查询中使用实际的对象名。

| 数据源对象为：

- 联合服务器上的另一个 DB2<sup>®</sup> 数据库实例中的表和视图。
- 另一个服务器上的 DB2 实例中的表和视图。
- 驻留在另一个数据源中的数据源对象，例如：Oracle、Sybase、Documentum 和 ODBC。

| 必须为这些对象注册昵称。有时候，数据源对象又称为远程对象。

| 当对联合服务器提交分布式请求时，该请求将按数据源对象的昵称来引用它们。昵称  
| 被映射至数据源中的特定对象名。映射可使得不需要按数据源名称来限定昵称。对于  
| 客户机应用程序或最终用户来说，数据源对象的位置是透明的。昵称不是数据源对象的  
| 备用名称。它们是联合服务器引用这些对象时所采用的指针。

| 例如，如果您定义昵称 *DEPT* 来表示一个称为 *NFX1.PERSON.DEPT* 的 Informix 数据库，  
| 则允许从联合服务器中使用语句 `SELECT * FROM DEPT`。但是，不允许使用语句  
| `SELECT * FROM NFX1.PERSON.DEPT`。

| 当注册昵称时，有关该昵称的元数据信息将存储在联合数据库系统目录中。对于关系  
| 数据源对象，将检索远程服务器中的目录数据，并将这些数据存储在联合数据库系统  
| 目录中。对于非关系数据源，将数据源信息存储在联合数据库系统目录中所采用的方式  
| 将随着数据源不同而不同。信息可能是从远程服务器中检索的，否则就要将此信息  
| 包括在 `CREATE NICKNAME` 语句中。

| “SQL 编译器”使用此元数据信息来使存取数据源对象更方便。例如，当为具有索引的  
| 表注册了昵称时，与该索引相关的元数据信息便存储在联合数据库系统目录中。当查  
| 询昵称时，“SQL 编译器”使用索引元数据信息，例如，索引键中每一列的名称。

| 过程：

| 可以从 DB2 控制中心或 DB2 命令行注册昵称：

- 要从 DB2 控制中心执行此任务，使用“联合对象”向导或者右键单击昵称文件夹并单击**创建**。使用“发现”工具来标识想要为其创建昵称的对象。
- 要从 DB2 命令行执行此任务，请使用 `CREATE NICKNAME` 语句。

| 可以为同一个数据源对象定义多个昵称。

#### 相关任务:

- 第 333 页的『为表结构文件注册昵称』
- 第 162 页的『为 Documentum 数据源注册昵称』
- 第 205 页的『为 Excel 数据源注册昵称』
- 第 98 页的『为 BLAST 数据源注册昵称』
- 第 394 页的『为 XML 数据源注册昵称』
- 第 185 页的『为 Entrez 数据源注册昵称』
- 第 350 页的『为 Teradata 表和视图注册昵称』
- 第 67 页的『为 BioRS 数据源注册昵称』
- 第 153 页的『为 DB2 表和视图注册昵称』
- 第 221 页的『为 Extended Search 数据源注册昵称』
- 第 246 页的『为 HMMER 数据源注册昵称』
- 第 266 页的『为 Informix 表、视图和同义词注册昵称』
- 第 279 页的『为 Microsoft SQL Server 表和视图注册昵称』
- 第 291 页的『为 ODBC 数据源表和视图注册昵称』
- 第 313 页的『为 Oracle 表和视图注册昵称』
- 第 327 页的『为 Sybase 表和视图注册昵称』
- 第 118 页的『为商业应用程序数据源注册昵称』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考:

- 第 19 页的『联合对象命名规则』

## 对昵称创建昵称

您可能偶尔需要对昵称创建昵称。

#### 过程:

假定您具有一个使用 AIX® 的联合服务器和一个使用 Windows 的联合服务器。您想要从这两个联合服务器存取 Excel 电子表格。但是，Excel 包装器只在使用 Windows 的联合服务器上才受支持。要从 AIX 联合服务器存取 Excel 电子表格，使用下列步骤：

1. 在 Windows 联合服务器上，安装 DB2 Information Integrator。
2. 配置 Windows 联合服务器来存取 Excel 数据源。
3. 在 Windows 联合服务器上，为 Excel 电子表格创建昵称。
4. 在 AIX 联合服务器上，安装 DB2 Information Integrator。
5. 配置 AIX 联合服务器来存取 DB2 系列数据源。
6. 在 AIX 联合服务器上，为 Windows 联合服务器上的 Excel 昵称创建昵称。

## 为非关系数据源指定昵称列

对于某些非关系数据源，注册昵称时必须定义一些列的列表。您指定的每一列将映射至数据源对象中的特定字段、列或元素。

某些非关系数据源的包装器要求一组固定输入列和固定输出列。当注册昵称时将自动定义固定列，并且会将这些固定列添加至联合数据库系统目录中。

#### 过程:

要在注册昵称时定义一些列的列表，应指定列名和数据类型。还可以对昵称列指定一个选项。

#### 相关任务:

- 第 56 页的『为数据源注册昵称』

---

## 可选的配置步骤

当配置联合服务器以存取数据源时，有几个可选步骤可能需要执行。

#### 索引规范:

为不具有索引的对象定义索引规范。例如，当表获得新索引时或者如果数据源对象（例如，视图）通常没有索引，您就要创建索引规范。

#### 数据类型映射:

只能为关系数据源指定备用数据类型映射。

指定备用数据类型映射，而不使用缺省数据类型映射。可以指定只用于特定数据源对象（例如，数据库中的特定表）的映射。

#### 函数映射:

只能为关系数据源指定函数映射。

定义备用的函数映射，而不使用缺省函数映射。这对于您想强制 DB2<sup>®</sup> 使用数据源中的用户定义的函数时特别有用。

#### 相关概念:

- 『联合系统中的数据类型映射』（《联合系统指南》）
- 『联合系统中的函数映射』（《联合系统指南》）
- 『联合系统中的索引规范』（《联合系统指南》）
- 第 49 页的『配置数据源的快速方法』



---

## 第 6 章 配置对 BioRS 数据源的存取

本章说明如何配置联合服务器以存取存储在 BioRS 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 BioRS 数据源的存取。

本章包含下列内容:

- 说明什么是 BioRS
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 BioRS 包装器相关联的错误消息

---

### 什么是 BioRS?

BioRS 是由 Biomax Informatics 开发的查询和检索系统。可以使用 BioRS 来从包括平面文件和关系数据库在内的多个数据源检索信息。您通常将公用数据（例如，SwissProt 和 GenBank）作为平面文件下载到 BioRS 系统。BioRS 可以将公用数据源和私有数据源（例如，由您所在的组织维护的私有数据库）集成到公共环境中。

数据源在集成到 BioRS 系统中之后，它被称为数据银行。包含在每个数据银行条目中的元素统称模式。数据银行的建立了索引的元素可以用于 BIOR.S.CONTAINS、BIORS.CONTAINS\_GE 和 BIOR.S.CONTAINS\_LE 函数中。BioRS 函数是在 SELECT 语句的 WHERE 子句中指定的。可以在 SELECT 列表和 WHERE 子句中的其它谓词中引用未建立索引的元素。未建立索引的元素由联合服务器来处理。

可以在数据银行中的条目之间建立关系，以便可以在 BioRS 系统中将数据银行链接到一起。

BioRS 数据银行可以具有父子关系（数据银行可以嵌套）。在这样的关系中，子数据银行包含名为 PARENT 的“引用”数据类型元素。PARENT 元素引用父数据银行的 \_ID\_ 元素。除了存在这个预定义的 PARENT 元素之外，嵌套的数据银行与不嵌套的数据银行包含相同的数据。

BioRS 提供了基于 Web 的界面，此界面使用户能够对 BioRS 数据银行中的数据运行查询。BioRS 包装器与 BioRS 的基于 Web 的界面使用相同的应用程序编程接口（API）来运行查询。

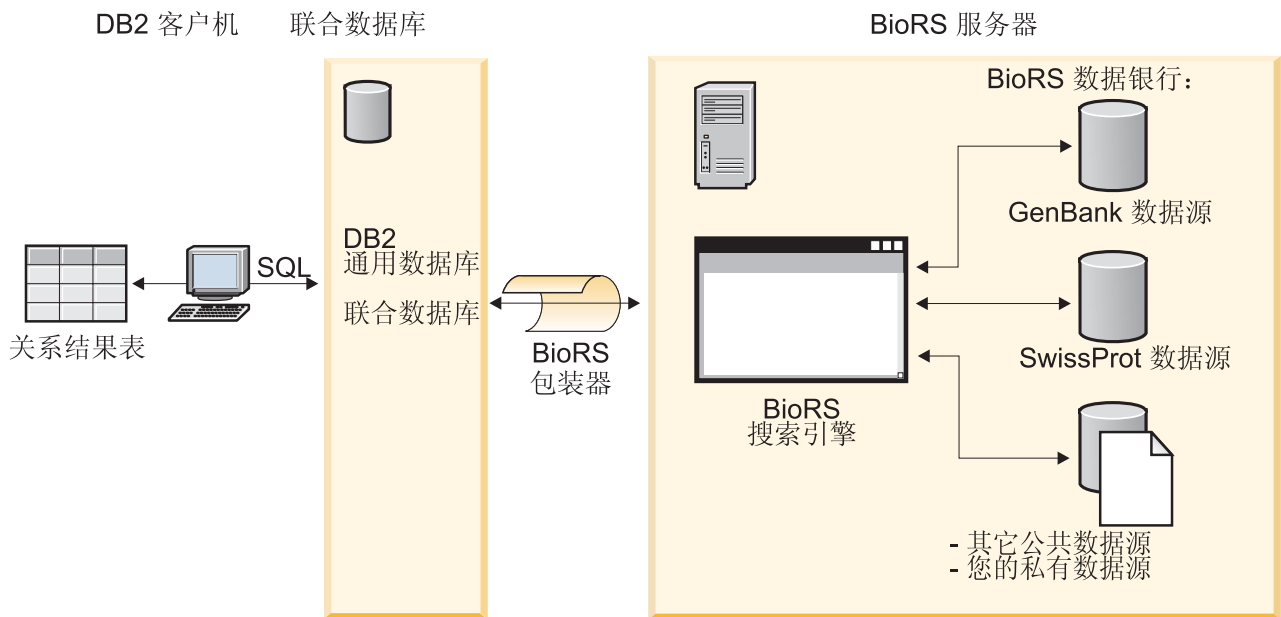


图 6. BioRS 包装器的工作方式

用户或应用程序使用 SQL 语句来从客户机提交查询。然后，将查询发送到安装了 BioRS 包装器的联合系统。根据查询的构造方式的不同，联合服务器和 BioRS 服务器都可以用来处理查询。BioRS 服务器与联合系统可以位于不同的计算机上。对于每个查询，联合系统都必须向 BioRS 服务器提供认证信息。此信息可以是用户标识和密码信息，也可以是未经认证的指示（通常是来宾帐户）。

BioRS 包装器与 BioRS V5.0.14 配合工作。

有关 BioRS 产品的详细信息，请参阅 Biomax Web 站点，网址为：  
<http://www.biomax.com>。

相关任务:

- 第 62 页的『将 BioRS 数据源添加至联合服务器』

相关参考:

- 第 74 页的『BioRS 包装器 - 示例查询』

## 将 BioRS 添加至联合服务器

### 将 BioRS 数据源添加至联合服务器

要配置联合服务器以存取 BioRS 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 BioRS 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上

- 联合数据库必须存在于联合服务器上

#### 过程:

要将 BioRS 数据源添加至联合服务器:

1. 为 BioRS 包装器注册定制函数。
2. 注册 BioRS 包装器。
3. 注册 BioRS 服务器定义。
4. 可选: 创建用户映射。
5. 为 BioRS 数据银行注册昵称。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 63 页的『为 BioRS 包装器注册定制函数』
- 第 65 页的『注册 BioRS 包装器』
- 第 66 页的『为 BioRS 数据源注册服务器定义』
- 第 66 页的『为 BioRS 数据源注册用户映射』
- 第 67 页的『为 BioRS 数据源注册昵称』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 为 BioRS 包装器注册定制函数

为 BioRS 包装器注册定制函数是将 BioRS 数据源添加至联合服务器这一大型任务的一部分。

BioRS 包装器的定制函数为:

- biors.contains
- biors.contains\_le
- biors.contains\_ge
- biors.search\_term

#### 先决条件:

- 必须使用模式名 biors 来注册 BioRS 包装器的所有定制函数。
- 必须为安装了 BioRS 包装器的每个联合数据库将每个定制函数都注册一次。

#### 过程:

要注册定制函数，请发出带有 AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION 关键字的 CREATE FUNCTION 语句。

每个函数的标准名为 biors.function\_name。

以下示例将注册一个版本的 CONTAINS 函数：

```
CREATE FUNCTION biors.contains (varchar(), varchar())  
    RETURNS INTEGER AS TEMPLATE  
    DETERMINISTIC NO EXTERNAL ACTION;
```

要注册定制函数，使用样本文件 create\_function\_mappings.ddl。该样本文件位于 sqllib/samples/lifesci/biors 目录中。样本文件包含每个定制函数的定义。可以运行此 DDL 文件来对安装了 BioRS 包装器的每个联合数据库注册定制函数。

在这一系列任务中的下一个任务是注册 BioRS 包装器。

#### 相关任务：

- 第 65 页的『注册 BioRS 包装器』

#### 相关参考：

- 『CREATE FUNCTION (Sourced or Template) statement』 (*SQL Reference, Volume 2*)
- 第 70 页的『定制函数和 BioRS 查询』
- 第 74 页的『BioRS 包装器 - 示例查询』
- 第 64 页的『定制函数表 - BioRS 包装器』

## 定制函数表 - BioRS 包装器

使用 CREATE FUNCTION 语句来注册 BioRS 定制函数。

下表列示了四个 BioRS 定制函数，并提供了当您注册函数时可以指定的数据类型的示例。

在函数中指定的第一种数据类型是已建立索引的列。在函数中指定的第二种数据类型是搜索项。

表 12. BioRS 包装器的定制函数

函数	描述
biors.contains (varchar(), varchar())	搜索已建立索引的列，以找到与您指定的值相等（根据 BioRS 查询语义来判断）的值。第一个自变量必须是对已建立索引的列的引用，第二个自变量是您指定的值。
biors.contains (varchar(), char())	
biors.contains (varchar(), date)	
biors.contains (varchar(), timestamp)	
biors.contains_LE (varchar(), varchar())	搜索已建立索引的列，以找到小于或等于（根据 BioRS 查询语义来判断）您指定的值的值。第一个自变量必须是对已建立索引的列的引用，第二个自变量是您指定的值。
biors.contains_LE (varchar(), smallint)	
biors.contains_LE (varchar(), bigint)	
biors.contains_LE (varchar(), decimal)	
biors.contains_LE (varchar(), double)	
biors.contains_LE (varchar(), real)	



表 12. BioRS 包装器的定制函数 (续)

函数	描述
biors.contains_GE (char(), char())	搜索已建立索引的列，以找到大于或等于（根据 BioRS 查询语义来判断）您指定的值的值。第一个自变量必须是对已建立索引的列的引用，第二个自变量是您指定的值。
biors.contains_GE (char(), date)	
biors.contains_GE (char(), timestamp)	
biors.contains_GE (char(), integer)	
biors.contains_GE (char(), smallint)	
biors.contains_GE (clob(), date)	
biors.search_term (varchar(), varchar())	将 BioRS 搜索项传递至 BioRS 搜索引擎。
biors.search_term (varchar(), char())	
biors.search_term (char(), varchar())	
biors.search_term (char(), char())	

**相关任务:**

- 第 63 页的『为 BioRS 包装器注册定制函数』

## 注册 BioRS 包装器

注册 BioRS 包装器是将 BioRS 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 BioRS 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

**过程:**

要注册包装器，使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 biors\_wrapper 的包装器，请发出以下语句：

```
CREATE WRAPPER biors_wrapper LIBRARY 'libdb2lsbiors.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 BioRS 包装器库文件的列表。

在这一系列任务中的下一个任务是为 BioRS 包装器注册服务器定义。

**相关参考:**

- 第 65 页的『BioRS 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## BioRS 包装器库文件

下表列示了 BioRS 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2lsbiors.a、libdb2lsbiorsF.a 和 libdb2lsbiorsU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 13. BioRS 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2lsbiors.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsbiors.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsbiors.so
Windows	%DB2PATH%\bin	db2lsbiors.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

#### 相关参考:

- 第 35 页的『包装器库文件』

## 为 BioRS 数据源注册服务器定义

为 BioRS 数据源注册服务器定义是将 BioRS 添加至联合系统这一大型任务的一部分。在注册包装器之后，必须注册相应的服务器定义。

#### 过程:

要在联合数据库中注册 BioRS 服务器定义，发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER brs_server WRAPPER wrap_biors OPTIONS(NODE 'biors_server2.com');
```

在这一系列任务中的下一个任务是为 BioRS 数据源注册用户映射。

#### 相关任务:

- 第 67 页的『为 BioRS 数据源注册昵称』

#### 相关参考:

- 第 495 页的『CREATE SERVER 语句选项 - BioRS 包装器』

## 为 BioRS 数据源注册用户映射

注册用户映射是将 BioRS 添加至联合系统这一大型任务的一部分。

根据帐户访问方法或 BioRS 系统中使用的方法的不同，可能不需要创建用户映射。

- 如果 BioRS 服务器是针对所有用户帐户的来宾访问配置的，则不需要在 DB2 Information Integrator 中创建用户映射。
- 如果 BioRS 服务器配置为使用标识和密码来认证用户帐户，则必须在联合数据库中为必须使用 BioRS 包装器的帐户创建用户映射。
- 如果 BioRS 服务器配置为既使用来宾也使用经过认证的用户帐户，则必须在联合数据库中为必须使用 BioRS 包装器的帐户创建经过认证的用户帐户的用户映射。

用户映射可以用来认证使用 BioRS 包装器查询 BioRS 数据源的用户或应用程序的存取权。如果用户或应用程序对已注册的 BioRS 昵称提交 SQL 查询，但是没有为该用户或

应用程序定义任何用户映射，则 BioRS 包装器使用缺省用户标识和密码来尝试从远程 BioRS 服务器检索数据。如果正在查询的数据银行要求认证，则可能返回错误消息。

为了确保将正确的用户标识和密码传递给 BioRS 服务器，请在联合数据库中为经授权可搜索 BioRS 数据源的用户创建用户映射。创建用户映射时，密码以加密格式存储在联合数据库系统目录表中。

#### 过程:

要注册 BioRS 用户映射，使用 CREATE USER MAPPING 语句。

例如，以下 CREATE USER MAPPING 语句将用户 Charlie 映射至 Biors\_Server1 服务器上的用户 Charlene。

```
CREATE USER MAPPING FOR Charlie SERVER Biors_Server1
OPTIONS(REMOTE_AUTHID 'Charlene', REMOTE_PASSWORD 'Charlene_pw');
```

还可以定义您自己的用户映射。在以下示例中，USER 是标识当前用户的关键字，而不是 USER 的用户名。

```
CREATE USER MAPPING FOR USER SERVER Biors_Server1
OPTIONS(REMOTE_AUTHID 'Yudong', REMOTE_PASSWORD 'Yudong_pw')
```

在这一系列任务中的下一个任务是为 BioRS 数据源注册昵称。

#### 相关任务:

- 第 67 页的『为 BioRS 数据源注册昵称』

#### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）
- 第 496 页的『CREATE USER MAPPING 语句选项 - BioRS 包装器』

## 为 BioRS 数据源注册昵称

为 BioRS 数据源注册昵称是将 BioRS 添加至联合服务器这一大型任务的一部分。

在注册服务器定义之后，必须为想要存取每个 BioRS 数据源注册昵称。当您在查询中引用 BioRS 数据源时就要使用昵称。

在将数据源集成到 BioRS 系统中之后，它在 BioRS 中就被称为数据银行。BioRS 中的数据银行等同于联合系统中的昵称。

#### 先决条件:

- 如果 BioRS 数据银行名不符合 CREATE NICKNAME 语句需要的语法，则注册昵称时必须使用 REMOTE\_OBJECT 昵称选项。
- 如果 BioRS 元素名称不符合 CREATE NICKNAME 语句需要的语法，则注册昵称时必须使用 ELEMENT\_NAME 列选项。

#### 限制:

不要使用 BioRS AllText 元素作为昵称的第一列。可以在任何其它列位置（例如，作为第二列或作为第三列）中使用 BioRS AllText 元素。

#### 过程:

要注册 BioRS 昵称，使用 CREATE NICKNAME 语句。

当创建 BioRS 昵称时，定义昵称列的列表。指定的昵称列必须与特定 BioRS 数据银行格式的元素相应。BioRS 为元素定义 5 种可能的数据类型：文本、数字、日期、作者和引用。BioRS 数据类型只能映射至 DB2 数据类型 CHAR、CLOB 或 VARCHAR。

为 BioRS 数据银行注册昵称的最简单的方法是对昵称指定与 BioRS 数据银行相同的名称。例如：

```
CREATE NICKNAME SwissProt
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR(128),
  ENTRYDATE VARCHAR (64))
FOR SERVER brs_server;
```

BioRS 数据银行 SwissProt 是昵称的名称。

使用这个简单的 CREATE NICKNAME 语法将限制您对每个 DB2 模式只能使用一个系列的昵称。例如，您具有两个具有父子关系的数据银行。这两个数据银行是 SWISSPROT 和 SPFEAT。这两个数据银行组成一个系列。如果对 CREATE NICKNAME 语句使用缺省语法，则将把一个昵称（SWISSPROT）用于 SWISSPROT 数据银行，而将另一个昵称（SPFEAT）用于 SPFEAT 数据银行。要在模式中使 SWISSPROT 具有多个昵称，则必须使用 REMOTE\_OBJECT 选项。

REMOTE\_OBJECT 昵称选项指定与昵称相关联的 BioRS 数据银行的名称。您在 REMOTE\_OBJECT 选项中指定的名称确定模式和昵称的 BioRS 数据银行。REMOTE\_OBJECT 选项还指定昵称与其它昵称的关系。

以下示例显示与上一个示例相同的一组昵称特征，但是更改了昵称名称，并使用 REMOTE\_OBJECT 选项来指定正在为其定义昵称的 BioRS 数据银行：

```
CREATE NICKNAME NewSP
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR(128),
  ENTRYDATE VARCHAR (64))
FOR SERVER brs_server
  OPTIONS (REMOTE_OBJECT 'SwissProt');
```

对于您想为其创建昵称的每个 BioRS 数据银行重复此步骤。

在这一系列任务中不需要再执行其它步骤。

#### 相关概念:

- 第 80 页的『BioRS 统计信息』

#### 相关任务:

- 第 81 页的『更新 BioRS 昵称基数统计信息』
- 第 58 页的『为非关系数据源指定昵称列』

#### 相关参考:

- 第 74 页的『BioRS AllText 元素』
- 第 69 页的『CREATE NICKNAME 语句 - BioRS 包装器的示例』
- 第 496 页的『CREATE NICKNAME 语句语法 - BioRS 包装器』

## CREATE NICKNAME 语句 - BioRS 包装器的示例

本主题提供了示例来说明如何使用 CREATE NICKNAME 语句来为 BioRS 包装器注册昵称。

### 示例 1:

以下示例说明如何为不符合 DB2 Information Integrator 语法的远程 BioRS 数据银行创建昵称:

```
CREATE NICKNAME SwissFT
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR (128),
  ENTRYDATE VARCHAR (64),
  FtLength VARCHAR (16),
  FOR SERVER biors1
  OPTIONS (REMOTE_OBJECT 'SwissProt.Features');
```

此昵称的名称是 SwissFT。表列是 ID、ALLTEXT、ENTRYDATE 和 FtLength。对 ID 列指定了 ELEMENT\_NAME 列选项。当 BioRS 元素的名称不符合列名的有效 DB2 联合语法时，必须指定 ELEMENT\_NAME 选项。在此示例中，BioRS 元素 \_ID\_ 符合 DB2 联合语法，但 \_ID\_ 对于 DB2 Information Integrator 用户而言是有可能令人混淆的名称。名称 ID 简单易懂。通常，在下列情况下使用 ELEMENT\_NAME 选项:

- 当 BioRS 元素名称不符合有效的 DB2 联合语法时
- 当 BioRS 元素名称的区分大小写性不符合所建立的 DB2 联合系统标准时
- 当 BioRS 元素名称对 DB2 Information Integrator 用户可能不明显时

另外，REMOTE\_OBJECT 选项用于指定与昵称等同的 BioRS 数据银行名。当 BioRS 数据银行的名称不符合有效的 DB2 联合语法时，必须指定 REMOTE\_OBJECT 选项。在此示例中，数据银行名“SwissProt.Features”不符合有效的 DB2 联合语法。通常，在下列情况下使用 REMOTE\_OBJECT 选项:

- 当 BioRS 数据银行名的区分大小写性不符合所建立的 DB2 联合系统标准时
- 当 BioRS 数据银行名不符合有效的 DB2 联合语法时
- 当 BioRS 数据银行名对 DB2 Information Integrator 用户可能不明显时

### 示例 2:

以下示例说明如何为使用链接到另一个 BioRS 数据银行的 BioRS 数据银行的表创建昵称:

```
CREATE NICKNAME SwissFT2
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR (1200),
  FtKey VARCHAR (32),
  FtLength VARCHAR (64),
  FtDescription VARCHAR (128),
  Parent VARCHAR (32) OPTIONS (REFERENCED_OBJECT 'SwissProt'))
  FOR SERVER biors1
  OPTIONS (REMOTE_OBJECT 'SwissProt.Features');
```

此昵称的名称是 SwissFT2。表列是 ID、ALLTEXT、FtKey、FtLength、FtDescription 和 Parent。对 ID 列指定了 ELEMENT\_NAME 列选项。REMOTE\_OBJECT 选项用于指定与昵称相应的 BioRS 数据银行名。

另外，Parent 列使用 REFERENCED\_OBJECT 选项。必须对与 BioRS 引用数据类型元素相应的列指定此选项。REFERENCED\_OBJECT 选项指定列所引用的 BioRS 数据银行的名称。在此情况下，Parent 元素引用 BioRS SwissProt 数据银行。

**相关任务:**

- 第 67 页的『为 BioRS 数据源注册昵称』

**相关参考:**

- 第 496 页的『CREATE NICKNAME 语句语法 - BioRS 包装器』

---

## BioRS 数据源的查询和定制函数

### 定制函数和 BioRS 查询

联合环境使用两个查询引擎。对于 BioRS 包装器，这些查询引擎是 DB2 通用数据库和 BioRS。可以通过使用 4 个 BioRS 定制函数来指定将谓词下推至 BioRS 引擎，这些定制函数是:

- BIOR.S.CONTAINS
- BIOR.S.CONTAINS\_LE
- BIOR.S.CONTAINS\_GE
- BIOR.S.SEARCH\_TERM

这四个定制函数是在 BioRS 模式中注册的。必须使用 BioRS 模式来引用这些函数。

定制函数 BIOR.S.CONTAINS、BIOR.S.CONTAINS\_LE 和 BIOR.S.CONTAINS\_GE 需要搜索项列自变量和查询文本自变量。以下示例显示了 BIOR.S.CONTAINS 语句:

```
BIOR.S.CONTAINS (<search term column>,<query term>)
```

搜索项列自变量的值必须引用已建立索引的 BioRS 列。使用未建立索引的列将产生错误消息 SQL30090N (“操作对应用程序执行环境无效”)。

查询项自变量的值只能是文字、主变量或列引用。不能使用算术或字符串并置。并且，即使将所使用的搜索项列定义为允许空值，查询项自变量的值也不能为 NULL。

查询项自变量的大小写无关紧要。

查询项自变量的有效数据类型和格式取决于所使用的搜索项列的 BioRS 数据类型。BioRS 定义了 5 种可能的数据类型: 文本、作者、日期、数字和引用。表 14 列示了 BioRS 数据类型和每种数据类型的有效函数查询项。

表 14. BioRS 数据类型和有效的定制函数查询项

搜索项列的数据类型	有效的查询项	格式
文本	VARCHAR() 或 CHAR()	BioRS 文本项，包括通配符。

表 14. BioRS 数据类型和有效的定制函数查询项 (续)

搜索项列的数据类型	有效的查询项	格式
作者	VARCHAR() 或 CHAR()	具有格式 “<last>, <init>” 的 BioRS 作者引用。“<last>” 是作者的姓。“<init>” 是作者姓名的首字母 (不带句点)。可以接受逗号与姓名首字母之间的空格。  另外, 也可以单独指定 <last> 而不带逗号或姓名首字母。
日期	VARCHAR()、CHAR()、DATE 或 TIMESTAMP	如果是字符串, 则具有 DB2 格式日期 yyyy/mm/dd。
数字	VARCHAR() 或 CHAR()、INTEGER、SMALLINT、 BIGINT REAL、DOUBLE 和 DECIMAL	DB2 格式数字。
引用	VARCHAR() 或 CHAR()	BioRS 文本项。

BioRS 数据类型搜索项列与查询项自变量的所有其它组合都将产生错误消息 SQL30090N (“操作对应用程序执行环境无效”)。只可以使用第 70 页的表 14 中显示的组合。

文本、作者和引用数据类型搜索项列的查询项自变量必须与 BioRS 查询语言模式相匹配。在 BioRS 中, 查询项自变量可以包含字母数字字符串和通配符。BIORS.CONTAINS 函数支持两个通配符: ? (问号) 和 \* (星号)。

? 通配符与单个字符相匹配。例如, 谓词 BioRS.CONTAINS (description, 'bacteri?')=1 与项 bacteria 相匹配, 但不与项 bacterial 相匹配。

\* 通配符与零个或多个字符相匹配。例如, 谓词 BioRS.CONTAINS (description, 'bacteri\*')=1 与项 bacteri、bacteria 和 bacterial 相匹配。

有关 BioRS 查询语言模式的详细信息, 请参阅 BioRS 文档。

可以对所有 BioRS 列类型指定 BIORS.CONTAINS 函数。

只能对其下层 BioRS 数据类型是数字或日期的列指定 BIORS.CONTAINS\_GE 和 BIORS.CONTAINS\_LE 定制函数。BIORS.CONTAINS\_GE 函数选择符合以下条件的行: 列包含的值大于或等于由查询项自变量表示的值。BIORS.CONTAINS\_LE 函数选择符合以下条件的行: 列包含的值小于或等于由查询项自变量表示的值。

BIORS.CONTAINS、BIORS.CONTAINS\_GE 和 BIORS.CONTAINS\_LE 函数返回整数结果。当在谓词中使用这三个 CONTAINS 函数的任何一个函数时, 必须使用 = 或 <> 运算符来将返回值与值 1 进行比较。例如:

```
SELECT * FROM s.MySP WHERE BIORS.CONTAINS (s.AllText, 'muscus') = 1;
```

具有 NOT (BioRS.Contains (col,value) = 1) 格式的表达式等同于 BioRS.CONTAINS (col,value) <> 1。

可以通过发出 `BIORS.SEARCH_TERM` 函数来运行采用别的方法不可能进行的查询。可以使用此函数来指定使用 BioRS 格式的搜索项。`BIORS.SEARCH_TERM` 函数需要两个自变量。第一个自变量是对项将要应用于的昵称的 `_ID_` 列的引用。第二个自变量是包含不带数据银行名的项的字符串。

以下示例从 MyEMBL 数据银行中选择 `SeqLength` 元素包含大于或等于 100 的值的条目的所有列。

```
SELECT * FROM MyEMBL s WHERE
  BIORs.SEARCH_TERM (s.ID, '[SeqLength GREATER number:100;]') = 1;
```

以下示例从 Swiss 昵称中选择 `MolWeight` 元素值大于或等于 100368 的 `MolWeight` 列。

```
SELECT s.molweight FROM Swiss s WHERE
  BIORs.SEARCH_TERM (s.ID, '[MolWeight GREATER number:100368;]') = 1;
```

#### 相关概念:

- 『下推分析』（《联合系统指南》）
- 第 79 页的『有关优化 BioRS 包装器性能的准则』
- 第 72 页的『BioRS 包装器的等值连接谓词』

#### 相关任务:

- 第 63 页的『为 BioRS 包装器注册定制函数』

#### 相关参考:

- 第 74 页的『BioRS 包装器 - 示例查询』
- 第 64 页的『定制函数表 - BioRS 包装器』

## BioRS 包装器的等值连接谓词

当使用 BioRS 定制函数时，必须为 BioRS 引擎指定谓词，但是有一个例外情况。在查询期间执行等值连接操作就是这种例外情况。连接操作涉及根据匹配列值来从两个或更多个表检索数据。等值连接是连接条件具有“表达式 = 表达式”格式的连接操作。对于 BioRS 查询，等值连接项必须包含一个数据银行的 `_ID_` 元素和另一个数据银行的“引用”类型元素。

#### 示例:

此示例显示样本昵称定义以及使用样本昵称的等值连接查询。

您想查询两个 BioRS 数据银行: `SwissProt` 和 `SwissProt.features`。`SwissProt.features` 数据银行是 `SwissProt` 数据银行的子数据银行，它包含名为 `Parent` 的元素。`Parent` 元素包含对 `SwissProt` 的 `_ID_` 元素标识的条目的引用。您为两个数据银行注册两个昵称定义。

#### 昵称定义 1:

```
CREATE NICKNAME tc600sprot (
  ID      VARCHAR (32) OPTIONS (ELEMENT_NAME '_ID_'),
  AllText  VARCHAR (128),
  EntryDate  VARCHAR (128),
  Update    VARCHAR (128),
  Description  VARCHAR (1200),
  Crossreference  VARCHAR (32),
  Authors    VARCHAR (256),
  Journal    VARCHAR (256),
```



```

|         JournalIssue  VARCHAR (64) OPTIONS (IS_INDEXED 'N'),
|         PublicationYear VARCHAR (1024),
|         Gene          VARCHAR (20) OPTIONS (IS_INDEXED 'Y'),
|         Remarks       VARCHAR (1200),
|         RemarkType    CHAR (20),
|         CatalyticActivity VARCHAR (20),
|         CoFactor      VARCHAR (64),
|         Disease       VARCHAR (128),
|         Function      VARCHAR (128),
|         Pathway       VARCHAR (128),
|         Similarity    VARCHAR (128),
|         Complex       VARCHAR (64),
|         FtKey         VARCHAR (32),
|         FtDescription VARCHAR (128),
|         FtLength      VARCHAR (256),
|         MolWeight     VARCHAR (64),
|         ProteinLen    VARCHAR (32) OPTIONS (ELEMENT_NAME 'Protein_length'),
|         Sequence      CLOB,
|         AccNumber     VARCHAR (32),
|         Taxonomy      VARCHAR (128),
|         Organelle     VARCHAR (128),
|         Organism      VARCHAR (128),
|         Keywords      VARCHAR (1200),
|         Localization  VARCHAR (128),
|         FtKey_count   VARCHAR (32)) FOR SERVER biors_server_600
|         OPTIONS (REMOTE_OBJECT 'SwissProt');

```

昵称定义 2:

```

| CREATE NICKNAME tc600feat (
|     ID      VARCHAR (32) OPTIONS (ELEMENT_NAME '_ID_'),
|     AllText VARCHAR (1200),
|     FtKey   VARCHAR (32),
|     FtLength VARCHAR (64),
|     FtDescription VARCHAR (128),
|     Parent VARCHAR (32) OPTIONS (REFERENCED_OBJECT 'SwissProt'))
|     FOR SERVER biors_server_600 OPTIONS (REMOTE_OBJECT 'SwissProt.features');

```

以下查询在等值连接中引用了这两个昵称:

```

| SELECT s.ID, f.ID, f.FtKey FROM tc600sprot s, tc600feat f
|     WHERE BioRS.CONTAINS (s.AllText, 'anopheles') = 1
|           AND BioRS.CONTAINS (s.PublicationYear, 1997) = 1
|           AND BioRS.CONTAINS (f.FtKey, 'signal') = 1
|           AND f.Parent = s.ID;

```

在上一个查询中，将两个谓词应用于 tc600sprot 昵称 (SwissProt 数据银行)。这两个谓词对包含 anopheles 项并且发布年份为 1997 的行进行过滤。一个谓词应用于 tc600feat 昵称 (SwissProt.features 数据银行)，它对 FtKey 元素包含 signal 项的那些行进行过滤。这两个昵称是使用 f.Parent = s.ID 项进行连接的。

最终结果集只包含满足这些条件并且 features 条目引用 SwissProt 数据银行中的匹配条目的行。

**相关概念:**

- 第 79 页的『有关优化 BioRS 包装器性能的准则』

**相关参考:**

- 第 70 页的『定制函数和 BioRS 查询』
- 第 74 页的『BioRS 包装器 - 示例查询』

## BioRS AllText 元素

BioRS 系统中的每个数据银行都包含一个称为 AllText 的元素。AllText 元素是 BioRS 自动为所有数据银行创建的已建立索引的元素。

通过使用 AllText 元素，可以对条目中的所有文本执行搜索，而不仅仅是对特定的已建立索引的元素进行搜索。例如，对项 `muscus` 执行的搜索可以返回标题、摘要、描述或有机体中出现了 `muscus` 一词的条目。

要在 DB2 Information Integrator 查询中使用 AllText 元素，必须将 AllText 元素映射到昵称列。当在 CREATE NICKNAME 语句中指定列时，将 AllText 元素映射至昵称列。映射至 AllText 元素的昵称列将在 SELECT 语句中返回 NULL 值。当将一列指定为 AllText 元素时，该列一定不能是在 CREATE NICKNAME 语句中声明的第一列。

在 AllText 元素正确地映射到昵称列之后，就可以在 CONTAINS 定制函数调用中使用该昵称列。

### 相关任务:

- 第 67 页的『为 BioRS 数据源注册昵称』

### 相关参考:

- 第 74 页的『BioRS 包装器 - 示例查询』

## BioRS 包装器 - 示例查询

本主题提供了数个使用昵称 `swiss` 和 `swissft` 的样本查询。

昵称 `swiss` 是使用以下 CREATE NICKNAME 语句注册的:

```
CREATE NICKNAME swiss
(
  ID                CHAR (30) OPTIONS (ELEMENT_NAME '_ID_'),
  EntryDate         VARCHAR (15),
  Update            CLOB (15),
  Description        CLOB (15),
  Crossreference     CLOB (15),
  Authors            CLOB (15),
  Journal            VARCHAR (15),
  JournalIssue       VARCHAR (15),
  PublicationYear    CLOB (15),
  PublicationTitle   CLOB (15),
  Gene               CLOB (15),
  Remarks            CLOB (15),
  RemarkType         VARCHAR (15),
  CatalyticActivity VARCHAR (15),
  CoFactor           VARCHAR (15),
  Disease            VARCHAR (15),
  Function           CLOB (15),
  Pathway            VARCHAR (15),
  Similarity         CLOB (15),
  Complex            VARCHAR (15),
  FtKey              VARCHAR (15),
  FtDescription      CLOB (15),
  FtLength           VARCHAR (15),
  MolWeight          CHAR (15),
  Protein_Length     VARCHAR (15),
  Sequence           CLOB (15),
  AccNumber          VARCHAR (15),
  Taxonomy           CLOB (15),
```

```

Organelle          VARCHAR (15),
Organism           VARCHAR (15),
Keywords           VARCHAR (15),
Localization       VARCHAR (15),
FtKey_count        VARCHAR (15),
AllText            CLOB (15)
)
)
FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'swissprot');

```

昵称 swissft 是使用以下 CREATE NICKNAME 语句注册的:

```

CREATE NICKNAME swissft
(
  ID              VARCHAR (30) OPTIONS (ELEMENT_NAME '_ID_'),
  FtKey           VARCHAR (15),
  FtLength        VARCHAR (15),
  FtDescription   VARCHAR (15),
  Parent          VARCHAR (30) OPTIONS (REFERENCED_OBJECT 'swissprot'),
  AllText         CLOB (15)
)
FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'swissprot.features');

```

表 15 中的查询和结果举例说明了可以如何构建查询以优化联合系统与 BioRS 服务器之间的工作负载。

表 15. 生成完全相同的结果的不同查询的样本

查询	结果
<pre>select s.id from Swiss s where biors.CONTAINS(s.id, '100K_RAT') = 1 fetch first 3 rows only</pre>	<pre>ID ----- 100K_RAT  1 record(s) selected.</pre>
<pre>select s.id from Swiss s where s.id LIKE '%100K_RAT%' fetch first 3 rows only</pre>	<pre>ID ----- 100K_RAT  1 record(s) selected.</pre>

表 15 中的两个查询产生相同的结果。但是，第一个查询的运行速度比第二个查询快得多。第一个查询使用 BIORS.CONTAINS 函数来指定输入谓词。因此，BioRS 选择 swissprot 数据银行中的数据并接着将所选数据传递到 DB2 Information Integrator。在第二个查询中，直接对 Swiss 昵称指定输入谓词 LIKE。因此，BioRS 将整个 swissprot 数据银行传输到 DB2 Information Integrator。在传输数据银行内容之后，DB2 Information Integrator 接着选择数据。

第 76 页的表 16 中的查询和结果显示了在 BIORS.CONTAINS 函数中使用通配符。尽管使用了不同的通配符，但是第 76 页的表 16 中的所有查询结果都是完全相同的。

表 16. 在 *BIORS.CONTAINS* 函数中使用通配符的样本查询

查询	结果
<pre>select s.crossreference from Swiss s where biors.CONTAINS(s.crossreference, 'MEDLINE') = 1 fetch first 3 rows only</pre>	<pre>CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081  3 record(s) selected.</pre>
<pre>select s.crossreference from Swiss s where biors.CONTAINS(s.crossreference, '?ED?IN?') = 1 fetch first 3 rows only</pre>	<pre>CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081  3 record(s) selected.</pre>
<pre>select s.crossreference from Swiss s where biors.CONTAINS(s.crossreference, '*D*N*') = 1 fetch first 3 rows only</pre>	<pre>CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081  3 record(s) selected</pre>

表 17 中的查询和结果显示了可以如何使用 *BIORS.CONTAINS* 函数来存取 *BioRS* 作者数据类型元素中的信息。

表 17 中的所有查询的语法几乎完全相同。唯一的区别是查询项中存在或不存在的名字首字母以及名字与姓氏首字母之间的空格数目。

表 17. 存取 *BioRS* 作者数据类型列的样本查询

查询	结果
<pre>select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller') = 1 fetch first 3 rows only</pre>	<pre>AUTHORS ----- Mueller D. Rehb Mayer K.F.X. Sc Zemmour J. Litt  3 record(s) selected.</pre>
<pre>select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller,D') = 1 fetch first 3 rows only</pre>	<pre>AUTHORS -----  0 record(s) selected.</pre>
<pre>select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller ,D') = 1 fetch first 3 rows only</pre>	<pre>AUTHORS -----  0 record(s) selected.</pre>
<pre>select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller, D') = 1 fetch first 3 rows only</pre>	<pre>AUTHORS ----- Mueller D. Rehb Zou P.J. Borovo Davies J.D. Mue  3 record(s) selected.</pre>

表 18 中的查询和结果举例说明可以如何使用 `BIORS.CONTAINS` 函数来存取 BioRS 日期类型元素中的信息。

当 BioRS 日期类型字段包含日期序列时，结果可以包含额外的信息，如表 18 的第二个示例所示。BioRS 数字数据类型元素（日期和数字）可以包含多个值。因此，对 BioRS 日期或数字元素运行的查询的结果也可以包含多个值。如果包含多个值，则始终用空格将值分隔开。

表 18. 存取 BioRS 日期数据类型列的样本查询

查询	结果
<pre>select e.entrydate from embl e where biors.CONTAINS(e.entrydate, date('11/01/1997')) = 1 fetch first 3 rows only</pre>	<pre>ENTRYDATE ----- 01-NOV-1997 01-NOV-1997 01-NOV-1997  3 record(s) selected.</pre>
<pre>select g.update from gen g where biors.CONTAINS(g.update, date('11/01/1997')) = 1 fetch first 3 rows only</pre>	<pre>UPDATE ----- 01-NOV-1997 11- 01-NOV-1997 12- 01-NOV-1997 06-  3 record(s) selected.</pre>

表 19 中的查询和结果显示了可以如何使用 `BIORS.CONTAINS_LE` 和 `BIORS.CONTAINS_GE` 函数。

表 19. 使用 `BIORS.CONTAINS_LE` 和 `BIORS.CONTAINS_GE` 函数的样本查询

查询	结果
<pre>select s.molweight from Swiss s where biors.CONTAINS_LE(s.molweight, 100368) = 1 fetch first 3 rows only</pre>	<pre>MOLWEIGHT ----- 100368 10576 8523  3 record(s) selected.</pre>
<pre>select s.molweight from Swiss s where biors.CONTAINS_GE(s.molweight, 100368) = 1 fetch first 3 rows only</pre>	<pre>MOLWEIGHT ----- 100368 103625 132801  3 record(s) selected.</pre>
<pre>select s.journalissue from Swiss s where biors.CONTAINS_GE(s.journalissue, 172) = 1 fetch first 3 rows only</pre>	<pre>JOURNALISSUE ----- 172 21 242 196  3 record(s) selected.</pre>

第 78 页的表 20 中的查询和结果显示了可以如何使用 `BIORS.SEARCH_TERM` 函数来指定使用 BioRS 格式搜索项。

表 20. 使用 `BIORS.SEARCH_TERM` 函数的样本查询

查询	结果
<pre>select s.publicationyear from Swiss s where biors.SEARCH_TERM (s.id, '[PublicationYear EQ number:1997;]')=1 fetch first 10 rows only</pre>	<pre>PUBLICATIONYEAR ----- 1997 1997 2000 1988 1991 1997 1994 1997 1997 1998 1994 1995 1997 1997 1999 1997 1994 1994 1995 1993 1992 1997  10 record(s) selected.</pre>
<pre>select s.molweight from Swiss s where biors.SEARCH_TERM (s.id, '[MolWeight EQ number:100368;]') = 1 fetch first 10 rows only</pre>	<pre>MOLWEIGHT ----- 100368 100368  2 record(s) selected.</pre>
<pre>select s.molweight from Swiss s where biors.SEARCH_TERM (s.id, '[MolWeight GREATER number:100368;]') = 1 fetch first 10 rows only</pre>	<pre>MOLWEIGHT ----- 100368 103625 132801 194328 130277 287022 289130 135502 112715 112599  10 record(s) selected.</pre>

以下查询显示如何使用关系谓词来构成具有父子关系的两个数据银行之间的等值连接:

```
select s.id, f.id, f.parent from Swiss s, Swissft f
where (f.parent = s.id) fetch first 10 rows only
```

查询结果如下:

ID	ID	PARENT
100K_RAT	100K_RAT.1	swissprot:100K_RAT
100K_RAT	100K_RAT.2	swissprot:100K_RAT
100K_RAT	100K_RAT.3	swissprot:100K_RAT
100K_RAT	100K_RAT.4	swissprot:100K_RAT
100K_RAT	100K_RAT.5	swissprot:100K_RAT
100K_RAT	100K_RAT.6	swissprot:100K_RAT
100K_RAT	100K_RAT.7	swissprot:100K_RAT
100K_RAT	100K_RAT.8	swissprot:100K_RAT
100K_RAT	100K_RAT.9	swissprot:100K_RAT
104K_THEPA	104K_THEPA.1	swissprot:104K_THEPA

10 record(s) selected.

在前面的查询结果中，100K\_RAT 记录是 9 个子记录（100K\_RAT.1 到 100K\_RAT.9）的父记录。

#### 相关概念:

- 第 79 页的『有关优化 BioRS 包装器性能的准则』
- 第 72 页的『BioRS 包装器的等值连接谓词』

#### 相关参考:

- 第 70 页的『定制函数和 BioRS 查询』
- 第 69 页的『CREATE NICKNAME 语句 - BioRS 包装器的示例』
- 第 496 页的『CREATE NICKNAME 语句语法 - BioRS 包装器』

---

## 优化 BioRS 包装器性能

### 有关优化 BioRS 包装器性能的准则

本主题提供有关使用 BioRS 包装器时如何优化查询性能的准则。

#### 最大程度地减少在搜索引擎之间传输的数据量。

联合环境使用两个查询引擎。对于 BioRS 包装器，这些查询引擎是 DB2<sup>®</sup> 通用数据库和 BioRS。DB2 引擎处理对昵称列指定的谓词（关系运算符，例如，=、BETWEEN、LIKE 和 <>）。BioRS 引擎处理使用 BioRS 包装器的 4 个定制函数指定的谓词。

要最大程度地减少在两个搜索引擎之间传输的数据量，请将查询构建成为尽可能将数据处理下推到 BioRS 系统。

如果需要在查询中执行连接操作，则请利用 BioRS 数据银行中已经存在的任何父子关系并尽可能地执行等值连接操作。等值连接操作是在 BioRS 中处理的，这还最大程度地减少了在 DB2 和 BioRS 查询引擎之间传输的数据量。

**注意：**不要中断对 BioRS 进行的 DB2 Information Integrator 查询（例如，在命令行处理器中使用 **Ctrl-D** 或 **Ctrl-Z**，或停止应用程序）。将查询中断会导致“死亡”进程继续在 BioRS 服务器上运行。这些“死亡”进程将迅速地降低 BioRS 和 DB2 Information Integrator 系统性能。如果有足够的此类“死亡”进程在运行的话，在 DB2 Information Integrator 查询处理期间就会发生意外的错误。例如，当预期会返回行的时候，有效的查询可能返回 0 行。在极端的情况下，BioRS、DB2 Information Integrator 或两个产品都会停止或异常结束。

#### 在联合环境中维护 BioRS 统计信息。

在联合系统中，联合数据库依靠带有昵称的对象的目录统计信息来优化查询处理。维护关于 BioRS 数据源的当前统计信息对于优化 BioRS 包装器性能而言是必需的。如果定义昵称所基于的远程对象的统计数据或结构特征已更改，则必须在联合系统中更新相应的昵称列基数统计信息。

要优化 BioRS 包装器性能，请在 DB2 Information Integrator 中定期执行这些更新。

#### 相关概念:

- 『调整查询处理』（《联合系统指南》）
- 第 72 页的『BioRS 包装器的等值连接谓词』

- 第 80 页的『BioRS 统计信息』

#### 相关参考:

- 第 70 页的『定制函数和 BioRS 查询』
- 第 74 页的『BioRS 包装器 - 示例查询』

## BioRS 统计信息

在联合系统中，联合数据库依靠带有昵称的对象的目录统计信息来优化查询处理。这些统计信息是使用 CREATE NICKNAME 语句创建昵称时从 BioRS 数据源中检索到的。联合数据库验证数据源中的对象是否存在，然后尝试收集现有数据源的统计数据。信息是从数据源目录中读取的，并且放入联合服务器上的 DB2® 联合数据库系统目录。

对于 BioRS 数据源，关键的统计信息包括:

- 昵称的基数。对于 BioRS 数据源，昵称基数等同于相应的 BioRS 数据银行中的条目数目。
- 与 BioRS \_ID\_ 元素相应的列的基数。此列的基数必须与引用列的昵称的基数相匹配。
- BioRS 包装器可能需要使用的所有列的基数。

必须维护关于 BioRS 数据源的当前统计信息以优化 BioRS 包装器的性能。如果定义昵称所基于的远程对象的统计数据或结构特征发生更改，则必须在联合系统中更新相应的基数统计信息。基数统计信息存储在 SYSSTAT.TABLES 目录视图和 SYSSTAT.COLUMNS 目录视图中。

执行下列任务来在联合系统中维护 BioRS 基数统计信息:

1. 确定必需的昵称的基数统计信息（如果有必要的话）。
2. 在必需的目录视图中适当地更新基数统计信息。

#### 相关概念:

- 『调整查询处理』（《联合系统指南》）

#### 相关任务:

- 第 80 页的『确定 BioRS 数据银行基数统计信息』
- 第 81 页的『更新 BioRS 昵称基数统计信息』
- 第 81 页的『更新 BioRS 列基数统计信息』
- 第 83 页的『更新 BioRS \_ID\_ 列基数』

## 确定 BioRS 数据银行基数统计信息

在可以更新昵称统计信息或更新与 BioRS \_ID\_ 元素相应的列的基数之前，必须确定 BioRS 数据银行基数统计信息。

#### 过程:

要确定 BioRS 中的特定数据银行的基数统计信息，请使用 BioRS 实用程序 admin\_find 或 www\_find.cgi。指定 -c（基数）选项。有关这两个 BioRS 实用程序的更多信息，请参阅 BioRS 文档。



|

| 相关概念:

- 第 80 页的『BioRS 统计信息』

|

| 相关任务:

- 第 81 页的『更新 BioRS 昵称基数统计信息』
- 第 81 页的『更新 BioRS 列基数统计信息』
- 第 83 页的『更新 BioRS \_ID\_ 列基数』

|

## 更新 BioRS 昵称基数统计信息

|

| 当要为其创建昵称的 BioRS 数据银行的内容发生显著更改时，必须更新 BioRS 昵称基数统计信息。通过维护昵称的正确基数统计信息，优化器和 BioRS 包装器就能够选择性能最佳的数据存取方案。

|

| 要更新 BioRS 昵称基数统计信息，您用正确的基数数目修改 SYSSTAT.TABLES 目录视图。

|

| 先决条件:

|

| 必须确定与要更新其统计信息的昵称相应的 BioRS 数据银行的基数数目。

|

| 过程:

|

| 使用以下语法来发出 UPDATE 语句:

|

```
UPDATE sysstat.tables SET card=cardinality
      WHERE tabschema=nickname-schema
      AND tablename=nickname-name;
```

- *cardinality* 是与要更新其统计信息的昵称相应的 BioRS 数据银行的基数数目。
- *nickname-schema* 是与要更新其统计信息的昵称相关联的模式名称。
- *nickname-name* 是要更新其统计信息的昵称名称。

|

| 相关概念:

- 第 80 页的『BioRS 统计信息』

|

| 相关任务:

- 第 80 页的『确定 BioRS 数据银行基数统计信息』
- 第 81 页的『更新 BioRS 列基数统计信息』
- 第 83 页的『更新 BioRS \_ID\_ 列基数』

|

## 更新 BioRS 列基数统计信息

|

| 要更新联合系统中的 BioRS 列基数统计信息，必须修改 SYSSTAT.COLUMNS 目录视图。

|

| 通过维护 BioRS 列的正确基数统计信息，优化器和 BioRS 包装器在处理查询期间就能够选择性能最佳的数据存取方案。

|

| 您可以可选地更新 BioRS 列基数统计信息来作为将 BioRS 添加至联合系统这一大型任务的一部分。当您想要改进 BioRS 数据源的查询性能时，也可以更新 BioRS 列基数统计信息。

## 限制:

不要使用此过程来更新与 BioRS \_ID\_ 元素相应的列的基数统计信息。必须使用另一个过程来更新与 BioRS \_ID\_ 元素相应的列的基数统计信息。

## 过程:

要更新 BioRS 列基数统计信息, 请使用以下语法来发出 UPDATE 语句:

```
UPDATE sysstat.columns SET colcard=(SELECT COUNT(DISTINCT column-name)
FROM nickname-schema.nickname-name)
WHERE
  tabschema=nickname-schema
  AND tablename=nickname-name
  AND colname=column-name;
```

- *column-name* 是要更新其基数统计信息的列的名称。
- *nickname-schema* 是与使用了指定的列的昵称相关联的模式名称。
- *nickname-name* 是使用了指定的列的昵称的名称。

因为必须检索昵称中指定的数据银行的所有条目, 所以此查询可能需要运行几分钟。

如果列可以包含多个值 (例如, 具有 SwissProt 数据库格式的 PublicationYear 元素), 则计算就会变得太复杂而导致无法使用 SQL 查询。对于这样的列, 必须手工计算基数, 然后更新 SYSSTAT.COLUMNS 目录视图。要计算基数, 请将列中的相异值的数目除以每行的值的平均数目。计算得出的基数不能大于表的基数。

## 示例:

假定某个昵称带有三行。这三行的 PublicationYear 列的值为:

- 1997 1992 1985
- 1997 1992 1982
- 1992 1991 1990 1976 1974 1971

共有 9 个相异值, 各行中的值的平均数目是 4。这个 PublicationYear 列的基数为 9/4 或 3 (2.25 舍入到下一个最大整数)。在计算了基数之后, 就可以使用以下 UPDATE 语句来更新 SYSSTAT.COLUMNS 目录视图:

```
UPDATE sysstat.columns SET colcard=3
WHERE
  tabschema=nickname-schema
  AND tablename=nickname-name
  AND colname=column-name
```

- 3 是列基数。
- *nickname-schema* 是与使用了指定的列的底层昵称相关联的模式名称。
- *nickname-name* 是使用了指定的列的底层昵称的名称。
- *column-name* 是要更新其基数统计信息的列的名称。

## 相关概念:

- 第 80 页的『BioRS 统计信息』

## 相关任务:

- 第 81 页的『更新 BioRS 昵称基数统计信息』
- 第 83 页的『更新 BioRS \_ID\_ 列基数』

## 更新 BioRS \_ID\_ 列基数

通过维护映射至 BioRS \_ID\_ 元素的列的正确基数统计信息，优化器和 BioRS 包装器就能够选择性能最佳的数据存取方案。

要更新映射到 BioRS \_ID\_ 元素的列的基数数目，必须修改 SYSSTAT.COLUMNS 目录视图。

### 先决条件:

必须确定与引用了列的昵称相应的 BioRS 数据银行的基数数目。映射到 BioRS \_ID\_ 元素的列的基数数目必须与引用列的昵称的基数相匹配。

### 过程:

要更新 BioRS \_ID\_ 列基数统计信息，请使用以下语法来发出 UPDATE 语句:

```
UPDATE sysstat.columns SET colcard=<cardinality>
WHERE
    tabschema=nickname-schema
    AND tabname=nickname-name
    AND colname IN (SELECT colname FROM syscat.coloptions
        WHERE
            tabschema=nickname-name
            AND tabname=nickname-name
            AND option='ELEMENT_NAME';
            AND setting='_ID_')
```

- *cardinality* 是与列的昵称相对应的 BioRS 数据银行的基数数目。
- *nickname-schema* 是与列的昵称相关联的模式的名称。
- *nickname-name* 是使用了列的昵称的名称。

### 相关概念:

- 第 80 页的『BioRS 统计信息』

### 相关任务:

- 第 80 页的『确定 BioRS 数据银行基数统计信息』
- 第 81 页的『更新 BioRS 昵称基数统计信息』
- 第 81 页的『更新 BioRS 列基数统计信息』

---

## BioRS 包装器的消息

本主题说明在使用 BioRS 的包装器时可能会接收到的消息。

表 21. 由 BioRS 的包装器发出的消息

错误代码	消息	解释
SQL0604N	列的长度、精度或小数位属性、单值类型、结构化类型、结构化类型的属性、函数或类型映射 <data-item>	昵称列的数据类型与下层数据银行元素的 BioRS 类型不兼容。检查 CREATE NICKNAME 语句中该列的数据类型。无效。

表 21. 由 BioRS 的包装器发出的消息 (续)

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “创建包装器对象时出错。”)	创建新的包装器对象时出错。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “BioRS <trace_point>。”)	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “内存分配失败: <trace_point>。”)	分配内存时出错。确保联合服务器主机有足够的内存可用并再次提交查询。如果问题仍存在, 请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “sqlno_crule_save_plans[100]: rc(-214272209) 方案列表是空的。”)	优化器程序和 BioRS 包装器未能在运行查询的方案上达成一致。简化查询并再次运行它。
SQL0401N	运算 “=” 的操作数的数据类型不兼容。	由于定制函数谓词右边的表达式必须是整数值, 所以该查询无效。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “”。相关联的文本和标记是 “找不到数据银行”。	在 BioRS 服务器上找不到 CREATE NICKNAME 语句中引用的 BioRS 数据银行。检查 CREATE NICKNAME 语句并确保所引用的数据银行的名称是正确的。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “”。相关联的文本和标记是 “连接已超时”。	BioRS 服务器未能在 TIMEOUT 选项指定的时间段内响应通信请求。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “<trace_point>”。相关联的文本和标记是 “从服务器中读取时出错”。	从 BioRS 服务器读取数据时发生通信错误。<trace_point> 错误代码的值可能提供了有关该错误的更多信息。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “<trace_point>”。相关联的文本和标记是 “找不到主机”。	找不到在 HOST 服务器选项中标识的 BioRS 服务器主机。检查 CREATE SERVER 语句并确保 HOST 服务器选项值是正确的。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “<trace_point>”。相关联的文本和标记是 “未能连接至服务器。”	包装器无法连接至由 HOST 服务器选项标识的服务器。<trace_point> 错误代码的值可能提供了有关该错误的更多信息。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “<trace_point>”。相关联的文本和标记是 “未能创建 TCPIP 套接字”。	包装器未能创建 TCPIP 套接字。<trace_point> 错误代码的值可能提供了有关该错误代码的更多信息。
SQL1822N	从数据源 “BioRS 包装器” 接收到意外的错误代码 “<trace_point>”。相关联的文本和标记是 “发送至服务器时出错”。	包装器未能将请求发送至 BioRS 服务器。<trace_point> 错误代码的值可能提供了有关该错误的更多信息。

表 21. 由 BioRS 的包装器发出的消息 (续)

错误代码	消息	解释
SQL30090N	操作对于应用程序执行环境无效。原因码 = “无法更改服务器对字符大小写的区分”。	不能使用 SQL 语句来更改 CASE_SENSITIVE 服务器选项的值。要更改此选项的值，必须删除服务器。然后，必须使用 CREATE SERVER 语句再次创建服务器并对 CASE_SENSITIVE 选项指定正确的值。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “两个昵称之间存在多个连接。”	由于在任何两个昵称之间只允许一个连接谓词，所以该查询无效。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “函数谓词的右边必须是常量”。	由于定制函数谓词右边的表达式必须是常量，所以该查询无效。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “定制函数的第一个自变量不是一列”。	由于定制函数的第一个自变量必须引用 BioRS 昵称的列，所以该查询无效。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “CONTAINS 函数的第一个自变量未建立索引”。	该查询无效。BIORS.CONTAINS、BIORS.CONTAINS_LE 或 BIOR.S.CONTAINS_GE 函数的第一个自变量中引用的列必须是已建立索引的列。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “<function-name> 函数的第一个自变量的类型错误”。	该查询无效。BIORS.CONTAINS、BIORS.CONTAINS_LE 或 BIOR.S.CONTAINS_GE 函数的第一个自变量中引用的列不具有正确的数据类型。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “SEARCH_TERM 的第一个自变量不是 _ID_ 列”。	该查询无效。SEARCH_TERM 函数的第一个自变量中引用的列没有映射 BioRS _ID_ 元素。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “绑定参数不能为 NULL”。	在 BIOR.S.CONTAINS 函数的第二个自变量中引用的列或主变量值是 NULL。BioRS 包装器无法处理空值。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “无法将值转换为 BioRS 文字”。	已在文字、列或主变量中将值提交给包装器，未能将该值转换为有效的 BioRS 文字。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “无法更改服务器版本”。	不能使用 ALTER SERVER 语句来更改服务器版本。要更改服务器版本，必须删除服务器。然后，必须使用 CREATE SERVER 语句再次创建具有正确版本的服务器。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “<function-name> 函数的第二个自变量的类型错误”。	该查询无效。BIORS.CONTAINS、BIORS.CONTAINS_LE 或 BIOR.S.CONTAINS_GE 函数的第二个自变量中引用的列不具有正确的数据类型。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “昵称没有任何列”。	没有在 CREATE NICKNAME 语句中指定列声明。需要列声明才能创建昵称。

相关概念:

- 『消息介绍』(《消息参考》第 1 卷)

|  
|

相关参考:

- 『SQLSTATE 消息』（《消息参考》第 2 卷）

---

## 第 7 章 配置对 BLAST 数据源的存取

本章说明如何配置联合服务器以存取存储在 BLAST 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 BLAST 数据源的存取。

本章包含下列内容:

- 说明什么是 BLAST
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 BLAST 包装器相关联的错误消息

---

### 什么是 BLAST?

BLAST (局部相似性基本查询工具) 是由“国家生物技术信息中心”(NCBI) 维护的一个实用程序。BLAST 用来扫描核苷酸或氨基酸序列数据库以找到“符合项”。BLAST 符合项包含一个或多个高分值片段对 (HSP)。HSP 是一对序列段, 其比对是局部最大的, 并且其相似性分数超过了某些阈值。NCBI 提供了可执行的 blastall, 用来对支持 BLAST 的数据源 (例如, GenBank 和 SWISS-PROT) 执行 BLAST 搜索。

BLAST 包装器支持所有五种类型的 BLAST 搜索: BLASTn、BLASTp、BLASTx、tBLASTn 和 tBLASTx。在表 22 中描述了这些搜索类型。

表 22. BLAST 包装器支持的 BLAST 搜索类型

BLAST 搜索类型	描述
BLASTn	一种 BLAST 搜索类型, 在该类型的搜索中, 将把核苷酸序列与核苷酸序列数据库的内容进行比较, 以找出有一些区域与原始序列的某些区域相似的那些序列。
BLASTp	一种 BLAST 搜索类型, 在该类型的搜索中, 将把氨基酸序列与氨基酸序列数据库的内容进行比较, 以找出有一些区域与原始序列的某些区域相似的那些序列。
BLASTx	一种 BLAST 搜索类型, 在该类型的搜索中, 将把核苷酸序列与氨基酸序列数据库的内容进行比较, 以找出有一些区域与原始序列的某些区域相似的那些序列。按所有六个读框架转换序列数据库中的序列, 并搜索获得的序列以找出与查询序列的某些区域同源的那些区域。
tBLASTn	一种 BLAST 搜索类型, 在该类型的搜索中, 将把氨基酸序列与核苷酸序列数据库的内容进行比较, 以找出有一些区域与原始序列的某些区域同源的那些序列。按所有六个读框架翻译序列数据库中的序列, 并搜索获得的序列以找出与查询序列的某些区域同源的那些区域。
tBLASTx	一种 BLAST 搜索类型, 在该类型的搜索中, 将把核苷酸序列与核苷酸序列数据库的内容进行比较, 以找出有一些区域与原始序列的某些区域同源的那些序列。在 tBLASTx 搜索中, 按所有六个读框架翻译查询序列和序列数据库, 并将获得的序列与发现的同源区域进行比较。

图 7 说明了 BLAST 如何使用联合系统。

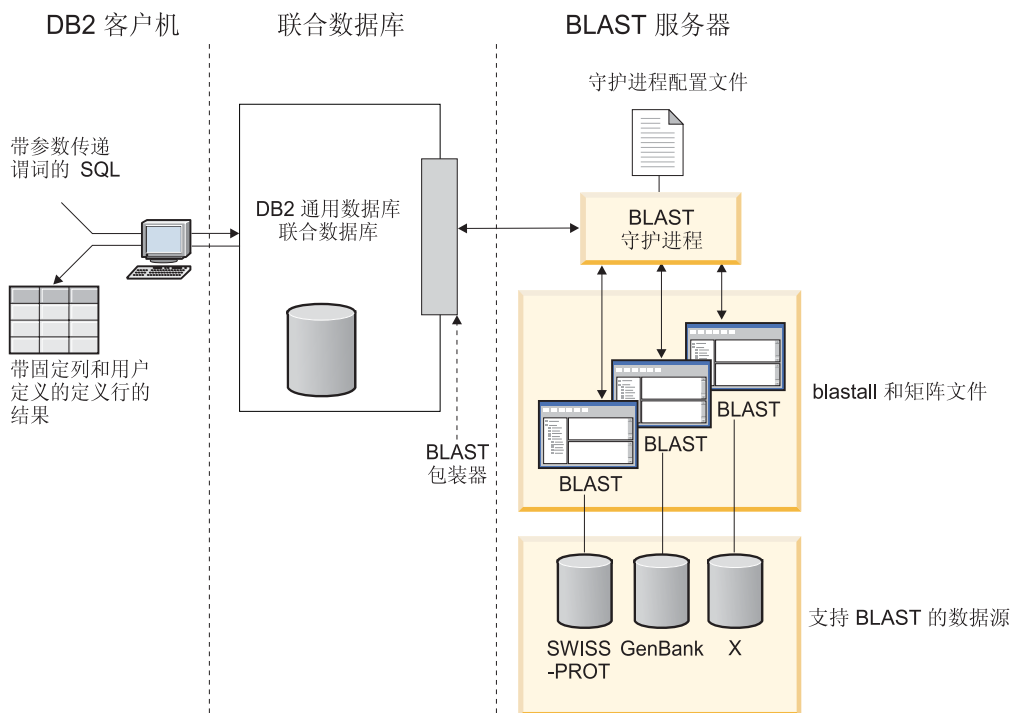


图 7. BLAST 包装器的工作方式

在客户端，用户或应用程序提交的 SQL 语句具有映射至标准 BLAST 选项的特定于 BLAST 的参数传递谓词。具有输入谓词的 SQL 语句被发送至安装了 BLAST 包装器的 DB2® 通用数据库联合数据库系统。

BLAST 包装器将查询转换成 BLAST 应用程序能够理解的格式，并将已变换的查询发送至 BLAST 服务器。此服务器可以是不同于具有联合系统的机器的另一台机器。特殊的守护进程在 BLAST 服务器上运行。此守护进程使用守护进程配置文件中的信息来接收来自联合系统的查询请求，并将查询请求发送至 BLAST 应用程序。然后，按常见方式对支持 BLAST 的数据源运行 BLAST 应用程序。

结果先返回到 BLAST，然后返回到守护进程。守护进程将把检索到的数据返回给 BLAST 包装器。包装器将该数据转换成关系表格式，然后将此表返回给您或应用程序。返回的数据包含两部分：

- BLAST 用户熟悉的一系列标准的固定列以及
- 用户配置的定义行信息。

以下示例说明了如何从支持 BLAST 的数据源中抽取关系信息。数据从原始的 fasta 文件格式移动到支持 BLAST 的数据集，再移动到可以与联合系统中的其它数据源连接的关系表。

第 89 页的图 8 是一个包含四个定义行和核苷酸序列记录的样本 fasta 文件。



```

>7:4986 PMON5744
GTTCTTCCAGTGCCCAAGTCCATTCTGACATCAATGAAGAAGGTAAAATCCCTGCGTGATCCCTCTGCC
AAGATGTGCGAAATCAGACCCGGATAAAGTACCTGCTGTGAGAAATACAGACAGCCCGGAGGAGATCGTGC
AGAAGTTCGCAAGGCTGTGACGGACTTACCTCGGAGGTACCTACGACCCGGCCAGGCAGGAGGCGT
GTCCAACTTGGTGGCCATCCACGCGGCAGTGACCGGACTCCCGGTGGAGGAGGTGGTCCGCCGAAGTGT
GGCATCAACACCGCTGGCTACAAGTTGGTGGTGGCGGAGGCTGTGATTGAGAGATTTGCACCAATTAAGA
GTGAAATGAAAACAGAGGGAACAAGGACCACCTAGAGAAGGTTTTACAAGTTGGGTCGGCAAAAGC
CAAAGAATTAGCATATCCCGTGTGCCAGGAGGTGAAGAAATTTGGTGGGTTTTCTATAGGCAGTCTCACCT
AGTCCAGAAAATGTTTTTATCTTGTGGTCTGCTTGCACACTCAGTCTAATAAAGGCAGCTTTCCTAAG
ACGCCAACAAATCCAGTTTGGGGATGCTTAGTTACT

>8:9747 PMON5699
AAGAAGTCTTGTAGAACTTTCCACCTCCGGCTTCCCCTCCACCTCTTACTGTCCCAACCTTCTGAG
ACGCTTTTTCTCCTCCCGAGGATTTATCTTCTTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCT
TTTTCCCCCGAGGCTGGTTTTGCTTTGGGGAGGGGGGTTTTTTAAAGGGCCGGGGGGCCCCCTTT
CTCCCCCTAATGGGGTTAATTAATAATGGGGGGGGGTTTTTTTTTTAAACCCTATTTGGTCCGG
CCCGGGGATTTCCCCCCCCCCTTGCCTGGTTCCGGGGCCGGAGGAGGGGGGAAAAGGGCGGGAA
CCTTTGGTAGTTCCCTCGGAAAAAATTTTTCGGGGGGAAAACCTCCCT

>13:6512 PMON5498
GATAAGAGGCAGAATAGAAGACTGGACTACTTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCTCT
CTCCACCAAATGGACCGAAACCTTAAAAAGAATCCTACTCTGAAGAAAAGAGGAGGACACATCAA
GAGGTAGAAGGGCGATTTTCATGATATAACAACCCATACCTCCAGAGTGGGAAGCTCCACAGACTGAA
AACTAACTGGTTCACAGAACTCACCTACAGGAGTGAGCCACATCAAACCTCGAATGTGGGGATCTG
GCACTGGTAGAAAAGGCCCTGGAGCATCTGGCATTGAAGGCCAGTGGGGCTGTGTGCAGGAGATCCAC
AGGACTAGGGGAAACGGAGACCCCACTTAAAAGGTGCACACAGACTTTTACGTGCAGTGGGTCCAG
TGCAAAGCAAAGTCTCCATAGGAATCTGGGTCAAACCTGACTGCAGTCTTGGAGGACCTCTGGGAAAG
CAAGGGTGAATGTGGCTTCTTGTGGGAAAGGACATTGGAAGCAAAGCTTTGGGAATATTCATCAGTGT
GC

>15:8924 PMON5426
GGAGAACTGACTCCTGAGCAGCTGCAATTCATGCGGCAGGTGCAGCTCGCCAGTGGCAGAAGACGCTG
CCACAGCGCGGACCCGGAACATCGTGACCGGCTGGGCATCGGGGCGTGGTGTGGCAATTTGTATCC
GTTTGGACTGTAGACTCAGGGAGACCGCATTTAGGGGAACAGGAAGGGCAGCAGGGCGTGTAGGAGGGC
AGTGTGGGGTGGTAGAAGGAGCCGAGATATGAAAACCTTGGCTCCTTTAACTGAAATCAAGCGTTT
GGTGTACCTTACGTTGTCAATTTAAAGGTGATTTTAGTATAATTGATTAATGATTACGGAGTCGGGTGA
GGGCTCCAGGAGCAGACGGCAGAAGATCGAATTTGGGAGGATGATCAGCAGCGGTGGTTGAGCAAGTGT
GGGAAAAGGGAATGCGCACATTCACGTGGTTTCTGAACCCACCTCCCAGATGGTTACACCTTCTACT
CGGTGTCCAGGAGCGTTTCTTGGATGAGCTGGAGGATGAGGCCAAAGCTGCTC

```

图 8. 样本 *fasta* 文件 *nucleo1*

标准的 `formatdb` 应用程序将 *fasta* 文件转换成支持 `BLAST` 的数据集。现在，就可以通过安装并注册了 `BLAST` 包装器的联合系统来对这些数据进行 `SQL` 查询。

由您或应用程序在客户机端发送的以下查询将被 `BLAST` 包装器变换。然后，对支持 `BLAST` 的数据集运行该查询。

```

SELECT Unique_ID, Experiment_Number, Organism_Number, HSP_Info, Score
FROM nucleo1
WHERE BlastSeq = 'ACATTTCTATAGATATTGCTACTCTCCAGGATAGAGTCATCTCT
GGTCTCCAGAGCCACCGCTGGCTACAAGTTGGTGGTGGCGGAGGCTGTGATTGAGAGATTTG
CACCAATACAGAACTCACCTACAGGAGTGAGCGGGTGGTAGAAGGAGCCGAGATATGAAA
ACCTTGTTCAGACCCATTGTCACCGGGG';

```

查询结果由 BLAST 包装器变换成关系表格式并显示在表 23 中。

表 23. 当集成到联合系统中时, BLAST 将以关系表的形式返回结果

唯一标识	实验编号	结构编号	HSP_INFO	分数
PMON5744	4986	7	Identities = 57/201 (28%), Positives = 57/201 (28%), Gaps = 0/201 (0%)	+1.13487000000000E+002
PMON5426	8924	15	Identities = 35/201 (17%), Positives = 35/201 (17%), Gaps = 0/201 (0%)	+6.98754000000000E+001
PMON5498	6512	13	Identities = 26/201 (13%), Positives = 26/201 (13%), Gaps = 0/201 (0%)	+5.20342000000000E+001

数据采用完全关系形式, 可以与实验室使用的来自其它数据源的数据连接起来。在实现联合系统之前, 将几个数据源的结果组合起来可能会导致不容易理解或效率不高。

#### 相关任务:

- 第 90 页的『将 BLAST 数据源添加至联合服务器』

## 将 BLAST 添加至联合服务器

### 将 BLAST 数据源添加至联合服务器

要配置联合服务器以存取 BLAST 数据源, 必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 BLAST 数据源。DB2 控制中心包括一个向导, 可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

#### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

#### 过程:

要将 BLAST 数据源添加至联合服务器:

1. 验证是否安装了正确版本的 blastall 可执行文件和矩阵文件
2. 配置 BLAST 守护进程。
3. 启动 BLAST 守护进程。
4. 注册包装器。
5. 注册服务器定义。
6. 为 BLAST 搜索注册昵称。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 91 页的『验证是否安装了正确版本的 blastall 可执行文件和矩阵文件』
- 第 91 页的『配置 BLAST 守护进程』
- 第 95 页的『启动 BLAST 守护进程』
- 第 97 页的『注册 BLAST 包装器』
- 第 98 页的『为 BLAST 数据源注册服务器』
- 第 98 页的『为 BLAST 数据源注册昵称』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 验证是否安装了正确版本的 blastall 可执行文件和矩阵文件

验证是否安装了正确版本的 blastall 可执行文件和矩阵文件是将 BLAST 添加至联合系统这一大型任务的一部分。

#### 先决条件:

验证是否在 BLAST 服务器上安装了最新版本的 blastall 可执行文件和 BLOSUM62、BLOSUM80、PAM30 和 PAM70 矩阵文件。如果您没有 blastall 可执行文件的最新版本,则必须安装二进制文件和矩阵文件。矩阵文件必须与 blastall 可执行文件位于同一目录中。

#### 过程:

要检查 blastall 可执行文件和矩阵文件的版本级别:

1. 从命令行运行 BLAST 搜索,并记下位于输出文件中的版本号。
2. 检查此产品的 Web 站点以获取已经使用此包装器测试的 BLAST 的版本,以便确保您具有受支持的版本。

在这一系列任务中的下一个任务是配置 BLAST 守护进程。

#### 相关任务:

- 第 91 页的『配置 BLAST 守护进程』

## 配置 BLAST 守护进程

配置 BLAST 守护进程是将 BLAST 添加至联合系统这一大型任务的一部分。

BLAST 包装器需要 BLAST 守护进程。BLAST 守护进程必须在可以从联合系统中通过 TCP/IP 访问的服务器上运行。此服务器可以是作为联合服务器运行的服务器，也可以是一个独立的 BLAST 服务器。

守护进程将与包装器和联合数据库分别运行。守护进程从包装器中侦听 BLAST 作业请求。

#### 先决条件:

BLAST 守护进程必须:

- 对 blastall 二进制文件具有执行存取权，以便它可以运行 BLAST 搜索。
- 对于它可以写入临时文件的目录具有写存取权。
- 至少对一个可以运行 BLAST 搜索的支持 BLAST 的数据源具有读存取权。blastall 可执行文件必须对数据文件和由 formatdb 程序生成的 BLAST 索引文件都具有读存取权。

#### 限制:

如果可执行文件路径或者数据库路径包含空格，则 BLAST 守护进程可能不能正确运行。例如，不能在 Windows 服务器上的 C:\Program Files 中安装 BLAST 可执行文件。

#### 过程:

要配置 BLAST 守护进程:

1. 确保 BLAST 守护进程可执行文件位于正确的服务器上。在安装 DB2 Information Integrator 期间，守护进程可执行文件安装在联合服务器上的一个目录中:

##### 在 UNIX 上

守护进程可执行文件为 db2blast\_daemon。此文件安装在 \$DB2PATH/bin 目录中。

##### 在 Windows 上

守护进程可执行文件为 db2blast\_daemon.exe 和 db2blast\_daemon\_svc.exe。这些文件安装在 %DB2PATH%\bin 目录中。

如果使用独立的 BLAST 服务器，则必须将守护进程可执行文件从联合服务器上的目录复制到 BLAST 服务器上的一个目录中。守护进程可执行文件可以在 BLAST 服务器上的目录路径中的名称中不包含空格的任何目录中运行。

2. 确保 BLAST 守护进程配置文件位于正确的服务器上。在安装 DB2 Information Integrator 期间，样本守护进程配置文件 BLAST\_DAEMON.config 安装在联合服务器上的一个目录中:

##### 在 UNIX 上

该守护进程配置文件安装在 \$DB2PATH/bin 目录中。\$DB2PATH 是 DB2 Information Integrator 的安装目录。

##### 在 Windows 上

该守护进程配置文件安装在 %DB2PATH%\bin 目录中。%DB2PATH% 是 DB2 Information Integrator 的安装目录，通常为 C:\SQLLIB\bin。

缺省情况下，守护进程期望在启动守护进程的工作目录中找到配置文件。可以将配置文件复制到另一个位置。如果使用 BLAST 服务器，则必须将守护进程配置文件从

联合服务器上的目录复制到 BLAST 服务器上的一个目录中。可以将守护进程配置文件复制到 BLAST 服务器上守护进程可以访问的任何目录中。

3. 编辑守护进程配置文件以使用数据源。还可以重命名配置文件。
  - 配置文件中的第一行必须是一个等号。如果缺少等号，则将不能启动守护进程。错误消息将指示未指定 DAEMON\_PORT。
  - 配置文件中的最后一行必须以换行符结束。随 DB2 Information Integrator 一起提供的样本配置文件以换行符结束。当您编辑文件时，必须确保文件中的最后一行以换行符结束。如果最后一行不是以换行符结束，则当您试图使用最后一行上列示的数据源来运行第一个 BLAST 查询时就会接收到错误消息。
  - 在配置文件中指定下列选项。对于需要路径的选项，可以指定相对路径。相对路径是相对于启动守护进程的目录的。

### DAEMON\_PORT

这是守护进程侦听由包装器提交的 BLAST 作业请求的网络端口。

### MAX\_PENDING\_REQUESTS

这是在守护进程上任何一次可以阻塞的 BLAST 作业请求的最大数目。此数目并不表示正在同时运行的 BLAST 作业数，而只是表示一次可以阻塞的作业请求数。建议将此数目设置为一个大于五的数。BLAST 守护进程并不会限制可以同时运行的 BLAST 作业数。

### DAEMON\_LOGFILE\_DIR

这是守护进程在其中创建它的日志文件的目录。此文件包含由 BLAST 守护进程生成的有用的状态信息和错误信息。

### Q\_SEQ\_DIR\_PATH

这是守护进程在其中创建临时查询序列数据文件的目录。一旦完成了 BLAST 作业，就会清除此临时文件。

### BLAST\_OUT\_DIR\_PATH

这是守护进程在其中创建临时文件用来存储 BLAST 输出数据的目录。从此文件中读取数据，并通过网络连接将数据传递回包装器。在将数据传递回包装器之后，守护进程将清除临时文件

### BLASTALL\_PATH

这是在运行守护进程的机器上 BLAST 可执行文件的标准名称。

### 数据库规范条目

指定支持 BLAST 的数据源的位置。记录下您在配置文件中指定的数据库 *data\_source\_name*。为了使守护进程正确工作，当您为数据源创建昵称时，必须指定数据库 *data\_source\_name*。该名称是区分大小写的。数据库 *data\_source\_name* 是在 CREATE NICKNAME 语句的 DATASOURCE 选项中指定的。

配置文件必须至少包含以下格式的一个数据库规范条目：

```
data_source_name = path to BLAST-able_data_source
```

### 在 UNIX 上

例如，要指定 GenBank 支持 BLAST 的数据源，应将下面这一行添加到守护进程配置文件中：

```
genbank=/dsk/1/nuc1_data/genbank
```

## 在 Windows 上

例如，要指定 GenBank 支持 BLAST 的数据源，应将下面这一行添加到守护进程配置文件中：

```
c:\vnr_data\genbank_nonest1.fasta
```

数据库规范条目中指示的路径必须包含三个索引文件。

- 对于核苷酸数据源，索引文件具有下列扩展名：
  - .nhr
  - .nin
  - .nsq
- 对于氨基酸数据源，索引文件具有下列扩展名：
  - .phr
  - .pin
  - .psq

数据库规范条目必须指示包含原始的 Fasta 格式的数据的文件的名称。三个索引文件与包含原始的 Fasta 格式的数据的文件必须具有相同的根名称。

在这一系列任务中的下一个任务是启动 BLAST 守护进程。

### 相关任务：

- 第 95 页的『启动 BLAST 守护进程』

### 相关参考：

- 第 94 页的『BLAST 守护进程配置文件 - 示例』

## BLAST 守护进程配置文件 - 示例

下列示例显示样本配置文件的内容。

### 示例 - BLAST 守护进程配置文件 (UNIX)：

此示例显示 GenBank 和 SWISS-PROT 的必需选项和支持 BLAST 的数据源规范。

```
=
DAEMON_PORT=4007
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=./
Q_SEQ_DIR_PATH=./
BLAST_OUT_DIR_PATH=./
BLASTALL_PATH=./blastall
genbank=/dsk/1/nuc1_data/genbank
swissprot=/dsk/1/prot_data/swissprot
```

### 示例 - BLAST 守护进程配置文件 (Windows)：

此示例显示 GenBank 和 SWISS-PROT 的必需选项和支持 BLAST 的数据源规范。

```
=
DAEMON_PORT=4007
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=.\
Q_SEQ_DIR_PATH=.\
```

```
BLAST_OUT_DIR_PATH=.\
BLASTALL_PATH=.\blastall.exe
genbank=c:\vnr_data\genbank_nonest1.fasta
swissprot=c:\vnr_data\swissprot
```

#### 相关任务:

- 第 90 页的『将 BLAST 数据源添加至联合服务器』
- 第 91 页的『配置 BLAST 守护进程』
- 第 95 页的『启动 BLAST 守护进程』

## 启动 BLAST 守护进程

启动 BLAST 守护进程是将 BLAST 添加至联合系统这一大型任务的一部分。在可以存取 BLAST 数据源之前，必须启动 BLAST 守护进程。

#### 先决条件:

在启动 BLAST 守护进程之前，您必须对列示在配置文件中的 DAEMON\_LOGFILE\_DIR、BLAST\_OUT\_DIR\_PATH 和 Q\_SEQ\_DIR\_PATH 条目下面的所有路径都具有写存取权。

#### 过程:

要在 UNIX 服务器上启动 BLAST 守护进程:

1. 打开守护进程可执行文件所在的目录。
2. 发出 db2blast\_daemon 命令:
  - 如果未更改守护进程配置文件的名称，并且配置文件与守护进程可执行文件处于同一目录中，则在命令行中输入以下命令:

```
db2blast_daemon
```
  - 如果您更改了守护进程配置文件的名称，或者如果守护进程配置文件与守护进程可执行文件不在同一目录中，则必须在包装器守护进程命令上使用 -c 选项来将守护进程可执行文件指向新名称或新位置。

例如，以下命令将导致包装器守护进程在子目录 cfg 中的称为 BLAST\_D.config 的文件中查找守护进程配置信息。

```
db2blast_daemon -c cfg/BLAST_D.config
```

可执行文件将启动新进程，BLAST 守护进程就在该进程中运行。

要在 Windows 服务器上启动 BLAST 守护进程:

1. 打开守护进程可执行文件所在的目录。
2. 发出带有您需要的参数的 db2blast\_daemon 命令。例如，要安装守护进程服务并打开调试:

```
db2blast_daemon -a install -d 2
db2blast_daemon -a start
```

在这一系列任务中的下一个任务是注册 BLAST 包装器。

#### 相关任务:

- 第 97 页的『注册 BLAST 包装器』

## db2blast\_daemon 命令 - 语法和示例

可以在 UNIX 和 Windows 服务器上使用 `db2blast_daemon` 命令。语法中列示的某些自变量只能在 Windows 服务器上使用。

`db2blast_daemon` 命令的语法为:

```
db2blast_daemon -a action -c config_file -d debug_level
                  -u user_id -p password
```

### **-a** *action*

执行指定的活动。有效操作为 *status*、*install*、*start*、*stop* 和 *remove*。

只能在 Windows 服务器上指定此自变量。

### **-c** *config\_file*

指示守护进程服务使用指定的配置文件。如果不指定配置文件，则守护进程将在安装了守护进程可执行文件的目录中搜索 `BLAST_DAEMON.config` 文件。可以将此选项与 *install* 和 *start* 操作配合使用。

可以在 UNIX 和 Windows 服务器上指定此自变量。

### **-d** *debug\_level*

将守护进程服务调试级别设置为指定的值。有效值为 1、2 或 3。可以将此选项与 *install* 和 *start* 操作配合使用。

可以在 UNIX 和 Windows 服务器上指定此自变量。

### **-u** *user\_id*

将守护进程服务设置为以指定的用户标识运行。可以将此选项与 *install* 操作配合使用。

只能在 Windows 服务器上指定此自变量。

### **-p** *password*

指定所指定的用户标识的密码。仅当指定了 `-u` 选项时，密码才有效并且是必需的。如果设置 `-u` 选项时未指定 `-p` 选项，则程序将提示您输入密码。可以将此选项与 *install* 操作配合使用。

只能在 Windows 服务器上指定此自变量。

随 *start* 操作一起指定的选项只影响守护进程的当前运行，并且将覆盖随 *install* 操作一起指定的值。

### 示例:

下列示例显示守护进程操作。这些示例假定 `BLAST_DAEMON.config` 文件与 `db2blast_daemon.exe` 位于同一目录中。

- 要检查守护进程的状态:  
`db2blast_daemon -a status`
- 要安装守护进程服务并打开调试:  
`db2blast_daemon -a install -d 2`
- 要启动守护进程:  
`db2blast_daemon -a start`
- 要停止守护进程:



```
db2blast_daemon -a stop
```

- 要除去守护进程服务:

```
db2blast_daemon -a remove
```

相关任务:

- 第 95 页的『启动 BLAST 守护进程』

## 注册 BLAST 包装器

注册 BLAST 包装器是将 BLAST 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 BLAST 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 blast\_wrapper 的包装器，请发出以下语句:

```
CREATE WRAPPER blast_wrapper LIBRARY 'libdb2lsblast.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 BLAST 包装器库文件的列表。

在这一系列任务中的下一个任务是为 BLAST 包装器注册服务器定义。

相关参考:

- 第 97 页的『BLAST 包装器库文件』
- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）

## BLAST 包装器库文件

下表列示了 BLAST 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2lsblast.a、libdb2lsblastF.a 和 libdb2lsblastU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 24. BLAST 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2lsblast.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsblast.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsblast.so
Windows	%DB2PATH%\bin	db2lsblast.dll

| %DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows  
| 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

| 相关任务:

- | • 第 97 页的『注册 BLAST 包装器』

## 为 BLAST 数据源注册服务器

为 BLAST 数据源注册服务器是将 BLAST 添加至联合系统这一大型任务的一部分。在注册包装器之后，必须注册相应的服务器。

过程:

要向联合系统注册 BLAST 服务器，使用 CREATE SERVER 语句。

对于您的环境中安装了 BLAST 可执行文件和守护进程的每台机器，必须使用 BLAST 可执行文件和守护进程的实例为您想要运行的每种类型的 BLAST 搜索注册一个服务器。

例如，要为使用 CREATE WRAPPER 语句创建的将用于 BLASTn 搜索的 my\_blast 包装器注册一个称为 blast\_server1 的服务器，提交以下语句:

```
CREATE SERVER blast_server1
  TYPE blastn
  VERSION 2.1.2
  WRAPPER my_blast
  OPTIONS (NODE 'big_rs.company.com', DAEMON_PORT '4007')
```

在这一系列任务中的下一个任务是为 BLAST 数据源注册昵称。

相关任务:

- 第 98 页的『为 BLAST 数据源注册昵称』

相关参考:

- 『ALTER SERVER statement』 (*SQL Reference, Volume 2*)
- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 498 页的『CREATE SERVER 语句自变量 - BLAST 包装器』

## 为 BLAST 数据源注册昵称

### 为 BLAST 数据源注册昵称

为 BLAST 数据源注册昵称是将 BLAST 添加至联合系统这一大型任务的一部分。

在注册服务器之后，必须注册相应的昵称。当您在查询中引用 BLAST 数据源时就要使用昵称。

过程:

要注册 BLAST 昵称，使用 CREATE NICKNAME 语句。

由于每种类型的 BLAST 搜索是由单独的服务器处理的，因此，必须为您想对给定的支持 BLAST 的数据源运行的每种类型的 BLAST 搜索定义一个单独的昵称。

当创建昵称时，为数据源的定义行部分指定列信息。所有其它列是固定的。

在这一系列任务中没有别的任务了。

#### 相关概念:

- 第 99 页的『定义行分析』

#### 相关任务:

- 第 90 页的『将 BLAST 数据源添加至联合服务器』
- 第 58 页的『为非关系数据源指定昵称列』

#### 相关参考:

- 第 498 页的『CREATE NICKNAME 语句语法 - BLAST 包装器』
- 第 103 页的『CREATE NICKNAME 语句 - BLAST 包装器的示例』
- 第 99 页的『BLAST 昵称的固定列』

## 定义行分析

定义行类似于支持 BLAST 的数据源中的每个序列的一个键，并且是作为每个 BLAST 符合项的一部分返回的。定义行又称为 *define*。

由 BLAST 包装器为定义行返回和解析的值将不会始终完全等同于原始 FASTA 文件中的定义行。例如，如果在 BLAST 符合项的“存取号”字段中具有数据，则返回的定义行将包含“存取号”数据，并且后跟“定义”字段数据。然后，包装器对返回的数据进行语法分析。

**建议：**要确定包装器将如何返回定义行和对定义行进行语法分析，使用单个定义行列创建昵称。然后运行查询以查看由特定数据源的定义行的包装器返回的格式。

要将定义行信息包括在结果表中，必须在 CREATE NICKNAME 语句中指定定义行列。指定的每一列都必须同时包括 INDEX 选项和 DELIMITER 选项。如果想要最后一列包含定义行信息的其余部分，则可以省略所指定的最后一列上的 DELIMITER 选项。

定义行列的有效数据类型为 CLOB、DOUBLE、FLOAT、INTEGER 和 VARCHAR。

#### 相关概念:

- 第 420 页的『定义行语法分析用户定义的函数 - 概述』

#### 相关任务:

- 第 98 页的『为 BLAST 数据源注册昵称』

#### 相关参考:

- 第 498 页的『CREATE NICKNAME 语句语法 - BLAST 包装器』
- 第 103 页的『CREATE NICKNAME 语句 - BLAST 包装器的示例』
- 第 99 页的『BLAST 昵称的固定列』

## BLAST 昵称的固定列

当对 BLAST 数据源发出 CREATE NICKNAME 语句时，将自动对昵称创建一组固定列。

固定列是昵称的定义的一部分，固定列是在联合数据库系统目录中创建的。可以在 SQL 查询中引用固定列。有两种类型的固定列：输入固定列和输出固定列。

**BLAST 昵称的固定输入列：** 固定输入列是在 WHERE 子句中指定的。在 SQL 查询中，输入列被用作参数传递谓词。它们将标准的 BLAST 开关传递给 BLAST。然后 BLAST 使用这些开关对指定的数据源进行运行。还可以在查询 SELECT 列表中引用固定输入列，固定输入列是作为结果表的一部分返回的。

下表列示了可以在 WHERE 子句中使用的固定列。

表 25. BLAST 昵称的固定输入列

名称	数据类型	运算符	描述
BlastSeq	V A R C H A R (32000) 或 CLOB	=	将查询序列传递给 BLAST 包装器。
E_Value	DOUBLE	<	既是输入参数，又是输出参数。作为一个输入参数，此列指示 BLAST 包装器应当从 blastall 返回的期望值的上限。
QueryStrands	INTEGER	=	指定在执行 BLASTn 搜索时应当比较哪些链。如果值为 1，则指示应当使用顶部的链，值为 2 则指示应当使用底部的链，值为 3 则指示应当同时比较顶部的链和底部的链。
GapAlign	CHAR(1)	=	指示包装器在 BLAST 输出中是否允许有空位的比对。
Matrix	VARCHAR(50)	=	确定 blastall 使用哪些替代矩阵来确定一些氨基酸对之间的同源程度。只有将氨基酸与氨基酸进行比较的那些 BLAST 搜索类型才会使用此谓词。
NMisMatchPenalty	INTEGER	=	如果在对应区域中的其中一个核苷酸对不匹配，则指定 blastall 从比对的分数中减去的值。只有将核苷酸与核苷酸进行比较的那些 BLAST 搜索类型才会使用此谓词。
NMatchReward	INTEGER	=	指定 blastall 为对应区域中相匹配的每一对核苷酸添加到比对的分数中的值。只有将核苷酸与核苷酸进行比较的那些 BLAST 搜索类型才会使用此谓词。
FilterSequence	CHAR(1)	=	指示 blastall 是否执行过滤以便从查询序列中除去生物学上不令人感兴趣的段。如果搜索类型为 BLASTn，则使用的过滤器为 DUST。否则，由 SEG 执行过滤。
NumberOfAlignments	INTEGER	=	指定 BLAST 输出中要包括多少个 HSP 比对。

表 25. BLAST 昵称的固定输入列 (续)

名称	数据类型	运算符	描述
GapCost	INTEGER	=	如果必须在查询序列或符合项序列中引入空位以便允许比对的长度增长, 则指定 blastall 从比对的分数中减去的值。
ExtendedGapCost	INTEGER	=	如果一个核苷酸或氨基酸必须扩展在查询序列或符合项序列中已经引入的空位以便允许比对的长度增长, 则指定 blastall 从比对的分数中减去的值。
WordSize	INTEGER	=	指示 blastall 最初在数据库中其搜索的初始符合项的长度。
ThresholdEx	INTEGER	=	指示分数阈值, 低于该阈值, BLAST 就不再尝试进一步扩展符合项。

当创建昵称时, 可以覆盖这些列的缺省数据类型。例如, 某些列可以返回大量数据, 例如, HSP\_H\_Seq 和 HSP\_Midline 列。要返回一列的前 50 个字节, 可以使用数据类型 VARCHAR(50) 来定义该列。将只把前 50 个字节复制到输出列中。

**固定输入列的 BLAST 搜索类型和开关:** 下表列示了每个固定输入列的受支持的 BLAST 搜索类型和开关。

表 26. 输入固定列支持的 BLAST 搜索类型和开关

名称	BLAST 搜索类型	BLAST 开 必需 关	缺省值
BlastSeq	n、p、x、tn 和 tx	-l	是 N/A
E_Value	n、p、x、tn 和 tx	-e	否 10
QueryStrands	n	S	否 3
GapAlign	n、p、x、tn 和 tx	-g	否 T
Matrix	p、x、tn 和 tx	-n	否 BLOSUM62
NMisMatchPenalty	n	-q	否 -3
NMatchReward	n	-r	否 1
FilterSequence	n、p、x、tn 和 tx	-F	否 T
NumberOfAlignments	n、p、x、tn 和 tx	-b	否 250
GapCost	n、p、x、tn 和 tx	-G	否 11
ExtendedGapCost	n、p、x、tn 和 tx	-E	否 1
WordSize (对于 Blastn, 小于 7 的值无效)	n、p、x、tn 和 tx	-W	否 11 -BLASTn 3 -BLASTp
ThresholdEx	n、p、x、tn 和 tx	-f	否 0

**BLAST 昵称的固定输出列:** 下表列示了可以在 WHERE 子句中使用的固定列。

表 27. BLAST 昵称的固定输出列

名称	数据类型	描述
Score	DOUBLE	在 BLAST 结果中报告的为 HSP 计算的分数。
E_value	DOUBLE	既是输入参数, 又是输出参数。作为一个输出参数, 此列提供了在 BLAST 结果中报告的为 HSP 计算的分数。
Length	INTEGER	在 BLAST 结果中报告的符合项序列的长度。
HIT_NUM	INTEGER	在 BLAST 结果中报告的符合项编号, 从 1 开始。
HSP_NUM	INTEGER	在 BLAST 结果中报告的 HSP 编号, 从 1 开始。
HSP_Info	VARCHAR(100)	由 BLAST 报告的给定 HSP 的信息字符串。此字符串包含有关查询序列与符合项序列之间相匹配的核苷酸或氨基酸数目的信息。
HSP_ALIGNMENT_LENGTH	INTEGER	HSP 比对的长度。
HSP_IDENTITY	INTEGER	比对的全等百分比定义为全等的数目除以比对长度。
HSP_GAPS	INTEGER	比对中的空位百分比定义为空位的数目除以比对长度。
HSP_POSITIVE	INTEGER	比对的正值 (positive) 百分比定义为正值的数目除以比对长度。
HSP_QUERY_FRAME	INTEGER	查询序列中比对的读框架。  只能用于 blastx、tblastn 和 tblastx 类型的服务器。
HSP_HIT_FRAME	INTEGER	符合项序列中比对的读框架。  只能用于 blastx、tblastn 和 tblastx 类型的服务器。
HSP_Q_Start	INTEGER	查询序列上第一个同源核苷酸或氨基酸的数字位置。
HSP_Q_End	INTEGER	查询序列上最后一个同源核苷酸或氨基酸的数字位置。
HSP_Q_Seq	VARCHAR(32000)	从 HSP_Q_Start 开始, 在 HSP_Q_End 结束的查询序列的段。  可以覆盖此列的缺省数据类型, 并指定 CLOB, 其最大长度为 5 兆字节。
HSP_H_Start	INTEGER	符合项序列上第一个同源的核苷酸或氨基酸的数字位置。
HSP_H_End	INTEGER	符合项序列上最后一个同源的核苷酸或氨基酸的数字位置。

表 27. BLAST 昵称的固定输出列 (续)

名称	数据类型	描述
HSP_H_Seq	VARCHAR(32000)	从 HSP_H_Start 开始, 在 HSP_H_End 结束的符合项序列的段。  可以覆盖此列的缺省数据类型, 并指定 CLOB, 其最大长度为 5 兆字节。
HSP_Midline	VARCHAR(32000)	由 BLAST 输出的一个字符串, 用来指示在查询序列和符合项序列的对应区域中的每个位置处氨基酸与核苷酸之间的同源程度。  可以覆盖此列的缺省数据类型, 并指定 CLOB, 其最大长度为 5 兆字节。

#### 相关概念:

- 第 99 页的『定义行分析』

#### 相关任务:

- 第 98 页的『为 BLAST 数据源注册昵称』

#### 相关参考:

- 第 498 页的『CREATE NICKNAME 语句语法 - BLAST 包装器』
- 第 103 页的『CREATE NICKNAME 语句 - BLAST 包装器的示例』

## CREATE NICKNAME 语句 - BLAST 包装器的示例

下列 CREATE NICKNAME 语句将定义昵称 genbank。

它假定 BLAST 结果中的定义字段包含以下信息:

```
>276342 15:8924 PMON5426
```

其中:

### 276342

BLAST 结果的入藏字段。

### 15:8924 PMON5426

BLAST 结果中的定义字段, 它包含一个组织编号, 接着是实验编号, 然后是唯一标识。

通过使用此信息, 创建了以下昵称:

```
CREATE NICKNAME genbank (
  acc_num integer  OPTIONS(INDEX '1', DELIMITER ' '),
  org_num integer  OPTIONS(INDEX '2', DELIMITER ':'),
  exp_num integer  OPTIONS(INDEX '3', DELIMITER ' '),
  u_id varchar(10) OPTIONS(INDEX '4'))
FOR SERVER blast_server1
OPTIONS(DATASOURCE 'genbank', TIMEOUT '300');
```

acc\_num 列将包含 276342, org\_num 将包含 15, exp\_num 列将包含 8924, 而 u\_id 列将包含 PMON5426。

在提交 CREATE NICKNAME 语句之后，可以使用昵称 genbank 来查询联合系统。还可以将昵称 genbank 与联合系统中的其它昵称和表连接起来。

**相关任务:**

- 第 98 页的『为 BLAST 数据源注册昵称』

**相关参考:**

- 『ALTER NICKNAME statement』 (*SQL Reference, Volume 2*)
- 『CREATE NICKNAME statement』 (*SQL Reference, Volume 2*)
- 第 498 页的『CREATE NICKNAME 语句语法 - BLAST 包装器』

---

## 设置 TurboBlast 以使用 BLAST 包装器

**限制:**

TurboBlast 不支持某些 blastall 命令选项。例如，带有空位的比对选项 -g F 不受支持。如果您对 BLAST 昵称中 GapAlign 的列的值指定 F，则 TurboBlast 将产生错误。有关不受支持的选项的完整列表，参阅 *TurboBlast 2.0 User Guide*。

**过程:**

要设置 TurboBlast 以使用 BLAST 包装器:

1. 安装和配置 BLAST 包装器。对支持 blast 的数据库运行查询来测试设置。
2. BLAST 包装器和 TurboBlast 支持 AIX、Linux、Solaris 和 Windows 操作系统。但是，Windows 操作系统上未提供 BLAST 守护进程。当那些操作系统上提供了 BLAST 守护进程时，守护进程将使用 Windows 操作系统上的 TurboBlast。
3. 根据 *TurboBlast 2.0 Installation and Reference Guide* 来安装和配置 TurboBlast。可以采用各种方法来安装和设置 TurboBlast 系统。要允许 BLAST 包装器使用 TurboBlast，需要在具有 BLAST 守护进程的计算机上安装并设置 TurboBlast Client。BLAST 守护进程可以调用 tblastall 命令。
4. 在安装和配置 TurboBlast 之后务必测试 TurboBlast 系统。遵循 *TurboBlast 2.0 Installation and Reference Guide* 中的指示信息。
5. 按如下所示更改 BLAST\_DAEMON.config 文件:
  - a. 将 BLASTALL\_PATH 参数指定为 tblastall 的完整路径。例如：  
BLASTALL\_PATH=/home/blastst/turboblast/TBlast-2.1/tblastall
  - b. 将支持 BLAST 的数据库规范条目指定为用来将支持 BLAST 的数据库上载至 TurboBlast 的支持 BLAST 的数据库名称。当在 TurboBlast 中输入 listdatabase -l 命令时就会显示数据库名称。应当使用此 TurboBlast 数据库名称，而不使用至支持 blast 的数据源的路径。例如：genbank=<the genbank database name in TurboBlast>
6. 重新启动 BLAST 守护进程。BLAST 守护进程将调用 tblastall 而不是 blastall 来在支持 blast 的数据库上执行搜索工作。
7. 与 tblastall 相关的日志文件被写入在 BLAST\_DAEMON.config 文件中指定的 DAEMON\_LOGFILE\_DIR。还要检查由 BLAST 守护进程在同一目录中生成的 STDERR.log 和 STDOUT.log。

**相关任务:**



- 第 90 页的『将 BLAST 数据源添加至联合服务器』
- 第 91 页的『配置 BLAST 守护进程』

---

## 构造 BLAST SQL 查询

输入列上的谓词用来将标准 BLAST 开关传递至 blastall 可执行文件。输出列上的谓词由联合服务器来处理。

**限制:**

传递给 BLAST 包装器的每个查询必须至少包含 BlastSeq 输入谓词才有效。所有其它谓词是可选的。

**过程:**

要构造 BLAST 查询，在 SQL 语句的 WHERE 子句中使用输入谓词。

以下示例显示了三个输入谓词: BlastSeq、GapCost 和 NMisMatchPenalty。

```
Select * from blast b where  
BlastSeq = 'GTCCAGCC...' AND  
GapCost = -10 AND  
NMisMatchPenalty = -4;
```

**相关任务:**

- 第 98 页的『为 BLAST 数据源注册昵称』

**相关参考:**

- 第 105 页的『BLAST 数据源 - 示例查询』

---

## BLAST 数据源 - 示例查询

提供了几个样本 BLAST 查询来说明如何为 BLAST 数据源构造查询。

要运行查询，将这些示例作为指南。

在这些查询中，用于每个昵称的名称指示 BLAST 搜索的类型和数据源。这已经完成了，所以不需要每个样本查询都列示注册语句。另外，某些查询利用其它假想数据源，以便这些示例可以说明包装器在与其它数据源连接时的行为。

### 查询 1

```
select *  
from blastn_genbank  
where BlastSeq =  
'caaccctccagccgagttgtcaatggcgaggaagctgttccccac';
```

当执行此 SQL 语句时，包装器将使用指示的序列对 GenBank 执行 BLASTn 搜索。包装器将返回所有可用的列，包括输入参数列和 BLAST 结果列。

### 查询 2

```
select *
from blastn_genbank
where BlastSeq =
'caaccctccagccgagttgtcaatggcgaggaagctgttccccac'
and GapCost = 8 and NmisMatchPenalty = -4;
```

当执行此 SQL 语句时，包装器将使用指示的序列对 GenBank 执行 BLASTn 搜索。另外，包装器将把指示的两个参数传递给守护进程，然后它们又将传递给 blastall 命令行。包装器将返回所有可用的列，包括输入参数列和 BLAST 结果列。

### 查询 3

```
select blp.*
from blastp_swissprot blp, protein_db prdb
where prdb.keyword = 'malic enzyme'
and blp.BlastSeq = prdb.sequence;
```

当执行此 SQL 语句时，包装器将根据从假想的蛋白质序列数据库中返回的序列数来对 SWISS-PROT 执行零次或多次 BLASTp 搜索。此语句将被 DB2 分成两个独立的查询，并且将对从假想的蛋白质数据库中返回的每一行运行一次 BLASTp 搜索。包装器将返回所有可用的列，包括输入参数列和 BLAST 结果列。

### 查询 4

```
select Score, E_Value, HSP_Info, HSP_Q_Seq, HSP_H_Seq, HSP_Midline
from blastx_swissprot
where BlastSeq = 'gagttgtcaatggcgagg'
and GapCost = 8;
```

当执行此 SQL 语句时，包装器将使用指示的序列对 SWISS-PROT 执行 BLASTx 搜索。在此情况下，blastall 将按照所有六个读框架翻译输入序列并使用新创建的六个蛋白质序列中的每个序列来执行相应的搜索。结果中的 HSP 将包含氨基酸 - 氨基酸比对，而不是核苷酸 - 核苷酸比对。所提供的参数将传递给守护进程，然后通过命令行传递给 blastall。包装器将只返回在查询中专门请求的那些列。

### 查询 5

```
select tblx.Score, tblx.E_Value, tblx.HSP_Info tblx.HSP_Q_Seq,
HSP_H_Seq, HSP_Midline
from tblastx_genbank tblx, gen_exp_database gedb
where tblx.BlastSeq = gedb.sequence
and gedb.organism = 'interesting organism'
and GapCost = 8
and FilterSequence = 'F';
```

当执行此 SQL 语句时，包装器将根据从假想的基因表达式数据库中返回的序列数来对 GenBank 执行零次或多次 tBLASTx 搜索。此语句将被 DB2 分成两个独立的查询，并且将对从假想的基因表达式数据库中返回的每一行运行一次 tBLASTx 搜索。在此情况下，blastall 将按照所有六个读框架翻译输入序列和 GenBank 中的所有序列并使用新创建的六个查询蛋白质序列中的每个序列和新创建的所有数据库蛋白质序列来执行相应的搜索。结果中的 HSP 将包含氨基酸 - 氨基酸比对，而不是核苷酸 - 核苷酸比对。所提供的参数将传递给守护进程，然后通过命令行传递给 blastall。包装器将只返回在查询中专门请求的那些列。

### 相关参考:

- 第 172 页的『Documentum 数据源 - 示例查询』
- 第 205 页的『Excel 数据源 - 示例查询』

## BLAST 包装器的优化提示

要提高网络通信性能，联合服务器和 BLAST 服务器应该位于不同的硬件上。BLAST 守护进程应该驻留在 BLAST 服务器上。

相关任务:

- 第 91 页的『配置 BLAST 守护进程』

## BLAST 包装器的消息

本节列示和描述了您在使用 BLAST 的包装器时可能会遇到的消息。

表 28. 由 BLAST 的包装器发出的消息

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “sqlno_crule_save_plans [100]:rc (-2144272209) 检测到空的计划列表”。)	包装器不能处理提交给 DB2 的 SQL 查询。更正语法，然后重新提交。
SQL1816N	包装器 “BLAST_WRAPPER” 不能用来存取您正在尝试向联合数据库定义的数据源的 “类型” ( “<server type>” )。	CREATE SERVER 语句使用了无效 TYPE。类型必须是受支持的其中一种 BLAST 类型。
SQL1817N	CREATE SERVER 语句未标识您想要向联合数据库定义的数据源的 “version”。	CREATE SERVER 语句未指定版本。
SQL1822N	从数据源 “Blast 包装器” 中接收到意外的错误代码 “未指定的错误”。相关联的文本和标记为 “未能连接至守护进程”。	blast 包装器未能连接至守护进程。守护进程可能未运行。也可能是进行了错误配置。可能无法访问它正在运行的机器。
SQL1822N	从数据源 “Blast 包装器” 中接收到意外的错误代码 “未指定的错误”。相关联的文本和标记为 “Blast 守护进程超时已到期”。	经过在 CREATE NICKNAME 语句上指定的超时之前未从守护进程中接收到结果。增大超时，或者检查守护进程是否有问题。
SQL1822N	从数据源 “Blast 包装器” 中接收到意外的错误代码 “未指定的错误”。相关联的文本和标记为 “Blast 守护进程失败”。	守护进程已停止通信或者返回的结果未正确格式化。
SQL1822N	从数据源 “Blast 包装器” 中接收到意外的错误代码 “未指定的错误”。相关联的文本和标记为 “blast 守护进程中存在未知错误”。	blast 包装器从它不能识别的守护进程中接收到错误代码。守护进程版本可能与包装器版本不兼容。
SQL1822N	从数据源 “Blast 包装器” 中接收到意外的错误代码 “未指定的错误”。相关联的文本和标记为 “不允许对列进行重命名”。	发出了 ALTER NICKNAME 语句来试图对其中的一列进行重命名。不允许对某列进行重命名。

表 28. 由 *BLAST* 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源 “Blast 包装器” 中接收到意外的错误代码 “未指定的错误”。相关联的文本和标记为 “XML 解析器错误”。	Xerces 解析器处于无效状态或者抛出了异常。
SQL1823N	对于服务器 “<server name>” 中的数据类型 “<data type name>” 不存在数据类型映射。	此列不支持所指定的数据类型。
SQL1881N	“DEFAULT” 不是 “<column-name>” 的有效 “COLUMN” 选项	在不支持 DEFAULT 选项的一列上使用了该选项。“仅输出” 列和 “定义行” 列没有缺省值。
SQL1882N	“COLUMN” 选项 “DEFAULT” 不能设置为 “<column-name>” 的 “<option-value>”。	为 DEFAULT 选项指定的值对于某列是不兼容类型或者格式不正确。

**相关概念:**

- 『消息介绍』（《消息参考》第 1 卷）

**相关参考:**

- 『SQLSTATE 消息』（《消息参考》第 2 卷）

---

## 第 8 章 配置对商业应用程序数据源的存取

本节说明如何使用 Websphere Business Integration 包装器来将商业应用程序数据源添加至联合系统。

---

### WebSphere Business Integration 包装器

WebSphere<sup>®</sup> Business Integration 包装器是一个只读包装器，它使用 WebSphere Business Integration 适配器来访问商业应用程序。有关受支持的适配器和应用程序的列表，请参阅《*IBM DB2 Information Integrator 联合系统指南*》。

WebSphere Business Integration 包装器提供了与一些商业应用程序（例如，由 SAP、Siebel 和 PeopleSoft 提供的那些商业应用程序）的 SQL 接口。通过使用 WebSphere Business Integration 包装器，可以使用联合系统功能来将商业应用程序中的业务数据与其它联合数据源中的数据连接起来。WebSphere Business Integration 包装器将业务对象定义抽取到昵称的层次结构中。

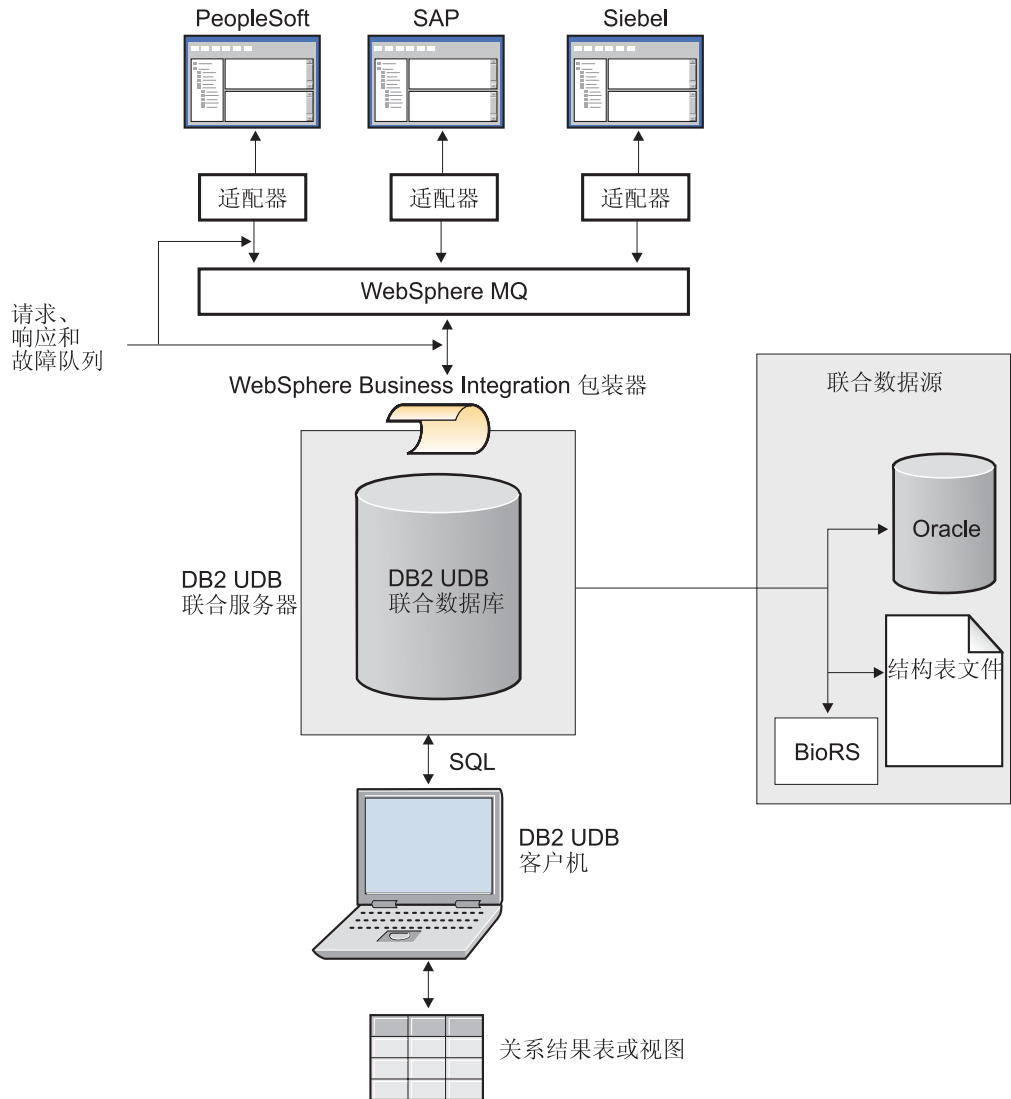


图 9. DB2® 通用数据库环境中的 WebSphere Business Integration 包装器

图 9 显示 WebSphere Business Integration 包装器与 DB2 Universal Database™ (DB2 通用数据库) 环境中的适配器之间的关系。下列步骤描述访问联合系统中的商业应用程序数据的过程:

1. 用户向联合服务器发送一个查询来存取映射至数据源 (例如, Siebel 应用程序) 的昵称。
2. 包装器将查询变换成业务对象。
3. 包装器将该业务对象放置到 WebSphere MQ 消息队列上。
4. 特定应用程序的 WebSphere Business Integration 适配器从消息队列中读取业务对象 (它是一个请求)。
5. WebSphere Business Integration 适配器使用商业应用程序来准备响应业务对象。
6. WebSphere Business Integration 适配器将响应业务对象放置到消息队列上。
7. 包装器从响应队列中读取响应业务对象。
8. 包装器根据使用昵称定义来定义的关系模式来将响应业务对象抽取到结果集中。

相关概念:

- 第 111 页的『业务对象定义』

#### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 116 页的『注册 WebSphere Business Integration 包装器』
- 第 112 页的『配置 WebSphere Business Integration 适配器』
- 第 118 页的『为商业应用程序数据源注册昵称』

#### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』

---

## 业务对象定义

业务对象定义是一个模板，WebSphere® Business Integration Adapter 根据它来创建业务对象的实例。业务对象定义表示商业应用程序数据实体。

业务对象是业务对象定义的一个实例，它可以是平面结构，也可以是分层结构。平面业务对象只包含一些简单属性。分层业务对象包含一种或多种复杂属性。业务对象定义的资源库存在于受支持的特定应用程序的每个 WebSphere Business Integration 适配器中。

以下示例显示一个平面业务对象:

```
Customer  
  Gomez  
  Juanita  
  Apt 2C  
  123 Main Street  
  Big City  
  California  
  91234  
  888  
  1111111
```

以下示例显示一个具有一些复杂属性的分层业务对象:

```
Contact (Parent)  
  ID  
  Customer ID  
  Date  
  Text  
  Authorization  
  Line items (there are 0 or more Line item elements)  
    (Child elements)  
    Business object 1  
    Business object 2  
    Business object 3
```

必须使用与每个 WebSphere Business Integration 适配器打包在一起的对象发现代理工具来生成业务对象定义。对象发现代理工具将为业务对象定义生成 XML 模式定义文件。如果正在定义的业务对象具有分层结构，则对象发现代理工具可以生成多个模式文件。

XML 模式定义是在 WebSphere Business Integration 配置中指定的目录中的一个文件类型为 .xsd 的文件。在为 WebSphere Business Integration 包装器创建昵称之前必须生成业务对象定义。有关 WebSphere Business Integration 适配器系列的更多信息，请参阅：[www.ibm.com/websphere/integration/wbiadapters](http://www.ibm.com/websphere/integration/wbiadapters)。

要创建昵称，使用对象发现代理工具创建的 xsd 文件。昵称提供了业务对象定义的关系模式表示。WebSphere Business Integration 包装器将分层业务对象映射到关系昵称的层次结构中。例如，基数为“n”的每个子业务对象映射至通过外键约束链接至父业务对象的昵称的一个单独昵称。

IBM® DB2® Information Integrator 可以存取的 WebSphere Business Integration 业务对象映射至下表中的特定应用程序实体：

表 29. 业务对象和相关的应用程序实体

业务对象	应用程序实体
Siebel	业务组件
PeopleSoft	组件接口
SAP	BAPI

#### 相关概念：

- 第 109 页的『WebSphere Business Integration 包装器』

#### 相关任务：

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 112 页的『配置 WebSphere Business Integration 适配器』

#### 相关参考：

- 第 142 页的『商业应用程序数据源 - 示例查询』

---

## 配置 WebSphere Business Integration 适配器

对于您想通过使用联合包装器功能并使用 SQL 语句来访问的每个商业应用程序，必须安装和配置 WebSphere Business Integration 适配器。每个适配器都映射至联合服务器定义。

#### 先决条件：

- 有关受支持的适配器的列表，请参阅《*IBM DB2 Information Integrator 联合系统指南*》。
- 有关每个适配器的安装信息，请参阅 IBM WebSphere Business Integration 信息中心。
- 有关特定商业应用程序的帮助，请参阅 WebSphere Business Integration 适配器文档。
- 安装想要使用的特定适配器的所有最新的修订包。可以从 WebSphere Business Integration Adapters Support 站点获得相应的支持信息。
- 有关配置消息队列的信息，请参阅 *WebSphere MQ System Administration Guide*。
- 有关适配器和配置属性的信息，请参阅 IBM WebSphere Business Integration 信息中心中的安装信息。

#### 过程：



要配置 WebSphere Business Integration 适配器:

1. 配置对象发现代理工具和业务对象设计器工具, 并构建业务对象定义。

当在业务对象设计器工具中配置业务对象定义时, 根据商业应用程序指定下列动词值:

表 30. 与商业应用程序配合使用的动词值

商业应用程序	动词
SAP	Retrieve
PeopleSoft	Retrieve
Siebel	RetrieveByContent

有关如何配置和使用对象发现代理工具的更多信息, 请参阅正在配置的适配器的文档。

2. 从“WebSphere Business Integration 适配器”界面中使用“连接器配置程序”工具来定义包含以下信息的配置文件:

- 适配器支持的业务对象。
- 适配器的配置属性。具有一些标准配置属性和特定于应用程序的配置属性。

#### 标准配置属性

必须定制一些属性值以便将适配器与 IBM DB2 Information Integrator 配合使用。以下列表中包括要配置的一些特定属性:

- 将集成代理的值指定为 WMQI。
- 指定适配器所拥有的元数据资源库的位置。XML 模式定义文件(它包含业务对象定义)就保存在此位置。
- 将传递传输的类型指定为 WMQI-MQ。
- 指定用来管理适配器所使用的队列的队列管理器的名称。
- 指定运行适配器所需要的八个队列的名称。

#### 特定于应用程序的配置属性

这些属性指定特定于应用程序的特定组件的值。您提供的值可以帮助与应用程序建立会话。这些属性还指导特定于应用程序的组件的处理行为。

3. 定义包装器需要的三个 WebSphere MQ 消息队列: request\_queue、response\_queue 和 fault\_queue。

WebSphere MQ 是适配器与包装器之间的消息传递和传输层。

#### request\_queue

将请求消息从 DB2 Information Integrator 传递给适配器。

#### response\_queue

将响应消息从适配器传递给 DB2 Information Integrator。

#### fault\_queue

将故障消息从适配器传递给 DB2 Information Integrator。当适配器不能将消息放置到回复队列上时就会将消息放置到此队列上。

这些队列是用来在适配器与包装器之间交换消息（包括数据对象和错误消息）的静态队列。

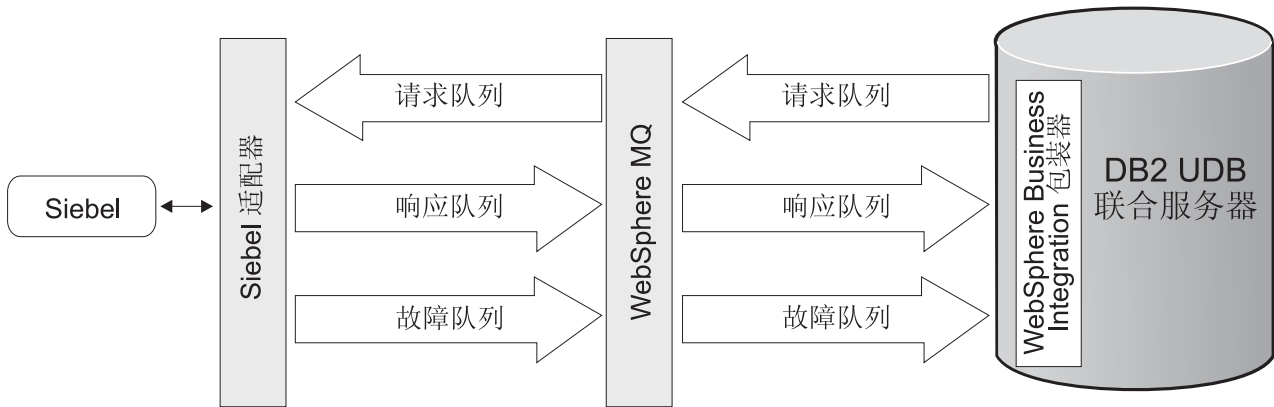


图 10. 在 Siebel 商业应用程序与 DB2 联合服务器之间传输信息的 WebSphere 消息队列的拓扑结构

#### 4. 定义适配器需要的五个附加消息队列:

- AdminInQueue
- AdminOutQueue
- SynchronousRequestQueue
- SynchronousResponseQueue
- DeliveryQueue

WebSphere Business Integration 适配器需要五个附加队列，当适配器与 WMQI 代理而不是与 DB2 Information Integrator 配合使用时就要使用这些队列。必须创建和配置这些附加消息队列才能启动适配器。

#### 5. 使用下列方法之一来定义 WebSphere MQ 用户权限:

- 将 DB2 实例所有者标识定义为 MQManager 组的一部分。
- 确保 MQManager 管理员在创建 ServerConnection 通道时设置 MCAUSER 值。MCAUSER 的值必须是作为 MQManager 组或 Administrator 组的一部分的用户标识。

#### 相关概念:

- 第 111 页的『业务对象定义』
- 第 109 页的『WebSphere Business Integration 包装器』

#### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 116 页的『注册 WebSphere Business Integration 包装器』

#### 相关参考:

- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』

## 将商业应用程序添加至联合服务器

### 将商业应用程序数据源添加至联合系统

要配置联合服务器以存取商业应用程序数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取商业应用程序数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

#### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上。
- 使用 32 位 DB2 UDB 实例的联合数据库必须存在于联合服务器上。

#### 过程:

要将商业应用程序数据源添加至联合系统:

1. 注册 WebSphere Business Integration 包装器。
2. 注册服务器定义。
3. 注册商业应用程序数据源的昵称。
4. 可选: 为 WebSphere Business Integration 昵称创建联合视图。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』
- 第 111 页的『业务对象定义』
- 第 109 页的『WebSphere Business Integration 包装器』

#### 相关任务:

- 『在安装了 DB2 Information Integrator 之后编辑 Oracle genclntsh 脚本并创建 libclntsh 文件 (HP-UX、Linux 和 Solaris)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 116 页的『注册 WebSphere Business Integration 包装器』
- 第 117 页的『为商业应用程序数据源注册服务器定义』
- 第 118 页的『为商业应用程序数据源注册昵称』
- 第 126 页的『为商业应用程序昵称创建联合视图』
- 第 33 页的『检查联合服务器的设置』
- 第 44 页的『创建联合数据库』
- 第 116 页的『注册 WebSphere Business Integration 包装器』
- 第 117 页的『为商业应用程序数据源注册服务器定义』
- 第 126 页的『为商业应用程序昵称创建联合视图』
- 第 112 页的『配置 WebSphere Business Integration 适配器』
- 第 118 页的『为商业应用程序数据源注册昵称』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统（32 位）』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）

## 注册 WebSphere Business Integration 包装器

注册 WebSphere Business Integration 包装器是将商业应用程序数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取商业应用程序数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。可以使用 DB2 控制中心或 DB2 命令行来注册包装器：DB2 控制中心包括一个向导，可以使用该向导来指导您完成注册包装器所需要执行的步骤。

### 先决条件:

1. 安装和配置适当的适配器。
2. 安装和配置 WebSphere MQ V5.3 (CSD 级别 5)。
3. 创建 WebSphere MQ 消息队列。
4. 如果 WebSphere MQ 管理器与 DB2 Information Integrator 不是安装在同一系统上，则在安装 DB2 Information Integrator 服务器实例的系统上安装 WebSphere MQ V5.3 (CSD 级别 5) 客户机。

### 过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如，要在使用 Windows 操作系统的联合服务器上注册一个名称为 wbi\_wrapper 的包装器，请发出以下语句：

```
CREATE WRAPPER wbi_wrapper LIBRARY 'db2wbi.dll';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 WebSphere Business Integration 包装器库文件的列表。

在这一系列任务中的下一个任务是为 WebSphere Business Integration 包装器注册服务器定义。

### 相关概念:

- 第 109 页的『WebSphere Business Integration 包装器』

### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』

### 相关参考:

- 第 116 页的『WebSphere Business Integration 包装器库文件』
- 第 116 页的『WebSphere Business Integration 包装器库文件』

## WebSphere Business Integration 包装器库文件

下表列示了 WebSphere Business Integration 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2wbi.a、libdb2wbiF.a 和 libdb2wbiU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 31. WebSphere Business Integration 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2wbi.a
Windows	%DB2PATH%\bin	db2wbi.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

#### 相关任务:

- 第 116 页的『注册 WebSphere Business Integration 包装器』

## 为商业应用程序数据源注册服务器定义

为业务对象数据源注册服务器定义是将业务对象应用程序添加至联合系统这一大型任务的一部分。

在注册包装器之后，必须注册相应的服务器。

#### 限制:

仅当指定 MQ\_CONN\_NAME 选项时才能指定 MQ\_SVRCONN\_CHANNELNAME 选项。在删除 MQ\_SVRCONN\_CHANNELNAME 选项之前不能删除 MQ\_CONN\_NAME 选项。如果未指定 MQ\_CONN\_NAME，则联合系统将使用 MQSERVER 环境变量的值。在 db2dj.ini 文件中设置 MQSERVER 环境变量。如果要编辑 db2dj.ini 文件，则必须先停止 DB2 通用数据库然后再重新启动它。

#### 过程:

要向联合系统注册商业应用程序的服务器定义，发出 CREATE SERVER 语句。

例如，要为 Siebel 商业应用程序注册服务器定义:

```
CREATE SERVER siebel_server
  VERSION 2.4
  WRAPPER wbi_wrapper
  OPTIONS ( App_Type 'siebel',
           Request_Queue 'myqueue3',
           Response_Queue 'myqueue4',
           Fault_Queue 'myqueue5',
           MQ_Manager 'mymq',
           MQ_REPONSE_TIMEOUT '55000',
           MQ_CONN_NAME '9.30.76.151(1420)',
           MQ_SVRCONN_CHANNELNAME 'SYSTEM.DEF.SVRCONN'
  )
```

在该示例中，商业应用程序是 Siebel 应用程序，它是用 APP\_TYPE 选项来标识的。有效值为 SIEBEL、PSOFT 和 SAP。VERSION 选项表示您正在使用的 WebSphere

Business Integration 适配器的版本。有效值为 2.3 和 2.4。服务器选项必须包含队列定义，如主题配置 WebSphere 业务集成适配器中所述。MQ\_RESPONSE\_TIMEOUT 的缺省值被设置为 50000 毫秒。值 -1 指定没有超时限制。

在这一系列任务中的下一个任务是为商业应用程序数据源注册昵称。

#### 相关概念:

- 第 109 页的『WebSphere Business Integration 包装器』

#### 相关任务:

- 第 112 页的『配置 WebSphere Business Integration 适配器』
- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 116 页的『注册 WebSphere Business Integration 包装器』
- 第 118 页的『为商业应用程序数据源注册昵称』

#### 相关参考:

- 第 529 页的附录 D, 『联合系统的服务器选项』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』

## 为商业应用程序数据源注册昵称

### 为商业应用程序数据源注册昵称

为商业应用程序数据源注册昵称是将商业应用程序添加至联合系统这一大型任务的一部分。

可以使用 DB2 控制中心或 DB2 命令行来注册昵称。DB2 控制中心包括一个向导，可以使用该向导来指导您完成注册昵称所需要执行的步骤。

必须创建与业务对象定义的层次结构相对应的昵称。父昵称至少包含一个子昵称。子昵称对应于包含大于 1 的基数并且嵌套在父昵称的元素中的那些元素。

#### 过程:

要从 DB2 命令行为商业应用程序数据源注册昵称，发出 CREATE NICKNAME 语句。

例如，要为 Siebel 业务对象定义注册一个称为 sieb\_ssa\_Contact\_Contact 的昵称，发出以下语句:

```
CREATE NICKNAME sieb_ssa_Contact_Contact_NN(
  Id VARCHAR(15) OPTIONS(XPATH './ns1:Id/text()'),
  TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50) OPTIONS(XPATH './ns1:FirstName/text()'),
  TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50) OPTIONS(XPATH './ns1:LastName/text()'),
  TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255) OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
  OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
  OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
  OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255) OPTIONS(XPATH './ns1:SalesRep/text()'),
```

```

State VARCHAR(255) OPTIONS(XPATH './ns1:State/text()')
FOR SERVER siebel_server
OPTIONS(XPATH '//ns1:sieb_ssa_Contact_Contact',
TEMPLATE '<ns1:sieb_ssa_Contact_Contact>
          &Id[1,1] &FirstName[0,1] &LastName[0,1]
          </ns1:sieb_ssa_Contact_Contact>',
BUSOBJ_NAME 'sieb_ssa_Contact_Contact',
NAMESPACES 'ns1="http://www.ibm.com/websphere/
            crossworlds/2002/BOSchema/
            sieb_ssa_Contact_Contact"');

```

BUSOBJ\_NAME 昵称选项是表示业务对象定义的 XML 模式定义 (XSD) 文件的名称。

**必需步骤:** 在 XSD 文件中对必需的输入列加标志

WebSphere Business Integration 适配器只能返回单个业务对象实例来作为对检索请求的响应。如果以输入业务对象的形式向适配器发出的检索请求标识应用程序中的多个业务对象, 则适配器将只返回第一个业务对象。包装器将发出一个错误来指示它未能检索整个结果集。

要确保只标识单个业务对象来作为检索请求的响应, 必须在请求业务对象中对适配器提供足够的谓词。包装器必须发送标识单个业务对象所需要的所有输入谓词。因此, 必须在昵称定义中使用正确的模板引用来标识列。下列步骤描述在使用 DB2 Control Center 来生成昵称定义之前标识正确的必需输入列而必须执行的操作:

1. 标识 SAP、Siebel 或 PeopleSoft 应用程序资源库中用来表示正在映射的应用程序实体的唯一键的那些列。

**SAP** 可以使用“SAP 业务对象资源库”来标识正在通过 WebSphere Business Integration 对象发现代理工具映射至 WebSphere Business Integration 业务对象定义的 BAPI 的必需输入参数。

**Siebel** 使用下列方法之一:

- Siebel 应用程序具有与每个“业务组件”相关联的唯一标识列, 并为“业务组件”的每个实例生成此列的十六进制值。此标识列存在于“业务组件”层次结构的最顶层, 已经使用生成的 xsd 文件中的元素的 isKey="true" 规范 (该范围在 xml 注释的 appSpecificInfo 部分) 作了标志。
- 可以使用 Siebel 工具来标识用来表示正在映射的“业务组件”的组合唯一键的那些数据库列。这些列必须全部都位于业务对象层次结构的最顶层或根层。

**PeopleSoft**

使用“应用程序设计器”工具来标识层次结构最顶层的“组件接口”中正在映射至 WebSphere Business Integration 业务对象定义的 getKey 列。

2. 编辑由 WebSphere Business Integration 对象发现代理工具为业务对象定义生成的 XSD 文件, 以便对必需的输入列作出标志。昵称和列级别的 TEMPLATE 选项这一主题提供了有关如何对列作出标志的指南。
3. 从 DB2 控制中心为业务对象定义生成昵称 DDL。

要从 DB2 控制中心为商业应用程序数据源注册昵称:

1. 展开**联合数据库对象**文件夹。
2. 展开想要为其注册昵称的包装器文件夹。

3. 展开**服务器定义**文件夹。
4. 展开想要为其注册昵称的服务器文件夹。
5. 右键单击**昵称**文件夹，并选择**创建**。
6. 在“创建昵称”窗口中，单击**发现**以定义可以帮助您选择数据源中的对象的搜索条件。
7. 指定 XML 模式定义文件，它包含您想让 DB2 Information Integrator 用户存取的业务对象的定义。
8. 单击**确定**以便根据所选择的 XML 模式定义文件来创建昵称。

DB2 控制中心将模式文件抽取到多个创建昵称 DDL 语句中，并提供适当的父子关系定义。创建的昵称表示在 XML 模式定义文件中定义的业务对象层次结构。

可选：在这一系列任务中的下一个任务是为商业应用程序昵称创建联合视图。

#### 相关概念:

- 第 120 页的『昵称级别和列级别的 TEMPLATE 选项』
- 第 111 页的『业务对象定义』
- 第 109 页的『WebSphere Business Integration 包装器』
- 第 120 页的『昵称级别和列级别的 TEMPLATE 选项』
- 第 125 页的『昵称和 XPATH 表达式』

#### 相关任务:

- 第 58 页的『为非关系数据源指定昵称列』
- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 116 页的『注册 WebSphere Business Integration 包装器』
- 第 126 页的『为商业应用程序昵称创建联合视图』

#### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## 昵称级别和列级别的 TEMPLATE 选项

本主题适用于 WebSphere® Business Integration 包装器和 Web 服务包装器。

WebSphere® Business Integration 包装器和 Web 服务包装器构建 WebSphere Business Integration 适配器和 Web 服务环境所必需的 XML 文档。这些包装器在创建昵称时需要昵称级别和列级别的模板片段，也就是 CREATE NICKNAME 语句上的 TEMPLATE 选项。这些包装器在查询规划和查询执行阶段将使用此信息。

### Web 服务包装器

对于 Web 服务包装器，必需的属性和可选属性将随着 WSDL 文档中的定义以及派生列的方式不同而有所不同。可以从一个元素或者元素的属性来派生列。

- 如果列是从元素派生的，则 minOccurs 值将确定一列是否是可选的。



- 如果 minOccurs 的值等于 0，则表示该列是可选的。
- 如果 minOccurs 的值等于 1，则表示该列是必需的。
- 如果列是从元素的属性派生的，则该属性上使用的值将确定一列是否是可选的。
- 如果属性包含值 use=optional，则该列是可选的。
- 如果属性包含值 use=required，则该列是必需的。

以下示例是模式定义中与某列相关联的一种属性:

```
<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="tns:ZooName"/>
    <xsd:element ref="tns:Count"/>
    <xsd:element ref="tns:LastModified"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>
```

## WebSphere Business Integration 包装器

对于 WebSphere Business Integration 包装器，必需的列和可选列将随着应用程序和相关联的适配器不同而有所不同。需要通过为这些列指定适当的模板选项值来标识必需的输入列和可选输入列。在使用 DB2® 控制中心来创建昵称之前，必须修改 XML 模式定义文件来标记必需的输入列和可选输入列。

### SAP BAPI

IBM® DB2 控制中心根据 XML 模式定义 (XSD) 文件中表示业务对象定义的特定标志的值来确定必需的输入列和可选输入列。

在业务对象层次结构的任何级别的元素 (父业务对象或子业务对象) 的 annotation 部分, appSpecificInfo 值中的 I 前缀指示业务对象定义映射至的 SAP BAPI 的导入参数。E 前缀指示 SAP BAPI 的导出参数。某些元素可以同时是 BAPI 的导入和导出参数。以下示例显示一个既是导入参数又是导出参数的元素:

```
<bx:appSpecificInfo>ICOMPANYCODE:ECOMPANYCODE</bx:appSpecificInfo>
```

前缀是由 WebSphere Business Integration 对象发现代理工具根据从 SAP 业务对象库中抽取的信息自动生成的。

如果表示导入参数 (appSpecificInfo 值中的 I 前缀) 的元素是使用属性 minOccurs=1 指定的, 则 DB2 控制中心将在昵称定义中将该元素标识为必需的输入参数并且将元素标记为必需的输入列。对于 SAP BAPI 的必需输入参数, WebSphere Business Integration 对象发现代理工具不会自动将 minOccurs 的值设置为 1。必须参考“SAP 业务对象库”来确定您想访问的 BAPI 的所有必需输入参数。然后, 必须通过将该属性手工设置为 minOccurs=1 来编辑 XML 模式文件中的相应元素。如果输入参数的 minOccurs 属性值仍然保持为缺省值 0, 则 DB2 控制中心将在生成的昵称层次结构将该列指定为可选输入列。

以下示例显示一个可选输入列:

```
<xsd:element name="Company_code" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICOMPANYCODE:</bx:appSpecificInfo>
```

```

        <bx:attributeInfo isForeignKey="false" isKey="true" />
    </bx:boAttribute>
</xsd:appinfo>
</xsd:annotation>
<xsd:simpleType>
    <xsd:restriction base="xsd:string">
        <xsd:maxLength value="4" />
    </xsd:restriction>
</xsd:simpleType>
</xsd:element>

```

以下示例显示一个必需的输入列:

```

<xsd:element name="Company_id" minOccurs="1">
    <xsd:annotation>
        <xsd:appinfo>
            <bx:boAttribute>
                <bx:appSpecificInfo>ICOMPANYID:</bx:appSpecificInfo>
                <bx:attributeInfo isForeignKey="true" isKey="true" />
            </bx:boAttribute>
        </xsd:appinfo>
    </xsd:annotation>
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="4" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>

```

SAP 商业应用程序的必需输入列和可选输入列是按表中显示的语法来指定的:

表 32. SAP 输入列信息的加标志模式

SAP XSD 文件中使用的标志	必需的输入列	昵称模板中的列引用
层次结构中前缀 = 'I' 并且 minOccurs=1 的任何元素	是	&columnname[1,1]
层次结构中前缀 = 'I' 并且 minOccurs=0 的任何元素	否	&columnname[0,1]

### Siebel 和 PeopleSoft

DB2 控制中心根据元素的 annotation 的 attributeInfo 部分中是否存在 isRequired 标志以及该标志的值来确定必需的输入列和可选输入列。如果没有 isRequired 标志, 该列就不是一个输入列。WebSphere Business Integration 对象发现代理工具不会在 XSD 文件中自动生成这些标志。在使用 DB2 控制中心来生成昵称 DDL 之前, 必须标识必需的输入列和可选输入列, 并在 XSD 文件中适当地标记它们。

以下示例显示 Siebel 或 PeopleSoft 业务对象定义的 XSD 文件中的必需输入列和可选输入列的标志。

---

```

<xsd:element name="sieb_ssa_Contact_Contact">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="1.0.0">
        <bx:appSpecificInfo>ON=Contact;CN=Contact</bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Id" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=Id</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false"
          isKey="true" isRequired="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
...

```

---

图 11. Siebel 业务对象定义部分 (1/2)

---

```

...
<xsd:element name="FirstName" minOccurs="1">
<xsd:annotation>
  <xsd:appinfo>
    <bx:boAttribute>
      <bx:appSpecificInfo>FN=First Name</bx:appSpecificInfo>
      <bx:attributeInfo isForeignKey="false" isKey="false"
        isRequired="false" />
    </bx:boAttribute>
  </xsd:appinfo>
</xsd:annotation>
<xsd:simpleType>
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="50" />
  </xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="LastName" minOccurs="1">
<xsd:annotation>
  <xsd:appinfo>
    <bx:boAttribute>
      <bx:appSpecificInfo>FN=Last Name</bx:appSpecificInfo>
      <bx:attributeInfo isForeignKey="false" isKey="false"
        isRequired="false" />
    </bx:boAttribute>
  </xsd:appinfo>
</xsd:annotation>
...

```

---

图 11. Siebel 业务对象定义部分 (2/2)

Siebel 和 PeopleSoft 商业应用程序的必需输入列和可选输入列是按下表中显示的语法来指定的:

表 33. Siebel 和 PeopleSoft 输入列信息的加标志模式

Siebel 和 PeopleSoft XSD 必需的输入列文件中使用的标志		昵称模板中的列引用
isRequired="true"	是	&columnname[1,1]
isRequired="false"	否	&columnname[0,1]

以下示例显示 DB2 控制中心根据标题为 *Siebel 业务对象定义部分* 的图中显示的 XSD 文件来创建的 DDL。在该图中的 XSD 文件中, isRequired 属性的值为 false。

```
CREATE NICKNAME sieb_ssa_Contact_Contact_NN(
  Id VARCHAR(15) OPTIONS(XPATH './ns1:Id/text()'),
    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50) OPTIONS(XPATH './ns1:FirstName/text()'),
    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50) OPTIONS(XPATH './ns1:LastName/text()'),
    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255) OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255) OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255) OPTIONS(XPATH './ns1:State/text()'))
FOR SERVER siebel_server
OPTIONS(XPATH '//ns1:sieb_ssa_Contact_Contact',
  TEMPLATE '<ns1:sieb_ssa_Contact_Contact>
    &Id[1,1] &FirstName[0,1] &LastName[0,1]
    </ns1:sieb_ssa_Contact_Contact>',
  BUSOBJ_NAME 'sieb_ssa_Contact_Contact',
  NAMESPACE 'ns1="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sieb_ssa_Contact_Contact"');
```

**相关概念:**

- 第 109 页的『WebSphere Business Integration 包装器』
- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

**相关任务:**

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 361 页的『为 Web 服务数据源注册昵称』

**相关参考:**

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 381 页的『Web 服务包装器的消息』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## 昵称和 XPATH 表达式

本主题适用于 WebSphere® Business Integration 包装器和 Web 服务包装器。

昵称对应于 XML 文档数据的树形结构。父昵称和子昵称对应于数据树形结构的根结构和嵌套元素。这些父昵称和子昵称是通过使用 CREATE NICKNAME 语句指定的主键和外键联系起来的。

每个昵称是由表示输出值的 XPath 表达式定义的。WebSphere Business Integration 包装器和 Web 服务包装器使用 XPath 表达式来建立 XML 文档中的数据与关系表中的各行之间的对应关系。这些 XPath 表达式标识 XML 文档中的值，并确定这些值如何与每一行的各列相对应。WebSphere Business Integration 包装器和 Web 服务包装器将只是读取 XML 文档数据。这些包装器不会更新数据。XPATH 选项包含通过 SOAP 包络和 SOAP 主体标记来查找 SOAP 消息所需要的信息。getQuote 消息包含在 SOAP 包络和主体元素中。

NICKNAME 选项 XPATH 表达式指向输出元素中的重复标记。XPath 表达式确定昵称中将具有多少行或者昵称中将包含哪些行。列选项 XPATH 表达式是相对于 NICKNAME XPATH 表达式的。列选项 XPATH 标识一行中的值。子昵称中的 NICKNAME 选项 XPATH 是相对于父昵称中的 NICKNAME 选项 XPATH 表达式的。

当创建昵称时，选择用来指定该昵称与 XML 文档之间的关联的选项。为 WebSphere Business Integration 包装器创建的昵称与 XML 模式定义 (XSD) 文档相关联。为 Web 服务包装器创建的昵称与 Web 服务描述语言 (WSDL) 文档相关联。

### 相关概念:

- 第 385 页的『什么是 XML?』
- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』
- 第 391 页的『昵称与 XML 文档之间的数据关联』

### 相关任务:

- 第 388 页的『将 XML 添加至联合系统』
- 第 394 页的『为 XML 数据源注册昵称』
- 第 399 页的『为非根昵称 (XML 包装器) 创建联合视图』
- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 118 页的『为商业应用程序数据源注册昵称』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 361 页的『为 Web 服务数据源注册昵称』

### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』
- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## 为商业应用程序昵称创建联合视图

为商业应用程序昵称创建联合视图是将商业应用程序添加至联合系统这一大型任务的一部分。

可以对描述业务对象层次结构的昵称的层次结构定义联合视图。定义联合视图可以确保用来连接商业应用程序昵称层次结构的各个部分的查询正确运行。

### 过程:

要为商业应用程序昵称创建联合视图:

1. 将每个商业应用程序昵称的视图定义为指向父昵称的路径上的所有昵称的连接。
2. 在视图的 WHERE 子句中, 将 PRIMARY\_KEY 和 FOREIGN\_KEY 列定义为连接谓词。
3. 在视图的 SELECT 列表中, 包括商业应用程序昵称的所有列 (但是使用 FOREIGN\_KEY 昵称列选项指定的列除外)。不要将指定为层次结构中的父昵称中的 PRIMARY\_KEY 的列包括在 SELECT 列表中
4. 将层次结构必需的输入列包括在 SELECT 列表中。这些列可能属于层次结构中的其它某些昵称。

以下示例显示一个基于从业务对象生成的昵称的视图。WHERE 子句中包含在 CREATE NICKNAME 语句中定义的主键和外键。

```
CREATE VIEW view1 (  
    customer, bankkey, bankact, customerno )  
AS (SELECT b.customer, b.bank_key, b.bank_acct,  
    a.customerno  
    FROM sap_bapi_customer_getdetail2_NN a,  
    sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b  
    WHERE a.NN_PKEY=b.NN_FKEY);
```

使用视图 view1 的查询必须包括必需列的谓词值, 如以下示例中所示:

```
SELECT * FROM view1  
    WHERE customerno='1234567890';
```

在这一系列任务中没有别的任务了。

### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』

### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』

## CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例

### 示例 1: 平面业务对象

第 127 页的图 12 是一个表示“Siebel 业务组件”的 WebSphere Business Integration 业务对象定义的 xsd 文件的一部分。业务对象定义层次结构由单层组成, 它只包含根业务对象。DB2 控制中心将只创建单个关系昵称来表示此业务对象定义。

在该 xsd 文件中，通过在 ID 元素的 annotation 部分添加 `isRequired="true"` 标志来将该元素标记为必需的输入列。通过添加 `isRequired="false"` 标志来将 `FirstName` 和 `LastName` 列指定为可选的输入列。

---

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<xsd:schema elementFormDefault="qualified"
  targetNamespace="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sieb_ssa_Contact_Contact"
  ...
  <xsd:element name="sieb_ssa_Contact_Contact">
    <xsd:annotation>
      <xsd:appinfo>
        <bx:boDefinition version="1.0.0">
          <bx:appSpecificInfo>ON=Contact;CN=Contact</bx:appSpecificInfo>
        </bx:boDefinition>
      </xsd:appinfo>
    </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Id" minOccurs="0">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:boAttribute>
              <bx:appSpecificInfo>FN=Id</bx:appSpecificInfo>
              <bx:attributeInfo isForeignKey="false"
                isKey="true" isRequired="true" />
            </bx:boAttribute>
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="FirstName" minOccurs="1">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:appSpecificInfo>FN=First Name</bx:appSpecificInfo>
            <bx:attributeInfo isForeignKey="false"
              isKey="false" isRequired="false" />
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="LastName" minOccurs="1">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:appSpecificInfo>FN=Last Name</bx:appSpecificInfo>
            <bx:attributeInfo isForeignKey="false"
              isKey="false" isRequired="false" />
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

---

图 12. 平面业务对象的 XML 模式文件 (1/2)

---

```

<xsd:element name="AccountId" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Account Id</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false"
      isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="PrimaryAccountName" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Primary Account Name</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="PrimaryPostalCode" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Primary Postal Code</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
</xsd:element>
<xsd:element name="PrimaryStreetAddress" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Primary Street Address</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="SalesRep" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:boAttribute>
    <bx:appSpecificInfo>FN=Sales Rep</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="State" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:boAttribute>
    <bx:appSpecificInfo>FN=State</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="ObjectEventId"
  type="xsd:string" minOccurs="0" />
</xsd:sequence>
...
</xsd:schema>

```

---

图 12. 平面业务对象的 XML 模式文件 (2/2)



DB2 控制中心将从第 127 页的图 12 中显示的 XSD 文件生成以下 CREATE NICKNAME 语句。为每个输入列都指定了 TEMPLATE 选项值。列选项模板与昵称选项模板是相关联的。在昵称级别 TEMPLATE 选项值中也引用了输入列。昵称选项模板提供了输入业务对象的结构。昵称模板值中的每个输入列引用的 minOccurs 属性的值将确定该输入列是一个必需列还是一个可选列。ID 列的引用都指定为 &Id[1,1]。FirstName 和 LastName 列的引用被指定为 &FirstName [0,1] 和 &LastName [0,1]。所有输出列都包括一个 XPATH 列选项值。昵称是针对不包含任何子代（基数为“n”的元素）的平面业务对象的：

```
CREATE NICKNAME sieb_ssa_Contact_Contact_NN(
  Id VARCHAR(15)  OPTIONS(XPATH './ns1:Id/text()'),
    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50)  OPTIONS(XPATH './ns1:FirstName/text()'),
    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50)  OPTIONS(XPATH './ns1:LastName/text()'),
    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255)
    OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255)  OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255)  OPTIONS(XPATH './ns1:State/text()'))
  FOR SERVER siebel_server
  OPTIONS(XPATH '//ns1:sieb_ssa_Contact_Contact',
    TEMPLATE '<ns1:sieb_ssa_Contact_Contact>
      &Id[1,1] &FirstName[0,1] &LastName[0,1]
      </ns1:sieb_ssa_Contact_Contact>',
    BUSOBJ_NAME 'sieb_ssa_Contact_Contact',
    NAMESPACE 'ns1="http://www.ibm.com/websphere/
      crossworl ds/2002/BOSchema/sieb_ssa_Contact_Contact"');
```

## 示例 2: 分层业务对象

在此示例中，业务对象定义由两层层次结构组成，该层次结构包括根业务对象和两个子业务对象或者三个 xsd 文件组成。只生成了两个昵称来表示业务对象定义层次结构。sap\_customeraddress 子业务对象的基数为 1，这是由元素定义中缺少 maxOccurs 属性规范来指示的。sap\_customeraddress 的所有列都包括在根昵称 sap\_bapi\_customer\_getdetail2\_NN 中。sap\_customerbankdetail 子业务对象的基数为 n，这是由元素定义中的 maxOccurs="unbounded" 规范来指示的。它将映射至单独的子昵称 sap\_bapi\_customer\_getdetail2\_sap\_customerbankdetail\_NN。子昵称通过特殊的主键 - 外键关系与根昵称相关联。

---

```

...
<xsd:element name="sap_bapi_customer_getdetail2">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="3.0.0" />
    </xsd:appinfo>
  </xsd:annotation>

  ...
  <xsd:element name="COMPANYCODE" minOccurs="0">
    <xsd:annotation>
      <xsd:appinfo>
        <bx:boAttribute>
          <bx:appSpecificInfo>ICOMPANYCODE:</bx:appSpecificInfo>
          <bx:attributeInfo isForeignKey="false" isKey="true" />
        </bx:boAttribute>
      </xsd:appinfo>
    </xsd:annotation>

    ...
  </xsd:element>
  <xsd:element name="CUSTOMERNO" minOccurs="1">
    <xsd:annotation>
      <xsd:appinfo>
        <bx:boAttribute>
          <bx:appSpecificInfo>ICUSTOMERNO:</bx:appSpecificInfo>
          <bx:attributeInfo isForeignKey="false" isKey="false" />
        </bx:boAttribute>
      </xsd:appinfo>
    </xsd:annotation>

    ...
  </xsd:element>

```

---

图 13. SAP 分层业务对象: *customer\_getdetail2* (1/2)

|

---

```

<xsd:element name="sap_customeraddress" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ECUSTOMERADDRESS</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
        <bx:childObjectInfo relationship="Containment" version="3.0.0" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
<xsd:element ref="sap_customeraddress:sap_customeraddress" />
</xsd:sequence>

</xsd:complexType>
</xsd:element>
<xsd:element name="sap_customerbankdetail" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>
          ICUSTOMERBANKDETAIL:ECUSTOMERBANKDETAIL
        </bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
        <bx:childObjectInfo relationship="Containment" version="3.0.0" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="sap_customerbankdetail:sap_customerbankdetail"
        maxOccurs="unbounded" />
    </xsd:sequence>
    <xsd:attribute name="size" type="xsd:positiveInteger"
      default="1" />
  </xsd:complexType>
</xsd:element>
<xsd:element name="ObjectEventId" type="xsd:string"
  minOccurs="0" />
</xsd:sequence>
...
<xsd:annotation>
  <xsd:appinfo>
    <bx:boVerb>
      <bx:appSpecificInfo>
        bapi.client.Bapi_customer_getdetail2
      </bx:appSpecificInfo>
    </bx:boVerb>
  </xsd:appinfo>
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Update" />
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
...

```

---

图 13. SAP 分层业务对象: *customer\_getdetail2* (2/2)

---

```

<xsd:element name="sap_customeraddress">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="3.0.0">
        <bx:appSpecificInfo>:ECUSTOMERADDRESS</bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CUSTOMER" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ECUSTOMER</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="NAME" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ENAME</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="CITY" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ECITY</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>

```

---

图 14. SAP 分层业务对象: *customer\_address* (1/2)

I

---

```

<xsd:element name="POSTL_CODE" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:EPOSTL_CODE</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="STREET" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ESTREET</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="REGION" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:EREGION</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
</xsd:sequence>
...

```

---

图 14. SAP 分层业务对象: *customer\_address* (2/2)

1

---

```
...
<xsd:element name= "sap_customerbankdetail ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version= "3.0.0 ">
        <bx:appSpecificInfo>
          ICUSTOMERBANKDETAIL:ECUSTOMERBANKDETAIL
        </bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name= "CUSTOMER " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICUSTOMER:ECUSTOMER</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "true " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
```

---

图 15. SAP 分层业务对象: *bank\_detail* (1/2)

|

---

```

<xsd:element name= "BANK_KEY " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>IBANK_KEY:EBANK_KEY</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name= "BANK_ACCT " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>IBANK_ACCT:EBANK_ACCT</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name= "CTRL_KEY " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICTRL_KEY:ECTRL_KEY</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name= "BANK_REF " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>IBANK_REF:EBANK_REF</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
...

```

---

图 15. SAP 分层业务对象: *bank\_detail* (2/2)

DB2 控制中心将从三个 SAP XSD 文件中生成两个昵称，如第 136 页的图 16 和第 137 页的图 17 中所示。由于 customerno 元素的 XSD 文件规范，customerno 列在 sap\_bapi\_customer\_getdetail2\_NN 昵称的昵称级别模板中被标记为必需的输入列。Customerno 以 I 前缀和 minOccurs=1 属性值作为标志。

---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
CUSTOMER VARCHAR(10)
    OPTIONS(XPATH './ns3:sap_customeraddress/
        ns1:sap_customeraddress/ns1:CUSTOMER/text()'),
NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
        ns1:sap_customeraddress/ns1:NAME/text()'),
CITY VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
        ns1:sap_customeraddress/ns1:CITY/text()'),
POSTL_CODE VARCHAR(10)
    OPTIONS(XPATH './ns3:sap_customeraddress/
        ns1:sap_customeraddress/ns1:POSTL_CODE/text()'),
STREET VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
        ns1:sap_customeraddress/ns1:STREET/text()'),
REGION VARCHAR(3)
    OPTIONS(XPATH './ns3:sap_customeraddress/
        ns1:sap_customeraddress/ns1:REGION/text()'),
NN_KEY VARCHAR(16) FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'),
COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()',
    TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()',
    TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
ObjectEventId VARCHAR(48) OPTIONS(XPATH './ns3:ObjectEventId/text()'))
FOR SERVER sap_server
OPTIONS(XPATH '/ns3:sap_bapi_customer_getdetail2',
TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
    &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
    &COMPANYCODE[0,1]
    &CUSTOMERNO[1,1]
</ns3:sap_bapi_customer_getdetail2>',
BUSOBJ_NAME 'sap_bapi_customer_getdetail2',
NAMESPACES 'ns1="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sap_customeraddress",
    ns2="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sap_customerbankdetail",
    ns3="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sap_bapi_customer_getdetail2"');

```

---

图 16. getdetail2 nickname

|



```

CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
  CUSTOMER VARCHAR(10)  OPTIONS(XPATH './ns2:CUSTOMER/text()',
    TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15)  OPTIONS(XPATH './ns2:BANK_KEY/text()',
    TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18)  OPTIONS(XPATH './ns2:BANK_ACCT/text()',
    TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2)   OPTIONS(XPATH './ns2:CTRL_KEY/text()',
    TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20)  OPTIONS(XPATH './ns2:BANK_REF/text()',
    TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
  NN_FKEY VARCHAR(16)   FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN')
  FOR SERVER sap_server
  OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
    TEMPLATE '<ns3:sap_customerbankdetail>
      <ns2:sap_customerbankdetail>
        &CUSTOMER[0,1]
        &BANK_KEY[0,1]
        &BANK_ACCT[0,1]
        &CTRL_KEY[0,1]
        &BANK_REF[0,1]
      </ns2:sap_customerbankdetail>
    </ns3:sap_customerbankdetail>',
  NAMESPACE 'ns1="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sap_customeraddress",
    ns2="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sap_customerbankdetail",
    ns3="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sap_bapi_customer_getdetail2"');

```

图 17. 客户银行详细信息昵称

### 示例 3: 主键和外键

PRIMARY\_KEY 和 FOREIGN\_KEY 列用来定义父昵称与子昵称之间的关系。每个父昵称必须具有主键列选项。使用引用父昵称的主键列的外键列选项来定义父昵称的子昵称。一个昵称可以具有多个子昵称，但是一个昵称只能具有一个父昵称。

WebSphere Business Integration 包装器的主键值和外键值仅在单个查询中有效，并且是唯一的。不能使用主键和外键来通过另一个查询来检索行。值不能在另一个表中继续有效，这是因为如果该表是使用多个查询来填充的，则不能保证值的唯一性。

下列 CREATE NICKNAME 语句是从第 130 页的图 13、第 132 页的图 14 和第 134 页的图 15 中显示的 XML 模式定义文件中派生出来的。外键 nn\_fkey 使子昵称 sap\_bapi\_customer\_getdetail2\_sap\_customerbankdetail\_nn 唯一与父昵称 sap\_bapi\_customer\_getdetail2\_nn 相关联。在昵称选项模板结构中，父昵称还使用对子昵称的引用。

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
  CUSTOMER VARCHAR(10)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:CUSTOMER/text()'),
  NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:NAME/text()'),
  ...
  NN_PKEY VARCHAR(16) FOR BIT DATA NOT NULL
    OPTIONS(PRIMARY_KEY 'YES'),

```

```

...
TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
        &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
        &COMPANYCODE[0,1]
        &CUSTOMERNO[1,1]
</ns3:sap_bapi_customer_getdetail2>',
...
CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
  CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()'),
    TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()'),
    TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()'),
    TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()'),
    TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()'),
    TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
  NN_FKEY VARCHAR(16) FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
FOR SERVER sap_server
OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
  TEMPLATE '<ns3:sap_customerbankdetail>
        <ns2:sap_customerbankdetail>
        &CUSTOMER[0,1]
        &BANK_KEY[0,1]
        &BANK_ACCT[0,1]
        &CTRL_KEY[0,1]
        &BANK_REF[0,1]
        </ns2:sap_customerbankdetail>
</ns3:sap_customerbankdetail>',
...

```

#### 示例 4: 使用名称空间来解析 XPath 表达式前缀

NAMESPACES 选项是名 - 值对的以逗号分隔的列表。它使用在 XML 模式中定义的名称空间 URI 来解析 XPath 表达式中使用的前缀。这些 XPath 表达式应用于从 WebSphere Business Integration 适配器返回的业务对象 (XML 文档)。以下示例包括名称空间前缀以及这些前缀的定义:

```

CREATE NICKNAME sap_customer
(
  sap_customeraddress_CUSTOMER VARCHAR(10)
    OPTIONS(XPATH './ns5:sap_customeraddress/
        ns2:sap_customeraddress/ns2:CUSTOMER/text()'),
  sap_customeraddress_NAME VARCHAR(35)
    OPTIONS(XPATH './ns5:sap_customeraddress/
        ns2:sap_customeraddress/ns2:NAME/text()'),
...
  sap_bapi_customer_getdet1_PKEY VARCHAR(16) FOR BIT DATA NOT NULL
    OPTIONS(PRIMARY_KEY 'YES'),
  COMPANYCODE VARCHAR(4)
    OPTIONS(XPATH './ns5:COMPANYCODE/text()'),
    TEMPLATE '<ns5:COMPANYCODE>&column</ns5:COMPANYCODE>'),
  CUSTOMERNO VARCHAR(10)
    OPTIONS(XPATH './ns5:CUSTOMERNO/text()'),
    TEMPLATE '<ns5:CUSTOMERNO>&column</ns5:CUSTOMERNO>'),
  ObjectEventId VARCHAR(48)
    OPTIONS(XPATH './ns5:ObjectEventId/text()')
)

```

```

FOR SERVER SAP_SOURCE
OPTIONS (
  XPATH '//ns5:sap_bapi_customer_getdetail2',
  TEMPLATE
    '<ns5:sap_bapi_customer_getdetail2>
      &customerbankdetail_NN[0,1] &COMPANYCODE[0,1] &CUSTOMERNO[1,1]
    </ns5:sap_bapi_customer_getdetail2>',
  BUSOBJ_NAME 'sap_bapi_customer_getdetail2',
  NAMESPACE '
ns2="http://www.ibm.com/websphere/
  crossworlds/2002/BOSchema/sap_customeraddress",
...
ns5="http://www.ibm.com/websphere/
  crossworlds/2002/BOSchema/sap_bapi_customer_getdetail2",
ns6="http://www.ibm.com/websphere/
  crossworlds/2002/BOSchema/sap_return"
);

```

#### 相关概念:

- 第 109 页的『WebSphere Business Integration 包装器』

#### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 118 页的『为商业应用程序数据源注册昵称』

#### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

---

## 对于商业应用程序和 Web 服务的包装器的查询限制

### Equal 谓词

在输入列上唯一有效的谓词是 equal 谓词。对于输出列，任何谓词都有效。

以下示例将返回一个错误，将发出一条消息来指示谓词在该列上不受支持。在此示例中，zipcode 列是一个输入列:

```
SELECT return FROM gettemp WHERE zipcode<'95141'
```

以下示例显示了一个在输入列上使用 equal 谓词的有效查询。customers 昵称与包含客户标识的本地 DB2 UDB 表相连接。该查询在 Sales 列（这是一个“仅输出”列）上包含附加谓词。

```

SELECT a.name, a.address
FROM customers a, local_table b
WHERE
  a.customer_id=b.custid AND
  a.Sales > 300000;

```

### 必需的输入列的谓词

对于您引用的昵称层次结构，必须为 SQL 查询中的所有必需的输入列提供全等谓词值。对于违反此限制的所有查询，包装器将返回 SQLCODE 901。

### IN 或 OR 谓词

对于 WebSphere Business Integration 包装器和 Web 服务包装器，输入列中不允许存在 IN 或 OR 谓词。

下列示例都是一些无效查询。customers 昵称具有一个必需的输入列 customer\_id:

```
SELECT * FROM customers
  WHERE customer_id IN (12345, 67890, 11223);
SELECT * FROM customers
  WHERE customer_id IN (SELECT custid FROM local_table); )
```

但是，对于 WebSphere Business Integration 包装器，如果您使用 SPECIFICATION ONLY 参数为必需的输入列定义一个唯一索引，则可以对必需的输入列使用 IN 列表谓词:

```
CREATE UNIQUE INDEX myuindex ON customers(customer_id) SPECIFICATION ONLY;
```

### 可选输入列的连接

以下示例演示了对于连接可选输入列的限制。不能从本地表或昵称连接可选输入列。如果 WSDL 生成了一个可选的输入昵称列，而您需要在连接中使用该列，则必须通过编辑 DDL 来将该列更改为必需的输入列。

在此示例中，名为 order 的 Web 服务包装器昵称是在将 shipping\_method 作为可选输入列的情况下创建的。因为以下语句在谓词中使用了文字，因此它是一个有效查询:

```
SELECT * FROM order
  WHERE part="hammer" AND shipping_method="FEDEX";
```

但是，如果查询中包括一个名为 orderparts 的本地表（该表定义部件和装运方法），并且该表包含一个名为 shipping\_method 的可选列，则以下语句无效:

```
SELECT * FROM
  order o, orderparts op
  WHERE
    o.part="hammer" AND
    o.shipping_method=op.shipping_method
```

对于 WebSphere Business Integration 包装器，昵称的可选输入列上的谓词可能会下推至 WebSphere Business Integration 适配器。DB2 UDB 可以决定在本地对从应用程序数据源访存的行应用那些谓词。为了确保始终将给定输入列的谓词下推至适配器，应将输入列声明为必需的输入列。昵称层次结构上的每个查询都必须对必需的输入列包括谓词值。

为了确保获得有效结果，互相连接的输入列必须是 Web 服务包装器的必需列。

### 外连接

使用父昵称中的主键的昵称与子昵称列中的外键之间的外连接是不受支持的。

如果 XML 文档中的父元素不包含任何子元素，而您使用了父称与子昵称之间的内连接时，将不会为该元素返回任何行。例如，对于给定的客户，如果 SAP 系统中没有他的任何银行详细信息，则对于特定客户不会为 sap\_bapi\_customer\_getdetail2\_sap\_customerbankdetail\_NN 昵称返回任何行。

下列 CREATE NICKNAME 语句将定义在示例查询中使用的列:

---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
...
NAME VARCHAR(35)
  OPTIONS(XPATH './ns3:sap_customeraddress/
            ns1:sap_customeraddress/ns1:NAME/text()'),
...
NN_PKEY VARCHAR(16) OPTIONS(PRIMARY_KEY 'YES'),
COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()'),
  TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()'),
  TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
...
FOR SERVER sap_server
OPTIONS(XPATH '//ns3:sap_bapi_customer_getdetail2',
  TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
            &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
            &COMPANYCODE[0,1]
            &CUSTOMERNO[1,1]
            </ns3:sap_bapi_customer_getdetail2>',
...

```

---

图 18. *getdetail2* 昵称的一部分内容

---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()'),
  TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()'),
  TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()'),
  TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()'),
  TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()'),
  TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
NN_FKEY VARCHAR(16) OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
FOR SERVER sap_server
OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
  TEMPLATE '<ns3:sap_customerbankdetail>
            <ns2:sap_customerbankdetail>
            &CUSTOMER[0,1]
            &BANK_KEY[0,1]
            &BANK_ACCT[0,1]
            &CTRL_KEY[0,1]
            &BANK_REF[0,1]
            </ns2:sap_customerbankdetail>
            </ns3:sap_customerbankdetail>',
...

```

---

图 19. 客户银行详细信息昵称的一部分内容

在以下示例中，因为两个昵称之间存在内连接条件，所以查询将不会返回任何件：

```

SELECT a.name, b.bank_key
FROM sap_bapi_customer_getdetail2_NN a,
     sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
WHERE a.customerno='1234567890'
AND a.NN_PKEY=b.NN_FKEY;

```

如果 WebSphere Business Integration 包装器或者 Web 服务包装器昵称定义中包含必需的输入列，则不支持此昵称与任何其它本地 DB2 UDB 表或其它昵称之间的左外连接。

#### 相关概念:

- 第 120 页的『昵称级别和列级别的 TEMPLATE 选项』

#### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』

#### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 378 页的『Web 服务数据源 - 示例查询』

---

## 商业应用程序数据源 - 示例查询

### 示例 1: 连接父昵称与子昵称

如果 SELECT 语句包含子昵称列，则必须指定与父昵称的连接谓词。必须设置连接条件以维护昵称层次结构中的父子关系。通过包含引用的子昵称与层次结构的父昵称之间的层次结构关联，来为层次结构中的每对父子昵称指定主键和外键连接条件。

下列查询是无效的，这是因为它们都不包含维护昵称层次结构所需要的所有元素：

```
SELECT * FROM <child_nickname>;
SELECT b.col1
FROM <parent_nickname> a,<child_nickname> b
WHERE a.required_column=<value>;
```

以下是一个维护了昵称层次结构的有效查询的示例：

```
SELECT b.col1, a.col2
FROM <parent_nickname> a,<child_nickname> b
WHERE a.primary_key_column=b.foreign_key_column
AND a.required_column=<value>;
```

在以下示例中，需要的所有父输入列和子输入列都包括在 WHERE 子句中的谓词中。WHERE 子句中包括用来指定父主键列与子外键列之间的相等性的连接谓词：

```
SELECT a.customer, a.name, b.bank_key, b.bank_acct
FROM sap_bapi_customer_getdetail2_NN a,
sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
WHERE a.nn_pkey=b.nn_fkey
AND a.customer = 'ABC'
```

### 示例 2: 视图

可以创建两种类型的视图：

- 从子昵称的各列中派生的视图，以便您可以直接对子昵称发出查询，而不必在查询中包括父子连接条件。

```

| CREATE VIEW view1 (
| customer, bankkey, bankact, customerno )
| AS (SELECT b.customer, b.bank_key, b.bank_acct,
| a.customerno
| FROM sap_bapi_customer_getdetail2_NN a,
| sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
| WHERE a.NN_PKEY=b.NN_FKEY);

```

视图定义包括昵称层次结构必需的输入列 customerno。使用视图 view1 的查询必须包括必需列的谓词值，如以下示例中所示：

```

| SELECT * FROM view1
| WHERE customerno='1234567890';

```

- 包括子昵称和父昵称的所有列（主键列和外键列除外）的全局视图。

### 示例 3: 必需的输入列

查询必须包括所有必需的输入列的谓词值。必需的输入列就是具有 TEMPLATE 列选项定义并且在昵称 TEMPLATE 选项值中引用值为 [1,1] 的那些列。对于谓词中不包括必需的输入列的任何查询，包装器都将返回错误。

以下示例显示了一个无效查询。customers 昵称包含必需的输入列 customer\_id。

```

| SELECT * FROM customers;

```

以下示例显示了一个有效查询。

```

| SELECT * FROM customers WHERE customer_id = 123;

```

以下示例显示 DB2 UDB 中的一个本地表，它在 local\_table 表的 custid 列中包含客户标识。此示例是 WebSphere Business Integration 昵称与该本地表之间的一个内连接。

```

| SELECT a.name, a.address
| FROM customers a, local_table b
| WHERE a.customer_id=b.custid;

```

#### 相关概念:

- 第 109 页的『WebSphere Business Integration 包装器』

#### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 118 页的『为商业应用程序数据源注册昵称』

#### 相关参考:

- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』





---

## 第 9 章 配置对 DB2 系列数据源的存取

本章说明如何配置联合服务器以存取存储在 DB2 系列数据库中的数据。这些数据库包括:

- DB2 UDB Linux 版、UNIX 版和 Windows 版
- DB2 UDB z/OS 和 OS/390 版
- DB2 UDB iSeries 版
- DB2 Server VM 和 VSE 版

本章说明如何配置联合服务器以存取存储在 DB2 系列数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 DB2 系列数据源的存取。

本章列示配置联合服务器时需要执行的任务, 包含您需要的 SQL 语句的示例, 而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 DB2 系列数据源添加至联合服务器

#### 将 DB2 系列数据源添加至联合服务器

要配置联合服务器以存取 DB2 系列数据源, 必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 DB2 系列数据源。DB2 控制中心包括一个向导, 可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

#### 先决条件:

- 为联合配置的 DB2 服务器。
- 联合服务器上必须存在联合数据库

#### 限制:

如果您正在存取存储在 DB2 Linux 版、UNIX 版和 Windows 版版本 8.1 中的数据, 则不能为 DB2 数据源别名创建昵称。

#### 过程:

要将 DB2 数据源添加至联合服务器:

1. 对节点进行编目。
2. 对远程数据库进行编目。
3. 注册包装器。
4. 注册服务器定义。
5. 创建用户映射。
6. 测试与 DB2 服务器的连接。
7. 为 DB2 表和视图注册昵称。

#### 相关概念:

- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 38 页的『检查 FEDERATED 参数』
- 第 44 页的『创建联合数据库』
- 第 146 页的『在联合节点目录中编目节点条目』
- 第 147 页的『在联合系统数据库目录中编目远程数据库』
- 第 147 页的『注册 DB2 包装器』
- 第 149 页的『为 DB2 数据源注册服务器定义』
- 第 151 页的『为 DB2 数据源创建用户映射』
- 第 152 页的『测试与 DB2 数据源服务器的连接』
- 第 153 页的『为 DB2 表和视图注册昵称』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 在联合节点目录中编目节点条目

在联合节点目录中编目节点条目是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

要指向 DB2 数据源所在的位置，在联合服务器的节点目录中编目一个条目。联合服务器使用此条目来确定连接至 DB2 数据源的正确存取方法。

#### 过程:

要在联合节点目录中编目一个节点条目:

1. 确定将使用的通信协议。
2. 发出适当的命令来编目节点条目。
  - 如果通信协议为“传输控制协议/网际协议”(TCP/IP)，则发出 **CATALOG TCPIP NODE** 命令。

例如:

```
CATALOG TCPIP NODE DB2NODE REMOTE SYSTEM42 SERVER DB2TCP42
```

*DB2NODE* 值是为正在编目的节点指定的名称。*REMOTE SYSTEM42* 是数据源所驻留的系统的主机名。*SERVER DB2TCP42* 是服务器数据库管理器实例的服务名称或主端口号。如果使用了服务名称，则它是区分大小写的。

- 如果通信协议为 SNA，则发出 **CATALOG APPC NODE** 命令。

例如:

```
CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

*DB2NODE* 值是为正在编目的节点指定的名称。*REMOTE DB2CPIC* 是远程伙伴节点的 SNA 伙伴逻辑单元 (LU) 名。*SECURITY PROGRAM* 指定用户名和密码都将包括在发送给伙伴 LU 的分配请求中。

在这一系列任务中的下一个任务是在联合系统数据库目录中编目远程数据库。

#### 相关任务:

- 第 147 页的『在联合系统数据库目录中编目远程数据库』

## 在联合系统数据库目录中编目远程数据库

在联合系统数据库目录中编目远程数据库是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

通过在联合服务器系统数据库目录中编目远程数据库来指定联合服务器将连接至哪个 DB2 数据源数据库。

#### 过程:

要在联合服务器系统数据库目录中编目远程数据库:

1. 使用“客户机配置助手”(CCA)。

对于 UNIX 上的联合服务器，可以通过另一种方法来使用 **CATALOG DATABASE** 命令。例如:

```
CATALOG DATABASE DB2DB390 AS CLIENTS390 AT NODE DB2NODE AUTHENTICATION DCS
```

值 *DB2DB390* 是您在联合服务器系统数据库目录中正在编目的远程数据库的名称。*AS CLIENTS390* 是正在编目的数据库的别名。如果未指定别名，则数据库管理器使用数据库名称 (例如, *DB2DB390*) 作为别名。*AT NODE DB2NODE* 是在节点目录中对节点条目进行编目时所指定的节点的名称。*AUTHENTICATION SERVER* 指定在 DB2 数据源节点上进行认证。

2. 如果远程数据库的名称超过了八个字符，则必须通过发出 **CATALOG DCS DATABASE** 命令来创建 DCS 目录条目。例如:

```
CATALOG DCS DATABASE SALES400 AS SALES_DB2DB400
```

值 *SALES400* 是要编目的远程数据库的别名。此名称应当跟与远程节点相关联的联合服务器系统数据库目录中的一个条目的名称相匹配。它就是您在 **CATALOG DATABASE** 命令中输入的同一名称。*AS SALES\_DB2DB400* 是想要编目的目标主机数据库的名称。

在这一系列任务中的下一个任务是注册 DB2 包装器。

#### 相关任务:

- 第 146 页的『在联合节点目录中编目节点条目』
- 第 147 页的『注册 DB2 包装器』

## 注册 DB2 包装器

注册 DB2 包装器是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 DB2 系列数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

### 过程:

要注册包装器，发出 CREATE WRAPPER 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER DRDA
```

**建议:** 使用称为 DRDA 的缺省包装器名称。当您使用缺省名称来注册包装器时，联合服务器将自动采用与该包装器名称相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突，则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于缺省名称的名称，则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 db2\_wrapper 的包装器，请发出以下语句:

```
CREATE WRAPPER db2_wrapper LIBRARY 'libdb2drda.a'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 DB2 包装器库文件的列表。

在这一系列任务中的下一个任务是为 DB2 包装器注册服务器定义。

### 相关任务:

- 第 149 页的『为 DB2 数据源注册服务器定义』

### 相关参考:

- 第 148 页的『DB2 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## DB2 包装器库文件

下表列示了 DB2 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2drda.a、libdb2drdaF.a 和 libdb2drdaU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 34. DB2 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2drda.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2drda.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2drda.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2drda.so
Windows	%DB2PATH%\bin	db2drda.dll

| %DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows  
| 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

| **相关任务:**

- | • 第 147 页的『注册 DB2 包装器』

## 为 DB2 数据源注册服务器定义

为 DB2 数据源注册服务器定义是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 DB2 服务器。当注册服务器定义时，联合服务器将连接至 DB2 服务器并将程序包绑定至数据库。因为有关权限和密码的信息未存储在联合全局目录中，所以，在服务器定义中必须包括这些信息。

**过程:**

要为 DB2 数据源注册服务器定义，发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name TYPE DB2/ZOS VERSION 6 WRAPPER DRDA  
AUTHORIZATION "name1" PASSWORD "passwd1"  
OPTIONS (DBNAME 'db_name')
```

为服务器指定的名称必须是唯一的。不允许存在重复的服务器名。

您指定的 VERSION 选项是想要访问的 DB2 数据库服务器的版本。

WRAPPER 参数的名称必须是您在 CREATE WRAPPER 语句中指定的名称。

尽管数据库名称被指定为 CREATE SERVER 语句中的一个选项，但是，它对于 DB2 数据源是必需的。

当发出 CREATE SERVER 语句时，联合服务器将测试与 DB2 数据源服务器的连接。

在注册服务器定义之后，可以通过发出 ALTER SERVER 语句来添加或删除服务器选项。

在这一系列任务中的下一个任务是为 DB2 数据源创建用户映射。

**相关任务:**

- 第 145 页的『将 DB2 系列数据源添加至联合服务器』
- 第 151 页的『为 DB2 数据源创建用户映射』

**相关参考:**

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 5 页的『受支持的数据源』
- 第 150 页的『CREATE SERVER 语句 - DB2 包装器的示例』

## CREATE SERVER 语句 - DB2 包装器的示例

此主题提供了几个示例来说明如何使用 CREATE SERVER 语句来为 DB2 系列数据源上的包装器注册服务器。此主题包括一个完整的示例，该示例说明如何使用所有必需的参数来创建服务器，它还包括一个带有可选的服务器选项的示例。

### 完整示例:

以下示例说明了如何使用 CREATE SERVER 语句来为 DB2 包装器创建服务器定义:

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
AUTHORIZATION "spalten" PASSWORD "db2guru"
OPTIONS (DBNAME 'CLIENTS390')
```

### DB2SERVER

为 DB2 数据库服务器指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

### TYPE DB2/ZOS

指定正在对其配置存取的数据源服务器的类型。

### VERSION 6

想要访问的 DB2 数据库服务器的版本。

### WRAPPER DRDA

在 CREATE WRAPPER 语句中指定的名称。

### AUTHORIZATION "spalten"

数据源中的授权标识。此标识在数据源中必须具有 BINDADD 权限。此值是区分大小写的。

### PASSWORD "db2guru"

与数据源中的授权标识相关联的密码。此值是区分大小写的。

### DBNAME 'CLIENTS390'

想要存取的数据源数据库的别名。此别名是在您使用 CATALOG DATABASE 命令对数据库进行编目时定义的。此值是区分大小写的。

对于 DB2 数据源，此数据库名称是必需的。

### 服务器选项示例:

当您注册服务器定义时，可以在 CREATE SERVER 语句中指定更多服务器选项。这些选项包括一般的服务器选项和特定于 DB2 数据源的服务器选项。

以下示例说明了具有 CPU\_RATIO 选项的服务器定义。

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
AUTHORIZATION "spalten" PASSWORD "db2guru"
OPTIONS (DBNAME 'CLIENTS390', CPU_RATIO '0.001')
```

如果将 CPU\_RATIO 选项设置为“0.001”，则它指示远程数据源中的 CPU 可用容量是联合服务器的 1000 倍。

### 相关任务:

- 第 149 页的『为 DB2 数据源注册服务器定义』

### 相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 523 页的附录 C, 『SQL 语句中的有效服务器类型』

## 为 DB2 数据源创建用户映射

为 DB2 数据源创建用户映射是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

当您尝试访问 DB2 服务器时, 联合服务器将使用对于数据源有效的用户标识和密码来与该数据源建立连接。必须定义联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联 (用户映射)。为将访问联合系统以便发送分布式请求的每个用户标识创建用户映射。

### 过程:

要将本地用户标识映射至 DB2 服务器用户标识和密码, 发出 CREATE USER MAPPING 语句。

例如:

```
CREATE USER MAPPING FOR USERID SERVER DB2SERVER
      OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

REMOTE\_AUTHID 是连接授权标识, 而不是绑定授权标识。

在这一系列任务中的下一个任务是测试与 DB2 数据源服务器的连接。

### 相关任务:

- 第 152 页的『测试与 DB2 数据源服务器的连接』

### 相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)
- 第 151 页的『CREATE USER MAPPING 语句 - DB2 包装器的示例』

## CREATE USER MAPPING 语句 - DB2 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将本地用户标识映射至 DB2 服务器用户标识和密码。此主题包括一个具有所有必需参数的完整示例, 还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

### 完整示例:

以下示例说明了如何将本地用户标识映射至 DB2 服务器用户标识:

```
CREATE USER MAPPING FOR DB2USER SERVER DB2SERVER
      OPTIONS (REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day2night')
```

#### DB2USER

指定您正在映射至在 DB2 系列数据源服务器中定义的用户标识的本地用户标识。

#### SERVER DB2SERVER

指定您在 CREATE SERVER 语句中定义的 DB2 系列数据源服务器的名称。

### **REMOTE\_AUTHID** 'db2admin'

指定您正在将 *DB2USER* 映射所至的 DB2 系列数据源服务器中的连接授权用户标识。使用单引号来保留此值的大小写（除非您在 `CREATE SERVER` 语句中将 `FOLD_ID` 服务器选项设置为 'U' 或 'L'）。

### **REMOTE\_PASSWORD** 'day2night'

指定与 'db2admin' 相关联的密码。使用单引号来保留此值的大小写（除非您在 `CREATE SERVER` 语句中将 `FOLD_PW` 服务器选项设置为 'U' 或 'L'）。

#### 专用寄存器示例:

以下是一个包括专用寄存器 `USER` 的 `CREATE USER MAPPING` 语句的示例:

```
CREATE USER MAPPING FOR USER SERVER DB2SERVER
  OPTIONS (REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day2night')
```

可以使用 DB2 专用寄存器 `USER` 来将正在发出 `CREATE USER MAPPING` 语句的人员的授权标识映射至在 `REMOTE_AUTHID` 用户选项中指定的数据源授权标识。

#### 相关任务:

- 第 151 页的『为 DB2 数据源创建用户映射』

#### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）

## 测试与 DB2 数据源服务器的连接

测试与 DB2 数据源服务器的连接是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

可以使用您定义的服务器定义和用户映射来测试与 DB2 服务器的连接。

#### 过程:

要测试连接:

1. 打开传递会话以便对 DB2 系统表发出 `SQL SELECT` 语句。

例如:

- 在 DB2 z/OS 和 OS/390 版上:

```
SET PASSTHRU server_name
SELECT count(*) FROM sysibm.systables
SET PASSTHRU RESET
```
- 在 DB2 iSeries 版上:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM qsys2.systables
SET PASSTHRU RESET
```

如果该 `SQL SELECT` 语句返回了一个计数，则表示正确设置了服务器定义和用户映射。

2. 如果 `SQL SELECT` 语句返回了错误，则可能需要:
  - 检查远程服务器以确保它已启动。
  - 检查远程服务器上的侦听器以确保为人局连接配置了该侦听器。



- 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 DB2 服务器的连接是有效的。
- 检查节点和数据库的 DB2 目录条目。
- 检查 DB2 联合变量的设置以验证您是否可以访问远程 DB2 服务器。这些变量包括系统环境变量、db2dj.ini 变量和“DB2 概要文件注册表”（db2set）DB2COMM 变量。
- 检查服务器定义。如果需要的话，删除服务器定义，然后再次创建它。
- 检查用户映射。如果需要的话，改变用户映射，或者创建另一个用户映射。

在这一系列任务中的下一个任务是为 DB2 表和视图注册昵称。

#### 相关任务:

- 第 153 页的『为 DB2 表和视图注册昵称』
- 第 52 页的『设置数据源环境变量』

#### 相关参考:

- 『ALTER USER MAPPING statement』（*SQL Reference, Volume 2*）

## 为 DB2 表和视图注册昵称

为 DB2 表和视图注册昵称是将 DB2 系列数据源添加至联合服务器这一大型任务的一部分。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是当您使用 CREATE NICKNAME 语句为数据源对象注册昵称时收集的。联合数据库验证数据源中的对象是否存在，然后尝试收集现有数据源的统计数据。对于优化器很有用的信息是从数据源目录中读取的，并且放入联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息，所以，建议在注册昵称之前（使用与 RUNSTATS 命令等价的数据源命令）更新数据源中的统计信息。

使用 CREATE NICKNAME 语句来为位于 DB2 系列数据源中的视图或表注册昵称。当您查询 DB2 系列数据源时，使用这些昵称而不是数据源对象的名称。

#### 限制:

不能对 DB2 别名创建昵称。

#### 过程:

要注册昵称，发出 CREATE NICKNAME 语句。

例如:

```
CREATE NICKNAME DB2NICKNAME FOR DB2SERVER.remote_schema.remote_table
```

昵称的长度最多可以为 128 个字符。

对于您想为其注册昵称的每个 DB2 表或视图重复此步骤。

当注册昵称时，联合服务器将使用连接来查询数据源目录。此查询将使用昵称来测试与数据源的连接。如果该连接不工作，则您将接收到错误消息。

#### 相关参考:

- 『RUNSTATS Command』 ( *Command Reference* )
- 『CREATE NICKNAME statement』 ( *SQL Reference, Volume 2* )
- 第 154 页的『CREATE NICKNAME 语句 - DB2 包装器的示例』

## CREATE NICKNAME 语句 - DB2 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE NICKNAME 语句来为您想存取的 DB2 表或视图注册昵称。

以下示例显示了一个 CREATE NICKNAME 语句:

```
CREATE NICKNAME DB2SALES FOR DB2SERVER.SALESDATA.EUROPE
```

#### *DB2SALES*

用来标识 DB2 表或视图的唯一昵称。

**注:** 昵称是一个由两部分组成的名称, 它包括模式和昵称。如果在注册昵称时省略了模式, 则昵称的模式将为创建该昵称的用户的授权标识。

#### *DB2SERVER.SALESDATA.EUROPE*

远程对象的由三部分组成的标识:

- *DB2SERVER* 是您在 CREATE SERVER 语句中为 DB2 数据库服务器指定的名称。
- *SALESDATA* 是表或视图所属于的远程模式的名称。此值是区分大小写的。
- *EUROPE* 是您想存取的远程表或视图的名称。

#### 相关任务:

- 第 485 页的『改变昵称』
- 第 153 页的『为 DB2 表和视图注册昵称』

#### 相关参考:

- 『CREATE NICKNAME statement』 ( *SQL Reference, Volume 2* )

---

## 第 10 章 配置对 Documentum 数据源的存取

本章说明如何配置联合服务器以存取存储在 Documentum 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Documentum 数据源的存取。

本章包含下列内容:

- 说明什么是 Documentum
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 Documentum 包装器相关联的错误消息

---

### 什么是 Documentum?

Documentum 是一种文档管理软件，它可以管理文档内容和属性，例如，检入、检出、工作流和版本管理。Documentum 产品是基于关系数据库构建的三层客户机 / 服务器系统。

Docbase 是一个 Documentum 资源库，它存储文档内容、属性、关系、版本、复制、格式、工作流和安全性。Documentum 查询语言 (DQL) 是一种用来查询 Documentum 数据的扩展 SQL 语言。Docbase 等同于 Oracle 实例或 DB2<sup>®</sup> 数据库加上文档内容文件。元数据被存储在底层关系数据库管理系统 (RDBMS) 中，内容是作为二进制大对象 (BLOB) 存储在数据库中，或者作为文件存储在服务器系统的文件系统。有关 Documentum 的更多信息，参阅 Documentum 手册。

Documentum 的包装器允许您将 Documentum 数据源添加至 DB2 联合系统。通过将 Documentum 数据源添加至联合系统，就可以使用 SQL 语句来存取和查询 Documentum Docbase 中的对象和已注册的表。然后，可以将此数据与联合系统中的其它数据源集成到一起，而不需要将数据移出本机数据源。Documentum 包装器使用客户机库来与 Documentum 服务器进行交互。Documentum 包装器提供对两个版本的 Documentum 服务器的访问权：EDMS 98（也称为版本 3）和 4i。第 156 页的图 20 说明了 Documentum 包装器的工作方式。

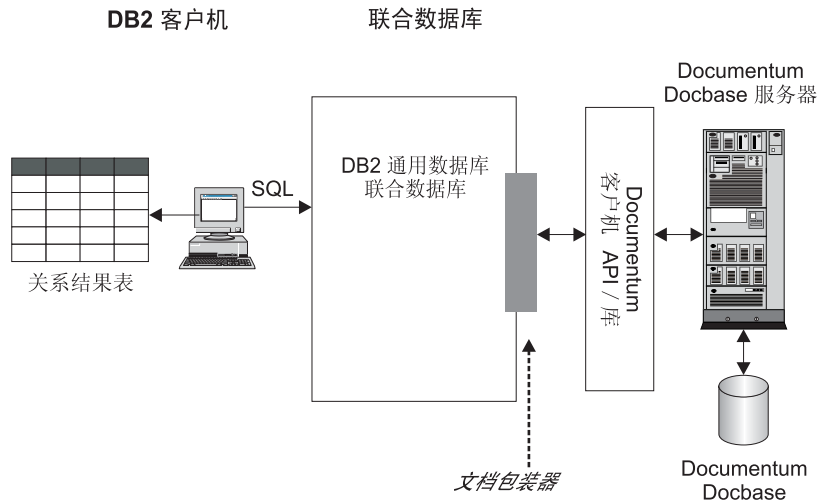


图 20. Documentum 包装器的工作方式

在注册 Documentum 包装器之后，就可以作为关系表来映射 Documentum Docbase 对象和已注册的表。这是通过将 Docbase 属性映射至 DB2 关系表中的列名来完成的。

例如，表 35 列示了 Documentum Docbase 缺省文档类型 dm\_document 的属性的子集以及相关的数据。您已经确定此属性子集对您很重要，并且您想将这些属性连接到联合数据库系统中。已将此数据子集命名为 DrugAB\_data。

表 35. DrugAB\_data

Title	Subject	Authors	Keywords
The effect of drug A on Drug A rabbits		Curran, L.	rabbits, drug A
Toxicity results for drug A	Drug A	Abelite, P., McMurtrey, K.	toxicity, drug A
Drug B interactions	Drug B	DeNiro, R., Stone, S.	interactions, drug B
Chemical structure of drug B	Drug B	Boyslim, F.	structure, drug B

注册 Documentum 包装器之后，可以使用 SQL 语句来查询数据。

以下查询将显示其主题 (subject) 为 Drug A 的标题 (title) 和作者 (author)。结果表显示在表 36 中。

```
SELECT title, authors
FROM drugAB_data
WHERE subject = 'Drug A'
```

表 36. 查询结果

Title	Authors
The effect of drug A on rabbits	Curran, L.
Toxicity results for drug A	Abelite, P., McMurtrey, K.

#### 相关任务:

- 第 157 页的『将 Documentum 数据源添加至联合服务器』

---

## 将 Documentum 添加至联合服务器

### 将 Documentum 数据源添加至联合服务器

要配置联合服务器以存取 Documentum 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Documentum 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

#### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

#### 过程:

要将 Documentum 数据源添加至联合服务器:

1. 使 Documentum 客户机库可供包装器使用。
2. 设置 Documentum 环境变量
3. 注册包装器。
4. 注册服务器定义。
5. 创建用户映射。
6. 为 Documentum Docbase 对象和已注册的表注册昵称。
7. 为 Documentum 包装器注册定制函数。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 158 页的『使 Documentum 客户机库可供包装器使用』
- 第 158 页的『设置 Documentum 环境变量』
- 第 160 页的『注册 Documentum 包装器』
- 第 161 页的『为 Documentum 数据源注册服务器』
- 第 161 页的『为 Documentum 数据源注册用户映射』
- 第 162 页的『为 Documentum 数据源注册昵称』
- 第 167 页的『为 Documentum 包装器注册定制函数』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 使 Documentum 客户机库可供包装器使用

使 Documentum 客户机库可供包装器使用是将 Documentum 添加至联合系统这一大型任务的一部分。

客户机库必须可供包装器使用才能使包装器正确工作。

### 先决条件:

Documentum 包装器使用版本 3.1.7a 的客户机库。如果您正在使用 Documentum 4i, 则将需要从 Documentum 中获取较旧版本的客户机库 (如果尚未安装它的话)。

### 过程:

要使 Documentum 客户机库可供包装器使用, 创建符号链接或者将客户机库复制到联合服务器上的适当目录中。下表列示了应该将库复制到的目录。

表 37. 按操作系统列出客户机库以及复制目标目录

联合服务器操作系统	客户机库	复制到的目录
AIX	libdmcl.a	sqllib/lib
Solaris	libdmcl.so	sqllib/lib
Windows	dmcl32.dll	x:\sqllib\bin

在这一系列任务中的下一个任务是设置 Documentum 包装器环境变量。

### 相关任务:

- 第 158 页的『设置 Documentum 环境变量』

## 设置 Documentum 环境变量

设置 Documentum 环境变量是将 Documentum 添加至联合服务器这一大型任务的一部分。

对 Documentum Docbases 的存取是通过 Documentum 客户机文件 dmcl.ini 来控制的。联合数据库实例必须将它的环境变量设置为 Documentum 客户机文件 dmcl.ini 才能获得对 Documentum Docbase 的存取权。

Documentum 的有效环境变量为:

- DOCUMENTUM
- DMCL\_CONFIG

### 先决条件:

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件 (例如, 数据源客户机软件)。

### 限制:

请参阅主题: db2dj.ini 文件的限制

### 过程:

要设置环境变量:

1. 编辑 db2dj.ini 文件, 并设置 DOCUMENTUM 或 DMCL\_CONFIG 环境变量。
  - 在运行 AIX 和 Solaris 的联合服务器上, db2dj.ini 文件位于 \$HOME/sqlllib/cfg 中
  - 在运行 Windows 的联合服务器上, db2dj.ini 文件位于 x:\sqlllib\cfg 中, 其中 **x:** 表示 sqlllib 目录所在的驱动器

下列示例说明这些变量在 UNIX 操作系统上的语法:

```
DOCUMENTUM=path
```

或者

```
DMCL_CONFIG=path/dmcl.ini
```

其中 *path* 是包含您想使用的 dmcl.ini 文件的全限定目录。有关 dmcl.ini 文件的路径的更多信息, 参阅随 Documentum 一起提供的文档。

如果在 db2dj.ini 文件中同时设置了这两个变量, 则使用 DMCL\_CONFIG 变量的值。如果在 db2dj.ini 文件中不设置这些变量中的任何一个变量, 则会返回错误。

2. 确保在 dmcl.ini 文件中指定了将 DB2 实例的所有可存取 Docbase 报告给的 docbroker 的名称, 如图 21 中所示。

---

```
##### DOCUMENTUM CLIENT CONFIGURATION FILE #####
#
# Copyright Documentum 1994.
# Version 3.1 of the Documentum Server.
#
# A generated client init file for the Documentum Server.
#
# The only REQUIRED information in this file is the
# [DOCBROKER_PRIMARY] section and an entry for host.
# The host value should be the name of host on which
# your network wide DocBroker is running

[DOCBROKER_PRIMARY]
host = server16.comp2.big.com
```

---

图 21. 指定了 docbroker 名称的样本 dmcl.ini 文件

3. 要确保在联合服务器上设置了环境变量, 重新启动 DB2 实例。发出下列命令来重新启动 DB2 实例:

```
db2stop
db2start
```

在这一系列任务中的下一个任务是注册 Documentum 包装器。

#### 相关任务:

- 第 160 页的『注册 Documentum 包装器』

#### 相关参考:

- 第 53 页的『对于 db2dj.ini 文件的限制』

## 注册 Documentum 包装器

注册 Documentum 包装器是将 Documentum 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Documentum 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

### 过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 dctm\_wrapper 的包装器，请发出以下语句：

```
CREATE WRAPPER dctm_wrapper LIBRARY 'libdb21sdctm.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 Documentum 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Documentum 数据源注册服务器定义。

### 相关参考:

- 第 160 页的『Documentum 包装器库文件』
- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）

## Documentum 包装器库文件

下表列示了 Documentum 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb21sdctm.a、libdb21sdctmF.a 和 libdb21sdctmU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 38. Documentum 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb21sdctm.a
Solaris	/opt/IBM/db2/V8.1/lib	libdb21sdctm.so
Windows	%DB2PATH%\bin	db21sdctm.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

### 相关任务:

- 第 160 页的『注册 Documentum 包装器』



## 为 Documentum 数据源注册服务器

为 Documentum 数据源注册服务器定义是将 Documentum 添加至联合系统这一大型任务的一部分。

### 限制:

在 DB2 的同一实例上运行的所有服务器必须共享 Documentum dmcl.ini 文件中的相同配置参数。

### 过程:

可以从 DB2 控制中心或 DB2 命令行注册服务器定义:

- 要从 DB2 控制中心执行此任务, 使用“联合对象”向导或者右键单击**服务器定义**文件夹并单击**创建**。“发现”工具将为 Documentum 服务器检索节点名。必须指定 RDBMS\_TYPE 和 OS\_TYPE 服务器选项的信息才能注册服务器定义。
- 要从 DB2 命令行执行此任务, 请使用 CREATE SERVER 语句。

例如, 要为包含在 AIX 上运行的 Docbase 并且使用 Oracle 来存储数据的 Documentum 服务器注册服务器定义 Dctm\_Server1, 请使用以下语句:

```
CREATE SERVER Dctm_Server1
  TYPE DCTM
  VERSION 3
  WRAPPER Dctm Wrapper
  OPTIONS( NODE 'Dctm_Docbase',
          OS_TYPE 'AIX',
          RDBMS_TYPE 'ORACLE');
```

在这一系列任务中的下一个任务是创建用户映射。

### 相关任务:

- 第 161 页的『为 Documentum 数据源注册用户映射』

### 相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 500 页的『CREATE SERVER 语句自变量和选项 - Documentum 包装器』

## 为 Documentum 数据源注册用户映射

为 Documentum 数据源创建用户映射是将 Documentum 添加至联合服务器这一大型任务的一部分。

当您尝试存取 Documentum 数据源时, 联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联 (用户映射)。为将访问联合系统以便发送分布式请求的每个用户标识创建用户映射。

### 过程:

要创建用户映射, 使用 CREATE USER MAPPING 语句。

例如，以下 CREATE USER MAPPING 语句将用户 Chuck 映射至 Dctm\_Server1 服务器上的用户 Charles。

```
CREATE USER MAPPING FOR Chuck SERVER Dctm_Server1
OPTIONS(REMOTE_AUTHID 'Charles', REMOTE_PASSWORD 'Charles_pw');
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。例如：

```
CREATE USER MAPPING FOR USER SERVER Dctm_Server1
    OPTIONS(REMOTE_AUTHID 'Lisa', REMOTE_PASSWORD 'Lisa_pw');
```

在这一系列任务中的下一个任务是为 Documentum 数据源注册昵称。

#### 相关任务:

- 第 162 页的『为 Documentum 数据源注册昵称』

#### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）
- 第 500 页的『CREATE USER MAPPING 语句选项 - Documentum 包装器』

## 为 Documentum 数据源注册昵称

### 为 Documentum 数据源注册昵称

为 Documentum 数据源注册昵称是将 Documentum 添加至联合服务器这一大型任务的一部分。

在创建用户映射之后，必须注册昵称。对于您定义的每个 Documentum 服务器，为想要存取的每个 Documentum Docbase 对象类型或已注册的表注册昵称。当您查询 Documentum 数据源时，使用这些昵称而不是 Docbase 的名称。

#### 过程:

要注册昵称，使用 CREATE NICKNAME 语句。

作为 CREATE NICKNAME 语句的一部分，可以定义伪列。如果未定义伪列，包装器就会创建伪列

在这一系列任务中的下一个任务是为 Documentum 数据源注册定制函数。

#### 相关任务:

- 第 167 页的『为 Documentum 包装器注册定制函数』
- 第 104 页的『设置 TurboBlast 以使用 BLAST 包装器』

#### 相关参考:

- 第 162 页的『伪列』
- 第 501 页的『CREATE NICKNAME 语句语法 - Documentum 包装器』
- 第 165 页的『CREATE NICKNAME 语句 - Documentum 包装器的示例』

### 伪列

与 Documentum 配合使用的 CREATE NICKNAME 语句定义了 6 个伪列。这些列用来存取对象内容和其它信息。

表 39 中列示了这些伪列及其定义。

表 39. 伪列名和定义。

伪列名	定义
GET_FILE	VARCHAR (nnn) <sup>1</sup>
GET_FILE_DEL	VARCHAR (nnn) <sup>1</sup>
GET_RENDITION	VARCHAR (nnn) <sup>1</sup>
GET_RENDITION_DEL	VARCHAR (nnn) <sup>1</sup>
HITS	INTEGER
SCORE	DOUBLE

注:

1. VARCHAR 的长度由联合服务器操作系统的最大路径长度确定。在 UNIX 联合服务器上, 此长度是 1024。在 Windows 联合服务器上, 此长度是 260。

表 40 列示了 SELECT 子句的伪列。

表 40. SELECT 子句的伪列

伪列名	描述
GET_FILE	<p>检索内容文件, 除了找到列值之外, 还要找到当前行。</p> <p>内容文件的扩展名是它的 Documentum 格式名。如果同名的文件存在, 则它将被覆盖。</p> <p>GET_FILE 试图获取对象的基本格式。它在一行中的值是文件的全限定文件名或者是字符串 “no_content”。</p> <p>例如:</p> <pre>SELECT object_name, get_file FROM ...</pre> <p>内容文件放置在由服务器的 CONTENT_DIR 选项指定的服务器目录中。它还放置在使用 DB2 本地名命名的子目录中。如果不存在该子目录, 则将创建它。</p> <p>它的扩展名将为在 Docbase 中为文档的格式类型定义的 DOS 扩展名。例如, MS Word 文档的扩展名为 “.doc”。</p>
GET_FILE_DEL	<p>此函数与 GET_FILE 相同, 只不过 GET_FILE_DEL 首先将删除在该查询中为先前检索的文件 (如果有的话)。它在一行中的值是文件的全限定文件名或者是字符串 “no_content”。</p>

表 40. SELECT 子句的伪列 (续)

伪列名	描述
GET_RENDITION	<p>检索该内容文件的复制（采用另一格式的原始文档的副本），除了找到列值之外，还要找到当前行。</p> <p>内容文件的扩展名是它的 Documentum 格式名。如果同名的文件存在，则它将被覆盖。</p> <p>要指定该复制格式，必须在 WHERE 子句中指定格式为 DCTM.RENDITION_FORMAT(&lt;format) = 1 的谓词。</p> <p>例如:</p> <pre>SELECT object_name, get_rendition FROM ... WHERE DCTM.RENDITION_FORMAT('pdf')=1</pre> <p>GET_RENDITION 试图获取对象的已命名复制。它在一行中的值是文件的全限定文件名或者是字符串 “no_content”。</p> <p>内容文件放置在由服务器的 CONTENT_DIR 选项指定的服务器目录中。它还放置在使用 DB2 本地名命名的子目录中。如果不存在该子目录，则将创建它。</p> <p>它的扩展名将为在 Docbase 中为文档的格式类型定义的 DOS 扩展名。例如，MS Word 文档的扩展名为 “.doc”。</p>
GET_RENDITION_DEL	<p>此函数与 GET_RENDITION 相同，只不过 GET_RENDITION_DEL 首先将删除在该查询中为先前检索的文件（如果有的话）。它在一行中的值是文件的全限定文件名或者是字符串 “no_content”。</p>

表 41 列示了包含搜索子句的查询中的 SELECT 子句的伪列。

表 41. 包含搜索子句的查询中的 SELECT 子句的伪列

伪列名	描述
HITS	<p>包含一个整数，它表示文档中与搜索条件相匹配的位置的数目。</p> <p>例如:</p> <pre>SELECT r_object_id, object_name, hits FROM std_doc WHERE DCTM.SEARCH_WORDS (''workflow'' OR ''flowchart'')=1</pre> <p>对于返回的每个文档，对文档内容中出现单词 “workflow” 和 “flowchart” 的次数进行求和，然后将结果作为 HITS 值返回。</p> <p>当文档只有一个内容文件时，HITS 伪列就是适当的。这是典型的情况。可以将此伪列用于 SELECT 语句的 WHERE 子句限制条件中。但是，还必须在 SELECT 子句中指定它。</p>

表 41. 包含搜索子句的查询中的 *SELECT* 子句的伪列 (续)

伪列名	描述
SCORE	包含文档的相关性等级。
	<p>将此伪列与 Documentum 的 ACCRUE 概念运算符一起使用。它们都会返回一个数，指示在返回的每个文档中找到了多少个指定的单词。</p> <p>例如:</p> <pre>SELECT object_name, score FROM std_doc WHERE DCTM.SEARCH_TOPIC('&lt;ACCRUE&gt;("document","management","workflow")')=1 AND SCORE &gt;=75</pre> <p>该语句将返回内容中有两个或三个指定单词的所有文档。如果一个文档中只有其中一个单词，则为该文档指定 50 分，因此，不能满足 WHERE 子句条件，将不会返回该文档。如果在一个文档中找到了三个单词中的两个单词，则为该文档指定 75 分。如果三个单词都找到了，则该文档的分数为 88。</p> <p>SCORE 伪列用于只有一个内容文件的文档。这是典型的情况。</p> <p>仅当 WHERE 中包含 SEARCH_WORDS() 或 SEARCH_TOPIC() 时，SELECT 子句中才能包含 SCORE。在 WHERE 子句中，将它与 ACCRUE 概念运算符一起使用。</p> <p>有关 ACCRUE 概念运算符的信息，请参阅 Documentum 文档。</p>

**相关任务:**

- 第 162 页的『为 Documentum 数据源注册昵称』
- 第 167 页的『为 Documentum 包装器注册定制函数』

**CREATE NICKNAME 语句 - Documentum 包装器的示例**

以下 CREATE NICKNAME 语句将定义昵称 std\_doc。Std\_doc 与对象类型为 dm\_document 的 Documentum Docbase 相关联。表 42 将 Documentum 属性和数据类型映射至要用来构造 CREATE NICKNAME 语句的 DB2 关系列名和数据类型。

表 42. 从 Documentum 属性至 std\_doc 昵称的 DB2 列的映射

Documentum 属性名	Documentum 数据类型	Documentum 数 DB2 列名	DB2 数据类型	可重复吗?	可空吗?
object_name	string(255)	object_name	varchar	否	否
r_object_id	ID	object_id	char(16)	否	否
r_object_type	string(32)	object_type	varchar	否	否
title	string(255)	title	varchar	否	否
subject	string(128)	subject	varchar	否	否
authors	string(32)	author	varchar	是	是
keywords	string(32)	keyword	varchar	是	是
r_creation_date	time	creation_date	timestamp	否	是
r_modify_date	time	modified_date	timestamp	否	是
a_status	string(16)	status	varchar	否	否

表 42. 从 *Documentum* 属性至 *std\_doc* 昵称的 DB2 列的映射 (续)

Documentum 属性名	Documentum 数据类型	Documentum 数 DB2 列名	DB2 数据类型	可重复吗?	可空吗?
a_content_type	string(32)	content_type	varchar	否	否
r_content_size	double	content_size	integer	否	否
owner_name	string(32)	owner_name	varchar	否	是

表 43 描述了在昵称中使用的每种 *Documentum* 属性。

表 43. 对昵称 *std\_doc* 的 *Documentum* 属性的描述

Documentum 属性名	描述
object_name	对象的用户定义的名称。
r_object_id	在创建时设置的此对象的唯一对象标识。
r_object_type	在创建对象时设置的对象的类型。
title	对象的用户定义的标题。
subject	对象的用户定义的主题。
authors	对象作者的用户定义的列表。
keywords	对象的用户定义的关键字列表。
r_creation_date	创建对象的日期和时间。
r_modify_date	最后一次修改对象的日期和时间。
a_status	转发路由器任务时由服务器设置。该值是从在路由器对象中为 attached_task_status 指定的那些值中获取的。
a_content_type	对象的内容的文件格式。
r_content_size	内容中的字节数。对于具有多页的文档，此属性将记录与该文档相关联的第一部分内容的大小。
owner_name	对象的所有者（即创建该对象的用户）的名称。

第 165 页的表 42 将转换成以下 CREATE NICKNAME 语句。

```
CREATE NICKNAME std_doc (
  object_name varchar(255) not null,
  object_id char(16) not null OPTIONS(REMOTE_NAME 'r_object_id'),
  object_type varchar(32) not null OPTIONS(REMOTE_NAME 'r_object_type'),
  title varchar(255) not null,
  subject varchar(128) not null,
  author varchar(32) OPTIONS(REMOTE_NAME 'authors', IS_REPEATING 'Y'),
  keyword varchar(32) OPTIONS(REMOTE_NAME 'keywords', IS_REPEATING 'Y'),
  creation_date timestamp OPTIONS(REMOTE_NAME 'r_creation_date'),
  modified_date timestamp OPTIONS(REMOTE_NAME 'r_modify_date'),
  status varchar(16) not null OPTIONS(REMOTE_NAME 'a_status'),
  content_type varchar(32) not null OPTIONS(REMOTE_NAME 'a_content_type'),
  content_size integer not null OPTIONS(REMOTE_NAME 'r_content_size'),
  owner_name varchar(32))
FOR SERVER Dctm_Server2 OPTIONS (REMOTE_OBJECT 'dm_document', IS_REG_TABLE 'N')
```

在提交 CREATE NICKNAME 语句之后，可以使用昵称 *std\_doc* 来查询联合系统。还可以将 *std\_doc* 昵称与联合系统中的其它昵称和表连接起来。

在目录中，由于存在伪列，因此，此昵称的列数为 6，超过了在 CREATE NICKNAME 语句中所指定的列数。

可以使用 CreateNicknameFile 实用程序来将 Documentum 类型自动映射至 DB2 类型，并创建初始 CREATE NICKNAME 语句。

**相关任务:**

- 第 162 页的『为 Documentum 数据源注册昵称』

**相关参考:**

- 第 501 页的『CREATE NICKNAME 语句语法 - Documentum 包装器』

## 双重定义重复属性 ( Documentum 包装器 )

为了使包装器的查询能力最大化，每种属性都必须定义为它的真实等价 DB2 数据类型。即必须将 Documentum 整数定义为 DB2 整数等等。但是，这些定义将阻止返回非 VARCHAR 重复属性的多个值。对于这些列，将只返回最后一个值。

存在此限制的原因是：只要有可能，包装器将只为每个 Docbase 对象返回一行结果。仅当选择了重复属性时此限制才是一个问题。但是，可以为同一远程重复属性定义第二列，但是该列具有数据类型 VARCHAR。

此列名将在 SELECT 列表中用来作为它的所有值的定界符分隔的列表来返回所有值。（每一列的 DELIMITER 选项指定要使用的定界符）。

应当使多值列的本地名称标准化。可以通过将前缀“m\_”添加至定义为它的真实数据类型的列的本地名称来使每个多值列的本地名称标准化。

例如，假定您具有称为 approval\_dates 的 Documentum 重复属性的昵称列并将它定义为数据类型 TIMESTAMP。您可以创建称为 m\_approval\_dates 的第二个昵称列并且可以将它定义为 VARCHAR 数据类型。然后，可以在 SELECT 列表中使用 m\_approval\_dates 来在用定界符分隔的列表中返回所有核准日期。

对于真实数据类型为 VARCHAR 的重复属性，不需要使用双重定义。

**相关任务:**

- 第 162 页的『为 Documentum 数据源注册昵称』
- 第 485 页的『改变昵称』

---

## Documentum 数据源的查询和定制函数

### 为 Documentum 包装器注册定制函数

为 Documentum 数据源注册定制函数是将 Documentum 添加至联合系统这一大型任务的一部分。必须使用 CREATE FUNCTION 语句来注册几个定制函数。可以使用这些函数来访问 Documentum 的某些唯一的功能，例如，全文本搜索和在查询中检索文档内容。

第 169 页的表 44 中列示了谓词的定制函数。

Verity, Inc 提供了一个 Documentum 函数作为第三方全文本索引系统的一部分，对 TOPIC 函数的引用就是对该函数的引用。

## 限制:

因为 DB2 不支持“布尔”类型，所以，WHERE 子句中使用的大多数定制函数（除了 USER 之外）必须对“=1”进行检查，这是因为这些函数被定义为要返回整数。

例如，

```
"... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1"
```

## 过程:

要注册定制函数，使用 CREATE FUNCTION 语句。

所有定制函数都必须使用模式名 DCTM 来注册。每个函数的标准名称为 DCTM.function\_name。

以下示例将注册 ANY\_EQ 定制函数。

```
CREATE FUNCTION DCTM.ANY_EQ (CHAR(), CHAR()) RETURNS INTEGER  
AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION
```

对于安装了 Documentum 包装器的每个联合数据库，必须同时注册每一个定制函数。

为了帮助您注册定制函数，在 sqllib/samples/lifesci/dctm 目录中提供了样本文件 create\_function\_mappings.ddl。此文件中包含每个定制函数的定义。可以运行此 DDL 文件来为安装了 Documentum 包装器的每个联合数据库注册定制函数。

## 定制函数字符串自变量规则

作为字符串传递的所有自变量都必须遵守下列规则:

- 将每个字符串都用单引号引起来。
- 字符串中的单引号是用两个单引号表示的。

## 在查询中使用定制函数

下列示例说明了如何在查询中使用定制函数。

要对具有一个或多个名为“Dave Winters”的作者的文档从 std\_doc 昵称中显示对象名和作者:

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1
```

要对具有一个或多个名为“Dave Winters”或“Jon Doe”的作者的文档从 std\_doc 昵称中显示对象名和作者:

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_IN(authors,'Dave Winters','Jon Doe')=1
```

要从作者列中包含类似于 'Dave Win%' 的字符串的文档的 std\_doc 昵称中显示对象名和 r\_object\_id 以及检索内容文件:

```
SELECT object_name, r_object_id, get_file FROM std_doc  
WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1
```



## 定制函数表

表 44 列示了谓词的定制函数。

表 44. 谓词的定制函数

函数名	描述
ANY_EQ(arg1, arg2)	<p>测试与指定值相等的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定要比较的值。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1</pre>
ANY_NE(arg1, arg2)	<p>测试不等于指定值的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定要比较的值。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_NE(authors,'Dave Winters')=1</pre>
ANY_LT(arg1, arg2)	<p>测试小于指定值的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定要比较的值。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_LT(num_approvers,4)=1</pre>
ANY_GT(arg1, arg2)	<p>测试大于指定值的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定要比较的值。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_GT(num_approvers,3)=1</pre>
ANY_LE(arg1, arg2)	<p>测试小于或等于指定值的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定要比较的值。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_LE(num_approvers,2)=1</pre>

表 44. 谓词的定制函数 (续)

函数名	描述
ANY_GE(arg1, arg2)	<p>测试大于或等于指定值的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定要比较的值。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_GE(num_approvers,1)=1</pre>
ANY_IN(arg1, arg2 - arg11)	<p>测试在指定的多个值的列表中任何十个值的重复属性。采用了 3 到 11 个同一数据类型的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2-arg11</b> 指定要比较的用逗号隔开的多个值的列表。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_IN(authors,'Crick','Watson')=1</pre> <p>对于单个语句, ANY_IN 定制函数中用于重复属性的值的最大个数为 10。可以将多个语句用 OR 连接。</p>
ANY_LIKE(arg1, arg2)	<p>测试与指定值相似的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定用来与子串进行比较的模式, 并用单引号引起来。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1 OR DCTM.ANY_LIKE(keywords,'%\_%')=1</pre> <p>在 ANY_LIKE() 谓词中不支持转义子句。</p>
ANY_NOT_LIKE(arg1, arg2)	<p>测试与指定值不相似的任何值的重复属性。采用了两个必需的自变量:</p> <p><b>arg1</b> 指定表示重复属性的一列的名称。</p> <p><b>arg2</b> 指定用来与子串进行比较的模式, 并用单引号引起来。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_NOT_LIKE(authors,'Dave Win%')=1 OR DCTM.ANY_NOT_LIKE(keywords,'%\_%')=1</pre> <p>在 ANY_NOT_LIKE() 谓词中不支持转义子句。</p>
ANY_NULL(arg)	<p>测试 IS NULL 的重复属性。采用一个必需的自变量, 也就是重复属性或单值 DATE 或 TIMESTAMP 属性的名称。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_NULL(authors)=1</pre>

表 44. 谓词的定制函数 (续)

函数名	描述
ANY_NOT_NULL(arg)	<p>测试 IS NOT NULL 的重复属性。采用一个必需的自变量，也就是重复属性的名称。</p> <p>例如:</p> <pre>... WHERE DCTM.ANY_NOT_NULL(authors)=1</pre>
ANY_SAME_INDEX(arg1 - arg10)	<p>测试在每种属性的同一索引中的多个值的重复属性。采用了二到十个其它 ANY_xx() 函数。</p> <p>以下示例将检查文档中是否至少有一个名为 Ken 的作者未加入 UCD。</p> <pre>... WHERE DCTM.ANY_SAME_INDEX( ANY_EQ(author_name,'Ken'), DCTM.ANY_NE(author_affiliation,'UCD'))=1</pre> <p>对处于重复属性的同一索引中的多个值的最大测试次数为 10。这些测试必须是按照从左到右的顺序进行求值的 AND 测试。</p>
CABINET(arg) 和 CABINET_TREE(arg)	<p>采用一个必需的自变量，也就是 Docbase 文件柜的标准名称。</p> <p>例如:</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 ... WHERE DCTM.CABINET_TREE('/MyDocs')=1</pre> <p>使用 CABINET 和 CABINET_TREE 的多个实例来指定多个文件柜。</p> <p>例如:</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 OR DCTM.CABINET_TREE('/Parts')=1</pre>
FOLDER(arg) 和 FOLDER_TREE(arg)	<p>采用一个必需的自变量，也就是 Docbase 文件夹或文件柜的标准名称。</p> <p>例如:</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 ... DCTM.FOLDER_TREE('/MyDocs/WhitePapers')=1</pre> <p>使用 FOLDER 和 FOLDER_TREE 的多个实例来指定多个文件夹。</p> <p>例如:</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 OR DCTM.FOLDER_TREE('/Animals/Horses')=1</pre>
RENDITION_FORMAT(format)	<p>与 GET_RENDITION 和 GET_RENDITION_DEL 伪列一起工作来建立要检索的重现版的格式。采用用来指定格式的单个字符串自变量。</p> <p>以下示例将检索采用 PDF 格式的文档:</p> <pre>SELECT get_rendition FROM .... WHERE DCTM.RENDITION_FORMAT('pdf')=1</pre>

表 44. 谓词的定制函数 (续)

函数名	描述
USER(1)	<p>将值与当前用户的 Documentum 作者标识进行比较。由于 DB2 的限制，定制函数 USER 是用未使用的整数自变量定义的。</p> <p>例如:</p> <pre>... WHERE approver = DCTM.USER(1)</pre> <p>要使 Documentum 作者标识与 DB2 作者标识相对应，使用 CREATE USER MAPPING 语句。</p>
SEARCH_WORDS(arg)	<p>采用一个必需的字符串自变量，也就是用单引号引起来、用 AND、OR 或 NOT 隔开并且使用圆括号来控制优先顺序的个别单词的列表。单词之间不能包含空格并且必须用单引号引起来。</p> <p>例如:</p> <pre>... DCTM.SEARCH_WORDS(''yeast'' AND (''bread'' OR ''cake'') AND NOT ''wedding'')=1</pre>
SEARCH_TOPIC(arg)	<p>采用一个必需的字符串自变量，它是一个要完全照原样传递给 Documentum 和 Verity 的 Verity TOPIC 查询语句。</p> <p>例如:</p> <pre>... WHERE DCTM.SEARCH_TOPIC('"quick")=1</pre>

在这一系列任务中没有别的任务了。

#### 相关参考:

- 『CREATE FUNCTION (Sourced or Template) statement』 (*SQL Reference, Volume 2*)

## Documentum 数据源 - 示例查询

注册包装器之后，就可以对 Documentum 数据源运行 SQL 查询。本节提供了几个示例查询。

要运行查询，按照您使用常规表名和表列的方式来在 SQL 语句中使用昵称和已定义的昵称列。

#### 连接局限性:

对于由 DB2 应用程序建立的与 DB2 数据库的每个连接，Documentum 包装器最多可以支持 10 个同时进行的 Documentum 会话，并且每个这样的会话最多可以同时管理 10 个 Documentum 查询。

单个 DB2 应用程序可以同时进行几个查询，当将查询提交给 DB2 时，查询的生存期就开始了，当结果集上相应的游标关闭时，查询就结束了。

在给定的任何时间，在此时正在进行的整个查询集中，引用一个 Documentum 服务器中的昵称数不能超过 10 个。在多个查询中提到的昵称或者在单个查询中多次引用的昵称必须是出现一次就计数一次。

## LIKE 谓词:

Documentum 服务器与 DB2 处理 LIKE 谓词的方式不同。当将 LIKE 谓词下推到 Documentum 服务器中时, Documentum 语义就适用。在以下示例中, 当 c1 列中包含长度为零的字符串时, 对于 Documentum, 谓词将为 true, 对于 DB2, 谓词将为 false。

```
c1 LIKE '%'
```

## 示例查询:

以下查询将显示名为 “Test Document” 的文档的所有 Docbase 文档:

```
SELECT object_name
FROM std_doc
WHERE object_name='Test Document';
```

以下查询使用定制函数 ANY\_EQ 来显示其中一个作者是 “Joe Doe” 的所有文档。

```
SELECT object_name
FROM std_doc
WHERE DCTM.ANY_EQ(author, 'Joe Doe')=1
```

以下查询使用 FOLDER\_TREE 函数和 SEARCH\_WORDS 函数来查找 Approved 文件柜中的所有包含文本 “protein” 文档。

```
SELECT object_name
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.SEARCH_WORDS('protein')=1
```

以下查询使用 GET\_FILE 伪列以及 FOLDER\_TREE 和 ANY\_IN 定制函数来检索 DB2 服务器上内容是 Approved 文件柜中具有任何所列作者的全部文档内容的文件名。

```
SELECT object_name, object_id, get_file
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.ANY_IN(author, 'Mary Black', 'Joe Carson', 'Peter Miller')=1
```

## 相关任务:

- 第 167 页的『为 Documentum 包装器注册定制函数』

## 相关参考:

- 第 167 页的『双重定义重复属性 (Documentum 包装器)』
- 第 173 页的『Documentum 包装器的访问控制』

## Documentum 包装器的访问控制

查询必须得到您在 Docbase 中的许可。只有您对它至少具有读存取权的那些文档才会包括在查询结果中。

## 相关参考:

- 第 335 页的『表结构文件包装器的文件访问控制模型』
- 第 207 页的『Excel 包装器的文件访问控制模型』

---

## Documentum 包装器的消息

本节列示和描述了您在使用 Documentum 的包装器时可能会遇到的消息。

表 45. 由 Documentum 的包装器发出的消息

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “sqlno_crule_save_plans [100]:rc (-2144272209) 检测到空的计划列表”。)	包装器不能处理提交给 DB2 的 SQL 查询。更正语法, 然后重新提交。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “dmAPI exec 失败: [DM_QUERY_E_BAD_QUAL] 错误: “属性 <column_name> 的属性限定符 A0 是无效限定符。””。)	为 REMOTE_OBJECT 昵称选项输入了不正确的 Documentum 类型或已注册的表。更改昵称以使用正确的 Documentum 对象类型或已注册的表。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “指定了无效的空列”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “昵称规范是空的”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: ““输出”对象是空的或者不完整”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “请求了意外的列数”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “找不到列信息”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “请求了不受支持的列类型”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “列定义不正确”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “类型不一致; DB2 请求 != 昵称类型”。)	内部编程错误。请与 IBM 软件支持机构联系。

表 45. 由 Documentum 的包装器发出的消息 (续)

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “输出参数不为 NULL”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “查询输出变量不为 NULL”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “时间戳记长度无效”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “列数不一致”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “转换值时未能存取数据”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “未能初始化 DMCL 客户机”。)	Documentum 客户机不能初始化。请与系统管理员联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “Get_User 返回了 NULL”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “Get_Local_User 返回了 NULL”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “开始事务失败”。)	Documentum 报告了开始事务失败。请与系统管理员联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “输入参数不为 NULL”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “Dctm 函数必须类似于 DCTM.function(...) =1”。)	未使用 =1 作为 Dctm 函数的谓词的 RHS。更正语法, 然后再次运行查询。

表 45. 由 Documentum 的包装器发出的消息 (续)

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“请求的列号无效”。)	内部编程错误。请与 IBM 软件支持机构联系。
SQL1881N	“DELIMITER”不是“<column-name>”的有效“COLUMN”选项	为 <column-name> 这一列指定了 DELIMITER 选项，但是未指定 IS_REPEATING 选项。
SQL1882N	“SERVER”选项“RDBMS_TYPE”不能设置为“<server-name>”的“<option-value>”。	为 RDBMS_TYPE 服务器选项指定的值无效。它必须是下列值之一： DB2、INFORMIX、ORACLE、SQLSERVER 或 SYBASE。
SQL1882N	“SERVER”选项“TRANSACTIONS”不能设置为“<server-name>”的“<option-value>”。	为 TRANSACTIONS 服务器选项指定的值无效。它必须是下列值之一： NONE、QUERY、PASSTHRU 或 ALL。
SQL1882N	“NICKNAME”选项“IS_REG_TABLE”不能设置为“<nickname>”的“<option-value>”。	为 IS_REG_TABLE 昵称选项指定的值无效。它必须是下列值之一：'Y' 或 'N'。
SQL1882N	“NICKNAME”选项“ALL_VERSIONS”不能设置为“<nickname>”的“<option-value>”。	为 ALL_VERSIONS 昵称选项指定的值无效。它必须是下列值之一：'Y' 或 'N'。
SQL1882N	“SERVER”选项“OS_TYPE”不能设置为“<server-name>”的“<option-value>”。	为 OS_TYPE 服务器选项指定的值无效。它必须是：AIX、HPUX、SOLARIS 或 WINDOWS。
SQL1882N	“NICKNAME”选项“FOLDERS”不能设置为“<nickname>”的“<option-value>”。	为 FOLDERS 昵称选项指定的值无效。不能对将 IS_REG_TABLE 设置为 'Y' 的表指定该值。
SQL1882N	“NICKNAME”选项“VERSIONS”不能设置为“<nickname>”的“<option-value>”。	为 VERSIONS 昵称选项指定的值无效。它必须是下列值之一：'Y' 或 'N'。对于 IS_REG_TABLE 选项设置为 'Y' 的表，不能将 VERSIONS 选项设置为 'Y'。
SQL30090N	对于应用程序执行环境，操作无效。原因码 = “在昵称中指定了无效列名、IS_REG_TABLE 或 IS_REPEATING”	检查昵称语句，以了解 IS_REG_TABLE、IS_REPEATING、REMOTE_NAME 选项和列名的正确规范。
SQL30090N	对于应用程序执行环境，操作无效。原因码 = “未能打开日志文件以便进行调试”	不能存取用于故障诊断的日志文件。请与系统管理员联系。
SQL30090N	对于应用程序执行环境，操作无效。原因码 = “只能指定一个搜索条件”	每个查询只能指定一个定制搜索函数。



表 45. 由 Documentum 的包装器发出的消息 (续)

错误代码	消息	解释
SQL30090N	对于应用程序执行环境，操作无效。原因码 = “未能创建内容目录”	确保 DB2 代理进程可以写入目的地目录。
SQL30090N	对于应用程序执行环境，操作无效。原因码 = “未能更改对内容文件的许可权”	确保 DB2 代理进程可以写入目标内容目录。
SQL5182N	尚未设置必需的环境变量“DMCL_CONFIG”。	DOCUMENTUM 和 DMCL_CONFIG 环境变量都未设置。请在 db2dj.ini 文件中设置它们。

**相关概念:**

- 『消息介绍』（《消息参考》第 1 卷）

**相关参考:**

- 『SQLSTATE 消息』（《消息参考》第 2 卷）



## 第 11 章 配置对 Entrez 数据源的存取

本章说明如何配置联合服务器以存取存储在 Entrez 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Entrez 数据源的存取。

本章包含下列内容:

- 说明什么是 Entrez
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 Entrez 包装器相关联的错误消息

### 什么是 Entrez?

Entrez 是由“国家生物技术信息中心”（NCBI）开发的一个查询检索系统。可以使用 Entrez 来存取由 NCBI 主管的几个已链接的数据库。

这些数据库包括:

- PubMed（生物医学文献）
- Nucleotide（也称为 GenBank 的序列数据库）
- OMIM（由 John Hopkins 大学开发的“孟德尔人类遗传学联机数据库”）
- Genome（完整基因组汇编）

可以通过统一的一组基于 Web 的工具来存取所有 Entrez 数据库。Entrez 包装器使用这些工具来将 Entrez 数据库联合到 DB2® 环境中。尽管 Entrez 接口支持许多数据库，但是 Entrez 包装器只支持 PubMed 和 Nucleotide。

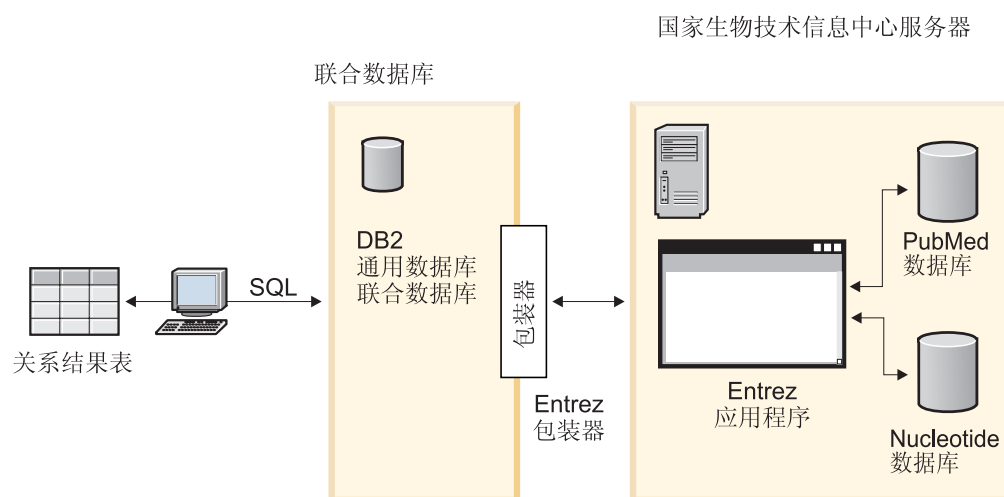


图 22. Entrez 包装器的工作方式

Entrez 包装器的许多元素是所有数据库共有的。这些元素包括:

- 通过 Web 和 Entrez ESearch 和 EFetch 实用程序与 NCBI 连接

- 将分层 XML 数据映射到关系表中
- 通过 XML 包装器技术来使相关表之间进行连接

**相关任务:**

- 第 180 页的『将 Entrez 数据源添加至联合服务器』

---

## 将 Entrez 添加至联合服务器

### 将 Entrez 数据源添加至联合服务器

要配置联合服务器以存取 Entrez 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Entrez 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

Entrez 包装器可以存取使用具有代理的防火墙的网络中的 PubMed 和 Nucleotide 数据源。受支持的代理是：HTTP、SOCKS4 和 SOCKS5。

**先决条件:**

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

**过程:**

要将 Entrez 数据源添加至联合服务器:

1. 注册 Entrez 包装器的定制函数。
2. 注册包装器。
3. 注册服务器定义。
4. 为 Entrez 数据库注册昵称。

**相关概念:**

- 『DB2 Information Integrator 安装过程 - 概述』(《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

**相关任务:**

- 第 44 页的『创建联合数据库』
- 第 181 页的『为 Entrez 包装器注册定制函数』

**相关参考:**

- 『DB2 Information Integrator 支持的操作系统 (32 位)』(《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』(《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 为 Entrez 包装器注册定制函数

为 Entrez 包装器注册定制函数是将 Entrez 添加至联合系统这一大型任务的一部分。注册定制函数之后，必须注册包装器。

### 限制:

- 必须使用模式名 `entrez` 来注册 Entrez 包装器的所有定制函数。
- 必须为安装了 Entrez 包装器的每个 DB2 数据库将每个定制函数都注册一次。

### 过程:

要注册定制函数，请发出带有 `AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION` 关键字的 `CREATE FUNCTION` 语句。

每个函数的标准名为 `entrez.function_name`。

以下示例将注册一个版本的 `CONTAINS` 函数:

```
CREATE FUNCTION entrez.contains (varchar(), varchar())
  RETURNS INTEGER AS TEMPLATE
  DETERMINISTIC NO EXTERNAL ACTION;
```

要注册定制函数，使用样本文件 `create_function_mappings.ddl`。样本文件位于 DB2 Information Integrator 的安装路径中（在 `samples/lifesci/entrez` 目录下面）。样本文件包含每个定制函数的定义。可以运行此 DDL 文件来为安装了 Entrez 包装器的每个联合数据库注册定制函数。

在这一系列任务中的下一个任务是注册 Entrez 包装器。

### 相关参考:

- 『CREATE FUNCTION (Sourced or Template) statement』 (*SQL Reference, Volume 2*)
- 第 186 页的『定制函数和 Entrez 查询』
- 第 181 页的『定制函数表 - Entrez 包装器』

## 定制函数表 - Entrez 包装器

使用 `CREATE FUNCTION` 语句来注册 Entrez 定制函数。

下表列示了 Entrez 定制函数和在注册函数时指定的自变量的数据类型。在函数中指定的第一个自变量是为带标记列的列名指定的。在函数中指定的第二个自变量是搜索项。

表 46. Entrez 包装器的定制函数

函数	描述
entrez.contains (varchar(), varchar())	使用您指定的项来搜索带标记列。
entrez.contains (integer, varchar())	
entrez.contains (smallint, varchar())	
entrez.contains (real, varchar())	
entrez.contains (double, varchar())	
entrez.contains (date, varchar())	
entrez.contains (time, varchar())	
entrez.contains (char(), varchar())	
entrez.contains (timestamp(), varchar())	
entrez.search_term (char(), varchar())	直接将 Entrez 搜索项传递给 Entrez 搜索引擎。

要注册定制函数，使用样本文件 `create_function_mappings.ddl`。该样本文件安装在 `samples/lifesci/entrez` 目录中。

#### 相关任务:

- 第 181 页的『为 Entrez 包装器注册定制函数』

#### 相关参考:

- 第 186 页的『定制函数和 Entrez 查询』

## 注册 Entrez 包装器

注册 Entrez 包装器是将 Entrez 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Entrez 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 `CREATE WRAPPER` 语句。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 `entrez_wrapper` 的包装器，请发出以下语句:

```
CREATE WRAPPER entrez_wrapper LIBRARY 'libdb2lsentrez.a'
  OPTIONS(EMAIL 'jeff@someplace.com');
```

注册 Entrez 包装器时必须指定电子邮件地址。所有查询中都包括此电子邮件地址，并且在有问题时（例如，具有太多查询使 NCBI 服务器重载时），允许 NCBI 与您联系。

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 `CREATE WRAPPER` 语句中应指定的正确名称，请参阅 Entrez 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Entrez 数据源注册服务器定义。

#### 相关参考:

- 第 183 页的『Entrez 包装器库文件』

- 『CREATE WRAPPER statement』 (SQL Reference, Volume 2)

## Entrez 包装器库文件

下表列示了 Entrez 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2lsentrez.a、libdb2lsentrezF.a 和 libdb2lsentrezU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 47. Entrez 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2lsentrez.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsentrez.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsentrez.so
Windows	%DB2PATH%\bin	db2lsentrez.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

相关任务:

- 第 182 页的『注册 Entrez 包装器』

## 为 Entrez 数据源注册服务器

为 Entrez 数据源注册服务器定义是将 Entrez 添加至联合服务器这一大型任务的一部分。

在注册包装器之后，必须注册相应的服务器定义

由特定数据源表示的数据库 PubMed 或 Nucleotide 是由 CREATE SERVER 语句中表示的服务器类型值标识的。此服务器类型值将控制所创建的任何昵称的结构。

过程:

要向联合系统注册 Entrez 服务器，发出 CREATE SERVER 语句。

例如，要为 entrez\_wrapper 包装器注册一个名为 pubmed\_server1 的服务器，发出以下语句:

```
CREATE SERVER pubmed_server1
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper;
```

另外，要为 entrez\_wrapper 包装器注册一个名为 nucleotid\_server1 的服务器，发出以下语句:

```
CREATE SERVER nucleotid_server1
  TYPE NUCLEOTIDE
  VERSION 1.0
  WRAPPER entrez_wrapper;
```

## 限制对 Entrez 查询检索的行数

可以使用 MAX\_ROWS 服务器选项来限制对使用 Entrez 包装器的查询返回的行数。

与 SQL 语句的 FETCH FIRST N ROWS ONLY 子句（该子句限制返回给用户或应用程序的行数）不同，MAX\_ROWS 服务器选项使您能够限制可从 NCBI Web 站点检索的行数。

MAX\_ROWS 选项值总是用作查询可以检索的行数的上限（最大值）。如果查询尝试检索比在 MAX\_ROWS 选项中指定的数目更多的行，结果集将会截断，并且发出警告消息。

可以在创建服务器时设置 MAX\_ROWS 服务器选项，也可以使用 ALTER SERVER 语句来更改选项值。

MAX\_ROWS 服务器选项不是必需的。如果不设置该选项，则使用缺省值。使用的特定缺省值取决于您的操作系统。对于 Microsoft Windows 操作系统，缺省值是 2000 行。对于基于 UNIX 的操作系统，缺省值是 5000 行。

只可以指定正数和 0（零）。当将该选项设置为 0（零）时，就可使查询能够从 NCBI Web 站点检索无限数目的行。但是，将 MAX\_ROWS 服务器选项设置为 0（零）或设置为一个很大的数目可能会影响查询性能。

## 通过代理服务器存取 Entrez

要通过代理服务器存取 Entrez 数据源，必须在创建服务器定义时指定一些选项。您指定的选项取决于想要访问的代理服务器类型。

### 为 HTTP 代理服务器注册服务器定义的示例:

要注册服务器定义和指定 HTTP 代理服务器，使用以下语句:

```
CREATE SERVER pubmed_server_h
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_h',
          PROXY_SERVER_PORT '8080');
```

### 为 SOCKS4 代理服务器注册服务器定义的示例:

要注册服务器定义和指定 SOCKS4 代理服务器，使用以下语句:

```
CREATE SERVER pubmed_server_s4
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS4', PROXY_SERVER_NAME 'proxy_4',
          PROXY_SERVER_PORT '1080');
```

### 为 SOCKS5 代理服务器注册服务器定义而不提供认证信息的示例:

要注册服务器定义和指定 SOCKS5 代理服务器而不提供认证信息，使用以下语句:

```
CREATE SERVER pubmed_server_s5
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081');
```



为 **SOCKS5** 代理服务器注册服务器定义并提供认证信息的示例:

要注册服务器定义和指定 **SOCKS5** 代理服务器并提供认证信息, 使用以下语句:

```
CREATE SERVER pubmed_server_s5a
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081', PROXY_AUTHID 'Khalid',
          PROXY_PASSWORD 'aaa', );
```

在这一系列任务中的下一个任务是为 Entrez 数据源注册昵称。

**相关任务:**

- 第 185 页的『为 Entrez 数据源注册昵称』

**相关参考:**

- 第 510 页的『CREATE SERVER 语句自变量 - Entrez 包装器』

## 为 Entrez 数据源注册昵称

为 Entrez 数据源注册昵称是将 Entrez 添加至联合系统这一大型任务的一部分。

**限制:**

每个 Entrez 数据库的模式是由包装器决定的, 不能够更改或修改。每个数据库都有固定的一组表, 而每个表都有各列的固定列表。数据库中的各个表之间具有分层关系。在数据库中, 是所有其它表的父代的一个表被称为根表(父表)。数据库中的所有其它表与根表之间都具有父子关系。

**过程:**

要为 Entrez 数据源注册昵称, 发出 CREATE NICKNAME 语句。

因为昵称的各列的列表是由包装器决定和提供的, 所以, 创建 Nucleotide 昵称的基本语法很简单。例如:

```
CREATE NICKNAME GBSeq FOR SERVER nucl;
CREATE NICKNAME GBFeatures FOR SERVER nucl;
CREATE NICKNAME GBIntervals FOR SERVER nucl;
CREATE NICKNAME GBQualifiers FOR SERVER nucl;
CREATE NICKNAME GBReference FOR SERVER nucl;
```

以下是创建 PubMed 昵称的基本语法的示例:

```
CREATE NICKNAME pparticles FOR SERVER pubmed_server;
CREATE NICKNAME PMACCESSION FOR SERVER pubmed_server;
CREATE NICKNAME PMCHEMICAL FOR SERVER pubmed_server;
CREATE NICKNAME PMMESH FOR SERVER pubmed_server;
CREATE NICKNAME PMCOMMENTS FOR SERVER pubmed_server;
CREATE NICKNAME PMARTICLEID FOR SERVER pubmed_server;
```

昵称的名称是基础表的名称。

使用此语法将限制您每种 DB2 模式只能使用一个系列的昵称。可以通过使用昵称选项 REMOTE\_OBJECT 和 PARENT 来使用其它名称。对于根昵称, 只有 REMOTE\_OBJECT 是必需的。对于其它任何昵称, 必须同时提供 REMOTE\_OBJECT 和 PARENT。

以下示例显示了使用重命名功能的同一组 Nucleotide 昵称:

```
CREATE NICKNAME NewSeq FOR SERVER nuc1 OPTIONS (REMOTE_OBJECT 'GBSEQ');
CREATE NICKNAME NewFeatures FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBFEATURES', PARENT 'NEWSEQ');
CREATE NICKNAME NewIntervals FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBINTERVALS', PARENT 'NEWFEATURES');
CREATE NICKNAME NewQualifiers FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBQUALIFIERS', PARENT 'NEWFEATURES');
CREATE NICKNAME NewReference FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBREFERENCE', PARENT 'NEWSEQ');
```

以下示例显示了使用重命名功能的同一组 PubMed 昵称:

```
CREATE NICKNAME newparticles FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMARTICLES');
CREATE NICKNAME NEWPMACCESSION FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMACCESSION', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMCHEMICAL FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMCHEMICAL', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMESH FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMMESH', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMCOMMENTS FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMCOMMENTS', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMARTICLEID FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMARTICLEID', PARENT 'NEWPMARTICLES');
```

在这一系列任务中的下一个任务是为 Entrez 数据源注册定制函数。

相关任务:

- 第 180 页的『将 Entrez 数据源添加至联合服务器』
- 第 181 页的『为 Entrez 包装器注册定制函数』
- 第 58 页的『为非关系数据源指定昵称列』

相关参考:

- 『CREATE NICKNAME statement』 (*SQL Reference, Volume 2*)
- 第 190 页的『PubMed 模式表』
- 第 194 页的『Nucleotide 模式表』

---

## Entrez 数据源的查询和定制函数

### 定制函数和 Entrez 查询

联合环境使用两个查询引擎。对于 Entrez 包装器, 这些查询引擎是 DB2 和 Entrez。有一种例外的情况是, 您通过定制函数来指定 Entrez 引擎的所有谓词。对于 DB2 引擎, 通过关系运算符来指定所有谓词。

主定制函数为 ENTREZ.CONTAINS。CONTAINS 函数需要具有搜索项列自变量和查询文本自变量。以下示例显示了一个 ENTREZ.CONTAINS 语句:

```
ENTREZ.CONTAINS (search_term_column, query_text)
```

模式表的 Q 列中的标记标识一个搜索项。查询文本必须位于已修改的 Entrez 查询语法中。此语法由使用布尔运算符 (OR、AND 和 NOT) 分隔并使用圆括号来分组的搜索条件组成。CONTAINS 查询文本自变量的语法与该搜索项限定符中的标准 Entrez 查询语法有所不同, 例如, 不允许使用 [pd]。

定制函数是在 Entrez 模式中注册的，它必须用来引用函数。当使用定制函数时，必须在等号谓词中将它们的返回值与 1 进行比较。

在某些情况下，可能会以某种方式来混合使用 DB2 和 Entrez 谓词，在该方式下，将不能处理 DB2 和 Entrez 谓词。在这些情况下，将产生错误消息 SQL0142N（“SQL 语句不受支持”）。

例如，在以下查询中，不能将由包装器处理的谓词部分（ENTREZ.CONTAINS 调用）与必须由 DB2 处理的部分（BaseCountA 上的关系谓词）分隔开。

```
WHERE
  ENTREZ.CONTAINS (Organism, 'drosophila') = 1
  OR (BaseCountA > 10 AND ENTREZ.CONTAINS (Keywords, 'glop') = 1)
```

某些搜索字段在 Entrez 模式没有相应的列。例如，在 nucleotide 数据库中，[ALL] 这一条件将搜索所有可搜索的字段，而 [WORD] 将搜索与记录相关联的所有自由文本。为这些搜索条件提供了伪列。如果在选择列表中引用了伪列，则返回的值为 NULL。

可以通过发出 ENTREZ.SEARCH\_TERM 主函数来运行没有此主函数可能就不能运行的查询。如果指定了 ENTREZ.SEARCH\_TERM 主函数，则它必须是查询中唯一的定制函数。对于每个查询，每个 Entrez 昵称只能有一个 ENTREZ.SEARCH\_TERM 主函数。另外，在同一个查询的同一昵称中不能混合 SEARCH\_TERM 和 CONTAINS 函数。第一个自变量（即列规范）必须是父昵称的主键列。第二个自变量（即查询文本）是包括搜索字段限定符的 Entrez 格式的搜索条件。此文本是未经修改就进行传递的，只不过在需要时 URI 将被 URI 语法转义为 Entrez。

以下示例显示了在 PubMed 昵称上具有 WHERE 子句的一个查询：

```
WHERE
  ENTREZ.CONTAINS (authors, 'kaufmann OR ito AND NOT rakesh')
  AND
  (ENTREZ.CONTAINS (title, 'drosophila')
   OR
  ENTREZ.CONTAINS(alltext, 'drosophila OR "fruit fly"'))
```

在此示例中，各个谓词为 authors、title 和 all text。

修改了各个谓词，以便将限定符添加到每个搜索项后面。然后，使用圆括号将搜索项括起来，以便强制执行 DB2 布尔运算符优先权。由于进行了这些修改，authors 谓词将变成：

```
((kaufmann[auth] OR ito[auth]) AND (NOT (rakesh[auth])))
```

title 谓词将变成：

```
(drosophila[titl])
```

而 all text 谓词将变成：

```
(drosophila[all] OR "fruit fly"[all])
```

当组合各个谓词时，使用圆括号来维护 DB2 布尔运算符的优先权。除了将字符串表示为 URI 的一部分需要进行的文本变换之外，提交给 Entrez 的最终搜索项字符串为：

```
((kaufman[auth] OR ito[auth]) AND (NOT (rakesh[auth]))) AND
((drosophila[titl]) OR (drosophila[all] OR "fruit fly"[all]))
```

**相关参考：**

- 第 181 页的『定制函数表 - Entrez 包装器』

## Entrez 包装器的关系谓词

Entrez 包装器支持昵称列上包含关系谓词，例如，=、BETWEEN、LIKE 和 <>。但是，Entrez 搜索引擎将只处理这些关系谓词中的某些关系谓词。Entrez 搜索引擎不处理的关系谓词将由 DB2<sup>®</sup> 来处理。Entrez 搜索引擎处理每种模式的某些标识列上的等号 (=) 和 IN 谓词。这些谓词允许 Entrez 包装器绕过搜索阶段而直接执行访存阶段。以下是有效谓词的示例：

```
WHERE pmid = '1234567'  
WHERE medlineid IN ('1234567', '9191919')
```

可以在此类谓词中使用的列是由模式表的 F 列标识的。此选项的值必须为 Y。

### 相关概念：

- 第 188 页的『Entrez 包装器的无效 WHERE 子句』

### 相关任务：

- 第 189 页的『Entrez 数据源 - 示例查询』
- 第 181 页的『为 Entrez 包装器注册定制函数』

## Entrez 包装器的无效 WHERE 子句

Entrez 包装器将拒绝会导致对 NCBI 数据库进行非限定扫描的任何查询。有效的 WHERE 子句必须在模式的主标识上或者在定制函数中包含等号 (或者 IN) 谓词。不满足这些条件的查询将被拒绝，并且会提供错误代码 SQL0142N 或 SQL30090N。

### 相关概念：

- 第 188 页的『Entrez 包装器的关系谓词』

### 相关任务：

- 第 189 页的『Entrez 数据源 - 示例查询』
- 第 181 页的『为 Entrez 包装器注册定制函数』

## 模式数据元素规范

当通过 SQL 模式来提供几个数据元素时，它们将被转换为一种规范格式。这些数据元素包括项列表、姓名和日期。

### 项列表

除非另有声明，被分解为不规范的单一列的各项的列表使各个项由分号和单个空格隔开。例如，如果一个条目包含关键字 (keywords) dnaA gene、dnaN gene 和 orf187，则相应的 Keywords 列将包含值 dnaA gene、dnaN gene 和 orf187。

### 姓名

NCBI 模式中的姓名由必需的姓和几个可选元素之一组成。这些可选元素中的某些元素可以同时出现，而其它元素是互相排斥的。要创建一个规范格式的姓名，应指定这些元素的优先顺序。按照从最高到最低的顺序，这些元素是：

- 名
- 名字或中间名

- 首字母

提供姓名时可以带有或者不带从属关系。如果不带从属关系，则姓名的格式为 <last name>, <first>, 其中 <first> 是可选元素之一。如果找不到 <first> 元素，则不使用逗号。可以按照 (<affiliation>) 格式来添加从属关系。

使用分号和空格来分隔不规范列表中的姓名。分隔姓名的正确方法的示例为:

Parker, M. J.; Ranjan, K. A.

## 日期

日期（特别是发布日期）在 NCBI 模式中具有各种各样的格式。为了接受这些格式和尽可能允许进行日期比较和日期运算，SQL 模式中的日期是以两种格式表示的。第一，日期可以是字符串。第二，日期可以是类型为 DATE 的一列。

如果日期值中只提供了月份而没有提及日，则该月份的第一天就是缺省日。如果提供了季节而没有提供月份，或者没有提供月份和日，则使用该季节的第一天。

### 相关任务:

- 第 185 页的『为 Entrez 数据源注册昵称』

### 相关参考:

- 第 186 页的『定制函数和 Entrez 查询』

## Entrez 数据源 - 示例查询

本主题提供了一些要对 Entrez 数据源运行的样本查询。

### 过程:

要运行查询，将下列示例用作指南。

### 在 PubMed 昵称上:

以下显示了在 PubMed 昵称上具有单个访存键的一个查询:

```
select PMID, ArticleTitle FROM pmarticles WHERE pmid = '12345';
```

以下显示了在 PubMed 昵称上具有混合访存键的一个查询:

```
select PMID, ArticleTitle FROM pmarticles
WHERE pmid = '12345' OR MedlineID = '12346';
```

以下显示了在 PubMed 昵称上具有 CONTAINS 函数的一个查询:

```
select PMID, ArticleTitle FROM pmarticles
WHERE entrez.contains (ArticleTitle, 'granulation') = 1
AND entrez.contains (PubDate, '1992') = 1;
```

以下显示了在 PubMed 昵称上搜索指定的 AuthorList 和 LanguageList 的一个查询:

```
select PMID, ArticleTitle FROM pmarticles
WHERE entrez.contains (AuthorList, 'Albarrak') = 1
AND entrez.contains (LanguageList, 'eng')=1;
```

以下显示了在 PubMed 昵称上具有复杂谓词的一个查询:

```
select PMID, ArticleTitle FROM pmarticles
WHERE entrez.contains (PublicationTypeList, 'Journal Article') = 1
AND entrez.contains (MedlineTA, 'sun')=1
OR entrez.contains (PersonalNameSubjectList, 'shine')=1;
```

在 **Nucleotide** 昵称上:

以下显示了在 **Nucleotide** 昵称上具有多个访问键的一个查询:

```
select PrimaryAccession, LocusName, SeqLength from gbseq
WHERE PrimaryAccession in ('NM_000890', 'NC_003106');
```

以下显示了在 **Nucleotide** 昵称上搜索所有可搜索字段的一个查询:

```
select PrimaryAccession, substr(Definition,1,300), GI from gbseq
WHERE entrez.contains(AllText, 'abcde')=1;
```

以下显示了在 **Nucleotide** 昵称上搜索所有自由文本的一个查询:

```
select * from gbseq WHERE entrez.contains(FreeText, 'abcde')=1;
```

以下显示了在 **Nucleotide** 昵称上搜索定义的一个查询:

```
select PrimaryAccession, substr(Definition,1,300), version, GI from gbseq
WHERE entrez.contains(Definition, 'Sulfolobus tokodaii
AND complete genome') = 1;
```

以下显示了在 **Nucleotide** 昵称上搜索关键字的一个查询:

```
select PrimaryAccession, substr(KeywordList,1,200), Segment from gbseq
WHERE entrez.contains(KeywordList, 'nkcc1 gene') = 1;
```

相关概念:

- 第 188 页的『Entrez 包装器的关系谓词』
- 第 188 页的『Entrez 包装器的无效 WHERE 子句』

相关任务:

- 第 181 页的『为 Entrez 包装器注册定制函数』

## PubMed 模式表

PubMed 模式定义 PubMed 类型服务器中的数据的外观。该模式由几个相关昵称组成:

- PMArticles
- PMAccession
- PMChemical
- PMMeSHHeading
- PMComments
- PMArticleID

下列各表列示了有关每个昵称中的列的信息。“标记”列包含这些列的有效搜索标记。有关有效搜索标记的列表，请参阅以下 Web 站点并找到指向 Search Field Descriptions and Tags 的链接:

[www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html](http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html)

当创建昵称时，可以覆盖这些列的缺省数据类型。Entrez 包装器支持 CLOB 数据类型，长度最大可为 5 兆字节。

当创建昵称时，可以覆盖这些列的缺省长度。例如，某些列可以返回大量数据，例如，PMArticles 昵称中的 Abstract 列。此列的缺省长度为 VARCHAR(32000)。要返回该列的前 100 个字节，可以使用数据类型 VARCHAR(100) 来定义该列。将只返回前 100 个字节。

### PMArticles 昵称:

下表描述了 PMArticles 昵称中的各列。F 列指示是指定访存键的列。使用访存键可以加快查询处理的速度。

表 48. PubMed PMArticles 昵称

列名	数据类型	描述	标记	访存键
PMID	VARCHAR(10) NOT NULL	用来连接 PMArticles 昵称与子昵称的主键列	UID	是
MedlineID	VARCHAR(10)	Medline ID	UID	是
Owner	VARCHAR(8) NOT NULL	出版物条目的所有者；值是由 NCBI 定义的，这些值可能是 NLM、NASA、PIP、KIE、HSR、HMD、SIS 和 NOTNLM。如果未提供值，则缺省值为 NLM。	无	否
Status	VARCHAR(32) NOT NULL	由 NCBI 定义的出版物状态。值可能包括：In-Process、Completed、Out-of-scope 和 PubMed-not_MEDLINE。	无	否
DateCreated	DATE NOT NULL	创建出版物条目的日期。	无	否
DateCompleted	DATE	完成出版物条目的日期。	无	否
DateRevised	DATE	修订出版物条目的日期。	无	否
ArticleTitle	VARCHAR(250) NOT NULL	文章的标题。	TI	否
Pagination	VARCHAR(32)	文章的完全标记页数。	无	否
Abstract	VARCHAR(32000)	文章的摘要。	TIAB	否
Affiliation	VARCHAR(250)	第一个作者的从属关系和地址	AD	否
AuthorList	VARCHAR(3200)	作者的列表；已规范化	AU	否
LanguageList	VARCHAR(250) NOT NULL	用分号分隔的列表	LA	否
PublicationTypeList	VARCHAR(250) NOT NULL	用分号分隔的列表	PT	否
VernacularTitle	VARCHAR(250)	文章的本国语标题。	无	否
DateOfElectronic 出版物	VARCHAR(32)	NCBI 模式没有指定此列的结构	无	否
Country	VARCHAR(128)	日志中列举的出版物的国家或地区。	无	否

表 48. PubMed PMArticles 昵称 (续)

列名	数据类型	描述	标记	访 存 键
MedlineTA	VARCHAR(250) NOT NULL	Medline 标题缩写。	TA	否
NlmUniqueID	VARCHAR(32)	包含 MedlineCode (如果未提供 NlmUniqueID 的话)	无	否
GeneSymbolList	VARCHAR(250)	用分号分隔的列表; 自从 1996 年以来都没有使用	无	否
NumberOfReferences	INTEGER	评论文章的书目引用数。	无	否
PersonalNameSubjectList	VARCHAR(250)	已被规范为用分号分隔的名称列表	PS	否
KeywordList	VARCHAR(3200)	用分号分隔的列表	无	否
SpaceFlightMissionList	VARCHAR(250)	用分号分隔的列表	无	否
InvestigatorList	VARCHAR(250)	已被规范为用分号分隔的名称列表	无	否
PublicationStatus	VARCHAR(32)	出版物的状态。	无	否
ProviderID	VARCHAR(32)	出版物供应商标识。	无	否
CitationSubsetList	VARCHAR(250)	用分号分隔的列表	SB	否
AllFields	VARCHAR(1)	伪列; 始终返回 NULL	ALL	否
TextWords	VARCHAR(1)	伪列; 始终返回 NULL	TW	否
PubDate	DATE	包括杂志和书籍出版日期和 medline 日期	DP	否
PubDateString	VARCHAR(32)	包括杂志和书籍出版日期和 medline 日期	DP	否
Title	VARCHAR(250)	书籍或杂志标题	TA	否
Journal_ISSN	CHAR(9)	日志的 ISSN。	TA	否
Journal_Volume	VARCHAR(10)	日志的容量。	VI	否
Journal_Issue	VARCHAR(10)	日志的发行。	IP	否
Journal_Coden	VARCHAR(32)	日志的代码号 (coden)。	无	否
Journal_ISOAbbreviation	VARCHAR(32)	日志的 ISO 缩写。	无	否
Book_Publisher	VARCHAR(128)	书籍的发行人。	无	否
Book_Authors	VARCHAR(250)	已规范化为其它作者列表	无	否
Book_CollectionTitle	VARCHAR(128)	书籍的收集标题。	无	否
Book_Volume	VARCHAR(10)	书籍的卷。	无	否

#### PMAccession 昵称:

下表描述了 PMAccession 昵称中的各列。

表 49. PubMed PMAccession 昵称

列名	数据类型	描述	标记
PMID	VARCHAR(10) NOT NULL	用来连接 PMAccession 子昵称和它的父昵称的键。	无



表 49. PubMed PMAccession 昵称 (续)

列名	数据类型	描述	标记
DataBankName	VARCHAR(250) NOT NULL	数据银行的名称。	SI
Accession	VARCHAR(32) NOT NULL	存取号。	SI

### PMChemical 昵称:

下表描述了 PMChemical 昵称中的各列。

表 50. PubMed PMChemical 昵称

列名	数据类型	描述	标记
PMID	VARCHAR(10) NOT NULL	用来连接 PMChemical 子昵称和它的父昵称的键。	无
NameOfSubstance	VARCHAR(128) NOT NULL	物质的名称。	NM
RegistryNumber	VARCHAR(32) NOT NULL	可能是 CAS 或其它注册编号	RN
CASRegistry	CHAR	Y 或 N	无

### PMMESHHeading 昵称:

下表描述了 PMMeSHHeading 昵称中的各列。

表 51. PubMed PMMeSHHeading 昵称

列名	数据类型	描述	标记
PMID	VARCHAR(10) NOT NULL	用来连接 PMMeSHHeading 子昵称和它的父昵称的键。	ID
DescriptorOrName	VARCHAR(128) NOT NULL	MeSH 的描述符的名称。	MH <sup>1</sup>
DescriptorIsMajor	CHAR NOT NULL	如果描述符是主要的, 则为 Y	无
QualifierOrSubhead	VARCHAR(128)	MeSH 的限定符或小标题。	SH
QSIsMajor	CHAR	如果限定符或小标题是主要的, 则为 Y	无

注:

1. 如果查询中包括谓词 “DescriptorIsMajor = Y”, 则搜索项为 MAJR。

### PMComments 昵称:

下表描述了 PMComments 昵称中的各列。

表 52. PubMed PMComments 昵称

列名	数据类型	描述	标记
PMID	VARCHAR(10) NOT NULL	用来连接 PMComments 子昵称和它的父昵称的键。	无

表 52. PubMed PMComments 昵称 (续)

列名	数据类型	描述	标记
RefSource	VARCHAR(128) NOT NULL	引用的源。	无
Type	VARCHAR(32) NOT NULL	CommentOn、CommentIn、ErratumIn、ErratumFor、RepublishedFrom、RepublishedIn、RetractionOf、RetractionIn、UpdateIn、UpdateOf、SummaryForPatents 和 OriginalReportIn	无
Note	VARCHAR(3200)	注释	无

### PMArticleID 昵称:

下表描述了 PMArticleID 昵称中的各列。

表 53. PubMed PMArticleID 昵称

列名	数据类型	描述	标记
PMID	VARCHAR(10) NOT NULL	用来连接 PMArticleID 子昵称和它的父昵称的键。	无
ArticleID	VARCHAR(32) NOT NULL	文章的标识。	无
IdType	VARCHAR(8) NOT NULL	doi、pii、pmcpid、pmpid、sici、pubmed、medline 和 pmcid	无

### 相关参考:

- 第 188 页的『模式数据元素规范』
- 第 186 页的『定制函数和 Entrez 查询』
- 第 194 页的『Nucleotide 模式表』

## Nucleotide 模式表

Nucleotide 模式定义 Nucleotide 类型服务器的数据的外观。该模式由几个相关昵称组成:

- GBSeq
- GBReference
- GBFeatures
- GBIntervals
- GBQualifiers

下列各表列示了有关昵称中的列的信息。“限定符”列包含该列的有效搜索限定符。有关有效搜索标记的列表, 请参阅以下 Web 站点并找到指向 Search Field Descriptions and Qualifiers 的链接:

[www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary\\_Matrices.html](http://www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary_Matrices.html)

当创建昵称时, 可以覆盖这些列的缺省数据类型。例如, GBSeq 昵称中的“序列”列的缺省数据类型为 VARCHAR(32000)。可以将此数据类型更改为 CLOB(1MB)。Entrez 包装器支持 CLOB 数据类型, 长度最大可为 5 兆字节。

## GBSeq:

下表描述了 GBSeq 昵称中的各列。F 列指示是指定访存键的列。使用访存键可以加快查询处理的速度。

表 54. Nucleotide GBSeq 昵称

列名	数据类型	描述	限定符	访存键
PrimaryAccession	VARCHAR(16) NOT NULL	主存取号	PACC	是
SequenceKey	VARCHAR(32) NOT NULL	用来连接 GBSeq 昵称与子昵称的主键列。		否
LocusName	VARCHAR(16) NOT NULL	轨迹的名称。	ACCN	否
SeqLength	INTEGER NOT NULL	序列的长度。	SLEN	否
Strandedness	VARCHAR(32)	not-set、single-stranded、double-stranded 和 mixed-stranded	无	否
MoleculeType	VARCHAR(16)	nucleic-acid、dna、rna、trna、rrna、mrna、urna、srna、snorna 和 peptide	PROP	否
Topology	VARCHAR(16)	直线型和圆形	无	否
Division	CHAR(3) NOT NULL	GenBank 部分。	PROP	否
UpdateDate	DATE NOT NULL	最新更新的日期。	MDAT	否
CreateDate	DATE NOT NULL	创建记录的日期。	无	否
Definition	VARCHAR(7000) NOT NULL	序列的定义行。	TITL	否
Version	INTEGER	序列的版本标识。	无	否
GI	VARCHAR(16)	GenInfo (GI) 序列标识。	无	否
KeywordList	VARCHAR(7000)	用分号分隔的列表	KYWD	否
Segment	VARCHAR(250)	段。	无	否
Source	VARCHAR(200) NOT NULL	源。	ORGN	否
Organism	VARCHAR(7000) NOT NULL	组织。	ORGN	否
Taxonomy	VARCHAR(7000) NOT NULL	分类法。	无	否
Comment	VARCHAR(7000)	注释	无	否
Primary	VARCHAR(7000)	主要。	无	否
SourceDB	VARCHAR(250)	源数据库。	无	否
Sequence	VARCHAR(32000)	序列。	无	否
AllText	VARCHAR(1)	伪列, 始终返回 NULL	ALL	否
FreeText	VARCHAR(1)	伪列, 始终返回 NULL	WORD	否

## GBReference:

下表描述了 GBReference 昵称中的各列。

表 55. Nucleotide GBReference 昵称

列名	数据类型	描述	限定符
SequenceKey	VARCHAR(32) NOT NULL	用来连接 GBReference 子昵称和它的父昵称的键。	无
ReferenceNum	INTEGER NOT NULL	从 GBReference_reference 进行语法分析	无
RangeLow	INTEGER NOT NULL	要参考的基本库（从 GBReference_reference 进行语法分析）	无
RangeHigh	INTEGER NOT NULL	要参考的高级库（从 GBReference_reference 进行语法分析）	无
Authors	VARCHAR(3200)	采用 GenBank 格式的用分号分隔的名称列表	AUTH
Consortium	VARCHAR(250)	财团。	无
Title	VARCHAR(250)	GenBank 引用标题。	WORD
Journal_Title	VARCHAR(250) NOT NULL	日志的标题。	JOUR
MedlineID	INTEGER	Medline 标识	无
PubMedID	INTEGER	PubMed 标识	无
Remarks	VARCHAR(3200)	注释	无

### GBFeatures:

下表描述了 GBFeatures 昵称中的各列。

表 56. Nucleotide GBFeatures 昵称

列名	数据类型	描述	限定符
SequenceKey	VARCHAR(32) NOT NULL	用来连接 GBFeatures 子昵称和它的父昵称的键。	无
FeatureJoinKey	VARCHAR(32) NOT NULL	用来连接 GBFeatures 昵称与子昵称的主键列。	无
FeatureKey	VARCHAR(20) NOT NULL		FKEY
FeatureLocation	VARCHAR(200) NOT NULL		无

### GBIntervals:

下表描述了 GBIntervals 昵称中的各列。

表 57. Nucleotide GBIntervals 昵称

列名	数据类型	描述	限定符
FeatureJoinKey	VARCHAR(32) NOT NULL	用来连接 GBIntervals 子昵称和它的父昵称的键。	无

表 57. Nucleotide GBIntervals 昵称 (续)

列名	数据类型	描述	限定符
IntervalFrom	INTEGER		无
IntervalTo	INTEGER		无
IntervalPoint	INTEGER		无
IntervalAccession	VARCHAR(32) NOT NULL		无

### GBQualifiers:

下表描述了 GBQualifiers 昵称中的各列。

表 58. Nucleotide GBQualifiers 昵称

列名	数据类型	描述	限定符
FeatureJoinKey	VARCHAR(32) NOT NULL	用来连接 GBQualifiers 子昵称和它的父昵称的键。	无
QualifierName	VARCHAR(50)	限定符的名称	无
QualifierValue	VARCHAR(32000)	限定符的值	无

### 相关参考:

- 第 188 页的『模式数据元素规范』
- 第 190 页的『PubMed 模式表』
- 第 186 页的『定制函数和 Entrez 查询』

## Entrez 包装器的消息

本主题描述了您在使用 Entrez 的包装器时可能会遇到的消息。对于未在此表、《消息参考: 第 1 卷》或《消息参考: 第 2 卷》中说明的消息, 请与 IBM 软件支持机构联系。

表 59. 由 Entrez 的包装器发出的消息

错误代码	消息	解释
SQL0142N	SQL 语句不受支持。	将无效的查询类型传递给了包装器。检查此包装器是否支持所发出的 SQL 语句。
SQL0204N	“<name>” 是一个未定义的名称。	指定的名称无效。检查 CREATE NICKNAME 语句。
SQL0405N	数字文字 “<literal>” 无效, 原因是它的值超出了范围。	已检索的 XML 数据中的一列或者 SQL 语句中的谓词包含超出了该数据类型的可能范围的值。检查此列的数据类型和数据源中的该列, 或者将该列重新定义为更合适的类型。

表 59. 由 Entrez 的包装器发出的消息 (续)

错误代码	消息	解释
SQL0408N	值与它的赋值目标的数据类型不兼容。目标名称为“<target_name>”。	XML 数据中的一列包含对于该数据类型无效的字符。检查此列的数据类型和数据源中的该列，或者将该列重新定义为更合适的类型。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“找不到数据库原型。”)	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“没有要解包的数据。”)	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“创建包装器对象时出错。”)	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“表达式类型错误。”)	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“找不到昵称。”)	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因：“内存分配错误。”)	没有足够的内存用来处理包装器内部的分配请求。
SQL1816N	包装器“<wrapper_name>”不能用来存取您正在尝试向联合服务器定义的数据源的“版本”(“<server_type>”和“<server_version>”)。	CREATE SERVER 语句的 VERSION 子句中的一个值无效。
SQL1816N	包装器“<wrapper_name>”不能用来存取您正在尝试向联合服务器定义的数据源的“类型”(“<server_type>”和“<server_version>”)。	CREATE SERVER 语句的 TYPE 子句中的一个值无效。
SQL1817N	CREATE SERVER 语句未标识您想要向联合数据库定义的数据源的“类型”。	CREATE SERVER 语句的 TYPE 子句是必需的，但是未指定。
SQL1822N	从数据源“Entrez 包装器”中接收到意外的错误代码“900”。相关联的文本和标记为“未定义父代昵称”。	这是一个内部错误。请与 IBM 软件支持机构联系。

表 59. 由 Entrez 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1823N	对于服务器 “<server_name>” 中的数据类型 “<data_type>” 不存在数据类型映射。	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL1881N	“<option_name>” 不是 “<option_name>” 的有效 “<option_type>”。	指定的选项是无效选项。检查 CREATE NICKNAME 语句。
SQL1882N	“<option_type>” 选项 “<option_name>” 不能设置为 “<option_name>” 的 “<option_value>”。	指定的值对于此选项无效。检查 CREATE NICKNAME 语句。
SQL1883N	“<option_name>” 是 “<option_name>” 的必需 “<option_type>” 选项。	对于该对象，指定的选项是必需的，但是未指定。检查 CREATE NICKNAME 语句。
SQL1884N	多次指定了 “FOREIGN_KEY” (这是一个 “COLUMN” 选项)。	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL1884N	多次指定了 “PRIMARY_KEY” (这是一个 “COLUMN” 选项)。	这是一个内部错误。请与 IBM 软件支持机构联系。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “不能更改服务器版本”。	不能通过发出 ALTER SERVER 语句来更改服务器的版本。必须使用新版本来创建一个新服务器。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “PARENT 昵称无效”。	在 PARENT 昵称选项中引用的昵称对于当前昵称无效。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “列名无效”。	在 CREATE NICKNAME 语句中指定的列名与昵称的任何可能列都不匹配。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “不能对访存键进行 AND 操作”。	在一个逻辑 “与” 操作中多次引用了一个访存键 (例如, PMArticles 昵称的 PMID 列)。例如, “PMID = 12346 AND PMID = 12348”。只能使用 OR 来关联访存键谓词。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “混合了 SEARCH_TERM 和 CONTAINS 函数”。	在查询中不能混合 SEARCH_TERM 和 CONTAINS 函数。每个查询中只允许存在一个 SEARCH_TERM 函数。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “函数中的第一个自变量无效”。	SEARCH_TERM 或 CONTAINS 函数的第一个自变量无效。此自变量必须是对一系列的引用。

表 59. 由 *Entrez* 的包装器发出的消息 (续)

错误代码	消息	解释
SQL30090N	操作对于应用程序执行环境无效。原因码 = “函数中的第二个自变量无效”。	SEARCH_TERM 或 CONTAINS 函数的第二个自变量无效。此自变量必须是字符串文字、主变量或列引用。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “CONTAINS 函数中具有未标记列”。	CONTAINS 函数的第一个自变量无效。此自变量必须是对一个已标记列的引用。
SQL30090N	操作对于应用程序执行环境无效。原因码 = “函数无效”。	这是一个内部错误。请与 IBM 软件支持机构联系。

相关概念:

- 『消息介绍』 (《消息参考》第 1 卷)

相关参考:

- 『SQLSTATE 消息』 (《消息参考》第 2 卷)



## 第 12 章 配置对 Excel 数据源的存取

本章说明如何配置联合服务器以存取存储在 Excel 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Excel 数据源的存取。

本章包含下列内容:

- 说明什么是 Excel
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 Excel 包装器相关联的错误消息

### 什么是 Excel?

Excel 电子表格或工作簿是使用 Microsoft® Excel 应用程序创建的文件，文件扩展名为 xls。DB2® Information Integrator 支持 Excel 97、Excel 2000 和 Excel 2002 的电子表格。图 23 说明了 Excel 包装器如何将您的电子表格连接至联合系统。

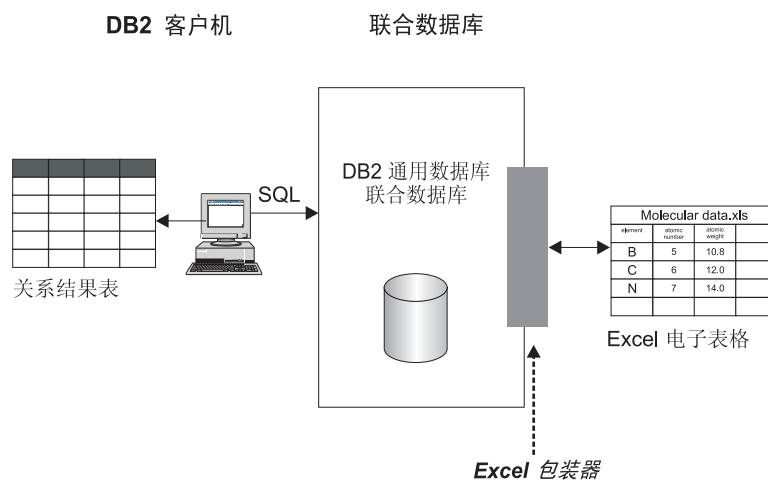


图 23. Excel 包装器的工作方式

Excel 包装器使用 CREATE NICKNAME 语句来将 Excel 电子表格中的列映射至 DB2 UDB 联合系统中的列。表 60 显示了存储在名为 Compound\_Master.xls 的文件中的样本电子表格数据。

表 60. Compound\_Master.xls 的样本电子表格

	A	B	C	D
1	COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
2	compound_A	1.23	367	tested
3	compound_G		210	
4	compound_F	0.000425536	174	tested
5	compound_Y	1.00256		tested

表 60. *Compound\_Master.xls* 的样本电子表格 (续)

	A	B	C	D
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested

通常，通过标准的 SQL 命令不能为您提供此信息。当安装并注册了 Excel 包装器之后，就可以将此信息当作标准的关系数据源来存取。例如，如果您想知道分子数大于 100 的化合物数据，则可以运行以下 SQL 查询：

```
SELECT * FROM compound_master WHERE mol_count > 100
```

查询结果显示在表 61 中。

表 61. 查询结果

COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
compound_A	1.23	367	tested
compound_G		210	
compound_F	0.000425536	174	tested
compound_Q		1024	

相关概念：

- 第 22 页的『用来存取 Excel 数据的方法』

相关任务：

- 第 202 页的『将 Excel 数据源添加至联合服务器』

## 将 Excel 添加至联合服务器

### 将 Excel 数据源添加至联合服务器

要配置联合服务器以存取 Excel 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Excel 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

先决条件：

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上
- 正确构造的 Excel 工作表，以便包装器可以存取数据

过程：

要将 Excel 数据源添加至联合服务器：

1. 注册包装器。

2. 注册服务器定义。
3. 为 Excel 工作表注册昵称。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 22 页的『用来存取 Excel 数据的方法』
- 第 201 页的『什么是 Excel?』
- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 203 页的『注册 Excel 包装器』
- 第 204 页的『为 Excel 数据源注册服务器』
- 第 205 页的『为 Excel 数据源注册昵称』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 22 页的『选择正确的包装器』

## 注册 Excel 包装器

注册 Excel 包装器是将 Excel 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Excel 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 限制:

- Excel 包装器只可用于支持“DB2 通用数据库企业服务器版”的 Microsoft Windows 操作系统。
- 必须在安装了 DB2 Information Integrator 的服务器上安装了 Excel 应用程序之后才能使用 Excel 包装器。
- 不允许进行传递会话。

#### 过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如，要注册一个名称为 excel\_wrapper 的包装器，发出以下语句：

```
CREATE WRAPPER excel_wrapper LIBRARY 'db2lsx1s.dll';
```

必须在 CREATE WRAPPER 语句中指定包装器库文件 db2lsx1s.dll。

在这一系列任务中的下一个任务是为 Excel 包装器注册服务器定义。

**相关任务:**

- 第 204 页的『为 Excel 数据源注册服务器』

**相关参考:**

- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）

## Excel 包装器库文件

下表列示了 Excel 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把此库文件添加到该表中列示的目录路径中。

当注册包装器时，只指定该表中列示的库文件名。

表 62. Excel 包装器库位置和文件名

操作系统	目录路径	包装器库文件
Windows	%DB2PATH%\bin	db2lsxls.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

**相关任务:**

- 第 203 页的『注册 Excel 包装器』

## 为 Excel 数据源注册服务器

为 Excel 数据源注册服务器是将 Excel 添加至联合系统这一大型任务的一部分。在注册包装器之后，必须注册相应的服务器。

对于 Excel，将会创建服务器定义，这是因为联合对象的层次结构需要数据源文件（由昵称标识）与特定服务器对象相关联。

**过程:**

要向联合系统注册 Excel 服务器，使用 CREATE SERVER 语句。

假定您想为包含生物化学数据的工作簿创建称为 `biochem_lab` 的服务器对象。该服务器对象必须与使用 CREATE WRAPPER 语句注册的 Excel 包装器相关联。注册此服务器对象的 CREATE SERVER 语句是：

```
CREATE SERVER biochem_lab WRAPPER Excel_2000_Wrapper;
```

在这一系列任务中的下一个任务是为 Excel 数据源注册昵称。

**相关任务:**

- 第 205 页的『为 Excel 数据源注册昵称』

**相关参考:**

- 『CREATE SERVER statement』（*SQL Reference, Volume 2*）
- 第 503 页的『CREATE SERVER 语句自变量 - Excel 包装器』

## 为 Excel 数据源注册昵称

为 Excel 数据源注册昵称是将 Excel 添加至联合系统这一大型任务的一部分。在注册服务器之后，必须注册相应的昵称。当您在查询中引用 Excel 数据源时就要使用昵称。

### 过程:

要将 Excel 数据源映射至关系表，使用 CREATE NICKNAME 语句创建一个昵称。

以下示例中的语句将从名为 CompoundMaster.xls 的 Excel 电子表格文件来创建 Compounds 昵称。该文件包含三列数据，它们向联合系统分别定义为 Compound\_ID、CompoundName 和 MolWeight。

```
CREATE NICKNAME Compounds (  
    Compound_ID INTEGER,  
    CompoundName VARCHAR(50),  
    MolWeight FLOAT)  
FOR SERVER biochem_lab  
OPTIONS (FILE_PATH 'C:\My Documents\CompoundMaster.xls',  
        RANGE 'B2:D5');
```

在这一系列任务中没有别的任务了。

### 相关任务:

- 第 202 页的『将 Excel 数据源添加至联合服务器』
- 第 58 页的『为非关系数据源指定昵称列』

### 相关参考:

- 第 205 页的『Excel 数据源 - 示例查询』
- 第 504 页的『CREATE NICKNAME 语句语法 - Excel 包装器』

---

## Excel 数据源 - 示例查询

本主题列示了几个使用示例昵称 Compounds 的样本 Excel 电子表格查询。

要运行查询，按照您使用常规表名和表列的相同方式在 SQL 语句中使用昵称和已定义的昵称列。

以下查询将显示分子量大于 2000 的所有 compound\_ID:

```
SELECT compound_ID  
FROM Compounds  
WHERE MolWeight > 200;
```

以下查询将显示化合物名称或分子量为空值的所有记录:

```
SELECT *  
FROM Compounds  
WHERE CompoundName IS NULL  
OR MolWeight IS NULL;
```

以下查询将显示化合物名称中包含字符串 ase 并且分子量大于或等于 300 的所有记录:

```
SELECT *  
FROM Compounds  
WHERE CompoundName LIKE '%ase%'  
AND MolWeight >=300;
```

### 相关参考:

- 第 172 页的『Documentum 数据源 - 示例查询』
- 第 206 页的『Excel 数据源 - 样本方案』

---

## Excel 数据源 - 样本方案

本节演示 Excel\_2000 包装器存取位于 C:\Data 目录中的 Excel 2000 工作表的样本实现。该方案将注册包装器、注册服务器和注册一个将用来存取工作表的昵称。该方案中显示的语句是使用 DB2 命令行输入的。在注册包装器之后，就可以对工作表运行查询了。

该方案从一个称为 Compound\_Master.xls 的化合物工作表开始，它一共有 4 列 9 行。该文件的全限定路径名为 C:\Data\Compound\_Master.xls。在表 63 中显示了其内容。

表 63. 样本工作表 *Compound\_Master.xls*

	A	B	C	D
1	COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
2	compound_A	1.23	367	tested
3	compound_G		210	
4	compound_F	0.000425536	174	tested
5	compound_Y	1.00256		tested
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested

### 过程:

要使用 Excel 包装器来存取 Excel 2000 工作表:

1. 注册 Excel\_2000 包装器:

```
db2 => CREATE WRAPPER Excel_2000 LIBRARY 'db2lsxls.dll'
```

2. 注册服务器:

```
db2 => CREATE SERVER biochem_lab WRAPPER Excel_2000
```

3. 注册表示 Excel 工作表的昵称:

```
db2 => CREATE NICKNAME Compound_Master (compound_name VARCHAR(40),  
weight FLOAT, mol_count INTEGER, was_tested VARCHAR(20))  
FOR biochem_lab  
OPTIONS ( FILE_PATH 'C:\Data\Compound_Master.xls')
```

这就完成了注册过程。Excel 数据源现在就是联合系统的一部分了，并且可以用于 SQL 查询。

下列示例使用 Excel 数据源来说明了样本 SQL 查询以及获得的结果。

- 样本 SQL 查询: “请提供 mol\_count 大于 100 的所有化合物数据。”

```
SELECT * FROM compound_master WHERE mol_count > 100
```

结果: 第 2、3、4、6 和 8 行的所有字段。

- 样本 SQL 查询: “请提供尚未确定 mol\_count 的所有化合物的 compound\_name 和 mol\_count。”

```
SELECT compound_name, mol_count FROM compound_master
WHERE mol_count IS NULL
```

结果: 工作表的第 5、7 和 10 行的字段 compound\_name 和 mol\_count。

- 样本 SQL 查询: “计算尚未测试并且重量大于 1 的化合物的数量。”

```
SELECT count(*) FROM compound_master
WHERE was_tested IS NULL AND weight > 1
```

结果: 记录数为 1, 它表示工作表中满足条件的单一行第 7 行。

- 样本 SQL 查询: “请提供 mol\_count 已经确定并且小于平均 mol\_count 的所有化合物的 compound\_name 和 mol\_count。”

```
SELECT compound_name, mol_count
FROM compound_master
WHERE mol_count IS NOT NULL
AND mol_count < (SELECT AVG(mol_count) FROM compound_master
WHERE mol_count IS NOT NULL AND was_tested IS NOT NULL)
```

子查询将平均值 368 返回给主查询, 然后主查询返回表 64:

表 64. 查询结果

COMPOUND_NAME	MOL_COUNT
compound_A	367
compound_G	210
compound_F	174
compound_S	67

**相关任务:**

- 第 202 页的『将 Excel 数据源添加至联合服务器』

**相关参考:**

- 第 205 页的『Excel 数据源 - 示例查询』

## Excel 包装器的文件访问控制模型

数据库管理系统将使用 DB2 数据库服务的 LOG ON AS 属性的权限来存取 Excel 文件。可以在 DB2 实例的 LOG ON 属性页面中查看此设置。通过 Windows NT “服务”控制面板来存取该属性页面。

**相关参考:**

- 第 335 页的『表结构文件包装器的文件访问控制模型』
- 第 173 页的『Documentum 包装器的访问控制』

## Excel 包装器的消息

本节列示和描述了您在使用 Excel 的包装器时可能会遇到的消息。

表 65. 由 Excel 的包装器发出的消息

错误代码	消息	解释
SQL1817N	CREATE SERVER 语句未标识您想要向联合数据库定义的数据源的“VERSION”。	在 CREATE SERVER 语句中未指定 VERSION 参数。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1000.<internal program code>”。相关联的文本和标记为“内存分配错误”	请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1001.<internal program code>”。相关联的文本和标记为“未知选项”。	不支持在 DDL 语句中指定的选项。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1002.<internal program code>”。相关联的文本和标记为“未能创建 DELTA 对象”。	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1100.<internal program code>”。相关联的文本和标记为“包装器选项不受支持”。	此包装器不支持包装器 OPTIONS。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1200.<internal program code>”。相关联的文本和标记为“<option> 是一个不受支持的 Server 选项”。	此包装器不支持所指定的选项。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1201.<internal program code>”。相关联的文本和标记为“获取服务器名时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1209. <internal program code>”。相关联的文本和标记为“转换 VARCHAR 数据时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1211.<internal program code>”。相关联的文本和标记为“转换 INTEGER 数据时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1212.<internal program code>”。相关联的文本和标记为“转换 FLOAT 数据时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。



表 65. 由 Excel 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1400.<internal program code>”。相关联的文本和标记为“<option> 是一个不受支持的 User 选项”	此包装器不支持所指定的选项。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1401.<internal program code>”。相关联的文本和标记为“未能创建 USER Delta 对象”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1500.<internal program code>”。相关联的文本和标记为“<option> 是一个不受支持的 Nickname 选项”	此包装器不支持所指定的选项。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1501.<internal program code>”。相关联的文本和标记为“未指定必需的选项 PATH”	注册 NICKNAME 时必需具有 PATH 选项。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1502.<internal program code>”。相关联的文本和标记为“未能创建 NICKNAME Delta 对象”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1503.<internal program code>”。相关联的文本和标记为“获取“昵称”列类型时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1504.<internal program code>”。相关联的文本和标记为“获取“昵称”列类型名称时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1505.<internal program code>”。相关联的文本和标记为“从数据源“Excel 包装器”中接收到”。	此包装器不支持所指定的 <data type>。更正 SQL 语句，然后再次运行它。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1506.<internal program code>”。相关联的文本和标记为“获取“昵称”列信息时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。

表 65. 由 Excel 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1507.<internal program code>”。相关联的文本和标记为“不能删除 <option> 选项”	不能删除指定的选项，因为它是一个必需的选项。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1508.VANI”。相关联的文本和标记为“不能改变列名”	Excel 包装器不允许改变列名。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1509.VCTS”。相关联的文本和标记为“找不到列信息”。	找不到列信息。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1701.<internal program code>”。相关联的文本和标记为“对 SQL 进行解析时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1702.<internal program code>”。相关联的文本和标记为“存取 NICKNAME 对象时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1703.<internal program code>”。相关联的文本和标记为“构建数据存贮区时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1704.<internal program code>”。相关联的文本和标记为“将 SQL 链接至“昵称数据”时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1705.<internal program code>”。相关联的文本和标记为“Excel 应用程序启动失败”	启动 Excel 应用程序失败。确认 Excel 是否已经安装在系统上并且已经以正确版本的包装器进行了注册。在 Windows NT “服务”控制面板中检查 DB2 实例的 LOG ON AS 属性。将使用此权限来访问 Excel 应用程序。确认此用户是否具有适当的权限，或者将此属性更改为已授权的帐户，然后重新启动 DB2 并再次运行 SQL 查询。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1706.<internal program code>”。相关联的文本和标记为“打开源电子表格时出错”	在打开由 SQL 查询中的昵称所引用的电子表格时产生了问题。确保文件存在于在注册期间在 CREATE NICKNAME 语句中指定的 PATH 中。

表 65. 由 Excel 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1707.<internal program code>”。相关联的文本和标记为“存取 DL 输出存储区时出错”	发生了内部程序错误。请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1708.<internal program code>”。相关联的文本和标记为“Excel 应用程序结束失败”	发生了内部程序错误。如果在重复查询之后此错误仍然存在，请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1711.<internal program code>”。相关联的文本和标记为“在访存期间出错，可能数据 / 列类型不匹配”	在 SQL 查询期间访存的数据的数据类型与在注册昵称期间指定的数据类型不相同。更正源电子表格中的数据，或者更正昵称中已注册的数据类型。如果此操作不能更正该问题，请与 IBM 软件支持机构联系。
SQL1822N	从数据源“Excel 包装器”中接收到意外的错误代码“-1900.<internal program code>”。相关联的文本和标记为“内存分配错误”	发生了内部程序错误。请与 IBM 软件支持机构联系。

相关概念:

- 『消息介绍』（《消息参考》第 1 卷）

相关参考:

- 『SQLSTATE 消息』（《消息参考》第 2 卷）



---

## 第 13 章 配置对 Extended Search 数据源的存取

本章说明如何配置联合服务器以存取存储在 IBM Lotus Extended Search 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Extended Search 数据源的存取。

本章包含下列内容:

- 说明什么是 Extended Search
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 Extended Search 包装器相关联的错误消息

---

### 什么是 Extended Search?

Extended Search 产品是一个多层的客户机 / 服务器系统, 它提供了大范围的搜索和检索功能。利用 Extended Search, 可以输入单个请求来同时搜索潜在的成千上万的数据资源库和因特网。这些资源库可能具有不同的内容和结构, 而在地理位置上可能遍布全世界。

Extended Search 支持通过单个存取点来对结构化和非结构化数据进行分布式全异搜索。它利用当前的数据管理资源, 并彻底地处理同时存取各种各样的源所需要的各种事项。

Extended Search 使用它的“通用查询语言”(GQL)作为通用搜索语法, 并在内部将每个搜索请求转换为想要搜索的数据源的本机搜索语言。它还使用对于那些源来说属于本机的方法来查找和检索信息, 而不管源所在的位置。

有关安装 Extended Search 服务器、配置搜索域和使用 GQL 的信息, 请参阅 Extended Search 产品文档。在 IBM® Lotus® Extended Search Web 站点的 Resources 页上可以找到下列文档:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

#### *Extended Search General Information*

描述了 Extended Search 系统中的组件以及它们相互之间如何交互作用, 还描述了后端数据系统。

#### *Extended Search Installation*

定义了系统先决条件, 并提供了有关安装该产品和验证安装过程的指示信息。

#### *Extended Search Administration*

提供了有关将数据源添加至搜索域、配置可搜索的字段以及使用样本搜索应用程序来查询 Extended Search 源的指示信息。

#### *Extended Search Programming*

讨论了可以用来将搜索支持扩展到在该产品的缺省配置中不受支持的数据源的应用程序开发工具。还包括了对 Extended Search 通用查询语言的描述。

## Extended Search 数据源

借助 Extended Search，可以搜索下列类型的数据源：

- 许多受欢迎的 Web 搜索站点和新闻站点。如果需要搜索您的内部网的搜索站点或者其它内部或外部搜索站点，可以很容易添加对执行这些操作的支持。
- 邮件系统，例如，您使用 Lotus Notes<sup>®</sup> 和 Microsoft<sup>®</sup> Exchange Server 管理的那些邮件系统。
- 文档管理系统，例如，DB2<sup>®</sup> Information Integrator for Content 数据库。
- 关系数据库，例如，IBM DB2、Oracle、Microsoft SQL Server、Microsoft Access 和其它符合“开放式数据库连接”（ODBC）标准的数据库。
- 全文索引，例如，使用 IBM WebSphere<sup>®</sup> Portal、Domino<sup>™</sup> Domain Index、Microsoft Index Server 和 Microsoft Site Server 创建的那些全文索引。
- Lotus 资源库，包括 Notes 数据库、Domino.Doc<sup>®</sup> 库和文件柜、Lotus QuickPlace 位置以及 Lotus Discovery Server<sup>™</sup> 知识图（K 图）。
- 即时消息传递系统，例如，Lotus Sametime<sup>®</sup>。此功能部件使您能够将查询提交给知识丰富的人员，而不只是可搜索的数据资源库。
- 轻量级目录访问协议（LDAP）目录，例如，您使用 IBM SecureWay<sup>®</sup>、Domino LDAP Server 和 Exchange LDAP Server 管理的那些目录。
- 文件系统，可以搜索存储在本地或者存储在网络驱动器上的文本文件。不能搜索已压缩或已加密的文件。

借助 Extended Search C++ 和 Java<sup>™</sup> 应用程序编程接口（API），可以将支持扩展到对其它类型的源（例如，此处未提到的专有数据库）的支持。

## Extended Search 包装器的工作方式

在结构化关系数据库模型中，各列是按一致格式命名和表示的。此功能允许您执行精确的计算操作和通过比较特定列值来连接来自不同表的数据。还可以执行其它类型的分析，例如，列示一个表中有而另一个表中没有的对象。

相反，非结构化数据通常是以自由文本格式存储的。通常，很少有或者没有使您能够按列名查询信息的元数据。搜索非结构化数据更多的是取决于查找与用户指定的关键字相匹配的数据，而不是取决于计算条件。

Extended Search 包装器将这两种搜索技术结合起来了。借助该包装器，可以使用结构化查询语言来搜索 Extended Search 域中的非结构化内容。然后可以对搜索结果执行分析操作或关系操作。

通过输入引用特殊用途的 DB2 表（昵称表）的 SQL 语句来发出查询。Extended Search 根据 SQL 条件来执行搜索，并用结果数据填充昵称表。因为搜索结果保留在表中，所以，数据可用于对其它数据库表（包括其它昵称表）的操作。

当使用包装器提交搜索请求时，可以从已映射至昵称表的任何 Extended Search 源中检索数据。可以将此数据与联合系统中的其它数据源集成到一起，而不需要将数据移出本机数据源。无论有多少个源对查询提供了响应，搜索结果都将作为单个结果集出现。

下图说明了 Extended Search 包装器如何将 Extended Search 域中的各种数据源连接到联合数据库系统。该包装器从一个或多个远程 Extended Search 服务器中存取和检索数

据。如果包装器与连接至其它 Extended Search 服务器的一个 Extended Search 服务器进行联系，则可能会从多个服务器中返回搜索结果。

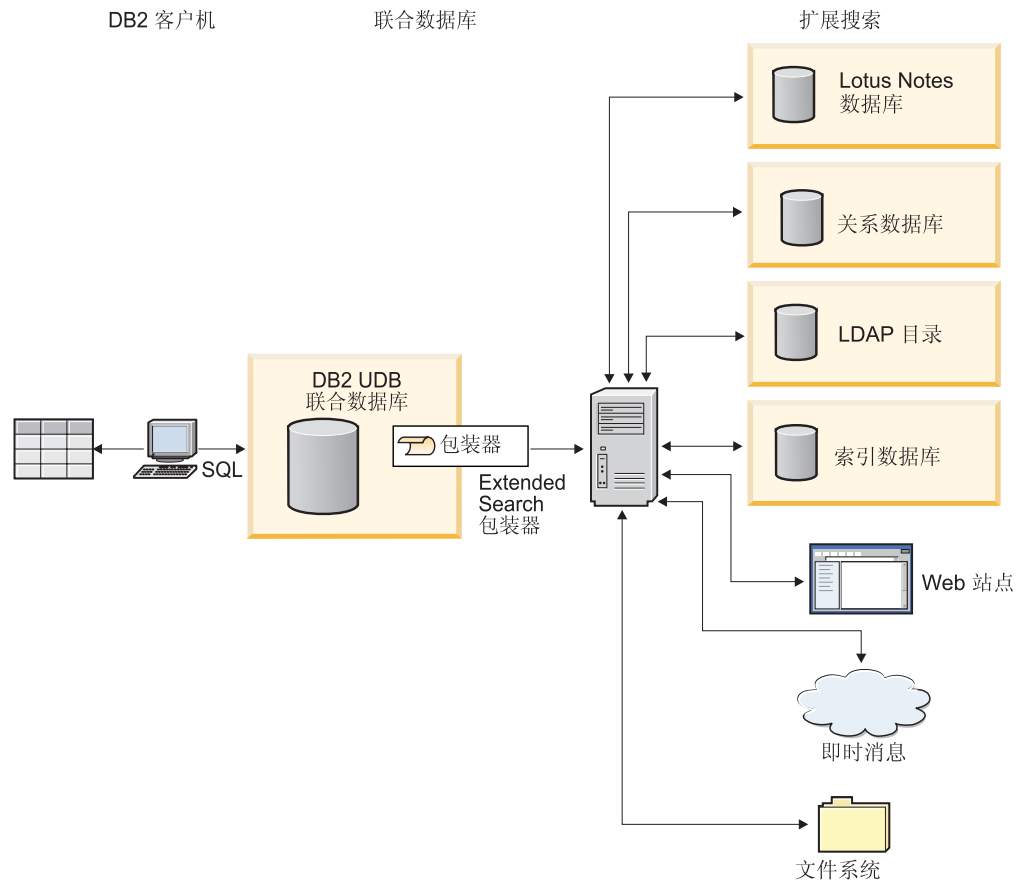


图 24. Extended Search 包装器的工作方式

相关任务:

- 第 218 页的『将 Extended Search 数据源添加至联合服务器』

## Extended Search 昵称

在 Extended Search 数据模型中，一个或多个字段组成一个文档。多个文档的集合组成数据源。可以将任意数目的数据源组合成一个类别，这样就使您能够将它们作为一个组来进行搜索和管理。

为了确保用户只存取他们需要的数据源，一个类别必须至少属于一个应用程序。可以将应用程序看作是为了控制存取和搜索功能而将用户分组的一种方法。例如，个人应用程序可以与金融应用程序包括相同的数据源，但是，每个应用程序的用户并不一定需要存取那些数据源中的相同字段。

当注册昵称时，标识想要搜索的应用程序、类别、数据源和数据源字段。Extended Search 配置数据库中必须存在这些实体。要使用 Extended Search 包装器来搜索 Extended Search 数据源，必须为该数据源创建昵称。

昵称表的内容将反映您注册昵称时 Extended Search 配置数据库的状态。如果 Extended Search 管理员更新了配置（例如，通过添加或删除源或字段），则昵称表中将不会反映那些更改。如果昵称表引用了已更改的数据，而您想与 Extended Search 配置数据库保持同步，则必须改变昵称，或者删除它然后创建新昵称。

如果您不改变或重新创建昵称，则当您试图搜索 Extended Search 域中不再存在的搜索项时，可能会接收到返回的错误和无结果报告。

尽管单个昵称表中可以包含有关在 Extended Search 中配置的所有源的信息，但是，创建几个昵称表可能会更有用。要充分使用 DB2® 的功能，为您计划使用 Extended Search 包装器搜索的每种类型的数据源创建独立的昵称。

例如，可以将一个昵称用于 Web 源，一个昵称用于 Notes 数据库，另一个昵称用于文件系统等等。如果具有独立的昵称表，您就能够对返回给包装器的数据更好地执行连接，根据字段值使各种不同的源相关联，并将结果数据与其它数据一起集成到联合系统中。

**相关概念:**

- 第 216 页的『Extended Search 垂直表』

**相关任务:**

- 第 221 页的『为 Extended Search 数据源注册昵称』

**相关参考:**

- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 507 页的『CREATE NICKNAME 语句语法 - Extended Search 包装器』

---

## Extended Search 垂直表

Extended Search 应用程序可以由许多类别组成，每个类别又可以包含许多数据源。由于每个数据源对字段名都使用它自己的约定，所以，字段的交集可能会导致空集。当将数据源字段映射至昵称表中用户定义的列并以水平表的形式来显示搜索结果时，该表中可能包含难以管理的列数。如果许多行中都只有少数列才具有数据，则该表看起来将是稀疏地填充了一些数据。例如：

Column_1	Column_2	Column_3
Value_11		
	Value_22	
Value_31		Value_33

在 Extended Search 中，可以通过定义已映射的字段来控制如何显示结果。通过已映射的字段可以将不同源中命名不同但是具有共同用途的内容组合起来。例如，可以创建一个名为 EmployeeNumber 的已映射字段来表示来自不同源中名为 EmpNum、EmpNo 和 EmpID 的字段的结果数据。如果没有此映射功能，则需要为每个唯一的字段名定义一个昵称列，这与已映射字段是映射至单个列不同。

当您知道需要使之相关的字段的名称时，映射字段就很有用。但是，某些应用程序需要使来自许多数据源的大量字段相关。特别是对于非结构化数据，可能预先并不知道



各个字段之间的关系。因此，很难定义和构造有意义的昵称表。为了支持这种类型的应用程序，Extended Search 包装器允许您创建垂直昵称表。

当为 Extended Search 创建昵称表时，可以启用 VERTICAL\_TABLE 选项。此选项将返回在数据源中配置为可返回的所有字段，如 Extended Search 配置数据库中定义的那样。当您不能确定哪些列将在搜索中相关，或者当您对结果集执行后处理查询或连接时不能确定哪些列将相关，则使用此选项。

垂直表中的每一行都包含有关在结果集中返回的字段的信息。对于每一行，Extended Search 都将返回一个字段所来源的名称、字段名、该字段的值及其数据类型（date 和 integer 等等）。与散布在水平表的各列中的结果不同，垂直表有较高的数据填充密度，并且包含很多行数据。例如：

Field_Name	Field_Value	Field_Datatype
Column_1	Value_11	VARCHAR
Column_2	Value_22	DATE
Column_1	Value_31	VARCHAR
Column_3	Value_33	VARCHAR

当查询表时，可以对此数据执行 SQL 操作，并且可以查询所有列标号。例如：

```
Field_Value LIKE '%IBM%'
```

因为 VERTICAL\_TABLE 选项将返回有关数据源中所有可返回字段的信息，所以，可能不需要查询用户定义的特定列。如果启用此选项，然后发出 SELECT 语句来搜索用户定义的列，则在搜索结果中可能会接收到重复信息。但是，如果您定义用户定义的列，则可以将那些列与联合系统中的其它表一起使用。

下面两个表总结了 Extended Search 为垂直昵称表中的每一行返回的由系统提供的各个列。

对于每个昵称，包装器始终都将返回下面三个固定列。

列名	数据类型	描述
DOC_ID	VARCHAR(512)	文档标识，对于一组搜索结果中的每一项都是唯一的。
DOC_RANK	INTEGER	文档的相关程度。
CLIENT_LOCALE	VARCHAR(5)	搜索请求的客户机语言环境。如果 SQL 查询未提供客户机语言环境，则查询将使用 enUS 作为缺省客户机语言环境。

仅当启用了 VERTICAL\_TABLE 选项时，包装器才会创建下列固定列。

列名	数据类型	描述
DATASOURCE_NAME	VARCHAR(128)	产生搜索结果的数据源的名称。
FIELD_NAME	VARCHAR(128)	搜索结果中返回的字段名称。

列名	数据类型	描述
FIELD_VALUE	VARCHAR(4096)	结果集中返回的字段值。如果字段值长于昵称列的最大长度（VARCHAR 值），则字段值将被截断。如果在一列的末尾具有 ES_TRUNCATE 标记，则表示该值是不完整的。
FIELD_DATATYPE	SMALLINT	表示字段值的实际数据类型的一个整数值。  384 DATE 448 VARCHAR 496 INTEGER

将结果数据作为 VARCHAR 值来存储的垂直表可能很难查询。要进行更加精确地搜索，可在 Extended Search 配置数据库中创建已映射的字段，然后在昵称表中定义它们。通过已映射的字段，可以创建搜索结果的简明水平表。还可以优化您对结果执行关系操作的能力，并在涉及到联合数据库系统中的其它表的查询中组合它们。

有关在 Extended Search 中定义已映射字段的信息，请参阅 *Extended Search Administration*，在 IBM® Lotus® Extended Search Web 站点的 Resources 页上可以找到它：

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

**相关概念:**

- 第 215 页的『Extended Search 昵称』

**相关任务:**

- 第 221 页的『为 Extended Search 数据源注册昵称』

**相关参考:**

- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 507 页的『CREATE NICKNAME 语句语法 - Extended Search 包装器』

---

## 将 Extended Search 添加至联合服务器

### 将 Extended Search 数据源添加至联合服务器

要配置联合服务器以存取 Extended Search 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

| 可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Extended Search  
| 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所  
| 需要执行的步骤。

**先决条件:**

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上
- 在使用 Extended Search 包装器之前，确保在 Extended Search 配置数据库中配置了您计划搜索的源。通过 Extended Search 客户机提交一些查询，以便在尝试使用 Extended Search 包装器搜索一些源之前验证您搜索源的能力。

### 过程:

要将 Extended Search 数据源添加至联合服务器:

1. 注册包装器。
2. 注册服务器定义。
3. 可选: 创建用户映射。
4. 为 Extended Search 数据源注册昵称。
5. 可选: 为 Extended Search 包装器注册定制函数。

### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 219 页的『注册 Extended Search 包装器』
- 第 220 页的『为 Extended Search 数据源注册服务器』
- 第 221 页的『为 Extended Search 数据源注册昵称』
- 第 221 页的『为 Extended Search 数据源注册用户映射』
- 第 222 页的『为 Extended Search 包装器注册定制函数』

### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 注册 Extended Search 包装器

注册 Extended Search 包装器是将 Extended Search 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Extended Search 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

### 过程:

要注册包装器, 使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如, 要在使用 Windows 操作系统的联合服务器上注册一个名称为 NotesDBwrapper 的包装器, 请发出以下语句:

```
CREATE WRAPPER NotesDBwrapper LIBRARY 'db2uies.dll'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称, 请参阅 Extended Search 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Extended Search 包装器注册服务器定义。

### 相关任务:

- 第 220 页的『为 Extended Search 数据源注册服务器』

**相关参考:**

- 第 220 页的『Extended Search 包装器库文件』
- 第 505 页的『CREATE WRAPPER 语句语法 - Extended Search 包装器』

## Extended Search 包装器库文件

下表列示了 Extended Search 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2uies.a、libdb2uiesF.a 和 libdb2uiesU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 66. *Extended Search* 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2uies.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2uies.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2uies.so
Windows	%DB2PATH%\bin	db2uies.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

**相关任务:**

- 第 219 页的『注册 Extended Search 包装器』

## 为 Extended Search 数据源注册服务器

此任务是将 Extended Search 数据源添加至联合系统这一主要任务的一部分。在注册包装器之后，必须创建相应的服务器定义来标识正在与联合系统进行集成的远程 Extended Search 服务器。此定义使包装器能够连接至 Extended Search 服务器。

**过程:**

要注册 Extended Search 服务器，从 DB2 “命令行处理器”中发出 CREATE SERVER 语句。

例如，要为一个称为 myESwrapper 的包装器注册一个称为 es1 的服务器，发出以下语句。Extended Search 服务器使用缺省端口值。

```
CREATE SERVER es1 WRAPPER myESwrapper OPTIONS (ES_HOST 'my.server.com')
```

要创建此同一服务器，对所有消息级别（关键、非关键、警告和信息）启用跟踪，并将跟踪消息写入 wrapper 目录中称为 es1wrapper.log 的文件，发出以下语句：

```
CREATE SERVER es1 WRAPPER myESwrapper OPTIONS (ES_HOST 'my.server.com',
ES_TRACING 'ON', ES_TRACELEVEL 'CNWI',
ESTRACEFILENAME '/wrapper/es1wrapper.log')
```

在这一系列任务中的下一个任务是为 Extended Search 数据源注册昵称。

**相关任务:**

- 第 221 页的『为 Extended Search 数据源注册昵称』

**相关参考:**

- 第 506 页的『CREATE SERVER 语句语法 - Extended Search 包装器』

## 为 Extended Search 数据源注册用户映射

此任务是将 Extended Search 数据源添加至联合系统这一主要任务的一个可选步骤。

用户映射可以用来认证使用 Extended Search 包装器查询 Extended Search 源的用户的存取权。如果用户对已注册的 Extended Search 昵称提交 SQL 查询，但是没有为该用户定义任何用户映射，则 Extended Search 包装器将使用缺省用户标识和密码来尝试从远程 Extended Search 服务器中检索数据。如果正在被查询的数据源要求认证，则可能会返回空的结果集。

为了确保将正确的用户标识和密码传递给 Extended Search 服务器，在联合系统中为已授权可搜索 Extended Search 源的用户创建用户映射。当创建用户映射时，密码将以加密格式存储在 DB2 目录表中。当通过 Extended Search 将密码从 DB2 传递给正在被搜索的源时，密码将保持安全格式。

Extended Search 配置数据库中的安全性设置将确定是否已授权给该用户标识和密码以存取正在被搜索的源，以及是否将执行该用户标识的任何其它映射。

**过程:**

要注册 Extended Search 用户映射，从 DB2 “命令行处理器”中发出 CREATE USER MAPPING 语句。

该语句必须标识需要映射的 DB2 用户标识、托管目标数据源的 Extended Search 服务器以及使用户能够存取那些数据源的用户标识和密码。

例如，以下语句将注册用户标识 user1，以便它可以使用 Extended Search 服务器 es1 来搜索远程数据库。

```
CREATE USER MAPPING FOR user1 SERVER es1 OPTIONS  
(REMOTE_AUTHID 'ESUserId', REMOTE_PASSWORD 'abc123def')
```

在这一系列任务中的下一个任务是注册 Extended Search 定制函数模板。

**相关任务:**

- 第 222 页的『为 Extended Search 包装器注册定制函数』

**相关参考:**

- 第 507 页的『CREATE USER MAPPING 语句语法 - Extended Search 包装器』

## 为 Extended Search 数据源注册昵称

此任务是将 Extended Search 数据源添加至联合系统这一主要任务的一部分。在注册服务器之后，必须至少注册一个昵称。昵称表是一个虚拟的 DB2 表，它标识 Extended Search 域中的一个或多个可搜索的源。当提交查询时，为想要搜索的源指定昵称。

### 先决条件:

确保正在为其创建昵称的 Extended Search 服务器正在运行。当创建昵称时，系统将验证 Extended Search 配置数据库中是否存在有关您计划搜索的各个源和字段的信息。

### 过程:

要注册 Extended Search 昵称，从 DB2 “命令行处理器” 中发出 CREATE NICKNAME 语句。

例如，发出以下语句来创建一个昵称表，用于搜索由 Extended Search 服务器 es1 托管的 Demo 应用程序中属于 Web 类别的所有数据源。该语句返回 WebTitle 和 WebDescription 字段，并使用缺省搜索处理选项。

```
CREATE NICKNAME allweb (WebTitle VARCHAR(255), WebDescription VARCHAR(1000))
  FOR SERVER es1 OPTIONS(APPLICATIONID 'Demo', CATEGORY 'Web')
```

发出以下语句来创建一个昵称表，用于搜索 Science 应用程序中的几个数据源。该语句以垂直的列列表的形式提供搜索结果、将超时值设置为 60 秒、允许每个源最多可返回 100 个结果文档、将结果集大小扩展到 1000 个条目并按作者名将结果排序。

```
CREATE NICKNAME stars (Title VARCHAR(80), Author VARCHAR(40),
  Abstract VARCHAR(200))
  FOR SERVER es1 OPTIONS (APPLICATIONID 'Science',
  DATASOURCES 'Astronomy;NASA Library;Astrophysics', VERTICAL_TABLE 'yes',
  TIMEOUT '60', MAXHITS '100', TOTALMAXHITS '1000', SORTFIELD 'Author')
```

在这一系列任务中的下一个任务是为 Extended Search 包装器注册用户映射。

### 相关概念:

- 第 215 页的『Extended Search 昵称』
- 第 216 页的『Extended Search 垂直表』

### 相关任务:

- 第 218 页的『将 Extended Search 数据源添加至联合服务器』
- 第 221 页的『为 Extended Search 数据源注册用户映射』
- 第 58 页的『为非关系数据源指定昵称列』

### 相关参考:

- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 507 页的『CREATE NICKNAME 语句语法 - Extended Search 包装器』

## 为 Extended Search 包装器注册定制函数

此任务是将 Extended Search 数据源添加至联合系统这一主要任务的一个可选步骤。

定制函数中不包含任何可执行代码。在注册函数之后，可以在查询中引用它来改变缺省搜索行为。Extended Search 包装器的定制函数使您能够指定未定义为昵称表中的一列的精确搜索表达式和搜索内容。

### 限制:

- 只能使用 WHERE 子句来调用 Extended Search 函数。

- WHERE 子句必须至少包含一个充当搜索谓词的谓词，它或者是 Extended Search 函数，或者是类型为“列名运算符常量”的谓词。
- Extended Search 函数是一个标量函数模板。它必须使用 EQUAL (=) 运算符，且比较值必须为一 (1)。
- Extended Search 函数中的第一个参数充当一个锚值，用来标识函数所适用的昵称，例如，搜索结果中文档的排列 (DOC\_RANK)。必须为此参数指定一个 INTEGER 字段。如果 SQL 查询中包含多个昵称或者是昵称和表的组合，则此参数特别重要（不会对此参数求值）。例如：

```
SELECT * FROM es_nickname1, es_nickname2
WHERE eswrapper.es_search(es_nickname1.DOC_RANK, 'IBM') = 1 AND
      eswrapper.es_search(es_nickname2.DOC_RANK, 'IBM') = 1
```

过程:

要注册 Extended Search 定制函数，发出以下 CREATE FUNCTION 语句:

```
CREATE FUNCTION eswrapper.es_search(integer, varchar(1024))
  RETURNS INTEGER AS TEMPLATE
  DETERMINISTIC NO EXTERNAL ACTION;
```

相关参考:

- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 227 页的『Extended Search 包装器 - 通用查询语言』
- 第 510 页的『CREATE FUNCTION 语句语法 - Extended Search 包装器』
- 第 223 页的『Extended Search 包装器 - 查询指南』

## 查询 Extended Search 数据源

### Extended Search 包装器 - 查询指南

Extended Search 包装器期望查询采用特定格式，并且不支持不满足精确的语言条件的查询。本主题提供了有关创建查询的指南，并提供了正确和不正确查询语法的示例。

#### 查询多种语言的 Web 源

Extended Search 用来链接至 Web 源的第三方软件支持使用 ISO-8859-1 代码页的语言（例如，英语、法语、德语、葡萄牙语和瑞典语）。因此，当您搜索 Web 源时，将不能搜索双字节字符集语言（例如，韩国语）、双向语言（例如，希伯来语）或者其它非 ISO-8859-1 语言。当处理搜索结果的解析器检测到它认为是非法字符代码的内容时就会失败。

#### 指定 CLIENT\_LOCALE 值

如果在 WHERE 子句中包括 CLIENT\_LOCALE 列来设置客户机语言环境的值，则必须使用 AND 谓词来指定搜索条件。不能将 OR 谓词与 CLIENT\_LOCALE 列配合使用。

示例 - 正确语法

下列示例说明了将 CLIENT\_LOCALE 列包括在 WHERE 子句中的正确方法:

```
WHERE CLIENT_LOCALE = 'enUS' AND  
ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1
```

```
WHERE ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1  
AND CLIENT_LOCALE = 'enUS'
```

示例 - 不正确的语法

下列示例是不正确的，原因是它们试图将 OR 谓词与 CLIENT\_LOCALE 列配合使用：

```
WHERE CLIENT_LOCALE = 'enUS' OR  
ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1
```

```
WHERE ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1  
OR CLIENT_LOCALE = 'enUS'
```

### 在 Extended Search 固定列上指定谓词

包含 Extended Search 昵称的 SQL 语句必须在 WHERE 子句中为昵称指定谓词。但是，Extended Search 固定列上的谓词并不算作谓词。

示例 - 不正确的语法

以下示例显示了一个不正确的查询，原因是它不包含谓词：

```
SELECT * FROM ES_NICKNAME
```

以下示例显示了一个不正确的查询，原因是唯一的谓词在一个固定列上：

```
SELECT * FROM ES_NICKNAME WHERE DOC_RANK < 20
```

### 指定未绑定的谓词

仅当谓词值为常量时，用户定义的列上的谓词才会由 Extended Search 包装器来处理。如果未绑定谓词值，则谓词将由 DB2 引擎来处理。如果未绑定的谓词是 SQL 语句中唯一的谓词，则将导致错误。Extended Search 昵称需要可以由 Extended Search 包装器处理的谓词。

示例 - 正确语法

以下示例中的 WHERE 语句说明了一个将由 Extended Search 包装器来处理的谓词：

```
SELECT *  
FROM ES_NICKNAME  
WHERE Author = 'Ernest Hemingway'
```

示例 - 不正确的语法

以下示例中的 WHERE 语句说明了一个将由 DB2 来处理的谓词：

```
SELECT *  
FROM ES_NICKNAME_1, ES_NICKNAME_2  
WHERE ES_NICKNAME_1.Author = ES_NICKNAME_2.Author
```

### 使用 OR 谓词来连接查询

Extended Search 包装器不能搜索不同的昵称表，也不能搜索使用简单的 OR 谓词连接的昵称表和数据库表。只能在同一昵称中使用 OR 谓词。

示例 - 不正确的语法



```

SELECT *
FROM   ES_Nickname as N1, TABLE as T1
WHERE  N1.Column1 = 'abc' OR T1.Column1 = 'abc'

SELECT *
FROM   ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE  N1.USerdefCol = 'abc' OR N2.USerdefCol = 'cdf'

SELECT *
FROM   ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE  ESWRAPPER.ES_SEARCH(N1.DOC_RANK, 'IBM')=1 OR
       ESWRAPPER.ES_SEARCH(N2.DOC_RANK, 'LOTUS')=1

```

#### 相关任务:

- 第 222 页的『为 Extended Search 包装器注册定制函数』

#### 相关参考:

- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 227 页的『Extended Search 包装器 - 通用查询语言』

## Extended Search 包装器 - 示例查询

要使用 Extended Search 包装器来运行查询，按照指定典型的 DB2 表名和表列的方式在 SQL 语句中指定已注册的昵称和昵称列。

在此样本搜索方案中，医院工作人员需要搜索和比较最新的医学研究成果。为了搜索各种各样的源，医院将使用 Extended Search 服务器。Extended Search 域包括名为 MedResearch 的应用程序以及为了搜索基于文档的数据库、邮件服务器和 Web 而配置的几个类别。

除了进行搜索之外，工作人员还需要将结果与各种搜索进行比较。例如，他们需要确定谁在一定时间段内发表了文章、谁最近采购了草药和维生素、通过电子邮件与同事讨论替代药品，并应用于更新药品许可证。Extended Search 包装器通过它的能够将非结构化 Extended Search 数据集成到 DB2 中以进行结构化检索的能力来提供了解决方案。

医院工作人员决定创建下面三个昵称，一个昵称用于搜索文档资源库，一个昵称用于搜索电子邮件系统，另一个昵称用于搜索特定的 Web 源。在 Extended Search 配置数据库中，Owner 和 Date 字段被定义为已映射的字段，这使您能够在连接中使用这些字段，而不管它们在本机数据源中是如何命名的。

#### 文档昵称:

```

CREATE NICKNAME MedDocs ( Owner   VARCHAR(80),
                          Date    DATE,
                          Title   VARCHAR(80),
                          Abstract VARCHAR(200) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              CATEGORY 'AMA Library;Medical Records;Pharmacy',
                              VERTICAL_TABLE 'YES',
                              TIMEOUT '60', MAXHITS '100',
                              TOTALMAXHITS '1000' )

```

#### 电子邮件昵称:

```

CREATE NICKNAME MedMail ( Owner   VARCHAR(80),
                           To     VARCHAR(80),
                           Date   DATE,
                           Subject VARCHAR(80) )

```

```
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                             CATEGORY 'Exchange Server;Lotus Notes',
                             VERTICAL_TABLE 'YES', )
                             TIMEOUT '60', MAXHITS '100',
                             TOTALMAXHITS '1000' )
```

Web 昵称:

```
CREATE NICKNAME MedWeb ( WebTitle VARCHAR(255),
                        WebDescription VARCHAR(1000) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                             DATASOURCES 'Google!;Alta Vista;CNN',
                             TOTALMAXHITS '500' )
```

以下查询将搜索标题中包含短语 Artificial Liver 且文档内容中包含缩写 MARS 的文档。结果集应排除在 2001 年之前发布的任何文档。

```
SELECT OWNER, DOC_CONTENT
FROM MedDocs
WHERE ESWRAPPER.ES_Search(DOC_RANK, '( ( TOKEN:EXACT "MARS") AND
                                     ("TITLE" IN "Artificial Liver") AND
                                     ("DATE" >= "01/01/2001") ) ) = 1
```

以下查询将搜索以前几个月编写的讨论替代药品的电子邮件:

```
SELECT *
FROM MedMail
WHERE ESWRAPPER.ES_Search(DOC_RANK, '(
                             ("SUBJECT" IN "alternative medicine") AND
                             ("DATE" BETWEEN "03/01/2002" AND
                             "09/30/2002") ) ) = 1
```

以下查询将搜索讨论了辅助替代药品 (CAM) 疗法以及美国民众对它的接受程度的 Web 源:

```
SELECT WebTitle, WebDescription
FROM MedWeb
WHERE ESWRAPPER.ES_Search(DOC_RANK, '(
                             TOKEN:EXACT "CAM therapy" ) AND
                             ( TOKEN:FUZZY "United States" ) ) = 1
```

以下查询将搜索从医院药房购买了大量草药或维他命并且是最近才领取执照的医生。然后, 该查询将那些医生的姓名与编写了有关替代药品的电子邮件的人员对应起来。

```
SELECT N2.OWNER, N2.DATE
FROM MedDocs as N1,
MedMail as N2
WHERE ESWRAPPER.ES_SEARCH(N1.DOC_RANK, '(
                                     ("LICENSE_DATE" >= "01/01/2002") AND
                                     ( ( ( "PRODUCT" = "HERB") OR ("PRODUCT" = "VITAMIN") ) AND
                                     ("QUANTITY" > "1000") ) ) ) = 1
AND ESWRAPPER.ES_SEARCH(N2.DOC_RANK, '( "SUBJECT" IN
                                     "alternative medicine" ) ) = 1
AND N1.OWNER = N2.OWNER
```

相关概念:

- 第 215 页的『Extended Search 昵称』
- 第 216 页的『Extended Search 垂直表』

相关任务:

- 第 221 页的『为 Extended Search 数据源注册昵称』
- 第 222 页的『为 Extended Search 包装器注册定制函数』

### 相关参考:

- 第 227 页的『Extended Search 包装器 - 通用查询语言』
- 第 510 页的『CREATE FUNCTION 语句语法 - Extended Search 包装器』
- 第 507 页的『CREATE NICKNAME 语句语法 - Extended Search 包装器』
- 第 223 页的『Extended Search 包装器 - 查询指南』

## Extended Search 包装器 - 通用查询语言

您通过 Extended Search 包装器传递给 Extended Search 服务器的查询可以包含用通用查询语言 (GQL) (它是 Extended Search 的查询语言) 编写的搜索表达式。

例如, 假定用户想在包含具有职员信息的表的关系数据库中查找其姓名以 JO 开头的所有职员。则可以发出用 GQL 编写的以下查询:

```
(LIKE "EMPLOYEE_NAME" "JO")
```

可以按如下所示发出用 SQL 编写的相同查询:

```
SELECT * FROM EMP.TABLE WHERE EMPLOYEE_NAME LIKE JO%
```

与 SQL 相似, 包装器支持中缀表示法, 这是一种要求运算符位于字段名与比较值之间的语法。本地 Extended Search GQL 语法使用前缀表示法, 这是一种要求运算符在想要对其进行求值的字段和值的前面的语法。

比较下列将搜索在 TITLE 字段中包含单词 IBM 的文档的查询表达式:

### 中缀 GQL

```
("TITLE" IN "IBM")
```

### 前缀 GQL

```
(IN "TITLE" "IBM")
```

当使用 Extended Search 包装器提交查询时, API 将把中缀 SQL 语句转换为前缀 GQL, 以便供 Extended Search 处理。

以下语法描述说明了在查询中可以使用的 Extended Search 语法的“巴科斯范式”规范。

```
expr:          pattern_expr
              | bool_expr
              | field_expr
              | prox_expr

pattern_expr:  STRING
              | token_expr

token_expr:   ( TOKEN [:CASE] [:STEM] [:EXACT] [:WEIGHT "x"]
              | [:WILD] [:FUZZY] STRING )

bool_expr:    (expr_list bool_operator [:WEIGHT "x"] expr )

bool_text_expr: (text_expr_list bool_operator [:WEIGHT "x"] text_expr )

text_expr:    pattern_expr
              | bool_text_expr
              | prox_expr

text_expr_list: text_expr
               | text_expr_list text_expr
```

```

expr_list:      expr
                | expr_list expr

field_expr:    ( field_name operator_1 [:WEIGHT "x"] text_expr )
                | ( field_name operator_2 [:WEIGHT "x"] value )
                | ( field_name operator_3 [:WEIGHT "x"] value_1 AND value_2 )
                | ( field_name operator_4 value )

prox_expr:    ( prox_op [:COUNT "x"][:ORDER][:MATH "y"][:WEIGHT "x"]
                expr_list expr )

prox_op:      DOCUMENT
                | PARAGRAPH
                | SENTENCE
                | WORD
                | CHARACTER

operator1:    START
                | END
                | IN
                | =

operator_2:   =
                | >
                | >=
                | <
                | <=
                | EQ
                | GT
                | GTE
                | LT
                | LTE

operator_3:   BETWEENI
                | BETWEENE
                | LIKE

bool_operator: AND
                | OR
                | NOT

```

有关 GQL 语法的完整信息，请参阅 *Extended Search Programming*（可以在 IBM Lotus Extended Search Web 站点的 Resources 页上找到它）：

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

#### 相关任务:

- 第 222 页的『为 Extended Search 包装器注册定制函数』

#### 相关参考:

- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 510 页的『CREATE FUNCTION 语句语法 - Extended Search 包装器』
- 第 223 页的『Extended Search 包装器 - 查询指南』

---

## Extended Search 包装器的消息

本主题描述了您在使用 Extended Search 包装器时可能会遇到的消息。

表 67. 由 *Extended Search* 的包装器发出的消息

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: 内部 <i>Extended Search</i> 包装器错误 - 原因码: xxx。)	记录原因码 (它是一个从 901 到 999 的数), 并与 IBM 软件支持机构联系。
SQL0973N	“应用程序”堆中没有足够的存储器用来处理语句。	<p><i>Extended Search</i> 包装器未能在“应用程序”堆中分配内存。要解决此问题, 增大“应用程序”堆大小, 然后再次尝试该语句。例如:</p> <pre>db2 update db cfg for db-name using heap-name heap-size</pre> <p>如果增大此值之后错误仍然存在, 请与 IBM 软件支持机构联系。</p>
SQL1822N	从数据源“ <i>Extended Search</i> 包装器”中接收到意外的错误代码“<error_code>”。相关联的文本和标记为“<tokens>”。	在处理搜索请求时远程 <i>Extended Search</i> 服务器返回了错误。该错误还返回了一个标记, 该标记指示远程服务器上产生错误的原因。如果对 <i>Extended Search</i> 服务器启用了跟踪, 则复查跟踪日志文件以获取诊断帮助。
SQL1823N	对于服务器“<server_name>”中的数据类型“<data_type>”不存在数据类型映射。	CREATE NICKNAME 语句或 ALTER NICKNAME 语句中的列使用了 <i>Extended Search</i> 系统不支持的数据类型。在查询处理期间也可能会发生此错误。如果正在处理查询时产生了该问题, 要解决该问题, 则应删除昵称表, 并创建新昵称。
SQL1825N	在联合环境中不能处理此 SQL 语句。	<i>Extended Search</i> 包装器不能处理当前 SQL 语句。要解决该问题, 请参阅 <i>Extended Search</i> 包装器文档, 根据需要更改 SQL 语句, 然后再次提交该请求。
SQL1833N	不能与端口“<port_number>”上的远程 <i>Extended Search</i> 服务器“<host_name>”建立连接, 或者连接被终止了。	<i>Extended Search</i> 包装器尝试了在指定端口连接至远程 <i>Extended Search</i> 服务器, 但是未能建立连接, 或者被远程服务器终止了。验证远程 <i>Extended Search</i> 服务器的主机名和端口号, 并确保 <i>Extended Search</i> 服务器正在运行, 然后再试。
SQL1834N	用户定义的列“<column_name>”与包装器“<wrapper_name>”的一个固定列完全相同, 但是使用了不同的数据类型。	CREATE NICKNAME 语句或 ALTER NICKNAME 语句包含一个用户定义的列, 该列与指定的 <i>Extended Search</i> 包装器的一个固定列具有相同的名称, 但是使用了不同的数据类型。不需要在 CREATE NICKNAME 语句的列定义中指定固定列。如果指定了固定列, 则应确保固定列名、数据类型和数据类型长度与固定列定义相匹配。不能改变 (ALTER) 固定列名或数据类型。

表 67. 由 *Extended Search* 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1835N	在远程 <i>Extended Search</i> 服务器 “<host_name>” 上找不到类型为 “<object_type>” 的 <i>Extended Search</i> 对象 “<object_name>”。	在指定的远程 <i>Extended Search</i> 服务器上找不到指定的 <i>Extended Search</i> 对象。验证在此 <i>Extended Search</i> 服务器上是否定义了该对象名，且该对象属于指定的对象类型。还要验证此对象的拼写是否正确。
SQL1836N	用户定义的列 “<column_name>” 与远程 <i>Extended Search</i> 服务器 “<host_name>” 上的字段名之间不存在列映射。	DATASOURCE 或 CATEGORY 选项中包括的任何数据源都不包含与指定的用户定义的列名相匹配的字段名。验证该列名是否是 DATASOURCE 选项中的至少其中一个数据源中的字段，或者在属于 CATEGORY 选项中的一个类别的至少其中一个数据源中，然后再次提交该语句。
SQL1837N	不能删除包装器 “<wrapper_name>” 上类型为 “<object_type>” 的必需选项 “<option_name>”。	不能删除必需的选项。更改 ALTER 语句以使用 SET 而不是 DROP。更正搜索语句，然后再次提交该请求。参阅 <i>DB2 SQL Reference</i> ，以获取有关创建有效的 SQL 搜索语句的信息。如果搜索语句中包括 ES_SEARCH 函数，则参阅 <i>Extended Search</i> 包装器文档，以获取有关使用 <i>Extended Search</i> 通用查询语言 (GQL) 的信息。
SQL1838N	搜索语句 “<option_name>” 是无效的 <i>Extended Search</i> 查询。	<i>Extended Search</i> 包装器尝试了处理指定的搜索语句，但是，由于语句未使用正确的查询语法而导致查询失败了。参阅 <i>DB2 SQL Reference</i> ，以获取有关创建有效的 SQL 搜索语句的信息。如果搜索语句中包括 ES_SEARCH 函数，则参阅 <i>Extended Search</i> 包装器文档，以获取有关使用 <i>Extended Search</i> 通用查询语言 (GQL) 的信息。
SQL1839N	一个或多个搜索参数无效。	<i>Extended Search</i> 包装器尝试了使用指定的搜索参数，但是这些参数对于 <i>Extended Search</i> 无效。参阅 <i>Extended Search</i> 包装器文档，更正无效参数，然后再次提交该请求。
SQL1881N	对于 “<object_name>”， “<option_name>” 是无效的 “<option_type>”。	指定的选项对于指定的对象（包装器、服务器、昵称、列或用户映射）无效。请参阅 <i>Extended Search</i> 包装器文档，除去或更改无效选项，然后再次提交该语句。
SQL1882N	不能将 “<option_type>” 选项 “<option_name>” 设置为 “<object_name>” 的 “<option_value>”。	指定的选项值对于指定的对象（包装器、服务器、昵称、列或用户映射）无效。请参阅 <i>Extended Search</i> 包装器文档，更改无效的选项值，然后再次提交该语句。
SQL1883N	对于 “<object_name>”， “<option_name>” 是必需的 “<option_type>”。	要创建、改变或初始化指定的对象（包装器、服务器、昵称或用户映射）的语句中缺少 <i>Extended Search</i> 包装器的必需选项。请参阅 <i>Extended Search</i> 包装器文档，添加必需的选项，然后再次提交该语句。

有关这些消息的更多信息，请参阅《*DB2 消息参考*》。您可能想查看 *Extended Search Administration* 中的 *Extended Search* 产品消息。如果您接收到有关不正确的 GQL 查

询语法的错误，则请参阅 *Extended Search Programming*。在 IBM Lotus Extended Search Web 站点的 Resources 页上提供了 Extended Search 文档：

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

**相关概念：**

- 『消息介绍』（《消息参考》第 1 卷）

**相关参考：**

- 『SQLSTATE 消息』（《消息参考》第 2 卷）
- 『sql0900』（《消息参考》第 2 卷）
- 『sql1800』（《消息参考》第 2 卷）





---

## 第 14 章 配置对 HMMER 数据源的存取

本章说明如何配置联合服务器以存取存储在 HMMER 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 HMMER 数据源的存取。

本章包含下列内容:

- 说明什么是 HMMER
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 HMMER 包装器相关联的错误消息

---

### 什么是 HMMER?

HMMER 是可以用来搜索使用统计模型或概要文件“隐马尔可夫模型”(HMM)的基因序列数据库的一个应用程序包。可以从 <http://hmmer.wustl.edu/> 处免费下载 HMMER 应用程序包。可以在独立的 HMMER 服务器上安装 HMMER 应用程序包,也可以在联合服务器上安装它。

HMM 是一种反映基因序列系列的主结构共有性的统计模型。HMM 基于各种概率模型。如果尚不知道可信的比对,则可以训练 HMM 来识别不一致的基因序列中的模式。训练和使用成功的 HMM 比仔细地构造概要文件需要的技巧和人工干预更少。可以使用经过训练的 HMM 来存取数百个概要文件 HMM 的库,并将它们大量应用于整个基因组或“表达序列标记”(EST)分析。

PFAM (比对和 HMM 蛋白质家族数据库)是各种蛋白质结构域模型的数据库。HMMER 应用程序包与 PFAM 数据库的构造和使用紧密联系在一起。

HMMER 应用程序包包含 9 个程序,但是 DB2<sup>®</sup> Information Integrator 只支持其中的两个程序: hmmpfam 和 hmmsearch 程序。

表 68. HMMER 包装器支持的 HMMER 程序

HMMER 程序	描述
hmmpfam	使用特定基因序列来搜索 HMM 数据库,并确定测试基因序列可能属于的系列。计算每种模型与指定序列和各种模型的数据库相匹配的程度。匹配程度是按照统计显著性来表示的。
hmmsearch	使用特定的 HMM 概要文件来搜索序列数据库,以找到极其相似的序列匹配项。

用户或应用程序对联合服务器发出带有特定于 HMMER 的谓词的 SQL 查询语句。这些语句中的谓词将映射至 hmmpfam 或 hmmsearch 程序中的命令行选项。

HMMER 包装器将查询语句变换成 HMMER 应用程序包可以解释的格式,并启动 hmmpfam 程序或 hmmsearch 程序来运行查询。

在安装 HMMER 应用程序包的服务器上特殊的守护进程在运行。此守护进程接收来自联合服务器的查询请求，并将查询请求发送至 HMMER 应用程序包。HMMER 应用程序包对概要文件数据库（例如，PFAM）运行查询。

图 25 说明了 HMMER 如何使用联合系统。

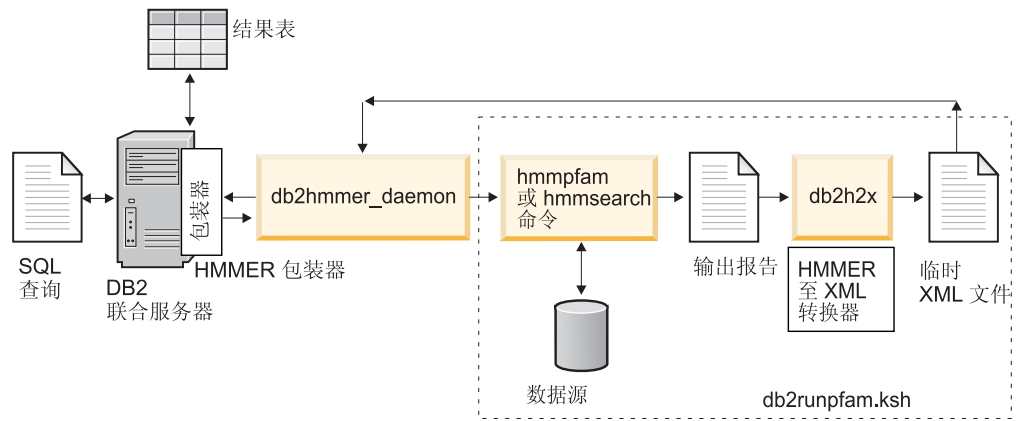


图 25. HMMER 包装器的工作方式

守护进程将结果返回给 HMMER 包装器。包装器将该数据变换成关系表，然后将此表返回给用户或应用程序。

以下示例说明了如何从概要文件数据库中抽取信息（它是由 HMMER 程序构造的）并将信息作为关系表来显示。HMMER User's Guide (<http://hmmmer.wustl.edu/>) 提供了创建概要文件数据库的一些示例以及 HMMER 教程。

第 235 页的图 26 显示了一个使用 7LES\_DROME 基因序列的样本查询。在查询的 WHERE 子句中指定一些序列。

```

SELECT Model, ModelScore, DomainNumber, DomainScore
FROM myhmms
WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTH
INQQAPGTSSSSNSQNASPKIVVRQQSSSFDLRQQLARLGRQLASGQDGHGGISTILINLLLL
ILLSICCDVCRSHNYTVHQSPEPVSKDQMRLLRPKLDSDVVEKVAIWHKHAAPPSIVEGIAISS
RPQSTMAHHPDDRDRDRDPSEEQHGVDERMVLERVTRDCVQRCEIVEDLDFEFGIQCEKADNGEK
CYKTRCTKGCQWYRALKELESCQEACLSLQFYPYDMPICACEMAQRDYWHLQRLAISHLVERTQ
PQLERAPRADGQSTPLTIRWAMHFPEHYLASRPFNIQYQFVDHGHGEELDEQEDQDASGETGSSAW
FNLADYDCDEYMCIELEALIPYTQYRFRFELPFGENRDEVLYSPATPAYQTPPEGAPISAPVIEH
LMGLDDSHLAVHWHPGRFTNGPIEGYRLRLSSSEG NATSEQLVPA GRGSIYFSQLQAGTNYTLALS
MINKQEGEPVAKGFVQTHSARNEKPAKDLTESVLLVGRRAVMWQSLEPAGENSMIYQSQEELADIA
WSKREQLWLLNVHGERLSLKFESGMVSPAQQLKLDLGNISSGRWVPRRSLFDWLHHRLYFAMES
PERNQSSFQIISTDLLGESAQKVGESFDLPVEQLEVDALNGWIFWRNEESLWRQDLHGRMIHRLLR
IRQPQWFLVQPQHFIHMLLPQEGKFLIISYDGGFKHPLPLPPSNGAGNGPASSHWQSFALLGRS
LLLPDSGQLILVEQQQAASPASWPLKNLPDCWAVILLVPESQPLTSAGGKPHSLKALLGAQAAK
TSWKEPERNPYQSADAARSWSYELEVLDVASQSASFIRNIRGPIFGLQRLQPNLYQLRVRAINVD
GEPGEWTEPLAARTWPLGPHRLRWASRQGSVIHTNELGEGLEVQQEQLERLPGPMTMVNESVGYVY
TGDGLLHCINLVHSQWGCPISEPLQHVGSVTYDWRGGRVYWTDLARNCVVRMDPWSGSRELLPVFE
ANFLALDPRQGHLYYATSSQLSRHGSTPDEAVTYRVNGLGEGSIASFVLDLTQQDQLFWLVKGSAL
RLYRAPLTAGGDSLQMIQKIGVFQAVPDSLQLLRPLGALLWLESGRRARLVRLAAPLDMELPT
PDQASPASALQLLDPQLPPRDEGVIPTVLPDSVRLDDGHWDFFHVRWQPSSTGGNHSVSYRLLL
EFGQLQTLDLSTPFARLTQLPQAQLQLKISITPRTAWRSGDTRVQLTTPPVAPSQRRLRVFVE
RLATALQEANVSAVLRWDAPEQGEAPMQALEYHISCWVGSELHEELRLNQSALAEARVEHLQPDQT
YHFQVEARVAATGAAAAGAASHALHVAPEVQAVPRVLYANAEFIGELDLDRNRRLVHTASPV EHL
VGI EGEQRLWVNEHV ELLTHVPGSAPAKLARMRAEVLALAVDWIQRIYVWAE DATAPQAAIIYR
LDLCNFEKGILQGERVWSTPRGRLLKDLVALPQAQSLIWLEYEQGSPRNGSLRGRNLTGSGLEWA
TVQPLIRLHAGSLEPGSETLNLVDNQGKLCVYDVARQLCTASALRAQLNLLGEDSIAGQLAQDSGY
LYAVKNWSIRAYGRRRQLEYTVELEPEEVRLLQAHNYQAYPPKNCLLPSGGSLKATDCEEQR
CLLNLPMITASEDCPLIPGVRYQLNLT LARGPGSEEHDHGVEPLGQWLLGAGESLNLTDLLPFTR
YRVSGILSSFYQKKLALPTLVLAPELLELTASATPSPRNFSVRVLSPRELEVSWLPPEQLRSESVY
YTLHWQQELDGENVQDRREWEAHERRLETAGTHRLTGKPGSGYSLWVQAHATPTKSNSSERLHVR
SFAELPELQLELGPYSLSLTWAGTPDPLGSLQLECRSSAEQLRRNVAGNHTKMVVEPLQPRTRYQ
CRLLLGYAATPGAPLYHGTA EYVETLGDAPSQPGKQLEHIAEEVFRVWTAARGNGAPIALYNLE
ALQARSDIRRRRRRRRRNSGGSLQLPWAEPEVVVEDQWDFCNTTELS CIVKSLHSSRLLFRVR
ARSLEHGWPYSEESERVAEPFVSPEKRGSLVLAIIAPAAIVSSCVLALVLRKVQKRRLRACKLL
QQSRPSIWSNLSTLQTQQQLMAVRNRAFSTLLSDADIALLPQINWSKLLRFLGSGAFGEVYEGQ
LKTEDSEEPQRVAIKSLRKGASEFAELLQEAQLMSNFKHENIVRLVGICFDTESISLIMEHMEAGD
LLSYLRAARATSTQEPQTAGLSSELLAMCIDVANGCSYLEDMHFVHRDLACRNLVTESTGSTD
RRRTVKIGDFGLARDIYKSDYYRKEGEGLLPVRWMSPESLVDGLFTTQSDVWAFVGLCWEILT LGQ
QPYAARNFEVLAHVKEGGRLQPPMCTEKLYSLLLLCWRTPWPERPSFRRCYNTLHAISTDLRRT
QMASATADVVSSCSRPEFKVRFDGQPLEEHREHNERPEDENLTLREVPLKDKQLYANEGVSR '

```

图 26. 对 7LES\_DROME 数据运行的样本查询

HMMER 包装器将查询结果变换成关系表，如表 69 中所示。

表 69. HMMER 结果被变换成关系表

Model	ModelScore	DomainNumber	DomainScore
pkinese	+3.04100000000000E+002	1	+3.04100000000000E+002
fn3	+1.76300000000000E+002	1	+4.90000000000000E+001
fn3	+1.76300000000000E+002	2	+1.36000000000000E+001
fn3	+1.76300000000000E+002	3	+1.62000000000000E+001
fn3	+1.76300000000000E+002	4	+6.35000000000000E+001
fn3	+1.76300000000000E+002	5	+1.46000000000000E+001
fn3	+1.76300000000000E+002	6	+1.94000000000000E+001
rrm	-4.45000000000000E+001	1	-4.45000000000000E+001

数据现在采用关系格式，可以与其它数据源中的数据连接起来。

**相关任务:**

- 第 236 页的『将 HMMER 数据源添加至联合服务器』

---

## 将 HMMER 添加至联合服务器

### 将 HMMER 数据源添加至联合服务器

要配置联合服务器以存取 HMMER 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 HMMER 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

**先决条件:**

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

**过程:**

要将 HMMER 数据源添加至联合服务器:

1. 验证是否安装了正确版本的 HMMER 程序可执行文件。
2. 配置 HMMER 守护进程。
3. 启动 HMMER 守护进程。
4. 注册包装器。
5. 注册服务器定义。
6. 注册昵称。

**相关概念:**

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 233 页的『什么是 HMMER?』

**相关任务:**

- 第 44 页的『创建联合数据库』
- 第 237 页的『验证 HMMER 程序可执行文件的版本』
- 第 237 页的『配置 HMMER 守护进程』
- 第 243 页的『注册 HMMER 包装器』
- 第 244 页的『为 HMMER 数据源注册服务器定义』
- 第 246 页的『为 HMMER 数据源注册昵称』
- 第 33 页的『检查联合服务器的设置』

**相关参考:**

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

- 『DB2 Information Integrator 支持的操作系统（64 位）』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 第 250 页的『HMMER 数据源 - 完整示例』

## 验证 HMMER 程序可执行文件的版本

验证是否安装了必需版本的 HMMER 程序可执行文件是将 HMMER 数据源添加至联合服务器这一大型任务的一部分。

在安装 HMMER 应用程序的服务器上必须安装 hmmpfam 和 hmmsearch 可执行文件的受支持版本。可以将 HMMER 应用程序安装在联合服务器上，也可以安装在独立的 HMMER 服务器上。

### 过程:

要检查可执行文件的版本级别:

1. 发出一个将返回版本号命令:
  - 对于 hmmpfam 程序, 命令为:  
hmmpfam -h
  - 对于 hmmsearch 程序, 命令为:  
hmmsearch -h
2. 在输出文件中, 检查可执行文件的版本。版本必须为 HMMER V2.2g (或更新版本)。
3. 如果版本不正确, 则从 <http://hmmer.wustl.edu/> 下载这些文件。

在这一系列任务中的下一个任务是配置 HMMER 守护进程。

### 相关任务:

- 第 237 页的『配置 HMMER 守护进程』

## 配置 HMMER 守护进程

配置 HMMER 守护进程是将 HMMER 添加至联合服务器这一大型任务的一部分。

HMMER 包装器需要 HMMER 守护进程。HMMER 守护进程必须在可以从 DB2 通用数据库联合系统中通过 TCP/IP 访问的服务器上运行。此服务器可以是作为联合服务器运行的服务器, 也可以是一个独立的 HMMER 服务器。

守护进程将与包装器和 DB2 通用数据库分别运行。守护进程从包装器中侦听 HMMER 作业请求。

### 先决条件:

HMMER 守护进程必须:

- 对 hmmpfam 和 hmmsearch 可执行文件具有执行存取权, 以便它可以运行 HMMER 搜索。
- 对它可以在其中编写临时文件的目录具有写存取权。
- 对可以在其上运行 HMMER 搜索的至少一个概要文件数据库具有读存取权。

## 限制:

如果可执行文件路径或者数据库路径包含空格, 则 HMMER 守护进程可能不能正确运行。例如, 不能在 Windows 服务器上的 C:\Program Files 中安装 HMMER 可执行文件。

## 过程:

要配置 HMMER 守护进程:

1. 确保 HMMER 守护进程可执行文件位于正确的服务器上。在安装 DB2 Information Integrator 期间, 守护进程可执行文件安装在联合服务器上的一个目录中:

### 在 UNIX 上

守护进程可执行文件安装在 \$DB2PATH/bin 目录中。

### 在 Windows 上

守护进程可执行文件安装在 %DB2PATH%\bin 目录中。

如果使用独立的 HMMER 服务器, 则必须将守护进程可执行文件从联合服务器上的目录复制到 HMMER 服务器上的一个目录中。守护进程可执行文件可以在 HMMER 服务器上的目录路径中的名称中不包含空格的任何目录中运行。

2. 确保安装 HMMER 的服务器上具有配置文件和其它必需的文件。某些必需的文件与 DB2 Information Integrator 一起安装在联合服务器上的一个目录中。必须提供其它的必需文件。

### 在 UNIX 上

在安装 HMMER 的服务器上必须具有下列文件:

- 守护进程可执行文件 %DB2PATH%/bin/db2hmmmer\_daemon
- HMMER 守护进程配置文件  
%DB2PATH%/samples/lifesci/HMMER\_DAEMON.config
- 转换实用程序 %DB2PATH%/bin/db2h2x
- shell 脚本 %DB2PATH%/bin/db2runpfam.ksh
- HMMER 可执行文件 hmmpfam 和 hmmsearch (它们不是由 IBM 提供的)
- HMMER 数据库文件 (它们不是由 IBM 提供的)

%DB2PATH% 是 DB2 Information Integrator 的安装路径。

### 在 Windows 上

在安装 HMMER 的服务器上必须具有下列文件:

- 守护进程可执行文件 %DB2PATH%\bin\db2hmmmer\_daemon.exe 和 %DB2PATH%\bin\db2hmmmer\_daemon\_svc.exe
- HMMER 守护进程配置文件  
%DB2PATH%\samples\lifesci\HMMER\_DAEMON.config
- 转换实用程序 %DB2PATH%\bin\db2h2x.exe
- HMMER 可执行文件 hmmpfam.exe 和 hmmsearch.exe (它们不是由 IBM 提供的)
- HMMER 数据库文件 (它们不是由 IBM 提供的)

%DB2PATH% 是 DB2 Information Integrator 的安装路径。

缺省情况下, 守护进程期望在启动守护进程的工作目录中找到配置文件。可以将配置文件复制到另一个位置。如果使用 HMMER 服务器, 则必须将守护进程配置文件

从联合服务器上的目录复制到 HMMER 服务器上的一个目录中。可以将守护进程配置文件复制到 HMMER 服务器上守护进程可以访问的任何目录中。

3. 在 UNIX 上，确保 HMMER 守护进程可执行文件、转换实用程序和 shell 脚本是可执行的。要使这些文件可执行，运行以下命令：

```
chmod a+x db2hmmmer_daemon db2h2x db2runpfam.ksh
```

4. 编辑守护进程配置文件以使用数据源。还可以重命名配置文件。
  - 配置文件中的第一行必须是一个等号。如果缺少等号，则将不能启动守护进程。错误消息将指示未指定 DAEMON\_PORT。
  - 配置文件中的最后一行必须以换行符结束。随 DB2 Information Integrator 一起提供的样本配置文件以换行符结束。当您编辑文件时，必须确保文件中的最后一行以换行符结束。如果最后一行不是以换行符结束，则当您试图使用最后一行上列示的数据源来运行第一个 HMMER 查询时就会接收到错误消息。
  - 在配置文件中指定下列选项。对于需要路径的选项，可以指定相对路径。相对路径是相对于启动守护进程的目录的。

#### **DAEMON\_PORT**

这是守护进程侦听由包装器提交的 HMMER 作业请求的网络端口。

#### **MAX\_PENDING\_REQUESTS**

这是在守护进程上任何一次可以阻塞的 HMMER 作业请求的最大数目。此数目并不表示同时运行的 HMMER 作业数，而只是可以同时阻塞的作业请求数。建议您将此数目设置为大于五的数。HMMER 守护进程并不限制可以同时运行的 HMMER 作业数。

#### **DAEMON\_LOGFILE\_DIR**

这是守护进程在其中创建它的日志文件的目录。此文件包含由 HMMER 守护进程生成的有用的状态信息和错误信息。

#### **Q\_SEQ\_DIR\_PATH**

这是守护进程在其中创建临时查询序列数据文件的目录。一旦完成了 HMMER 作业，就会清除此临时文件。

#### **HMMER\_OUT\_DIR\_PATH**

这是守护进程在其中创建临时文件用来存储 HMMER 输出数据的目录。从此文件中读取数据，并通过网络连接将数据传递回包装器。在将数据传递回包装器之后，守护进程将清除临时文件。

#### **RUNPFAM\_PATH**

这是随 DB2 Information Integrator 一起提供的 db2runpfam.ksh shell 脚本的标准名称。如果在 Windows 上指定了此选项，则将忽略它。

#### **HMMERPFAM\_PATH**

这是在运行守护进程的机器上 HMMER 可执行文件的标准名称。在 UNIX 上，该文件的名称为 hmmpfam。在 Windows 上，该文件的名称为 hmmpfam.exe。

#### **HMMSEARCH\_PATH**

这是在运行守护进程的机器上 HMMER 可执行文件的标准名称。在 UNIX 上，该文件的名称为 hmmsearch。在 Windows 上，该文件的名称为 hmmsearch.exe。

## H2X\_PATH

这是与守护进程一起提供的转换程序（从 HMMER 转换为 XML）的标准名称。在 UNIX 上，该程序的名称为 db2h2x。在 Windows 上，该程序的名称为 db2h2x.exe。

## 数据库规范条目

指定概要文件数据库或序列文件的位置。记录下您在配置文件中指定的数据库 *data\_source\_name*。为了使守护进程正确工作，当您为数据源创建昵称时，必须指定数据库 *data\_source\_name*。该名称是区分大小写的。数据库 *data\_source\_name* 是在以下位置指定的：

- CREATE NICKNAME 语句的 DATASOURCE 选项（对于 hmmpfam）
- CREATE NICKNAME 语句的 MODEL 谓词（对于 hmmsearch）

配置文件必须至少包含以下格式的一个数据库规范条目：

```
data_source_name=fully_qualified_name_of_profile_or_sequence_database
```

## 在 UNIX 上

例如，要指定 MYHMMS 概要文件数据库，在守护进程配置文件中应添加下面这一行：

```
myhmms=/home/user_ID/myhmms
```

## 在 Windows 上

例如，要指定 MYHMMS 概要文件数据库，在守护进程配置文件中应添加下面这一行：

```
myhmms=c:\hmm\tutorial\myhmms
```

在这一系列任务中的下一个任务是启动 HMMER 守护进程。

## 相关任务：

- 第 241 页的『启动 HMMER 守护进程』

## 相关参考：

- 第 240 页的『HMMER 守护进程配置文件 - 示例』

## HMMER 守护进程配置文件 - 示例

下列示例显示 PFAM 和 SEARCH 的样本配置文件的内容。

### 示例 - UNIX 的 HMMER\_DAEMON.config 文件：

此示例显示 UNIX 的必需选项和概要文件数据库规范。

```
=
DAEMON_PORT=4098
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=./
Q_SEQ_DIR_PATH=./
HMMER_OUT_DIR_PATH=./
RUNPFAM_PATH=./db2runpfam.ksh
HMMPFAM_PATH=/home/user_id/hmmer/bin/hmmpfam
HMMSEARCH_PATH=/home/user_id/hmmer/bin/hmmsearch
H2X_PATH=/home/user_id/sql1lib/bin/db2h2x
myhmms=/home/user_id/hmmer/tutorial/myhmms
globin=/home/user_id/hmmer/tutorial/globin.hmm
pfamls=/home/user_id/hmmer/pfam/Pfam_ls
```



示例 - Windows 的 HMMER\_DAEMON.config 文件:

此示例显示 Windows 的必需选项和概要文件数据库规范。

```
=  
DAEMON_PORT=4098  
MAX_PENDING_REQUESTS=10  
DAEMON_LOGFILE_DIR=.\  
Q_SEQ_DIR_PATH=.\  
HMMER_OUT_DIR_PATH=.\  
HMMPFAM_PATH=c:\hmm\bin\hmmpfam.exe  
HMMSEARCH_PATH=c:\hmm\bin\hmmsearch.exe  
H2X_PATH=.\db2h2x.exe  
myhmms=c:\hmm\tutorial\myhmms  
globin=c:\hmm\tutorial\globin.hmm  
pfamseq=c:\hmm\pfam\pfamseq
```

相关任务:

- 第 237 页的『配置 HMMER 守护进程』

## 启动 HMMER 守护进程

启动 HMMER 守护进程是将 HMMER 数据源添加至联合服务器这一大型任务的一部分。在可以存取 HMMER 数据源之前，必须启动 HMMER 守护进程。

先决条件:

在启动 HMMER 守护进程之前，您必须对列示在配置文件中的 DAEMON\_LOGFILE\_DIR、HMMER\_OUT\_DIR\_PATH 和 Q\_SEQ\_DIR\_PATH 条目下面的所有路径都具有写存取权。

过程:

要在 UNIX 服务器上启动 HMMER 守护进程:

1. 打开守护进程可执行文件所在的目录。
2. 发出 db2hmmmer\_daemon 命令:
  - 如果未更改守护进程配置文件的名称，并且配置文件与守护进程可执行文件处于同一目录中，则在命令行中输入以下命令:  

```
db2hmmmer_daemon
```
  - 如果您更改了守护进程配置文件的名称，或者如果守护进程配置文件与守护进程可执行文件不在同一目录中，则必须在包装器守护进程命令上使用 -c 选项来将守护进程可执行文件指向新名称或新位置。

例如，以下命令将导致包装器守护进程在子目录 cfg 中的称为 HMMER\_D.config 的文件中查找守护进程配置信息。

```
db2hmmmer_daemon -c cfg/HMMER_D.config
```

可执行文件将启动新进程，HMMER 守护进程就在该进程中运行。

要在 UNIX 服务器上停止守护进程:

1. 通过使用以下 UNIX 命令来列示 db2hmmmer\_daemon 的进程标识:  

```
ps -ef | grep db2hmmmer
```
2. 使用进程标识来停止守护进程。使用以下命令:  

```
kill nnnn
```

其中 `nnnn` 是 `db2hmmmer_daemon` 的进程标识。

要在 Windows 服务器上启动 HMMER 守护进程:

1. 打开守护进程可执行文件所在的目录。
2. 发出带有您需要的参数的 `db2hmmmer_daemon` 命令。例如, 要安装守护进程服务并打开调试并且启动守护进程, 应发出下列命令:

```
db2hmmmer_daemon -a install -d 2
db2hmmmer_daemon -a start
```

要停止守护进程, 使用以下 Windows 命令:

```
db2hmmmer_daemon -a stop
```

在这一系列任务中的下一个任务是注册 HMMER 包装器。

**相关任务:**

- 第 237 页的『配置 HMMER 守护进程』
- 第 243 页的『注册 HMMER 包装器』
- 第 236 页的『将 HMMER 数据源添加至联合服务器』

**相关参考:**

- 第 242 页的『`db2hmmmer_daemon` 命令 - 语法和示例』

## db2hmmmer\_daemon 命令 - 语法和示例

可以在 UNIX 和 Windows 服务器上使用 `db2hmmmer_daemon` 命令。语法中列示的某些自变量只能在 Windows 服务器上使用。

`db2hmmmer_daemon` 命令的语法为:

```
db2hmmmer_daemon -a action -c config_file -d debug_level
                  -u user_id -p password
```

**-a action**

执行指定的活动。有效操作为 `status`、`install`、`start`、`stop` 和 `remove`。

只能在 Windows 服务器上指定此自变量。

**-c config\_file**

指示守护进程服务使用指定的配置文件而不是缺省配置文件。如果不指定配置文件, 则守护进程将在安装了守护进程可执行文件的目录中搜索 `HMMER_DAEMON.config` 文件。可以将此选项与 `install` 和 `start` 操作配合使用。

可以在 UNIX 和 Windows 服务器上指定此自变量。

**-d debug\_level**

将守护进程服务调试级别设置为指定的值。有效值为 1、2 或 3。可以将此选项与 `install` 和 `start` 操作配合使用。

可以在 UNIX 和 Windows 服务器上指定此自变量。

**-u user\_id**

将服务设置为以指定的用户标识运行。可以将此选项与 `install` 操作配合使用。

只能在 Windows 服务器上指定此自变量。

### **-p** *password*

指定所指定的用户标识的密码。仅当指定了 **-u** 选项时，密码才有效并且是必需的。如果设置 **-u** 选项时未指定 **-p** 选项，则程序将提示您输入密码。可以将此选项与 *install* 操作配合使用。

只能在 Windows 服务器上指定此自变量。

随 *start* 操作一起指定的选项只影响守护进程的当前运行，并且将覆盖随 *install* 操作一起指定的值。

### 示例:

下列示例显示 Windows 上的守护进程操作。这些示例假定 *HMMER\_DAEMON.config* 文件与 *db2hmmmer\_daemon.exe* 位于同一目录中。

- 要检查守护进程的状态:  
`db2hmmmer_daemon -a status`
- 要安装守护进程服务并打开调试:  
`db2hmmmer_daemon -a install -d 2`
- 要启动守护进程:  
`db2hmmmer_daemon -a start`
- 要停止守护进程:  
`db2hmmmer_daemon -a stop`
- 要除去或卸载守护进程服务:  
`db2hmmmer_daemon -a remove`

### 相关任务:

- 第 241 页的『启动 HMMER 守护进程』

## 注册 HMMER 包装器

注册 HMMER 包装器是将 HMMER 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 HMMER 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

### 过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 *CREATE WRAPPER* 语句。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 *hmmmer\_wrapper* 的包装器，请发出以下语句:

```
CREATE WRAPPER hmmmer_wrapper LIBRARY 'libdb21shmmmer.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 *CREATE WRAPPER* 语句中应指定的正确名称，请参阅 HMMER 包装器库文件的列表。

在这一系列任务中的下一个任务是为 HMMER 包装器注册服务器定义。

### 相关任务:

- 第 244 页的『为 HMMER 数据源注册服务器定义』

**相关参考:**

- 第 244 页的『HMMER 包装器库文件』
- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）

## HMMER 包装器库文件

下表列示了 HMMER 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2lshmm.a、libdb2lshmmF.a 和 libdb2lshmmU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 70. HMMER 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2lshmm.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lshmm.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lshmm.so
Windows	%DB2PATH%\bin	db2lshmm.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

**相关任务:**

- 第 243 页的『注册 HMMER 包装器』

## 为 HMMER 数据源注册服务器定义

为 HMMER 数据源注册服务器定义是将 HMMER 数据源添加至联合服务器这一大型任务的一部分。

在注册包装器之后，必须注册相应的服务器定义。

**过程:**

要注册 HMMER 服务器定义，发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name TYPE HMMER_search_type
  VERSION version WRAPPER wrapper_name
  OPTIONS (NODE 'node_name', DAEMON_PORT 'port_number')
```

必须对想要运行 HMMER 搜索的每个服务器注册定义。

在这一系列任务中的下一个任务是为 HMMER 数据源注册昵称。

**相关任务:**

- 第 246 页的『为 HMMER 数据源注册昵称』

### 相关参考:

- 第 529 页的附录 D, 『联合系统的服务器选项』
- 第 245 页的『CREATE SERVER 语句 - HMMER 包装器的示例』

## CREATE SERVER 语句 - HMMER 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE SERVER 语句来为 HMMER 包装器注册服务器定义。

要使用 hmmpfam 程序的搜索注册服务器定义 hmmpfam\_server, 发出以下语句:

```
CREATE SERVER hmmpfam_server
  TYPE pfam
  VERSION 2.2
  WRAPPER hmmer_wrapper
  OPTIONS (NODE 'someserver.someschool.edu', DAEMON_PORT '4098')
```

*hmmpfam\_server*

为 HMMER 服务器定义指定的名称。此名称必须是唯一的。

**TYPE** *pfam*

服务器定义支持的搜索类型。可以指定的类型为 PFAM (对于 hmmpfam) 或 SEARCH (对于 hmmsearch)。

**VERSION** 2.2

正在使用的 hmmpfam 或 hmmsearch 可执行文件的版本。受支持的版本为 HMMER 2.2g (或更新版本)。

**WRAPPER** *hmmer\_wrapper*

在 CREATE WRAPPER 语句中指定的名称。

**NODE** '*someserver.someschool.edu*'

HMMER 守护进程运行于的服务器的主机名或 IP 地址。

尽管节点名被指定为 CREATE SERVER 语句中的一个选项, 但是, 它对于 HMMER 数据源是必需的。

**DAEMON\_PORT** '*4098*'

守护进程侦听 HMMER 作业请求的端口号。此端口号必须与在守护进程配置文件的 DAEMON\_PORT 选项中指定的端口号相同。缺省值为 4098。

### 其它服务器选项:

当您创建服务器定义时, 可以在 CREATE SERVER 语句中指定其它服务器选项。

**PROCESSORS**

HMMER 程序使用的处理器数目。此选项等效于 hmmpfam 和 hmmsearch 命令的 --cpu 选项。示例: PROCESSORS '2'。

**HMPFAM\_OPTIONS**

使用此服务器选项来将选项传递给不能在谓词中指定的 hmmpfam 命令。例如: HMPFAM\_OPTIONS '--null2 --pvm'。在此示例中, 每当对此服务器运行查询时, hmmpfam 始终都将带两个附加选项 --null2 和 --pvm 来运行。HMPFAM\_OPTIONS 选项仅对于指定为 PFAM 类型的服务器有效。

**HMMSEARCH\_OPTIONS**

使用此服务器选项来将选项传递给不能在谓词中指定的 hmmsearch 命令。例

| 如: HMMSEARCH\_OPTIONS '--null2 --pvm'。在此示例中, 每当对此服务器运行  
| 查询时, hmmsearch 始终都将带两个附加选项 --null2 和 --pvm 来运行。  
| HMMSEARCH\_OPTIONS 选项仅对于指定为 SEARCH 类型的服务器有效。

| **相关任务:**

- | • 第 244 页的『为 HMMER 数据源注册服务器定义』

| **相关参考:**

- | • 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)  
| • 第 529 页的附录 D, 『联合系统的服务器选项』

## 为 HMMER 数据源注册昵称

为 HMMER 数据源注册昵称是将 HMMER 数据源添加至联合服务器这一大型任务的一部分。

在注册服务器定义之后, 必须注册相应的昵称。当您在查询中引用 HMMER 数据源时就要使用昵称。

**过程:**

要注册 HMMER 昵称, 发出 CREATE NICKNAME 语句。例如:

```
CREATE NICKNAME nickname  
  FOR SERVER server_name  
  OPTIONS(DATASOURCE 'data_source')
```

必须为想要查询的每个概要文件数据库定义独立的昵称。data\_source 名称必须与 HMMER 服务器上的 HMMER\_DAEMON.config 文件中的现有 data\_source\_name 相匹配。

| 当为 HMMER 数据库创建昵称时, 在联合数据库系统目录中为概要文件数据库注册了  
| 一组输入和输出固定列。

在这一系列任务中没有别的任务了。

**相关任务:**

- 第 58 页的『为非关系数据源指定昵称列』
- 第 237 页的『配置 HMMER 守护进程』
- 第 252 页的『使用样本来构造新的 HMMER 查询』
- 第 236 页的『将 HMMER 数据源添加至联合服务器』

**相关参考:**

- 第 247 页的『HMMER 昵称的固定列』
- 第 250 页的『HMMER 数据源 - 完整示例』
- 第 250 页的『CREATE NICKNAME 语句 - HMMER 包装器的示例』
- 第 247 页的『HMMER 昵称的固定列』

## HMMER 昵称的固定列

当对 HMMER 数据源发出 CREATE NICKNAME 语句时，将自动对昵称创建一组固定输入列和固定输出列。如果想更改为固定列指定的缺省数据类型，可以在 CREATE NICKNAME 语句中指定列名和数据类型。例如，要将 AlignmentConsensus 列输出限制为不超过前 100 个字符，发出以下语句：

```
CREATE NICKNAME nucleo1 (AlignmentConsensus VARCHAR(100))
FOR SERVER searchtest
OPTIONS(DATASOURCE 'nucleo1', TIMEOUT '1');
```

可以在 SQL 查询中引用固定列并作为昵称定义的一部分。有两种类型的固定列：输入和输出。

### HMMER 昵称的固定输入列

固定输入列是在 WHERE 子句中指定的。在 SQL 查询中，输入列被用作参数传递谓词。它们将标准的 HMMER 开关传递给 hmmpfam 或 hmmsearch。然后 HMMER 使用这些开关对指定的数据源进行运行。还可以在查询 SELECT 列表中引用固定输入列，固定输入列是作为结果表的一部分返回的。

#### PFAM 类型的服务器的输入固定列：

下表列示了可以在 WHERE 子句中使用的固定列。

表 71. PFAM 类型的服务器的固定输入列

名称	数据类型	描述	运算符	开关	返回的值
HmmQSeq	varchar(32000)	用来搜索的输入基因序列	=		与您指定的输入值相同。此列是必需的。
ModelEValue	double	估计的 E 值	<	-E <i>n</i>	请参阅输出。
ModelScore	double	原始分数	>	-T <i>n</i>	请参阅输出。
DBSize	integer	就象数据库具有“n”个基因序列一样来计算 E 值	=	-Z <i>n</i>	与您指定的输入值相同。如果未指定，则使用 hmmpfam 缺省值。
CutMode	char(2)	截止方式；可以为 ga、tc 或 nc（区分大小写）	=	--cut_ga --cut_tc --cut_nc	与您指定的输入值相同。如果未指定，则为 NULL。
DomainScore	double	域分数	>	--domT <i>n</i>	请参阅输出。
DomainEValue	double	域 E 值	<	--domE <i>n</i>	请参阅输出。
ForwardAlgorithm	char	使用 Forward 算法而不是 Viterbi；值可以为 'Y' 或 'N'	=	--forward	与您指定的输入值相同。'N' 为缺省值。

#### SEARCH 类型的服务器的输入固定列：

下表列示了可以在 WHERE 子句中使用的固定列。

表 72. SEARCH 类型的服务器的固定输入列

名称	数据类型	描述	运算符	选项	返回的值
Model	varchar(32000)	搜索中使用的 HMM 概要文件的名称。该名称必须是在 HMMER_DAEMON.config 文件中的数据库规范条目中列示的其中一个数据源名称。	=		与您指定的输入值相同。此列是必需的。
SequenceEValue	double	估计的 E 值	<	-E <i>n</i>	请参阅输出。
SequenceScore	double	原始分数	>	-T <i>n</i>	请参阅输出。
DBSize	integer	就象数据库具有“ <i>n</i> ”个基因序列一样来计算 E 值	=	-Z <i>n</i>	与您指定的输入值相同。如果未指定，则使用 hmmpfam 缺省值。
CutMode	char(2)	截止方式；可以为 ga、tc 或 nc（区分大小写）	=	--cut_ga --cut_tc --cut_nc	与您指定的输入值相同。如果未指定，则为 NULL。
DomainScore	double	域分数	>	--domT <i>n</i>	请参阅输出。
DomainEValue	double	域 E 值	<	--domE <i>n</i>	请参阅输出。
ForwardAlgorithm	char	使用 Forward 算法而不是 Viterbi；值可以为 'Y' 或 'N'	=	--forward	与您指定的输入值相同。'N' 为缺省值。

## HMMER 昵称的固定输出列

可以在 SELECT 列表中指定任何固定输出列。也可以在 WHERE 子句中指定固定输出列（作为谓词）。

### PFAM 的固定输出列:

下表列示了可以作为 PFAM 的输出返回的固定列。

表 73. PFAM 的固定输出列

名称	数据类型	描述
Model	varchar(32)	模型的名称。
ModelDescription	varchar(64)	模型的文本描述。
ModelScore	double	原始分数（“位分数”）。



表 73. PFAM 的固定输出列 (续)

名称	数据类型	描述
ModelEValue	double	估计的 E 值。
ModelHits	integer	模型中域符合项的数目。
DomainNumber	integer	特定域 (在一个模型中)。
SequenceFrom	integer	基因序列的起始点。
SequenceFromGlobal	char	如果比对从基因序列的开头开始, 则为 'Y'。
HmmFrom	integer	共有模型的起始点。
HmmFromGlobal	char	如果比对从共有模型的开头开始, 则为 'Y'。
HmmTo	integer	共有模型中的结束点。
HmmToGlobal	char	如果比对在共有模型的末尾结束, 则为 'Y'。
DomainScore	double	已隔离的域的原始分数 (“位分数”)。
DomainEValue	double	已隔离的域的期望值。
AlignmentConsensus	varchar(32000)	HMM 共有性。根据 HMM, 显示为共有的氨基酸是该位置处出现概率最高的氨基酸, 而不一定是最高分值的氨基酸。
AlignmentExactMatch	varchar(32000)	与 HMM 中出现概率最高的残余物相匹配。
AlignmentSubSequence	varchar(32000)	显示基因序列本身。

### SEARCH 的固定输出列:

下表列示了可以作为 SEARCH 的输出返回的固定列。

表 74. SEARCH 的固定输出列

名称	数据类型	描述
Sequence	varchar(32)	序列标识。
SequenceDescription	varchar(64)	序列的文本描述。
SequenceScore	double	原始分数 (“位分数”)。
SequenceEValue	double	估计的 E 值。
SequenceHits	integer	序列中域符合项的数目。
DomainNumber	integer	特定域 (在一个序列中)。
SequenceFrom	integer	基因序列的起始点。
SequenceFromGlobal	char	如果比对从基因序列的开头开始, 则为 'Y'。
HmmFrom	integer	共有模型的起始点。
HmmFromGlobal	char	如果比对从共有模型的开头开始, 则为 'Y'。
HmmTo	integer	共有模型中的结束点。
HmmToGlobal	char	如果比对在共有模型的末尾结束, 则为 'Y'。
DomainScore	double	已隔离的域的原始分数 (“位分数”)。
DomainEValue	double	已隔离的域的期望值。
AlignmentConsensus	varchar(32000)	HMM 共有性。根据 HMM, 显示为共有的氨基酸是该位置处出现概率最高的氨基酸, 而不一定是最高分值的氨基酸。
AlignmentExactMatch	varchar(32000)	与 HMM 中出现概率最高的残余物相匹配。
AlignmentSubSequence	varchar(32000)	显示基因序列本身。

#### 相关任务:

- 第 246 页的『为 HMMER 数据源注册昵称』

#### 相关参考:

- 第 250 页的『HMMER 数据源 - 完整示例』
- 第 250 页的『CREATE NICKNAME 语句 - HMMER 包装器的示例』

## CREATE NICKNAME 语句 - HMMER 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE NICKNAME 语句来为 HMMER 数据源注册昵称。

要注册使用服务器定义 `hmmpfam_server` 的昵称 `hmmpfam_nickname`，发出以下语句：

```
CREATE NICKNAME hmmpfam_nickname
  FOR SERVER hmmpfam_server
  OPTIONS(DATASOURCE 'myhmms',TIMEOUT '30')
```

#### *hmmpfam\_nickname*

为昵称指定的名称。此名称必须是唯一的。

#### **SERVER** *hmmpfam\_server*

想要与此昵称相关联的服务器定义的名称。

#### **DATASOURCE** *'myhmms'*

将对其运行 HMMER 搜索的数据库的名称。此数据库必须列示在 HMMER 守护进程配置文件中。

尽管数据源被指定为 CREATE NICKNAME 语句中的一个选项，但是，它对于 HMMER 数据源是必需的。

#### **TIMEOUT** *'30'*

包装器等待来自守护进程的结果的最长时间（以分钟计）。缺省值为 60 分钟。

#### 相关任务:

- 第 246 页的『为 HMMER 数据源注册昵称』

#### 相关参考:

- 第 543 页的附录 F，『联合系统的昵称选项』

## HMMER 数据源 - 完整示例

本主题提供了将 HMMER 数据源添加至联合服务器时需要发出的所有 SQL 语句的示例。此示例还显示使用在 SQL 语句中创建的昵称运行的查询。

HMMER 数据源的 SQL 语句必须包含用来将标准 HMMER 选项传递给程序可执行文件的特殊输入谓词。传递给 HMMER 包装器的每个查询必须至少包含 `HmmQSeq` 输入谓词（对于 TYPE PFAM）或 `model` 谓词（对于 TYPE SEARCH）。所有其它谓词是可选的。

要对昵称构造 HMMER 查询，在 WHERE 子句中指定输入列，在 SELECT 列表中指定输出列。

### hmmpfam 程序的示例:

此示例将在 AIX 联合服务器上为 hmmpfam 程序创建包装器、服务器定义和昵称。此示例还要运行一个查询，该查询将字符串文字用于搜索序列。

```
CREATE WRAPPER hmmer_wrapper
  LIBRARY 'libdb2lshmmmer.a';

CREATE SERVER hmmpfam_server
  TYPE pfam VERSION 2.2
  WRAPPER hmmer_wrapper
  OPTIONS(NODE 'HMMERServ.MyCompany.com');

CREATE NICKNAME hmmpfam_nickname
  FOR SERVER hmmpfam_server
  OPTIONS(DATASOURCE 'myhmms', TIMEOUT '1');

-- Run the 7LES_DROME gene sequence on the hmmpfam_nickname
SELECT Model, substr(ModelDescription,1,50) as ModelDescription,
  ModelScore, ModelEValue, ModelHits, DomainNumber,
  SequenceFrom, SequenceTo, SequenceFromGlobal, SequenceToGlobal,
  HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal,
  DomainScore, DomainEValue,
  length(HmmQSeq)          as "length(HmmQSeq)",
  length(AlignmentConsensus) as "length(AConsensus)",
  length(AlignmentMatch)   as "length(AMatch)",
  length(AlignmentSubSeq)  as "length(ASubSeq)",
  substr(HmmQSeq,1,64)      as HmmQSeq,
  substr(AlignmentConsensus,1,64) as AlignmentConsensus,
  substr(AlignmentMatch, 1,64) as AlignmentMatch,
  substr(AlignmentSubSeq, 1,64) as AlignmentSubSeq
FROM hmmpfam_nickname
WHERE HmmQSeq =
  'MTMFWQQNVHDHQSDEQDKAKGAAPT KRLNISFNVKIAVNVNTKMTTTHINQAPGTSS...';
```

### hmmsearch 程序的示例:

此示例将在 Windows 联合服务器上为 hmmsearch 程序创建包装器、服务器定义和昵称。

```
CREATE WRAPPER hmmer_wrapper
  LIBRARY 'db2lshmmmer.dll'
  OPTIONS(DB2_FENCED 'Y');

CREATE SERVER hmmsearch_serv
  TYPE search VERSION 2.2
  WRAPPER hmmer_wrapper
  OPTIONS(NODE 'localhost');

CREATE NICKNAME artemia
  FOR SERVER hmmsearch_server
  OPTIONS(DATASOURCE 'artemia', TIMEOUT '1');

SELECT Model, Sequence, substr(SequenceDescription,1,50)
  as SequenceDescription, SequenceScore, SequenceEValue,
  SequenceHits, DomainNumber, SequenceFrom,
  SequenceTo, SequenceFromGlobal, SequenceToGlobal,
  HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal, DomainScore,
  DomainEValue,
  length(AlignmentConsensus) as "length(AConsensus)",
  length(AlignmentMatch)     as "length(AMatch)",
  length(AlignmentSubSeq)    as "length(ASubSeq)",
  substr(AlignmentConsensus,1,200) as AlignmentConsensus,
  substr(AlignmentMatch, 1,200)  as AlignmentMatch,
  substr(AlignmentSubSeq, 1,200)  as AlignmentSubSeq
FROM artemia
WHERE Model = 'globin' and DomainScore > 50;
```

### 相关任务:

- 第 246 页的『为 HMMER 数据源注册昵称』
- 第 252 页的『使用样本来构造新的 HMMER 查询』
- 第 236 页的『将 HMMER 数据源添加至联合服务器』

---

## 使用样本来构造新的 HMMER 查询

下列样本 HMMER 查询说明如何为 HMMER 数据源构造查询。

### 过程:

要运行查询，将下列示例用作指南。

在这些查询中，昵称是用来描述 HMMER 搜索和数据源的类型的一个名称。某些示例还说明了如何将 HMMER 包装器与其它数据源配合使用。

### 查询 1

```
SELECT Model, ModelScore, ModelEValue, DomainNumber, DomainScore, DomainEValue
  FROM hmmpfam_nickname
 WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPTKRLNISFNVKIAVNVNTKMTTTHINQ...'
```

当此 SQL 语句运行时，包装器使用指示的序列和由昵称定义的 HMM 数据库来运行 hmmpfam 程序。包装器将返回在 SELECT 语句中列示的列。

### 查询 2

```
SELECT Model, ModelScore, ModelEValue
  FROM hmmpfam_nickname
 WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPTKRLNISFNVKIAVNVNTKMTTTHINQ...'
 AND ModelScore > 0
```

当此 SQL 语句运行时，包装器将对使用所指示的基因序列的 hmmpfam\_nickname 执行 hmmpfam 搜索。另外，包装器将把 -T 0 选项传递给 hmmpfam 命令。此选项来自于 HMMER 昵称的固定列的列表。包装器将返回列示在 SELECT 后面的三列。

### 查询 3

```
SELECT Model, DomainNumber, DomainScore, DomainEValue
  FROM hmmpfam_nickname
 WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPTKRLNISFNVKIAVNVNTKMTTTHINQ...'
 AND ModelEValue < 1
 ORDER BY DomainScore DESC
```

当此 SQL 语句运行时，包装器将对使用所指示的基因序列的 hmmpfam\_nickname 执行 hmmpfam 搜索。另外，包装器将把 -E 1 选项传递给 hmmpfam 命令。此选项来自于 HMMER 昵称的固定列的列表。包装器将返回列示在 SELECT 后面的四列，并将结果按照 DomainScore 从最高到最低的顺序进行排序。

### 查询 4

```
CREATE WRAPPER hmmer_wrapper
  LIBRARY 'db21shmmer.dll';

CREATE SERVER hmsearch_server
  TYPE search VERSION 2.2
 WRAPPER hmmer_wrapper
 OPTIONS(NODE 'HMMERServ.MyCompany.com');
```

```

CREATE NICKNAME artemia_nickname
FOR SERVER hmmsearch_server
OPTIONS(DATASOURCE 'artemia', TIMEOUT '1');

SELECT Model, Sequence, substr(SequenceDescription,1,50)
as SequenceDescription, SequenceScore, SequenceEValue,
SequenceHits, DomainNumber, SequenceFrom,
SequenceTo, SequenceFromGlobal, SequenceToGlobal,
HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal, DomainScore,
DomainEValue,
length(AlignmentConsensus) as "length(AConsensus)",
length(AlignmentMatch) as "length(AMatch)",
length(AlignmentSubSeq) as "length(ASubSeq)",
substr(AlignmentConsensus,1,200) as AlignmentConsensus,
substr(AlignmentMatch, 1,200) as AlignmentMatch,
substr(AlignmentSubSeq, 1,200) as AlignmentSubSeq
FROM artemia_nickname
WHERE Model = 'globin' and DomainScore > 50;

```

当此 SQL 语句运行时，包装器将使用由 globin 指定的 HMM 来对序列文件 artemia 运行 hmmsearch。将返回 DomainScore 大于 50 的那些行，这是因为包装器将 --domT 50 选项传递给了 **hmmsearch** 命令。包装器将返回在 SELECT 后面指定的列。长于 200 个字符的列值将被截断。将只返回这些列中的前 200 个字符。

#### 相关任务:

- 第 236 页的『将 HMMER 数据源添加至联合服务器』

#### 相关参考:

- 第 247 页的『HMMER 昵称的固定列』
- 第 250 页的『HMMER 数据源 - 完整示例』
- 第 253 页的『HMMER 包装器的消息』

## HMMER 包装器的消息

为了使 HMMER 包装器工作，必须指定在 HmmQSeq 列上包含谓词的查询。当查询在 HmmQSeq 列上缺少谓词的片段时，将产生错误。

本节列示和描述了您在使用 HMMER 包装器时可能会遇到的消息。

表 75. HMMER 包装器消息

错误代码	消息	解释
SQL0142N	SQL 语句不受支持。	包装器不能处理提交给 DB2 的 SQL 查询。添加必需的谓词，然后重新提交。验证谓词中使用的运算符对于该列是否有效。请参阅 HMMER 昵称的固定列。
SQL1822N	从数据源“Hmmer 包装器”中接收到意外的错误代码“未指定的错误”。相关联的文本和标记为“不能解析 NODE 主机名”。	在 CREATE SERVER 中指定的 TCP/IP NODE 名无效。

表 75. HMMER 包装器消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“Hmmer 包装器”中接收到意外的错误代码“未指定的错误”。相关联的文本和标记为“未能连接至守护进程”。	hmmmer_daemon 程序当前未在目标节点上运行, 或者在 CREATE SERVER 命令中指定的 DAEMON_PORT 与在守护进程配置文件 HMMER_DAEMON.config 中指定的 DAEMON_PORT 值不匹配。
SQL1822N	从数据源“Hmmer 包装器”中接收到意外的错误代码“未指定的错误”。相关联的文本和标记为“hmmmer 守护进程中存在未知错误”。	在 CREATE NICKNAME 语句中指定的 DATASOURCE 名称不能与守护进程配置文件 HMMER_DAEMON.config 中列示的任何概要文件数据库名称相匹配。
SQL1822N	从数据源“Hmmer 包装器”中接收到意外的错误代码“未指定的错误”。相关联的文本和标记为“致命错误: 没有这样的选项“--cut_TC””。	必须用小写来指定 CutMode 谓词。示例: WHERE CutMode = 'tc'

**相关概念:**

- 『消息介绍』(《消息参考》第 1 卷)

**相关参考:**

- 『SQLSTATE 消息』(《消息参考》第 2 卷)

---

## 第 15 章 配置对 Informix 数据源的存取

本章说明如何配置联合服务器以存取存储在 Informix 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Informix 数据源的存取。

本章列示配置联合服务器时需要执行的任务，包含您需要的 SQL 语句的示例，而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 Informix 添加至联合服务器

#### 将 Informix 数据源添加至联合服务器

要配置联合服务器以存取 Informix 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Informix 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

##### 先决条件:

- 为联合配置的 DB2 服务器。
- 联合数据库必须存在于联合服务器上
- 在联合服务器上已经安装和配置了 Informix Client SDK 软件。
- 在 AIX 联合服务器上的“AIX 基本应用程序开发数学库”。可以通过发出 AIX 命令 **lslpp -l bos.adt.libm** 来确定是否安装了该库。

##### 过程:

要将 Informix 数据源添加至联合服务器:

1. 设置和测试 Informix 客户机配置文件。
2. 设置 Informix 环境变量。
3. 注册包装器。
4. 注册服务器定义。
5. 创建用户映射。
6. 测试与 Informix 服务器的连接。
7. 为 Informix 表、视图和同义词注册昵称。

##### 相关概念:

- 第 49 页的『配置数据源的快速方法』

##### 相关任务:

- 第 38 页的『检查 FEDERATED 参数』
- 第 44 页的『创建联合数据库』
- 第 256 页的『设置和测试 Informix 客户机配置文件』

- 第 260 页的『注册 Informix 包装器』
- 第 261 页的『为 Informix 数据源注册服务器定义』
- 第 263 页的『为 Informix 数据源创建用户映射』
- 第 265 页的『测试与 Informix 服务器的连接』
- 第 266 页的『为 Informix 表、视图和同义词注册昵称』
- 第 267 页的『对 Informix 数据源的配置进行调整和故障诊断』
- 第 257 页的『设置 Informix 环境变量』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 设置和测试 Informix 客户机配置文件

设置和测试 Informix 客户机配置文件是将 Informix 数据源添加至联合服务器这一大型任务的一部分。

客户机配置文件用来使用安装在联合服务器上的客户机库来连接至 Informix。此文件指定每个 Informix 数据库服务器的位置以及数据库服务器的连接 (协议) 的类型。

客户机配置文件的缺省位置取决于联合服务器使用的操作系统。

- 在 UNIX 操作系统上, 此文件的缺省位置和名称为 \$INFORMIXDIR/etc/sqlhosts。sqlhosts 文件是随 Informix Client SDK 一起安装的。
- 在 Windows 操作系统上, sqlhosts 注册表的缺省位置为本地计算机。

在 *Administrator's Guide for Informix Dynamic Server* 中描述了 sqlhosts 的格式。

#### 过程:

要设置和测试 Informix 客户机配置文件:

1. 配置 Informix Client SDK。
  - 在 UNIX 上, 可以通过编辑 sqlhosts 文件来配置 Informix Client SDK。还可以从安装了 Informix Connect 或 Informix Client SDK 的另一个系统中复制 sqlhosts 文件。
  - 在 Windows 上, 可以使用 Informix Setnet32 实用程序来配置 Informix Client SDK。Setnet32 实用程序将设置 sqlhosts 注册表。
2. 验证 sqlhosts 文件或注册表的位置。
  - 在 UNIX 操作系统上, sqlhosts 文件位于 \$INFORMIXDIR/etc/ 目录中。
  - 在 Windows 操作系统上, sqlhosts 信息保存在 Windows 注册表中的以下键中:  
HKEY\_LOCAL\_MACHINE\SOFTWARE\INFORMIX\SQLHOSTS
3. 如果 sqlhosts 文件或注册表未在缺省位置, 则应设置环境变量 INFORMIXSQLHOSTS。



- 在 UNIX 操作系统上，将环境变量 INFORMIXSQLHOSTS 设置为 sqlhosts 文件的标准名称。
  - 在 Windows 操作系统上，将环境变量 INFORMIXSQLHOSTS 设置为用来存储注册表的 Windows 计算机的名称。
4. 测试连接以确保客户机软件能够连接至 Informix 服务器。如果 Informix **dbaccess** 工具在联合服务器上，则使用此工具来测试连接。否则，运行 Informix 演示程序来测试客户机设置。

在这一系列任务中的下一个任务是设置 Informix 环境变量。

#### 相关任务:

- 第 260 页的『注册 Informix 包装器』
- 第 267 页的『对 Informix 数据源的配置进行调整和故障诊断』

## 设置 Informix 环境变量

设置 Informix 环境变量是将 Informix 添加至联合服务器这一大型任务的一部分。

当安装 DB2 Information Integrator 时，安装过程将尝试在 db2dj.ini 文件中设置 Informix 环境变量。

如果存在下列情况，则不会在 db2dj.ini 文件中设置环境变量:

- 在设置 DB2 联合服务器之后才安装 Informix 客户机软件。
- 尚未安装 Informix 客户机软件。

Informix 的有效环境变量为:

- INFORMIXDIR
- INFORMIXSERVER
- INFORMIXSQLHOSTS (可选)
- CLIENT\_LOCALE (可选)
- DB\_LOCALE (可选)
- DBNLS (可选)

必须手工设置可选环境变量。

#### 先决条件:

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件 (例如，数据源客户机软件)。

#### 限制:

请参阅主题: db2dj.ini 文件的限制

#### 过程:

要自动设置必需的环境变量:

1. 在 DB2 联合服务器上安装和配置客户机软件 (如果需要的话)。

2. 设置必需的环境变量。可以通过再次运行 DB2 服务器安装并指定“典型”安装选项来自动设置环境变量。

要手工设置环境变量:

1. 编辑 db2dj.ini 文件。

- 在运行 Windows 的联合服务器上, 此文件位于 %DB2PATH%\cfg 目录中。
- 在运行 UNIX 的联合服务器上, 此文件位于 sqllib/cfg 目录中。

db2dj.ini 文件包含有关安装在联合服务器上的 Informix 客户机软件的配置信息。如果该文件不存在, 则可以使用任何文本编辑器来创建名称为 db2dj.ini 的新文件。在 db2dj.ini 文件中, 必须指定变量的全限定路径, 否则将遇到错误。

2. 根据需要设置下列环境变量:

### INFORMIXDIR

将 INFORMIXDIR 环境变量设置为安装了 Informix Client SDK 软件的目录路径。例如:

在运行 Windows 的联合服务器上, 将该路径设置为:

```
INFORMIXDIR=C:\informix\csdk
```

在运行 UNIX 的联合服务器上, 将该路径设置为:

```
INFORMIXDIR=/informix/csdk
```

### INFORMIXSERVER

此变量标识缺省 Informix 服务器的名称。此设置必须是 sqlhosts 文件 (UNIX) 或 SQLHOSTS 注册表键 (Windows) 中的有效条目。要获得 INFORMIXSERVER 的值, 请阅读 sqlhosts 文件。选择其中一个 *dbservername* 值。*dbservername* 是 sqlhosts 中的每个条目中的第一个值。例如:

```
INFORMIXSERVER=inf93
```

**需求:** 尽管 Informix 包装器不使用此变量的值, 但是 Informix 客户机要求设置此变量。包装器使用 **NODE** 服务器选项的值, 该选项指定想要访问的 Informix 数据库服务器。

### INFORMIXSQLHOSTS

如果您正在使用 Informix sqlhosts 文件的缺省路径, 则不需要设置此变量。但是, 如果您正在使用 Informix sqlhosts 文件的其它路径, 则需要将此变量设置为 Informix sqlhosts 文件驻留的全路径名。

- 在运行 UNIX 的联合服务器上, 缺省路径为 \$INFORMIXDIR/etc。
- 在运行 Windows 的联合服务器上, 如果 SQLHOSTS 注册表键不驻留在本地计算机上, 则 INFORMIXSQLHOSTS 是存储注册表的 Windows 计算机的名称。

UNIX 中, 将此变量设置为另一个路径的示例为:

```
INFORMIXSQLHOSTS=/informix/csdk/etc/my_sqlhosts
```

3. 要确保在联合服务器上设置了环境变量, 重新启动 DB2 实例。发出下列命令来重新启动 DB2 实例:

```
db2stop  
db2start
```

## 设置 Informix 代码页转换:

对于 Informix 代码页转换, 可以设置下列可选环境变量:

- CLIENT\_LOCALE
- DB\_LOCALE
- DBNLS

每当 Informix 包装器连接至 Informix 数据源, 包装器就会确定将哪个代码页值用于该连接。如果在联合服务器上的 db2dj.ini 文件中设置了 Informix 环境变量 CLIENT\_LOCALE, 则包装器将使用 db2dj.ini 文件中的值。

可以通过在 Informix 服务器上发出 **glfiles** 命令来获得有效 Informix 语言环境的列表。有关代码页转换的更多信息, 参阅 *GLS 功能的 Informix 指南*。

Informix 代码页环境变量为:

### CLIENT\_LOCALE

将 CLIENT\_LOCALE 环境变量设置为想要使用的 Informix 语言环境。如果未设置 CLIENT\_LOCALE, 则包装器将确定联合数据库的代码页和地域。包装器将 CLIENT\_LOCALE 变量设置为最匹配的 Informix 语言环境。如果没有匹配 Informix 语言环境, 则对于 UNIX 系统, 包装器将把 CLIENT\_LOCALE 变量设置为 en\_us.8859-1 语言环境, 而对于 Windows 系统, 将把该变量设置为 en\_us.CP1252 语言环境。可以通过使用 Informix **glfiles** 命令来查看语言环境名称的列表。

```
CLIENT_LOCALE=Informix_client_locale_value
```

### DB\_LOCALE

如果 Informix 数据库使用与客户机语言环境不同的代码页, 并且您想要 Informix 在这两种代码页之间执行转换, 则设置此环境变量。将 Informix 环境变量 DB\_LOCALE 设置为 Informix 数据库语言环境的名称, 例如:

```
DB_LOCALE=Informix_db_locale_value
```

### DBNLS

要让 Informix 验证 DB\_LOCALE 设置与 Informix 数据库的实际语言环境是否相匹配, 将此 Informix 环境变量设置为 1。

```
DBNLS=1
```

### 代码页环境变量示例:

假定 Informix 数据库使用与客户机语言环境不同的代码页, 并且您想要 Informix 在这两种代码页之间执行转换。需要将 Informix 环境变量 DB\_LOCALE 设置为 Informix 数据库语言环境的名称。在联合服务器上的 db2dj.ini 文件中设置此变量。

如果想要 Informix 验证 DB\_LOCALE 设置与 Informix 数据库的实际语言环境是否相匹配, 则需要将 Informix 环境变量 DBNLS 设置为 1。必须在联合服务器上的 db2dj.ini 文件中设置此变量。

如果存取包含使用中文代码页 GB 18030 的数据的数据源, 则联合数据库必须使用 UTF-8 代码页。Informix 包装器将 Informix 环境变量设置为:

```
CLIENT_LOCALE=zh_cn.UTF8  
GL_USEGLU=1
```

必须将以下设置添加至 db2dj.ini 文件，以便 Informix 客户机正确地将 GB 18030 数据转换为 Unicode:

```
DB_LOCALE=zh_cn.GB18030-2000
```

在这一系列任务中的下一个任务是注册 Informix 包装器。

#### 相关任务:

- 第 266 页的『为 Informix 表、视图和同义词注册昵称』

#### 相关参考:

- 第 53 页的『对于 db2dj.ini 文件的限制』
- 第 53 页的『对于 db2dj.ini 文件的限制』

## 注册 Informix 包装器

注册 Informix 包装器是将 Informix 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Informix 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器，发出 CREATE WRAPPER 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER INFORMIX
```

**建议:** 使用称为 INFORMIX 的缺省包装器名称。当您使用缺省名称来注册包装器时，联合服务器将自动采用与该包装器名称相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突，则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于其中一个缺省名称的名称，则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 inf\_wrapper 的包装器，请发出以下语句:

```
CREATE WRAPPER inf_wrapper LIBRARY 'libdb2informix.a'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 Informix 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Informix 包装器注册服务器定义。

#### 相关任务:

- 第 261 页的『为 Informix 数据源注册服务器定义』

#### 相关参考:

- 第 261 页的『Informix 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## Informix 包装器库文件

下表列示了 Informix 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 `libdb2informix.a`、`libdb2informixF.a` 和 `libdb2informixU.a`。

当注册包装器时，只指定该表中列示的库文件名。

表 76. Informix 包装器库位置和文件名

操作系统	目录路径	库文件名
AIX	/usr/opt/db2_08_01/lib/	libdb2informix.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2informix.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2informix.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2informix.so
Windows	%DB2PATH%\bin	db2informix.dll

`%DB2PATH%` 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 `C:\Program Files\IBM\SQLLIB`。

### 相关任务:

- 第 260 页的『注册 Informix 包装器』

## 为 Informix 数据源注册服务器定义

为 Informix 数据源注册服务器定义是将 Informix 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 Informix 服务器。

### 过程:

可以从 DB2 控制中心或 DB2 命令行注册服务器定义:

- 要从 DB2 控制中心执行此任务，使用“联合对象”向导或者右键单击**服务器定义**文件夹并单击**创建**。“发现”工具将为 Informix 服务器检索节点名。必须指定 `DBNAME` 服务器选项的信息才能注册服务器定义。
- 要从 DB2 命令行执行此任务，请使用 `CREATE SERVER` 语句:
  1. 在 Informix `sqlhosts` 文件或注册表中找到节点名。

### 样本 `sqlhosts` 文件:

```
inf724 onsoctcp anaconda inmx724
inf731 onsoctcp boa ifmx731
inf92 onsoctcp python ifmx92
```

每一行中的第一个值是 `node_name`，例如 `inf724`。

每一行中的第二个值是 `nettype` 或者是连接类型。在此示例中，`onsoctcp` 指示这是一个 TCP/IP 连接。

每一行中的第三个值是主机名称，例如，`anaconda`、`boa` 和 `python`。

每一行中的第四个值是服务名称，例如，`inmx724`。服务名称字段取决于第二个值中列示的 `nettype`。

尽管 `node_name` 被指定为 `CREATE SERVER SQL` 语句中的一个选项，但是，它对于 Informix 数据源也是必需的。

有关此文件的格式和这些字段的含义的更多信息，请参阅 Informix 手册 *Administrators Guide for Informix Dynamic Server*。

## 2. 发出 `CREATE SERVER` 语句。

例如：

```
CREATE SERVER server_name TYPE informix
VERSION 9 WRAPPER INFORMIX
OPTIONS (NODE 'node_name', DBNAME 'db_name');
```

在注册了服务器定义之后，使用 `ALTER SERVER` 语句来添加或删除服务器选项。

在这一系列任务中的下一个任务是为 Informix 数据源创建用户映射。

**相关任务：**

- 第 263 页的『为 Informix 数据源创建用户映射』

**相关参考：**

- 『`CREATE SERVER` statement』 (*SQL Reference, Volume 2*)
- 第 529 页的附录 D，『联合系统的服务器选项』
- 第 262 页的『`CREATE SERVER` 语句 - Informix 包装器的示例』

## CREATE SERVER 语句 - Informix 包装器的示例

此主题提供了几个示例来说明如何使用 `CREATE SERVER` 语句来为 Informix 包装器注册服务器定义。此主题包括一个完整的示例，该示例说明如何使用必需的参数来注册服务器定义，它还包括一个带有其它服务器选项的示例。

**完整示例：**

以下示例说明了如何使用 `CREATE SERVER` 语句来为 Informix 包装器注册服务器定义：

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER INFORMIX
OPTIONS (NODE 'abc', DBNAME 'sales', IUD_APP_SVPT_ENFORCE 'N')
```

`asia` 为 Informix 数据库服务器指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

**TYPE `informix`**

指定正在对其配置存取权的数据源服务器的类型。对于 Informix 包装器，服务器类型必须是 `informix`。

**VERSION 9**

想要访问的 Informix 数据库服务器的版本。受支持的 Informix 版本为 7、8 和 9。

## WRAPPER *INFORMIX*

在 CREATE WRAPPER 语句中指定的名称。

## NODE 'abc'

Informix 数据库服务器所驻留的节点的名称。从 sqlhosts 文件中可获取节点名。此值是区分大小写的。

尽管节点名被指定为 CREATE SERVER 语句中的一个选项，但是，它对于 Informix 数据源是必需的。

## DBNAME 'sales'

想要存取的 Informix 数据库的名称。此值是区分大小写的。

尽管数据库名称被指定为 CREATE SERVER 语句中的一个选项，但是，它对于 Informix 数据源是必需的。

## IUD\_APP\_SVPT\_ENFORCE 'N'

指定 DB2 联合系统是否应该强制检测或构建应用程序保存点语句。Informix 不支持应用程序保存点语句。当设置为 'N' 时，联合服务器在遇到错误时将不回滚事务。应用程序必须处理错误恢复。

必须将 IUD\_APP\_SVPT\_ENFORCE 服务器选项设置为 'N' 才能复制至 Informix 数据源或者从 Informix 数据源中进行复制。

## 服务器选项示例:

当您创建服务器定义时，可以在 CREATE SERVER 语句中指定更多服务器选项。这些服务器选项可以是一般的服务器选项和特定于 Informix 的服务器选项。

以下示例显示了具有更多服务器选项的 Informix 服务器定义:

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER INFORMIX
  OPTIONS (NODE 'abc', DBNAME 'sales', FOLD_ID 'N', FOLD_PW 'N')
```

当联合服务器连接至数据源时，联合服务器将使用用户标识和密码的大小写的所有可能存在的组合以及当前的大小写来尝试进行连接。在成功连接至数据源服务器之前，联合服务器最多可以尝试进行九次连接。这些尝试会减少连接次数，并且可能会导致用户标识被锁定。可以通过指定 FOLD\_ID 和 FOLD\_PW 服务器选项的值来防止被锁定。

例如，可以将 FOLD\_ID 和 FOLD\_PW 服务器选项设置为 'N'（不转变用户标识或密码的大小写）。如果建立了这些设置，则必须按正确的大小写指定用户标识和密码。将这些选项设置为 'N' 的优点是：当指定了无效用户标识或密码时，包装器将不会继续尝试各种大小写组合。这两个服务器选项可以减少超过登录尝试失败的最大次数和用户标识被锁定的机会。

## 相关任务:

- 第 261 页的『为 Informix 数据源注册服务器定义』

## 相关参考:

- 『CREATE SERVER statement』（*SQL Reference, Volume 2*）

## 为 Informix 数据源创建用户映射

为 Informix 数据源创建用户映射是将 Informix 数据源添加至联合服务器这一大型任务的一部分。

当您尝试访问 Informix 服务器时，联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联（用户映射）。为将访问联合系统以便对 Informix 数据源发送分布式请求的每个用户标识创建用户映射。

#### 过程:

要将本地用户标识映射至 Informix 服务器用户标识和密码，发出 CREATE USER MAPPING 语句。

例如:

```
CREATE USER MAPPING FOR USERID SERVER INFORMIXSERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

在这一系列任务中的下一个任务是测试与 Informix 服务器的连接。

#### 相关任务:

- 第 265 页的『测试与 Informix 服务器的连接』

#### 相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)
- 第 264 页的『CREATE USER MAPPING 语句 - Informix 包装器的示例』

## CREATE USER MAPPING 语句 - Informix 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将联合服务器用户标识映射至 Informix 服务器用户标识和密码。此主题包括一个具有必需参数的完整示例，还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

#### 完整示例:

以下示例说明如何将联合服务器用户标识 (*VINCENT*) 映射至 Informix 服务器用户标识和密码 (*'vinnie'* 和 *'close2call'*) :

```
CREATE USER MAPPING FOR VINCENT SERVER asia
    OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call')
```

#### *VINCENT*

指定您正在映射至在 Informix 服务器中定义的用户标识的本地用户标识。

#### **SERVER** *asia*

指定您在 CREATE SERVER 语句中注册的 Informix 服务器的名称。

#### **REMOTE\_AUTHID** '*vinnie*'

指定您正在将 *VINCENT* 映射至的 Informix 数据库服务器的用户标识。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_ID 服务器选项设置为 'U' 或 'L'）。

#### **REMOTE\_PASSWORD** '*close2call*'

指定与 '*vinnie*' 相关联的密码。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_PW 服务器选项设置为 'U' 或 'L'）。

#### 专用寄存器示例:



以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER asia
    OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

#### 相关任务:

- 第 263 页的『为 Informix 数据源创建用户映射』

#### 相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)

## 测试与 Informix 服务器的连接

测试与 Informix 服务器的连接是将 Informix 数据源添加至联合服务器这一大型任务的一部分。

以使用您定义的服务器定义和用户映射来测试与 Informix 服务器的连接。

#### 过程:

要测试连接:

1. 打开传递会话以便对 Informix 系统表发出 SQL SELECT 语句。

例如:

```
SET PASSTHRU server_name
SELECT count(*) FROM informix.systables
SET PASSTHRU RESET
```

如果该 SQL SELECT 语句返回了一个计数,则表示正确设置了服务器定义和用户映射。

2. 如果 SQL SELECT 语句返回了错误,则可能需要:

- 检查 Informix 服务器以确保为入局连接配置了 Informix 服务器。
- 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 Informix 服务器的连接是有效的。在需要时,改变用户映射,或者创建另一个用户映射。
- 检查 DB2 联合服务器上的 Informix Client SDK 软件,以确保正确安装和配置了该软件以便连接至 Informix 服务器。
- 检查 DB2 联合变量的设置以验证它们对于 Informix 服务器是否正确。这些变量包括系统环境变量、db2dj.ini 变量和“DB2 概要文件注册表”(db2set)变量。
- 检查服务器定义。如果需要的话,删除服务器定义,然后再次创建它。

在这一系列任务中的下一个任务是为 Informix 表、视图和同义词注册昵称。

#### 相关任务:

- 第 255 页的『将 Informix 数据源添加至联合服务器』
- 第 266 页的『为 Informix 表、视图和同义词注册昵称』
- 第 257 页的『设置 Informix 环境变量』

#### 相关参考:

- 『ALTER USER MAPPING statement』 ( *SQL Reference, Volume 2* )

## 为 Informix 表、视图和同义词注册昵称

为 Informix 表、视图和同义词注册昵称是将 Informix 数据源添加至联合服务器这一大型任务的一部分。

对于您定义的每个 Informix 服务器，为想要存取每个表、视图或同义词注册昵称。当您查询 Informix 服务器时，使用这些昵称而不是数据源对象的名称。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是您使用 CREATE NICKNAME 语句为数据源对象注册昵称时收集的。联合数据库验证数据源中的对象是否存在，然后尝试收集现有数据源的统计数据。对于优化器很有用的信息是从数据源目录中读取的，并且放置到联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息，所以，在注册昵称之前（使用与 DB2 RUNSTATS 命令等价的数据源命令）更新数据源中的统计信息。

#### 过程:

要注册昵称，发出 CREATE NICKNAME 语句。

例如:

```
CREATE NICKNAME informix_name FOR INFOSERVER."remote_schema"."remote.table"
```

昵称的长度最多可以为 128 个字符。

对于您想为其创建昵称的每个 Informix 表、视图或同义词重复此步骤。

当创建昵称时，DB2 将使用连接来查询数据源目录。此查询将使用昵称来测试与数据源连接。如果该连接不工作，则您将接收到错误消息。

#### 相关参考:

- 『RUNSTATS Command』 ( *Command Reference* )
- 『CREATE NICKNAME statement』 ( *SQL Reference, Volume 2* )
- 第 266 页的『CREATE NICKNAME 语句 - Informix 包装器的示例』

## CREATE NICKNAME 语句 - Informix 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE NICKNAME 语句来为您想存取的 Informix 表、视图或同义词注册昵称。

此示例说明了如何为在其中指定昵称的 Informix 服务器指定远程对象:

```
CREATE NICKNAME JPSALES FOR asia."salesdata"."japan"
```

*JPSALES*

用来标识 Informix 表、视图或同义词的唯一昵称。

注: 昵称是一个由两部分组成的名称 - 模式和昵称。如果在注册昵称时省略了模式，则昵称的模式将是注册该昵称的用户的授权标识。

*asia."salesdata"."japan"*

远程对象的由三部分组成的标识。

- *asia* 是您在 CREATE SERVER 语句中为 Informix 数据库服务器指定的名称。
- *salesdata* 是表、视图或同义词所属于的远程模式的名称。
- *japan* 是您想存取的远程表、视图或同义词的名称。

除非用引号将 Informix 模式和表的名称引起来，否则，联合服务器将把 Informix 模式和表的名称转变为大写。

相关任务:

- 第 266 页的『为 Informix 表、视图和同义词注册昵称』

相关参考:

- 『CREATE NICKNAME statement』 (*SQL Reference, Volume 2*)

---

## 对 Informix 数据源的配置进行调整和故障诊断

在设置 Informix 数据源的配置之后，您可能想修改该配置以提高性能。

### 通过设置 FOLD\_ID 和 FOLD\_PW 服务器选项来提高性能

当联合服务器连接至数据源时，服务器将使用用户标识和密码的大小写的所有可能存在的组合来尝试进行连接。在成功连接至数据源服务器之前，服务器最多可以尝试进行九次连接。这些尝试会减少连接次数，并且可能会导致用户标识被锁定。

过程:

要提高性能，使用 ALTER SERVER OPTION 语句来指定 FOLD\_ID 和 FOLD\_PW 服务器选项的值。

- 如果所有 Informix 用户标识和密码都是采用小写的，则将 FOLD\_ID 和 FOLD\_PW 服务器选项设置为值 'L' 可以缩短连接时间。例如:

```
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_ID 'L');
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_PW 'L');
```

- 联合服务器将尝试用户标识和密码的大写和小写值的每种组合。可以通过将这些选项设置为 'N' (不转变用户标识和密码的大小写) 可以减少超过登录尝试失败的最大次数的机会。如果建立了这些设置，则需要始终按正确的大小写指定用户标识和密码。如果指定了无效用户标识和密码，则包装器将不会继续尝试各种组合。例如:

```
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_ID 'N');
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_PW 'N');
```

相关任务:

- 第 255 页的『将 Informix 数据源添加至联合服务器』

相关参考:

- 『db2set - DB2 Profile Registry Command』 (*Command Reference*)

- 『ALTER SERVER statement』 (*SQL Reference, Volume 2*)

---

## 第 16 章 配置对 Microsoft SQL Server 数据源的存取

本章说明如何配置联合服务器以存取存储在 Microsoft SQL Server 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Microsoft SQL Server 数据源的存取。

本章列示配置联合服务器时需要执行的任务，包含您需要的 SQL 语句的示例，而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 Microsoft SQL Server 添加至联合服务器

#### 将 Microsoft SQL Server 数据源添加至联合服务器

要配置联合服务器以存取 Microsoft SQL Server 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Microsoft SQL Server 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

##### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上
- 在联合服务器上必须安装和配置了 ODBC 驱动程序。受支持的驱动程序为 Microsoft ODBC 驱动程序 (Windows) 和 DataDirect Technologies Connect for ODBC 驱动程序 (UNIX)。

##### 过程:

要将 Microsoft SQL Server 数据源添加至联合服务器:

1. 准备联合服务器和联合数据库。
  - 在 Windows 上，确认是否正确设置了“ODBC 系统 DSN”，并测试与 Microsoft SQL Server 远程服务器的连接。
  - 在 UNIX 系统上，更新或创建 odbc.ini 文件，并测试与 Microsoft SQL Server 远程服务器的连接。
2. 为 Microsoft SQL Server 包装器设置环境变量。
3. 注册包装器。
4. 注册服务器定义。
5. 创建用户映射。
6. 测试与 Microsoft SQL Server 远程服务器的连接。
7. 为 Microsoft SQL Server 表和视图注册昵称。

##### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 270 页的『准备联合服务器以存取 Microsoft SQL Server 数据源』
- 第 273 页的『注册 Microsoft SQL Server 包装器』
- 第 274 页的『为 Microsoft SQL Server 数据源注册服务器定义』
- 第 277 页的『为 Microsoft SQL Server 数据源创建用户映射』
- 第 278 页的『测试与 Microsoft SQL Server 远程服务器的连接』
- 第 279 页的『为 Microsoft SQL Server 表和视图注册昵称』
- 第 281 页的『对 Microsoft SQL Server 数据源的配置进行调整和故障诊断』
- 第 33 页的『检查联合服务器的设置』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统（32 位）』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 『DB2 Information Integrator 支持的操作系统（64 位）』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）

## 准备联合服务器以存取 Microsoft SQL Server 数据源

准备联合服务器和数据库以存取 Microsoft SQL Server 数据源是将 Microsoft SQL Server 数据源添加至联合服务器这一大型任务的一部分。

准备联合服务器以存取 Microsoft SQL Server 数据源所需要的步骤取决于联合服务器上正在运行的操作系统。

#### 过程:

要准备联合服务器:

#### 在 Windows 上:

1. 通过在“控制面板”中检查此设置来验证是否设置了“ODBC 系统 DSN”以连接至 Microsoft SQL Server 数据源。找到 Microsoft SQL Server 远程服务器的现有条目或者创建新的条目。该条目是您在联合数据库中注册服务器时将用于 NODE 服务器选项的值。
2. 从“ODBC 数据源管理器”窗口中，选择配置以测试从“ODBC 系统 DSN”至 Microsoft SQL Server 数据源的连接。另外，可以使用 Microsoft SQL Server 查询工具来测试连接。

#### 在 UNIX 上:

1. 验证联合服务器上是否更新了 `odbc.ini` 文件（或者是否需要创建）。

**建议：**将 `odbc.ini` 文件或者此文件的副本放在 DB2 实例所有者的主目录中。

2. 验证指向 `odbc.ini` 的路径是否在 ODBCINI 环境变量中。从操作系统命令提示符处发出以下命令:

```
export ODBCINI=$HOME/.odbc.ini
```

### 3. 创建适当的符号链接:

- 在 HP-UX 上, 需要创建以下符号链接:

```
ln -s $DJX_ODBC_LIBRARY_PATH/libodbcinst.sl /usr/exe/libodbcinst.sl
```

如果您正在使用 DataDirect Technologies Connect for ODBC 4.2 驱动程序, 则还必须创建以下符号链接:

```
ln -s $DJX_ODBC_LIBRARY_PATH/libivicu19.sl /ivicu/exe/libivicu19.sl
```

- 在 Linux 上, 需要创建下列符号链接:

```
ln -s $DJX_ODBC_LIBRARY_PATH/./locale /usr/local/locale
```

```
ln -s $DJX_ODBC_LIBRARY_PATH/libodbcinst.so /usr/lib/libodbcinst.so
```

如果您正在使用 DataDirect Technologies Connect for ODBC 4.2 驱动程序, 则还必须创建以下符号链接:

```
ln -s $DJX_ODBC_LIBRARY_PATH/libivicu19.so /usr/lib/libivicu19.so
```

- 在 Solaris 上, 需要创建以下符号链接:

```
ln -s $DJX_ODBC_LIBRARY_PATH/./locale $HOME/sql/lib/locale
```

\$HOME 是 DB2 实例所有者的主目录。

### 4. 使用 DataDirect Connect ODBC **demoodbc** 工具来测试从联合服务器至 Microsoft SQL Server 数据源的连接。

- a. 运行 **/opt/odbc/odbc.sh** 脚本。此脚本设置几个特定于操作系统的环境变量。
- b. 使用 DataDirect Connect ODBC **demoodbc** 工具来测试与 Microsoft SQL Server 数据源的连接。**demoodbc** 工具位于 Connect ODBC 库的 /demo 子目录中。

在这一系列任务中的下一个任务是设置 Microsoft SQL Server 环境变量。

#### 相关任务:

- 第 273 页的『注册 Microsoft SQL Server 包装器』

## 设置 Microsoft SQL Server 环境变量

设置 Microsoft SQL Server 环境变量是将 Microsoft SQL Server 添加至联合服务器这一大型任务的一部分。

当安装 DB2 Information Integrator 时, 安装过程将尝试在 db2dj.ini 文件中设置 Microsoft SQL Server 环境变量。

如果在安装 DB2 Information Integrator 之后才安装 Microsoft SQL Server ODBC 驱动程序, 则不会在 db2dj.ini 文件中设置环境变量。

Microsoft SQL Server 的有效环境变量为:

- DJX\_ODBC\_LIBRARY\_PATH
- ODBCINI
- LD\_LIBRARY\_PATH (仅适用于 Solaris)
- SHLIB\_PATH (仅适用于 HP-UX)
- DB2LIBPATH

- DB2ENVLIST

如果联合服务器在运行 HP-UX，而您具有多分区实例配置，则必须导出 `userprofile` 中的 `SHLIB_PATH` 值。

**先决条件:**

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件（例如，数据源客户机软件）。

**限制:**

请参阅主题: `db2dj.ini` 文件的限制

**过程:**

要自动设置 Microsoft SQL Server ODBC 驱动程序环境变量:

1. 如果尚未在 DB2 联合服务器上安装 Microsoft SQL Server ODBC 驱动程序，则安装它。
2. 设置必需的环境变量。可以通过再次运行 DB2 Information Integrator 安装来自动设置环境变量。单击启动板中的 **安装产品**，并遵循向导中的指示信息。

要手工设置 Microsoft SQL Server ODBC 驱动程序环境变量:

1. 编辑 `db2dj.ini` 文件。
  - 在运行 Windows 的联合服务器上，此文件位于 `sql1lib\cfg` 目录中。
  - 在运行 UNIX 的联合服务器上，此文件位于 `sql1lib/cfg` 目录中。

`db2dj.ini` 文件包含有关安装在联合服务器上的 Microsoft SQL Server ODBC 驱动程序的配置信息。如果该文件不存在，则可以使用任何文本编辑器来创建名称为 `db2dj.ini` 的新文件。在 `db2dj.ini` 文件中，必须指定变量的全限定路径，否则将遇到错误。

2. 设置下列环境变量（如果需要）:

**DJX\_ODBC\_LIBRARY\_PATH**

设置指向 ODBC 库文件的目录路径。例如:

```
DJX_ODBC_LIBRARY_PATH=ODBC_driver_directory/lib
```

`ODBC_driver_directory` 是安装 ODBC 驱动程序的目录路径。

**ODBCINI**

将 ODBCINI 环境变量设置为 ODBC 配置文件 (`odbc.ini`) 所在的目录路径。不要将 ODBCINI 环境变量设置为系统变量。例如:

```
ODBCINI=/home/db2inst1/.odbc.ini
```

**LD\_LIBRARY\_PATH**

在 Solaris 上，设置指向 ODBC 库文件的目录路径。例如:

```
LD_LIBRARY_PATH=ODBC_driver_directory/lib
```

**SHLIB\_PATH**

在 HP-UX 上，设置指向 ODBC 库文件的目录路径。例如:

```
SHLIB_PATH=ODBC_driver_directory/lib
```



3. 要存取 Microsoft SQL Server, 需要设置指向 lib 子目录中的 ODBC 库文件的目录路径。例如:

```
db2set DB2LIBPATH=ODBC_driver_directory/lib
```

4. 要使用 Connect ODBC 驱动程序来存取 Microsoft SQL Server 数据源, 将 DB2ENVLIST 设置为 LIBPATH。例如:

```
db2set DB2ENVLIST=LIBPATH
```

LIBPATH 是安装 ODBC 驱动程序的目录路径。

5. 要确保在联合服务器上设置了环境变量, 重新启动 DB2 实例。发出下列命令来重新启动 DB2 实例:

```
db2stop  
db2start
```

#### 在 HP-UX 多分区实例配置上:

1. 如果 userprofile 尚不存在, 则创建它。userprofile 位于 \$HOME/sql1lib/ 目录中。
2. 将 SHLIB\_PATH 添加至 userprofile。SHLIB\_PATH 的值是安装 ODBC 驱动程序的目录路径。例如, 发出以下命令:

```
export SHLIB_PATH=$SHLIB_PATH:/home/DataDirectODBC/lib
```

在这一系列任务中的下一个任务是注册 Microsoft SQL Server 包装器。

#### 相关任务:

- 第 279 页的『为 Microsoft SQL Server 表和视图注册昵称』

#### 相关参考:

- 第 53 页的『对于 db2dj.ini 文件的限制』
- 第 53 页的『对于 db2dj.ini 文件的限制』

## 注册 Microsoft SQL Server 包装器

注册 Microsoft SQL Server 包装器是将 Microsoft SQL Server 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Microsoft SQL Server 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器, 发出 CREATE WRAPPER 语句。

例如:

```
CREATE WRAPPER MSSQLODBC3
```

**建议:** 使用称为 MSSQLODBC3 的缺省包装器名称。当您使用缺省名称来注册包装器时, 联合服务器将自动采用与该包装器名称相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突, 则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于缺省名称的名称, 则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 `sqlserver_wrapper` 的包装器，请发出以下语句：

```
CREATE WRAPPER sqlserver_wrapper LIBRARY 'libdb2mssql3.a'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 `CREATE WRAPPER` 语句中应指定的正确名称，请参阅 Microsoft SQL Server 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Microsoft SQL Server 包装器注册服务器定义。

#### 相关任务：

- 第 274 页的『为 Microsoft SQL Server 数据源注册服务器定义』

#### 相关参考：

- 第 274 页的『Microsoft SQL Server 包装器库文件』
- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）

## Microsoft SQL Server 包装器库文件

下表列示了 Microsoft SQL Server 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 `libdb2mssql3.a`、`libdb2mssql3F.a` 和 `libdb2mssql3U.a`。

当注册包装器时，只指定该表中列示的库文件名。

表 77. Microsoft SQL Server 客户机库位置和文件名

操作系统	目录路径	包装器库文件
AIX	<code>/usr/opt/db2_08_01/lib/</code>	<code>libdb2mssql3.a</code>
HP-UX	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2mssql3.sl</code>
Linux	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2mssql3.so</code>
Solaris	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2mssql3.so</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2mssql3.dll</code>

`%DB2PATH%` 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 `C:\Program Files\IBM\SQLLIB`。

#### 相关任务：

- 第 273 页的『注册 Microsoft SQL Server 包装器』

## 为 Microsoft SQL Server 数据源注册服务器定义

为 Microsoft SQL Server 数据源注册服务器定义是将 Microsoft SQL Server 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 Microsoft SQL Server 远程服务器。必须首先找到 Microsoft SQL Server 远程服务器的节点名，然后在通过发出 `CREATE SERVER` 语句来注册服务器定义时使用此节点名。

## 过程:

要为 Microsoft SQL Server 数据源注册服务器定义:

### 1. 找到节点名。

- 如果联合服务器正在使用 Windows, 则 NODE 是您为正在访问的 Microsoft SQL Server 远程服务器指定的“系统 DSN”名称。
- 如果联合服务器正在使用 UNIX, 则 NODE 是在 .odbc.ini 文件中定义的。

以下是 AIX 上的 .odbc.ini 文件的一个示例。

### AIX 上的示例 .odbc.ini 文件:

```
rawilson=MS SQL Server 7.0
medusa=MS SQL Server 7.0
[rawilson]
Driver=/opt/odbc/lib/ivmsss16.so
Description=MS SQL Server Driver for AIX
Address=9.112.30.39,1433
[medusa]
Driver=/opt/odbc/lib/ivmsss16.so
Description=MS SQL Server Driver for AIX
Address=9.112.98.123,1433
```

在 .odbc.ini 文件的顶部有一节标记为 [ODBC Data Sources], 它列示了各个节点。每个节点都具有描述每个节点的一节 [node\_name]。

尽管节点名被指定为 CREATE SERVER 语句中的一个选项, 但是, 它对于 Microsoft SQL Server 数据源是必需的。

### 2. 发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name TYPE MSSQLSERVER VERSION 7.0 WRAPPER mssqlodbc
OPTIONS (NODE 'sqlnode', DBNAME 'mssdb');
```

在创建了服务器定义之后, 使用 ALTER SERVER 语句来添加或删除服务器选项。

在这一系列任务中的下一个任务是为 Microsoft SQL Server 数据源创建用户映射。

### 相关任务:

- 第 277 页的『为 Microsoft SQL Server 数据源创建用户映射』

### 相关参考:

- 『ALTER SERVER statement』(SQL Reference, Volume 2)
- 『CREATE SERVER statement』(SQL Reference, Volume 2)
- 第 529 页的附录 D, 『联合系统的服务器选项』
- 第 275 页的『CREATE SERVER 语句 - Microsoft SQL Server 包装器的示例』

## CREATE SERVER 语句 - Microsoft SQL Server 包装器的示例

此主题提供了几个示例来说明如何使用 CREATE SERVER 语句来为 Microsoft SQL Server 包装器注册服务器。此主题包括一个完整的示例, 该示例说明如何使用必需的参数来注册服务器, 它还包括一个带有更多服务器选项的示例。

### 完整示例:

以下示例说明了如何通过发出 CREATE SERVER 语句来为 Microsoft SQL Server 包装器注册服务器定义:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 7.0 WRAPPER mssqlodbc3
    OPTIONS (NODE 'sqlnode', DBNAME 'africa');
```

#### *sqlserver*

为 Microsoft SQL Server 远程服务器指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

#### **TYPE** *MSSQLSERVER*

正在对其存取权进行配置的数据源的类型。Microsoft SQL Server 包装器的 TYPE 参数必须为 *MSSQLSERVER*。

#### **VERSION** *7.0*

您想访问的 Microsoft SQL Server 数据库服务器软件版本。受支持的版本为 6.5、7.0 和 2000。

#### **WRAPPER** *mssqlodbc3*

在 CREATE WRAPPER 语句中指定的包装器名称。

#### **NODE** *'sqlnode'*

在 Windows 上, 这是您正在访问的 Microsoft SQL Server 远程服务器的“系统 DSN”名称。在 UNIX 上, 这是在 .odbc.ini 文件中定义的节点。此值是区分大小写的。

尽管节点的名称被指定为 CREATE SERVER 语句中的一个选项, 但是, 它对于 Microsoft SQL Server 数据源是必需的。

#### **DBNAME** *'africa'*

想要存取的数据库的昵称。此值是区分大小写的。

尽管数据库的名称被指定为 CREATE SERVER 语句中的一个选项, 但是, 它对于 Microsoft SQL Server 数据源是必需的。

### 服务器选项示例:

当您注册服务器时, 可以在 CREATE SERVER 语句中指定更多服务器选项。这些服务器选项包括一般的服务器选项和特定于 Microsoft SQL Server 的服务器选项。

以下示例说明了如何使用 COLLATING\_SEQUENCE 服务器选项:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER
    VERSION 7.0
    WRAPPER mssqlodbc3
    OPTIONS (NODE 'sqlnode', DBNAME 'africa', COLLATING_SEQUENCE 'I');
```

COLLATING\_SEQUENCE 服务器选项指定数据源是否与联合服务器使用相同的整理顺序。在正在运行 Windows NT 或 Windows 2000 的 Microsoft SQL Server 数据库服务器上, 缺省整理顺序是不区分大小写的 (例如, 'STEWART' 和 'StewART' 被认为是相同的)。为了保证联合服务器中获得正确的结果, 将 COLLATING\_SEQUENCE 服务器选项设置为 'I'。此设置指示 Microsoft SQL Server 数据源是不区分大小写的。

如果从数据源返回的结果将不同于在联合服务器中处理查询时所返回的结果，则联合服务器将不会下推查询。当将 `COLLATING_SEQUENCE` 服务器选项设置为 'I' 时，则联合服务器不会下推具有字符串数据或表达式并且包括下列子句、谓词或函数的查询：

- `GROUP BY` 子句
- `DISTINCT` 子句
- 基本谓词，例如，等号 (=)
- 聚集函数，例如，`MIN` 或 `MAX`

相关任务：

- 第 274 页的『为 Microsoft SQL Server 数据源注册服务器定义』

相关参考：

- 『`CREATE SERVER` statement』 (*SQL Reference, Volume 2*)

## 为 Microsoft SQL Server 数据源创建用户映射

为 Microsoft SQL Server 数据源创建用户映射是将 Microsoft SQL Server 数据源添加至联合服务器这一大型任务的一部分。

当您尝试存取 Microsoft SQL Server 数据源时，联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联（用户映射）。为将访问联合系统以便发送分布式请求的每个用户标识创建用户映射。

过程：

要将本地用户标识映射至 Microsoft SQL Server 远程服务器用户标识和密码，发出 `CREATE USER MAPPING` 语句。

例如：

```
CREATE USER MAPPING FOR userid SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

在这一系列任务中的下一个任务是测试与 Microsoft SQL Server 远程服务器的连接。

相关任务：

- 第 278 页的『测试与 Microsoft SQL Server 远程服务器的连接』

相关参考：

- 『`CREATE USER MAPPING` statement』 (*SQL Reference, Volume 2*)
- 第 278 页的『`CREATE USER MAPPING` 语句 - Microsoft SQL Server 包装器的示例』

## CREATE USER MAPPING 语句 - Microsoft SQL Server 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将联合服务器用户标识映射至 Microsoft SQL Server 远程服务器用户标识和密码。此主题包括一个具有必需参数的完整示例，还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

### 完整示例:

以下示例说明了如何将联合服务器用户标识映射至 Microsoft SQL Server 远程服务器用户标识和密码:

```
CREATE USER MAPPING FOR elizabeth SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123')
```

#### *elizabeth*

指定您正在映射至在 Microsoft SQL Server 远程服务器中定义的用户标识的本地用户标识。

#### SERVER *sqlserver*

指定您在 CREATE SERVER 语句中定义的 Microsoft SQL Server 远程服务器的名称。

#### REMOTE\_AUTHID '*liz*'

指定您正在将 *elizabeth* 映射至的 Microsoft SQL Server 远程服务器中的用户标识。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_ID 服务器选项设置为 'U' 或 'L'）。

#### REMOTE\_PASSWORD '*abc123*'

指定与 '*liz*' 相关联的密码。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_PW 服务器选项设置为 'U' 或 'L'）。

### 专用寄存器示例:

以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

### 相关任务:

- 第 277 页的『为 Microsoft SQL Server 数据源创建用户映射』

### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）

## 测试与 Microsoft SQL Server 远程服务器的连接

测试与 Microsoft SQL Server 远程服务器的连接是将 Microsoft SQL Server 数据源添加至联合服务器这一大型任务的一部分。

可以使用您定义的服务器定义和用户映射来测试与 Microsoft SQL Server 远程服务器的连接。

### 过程:

要测试连接:

1. 打开传递会话以便对 Microsoft SQL Server 系统表发出 SQL SELECT 语句。

例如:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

如果该 SQL SELECT 语句返回了一个计数,则表示正确设置了服务器定义和用户映射。

2. 如果 SQL SELECT 语句返回了错误,则可能需要:
  - 检查 Microsoft SQL Server 远程服务器以确保它已启动。
  - 检查 Microsoft SQL Server 远程服务器以确保为入局连接配置了 Microsoft SQL Server 远程服务器。
  - 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 Microsoft SQL Server 远程服务器的连接是有效的。在需要时,改变用户映射,或者创建另一个用户映射。
  - 检查 DB2 联合服务器上的 ODBC 驱动程序,以确保正确安装和配置了这些驱动程序以便连接至 Microsoft SQL Server 远程服务器。
  - 检查 DB2 联合变量的设置以验证它们对于 Microsoft SQL Server 远程服务器是否正确。这些变量包括系统环境变量、db2dj.ini 变量和“DB2 概要文件注册表”(db2set)变量。
  - 检查服务器定义。如果需要的话,删除服务器定义,然后再次创建它。

在这一系列任务中的下一个任务是为 Microsoft SQL Server 表和视图注册昵称。

### 相关任务:

- 第 269 页的『将 Microsoft SQL Server 数据源添加至联合服务器』
- 第 279 页的『为 Microsoft SQL Server 表和视图注册昵称』
- 第 271 页的『设置 Microsoft SQL Server 环境变量』

### 相关参考:

- 『ALTER USER MAPPING statement』(SQL Reference, Volume 2)

## 为 Microsoft SQL Server 表和视图注册昵称

为 Microsoft SQL Server 表和视图注册昵称是将 Microsoft SQL Server 数据源添加至联合服务器这一大型任务的一部分。

对于您定义的每个 Microsoft SQL Server 远程服务器,为想要存取每个表或视图注册昵称。当您查询 Microsoft SQL Server 远程服务器时,使用这些昵称而不是数据源对象的名称。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是当您使用 `CREATE NICKNAME` 语句为数据源对象注册昵称时收集的。联合数据库验证数据源中的对象是否存在，然后尝试收集现有数据源的统计数据。对于优化器很有用的信息是从数据源目录中读取的，并且放置到联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息，所以，在注册昵称之前（使用与 `DB2 RUNSTATS` 命令等价的数据源命令）更新数据源中的统计信息。

#### 过程:

要注册昵称，发出 `CREATE NICKNAME` 语句。

例如:

```
CREATE NICKNAME mss_name FOR sqlserver."remote_schema"."remote.table"
```

昵称的长度最多可以为 128 个字符。

对于您想为其创建昵称的每个 Microsoft SQL Server 表或视图重复此步骤。

当创建昵称时，DB2 使用连接来查询数据源目录表（Microsoft SQL Server 将这些表称为系统表）。此查询将使用昵称来测试与数据源的连接。如果该连接不工作，则您将接收到错误消息。

#### 相关参考:

- 『`RUNSTATS` Command』 (*Command Reference*)
- 『`CREATE NICKNAME` statement』 (*SQL Reference, Volume 2*)
- 第 280 页的『`CREATE NICKNAME` 语句 - Microsoft SQL Server 包装器的示例』

## CREATE NICKNAME 语句 - Microsoft SQL Server 包装器的示例

此主题提供了一个示例来说明如何使用 `CREATE NICKNAME` 语句来为您想存取的 Microsoft SQL Server 表或视图注册昵称。

此示例说明了如何为在其中指定昵称的 Microsoft SQL Server 远程服务器指定远程对象:

```
CREATE NICKNAME cust_africa FOR sqlserver.customers.egypt
```

*cust\_africa*

Microsoft SQL Server 表或视图的唯一昵称。

**注:** 昵称是一个由两部分组成的名称，它包括模式和昵称。如果在注册昵称时省略了模式，则昵称的模式将为创建该昵称的用户的认证标识。

*sqlserver.customers.egypt*

远程对象的由三部分组成的标识。

- *sqlserver* 是您在 `CREATE SERVER` 语句中为 Microsoft SQL Server 数据库服务器指定的名称。
- *customers* 是表或视图所属于的远程模式的名称。
- *egypt* 是您想存取的远程表或视图的名称。



除非将 Microsoft SQL Server 模式和表的名称用引号引起来，否则，联合服务器将把这些名称转变为大写。

**相关任务:**

- 第 279 页的『为 Microsoft SQL Server 表和视图注册昵称』

**相关参考:**

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）

---

## 对 Microsoft SQL Server 数据源的配置进行调整和故障诊断

在设置 Microsoft SQL Server 数据源的配置之后，您可能想修改该配置以提高性能。

### 获取 ODBC 跟踪

如果您在存取数据源时遇到了问题，则可以获取 ODBC 跟踪信息来分析和解决问题。但是，激活跟踪将影响系统性能。在解决问题之后应关闭跟踪。

在运行 Windows 的联合服务器上，使用由“ODBC 数据源管理器”提供的跟踪工具来确保 ODBC 跟踪正确工作。

在运行 UNIX 的联合服务器上，可以通过更改 `odbc.ini` 文件来将跟踪设置为“打开”。例如，如果使用 DataDirect ODBC 3.x 驱动程序，则在客户机目录中查找 `odbc.ini` 文件的示例。`odbc.ini` 文件包含跟踪文件所需要的内容的样本：

```
[ODBC]
Trace=0
TraceFile=/home/user1/trace_dir/filename.xxx
TraceDll==ODBC_driver_directory/odbctrac.so
InstallDir=/opt/odbc
```

要打开跟踪，应将第一行设置为 `Trace=1`。要关闭跟踪，应将第一行设置为 `Trace=0`。`TraceFile` 应当指向联合数据库实例对其具有写存取权的路径和文件名。

**相关任务:**

- 第 269 页的『将 Microsoft SQL Server 数据源添加至联合服务器』

**相关参考:**

- 『db2set - DB2 Profile Registry Command』（*Command Reference*）



---

## 第 17 章 配置对 ODBC 数据源的存取

本章说明如何配置联合服务器以存取存储在 ODBC 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 ODBC 数据源的存取。

本章列示配置联合服务器时需要执行的任务，包含您需要的 SQL 语句的示例，而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 ODBC 添加至联合系统

#### 将 ODBC 数据源添加至联合服务器

要配置联合服务器以存取 ODBC 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 ODBC 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

通过 ODBC API 存取的数据源在本文中称作 ODBC 数据源。

根据需要，您可以使用 ODBC 包装器来存取 Excel 数据而不必使用 Excel 包装器。在单独的主题中说明了配置 ODBC 包装器以存取 Excel 数据的特定步骤。

#### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上
- 在联合服务器上必须安装和配置了 ODBC 驱动程序。ODBC 包装器支持 ODBC 3.x。
- 已正确设置了系统环境变量、db2dj.ini 变量和 DB2 概要文件注册表 (db2set) 变量。检查供应商文档以了解 ODBC 客户机必需的变量。LIBPATH 变量可能是必需的。

#### 限制:

- ODBC 包装器不能用来存取任何 DB2 系列的数据源。请使用 DRDA 包装器来存取 DB2 系列数据源。
- ODBC 包装器不支持下列功能和语句:
  - 有关昵称的 LOCK TABLE 语句
  - ODBC 3.x 中不推荐使用的功能
  - X/Open 或 SQL/CLI 驱动程序
  - 存储过程昵称
  - 使用远程保存点语句来实施语句级别可分性
  - 64 位客户机
  - WITH HOLD 游标

- 对于不支持已定位的更新和删除操作的数据源，如果在昵称或者它的相应远程表上不存在不可空列的唯一索引，则针对该昵称的已定位的 UPDATE 和 DELETE 语句及某些已搜索的 UPDATE 和 DELETE 语句将失败。当这些语句失败时，将返回错误 SQL30090，原因码为 21。
- ODBC 包装器不支持对限制每个连接的活动语句数的数据源执行 INSERT、UPDATE 或 DELETE 语句。参阅有关数据源的文档以便确定该数据源是否限制每个连接的活动语句数。此限制适用于 IBM Red Brick Warehouse。

#### 过程:

要将 ODBC 数据源添加至联合服务器:

1. 准备联合服务器和联合数据库。
2. 注册包装器。
3. 注册服务器定义。
4. 创建用户映射。
5. 测试与 ODBC 数据源的连接。
6. 为 ODBC 数据源表和视图注册昵称。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 ( 《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》 )
- 第 49 页的 『配置数据源的快速方法』
- 第 22 页的 『用来存取 Excel 数据的方法』

#### 相关任务:

- 第 293 页的 『使用 ODBC 包装器存取 Excel 数据』
- 第 44 页的 『创建联合数据库』
- 第 284 页的 『准备联合服务器以通过 ODBC 来存取数据源』
- 第 285 页的 『注册 ODBC 包装器』
- 第 287 页的 『为 ODBC 数据源注册服务器定义』
- 第 289 页的 『为 ODBC 数据源创建用户映射』
- 第 290 页的 『测试与 ODBC 数据源服务器的连接』
- 第 291 页的 『为 ODBC 数据源表和视图注册昵称』

#### 相关参考:

- 第 22 页的 『选择正确的包装器』

## 准备联合服务器以通过 ODBC 来存取数据源

准备联合服务器以通过 ODBC 来存取数据源是将 ODBC 数据源添加至联合服务器这一大型任务的一部分。

准备联合服务器以通过 ODBC 来存取数据源所需要遵循的步骤取决于联合服务器上正在运行的操作系统。

**注意:** 您使用的 ODBC 驱动程序和操作系统具有唯一的库路径位置。

#### 过程:

要准备联合服务器:

#### 在 Windows 上:

验证在联合服务器上是否已经安装和配置了 ODBC 3.x 驱动程序。必须在“系统 DSN”中定义 ODBC 数据源的节点名。请参阅 ODBC 驱动程序文档，了解安装和配置过程。

如果使用了 Microsoft ODBC 数据源管理器来配置 DSN，您就可以在“控制面板”中检查此设置。请确保 ODBC 数据源已作为“系统 DSN”注册。否则，DB2 可能会找不到该 DSN。

#### 在 UNIX 上:

1. 参阅 ODBC 客户机供应商提供的文档，以获取有关如何配置 ODBC 客户机的指示信息。
2. 如果客户机是 DataDirect ODBC 或 RedBrick，则验证是否创建了适当的符号链接：
  - 在 HP-UX 上，符号链接是从 `/usr/exe/libodbcinst.sl` 到 `$ODBC_LIBRARY_PATH/libodbcinst.sl`。
  - 在 Linux 上，符号链接是从 `/usr/lib/libodbcinst.so` 到 `$DJX_ODBC_LIBRARY_PATH/libodbcinst.so`。
  - 在 Solaris 上，符号链接是从 `$HOME/sql1lib/locale` 到 `$DJX_ODBC_LIBRARY_PATH/./locale`。\$HOME 是 DB2 实例所有者的主目录。

在这一系列任务中的下一个任务是注册 ODBC 包装器。

#### 相关任务:

- 第 285 页的『注册 ODBC 包装器』

## 注册 ODBC 包装器

注册 ODBC 包装器是将 ODBC 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 ODBC 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器，发出 `CREATE WRAPPER` 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER ODBC
```

**建议:** 使用称为 ODBC 的缺省包装器名称。当您注册使用缺省名称的包装器时，联合服务器将自动采用与该包装器相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突，则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于缺省名称的名称，则在 `CREATE WRAPPER` 语句中必须包括 `LIBRARY` 参数。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 `odbc_wrapper` 的包装器，请发出以下语句:

```
CREATE WRAPPER odbc_wrapper
  LIBRARY 'libdb2rcodbc.a' OPTIONS (MODULE '/usr/lib/odbc.a')

```

MODULE '*/usr/lib/odbc.a*' 是包含“ODBC 驱动程序管理器”的库的全路径。

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 ODBC 包装器库文件的列表。

在这一系列任务中的下一个任务是为 ODBC 包装器注册服务器定义。

**相关任务:**

- 第 287 页的『为 ODBC 数据源注册服务器定义』

**相关参考:**

- 第 286 页的『ODBC 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)
- 第 286 页的『CREATE WRAPPER 语句 - ODBC 包装器的示例』

## ODBC 包装器库文件

下表列示了 ODBC 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2rcodbc.a、libdb2rcodbcF.a 和 libdb2rcodbcU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 78. ODBC 客户机库位置和文件名

操作系统	目录路径	包装器库文件
AIX	<i>/usr/opt/db2_08_01/lib/</i>	<i>libdb2rcodbc.a</i>
HP-UX	<i>/opt/IBM/db2/V8.1/lib</i>	<i>libdb2rcodbc.sl</i>
Linux	<i>/opt/IBM/db2/V8.1/lib</i>	<i>libdb2rcodbc.so</i>
Solaris	<i>/opt/IBM/db2/V8.1/lib</i>	<i>libdb2rcodbc.so</i>
Windows	<i>%DB2PATH%\bin</i>	<i>db2rcodbc.dll</i>

*%DB2PATH%* 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 *C:\Program Files\IBM\SQLLIB*。

**相关任务:**

- 第 285 页的『注册 ODBC 包装器』

## CREATE WRAPPER 语句 - ODBC 包装器的示例

此主题提供了几个示例来说明如何使用 CREATE WRAPPER 语句来为 ODBC 数据源注册包装器。

用于 UNIX 系统的示例:

以下示例说明了在 UNIX 操作系统上如何通过发出 CREATE WRAPPER 语句来注册包装器:

```
CREATE WRAPPER odbc OPTIONS (MODULE '/usr/lib/odbc.so')
```

在此示例中, *odbc* 是为正在联合数据库中注册的包装器指定的名称。MODULE '/usr/lib/odbc.so' 是包含“ODBC 驱动程序管理器”的库的全路径。

在 UNIX 操作系统上必须指定 MODULE 选项。在 Windows 上, MODULE 选项被缺省设置为 '*odbc32.dll*'。

用于 **Windows** 的示例:

以下示例说明了在 Windows 操作系统上如何通过发出 CREATE WRAPPER 语句来注册包装器:

```
CREATE WRAPPER odbc LIBRARY 'db2rcodbc.dll'
```

在此示例中, *odbc* 是为正在联合数据库中注册的包装器指定的名称。LIBRARY '*db2rcodbc.dll*' 是 ODBC 包装器的库名。

相关任务:

- 第 285 页的『注册 ODBC 包装器』

相关参考:

- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## 为 ODBC 数据源注册服务器定义

为 ODBC 数据源注册服务器定义是将 ODBC 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中, 必须定义想要访问的每个 ODBC 数据源服务器。

过程:

要为 ODBC 数据源注册服务器定义:

发出 CREATE SERVER 语句。

例如:

```
| CREATE SERVER server_name TYPE data_source_type  
|             VERSION version WRAPPER odbc  
|             OPTIONS (NODE 'node_name')  
|
```

TYPE 和 VERSION 参数是可选的。

尽管 NODE 被指定为 CREATE SERVER 语句中的一个选项, 但对于 ODBC 数据源来说它是必需的。

在创建了服务器定义之后, 可使用 ALTER SERVER 语句来添加或删除服务器选项。

在这一系列任务中的下一个任务是为 ODBC 数据源创建用户映射。

相关任务:

- 第 289 页的『为 ODBC 数据源创建用户映射』

#### 相关参考:

- 『ALTER SERVER statement』 (*SQL Reference, Volume 2*)
- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 529 页的附录 D, 『联合系统的服务器选项』
- 第 288 页的『CREATE SERVER 语句 - ODBC 包装器的示例』

## CREATE SERVER 语句 - ODBC 包装器的示例

此主题提供了一些示例, 说明如何使用 CREATE SERVER 语句来为 ODBC 包装器注册服务器。此主题包括一个完整的示例, 说明如何使用必需的参数来注册服务器, 还包括一个具有更多服务器选项的示例。

#### 完整示例:

以下示例说明了如何通过发出 CREATE SERVER 语句来为 ODBC 包装器注册服务器定义:

```
CREATE SERVER mysql_server TYPE mysql
          VERSION 4.0 WRAPPER odbc
          OPTIONS (NODE 'odbc_node', DBNAME 'venice')
|
```

#### *mysql\_server*

为 ODBC 数据源服务器指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

#### TYPE *mysql*

正在对其配置存取权的数据源的类型。此参数是可选的。

#### VERSION 4.0

您想存取的 ODBC 数据源的版本。此参数是可选的。

#### WRAPPER *odbc*

在 CREATE WRAPPER 语句中指定的包装器名称。

#### NODE '*mysql\_node*'

定义 DSN 时为 ODBC 数据源指定的节点的名称 (系统 DSN 名称)。此值是区分大小写的。在 Windows 上, 此值必须是“ODBC 数据管理”窗口中的系统 DSN 的名称。在 UNIX 上, 参阅 ODBC 客户机供应商文档, 以获取有关要使用的值的信息。

尽管 NODE 被指定为 CREATE SERVER 语句中的一个选项, 但是, 它对于 ODBC 数据源是必需的。

#### DBNAME '*venice*'

想要存取的数据库的名称。此值是区分大小写的。

#### 服务器选项示例:

以下示例显示如何使用 DB2\_TABLE\_QUOTE\_CHAR、DB2\_ID\_QUOTE\_CHAR 和 DB2\_AUTHID\_QUOTE\_CHAR 服务器选项。

某些 ODBC 数据源 (例如, MySQL) 不能处理 SQL 语句中表名和列名两边的引号。要存取这些数据源, 必须在 CREATE SERVER 语句中包括下列服务器选项:



- DB2\_TABLE\_QUOTE\_CHAR ' ` ' ,
- DB2\_ID\_QUOTE\_CHAR ' ` ' ,
- DB2\_AUTHID\_QUOTE\_CHAR ' ` ' ,

` 字符是标识（例如，模式名、表名和列名）的定界符。

例如:

```
CREATE SERVER mysql_server TYPE mysql
    VERSION 4.0 WRAPPER odbc
    OPTIONS (NODE 'mysql_node', DB2_TABLE_QUOTE_CHAR ' ` ',
            DB2_ID_QUOTE_CHAR ' ` ' DB2_AUTHID_QUOTE_CHAR ' ` ')
```

相关任务:

- 第 287 页的『为 ODBC 数据源注册服务器定义』

相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)

## 为 ODBC 数据源创建用户映射

为 ODBC 数据源创建用户映射是将 ODBC 数据源添加至联合服务器这一大型任务的一部分。

当您尝试存取 ODBC 数据源时，联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联（用户映射）。为将访问联合系统以便发送分布式请求的每个用户标识创建用户映射。

过程:

要将本地用户标识映射至 ODBC 数据源用户标识和密码，发出 CREATE USER MAPPING 语句。

例如:

```
CREATE USER MAPPING FOR userid SERVER server_name
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

在这一系列任务中的下一个任务是测试与 ODBC 数据源服务器的连接。

相关任务:

- 第 290 页的『测试与 ODBC 数据源服务器的连接』

相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)
- 第 290 页的『CREATE USER MAPPING 语句 - ODBC 包装器的示例』

## CREATE USER MAPPING 语句 - ODBC 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将本地用户标识映射至 ODBC 数据源用户标识和密码。此主题包括一个具有必需参数的完整示例，还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

### 完整示例:

以下示例说明了如何将本地用户标识映射至 ODBC 数据源用户标识和密码:

```
CREATE USER MAPPING FOR arturo SERVER server_name
    OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue')
```

*arturo* 指定您正在映射至在 ODBC 数据源中定义的用户标识的本地用户标识。

*server\_name*

指定您在 CREATE SERVER 语句中定义的 ODBC 数据源的名称。

*'art'* 指定您正在将 *arturo* 映射至的 ODBC 数据源中的用户标识。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_ID 服务器选项设置为 'U' 或 'L'）。

*'red4blue'*

指定与 *'art'* 相关联的密码。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_PW 服务器选项设置为 'U' 或 'L'）。

### 专用寄存器示例:

以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER server_name
    OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

### 相关任务:

- 第 289 页的『为 ODBC 数据源创建用户映射』

### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）

## 测试与 ODBC 数据源服务器的连接

测试与 ODBC 数据源服务器的连接是将 ODBC 数据源添加至联合服务器这一大型任务的一部分。

可以使用您定义的服务器定义和用户映射来测试与 ODBC 数据源服务器的连接。

### 先决条件:

正在使用的数据源必须支持传递会话。

### 过程:

要测试连接:

1. 打开传递会话以便对 ODBC 数据源系统表发出 SQL SELECT 语句。

例如:

```
SET PASSTHRU server_name
SELECT COUNT(*) FROM schema_name.table_name
SET PASSTHRU RESET
```

*server\_name* 是您在 CREATE SERVER 语句中定义的 ODBC 数据源的名称。

*schema\_name* 是远程 ODBC 数据源中的模式的名称。如果 ODBC 数据源不支持该模式, 则从语句中省略该模式。

*table\_name* 是远程 ODBC 数据源中的表的名称。

如果 SQL SELECT 语句返回了一个计数, 则表示正确设置了服务器定义和用户映射。

2. 如果 SQL SELECT 语句返回了错误, 则可能需要:

- 验证数据源是否可用。
- 如果可用的话, 则检查数据源服务器以确保为入局连接配置了该数据源服务器。
- 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 ODBC 数据源的连接是有效的。在需要时, 改变用户映射, 或者创建另一个用户映射。
- 检查 DB2 联合服务器上的 ODBC 驱动程序, 以确保正确安装和配置了该驱动程序以便连接至 ODBC 数据源服务器。在 Windows 操作系统上, 使用“ODBC 数据源管理器”工具来检查驱动程序。在 UNIX 操作系统上, 参阅 ODBC 客户机供应商的文档。
- 检查服务器定义。如果需要的话, 删除服务器定义, 然后再次创建它。

在这一系列任务中的下一个任务是为 ODBC 数据源表和视图注册昵称。

相关任务:

- 第 283 页的『将 ODBC 数据源添加至联合服务器』
- 第 284 页的『准备联合服务器以通过 ODBC 来存取数据源』
- 第 291 页的『为 ODBC 数据源表和视图注册昵称』

相关参考:

- 『ALTER USER MAPPING statement』 (SQL Reference, Volume 2)

## 为 ODBC 数据源表和视图注册昵称

为 ODBC 数据源表和视图注册昵称是将 ODBC 数据源添加至联合服务器这一大型任务的一部分。

对于您定义的每个 ODBC 数据源服务器, 为想要存取的每个表或视图注册昵称。当您查询 ODBC 数据源时, 使用这些昵称而不是数据源对象的名称。

例如，假定您定义昵称 *cust\_europe* 来表示称为 *italy* 的 Microsoft SQL Server 表，该表的模式名为 *customers*。联合服务器中允许使用 SQL 语句 `SELECT * FROM cust_europe`。但是，不允许使用语句 `SELECT * FROM server_name."customers"."italy"`。

除了为 ODBC 数据源表和视图注册昵称之外，还可以为远程系统表注册昵称。

如果 ODBC 数据源不支持该模式，则从 `CREATE NICKNAME` 语句中省略该模式。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是您使用 `CREATE NICKNAME` 语句为数据源对象注册昵称时收集的。联合数据库验证数据源中的对象是否存在，然后尝试收集现有数据源的统计数据。对于优化器很有用的信息是从数据源目录中读取的，并且放置到联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息，所以，在注册昵称之前（使用与 `DB2 RUNSTATS` 命令等价的数据源命令）更新数据源中的统计信息。

#### 限制:

当对包含索引的 ODBC 数据源创建昵称时，ODBC 包装器不会将索引信息记录在联合数据库系统目录中。必须通过将 `CREATE INDEX` 语句与 `SPECIFICATION ONLY` 子句配合使用来为表创建索引规范。

#### 过程:

要注册昵称，发出 `CREATE NICKNAME` 语句。

例如:

```
CREATE NICKNAME odbc_name FOR server_name."remote_schema"."remote.table"
```

昵称的长度最多可以为 128 个字符。

对于您想为其创建昵称的每个 ODBC 表或视图重复此步骤。

当创建昵称时，DB2 将使用连接来查询数据源目录表。此查询将使用昵称来测试与 ODBC 数据源的连接。如果该连接不工作，则您将接收到错误消息。

#### 相关任务:

- 『为数据源对象创建索引规范』（《联合系统指南》）

#### 相关参考:

- 『RUNSTATS Command』（*Command Reference*）
- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 292 页的『CREATE NICKNAME 语句 - ODBC 包装器的示例』

## CREATE NICKNAME 语句 - ODBC 包装器的示例

此主题提供了一个示例来说明如何使用 `CREATE NICKNAME` 语句来为您想存取的 ODBC 数据源表或视图注册昵称。

此示例说明了如何为在其中指定昵称的 ODBC 数据源指定远程对象:

```
CREATE NICKNAME cust_europe FOR server_name."customers"."italy"
```

*cust\_europe*

表或视图的唯一昵称。昵称在模式中必须是唯一的。

**注：**昵称是一个由两部分组成的名称，它包括模式和昵称。如果在注册昵称时省略了模式，则昵称的模式将是注册该昵称的用户的认证标识。

*server\_name."customers"."italy"*

远程对象的由三部分组成的标识。

- *server\_name* 是您在 CREATE SERVER 语句中为 ODBC 数据库服务器指定的名称。
- *customers* 是表或视图所属于的远程模式的名称。如果 ODBC 数据源不支持该模式，则从 CREATE NICKNAME 语句中省略该模式。
- *italy* 是您想存取的远程表或视图的名称。

ODBC 数据源对象可能是区分大小写的。将远程模式名和远程表名都用引号引起来。否则，DB2 会将这些名称转变为大写。

**相关任务：**

- 第 291 页的『为 ODBC 数据源表和视图注册昵称』

**相关参考：**

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）

---

## 使用 ODBC 包装器存取 Excel 数据

通过使用 Excel ODBC 驱动程序，可以用 ODBC 包装器存取 Excel 工作簿。Excel 应用程序不必安装在联合服务器上。Excel ODBC 驱动程序会自动随 Microsoft Windows 一起安装。

借助于 ODBC 包装器和 Excel ODBC 驱动程序，可以从工作簿内的任何工作表存取数据。Excel ODBC 驱动程序将工作簿解释为数据库，而将工作簿内的每一个工作表解释为表。

即使生成工作簿的 Excel 应用程序的版本不再受支持，Excel ODBC 驱动程序也能支持 Excel 工作簿的先前版本。例如，Microsoft 不再支持在 Excel 版本 4.0 中创建的工作表，但是驱动程序支持在该版本中创建的 Excel 工作表。

**先决条件：**

Excel ODBC 驱动程序必须在联合服务器上。

联合服务器必须能够打开和读取 Excel 工作簿中的工作表才能检索数据。因此，Excel 工作簿必须处于 DB2 联合服务器所在的那台计算机上，或处于可访问的映射网络驱动器上。

**限制：**

- 当工作簿已由用户或应用程序以读 / 写方式打开时，ODBC 包装器不能存取工作表。但是，如果 ODBC 包装器在用户或应用程序打开工作簿之前已打开该工作簿，则用户或应用程序只能以只读方式打开该工作簿。

- Excel ODBC 驱动程序期望第一个非空白行包含工作表列的标题。如果工作表没有标题，必须在工作表中插入一行列标题。
- 因为 Excel ODBC 驱动程序只可用于 Windows 操作系统，所以只可在运行 Windows 操作系统的联合服务器上使用 ODBC 包装器来存取 Excel 数据。
- 可以对 Excel 工作表执行插入和更新操作，但不能执行删除操作。Excel ODBC 驱动程序不支持删除操作。要从工作表删除数据，必须在 Excel 中打开工作表来进行更改。

#### 过程:

要使用 ODBC 包装器存取 Excel 工作表:

1. 确保您想存取的 Excel 工作簿在联合服务器上或可访问的映射网络驱动器上。
2. 存取 Excel 数据的方法:
  - 如果有必要，可更改 Excel 工作表中数据的布局来满足 Excel ODBC 驱动程序的需求。
  - 如果有必要，可创建您想存取的任何命名区域。
3. 为您想存取的工作簿创建系统 DSN。可以使用 ODBC 数据源管理器来配置系统 DSN。在您创建系统 DSN 时指定的名称被指定为 CREATE SERVER 语句中 NODE 选项的值。
4. 发出 CREATE WRAPPER 语句。

例如:

```
CREATE WRAPPER odbc
```

5. 通过在联合数据库系统目录中注册服务器对象来指定工作簿的位置。对于 ODBC 包装器，每个 DSN 都需要一个服务器对象。使用 Excel ODBC 驱动程序时，DSN 与工作簿相关联。

要指定工作簿的位置，发出 CREATE SERVER 语句并将该 DSN 用作 NODE 选项的系统 DSN。

例如:

```
CREATE SERVER compounds_workbook WRAPPER odbc
  OPTIONS (NODE 'compounds_workbook_dsn', PASSWORD 'n')
```

NODE *compounds\_workbook\_dsn* 是您创建的系统 DSN。需要有 NODE 和 PASSWORD 选项，ODBC 包装器才能存取 Excel 工作表。

对要存取的每个工作簿重复此步骤。

6. 发出 CREATE NICKNAME 语句为您想存取的工作表创建昵称。语法是:

```
CREATE NICKNAME nickname FOR server_name.remote_table
```

- 如果您创建了命名区域来存取数据，则将该区域的名称指定为 CREATE NICKNAME 语句的 remote\_table 部分。

例如，如果该区域的名称是 *testing*，则 CREATE NICKNAME 语句为:

```
CREATE NICKNAME compounds_nickname FOR compounds_workbook.testing
```

- 要存取整个工作表中的数据而不是某一区域中的数据，指定工作表名称，后跟 \$ 符号。

例如，如果工作表的名称是 *Sheet1*，则 CREATE NICKNAME 语句为：

```
CREATE NICKNAME compounds_nick FOR compounds_workbook.Sheet1$
```

对您想存取的工作表或命名区域重复此步骤。

## 改变缺省数据类型映射

使用 ODBC 包装器时，受支持的数据类型是由 Excel ODBC 驱动程序确定的。Excel ODBC 驱动程序将 Excel 数据类型映射至 ODBC 数据类型。然后，ODBC 包装器将 ODBC 数据类型映射至 DB2 数据类型。每列的 DB2 数据类型存储在联合数据库目录表中。这些数据类型称为本地数据类型。下表列出了缺省数据类型映射：

表 79. 使用 ODBC 包装器时 Excel 与 DB2 之间的缺省数据类型映射

Excel 数据类型	ODBC 数据类型	DB2 数据类型
CURRENCY	SQL_NUMERIC	DECIMAL/DOUBLE
DATETIME	SQL_TIMESTAMP	TIMESTAMP
LOGICAL	SQL_BIT	SMALLINT
NUMBER	SQL_DOUBLE	DOUBLE
TEXT	SQL_VARCHAR	VARCHAR

但是，ODBC 包装器支持许多不同的数据类型。您可以将缺省数据类型映射至其它 DB2 数据类型。使用 ALTER NICKNAME 语句来更改本地数据类型。可通过 ODBC 包装器使用的数据类型的列表很全面，它包括诸如 LOB 之类的数据类型以及其它双字节数据类型。

**注意：**当将本地数据类型改变为除原始映射之外的某种类型时，有可能会遇到数据类型不匹配的情况。

## ALTER SERVER 语句 - ODBC 包装器存取 Excel 数据的示例

在整个 Excel ODBC 驱动程序中，ODBC 包装器允许将谓词和聚集函数下推至数据源以供处理。该驱动程序也支持工作表之间的连接。下推处理可以提高性能，原因是当查询引用工作表时，所有数据源行不返回到 DB2。

假定您定义了服务器 *compounds\_workbook* 来标识包含您想要存取的数据的工作簿。对于此工作簿，您想要指定联合服务器可以将操作下推至 Excel ODBC 驱动程序以供处理。ALTER SERVER 语句为：

```
ALTER SERVER compounds_workbook  
  OPTIONS (ADD PUSHDOWN 'Y')
```

要利用下推能力，需要使用 CREATE SERVER 语句或通过使用 ALTER SERVER 语句来设置下列 ODBC 服务器选项。

- PUSHDOWN 'Y'
- DB2\_GROUP\_BY 'Y'
- DB2\_COLFUNC 'Y'

相关概念：

- 第 22 页的『用来存取 Excel 数据的方法』

相关任务：

- 第 284 页的『准备联合服务器以通过 ODBC 来存取数据源』

**相关参考:**

- 第 22 页的『选择正确的包装器』

---

## 对 ODBC 数据源的配置进行调整和故障诊断

在设置 ODBC 数据源的配置之后，您可能想修改该配置以提高性能。

### 获取 ODBC 跟踪

如果您在存取数据源时遇到了问题，则可以获取 ODBC 跟踪信息来分析和解决问题。但是，激活跟踪将影响系统性能。在解决问题之后应关闭跟踪。

在运行 Windows 的联合服务器上，单击“ODBC 数据源管理器”窗口中的跟踪。

在运行 UNIX 的联合服务器上，参阅 ODBC 客户机供应商的文档。

**相关任务:**

- 第 283 页的『将 ODBC 数据源添加至联合服务器』

**相关参考:**

- 『db2set - DB2 Profile Registry Command』 (*Command Reference*)



---

## 第 18 章 配置对 OLE DB 数据源的存取

本章说明如何配置联合服务器以存取存储在 OLE DB 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 OLE DB 数据源的存取。

本章列示需要执行的任务并且包含配置联合服务器时所需要的 SQL 语句的示例。

---

### 将 OLE DB 数据源添加至联合服务器

Microsoft OLE DB 是一组 OLE/COM 接口，它们使应用程序能够一致地存取存储在各种不同信息源中的数据。OLE DB 组件 DBMS 体系结构定义 OLE DB 使用程序和 OLE DB 提供程序。OLE DB 使用程序就是使用 OLE DB 接口的任何系统或应用程序。OLE DB 提供程序是用来公开 OLE DB 接口的一个组件。

OLE DB 包装器使您能够存取符合 Microsoft OLE DB 2.0（或更新版本）的 OLE DB 提供程序。

使用 OLE DB 包装器来创建表函数。不能使用该包装器来对数据源表和视图创建昵称。

要配置联合服务器以存取 OLE DB 数据源，必须为联合服务器提供有关 OLE DB 提供程序的信息。

可以通过 DB2 命令中心或者 DB2 命令行来配置对 OLE DB 数据源的存取权。

**先决条件:**

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

**限制:**

OLE DB 包装器仅用来帮助您注册用户定义的 OLE DB 外部表函数。与其它包装器不同，OLE DB 包装器不使用昵称来存取存储在数据源中的数据。

**过程:**

要将 OLE DB 数据源添加至联合服务器:

1. 注册包装器。
2. 注册服务器定义。
3. 创建用户映射。

在配置对 OLE DB 数据源的存取权之后，使用 CREATE FUNCTION 语句在联合数据库中注册用户定义的 OLE DB 外部表函数。

**相关概念:**

- 『DB2 Information Integrator 安装过程 - 概述』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 第 49 页的『配置数据源的快速方法』

**相关任务:**

- 第 44 页的『创建联合数据库』
- 第 298 页的『注册 OLE DB 包装器』

---

## 注册 OLE DB 包装器

注册 OLE DB 包装器是将 OLE DB 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 OLE DB 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

### 过程:

要注册包装器，发出 CREATE WRAPPER 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER OLEDB
```

**建议:** 使用称为 OLEDB 的缺省包装器名称。当您注册使用缺省名称的包装器时，联合服务器将自动采用与该包装器相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突，则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于缺省名称的名称，则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如，要在使用 Windows 操作系统的联合服务器上注册一个名称为 oledb\_wrapper 的包装器，请发出以下语句:

```
CREATE WRAPPER oledb_wrapper LIBRARY 'db2oledb.dll'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 OLE DB 包装器库文件的列表。

在这一系列任务中的下一个任务是为 OLE DB 包装器注册服务器定义。

### 相关任务:

- 第 299 页的『为 OLE DB 数据源注册服务器定义』

### 相关参考:

- 第 298 页的『OLE DB 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

---

## OLE DB 包装器库文件

下表列示了 OLE DB 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把库文件添加到该表中列示的目录路径中。

当注册包装器时，只指定该表中列示的库文件名。

表 80. OLE DB 客户机库位置和文件名

操作系统	目录路径	包装器库文件
Windows	%DB2PATH%\bin	db2oledb.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

**相关任务:**

- 第 298 页的『注册 OLE DB 包装器』

---

## 为 OLE DB 数据源注册服务器定义

为 OLE DB 数据源注册服务器定义是将 OLE DB 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 OLE DB 数据源服务器。

**过程:**

要为 OLE DB 数据源注册服务器定义:

发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name WRAPPER OLEDB
      OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
                          Data Source=c:\msdasdk\bin\oledb\wind.mdb')
```

在这一系列任务中的下一个任务是为 OLE DB 数据源创建用户映射。

**相关任务:**

- 第 300 页的『为 OLE DB 数据源创建用户映射』

**相关参考:**

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 299 页的『CREATE SERVER 语句 - OLE DB 包装器的示例』

---

## CREATE SERVER 语句 - OLE DB 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE SERVER 语句来为 OLE DB 包装器注册服务器。

以下示例显示了一个 CREATE SERVER 语句:

```
CREATE SERVER Nwind WRAPPER OLEDB
      OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
                          Data Source=c:\msdasdk\bin\oledb\wind.mdb',
      COLLATING_SEQUENCE 'Y')
```

Nwind 为 OLE DB 数据源指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

## WRAPPER OLEDB

在 CREATE WRAPPER 语句中指定的包装器名称。

**CONNECTSTRING** 'Provider=Microsoft.Jet.OLEDB.4.0; Data Source=c:\msdasdk\bin\oledb\nwind.mdb'

提供连接至数据源所需要的初始化属性。

字符串包含用分号隔开的一系列关键字与值的配对。等号 (=) 将每个关键字与它的值分开。关键字是对 OLE DB 初始化属性 (属性集 DBPROPSET\_DBINT) 或特定于提供程序的关键字的描述。

有关 CONNECTSTRING 选项的完整语法和语义, 请参阅 Microsoft Press 在 1998 年出版的 *Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK*。

## COLLATING\_SEQUENCE 'Y'

指定数据源是否使用与 DB2 UNIX 版和 Windows 版整理顺序相同的整理顺序。

有效值为 'Y' (数据源使用 DB2 UNIX 版和 Windows 版整理顺序) 和 'N' (数据源使用不同于 DB2 UNIX 版和 Windows 版整理顺序的整理顺序)。缺省值为 'N'。

### 相关任务:

- 第 299 页的『为 OLE DB 数据源注册服务器定义』

### 相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)

---

## 为 OLE DB 数据源创建用户映射

为 OLE 数据源创建用户映射是将 OLE 数据源添加至联合服务器这一大型任务的一部分。

当您尝试存取 OLE 数据源时, 联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联 (用户映射)。为将访问联合系统以便发送分布式请求的每个用户标识创建用户映射。

### 过程:

要将本地用户标识映射至 OLE 数据源用户标识和密码, 发出 CREATE USER MAPPING 语句。

例如:

```
CREATE USER MAPPING FOR userid SERVER server_name
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

如果 OLE DB 数据源上的密码或联合服务器上的密码的长度少于八个字符, 则存取 OLE DB 数据源的 SQL 语句将失败。将出现以下错误消息:

```
SQL30082N 由于安全性原因“15”而尝试建立连接失败
(“处理失败”)。SQLSTATE=08001
```

为了避免发生此问题，应将 OLE DB 数据源密码或联合服务器上的密码更改为八个字符或者超过八个字符。

**相关参考:**

- 『CREATE USER MAPPING statement』 ( *SQL Reference, Volume 2* )
- 第 301 页的 『CREATE USER MAPPING 语句 - OLE DB 包装器的示例』

---

## CREATE USER MAPPING 语句 - OLE DB 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将本地用户标识映射至 OLE 数据源用户标识和密码。此主题包括一个具有必需参数的完整示例，还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

**完整示例:**

以下示例说明了如何将本地用户标识映射至 OLE 数据源用户标识和密码:

```
CREATE USER MAPPING FOR laura SERVER Nwind
    OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

*laura* 您正在映射至在 OLE DB 数据源中定义的用户标识的本地用户标识。

**SERVER *Nwind***

您在 CREATE SERVER 语句中定义的 OLE DB 服务器的名称。

**REMOTE\_AUTHID '*lulu*'**

您正在将 *laura* 映射至的 OLE DB 服务器的用户标识。此值是区分大小写的。

**REMOTE\_PASSWORD '*raiders*'**

与 '*lulu*' 相关联的密码。此值是区分大小写的。

**专用寄存器示例:**

以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER Nwind
    OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

**相关任务:**

- 第 300 页的 『为 OLE DB 数据源创建用户映射』

**相关参考:**

- 『CREATE USER MAPPING statement』 ( *SQL Reference, Volume 2* )



---

## 第 19 章 配置对 Oracle 数据源的存取

本章说明如何配置联合服务器以存取存储在 Oracle 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Oracle 数据源的存取。

本章列示配置联合服务器时需要执行的任务，包含您需要的 SQL 语句的示例，而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 Oracle 添加至联合系统

#### 将 Oracle 数据源添加至联合服务器

要配置联合服务器以存取 Oracle 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Oracle 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

##### 先决条件:

- 在联合服务器上已经安装和配置了 Oracle 客户机软件。
- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

##### 过程:

要将 Oracle 数据源添加至联合服务器:

1. 设置和测试 Oracle 客户机配置文件。
2. 设置 Oracle 环境变量。
3. 注册包装器。
4. 注册服务器定义。
5. 创建用户映射。
6. 测试与 Oracle 服务器的连接。
7. 为 Oracle 表、视图和同义词注册昵称。

##### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』(《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』

##### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 307 页的『设置和测试 Oracle 客户机配置文件』
- 第 308 页的『注册 Oracle 包装器』

- 第 309 页的『为 Oracle 数据源注册服务器定义』
- 第 311 页的『为 Oracle 数据源创建用户映射』
- 第 312 页的『测试与 Oracle 服务器的连接』
- 第 313 页的『为 Oracle 表和视图注册昵称』
- 第 314 页的『对 Oracle 数据源的配置进行调整和故障诊断』
- 第 304 页的『设置 Oracle 环境变量』
- 第 33 页的『检查联合服务器的设置』

**相关参考:**

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 设置 Oracle 环境变量

设置 Oracle 环境变量是将 Oracle 添加至联合服务器这一大型任务的一部分。

当安装 DB2 Information Integrator 时, 安装过程将尝试在 db2dj.ini 文件中设置 Oracle 环境变量。

如果存在下列情况, 则不会在 db2dj.ini 文件中设置环境变量:

- 在设置 DB2 联合服务器之后才安装 Oracle 客户机软件。
- 尚未安装 Oracle 客户机软件。

Oracle 的有效环境变量为:

- ORACLE\_HOME
- ORACLE\_BASE (可选)
- ORA\_NLS (可选)
- TNS\_ADMIN (可选)
- NLS\_LANG (可选)

必须手工设置可选环境变量。

**先决条件:**

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件 (例如, 数据源客户机软件)。

**限制:**

请参阅主题: db2dj.ini 文件的限制

**过程:**

要自动设置必需的环境变量:

1. 在 DB2 联合服务器上安装和配置客户机软件 (如果尚未安装它的话)。



2. 设置必需的环境变量。可以通过再次运行 DB2 Information Integrator 安装来自动设置环境变量。单击启动板中的**安装产品**，并遵循向导中的指示信息。

要手工设置 Oracle 环境变量：

1. 编辑 db2dj.ini 文件。

- 在运行 Windows 的联合服务器上，此文件位于 sqllib\cfg 目录中。
- 在运行 UNIX 的联合服务器上，此文件位于 sqllib/cfg 目录中。

db2dj.ini 文件包含有关安装在联合服务器上的 Oracle 客户机软件的配置信息。如果该文件不存在，则可以使用任何文本编辑器来创建名称为 db2dj.ini 的新文件。在 db2dj.ini 文件中，必须指定变量的全限定路径，否则将遇到错误。

2. 根据需要设置下列环境变量：

### ORACLE\_HOME

将 ORACLE\_HOME 环境变量设置为安装了 Oracle 客户机软件的目录路径。指定该变量的全限定路径 ORACLE\_HOME=*oracle\_home\_directory*。例如，如果 Oracle 主目录为 \usr\oracle\8.1.7，则 db2dj.ini 中的条目为 ORACLE\_HOME=\usr\oracle\8.1.7

**注意：**如果联合实例的单个用户设置了 ORACLE\_HOME 环境变量，则联合实例将不使用该设置。联合实例只使用您在 db2dj.ini 文件中设置的 ORACLE\_HOME 的值。

### ORACLE\_BASE

ORACLE\_BASE 表示 Oracle 客户机目录树的根。如果在安装 Oracle 客户机软件时设置了 ORACLE\_BASE 变量，则在联合服务器上设置 ORACLE\_BASE 环境变量。例如：

```
ORACLE_BASE=oracle_root_directory
```

### ORA\_NLS\*

如果您的系统正在使用多个版本的 Oracle，则必须确保：

- 设置了适当的 ORA\_NLS 变量。
- 提供了正在使用的各个版本的相应 NLS 数据文件。

特定于位置的数据存储在由 ORA\_NLS\* 环境变量指定的目录中。对于每个新版本的 Oracle，具有不同的 ORA\_NLS 数据目录。

表 81. 按版本指定特定于语言环境的数据的位置的 Oracle ORA\_NLS 变量。

Oracle 版本	环境变量
7.2	ORA_NLS
7.3	ORA_NLS32
8.x, 9.x	ORA_NLS33
10g	ORA_NLS10

例如，对于存取 Oracle 8.1 数据源的 UNIX 联合服务器，设置 ORA\_NLS33 环境变量：

```
ORA_NLS33=oracle_home_directory/ocommon/nls/admin/<data>
```

### TNS\_ADMIN

- 在运行 Windows 的联合服务器上，Oracle 客户机在 %ORACLE\_HOME%\NETWORK\ADMIN 目录中查找 tnsnames.ora 文件，其中 %ORACLE\_HOME% 是在 db2dj.ini 文件中定义的。如果 tnsnames.ora 文件不在 %ORACLE\_HOME%\NETWORK\ADMIN 目录中，则需要联合服务器上设置 TNS\_ADMIN 环境变量。
- 在运行 AIX、Linux 和 HP-UX 的联合服务器上，客户机在 /etc 目录中查找 tnsnames.ora 文件。如果客户机在 /etc 目录中找不到 tnsnames.ora 文件，则客户机将在 \$ORACLE\_HOME/network/admin 目录中查找该文件，其中 \$ORACLE\_HOME 是在 db2dj.ini 文件中定义的。如果客户机找不到 tnsnames.ora 文件，则需要设置 TNS\_ADMIN 环境变量。在 db2dj.ini 文件中将该变量设置为 tnsnames.ora 文件所在的路径。

例如，如果 tnsnames.ora 文件在 /home/oracle 目录中，则需要将该变量设置为：

```
TNS_ADMIN=/home/oracle
```

- 在运行 Solaris 的联合服务器上，客户机在 /var/opt/oracle 目录中查找 tnsnames.ora 文件。如果客户机在 /var/opt/oracle 目录中找不到 tnsnames.ora 文件，则客户机将在 \$ORACLE\_HOME/network/admin 目录中查找该文件，其中 \$ORACLE\_HOME 是在 db2dj.ini 文件中定义的。如果客户机找不到 tnsnames.ora 文件，则需要设置 TNS\_ADMIN 环境变量。在 db2dj.ini 文件中将该变量设置为 tnsnames.ora 文件所在的路径。

例如，如果 tnsnames.ora 文件在 /home/oracle 目录中，则需要将该变量设置为：

```
TNS_ADMIN=/home/oracle
```

3. 在 UNIX 上，使用 Oracle 环境变量来更新 DB2 实例的 .profile 文件。可以发出以下命令来完成此任务：

```
export ORACLE_HOME=oracle_home_directory
export PATH=$ORACLE_HOME/bin:$PATH
```

其中 *oracle\_home\_directory* 是 Oracle 客户机软件的安装目录。

4. 在 UNIX 上，通过输入以下命令来执行 DB2 实例 .profile：
  - \$HOME/ .profile
5. 要确保在联合服务器上设置了环境变量，重新启动 DB2 实例。发出下列命令来重新启动 DB2 实例：

```
db2stop
db2start
```

### 设置 Oracle 代码页转换：

对于 Oracle 代码页转换，可以设置可选环境变量 NLS\_LANG。

每当 Oracle 包装器连接至 Oracle 数据源，包装器就会确定将哪个代码页值用于该连接。如果在联合服务器上的 db2dj.ini 文件中设置了 NLS\_LANG，则包装器将使用 db2dj.ini 文件中的值。db2dj.ini 文件包含有关安装在联合服务器上的 Oracle 客户机软件的配置信息。

如果在联合服务器上未设置 NLS\_LANG 变量，则包装器将确定联合数据库的地域和代码页。包装器将 NLS\_LANG 设置为最接近匹配的 Oracle 语言环境。如果没有最接近匹配的语言环境，则将 NLS\_LANG 设置为 American\_America.US7ASCII。

如果存取包含使用中文代码页 GB 18030 的数据的数据源，则联合数据库必须使用 UTF-8 代码页。对于 Oracle 数据源，Oracle 包装器将 Oracle NLS\_LANG 环境变量设置为：

```
NLS_LANG=Simplified Chinese_China.UTF8
```

如果您正在使用 Oracle 9i 客户机，则将 db2dj.ini 文件中的 NLS\_LANG 设置更改为 Simplified Chinese\_China.AL32UTF8，以便 Oracle 9i 客户机能够正确地将 GB 18030 数据转换为 Unicode。例如：

```
NLS_LANG=Simplified Chinese_China.AL32UTF8
```

有关有效语言环境的列表，请参阅随 Oracle 软件一起提供的文档。

#### 相关任务：

- 第 313 页的『为 Oracle 表和视图注册昵称』

#### 相关参考：

- 第 53 页的『对于 db2dj.ini 文件的限制』
- 第 53 页的『对于 db2dj.ini 文件的限制』

## 设置和测试 Oracle 客户机配置文件

设置和测试 Oracle 客户机配置文件是将 Oracle 数据源添加至联合服务器这一大型任务的一部分。

客户机配置文件用来使用安装在联合服务器上的客户机库来连接至 Oracle 数据库。此文件指定每个 Oracle 数据库服务器的位置以及数据库服务器的连接（协议）的类型。Oracle 客户机配置文件的缺省名称为 tnsnames.ora。

#### 过程：

要设置和测试 Oracle 客户机配置文件：

1. 使用随 Oracle 客户机软件一起提供的“Oracle NET8/NET 配置”实用程序。

有关使用此实用程序的更多信息，请参阅 Oracle 的安装文档。在 tnsnames.ora 文件中，SID（或 SERVICE\_NAME）是 Oracle 实例的名称，HOST 是 Oracle 服务器所在的主机名。

创建 tnsnames.ora 文件所在的目录取决于联合服务器上正在运行的操作系统。

- 在 UNIX 操作系统上，此文件的缺省路径和名称为 \$ORACLE\_HOME/network/admin。
  - 在 Windows 操作系统上，此文件的缺省路径和名称为 %ORACLE\_HOME%\NETWORK\ADMIN。
2. 如果想将 tnsnames.ora 文件放置在一个不同于缺省搜索路径的路径中，则设置 TNS\_ADMIN 环境变量以指定文件位置。
    - a. 编辑位于 sqllib/cfg 目录中的 db2dj.ini 文件，并设置 TNS\_ADMIN 环境变量：

```
TNS_ADMIN=x:/path/
```

- b. 发出下列命令来重新启动 DB2 实例，并确保在程序中设置了环境变量：

```
db2stop  
db2start
```

3. 使用 Oracle **sqlplus** 工具来测试连接，以确保客户机软件能够连接至 Oracle 服务器。

在这一系列任务中的下一个任务是注册 Oracle 包装器。

#### 相关任务:

- 第 308 页的『注册 Oracle 包装器』
- 第 314 页的『对 Oracle 数据源的配置进行调整和故障诊断』

## 注册 Oracle 包装器

注册 Oracle 包装器是将 Oracle 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Oracle 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器，发出 CREATE WRAPPER 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER NET8
```

**建议:** 使用称为 NET8 的缺省包装器名称。当您使用其中一个缺省名称来注册包装器时，联合服务器将自动采用与该包装器相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突，则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于其中一个缺省名称的名称，则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 oracle\_wrapper 的包装器，请发出以下语句:

```
CREATE WRAPPER oracle_wrapper LIBRARY 'libdb2net8.a'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 Oracle 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Oracle 包装器注册服务器定义。

#### 相关任务:

- 第 309 页的『为 Oracle 数据源注册服务器定义』

#### 相关参考:

- 第 309 页的『Oracle 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## Oracle 包装器库文件

下表列示了 Oracle 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2net8.a、libdb2net8F.a 和 libdb2net8U.a。

当注册包装器时，只指定该表中列示的库文件名。

表 82. Oracle 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2net8F.a (NET8)
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2net8F.sl (NET8)
Linux	/opt/IBM/db2/V8.1/lib	libdb2net8F.so (NET8)
Solaris	/opt/IBM/db2/V8.1/lib	libdb2net8F.so (NET8)
Windows	%DB2PATH%\bin	db2net8.dll (NET8)

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

### 相关任务:

- 第 309 页的『为 Oracle 数据源注册服务器定义』

### 相关参考:

- 『CREATE WRAPPER statement』 (SQL Reference, Volume 2)

## 为 Oracle 数据源注册服务器定义

为 Oracle 数据源注册服务器定义是将 Oracle 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 Oracle 服务器。必须首先找到 Oracle 数据源的节点名，然后在注册服务器时使用此节点名。

### 过程:

要为 Oracle 数据源注册服务器定义:

1. 在 Oracle tnsnames.ora 文件中找到节点名。

#### 示例 tnsnames.ora 文件:

```
paris_node =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = somehost)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = ora9i.see1)))
```

在此示例中，在 CREATE SERVER 语句中要使用的节点值为 paris\_node。

尽管 node\_name 被指定为 CREATE SERVER SQL 语句中的一个选项，但是，它对于 Oracle 数据源是必需的。

2. 发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name TYPE oracle VERSION 8.1.7 WRAPPER net8
OPTIONS (NODE 'node_name')
```

在创建了服务器定义之后, 使用 ALTER SERVER 语句来添加或删除服务器选项。

在这一系列任务中的下一个任务是为 Oracle 数据源创建用户映射。

相关任务:

- 第 311 页的『为 Oracle 数据源创建用户映射』

相关参考:

- 『ALTER SERVER statement』(SQL Reference, Volume 2)
- 『CREATE SERVER statement』(SQL Reference, Volume 2)
- 第 529 页的附录 D, 『联合系统的服务器选项』
- 第 310 页的『CREATE SERVER 语句 - Oracle 包装器的示例』

## CREATE SERVER 语句 - Oracle 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE SERVER 语句来为 Oracle 包装器注册服务器。此主题包括一个完整的示例, 该示例说明如何使用必需的参数来注册服务器, 它还包括一个带有更多服务器选项的示例。

完整示例:

以下示例说明了如何使用 CREATE SERVER 语句来为 Oracle 包装器注册服务器定义:

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER net8
OPTIONS (NODE 'paris_node')
```

**oraserver**

为 Oracle 数据库服务器指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

**TYPE oracle**

指定正在对其配置存取权的数据源服务器的类型。SQLNET 和 NET8 包装器的 TYPE 参数必须为 *oracle*。

**VERSION 8.1.7**

想要访问的 Oracle 数据库服务器的版本。受支持的 Oracle 版本为 7.3.4、8.x 和 9.x。

**WRAPPER net8**

在 CREATE WRAPPER 语句中指定的名称。

**NODE 'paris\_node'**

Oracle 数据库服务器所驻留的节点的名称。从 tnsnames.ora 文件中可获取节点名。

尽管节点名被指定为 CREATE SERVER 语句中的一个选项, 但是, 它对于 Oracle 数据源是必需的。

服务器选项示例:

当您创建服务器定义时，可以在 `CREATE SERVER` 语句中指定更多服务器选项。这些服务器选项包括一般的服务器选项和特定于 Oracle 的服务器选项。

DB2 假定所有 Oracle `VARCHAR` 列都包含结尾空白。如果您能肯定 Oracle 数据库中的所有 `VARCHAR` 列都不包含结尾空白，则可以设置一个服务器选项来指定数据源使用非空白填充的 `VARCHAR` 比较语义。

以下示例显示了具有此服务器选项的 Oracle 服务器定义：

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER net8
      OPTIONS (NODE 'paris_node', VARCHAR_NO_TRAILING_BLANKS 'Y')
```

当没有任何列包含结尾空白时，使用 `VARCHAR_NO_TRAILING_BLANKS` 服务器选项。如果只有某些 `VARCHAR` 列包含结尾空白，则可以使用 `ALTER NICKNAME` 语句来对这些特定列设置一个选项。

**相关任务：**

- 第 309 页的『为 Oracle 数据源注册服务器定义』

**相关参考：**

- 『`CREATE SERVER` statement』 (*SQL Reference, Volume 2*)

## 为 Oracle 数据源创建用户映射

为 Oracle 数据源创建用户映射是将 Oracle 数据源添加至联合服务器这一大型任务的一部分。

当您尝试访问 Oracle 服务器时，联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联（用户映射）。为将访问联合系统以便对 Oracle 数据源发送分布式请求的每个用户标识创建用户映射。

**过程：**

要将本地用户标识映射至 Oracle 服务器用户标识和密码，发出 `CREATE USER MAPPING` 语句。

例如：

```
CREATE USER MAPPING FOR userid SERVER oraserver
      OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

在这一系列任务中的下一个任务是测试与 Oracle 服务器的连接。

**相关任务：**

- 第 312 页的『测试与 Oracle 服务器的连接』

**相关参考：**

- 『`CREATE USER MAPPING` statement』 (*SQL Reference, Volume 2*)
- 第 312 页的『`CREATE USER MAPPING` 语句 - Oracle 包装器的示例』

## CREATE USER MAPPING 语句 - Oracle 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将联合服务器用户标识映射至 Oracle 服务器用户标识和密码。此主题包括一个具有必需参数的完整示例，还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

### 完整示例:

以下示例说明了如何将联合服务器用户标识映射至 Oracle 服务器用户标识和密码:

```
CREATE USER MAPPING FOR robert SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now')
```

*robert* 指定您正在映射至在 Oracle 服务器中定义的用户标识的本地用户标识。

### SERVER *oraserver*

指定您在 CREATE SERVER 语句中定义的 Oracle 服务器的名称。

### REMOTE\_AUTHID '*rob*'

指定您正在将 *robert* 映射至的 Oracle 数据库服务器的用户标识。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_ID 服务器选项设置为 'U' 或 'L'）。

### REMOTE\_PASSWORD '*then4now*'

指定与 '*rob*' 相关联的密码。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_PW 服务器选项设置为 'U' 或 'L'）。

### 专用寄存器示例:

以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

**限制:** 必须已经使用带有 'identified by' 子句而不是 'identified externally' 子句的 Oracle **create user** 命令创建了 Oracle 数据源中的用户标识。

### 相关任务:

- 第 311 页的『为 Oracle 数据源创建用户映射』

### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）

## 测试与 Oracle 服务器的连接

测试与 Oracle 服务器的连接是将 Oracle 数据源添加至联合服务器这一大型任务的一部分。

可以使用您定义的服务器定义和用户映射来测试与 Oracle 服务器的连接。

### 过程:



要测试连接:

1. 打开传递会话以便对 Oracle 系统表发出 SQL SELECT 语句。

例如:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM sys.all_tables
SET PASSTHRU RESET
```

如果该 SQL SELECT 语句返回了一个计数,则表示正确设置了服务器定义和用户映射。

2. 如果 SQL SELECT 语句返回了错误,则可能需要:
  - 检查 Oracle 服务器以确保为入局连接配置了 Oracle 服务器。
  - 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 Oracle 服务器的连接是有效的。在需要时,改变用户映射,或者创建另一个用户映射。
  - 检查 DB2 联合服务器上的 Oracle 客户机软件,以确保正确安装和配置了该软件以便连接至 Oracle 服务器。
  - 检查 DB2 联合变量的设置以验证它们对于 Oracle 服务器是否正确。这些变量包括系统环境变量、db2dj.ini 变量和“DB2 概要文件注册表”(db2set)变量。
  - 检查服务器定义。如果需要的话,删除服务器定义,然后再次创建它。

在这一系列任务中的下一个任务是为 Oracle 表和视图注册昵称。

相关任务:

- 第 303 页的『将 Oracle 数据源添加至联合服务器』
- 第 307 页的『设置和测试 Oracle 客户机配置文件』
- 第 313 页的『为 Oracle 表和视图注册昵称』
- 第 304 页的『设置 Oracle 环境变量』

相关参考:

- 『ALTER USER MAPPING statement』(SQL Reference, Volume 2)

## 为 Oracle 表和视图注册昵称

为 Oracle 表和视图注册昵称是将 Oracle 数据源添加至联合服务器这一大型任务的一部分。

对于您定义的每个 Oracle 服务器,为想要存取每个表或视图注册昵称。当您查询 Oracle 服务器时,使用这些昵称而不是数据源对象的名称。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是您使用 CREATE NICKNAME 语句为数据源对象注册昵称时收集的。联合数据库验证数据源中的对象是否存在,然后尝试收集现有数据源的统计数据。对于优化器很有用的信息是从数据源目录中读取的,并且放置到联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息,所以,在注册昵称之前(使用与 DB2 RUNSTATS 命令等价的数据源命令)更新数据源中的统计信息。

过程:

要注册昵称，发出 CREATE NICKNAME 语句。

例如：

```
CREATE NICKNAME oracle_name FOR oraserver."remote_schema"."remote.table"
```

昵称的长度最多可以为 128 个字符。

对于您想为其创建昵称的每个 Oracle 表或视图重复此步骤。

当创建昵称时，DB2 将使用连接来查询数据源目录。此查询将使用昵称来测试与数据源的连接。如果该连接不工作，则您将接收到错误消息。

**相关参考：**

- 『RUNSTATS Command』（*Command Reference*）
- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 314 页的『CREATE NICKNAME 语句 - Oracle 包装器的示例』

## CREATE NICKNAME 语句 - Oracle 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE NICKNAME 语句来为您想存取的 Oracle 表或视图注册昵称。

此示例说明了如何为在其中指定昵称的 Oracle 服务器指定远程对象：

```
CREATE NICKNAME PARISINV FOR oraserver."france"."inventory"
```

*PARISINV*

用来标识 Oracle 表或视图的唯一昵称。

**注：**昵称是一个由两部分组成的名称 - 模式和昵称。如果在注册昵称时省略了模式，则昵称的模式将是注册该昵称的用户的授权标识。

*oraserver."france"."inventory"*

远程对象的由三部分组成的标识：

- *oraserver* 是您在 CREATE SERVER 语句中为 Oracle 数据库服务器指定的名称。
- *france* 是表或视图所属于的远程模式的名称。
- *inventory* 是您想存取的远程表或视图的名称。

除非用引号将 Oracle 模式和表的名称引起来，否则，联合服务器将把 Oracle 模式和表的名称转变为大写。

**相关任务：**

- 第 313 页的『为 Oracle 表和视图注册昵称』

**相关参考：**

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）

---

## 对 Oracle 数据源的配置进行调整和故障诊断

在设置 Oracle 数据源的配置之后，可以修改该配置以提高性能。

## 连接问题

对于 `tnsnames.ora` 文件的 `DESCRIPTION` 部分中的每个 `HOST`，可能需要更新 `TCP/IP` `hosts` 文件。是否需要更新此文件取决于您的网络上是如何配置 `TCP/IP` 的。网络的部件必须将在 `tnsnames.ora` 文件的 `DESCRIPTION` 部分指定的远程主机名转换成一个地址。

如果网络中具有一个已指定的服务器能够识别主机名，则不需要更新 `TCP/IP` `hosts` 文件。否则，需要具有远程主机的条目。咨询网络管理员，以确定您的网络是如何配置的。如果需要更新 `hosts` 文件，该文件的位置取决于联合服务器使用的操作系统：

在运行 **UNIX** 的联合服务器上

更新 `/etc/hosts` 文件。

在运行 **Windows** 的联合服务器上

更新 `x:\winnt\system32\drivers\etc\hosts` 文件。

相关任务:

- 第 303 页的『将 Oracle 数据源添加至联合服务器』

相关参考:

- 『`db2set - DB2 Profile Registry Command`』（*Command Reference*）



---

## 第 20 章 配置对 Sybase 数据源的存取

本章说明如何配置联合服务器以存取存储在 Sybase 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Sybase 数据源的存取。

本章列示配置联合服务器时需要执行的任务，包含您需要的 SQL 语句的示例，而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 Sybase 添加至联合系统

#### 将 Sybase 数据源添加至联合服务器

要配置联合服务器以存取 Sybase 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Sybase 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

##### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

##### 过程:

要将 Sybase 数据源添加至联合服务器:

1. 设置和测试 Sybase 客户机配置文件。
2. 设置 Sybase 环境变量。
3. 注册包装器。
4. 注册服务器定义。
5. 创建用户映射。
6. 测试与 Sybase 服务器的连接。
7. 为 Sybase 表和视图注册昵称。

##### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』(《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』

##### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 33 页的『检查联合服务器的设置』
- 第 320 页的『设置和测试 Sybase 客户机配置文件』
- 第 321 页的『注册 Sybase 包装器』

- 第 322 页的『为 Sybase 数据源注册服务器定义』
- 第 325 页的『为 Sybase 数据源创建用户映射』
- 第 326 页的『测试与 Sybase 服务器的连接』
- 第 327 页的『为 Sybase 表和视图注册昵称』
- 第 328 页的『对 Sybase 数据源的配置进行调整和故障诊断』
- 第 318 页的『设置 Sybase 环境变量』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 设置 Sybase 环境变量

设置 Sybase 环境变量是将 Sybase 添加至联合服务器这一大型任务的一部分。

当安装 DB2 Information Integrator 时, 安装过程将尝试在 db2dj.ini 文件中设置 Sybase 环境变量。

如果存在下列情况, 则不会在 db2dj.ini 文件中设置环境变量:

- 在设置联合服务器之后才安装 Sybase 客户机软件。
- 尚未安装 Sybase 客户机软件。

Sybase 的有效环境变量为:

- SYBASE
- SYBASE\_OCS (对于 Sybase V12 或更新版本是必需的)
- SYBASE\_CHARSET (可选)

必须手工设置可选环境变量。

#### 先决条件:

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件 (例如, 数据源客户机软件)。

#### 限制:

请参阅主题: db2dj.ini 文件的限制

#### 过程:

要自动设置必需的环境变量:

1. 在联合服务器上安装和配置客户机软件 (如果尚未安装它的话)。
2. 设置必需的环境变量。可以通过再次运行 DB2 Information Integrator 安装来自动设置环境变量。单击启动板中的**安装产品**, 并遵循向导中的指示信息。

要手工设置环境变量:

1. 编辑 db2dj.ini 文件。

- 在运行 Windows 的联合服务器上，此文件位于 `sqllib\cfg` 目录中。
- 在运行 UNIX 的联合服务器上，此文件位于 `sqllib/cfg` 目录中。

`db2dj.ini` 文件包含有关安装在联合服务器上的 Sybase 客户机软件的配置信息。如果该文件不存在，则可以使用任何文本编辑器来创建名称为 `db2dj.ini` 的新文件。在 `db2dj.ini` 文件中，必须指定变量的全限定路径，否则将遇到错误。

2. 根据需要设置下列环境变量：

### SYBASE

将 SYBASE 环境变量设置为安装了 Sybase Open Client 软件的目录路径。指定此变量的全限定路径。例如：

```
SYBASE=sybase_home_directory
```

例如，如果目录路径为 `D:/djxclient/sybase/V12/OCS-12_5`，则您指定的 SYBASE 变量为：

```
SYBASE=D:/djxclient/sybase/V12
```

### SYBASE\_OCS

对于 Sybase Open Client V12 或更新版本，将 SYBASE\_OCS 环境变量设置为 OCS 目录的名称。不要指定全限定路径。SYBASE\_OCS 环境变量指定所安装的 Sybase Open Client 的版本和发行版。

```
SYBASE_OCS=OCS-version_release
```

例如，如果目录路径为 `D:/djxclient/sybase/V12/OCS-12_5`，则您为 SYBASE\_OCS 变量指定的目录为：

```
SYBASE_OCS=OCS-12_5
```

### SYBASE\_CHARSET

将 SYBASE\_CHARSET 变量设置为想要使用的字符集的名称。Sybase 包装器使用 SYBASE\_CHARSET 来确定要使用的字符集。如果未设置 SYBASE\_CHARSET 变量，则包装器将使用与在联合服务器的代码页上指定的字符集相匹配的 Sybase 字符集。如果没有相匹配的 Sybase 字符集，则包装器将使用 `iso_1` 字符集。可以在 `$SYBASE\charsets` 目录中查看有效字符集名称的列表。例如：

```
SYBASE_CHARSET=iso_1
```

`iso_1` 是您想要使用的字符集的名称。

3. 使用 SYBASE 环境变量来更新联合数据库实例的 `.profile` 文件。可以通过发出下列命令来完成此任务：

```
export SYBASE=<sybase_home_directory>
export SYBASE_OCS=OCS-version_release
export PATH=$SYBASE/bin:$PATH
```

4. 从主目录中，通过输入以下命令来运行联合数据库实例 `.profile`：

```
..profile
```

5. 要确保在联合服务器上设置了环境变量，重新启动联合数据库实例。发出下列命令来重新启动联合数据库实例：

```
db2stop
db2start
```

在这一系列任务中的下一个任务是注册 Sybase 包装器。

**相关任务:**

- 第 327 页的『为 Sybase 表和视图注册昵称』

**相关参考:**

- 第 53 页的『对于 db2dj.ini 文件的限制』
- 第 53 页的『对于 db2dj.ini 文件的限制』

## 设置和测试 Sybase 客户机配置文件

设置和测试 Sybase 客户机配置文件是将 Sybase 数据源添加至联合服务器这一大型任务的一部分。

客户机配置文件用来通过使用安装在联合服务器上的 Sybase Open Client 库来连接至 Sybase。此文件指定每个 Sybase SQL Server 和 Adaptive Server Enterprise 实例的位置以及数据库服务器的连接（协议）类型。

在联合服务器中的每个实例上，必须设置将用来连接至 Sybase 的客户机配置文件。设置和测试此文件必须执行的步骤取决于联合服务器上正在运行的操作系统。

**过程:**

要设置和测试客户机配置文件:

**在 UNIX 操作系统上:**

1. 可以使用随 Sybase Open Client 软件一起提供的实用程序来设置客户机配置文件。此文件是在 \$SYBASE 目录中创建的。该文件的名称为 interfaces。有关使用此实用程序的更多信息，请参阅 Sybase 文档。
2. 通过使用下列方法之一来使联合数据库实例可存取 interfaces 文件:
  - 将此文件复制到联合数据库实例的 \$HOME/sqllib 目录中。
  - 使用 **ln** 命令来创建从 \$sybase 目录到实例 \$HOME/sqllib 目录中的 interfaces 文件的链接。例如:

```
ln -s -f /home/sybase/interfaces /home/db2djinst1/sqllib
```
  - 使用 IFILE 服务器选项来指定指向 Sybase interfaces 文件的全路径。
3. 测试该连接以确保 Sybase Open Client 软件能够连接至 Sybase 服务器。使用适当的 Sybase 查询实用程序，例如 **isql**。例如:

```
isql -Ssybnode -Umary -I/home/db2djinst1/sqllib/interfaces
```

**在 Windows 操作系统上:**

1. 可以使用随 Sybase Open Client 软件一起提供的实用程序来设置客户机配置文件。此文件是在 %SYBASE%\ini 目录中创建的。该文件的名称为 sql.ini。有关使用此实用程序的更多信息，请参阅 Sybase 文档。
2. 通过将此文件复制到联合数据库实例的 %DB2PATH% 目录来使联合数据库实例可存取此 sql.ini 文件。缺省路径为 c:\Program Files\IBM\SQLLIB。



因为 DB2 Information Integrator 使用 interfaces 作为 Sybase 客户机配置文件的缺省名称, 所以, 应将 c:\Program Files\IBM\SQLLIB 目录中的 Windows sql.ini 文件重命名为 interfaces。

**要求:** 如果不将 sql.ini 文件重命名为 interfaces, 则当您创建服务器定义时必须使用 IFILE 服务器选项。

3. 测试该连接以确保 Sybase Open Client 软件能够连接至 Sybase 服务器。使用适当的 Sybase 查询实用程序, 例如 **isql**。例如:

```
isql -Ssybnode -Umary -I"c:\Program Files\IBM\SQLLIB\interfaces"
```

在这一系列任务中的下一个任务是注册 Sybase 包装器。

**相关任务:**

- 第 321 页的『注册 Sybase 包装器』

## 注册 Sybase 包装器

注册 Sybase 包装器是将 Sybase 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Sybase 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

**过程:**

要注册包装器, 发出 CREATE WRAPPER 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER CTLIB
```

**建议:** 使用称为 CTLIB 的缺省包装器名称。当您使用缺省名称来注册包装器时, 联合服务器将自动采用与该包装器名称相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突, 则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于缺省名称的名称, 则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如, 要在使用 AIX 操作系统的联合服务器上注册一个名称为 sybase\_wrapper 的包装器, 请发出以下语句:

```
CREATE WRAPPER sybase_wrapper LIBRARY 'libdb2ctlib.a'
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称, 请参阅 Sybase 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Sybase 包装器注册服务器定义。

**相关任务:**

- 第 322 页的『为 Sybase 数据源注册服务器定义』

**相关参考:**

- 第 322 页的『Sybase 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## Sybase 包装器库文件

下表列示了 Sybase 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2ctlib.a、libdb2ctlibF.a 和 libdb2ctlibU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 83. Sybase 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2ctlib.a (CTLIB)
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2ctlib.sl (CTLIB)
Linux	/opt/IBM/db2/V8.1/lib	libdb2ctlib.so (CTLIB)
Solaris	/opt/IBM/db2/V8.1/lib	libdb2ctlib.so (CTLIB)
Windows	%DB2PATH%\bin	db2ctlib.dll (CTLIB)

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

### 相关任务:

- 第 321 页的『注册 Sybase 包装器』

## 为 Sybase 数据源注册服务器定义

为 Sybase 数据源注册服务器定义是将 Sybase 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 Sybase 服务器。

### 限制:

如果使用 DB2 控制中心来为 Sybase 数据源注册服务器定义，则 Sybase 客户机配置文件 interfaces 必须存在于以下缺省目录中:

- 在运行 UNIX 的联合服务器上，缺省目录为 \$HOME/sqllib/
- 在运行 Windows 的联合服务器上，缺省目录为 c:\Program Files\IBM\SQLLIB

### 过程:

可以从 DB2 控制中心或 DB2 命令行注册服务器定义:

- 要从 DB2 控制中心执行此任务，使用“联合对象”向导或者右键单击**服务器定义**文件夹并单击**创建**。“发现”工具将为 Sybase 服务器检索节点名。必须指定 DBNAME 服务器选项的信息才能注册服务器定义。
- 要从 DB2 命令行执行此任务，请使用 CREATE SERVER 语句:
  1. 在 Sybase interfaces 文件中找到节点名。

**UNIX 操作系统上的示例 interfaces 文件:**

```
sybase119
query tcp ether anaconda 4100
```

### Windows NT 或 Windows 2000 操作系统上的示例 interfaces 文件:

```
[sybase119]
query=TCP,anaconda,4100
```

在这些示例中，节点名为 sybase119。节点名后面跟着连接类型（TCP/IP）和主机名（anaconda）。

尽管节点名被指定为 CREATE SERVER 语句中的一个选项，但是，它对于 Sybase 数据源是必需的。

#### 2. 发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name TYPE SYBASE
        VERSION 12.0 WRAPPER CTLIB
        OPTIONS (NODE 'sybnode', DBNAME 'sybdb')
```

在创建了服务器定义之后，使用 ALTER SERVER 语句来添加或删除服务器选项。

**要点:** 如果当您设置 Sybase 客户机配置文件时不将 sql.ini 文件重命名为 interfaces，则当您注册服务器定义时必须使用 IFILE 服务器选项。

在这一系列任务中的下一个任务是为 Sybase 数据源创建用户映射。

#### 相关任务:

- 第 325 页的『为 Sybase 数据源创建用户映射』

#### 相关参考:

- 『ALTER SERVER statement』 (*SQL Reference, Volume 2*)
- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 529 页的附录 D, 『联合系统的服务器选项』
- 第 323 页的『CREATE SERVER 语句 - Sybase 包装器的示例』

## CREATE SERVER 语句 - Sybase 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE SERVER 语句来为 Sybase 包装器注册服务器定义。此主题包括一个完整的示例，该示例说明如何使用必需的参数来注册服务器定义，它还包括一个带有其它服务器选项的示例。

#### 完整示例:

以下示例说明了如何通过发出 CREATE SERVER 语句来为 Sybase 包装器注册服务器定义:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
        OPTIONS (NODE 'sybnode', DBNAME 'sybdb')
```

#### SYBSERVER

为 Sybase 服务器指定的名称。此名称必须是唯一的。不允许存在重复的服务器名。

#### TYPE SYBASE

将 Sybase 指定为正在对其存取权进行配置的数据源的类型。CTLIB 包装器的 TYPE 参数必须为 SYBASE。

#### VERSION 12.0

想要访问的 Sybase 数据库服务器软件的版本。受支持的版本为 11、11.5、11.9、12 和 12.5。

#### WRAPPER CTLIB

在 CREATE WRAPPER 语句中指定的包装器名称。

#### NODE 'sybnode'

SYBSERVER 所驻留的节点的名称。可从 interfaces 文件中获取节点名。此值是区分大小写的。

尽管节点名被指定为 CREATE SERVER 语句中的一个选项，但是，它对于 Sybase 数据源是必需的。

#### DBNAME 'sybdb'

想要存取的 Sybase 数据库的名称。可从 Sybase 服务器中获取此名称。此值是区分大小写的。

尽管数据库名称被指定为 CREATE SERVER 语句中的一个选项，但是，它对于 Sybase 数据源是必需的。

**要点：** 如果当您设置 Sybase 客户机配置文件时不将 sql.ini 文件重命名为 interfaces，则当您注册服务器定义时必须使用 IFILE 服务器选项。

#### 服务器选项示例:

当您注册服务器时，可以在 CREATE SERVER 语句中指定更多服务器选项。这些服务器选项包括一般的服务器选项和特定于 Sybase 的服务器选项。

以下示例说明了当在运行 UNIX 的联合服务器上注册服务器定义时如何使用 TIMEOUT 服务器选项:

```
CREATE SERVER SYBSERVER TYPE SYBASE
      VERSION 12.0 WRAPPER CTLIB
      OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
              TIMEOUT '60')
```

超时值就是包装器等待来自 Sybase 服务器的响应所花的秒数。使用 TIMEOUT 选项来避免事务发生死锁。

以下示例说明了当在运行 Windows 的联合服务器上注册服务器定义时如何使用 IFILE 服务器选项:

```
CREATE SERVER SYBSERVER TYPE SYBASE
      VERSION 12.0 WRAPPER CTLIB
      OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
              IFILE 'C:\Sybase\ini\sql.ini')
```

IFILE 值是 Sybase Open Client interfaces 文件的全路径和名称。如果未将 sql.ini 文件作为 \$SQLLIB\interfaces (在 UNIX 系统上) 或者作为 %SQLLIB%\interfaces (在 Windows 操作系统上) 来进行复制或链接，则使用此服务器选项。

特定于 Sybase 的其它服务器选项有:

- LOGIN\_TIMEOUT
- PACKET\_SIZE

相关任务:

- 第 322 页的『为 Sybase 数据源注册服务器定义』

相关参考:

- 『CREATE SERVER statement』（*SQL Reference, Volume 2*）
- 第 529 页的附录 D, 『联合系统的服务器选项』

## 为 Sybase 数据源创建用户映射

为 Sybase 数据源创建用户映射是将 Sybase 数据源添加至联合服务器这一大型任务的一部分。

当您尝试访问 Sybase 服务器时, 联合服务器将使用对于该数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联(用户映射)。为将访问联合系统以便对 Sybase 数据源发送分布式请求的每个用户标识创建用户映射。

过程:

要将本地用户标识映射至 Sybase 服务器用户标识和密码, 发出 CREATE USER MAPPING 语句。

例如:

```
CREATE USER MAPPING FOR userid SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

在这一系列任务中的下一个任务是测试与 Sybase 服务器的连接。

相关任务:

- 第 326 页的『测试与 Sybase 服务器的连接』

相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）
- 第 325 页的『CREATE USER MAPPING 语句 - Sybase 包装器的示例』

## CREATE USER MAPPING 语句 - Sybase 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将联合服务器用户标识映射至 Sybase 服务器用户标识和密码。此主题包括一个具有必需参数的完整示例, 还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

完整示例:

以下示例说明了如何将联合服务器用户标识映射至 Sybase 服务器用户标识和密码:

```
CREATE USER MAPPING FOR maria SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night')
```

*maria* 指定您正在映射至在 Sybase 服务器中定义的用户标识的本地用户标识。

## SERVER SYBSERVER

指定您在 CREATE SERVER 语句中定义的 Sybase 服务器的名称。

## REMOTE\_AUTHID 'mary'

指定您正在将 *maria* 映射至的 Sybase 服务器上的用户标识。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_ID 服务器选项设置为 'U' 或 'L'）。

## REMOTE\_PASSWORD 'day2night'

指定与 'mary' 相关联的密码。使用单引号来保持此值的大小写（除非您在 CREATE SERVER 语句中将 FOLD\_PW 服务器选项设置为 'U' 或 'L'）。

### 专用寄存器示例:

以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER SYBSERVER
  OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

### 相关任务:

- 第 325 页的『为 Sybase 数据源创建用户映射』

### 相关参考:

- 『CREATE USER MAPPING statement』（*SQL Reference, Volume 2*）

## 测试与 Sybase 服务器的连接

测试与 Sybase 服务器的连接是将 Sybase 数据源添加至联合服务器这一大型任务的一部分。

可以使用您定义的服务器定义和用户映射来测试与 Sybase 服务器的连接。

### 过程:

要测试连接:

1. 打开传递会话以便对 Sybase 系统表发出 SQL SELECT 语句。

例如:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

其中 *remote\_server\_name* 是您在 CREATE SERVER 语句中注册服务器定义时为远程服务器指定的名称。如果该 SELECT 语句返回了一个计数，则表示正确设置了服务器定义和用户映射。

2. 如果 SELECT 语句返回了错误，则可能需要:

- 检查 Sybase 服务器以确保为入局连接配置了 Sybase 服务器。
- 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 Sybase 服务器的连接是有效的。在需要时，改变用户映射，或者创建另一个用户映射。

- 检查 DB2 联合服务器上的 Sybase 客户机软件，以确保正确安装和配置了该软件以便连接至 Sybase 服务器。
- 检查 DB2 联合变量的设置以验证它们对于 Sybase 服务器是否正确。这些变量包括系统环境变量、db2dj.ini 变量和“DB2 概要文件注册表”（db2set）变量。
- 检查服务器定义。如果需要的话，删除服务器定义，然后再次创建它。

在这一系列任务中的下一个任务是为 Sybase 表和视图注册昵称。

#### 相关任务:

- 第 317 页的『将 Sybase 数据源添加至联合服务器』
- 第 320 页的『设置和测试 Sybase 客户机配置文件』
- 第 322 页的『为 Sybase 数据源注册服务器定义』
- 第 327 页的『为 Sybase 表和视图注册昵称』
- 第 318 页的『设置 Sybase 环境变量』

#### 相关参考:

- 『ALTER USER MAPPING statement』（*SQL Reference, Volume 2*）

## 为 Sybase 表和视图注册昵称

为 Sybase 表和视图注册昵称是将 Sybase 数据源添加至联合服务器这一大型任务的一部分。

对于您定义的每个 Sybase 服务器，为想要存取每个表或视图注册昵称。当您查询 Sybase 服务器时，使用这些昵称而不是数据源对象的名称。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是您使用 CREATE NICKNAME 语句为数据源对象注册昵称时收集的。联合数据库验证数据源中的对象是否存在，然后尝试收集现有数据源的统计数据。对于优化器很有用的信息是从数据源目录中读取的，并且放置到联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息，所以，在注册昵称之前（使用与 DB2 RUNSTATS 命令等价的数据源命令）更新数据源中的统计信息。

#### 过程:

要注册昵称，发出 CREATE NICKNAME 语句。

例如:

```
CREATE NICKNAME sybase_name FOR SYBSERVER."remote_schema"."remote.table"
```

昵称的长度最多可以为 128 个字符。

对于您想为其创建昵称的每个 Sybase 表或视图重复此步骤。

当创建昵称时，联合服务器将使用您指定的信息来查询数据源目录。此查询将使用昵称来测试与数据源的连接。如果该连接不工作，则您将接收到错误消息。

#### 相关参考:

- 『RUNSTATS Command』（*Command Reference*）
- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）

- 第 328 页的『CREATE NICKNAME 语句 - Sybase 包装器的示例』

## CREATE NICKNAME 语句 - Sybase 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE NICKNAME 语句来为您想存取的 Sybase 表或视图注册昵称。

此示例说明了如何为在其中指定昵称的 Sybase 服务器指定远程对象:

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."salesdata"."europe"
```

*SYBSALES*

是 Sybase 表或视图的唯一昵称。

昵称是一个由两部分组成的名称 - 模式和昵称。如果在创建昵称时省略了模式, 则该昵称的模式将是创建该昵称的用户的认证标识。

*SYBSERVER."salesdata"."europe"*

是远程对象的由三部分组成的标识。

- *SYBSERVER* 是您在 CREATE SERVER 语句中为 Sybase 数据库服务器指定的名称。
- *salesdata* 是表或视图所属的远程模式的名称。
- *europe* 是您想存取的远程表或视图的名称。

除非用引号将 Sybase 模式和表的名称引起来, 否则, 联合服务器将把 Sybase 模式和表的名称转变为大写。

**相关任务:**

- 第 327 页的『为 Sybase 表和视图注册昵称』

**相关参考:**

- 『CREATE NICKNAME statement』 (*SQL Reference, Volume 2*)

---

## 对 Sybase 数据源的配置进行调整和故障诊断

在设置 Sybase 数据源的配置之后, 您可能想修改该配置以提高性能。

### 解决 sp\_helpindex 错误

联合系统依赖于其中一个 Sybase 目录存储过程 (sp\_helpindex)。如果接收到以下 SQL 错误, 则 Sybase 目录存储过程可能未安装在 Sybase 服务器上。

SQL0204N “sp\_helpindex” 是一个未定义的名称。

请 Sybase 管理员在 Sybase 服务器上安装目录存储过程。

**相关任务:**

- 第 317 页的『将 Sybase 数据源添加至联合服务器』

**相关参考:**

- 『db2set - DB2 Profile Registry Command』 (*Command Reference*)



---

## 第 21 章 配置对表结构文件数据源的存取

本章说明如何配置联合服务器以存取存储在表结构文件数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对表结构文件数据源的存取。

本章包含下列内容:

- 说明什么是表结构文件
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与表结构文件包装器相关联的错误消息

---

### 什么是表结构文件?

表结构文件具有常规结构, 它由一系列记录组成, 每条记录都包含相同数目的字段, 字段之间用任意定界符隔开。空值由两个相邻的定界符来表示。

以下示例显示名为 DRUGDATA1.TXT 的文件的内容。该文件包含三条记录, 每条记录包含三个字段, 字段之间用逗号隔开:

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

第一个字段是药品的唯一标识号。第二个字段是药品名称。第三个字段是药品制造商的名称。

**相关概念:**

- 第 329 页的『表结构文件的属性』
- 第 330 页的『DB2 Information Integrator 如何使用表结构文件』

**相关任务:**

- 第 331 页的『将表结构文件数据源添加至联合服务器』

---

### 表结构文件的属性

表结构文件可能已排序, 也可能未排序。表结构文件包装器搜索已排序的数据文件比搜索未排序的文件效率更高。

#### 已排序的文件

DRUGDATA1.TXT 包含已排序的记录。该文件是按第一个字段(药品的唯一标识号)排序的。此字段是主键, 因为对于每种药品来说此字段是唯一的。已排序的文件必须按升序排序。

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

## 未排序的文件

DRUGDATA2.TXT 包含未排序的记录。对于在文件中列示记录的方式未进行排序。

```
332,DrugnameB,Manufacturer2
234,DrugnameA,Manufacturer1
333,DrugnameC,Manufacturer2
```

相关概念:

- 第 329 页的『什么是表结构文件?』
- 第 330 页的『DB2 Information Integrator 如何使用表结构文件』

相关任务:

- 第 331 页的『将表结构文件数据源添加至联合服务器』

---

## DB2 Information Integrator 如何使用表结构文件

通过使用一个称为包装器的模块, DB2® Information Integrator 可以处理这样的 SQL 语句: 这些语句查询表结构文件中的数据, 就好象这些数据包含在普通的关系表或视图中一样。这样就使表结构文件中的数据能够与关系数据或者其它表结构文件中的数据连接起来。在图 27 中说明了此过程。

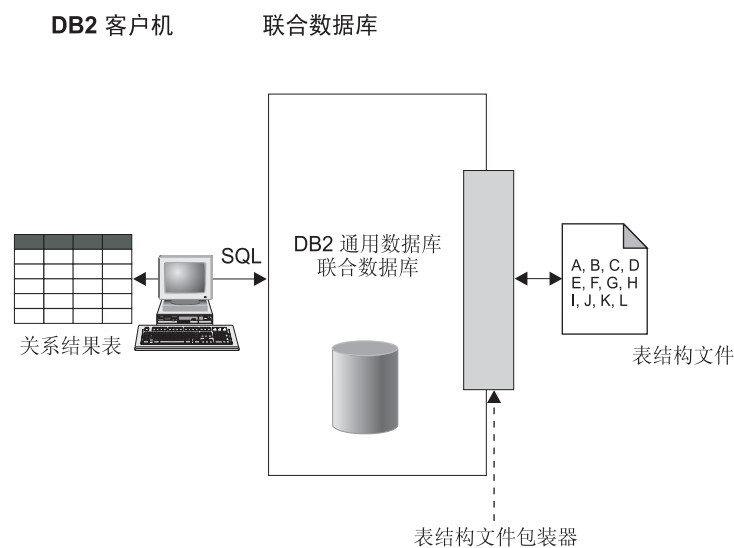


图 27. 表结构文件包装器的工作方式

例如, 假定表结构文件 DRUGDATA2.TXT 位于您的实验室中的计算机上。尝试查询此数据并使它与您使用的其它数据源中的其它表相匹配是很乏味的。

向 DB2 Information Integrator 注册 DRUGDATA2.TXT 之后, 该文件就相当于关系数据源一样。现在, 就可以同时查询该文件和其它关系数据源和非关系数据源, 并且可以同时分析它们的数据。

例如, 可以运行以下查询:

```
SELECT * FROM DRUGDATA2 ORDER BY DCODE
```

此查询将产生下列结果。

Dcode	Drug	Manufacturer
234	DrugnameA	Manufacturer1
332	DrugnameB	Manufacturer2
333	DrugnameC	Manufacturer2

相关概念:

- 第 329 页的『什么是表结构文件?』
- 第 329 页的『表结构文件的属性』

相关任务:

- 第 331 页的『将表结构文件数据源添加至联合服务器』

---

## 将表结构文件添加至联合系统

### 将表结构文件数据源添加至联合服务器

要配置联合服务器以存取表结构文件数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取表结构文件数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

过程:

要将表结构文件数据源添加至联合服务器:

1. 注册包装器。
2. 注册服务器定义。
3. 注册表结构文件的昵称。

相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』(《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

相关任务:

- 第 44 页的『创建联合数据库』
- 第 332 页的『注册表结构文件包装器』
- 第 333 页的『为表结构文件注册服务器』
- 第 333 页的『为表结构文件注册昵称』

相关参考:

- 『DB2 Information Integrator 支持的操作系统（32 位）』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 『DB2 Information Integrator 支持的操作系统（64 位）』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）

## 注册表结构文件包装器

注册表结构文件包装器是将表结构文件数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取表结构文件数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

### 过程:

要注册包装器，使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 laboratory\_flat\_files 的包装器，请发出以下语句：

```
CREATE WRAPPER laboratory_flat_files LIBRARY 'libdb2lsfile.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅表结构文件包装器库文件的列表。

在这一系列任务中的下一个任务是为表结构文件注册服务器定义。

### 相关参考:

- 第 332 页的『表结构文件包装器库文件』
- 『CREATE WRAPPER statement』（*SQL Reference, Volume 2*）

## 表结构文件包装器库文件

下表列示了表结构文件包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2lsfile.a、libdb2lsfileF.a 和 libdb2lsfileU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 84. 表结构文件客户机库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2lsfile.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2lsfile.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsfile.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsfile.so
Windows	%DB2PATH%\bin	db2lsfile.dll

| %DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows  
| 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

| 相关任务:

- | • 第 332 页的『注册表结构文件包装器』

## 为表结构文件注册服务器

为表结构文件注册服务器是将表结构文件添加至联合系统这一大型任务的一部分。在注册包装器之后，必须注册相应的服务器。

过程:

要向联合系统注册表结构文件服务器，使用 CREATE SERVER 语句。例如:

```
CREATE SERVER biochem_lab WRAPPER laboratory_flat_files
```

在此示例中，biochem\_lab 是为表结构文件服务器指定的名称。对于注册服务器的数据库，该名称必须是唯一的。

在这一系列任务中的下一个任务是为表结构文件注册昵称。

相关任务:

- 第 333 页的『为表结构文件注册昵称』

相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)

## 为表结构文件注册昵称

为表结构文件注册昵称是将表结构文件添加至联合系统这一大型任务的一部分。在注册服务器之后，必须注册相应的昵称。当您在查询中引用表结构文件数据源时就要使用昵称。

昵称以下面两种方式之一与表结构文件相关联:

- 以使用 FILE\_PATH 昵称选项的固定方式。当使用此选项时，昵称表示特定表结构文件中的数据。
- 在查询时使用 DOCUMENT 昵称列选项来指定文件名。当使用此选项时，昵称可以用来表示来自其模式与昵称定义相匹配的任何表结构文件中的数据。

限制:

如果非数字字段对于它的列类型来说太长了，则多余的数据会被截断。如果文件中的十进制字段的基数字符后面的位数超过了它的列类型的小数位参数允许的位数，则多余的数据会被截断。基数字符是由 LC\_NUMERIC “本地语言支持”类别的 RADIXCHAR 项确定的。

| 最大行长度为 10 MB (10485760 个字节)。

过程:

要注册昵称，对想要存取每个表结构文件使用 CREATE NICKNAME 语句。

在这一系列任务中没有别的任务了。

**相关任务:**

- 第 331 页的『将表结构文件数据源添加至联合服务器』
- 第 58 页的『为非关系数据源指定昵称列』

**相关参考:**

- 第 335 页的『表结构文件包装器的文件访问控制模型』
- 第 335 页的『表结构文件包装器的优化提示和注意事项』
- 第 511 页的『CREATE NICKNAME 语句语法 - 表结构文件包装器』
- 第 334 页的『CREATE NICKNAME 语句 - 表结构文件包装器的示例』

## CREATE NICKNAME 语句 - 表结构文件包装器的示例

此主题提供了使用 CREATE NICKNAME 语句来为表结构文件包装器注册昵称的完整示例。它还包括特定选项的示例。

**完整示例:**

以下示例说明了表结构文件 DRUGDATA1.TXT 的 CREATE NICKNAME 语句:

```
CREATE NICKNAME DRUGDATA1(Dcode Integer NOT NULL, Drug CHAR(20),
    Manufacturer CHAR(20))
FOR SERVER biochem_lab OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',
    COLUMN_DELIMITER ',', SORTED 'Y', KEY_COLUMN 'DCODE', VALIDATE_DATA_FILE 'Y')
```

**KEY COLUMN 选项示例:**

下列示例说明了通过在昵称语句中将 NOT NULL 选项添加至指定为键的列的定义中, 这些列就会被指定为不可空:

```
CREATE NICKNAME tox (tox_id INTEGER NOT NULL, toxicity VARCHAR(100))
FOR SERVER tox_server1
    OPTIONS (FILE_PATH'/tox_data.txt', SORTED 'Y')

CREATE NICKNAME weights (mol_id INTEGER, wt VARCHAR(100) NOT NULL)
FOR SERVER wt_server
    OPTIONS (FILE_PATH'/wt_data.txt', SORTED 'Y', KEY_COLUMN 'WT')
```

此选项是区分大小写的。但是, DB2 将把列名转换为大写(除非该列是用双引号定义的)。以下示例将不能正确工作, 这是因为 empno 列将被 DB2 转变为大写, 而 empno 键列将按小写提交。因此, 将找不到指定为键的列。

```
CREATE NICKNAME depart (
    empno char(6) NOT NULL)
FOR SERVER DATASTORE
    OPTIONS(FILE_PATH'data.txt', SORTED 'Y', KEY_COLUMN 'empno');
```

**相关任务:**

- 第 333 页的『为表结构文件注册昵称』

**相关参考:**

- 第 511 页的『CREATE NICKNAME 语句语法 - 表结构文件包装器』

---

## 表结构文件包装器的文件访问控制模型

数据库管理系统将使用 DB2 实例所有者的权限来存取表结构文件。包装器只能存取此用户标识（或组标识）可以读取的文件。应用程序的授权标识（与联合数据库建立连接所采用的标识）不相关。

在 DB2 通用数据库企业服务器版上，必须能够从每个节点中使用同一路径名来存取已经为其创建了呢称的任何表结构文件。只要能够通过公共路径从任何节点存取某个文件，该文件就不必在 DB2 通用数据库节点上。

如果网络中已经配置了 Windows NT 或 Windows 2000 域，要存取已映射的驱动器上的表结构文件，DB2 服务登录帐户必须是该域中对数据文件所驻留的共享文件夹具有存取权的一个帐户。

如果网络中没有 Windows NT 或 Windows 2000 域，并且用户在本地登录到每个工作站，那么，要存取已映射的驱动器上的表结构文件，DB2 服务登录帐户与机器上共享该文件夹的有效用户应当具有相同的用户名和密码。该用户必须处于共享文件夹的许可列表上，并且至少具有读存取权。

### 相关参考:

- 第 173 页的『Documentum 包装器的访问控制』
- 第 207 页的『Excel 包装器的文件访问控制模型』

---

## 表结构文件包装器的优化提示和注意事项

- 系统搜索已排序的数据文件比搜索未排序的文件效率更高。
- 对于已排序的文件，可以通过在提交查询时为键列指定一个值或范围来提高性能。
- 必须通过更新 SYSSTAT 和 SYSCAT 视图来手工更新表结构文件的呢称的统计信息。

### 相关参考:

- 第 107 页的『BLAST 包装器的优化提示』

---

## 表结构文件包装器的消息

本节列示和描述了您在使用表结构文件的包装器时可能会遇到的消息。

表 85. 由表结构文件的包装器发出的消息

错误代码	消息	解释
SQL0405N	数字文字 “<literal>” 无效，原因是它的值超出了范围。	数据文件中的一列或者 SQL 语句中的谓词值包含超出了该数据类型的可能范围的值。更正该数据文件，或者将该列重新定义为更合适的类型。
SQL0408N	值与它的赋值目标的数据类型不兼容。目标名称为 “<column_name>”。	数据文件中的一列包含对于该数据类型无效的字符。更正该数据文件，或者将该列重新定义为更合适的类型。

表 85. 由表结构文件的包装器发出的消息 (续)

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “数据源路径为 NULL”。)	请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “键列检索失败”。)	请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “STAT 在数据源上失败。ERRNO = <error_number>”。)	确保您具有正确的目录许可权。确保文件存在。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “找不到列信息”。)	请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “运算符不受支持”。)	请与 IBM 软件支持机构联系。
SQL1816N	包装器 “<wrapper_name>” 不能用来存取您正在尝试向联合数据库定义的数据源的 “类型” ( “<type>” ” )。	服务器类型无效。在 CREATE SERVER 语句中不应指定服务器类型。除去 TYPE 关键字和值, 然后重新运行它。
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “ERRNO = <error_number>”。相关联的文本和标记为 “不能读取文件”。	检查错误号的值。确保 DB2 实例所有者可以读取文件。然后重新运行 SQL 命令。
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “数据错误”。相关联的文本和标记为 “数据源不是标准文件”。	数据源文件是目录、套接字或 FIFO。只有标准文件才能作为数据源来存取。更改 FILE_PATH 选项, 使它指向有效文件, 然后重新发出 SQL 命令。
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “ERRNO = <error_number>”。相关联的文本和标记为 “文件打开错误”。	包装器无法打开文件。检查错误号以确定产生错误的原因。使用数据源来更正该问题, 然后重新发出 SQL 命令。
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “数据错误”。相关联的文本和标记为 “缺少键列”。	从数据源中检索到的记录缺少键字段。键列一定不能为空。更正数据, 或者使用未排序的昵称来注册文件。



表 85. 由表结构文件的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “数据错误”。相关联的文本和标记为 “文件未排序”。	文件在键列上未排序。执行下列操作之一：更改 KEY_COLUMN 选项以指向正确的列；将数据文件重新排序；或者将昵称注册为未排序的昵称。
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “数据错误”。相关联的文本和标记为 “键超过了定义大小”。	从数据源读取的键列字段大于 DB2 列定义，这可能会导致包装器搜索例程不正确地工作。更正数据或者更正昵称定义，然后重新注册该昵称。
SQL1822N	从数据源 “<server_name>” 中接收到意外的错误代码 “数据错误”。相关联的文本和标记为 “数据文件中的一行超过了 32k”。	数据文件中的一行超过了包装器允许的最大行长度。行长度不能大于 32768。缩短数据文件中该行的长度。
SQL1823N	对于服务器 “<server_name>” 中的数据类型的 “<data_type>” 不存在数据类型映射。	昵称是使用不受支持的数据类型定义的。只使用受支持的数据类型来重新定义昵称。
SQL1881N	“<option_name>” 不是 “<object_name>” 的有效 “<component>”。	所列表的值不是所列表的对象的有效选项。除去或更改无效选项，然后重新提交 SQL 语句。
SQL1882N	“昵称” 选项 “COLUMN_DELIMITER” 不能设置为 “<nickname_name>” 的 “<delimiter>”。	列定界符的长度超过了一个字符。使用单个字符重新定义该选项。然后重新运行 SQL 语句命令。
SQL1882N	“昵称” 选项 “KEY_COLUMN” 不能设置为 “<nickname_name>” 的 “<column_name>”。	没有为此昵称定义被选择作为键列的那一列。将 KEY_COLUMN 选项更正为此昵称的已排序的其中一列，然后重新发出 SQL 命令。
SQL1882N	“昵称” 选项 “VALIDATE_DATA_FILE” 不能设置为 “<nickname_name>” 的 “<option_value>”。	选项值无效。有效值为 “Y” 或 “N”。更正选项，并再次注册昵称。
SQL1883N	“<option_name>” 是 “<object_name>” 的必需 “<component>” 选项。	SQL 语句中缺少包装器的必需选项。添加必需的选项，并重新提交 SQL 语句。
SQL30090N	对于应用程序执行环境，操作无效。原因码 = “21”。	尝试进行了传递会话。表结构文件包装器不支持传递会话。

**相关概念:**

- 『消息介绍』（《消息参考》第 1 卷）

**相关参考:**

- 『SQLSTATE 消息』（《消息参考》第 2 卷）



---

## 第 22 章 配置对 Teradata 数据源的存取

本章说明如何配置联合服务器以存取存储在 Teradata 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 Teradata 数据源的存取。

本章列示配置联合服务器时需要执行的任务，包含您需要的 SQL 语句的示例，而且还提供了用于配置联合服务器的调整和故障诊断信息。

---

### 将 Teradata 添加至联合系统

#### 将 Teradata 数据源添加至联合服务器

要配置联合服务器以存取 Teradata 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Teradata 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

##### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上
- 已在联合服务器上安装和配置的 Teradata 客户机软件。
  - 要存取 Teradata 版本 V2R5, Teradata 客户机必须支持 Teradata Call-Level Interface V2 (CLIV2) R04.07 (或更新版本)。
  - 要存取 Teradata 版本 V2R3 或 V2R4, Teradata 客户机必须支持 Teradata Call-Level Interface V2 (CLIV2) R04.06 (或更新版本)。

##### 过程:

要将 Teradata 数据源添加至联合服务器:

1. 可选: 测试与 Teradata 服务器的连接。
2. 验证是否为运行时链接启用了 Teradata 库 (AIX)。
3. 为 Teradata 包装器设置环境变量。
4. 注册包装器。
5. 注册服务器定义。
6. 创建用户映射。
7. 测试从联合服务器至 Teradata 服务器的连接。
8. 为 Teradata 表和视图注册昵称。

##### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 49 页的『配置数据源的快速方法』

#### 相关任务:

- 第 44 页的『创建联合数据库』
- 第 351 页的『对 Teradata 数据源的配置进行调整和故障诊断』
- 第 340 页的『测试与 Teradata 服务器的连接』
- 第 344 页的『注册 Teradata 包装器』
- 第 345 页的『为 Teradata 数据源注册服务器定义』
- 第 347 页的『为 Teradata 数据源创建用户映射』
- 第 350 页的『为 Teradata 表和视图注册昵称』
- 第 33 页的『检查联合服务器的设置』
- 第 341 页的『设置 Teradata 环境变量』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 测试与 Teradata 服务器的连接

测试与 Teradata 服务器的连接是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

在创建包装器、服务器定义或用户映射之前，可以测试与 Teradata 服务器的连接。首先测试连接以验证是否正确设置了客户机软件并防止在发出 CREATE WRAPPER、CREATE SERVER 和 CREATE USER MAPPING 语句时出错。

可以使用“基本 Teradata 查询” (BTEQ) 实用程序来提交 SQL 查询以验证您是否可以连接至 Teradata 服务器。有关 BTEQ 实用程序的更多信息，请参阅 Teradata 文档。

#### 先决条件:

确保在 Teradata 客户机软件安装过程中已安装了 BTEQ 实用程序和“Teradata 数据连接器应用程序接口” (PIOM)。

#### 过程:

要测试与 Teradata 服务器的连接:

1. 启动 BTEQ 实用程序会话，并登录至 Teradata 服务器。
2. 发出 SQL 命令以验证您是否可以成功地连接至 Teradata 服务器。例如:

```
select count(*) from dbc.tables;
```

如果连接成功，则应当在屏幕上看到查询输出。例如:

```
*** Query completed. One row found. One column returned.
*** Total elapsed time was 1 second.
    Count(*)
```

如果连接不成功，则检查 Teradata 客户机软件以验证在联合服务器上是否正确安装和配置了该软件。

3. 从 Teradata 服务器中注销，并结束 BTEQ 实用程序会话。

在这一系列任务中的下一个任务是验证是否为运行时链接启用了 Teradata 库。

**相关任务:**

- 第 339 页的『将 Teradata 数据源添加至联合服务器』
- 第 341 页的『验证是否为运行时链接启用了 Teradata 库 (AIX)』

## 验证是否为运行时链接启用了 Teradata 库 (AIX)

验证是否为运行时链接启用了 Teradata 库是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

当您在 AIX 上将 Teradata 数据源添加至联合服务器时，在注册包装器或服务之前必须验证是否启用了运行时链接。

**过程:**

要验证是否为运行时链接启用了 Teradata 库:

1. 转至 libcliv2.so 文件所驻留的目录。

缺省情况下，安装过程将把此文件放置在 /usr/lib 目录中。

2. 发出以下 UNIX 命令。

```
dump -H libcliv2.so | grep libtli.a
```

3. 检查屏幕上出现的文件名。

如果出现文件名 libtli.a，则表示已经为运行时链接启用了 Teradata 库。

4. 如果不出现 libtli.a 文件名，则发出下列 UNIX 命令。

```
rtl_enable libcliv2.so -F libtli.a
mv libcliv2.so libcliv2.so.old
mv libcliv2.so.new libcliv2.so
chmod a+r libcliv2.so
```

这些命令将对 Teradata 库启用运行时链接。

在这一系列任务中的下一个任务是注册 Teradata 包装器。

**相关任务:**

- 第 344 页的『注册 Teradata 包装器』

## 设置 Teradata 环境变量

设置 Teradata 环境变量是将 Teradata 添加至联合服务器这一大型任务的一部分。

当安装 DB2 Information Integrator 时，安装过程将尝试在 db2dj.ini 文件中设置 Teradata 环境变量。

如果存在下列情况，则不会在 db2dj.ini 文件中设置环境变量:

- 在设置 DB2 联合服务器之后才安装 Teradata 客户机软件。
- 尚未安装 Teradata 客户机软件。

Teradata 的有效环境变量为:

- COPERR
- COPLIB
- TERADATA\_CHARSET (可选)
- NETRACE (可选)
- COPANOMLOG (可选)

必须手工设置可选环境变量。

#### 先决条件:

正确设置了联合服务器来存取数据源。这包括安装和配置任何必需的软件 (例如, 数据源客户机软件)。

#### 限制:

请参阅主题: db2dj.ini 文件的限制

#### 过程:

要自动设置必需的环境变量:

1. 在 DB2 联合服务器上安装和配置客户机软件 (如果尚未安装它的话)。
2. 设置必需的环境变量。可以通过再次运行 DB2 Information Integrator 安装来自动设置环境变量。单击启动板中的 **安装产品**, 并遵循向导中的指示信息。

要手工设置环境变量:

1. 编辑 db2dj.ini 文件。
  - 在运行 Windows 的联合服务器上, 此文件位于 sql1lib\cfg 目录中。
  - 在运行 UNIX 的联合服务器上, 此文件位于 sql1lib/cfg 目录中。

db2dj.ini 文件包含有关安装在联合服务器上的 Teradata 客户机软件的配置信息。如果该文件不存在, 则可以使用任何文本编辑器来创建名称为 db2dj.ini 的新文件。在 db2dj.ini 文件中, 必须指定变量的全限定路径, 否则将遇到错误。

2. 根据需要设置下列环境变量。

#### **COPERR**

将 COPERR 环境变量设置为 errmsg.txt 文件驻留在的目录路径。指定该变量的全限定路径 `COPERR=teradata_lib_directory`。例如:

```
COPERR=/usr/lib
```

#### **COPLIB**

将 COPLIB 环境变量设置为 libcliv2.so 文件驻留在的目录路径。指定该变量的全限定路径 `COPLIB=teradata_lib_directory`。例如:

```
COPLIB=/usr/lib
```

libcliv2.so 和 errmsg.txt 文件通常驻留在同一目录中。

## TERADATA\_CHARSET

如果不设置 TERADATA\_CHARSET 变量，则 DB2 Information Integrator 将根据数据库的代码页来检测客户机字符集。当设置了此变量时，DB2 Information Integrator 将把变量值用作客户机字符集。TERADATA\_CHARSET 变量中的值未经过验证，但是，如果未将它设置为正确的值，则远程数据源将返回错误。

在 db2dj.ini 文件中将 TERADATA\_CHARSET 环境变量设置为下列其中一个有效字符集：

在运行 UNIX 的联合服务器上设置为：

- HANGULKSC5601\_2R4
- KanjiEUC\_OU
- LATIN1\_0A
- LATIN9\_0A
- LATIN1252\_0A
- SCHGB2312\_1T0
- TCHBIG5\_1R0
- UTF8

在运行 Windows 的联合服务器上设置为：

- HANGULKSC5601\_2R4
- KanjiSJIS\_OS
- LATIN1\_0A
- LATIN1252\_0A
- SCHGB2312\_1T0
- TCHBIG5\_1R0
- UTF8

3. 要确保在联合服务器上设置了环境变量，重新启动 DB2 实例。发出下列命令来重新启动 DB2 实例：

```
db2stop  
db2start
```

### 启用和禁用 Teradata 跟踪：

Teradata NETRACE 和 COPANOMLOG 变量用来启用和禁用 Teradata 跟踪。如果需要保留发送给 Teradata 服务器的 SQL 语句的列表，可以启用 Teradata 跟踪。

要启用 Teradata 跟踪，编辑 db2dj.ini 文件，并对这些环境变量使用下列设置：

```
NETRACE=1  
COPANOMLOG=trace_file
```

*trace\_file* 是将包含跟踪数据的文件的标准名称。这些变量将只启用 Teradata 跟踪设施，而不会影响 DB2 跟踪。

要禁用 Teradata 跟踪，编辑 db2dj.ini 文件，并同时除去 NETRACE 和 COPANOMLOG 变量。

在这一系列任务中的下一个任务是注册 Teradata 包装器。

#### 相关任务:

- 第 350 页的『为 Teradata 表和视图注册昵称』

#### 相关参考:

- 第 53 页的『对于 db2dj.ini 文件的限制』
- 第 53 页的『对于 db2dj.ini 文件的限制』

## 注册 Teradata 包装器

注册 Teradata 包装器是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Teradata 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器，发出 CREATE WRAPPER 语句并指定包装器的缺省名称。

例如:

```
CREATE WRAPPER TERADATA
```

**建议:** 使用称为 TERADATA 的缺省包装器名称。当您使用缺省名称来注册包装器时，联合服务器将自动采用与该包装器名称相关联的缺省库名。

如果包装器名称与联合数据库中的现有包装器名称相冲突，则可以将缺省包装器名称替代为您选择的名称。如果您使用一个不同于缺省名称的名称，则在 CREATE WRAPPER 语句中必须包括 LIBRARY 参数。

例如，要在使用 AIX 操作系统的联合服务器上注册一个名称为 `tera_wrapper` 的包装器，请发出以下语句:

```
CREATE WRAPPER tera_wrapper LIBRARY 'libdb2teradata.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 Teradata 包装器库文件的列表。

在这一系列任务中的下一个任务是为 Teradata 包装器注册服务器定义。

#### 相关任务:

- 第 345 页的『为 Teradata 数据源注册服务器定义』

#### 相关参考:

- 第 344 页的『Teradata 包装器库文件』
- 『CREATE WRAPPER statement』 (*SQL Reference, Volume 2*)

## Teradata 包装器库文件

下表列示了 Teradata 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 `libdb2teradata.a`、`libdb2teradataF.a` 和 `libdb2teradataU.a`。



当注册包装器时，只指定该表中列示的库文件名。

表 86. Teradata 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2teradata.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2teradata.sl
Solaris	/opt/IBM/db2/V8.1/lib	libdb2teradata.so
Windows	%DB2PATH%\bin	db2teradata.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

#### 相关任务:

- 第 344 页的『注册 Teradata 包装器』

## 为 Teradata 数据源注册服务器定义

为 Teradata 数据源注册服务器定义是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

在联合数据库中，必须定义想要访问的每个 Teradata 服务器。必须首先找到 Teradata 数据源的节点名，然后在注册服务器时使用此节点名。

#### 过程:

要为 Teradata 数据源注册服务器定义:

1. 找到节点名。
  - a. 查找 hosts 文件。
- b. 搜索 hosts 文件以找出远程服务器的别名。

在 **AIX** 操作系统上，hosts 文件为 /etc/hosts。

在 **Windows** 操作系统上，hosts 文件为 x:\WINNT\system32\drivers\etc\hosts。x: 是 \WINNT 目录所驻留的驱动器。

此别名以字母字符串开头，以后缀 COP $n$  结尾。值  $n$  是与 Teradata 通信处理器相关联的应用程序处理器的数目。

- c. 在 hosts 文件中找到包含此别名的那一行。
- d. 找出该行上的第一个非数字字段。

#### 示例 hosts 文件:

```
127.0.0.1      localhost
9.22.5.77     nodexyz      nodexyzCOP1   # teradata server
9.66.111.133  rtplib05.data.xxx.com aap
9.66.111.161  rtpscm11.data.xxx.com aaprwr
9.66.111.161  rtpscm11.data.xxx.com accessm
```

在此示例中，**nodexyz** 字段是节点名。

2. 发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER server_name TYPE TERADATA VERSION 2.4 WRAPPER wrapper
OPTIONS (NODE 'node_name')
```

必须指定服务器名。指定的名称必须是唯一的。

对于所有 Teradata 服务器，必须将 TYPE 参数设置为 *TERADATA*。

Teradata 包装器支持 Teradata V2R3、V2R4 和 V2R5 的所有版本。将版本号指定为带小数点的两位数。有效版本号的示例为 2.3、2.4 和 2.5。

必须指定包装器的名称。您指定的名称必须与使用 CREATE WRAPPER 语句注册的 Teradata 包装器相对应。

还必须指定 Teradata 服务器所驻留的节点的名称。此节点名是区分大小写的。

当您注册 Teradata 服务器定义时，如果需要的话，可以在 CREATE SERVER 语句中指定更多服务器选项。

在注册服务器定义之后，可以通过发出 ALTER SERVER 语句来添加或删除服务器选项。

在这一系列任务中的下一个任务是为 Teradata 数据源创建用户映射。

相关任务:

- 第 347 页的『为 Teradata 数据源创建用户映射』

相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 346 页的『CREATE SERVER 语句 - Teradata 包装器的示例』

## CREATE SERVER 语句 - Teradata 包装器的示例

此主题提供了几个示例来说明如何使用 CREATE SERVER 语句来为 Teradata 包装器注册服务器。此主题包括一个完整的示例，该示例说明如何使用所有必需参数来创建服务器，它还包括一个带有可选的服务器选项的示例。

完整示例:

以下示例说明了如何使用 CREATE SERVER 语句来为 Teradata 包装器创建服务器定义:

```
CREATE SERVER TERASERVER TYPE TERADATA
VERSION 2.4 WRAPPER my_wrapper
OPTIONS (NODE 'tera_node');
```

服务器选项 *TERASERVER* 指定您为 Teradata 服务器指定的名称。TYPE *TERADATA* 指定您正在配置对 Teradata 数据源的存取权。VERSION 2.4 是您想访问的 Teradata 服务器软件的版本。WRAPPER *my\_wrapper* 指定您通过 CREATE WRAPPER 语句注册的 Teradata 包装器的名称。NODE '*tera\_node*' 是 Teradata 服务器所驻留的节点的名称。

服务器选项示例:

以下示例说明了具有优化器的统计信息的 Teradata 服务器定义:

```
CREATE SERVER TERASERVER1 TYPE TERADATA
VERSION 2.4 WRAPPER WRAPPERNAME1
OPTIONS (NODE 'tera_node1', CPU_RATIO '2.0', IO_RATIO '3.0');
```

在此示例中, *TERASERVER1* 是 Teradata 服务器的名称, *WRAPPERNAME1* 是您通过 `CREATE WRAPPER` 语句注册的包装器名称, 而 *'tera\_node1'* 是 Teradata 服务器所驻留的节点的名称。CPU\_RATIO 和 IO\_RATIO 服务器选项将为优化器提供下列信息:

- 联合服务器的 CPU 资源是 Teradata 服务器的 CPU 资源的两倍。
- 联合服务器的 I/O 设备处理数据的速度是 Teradata 服务器的 I/O 设备的三倍。

**相关任务:**

- 第 345 页的『为 Teradata 数据源注册服务器定义』

**相关参考:**

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)

## 为 Teradata 数据源创建用户映射

为 Teradata 数据源创建用户映射是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

当您尝试访问 Teradata 服务器时, 联合服务器将使用对于数据源有效的用户标识和密码来与该数据源建立连接。必须定义每个联合服务器用户标识和密码与相应的数据源用户标识和密码之间的关联 (用户映射)。为每个将访问联合系统的用户标识创建用户映射, 以便将分布式请求发送至 Teradata 数据源。

必须为您在相关联的 `CREATE SERVER` 语句中注册的每个 Teradata 服务器创建用户映射。

**过程:**

要将联合用户标识映射至 Teradata 服务器用户标识和密码, 发出 `CREATE USER MAPPING` 语句。

例如:

```
CREATE USER MAPPING FOR USERID SERVER TERASERVER
OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

另外, 可以通过使用 DB2 控制中心的“创建用户映射”窗口来创建用户映射。

在这一系列任务中的下一个任务是测试从联合服务器至 Teradata 服务器的连接。

**相关任务:**

- 第 348 页的『测试从联合服务器至 Teradata 服务器的连接』

**相关参考:**

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)
- 第 348 页的『CREATE USER MAPPING 语句 - Teradata 包装器的示例』

## CREATE USER MAPPING 语句 - Teradata 包装器的示例

此主题提供了一些示例来说明如何使用 CREATE USER MAPPING 语句来将本地联合用户标识映射至 Teradata 服务器用户标识和密码。此主题包括一个具有所有必需参数的完整示例，还包括一个说明如何将 DB2 专用寄存器 USER 与 CREATE USER MAPPING 语句配合使用的示例。

### 完整示例:

以下示例说明如何将本地联合用户标识 (*MICHAEL*) 映射至 Teradata 服务器用户标识和密码 (*'mike'* 和 *'passxyz123'*) :

```
CREATE USER MAPPING FOR MICHAEL SERVER TERASERVER
  OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123')
```

选项 *MICHAEL* 指定联合用户标识，该联合用户标识被映射至在 Teradata 服务器定义的用户标识。SERVER *TERASERVER* 指定您在 CREATE SERVER 语句中定义的 Teradata 服务器的名称。REMOTE\_AUTHID *'mike'* 是 Teradata 服务器的用户标识，您正在将称为 *MICHAEL* 的本地用户标识映射至这个用户标识。REMOTE\_PASSWORD *'passxyz123'* 是与 REMOTE\_AUTHID 值 *'mike'* 相关联的密码。

### 专用寄存器示例:

以下示例说明了包括专用寄存器 USER 的 CREATE USER MAPPING 语句:

```
CREATE USER MAPPING FOR USER SERVER TERASERVER
  OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123')
```

可以使用 DB2 专用寄存器 USER 来将正在发出 CREATE USER MAPPING 语句的人员的授权标识映射至在 REMOTE\_AUTHID 用户选项中指定的数据源授权标识。

### 相关任务:

- 第 347 页的『为 Teradata 数据源创建用户映射』

### 相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)

## 测试从联合服务器至 Teradata 服务器的连接

测试从联合服务器至 Teradata 服务器的连接是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

可以使用您定义的服务器定义和用户映射来测试从联合服务器至 Teradata 服务器的连接。

### 过程:

要测试连接:

1. 从 DB2 命令行处理器中，打开传递会话以便对 Teradata 系统表发出 SQL SELECT 语句。

例如:

```
SET PASSTHRU server_name
SELECT count(*) FROM dbc.tables
SET PASSTHRU RESET
```

如果该 SQL SELECT 语句返回了一个计数，则表示正确设置了服务器定义和用户映射。

2. 如果 SQL SELECT 语句返回了错误，则可能需要：

- 检查 Teradata 服务器以确保为入局连接配置了 Teradata 服务器。
- 检查用户映射以确保 REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项的设置对于与 Teradata 服务器的连接是有效的。在需要时，改变用户映射，或者创建另一个用户映射。
- 检查 DB2 联合服务器上的 Teradata 客户机软件，以确保正确安装和配置了该软件以便连接至 Teradata 服务器。
- 检查 DB2 联合变量的设置以验证您是否可以访问 Teradata 服务器。这些变量包括系统环境变量、db2dj.ini 变量和“DB2 概要文件注册表”（db2set）变量。
- 检查服务器定义。如果需要的话，删除服务器定义，然后再次创建它。

当您启动传递会话以便对 Teradata 对象发出 SQL 语句时，不能提交带有 INTO 参数的 SQL PREPARE 语句（如果该语句中包含主变量的话）。

在这一系列任务中的下一个任务是为 Teradata 表和视图注册昵称。

#### 相关任务：

- 第 339 页的『将 Teradata 数据源添加至联合服务器』
- 第 340 页的『测试与 Teradata 服务器的连接』
- 第 345 页的『为 Teradata 数据源注册服务器定义』
- 第 350 页的『为 Teradata 表和视图注册昵称』
- 第 341 页的『设置 Teradata 环境变量』

#### 相关参考：

- 『ALTER USER MAPPING statement』（*SQL Reference, Volume 2*）

## 联合服务器上的 Teradata 昵称

在您定义的每个 Teradata 服务器上，必须为您想存取的每个 Teradata<sup>®</sup> 表和视图创建昵称。当您查询 Teradata 服务器时，使用这些昵称而不是数据源对象的名称。

联合服务器使用您用 CREATE NICKNAME 语句指定的昵称来连接至 Teradata 数据源。然后，联合服务器查询数据源目录并验证与数据源的连接。如果该连接不工作，则 DB2<sup>®</sup> 将生成错误消息。

联合数据库依赖已指定昵称的对象的目录统计信息来优化查询处理。这些统计信息是当您为数据源对象创建昵称时收集的。

联合数据库验证数据源中的对象是否存在，然后尝试从该数据源中收集现有的统计数据。对于优化器很有用的信息是从数据源目录中读取的，并且放置到联合服务器上的全局目录中。因为优化器可能要使用某些或所有数据源目录信息，所以，在创建昵称之前要更新数据源中的统计信息。通过使用与 DB2 **RUNSTATS** 命令等价的命令或实用程序来更新数据源中的这些统计信息。

不能对引用了可更新的 Teradata 视图的昵称提交 SQL INSERT、UPDATE 或 DELETE 语句，除非该 SQL 语句可以完全下推到 Teradata 数据源中。

**相关任务:**

- 第 350 页的『为 Teradata 表和视图注册昵称』

**相关参考:**

- 『RUNSTATS Command』（*Command Reference*）
- 第 350 页的『CREATE NICKNAME 语句 - Teradata 包装器的示例』

## 为 Teradata 表和视图注册昵称

为 Teradata 表和视图注册昵称是将 Teradata 数据源添加至联合服务器这一大型任务的一部分。

对于您定义的每个 Teradata 服务器，为想要存取的每个表和视图注册昵称。

**过程:**

要注册昵称，发出 CREATE NICKNAME 语句。

例如:

```
CREATE NICKNAME TERANICKNAME FOR TERASERVER."remote_schema"."remote.table"
```

**建议:** 因为联合数据库使用已指定了昵称的对象的路径统计信息来优化查询处理，所以，在注册昵称之前在 Teradata 数据源中更新统计信息。可以使用与 DB2 RUNSTATS 命令等价的命令或实用程序。

昵称的长度最多可以为 128 个字符。

当您发出 CREATE NICKNAME 语句时可以指定 NUMERIC\_STRING 列选项。还可以使用 ALTER NICKNAME 语句来指定此列选项。

**相关概念:**

- 第 349 页的『联合服务器上的 Teradata 昵称』

**相关参考:**

- 『RUNSTATS Command』（*Command Reference*）
- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 350 页的『CREATE NICKNAME 语句 - Teradata 包装器的示例』

## CREATE NICKNAME 语句 - Teradata 包装器的示例

此主题提供了一个示例来说明如何使用 CREATE NICKNAME 语句来为您想存取的 Teradata 表或视图注册昵称。

此示例说明了如何为在其中指定昵称的 Teradata 服务器指定远程对象:

```
CREATE NICKNAME TERASALES FOR TERASERVER."salesdata"."europe"
```

*TERASALES* 是您为 Teradata 表或视图指定的唯一昵称。昵称是一个由两部分组成的名称：模式和实际的昵称。如果在创建昵称时省略了模式，则 DB2 将使用您的认证标识作为模式来创建昵称。

*TERASERVER."salesdata"."europe"* 为远程对象指定由三部分组成的标识：

- *TERASERVER* 是您在 CREATE SERVER 语句中为 Teradata 数据库服务器指定的名称。
- *salesdata* 是表或视图所属的远程模式的名称。
- *europe* 是您想存取的远程表或视图的名称。

相关概念：

- 第 349 页的『联合服务器上的 Teradata 昵称』

相关任务：

- 第 350 页的『为 Teradata 表和视图注册昵称』

相关参考：

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）

---

## 对 Teradata 数据源的配置进行调整和故障诊断

在设置 Teradata 数据源的配置之后，可以更改该配置以提高性能和消除潜在的错误。

### 对昵称的 UPDATE 或 DELETE 操作错误

缺省情况下，在 Teradata 数据源表上不会唯一地标识各行。当您试图更新或删除与 Teradata 表或 Teradata 视图相关联的昵称时，可能会接收到 SQL30090N, RC="21" 错误。如果发生了 SQL30090N, RC="21" 错误，则对所更新或删除的 Teradata 表至少创建一个唯一索引，然后再次尝试执行该操作。

### 调整和禁用 Teradata 存取记录

Teradata 产品提供了一个存取记录功能部件，当 Teradata 检查一个或多个数据库上的各种用户的特定安全性特权时，该功能部件就会生成日志条目。尽管存取记录提供了大量的有意义的安全性信息，但是，此功能部件显著地增加了处理器的使用率，并且可能会降低系统性能。

如果需要提高系统性能，则应评估您为存取记录定义的检查特权规则。然后，通过定义 END LOGGING 语句来终止任何不需要的规则。

为了获得最佳性能，应关闭所有存取记录。删除 **Teradata DBC.AccLogRules** 宏，然后强制重新设置“可信并行应用程序”（TPA）以便彻底停止存取日志记录。

有关更多信息，请参阅 Teradata 文档。

### 对 libcliv2.so 启用运行时链接（AIX）

如果您运行 `djxlinkTeradata.sh` 文件来链接至称为 `libcliv2.so` 的 Teradata 共享库，则当您发出 CREATE NICKNAME 语句时可能会接收到错误消息。

您可能会接收到的错误消息的一个示例为：

DB21034E 将命令当作 SQL 语句进行了处理，原因是该命令不是有效的“命令行处理器”命令。在进行 SQL 处理期间，它将返回：  
SQL30081N 检测到了通信错误。  
所使用的通信协议为：“TCP/IP”。  
所使用的通信 API：“SOCKETS”。  
检测到错误的位置：“9.112.26.28”。  
检测该错误的通信功能：“recv”。  
特定于协议的错误代码：“\*”、“\*”和“0”。  
SQLSTATE=08001

如果接收到错误消息，则检查 /sql1lib/db2dump 目录以获取任何陷阱文件。陷阱文件名以字母 t 开头，以后缀 000 结束。例如：

```
t123456.000
```

检查陷阱文件中的跟踪信息，以获取指示导致联合服务器停止的 OsCall 函数的任何 OsCall 函数引用。

以下示例说明了在陷阱文件中可能会找到的跟踪信息以及 OsCall 函数引用：

```
*** 启动堆栈回溯 ***  
  
0x239690E0 OsCall + 0x28C  
0x23973FB0 mtdpassn + 0x8A4  
0x239795A4 mtdp + 0x208  
0x2395A928 MTDPIO + 0x28C  
0x239609C4 CLICON + 0xD50  
0x23962350 DBCHCL + 0xC4
```

如果您在其中一个陷阱文件中找到了 OsCall 函数引用，则发出下列 UNIX 命令：

```
cd /usr/lib  
rtl_enable libcliv2.so -F libtli.a  
mv libcliv2.so libcliv2.so.old  
mv libcliv2.so.new libcliv2.so  
chmod a+r libcliv2.so
```

这些命令对 libcliv2.so 共享库启用运行时链接。

#### 相关任务:

- 第 339 页的『将 Teradata 数据源添加至联合服务器』
- 第 341 页的『验证是否为运行时链接启用了 Teradata 库 (AIX)』

#### 相关参考:

- 『db2set - DB2 Profile Registry Command』 (*Command Reference*)



## 第 23 章 配置对 Web 服务数据源的存取

本节中的信息说明如何将 Web 服务数据源添加至联合系统。

### Web 服务包装器和 Web 服务描述语言文档

Web 服务提供程序是由“Web 服务描述语言”（WSDL）文档描述的。可以使用 Web 服务包装器来访问 Web 服务提供程序。如图 28 中所示，Web 服务提供程序将实现服务并将接口发布至服务代理，例如 UDDI。然后，服务请求者就可以使用服务代理来查找 Web 服务。当请求者查找服务时，请求者将绑定至服务提供程序，以便请求者可以使用 Web 服务。请求者通过在请求者与提供程序之间交换 SOAP（简单对象访问协议）消息来调用服务。

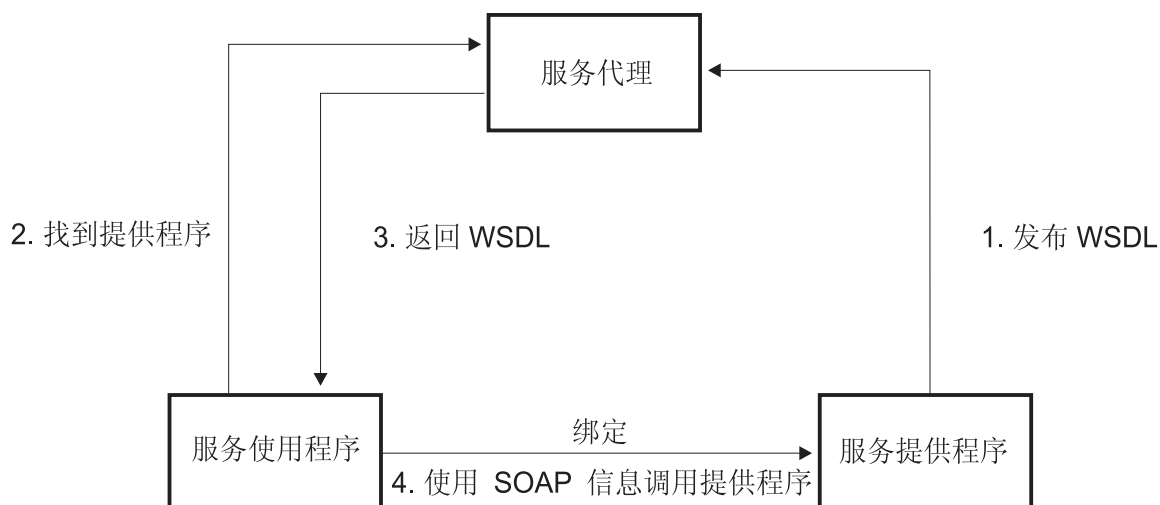


图 28. Web 服务：面向服务的体系结构

SOAP 规范定义基于 XML 的消息的布局。SOAP 消息包含在 SOAP 包络中。包络由可选的 SOAP 头和必需提供的 SOAP 主体组成。SOAP 头可以包含关于消息的信息，例如，加密信息或认证信息。SOAP 主体包含消息。SOAP 规范还定义编程语言绑定的缺省编码（称为 SOAP 编码）。

Web 服务的关键是 WSDL 文档。WSDL 文档是一种 XML 文档，它利用它发送和接收的消息来描述 Web 服务。消息是使用类型系统（通常为 XML 模式）来描述的。Web 服务操作使消息交换模式与一条或多条消息相关联。消息交换模式标识发送或接收的消息的顺序或基数，还标识消息在逻辑上发送给谁或者从何处接收。接口将操作集中在一起，但是对传输或有线格式没有任何保证。WSDL 绑定为一个或多个接口指定传输和有线格式详细信息。端点是网络地址与绑定相关联。服务将实现通用接口的端点集中起来。消息可以包含面向文档的信息或者面向过程的信息（也称为远程过程调用（RPC））。WSDL 文档可以包含一种或多种 Web 服务。

第 355 页的图 29 中的示例显示提供股票行情的简单服务的 WSDL 定义。Web 服务支持称为 GetLastTradePrice 的简单操作。可以使用基于 HTTP 的 SOAP 1.1 协议来访问

| 服务。请求读取股票报价机中的股票代码作为输入（它是字符串数据类型），返回值为价格（它是浮点数据类型）。字符串和浮点数据类型是 XML 模式标准中预定义的类型。Web 服务还可以定义数据类型和使用消息中那些用户定义的数据类型。预定义的数据类型和用户定义的 XML 数据类型将映射至昵称中的列。W3C Web 站点提供了完整的示例和 WSDL 规范。

---

```

<?xml version="1.0"?>
<definitions name="StockQuote"
...

<types>
  <schema targetNamespace="http://example.com/stockquote.xsd"
    xmlns="http://www.w3.org/2000/10/XMLSchema">
    <element name="TradePriceRequest">
      <complexType>
        <all>
          <element name="tickerSymbol" type="string"/>
        </all>
      </complexType>
    </element>
    <element name="TradePrice">
      <complexType>
        <all>
          <element name="price" type="float"/>
        </all>
      </complexType>
    </element>
  </schema>
</types>

<message name="GetLastTradePriceInput">
...
</message>

<portType name="StockQuotePortType">
  <operation name="GetLastTradePrice">
    <input message="tns:GetLastTradePriceInput"/>
    <output message="tns:GetLastTradePriceOutput"/>
  </operation>
</portType>

<binding name="StockQuoteSoapBinding"
  type="tns:StockQuotePortType">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetLastTradePrice">
    <soap:operation soapAction="http://example.com/GetLastTradePrice"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>

<service name="StockQuoteService">
  <documentation>My first service</documentation>
  <port name="StockQuotePort" binding="tns:StockQuoteBinding">
    <soap:address location="http://example.com/stockquote"/>
  </port>
</service>
</definitions>

```

---

图 29. WSDL 文档示例

Web 服务包装器使用与 HTTP 传输方法具有 SOAP 绑定的端口类型中的操作。操作中的输入消息以及相关类型或元素成为昵称中的列。操作中的输出消息被抽取到昵称层次结构中。可以为 WSDL 文档中的每个操作创建单独的昵称层次结构。

图 30 使用一个 WSDL 文档，该 WSDL 文档包含具有操作名称 GETTEMP 的 portType。对于此 Web 服务，输入邮政编码作为输入，然后接收该邮政编码所属地区的温度。

```
<?xml version="1.0"?>
<definitions name="TemperatureService" targetNamespace=http://www.xmethods.net/
  sd/TemperatureService.wsdl"
  xmlns:tns="http://www.xmethods.net/sd/TemperatureService.wsdl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
  <message name="getTempRequest">
    <part name="zipcode" type="xsd:string"/>
  </message>
  <message name="getTempResponse">
    <part name="return" type="xsd:float"/>
  </message>
  <portType name="TemperaturePortType">
    <operation name="getTemp">
      <input message="tns:getTempRequest"/>
      <output message="tns:getTempResponse"/>
    </operation>
  </portType>
  <binding name="TemperatureBinding" type="tns:TemperaturePortType">
    <soap:binding style="rpc"
      transport="http://schemas.xmlsoap.org/soap/http" />
    <operation name="getTemp">
      <soap:operation soapAction="" />
      <input>
        <soap:body use="encoded" namespace="urn:xmethods-Temperature"
          encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
      </input>
      <output>
        <soap:body use="encoded" namespace="urn:xmethods-Temperature"
          encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
      </output>
    </operation>
  </binding>
  <service name="TemperatureService">
    <documentation>
      Returns current temperature in a given U.S. zipcode
    </documentation>
    <port name="TemperaturePort" binding="tns:TemperatureBinding">
      <soap:address
        location="http://services.xmethods.net:80/soap/servlet/rpcrouter" />
    </port>
  </service>
</definitions>
```

图 30. GETTEMP Web 服务

输入值是由 zipcode 列描述的。输出值是由 return 列描述的。在 WSDL 文档中，这些列是在消息元素中标识的。消息元素表示在 Web 服务提供程序与 Web 服务使用程序之间发送的数据的逻辑定义。如果需要提供有关消息元素中的信息的更多解释，则 WSDL 文档还可以包含类型元素。类型元素可以表示基于 XML 模式规范的预定义类型或者是由用户定义的类型。

图 31 显示了 DB2® 控制中心“发现”工具从 WSDL 文档中生成的昵称。由于昵称 TEMPLATE 语法, zipcode 列是必需的输入列:

```
CREATE NICKNAME GETTEMP (
  ZIPCODE VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  RETURN VARCHAR (48) OPTIONS(XPATH './return/text()')
)
FOR SERVER "EHPWSSERV"
  OPTIONS(URL 'http://services.xmethods.net:80/soap/servlet/rpcrouter',
    SOAPACTION ' ',
    TEMPLATE '<soapenv:Envelope>
      <soapenv:Body>
        <ns2:getTemp>
          <zipcode>&zipcode[1,1]</zipcode>
        </ns2:getTemp>
      </soapenv:Body>
    </soapenv:Envelope>',
    XPATH '/soapenv:Envelope/soapenv:Body/*',
    NAMESPACES ' ns1="http://www.xmethods.net/sd/TemperatureService.wsdl",
      ns2="urn:xmethods-Temperature" ,
      soapenv="http://schemas.xmlsoap.org/soap/envelope/"');
```

图 31. GETTEMP 昵称

Web 服务包装器昵称选项 URL 和 SOAPACTION 能够覆盖您在创建昵称时所指定的端点或地址。当在查询中使用启用了 URLCOLUMN 或 SOAPACTIONCOLUMN 的列时, 可以对相同的昵称使用动态地址。如果您在创建昵称时定义了昵称选项 URL 和 SOAPACTION, 并且对列选项启用了 URLCOLUMN 和 SOAPACTIONCOLUMN, 则您就在使用 Web 服务包装器的延迟绑定功能。SOAPACTION 昵称选项的值就成为 HTTP 头中的属性。URL 昵称选项的值是将请求发送至的 HTTP URL。

URL 和 SOAPACTION 昵称选项提供了动态昵称关联。如果由几个公司来实现 Web 服务 portType, 则这些动态地址很有用。Web 服务包装器要求 WSDL 文档之间的唯一区别是 URL 和 SOAPACTIONS 不同。可以使用延迟绑定功能来为不同公司可能想使用的不同服务端点创建和使用相同的昵称。URL 和 SOAPACTION 值是从 WSDL 文档中派生出来的。

以下示例显示可以如何使用 URLCOLUMN 和 SOAPACTIONCOLUMN 列选项:

```
CREATE NICKNAME GetPartQuote(
  partnumber INTEGER OPTIONS (TEMPLATE '&column'),
  price FLOAT OPTIONS (XPATH './price'),
  urlcol VARCHAR(100) OPTIONS (URLCOLUMN 'Y'),
  soapactioncol VARCHAR(100) OPTIONS (SOAPACTIONCOLUMN 'Y'),
  FOR SERVER myServer
  OPTIONS (
    ...
    SOAPACTION 'http://example.com/GetPartPrice' ,
    URL 'http://mycompany.com:9080/GetPartPrice',
    ...
  )
```

图 32. GetPartQuote 昵称

以下示例使用在启用了 URLCOLUMN 列选项并且启用了 SOAPACTIONCOLUMN 列选项的情况下定义的 URLCOL 和 SOAPACTIONCOL 列:

```
SELECT * FROM supplier_endpoints p,  
       GetPartQuote q  
WHERE partnumber=1234 AND  
       p.url=q.urlcol AND  
       p.soapaction=q.soapactioncol;
```

SQL 应用程序可以将选择要使用的端点这项工作延迟到运行查询时才进行, 无须在创建昵称时定义特定端点。

Web 服务包装器可以将大量的 WSDL 文档数据分成很多片段, 从而减少使用的总内存量。当创建 Web 服务昵称时, 在 DB2 控制中心的“属性”窗口的“设置”页中指定 **STREAMING** 选项。Web 服务包装器处理获得的 XML 数据流, 并抽取由查询片段请求的信息。Web 服务包装器一次只对一个片段进行语法分析。只使用 **STREAMING** 选项来对大型 XML 文档进行语法分析。

#### 相关概念:

- 『DADX 文件中的 WSDL』 (《IBM DB2 Information Integrator 应用程序开发者指南》)
- 『Web 服务描述语言』 (《IBM DB2 Information Integrator 应用程序开发者指南》)

#### 相关任务:

- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 359 页的『注册 Web 服务包装器』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考:

- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 381 页的『Web 服务包装器的消息』

---

## 将 Web 服务添加至联合系统

### 将 Web 服务数据源添加至联合服务器

要配置联合服务器以存取 Web 服务数据源, 必须为联合服务器提供有关想要存取的数据源和对象 (例如, 有效的 Web 服务描述语言 (WSDL) 文档) 的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 Web 服务数据源。DB2 控制中心包括一个向导, 可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

#### 先决条件:

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

#### 过程:

要将 Web 服务数据源添加至联合服务器:

1. 注册包装器。

2. 注册服务器定义。
3. 可选: 创建用户映射。
4. 注册 Web 服务数据源的昵称。
5. 可选: 为 Web 服务昵称创建联合视图。

#### 相关概念:

- 『DB2 Information Integrator 安装过程 - 概述』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 381 页的『Web 服务包装器安全性』
- 第 381 页的『Web 服务包装器安全性』
- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

#### 相关任务:

- 『在安装了 DB2 Information Integrator 之后编辑 Oracle genclntsh 脚本并创建 libclntsh 文件 (HP-UX、Linux 和 Solaris)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 359 页的『注册 Web 服务包装器』
- 第 360 页的『为 Web 服务数据源注册服务器定义』
- 第 361 页的『为 Web 服务数据源注册昵称』
- 第 368 页的『为 Web 服务昵称创建联合视图』
- 第 44 页的『创建联合数据库』
- 第 359 页的『注册 Web 服务包装器』

#### 相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 381 页的『Web 服务包装器的消息』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## 注册 Web 服务包装器

注册 Web 服务包装器是将 Web 服务数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 Web 服务数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

#### 过程:

要注册包装器, 使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如, 要在使用 Windows 操作系统的联合服务器上注册一个名称为 websr\_wrapper 的包装器, 请发出以下语句:

```
CREATE WRAPPER websr_wrapper LIBRARY 'db2ws.d11';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称，请参阅 Web 服务包装器库文件的列表。

在这一系列任务中的下一个任务是为 Web 服务包装器注册服务器定义。

**相关概念:**

- 第 381 页的『Web 服务包装器安全性』
- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

**相关任务:**

- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 360 页的『为 Web 服务数据源注册服务器定义』

**相关参考:**

- 第 360 页的『Web 服务包装器库文件』

## Web 服务包装器库文件

下表列示了 Web 服务包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时，将把三个库文件添加到该表中列示的目录路径中。例如，如果联合服务器正在 AIX 上运行，则添加到目录路径中的包装器库文件为 libdb2ws.a、libdb2wsF.a 和 libdb2wsU.a。

当注册包装器时，只指定该表中列示的库文件名。

表 87. Web 服务包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2ws.a
Windows	%DB2PATH%\bin	db2ws.dll

%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

**相关任务:**

- 第 359 页的『注册 Web 服务包装器』

## 为 Web 服务数据源注册服务器定义

为 Web 服务数据源注册服务器定义是将 Web 服务添加至联合系统这一大型任务的一部分。

在注册包装器之后，必须注册相应的服务器。必须为想要访问的每个 Web 服务注册服务器定义。

可以从 DB2 命令行或 DB2 控制中心注册服务器定义。

**过程:**

要为 Web 服务包装器向联合系统注册服务器定义，发出 CREATE SERVER 语句。



例如，要在 Windows 上注册一个名称为 `ws_server` 的 Web 服务服务器定义，发出以下语句：

```
CREATE SERVER ws_server WRAPPER websr_wrapper;
```

在这一系列任务中的下一个任务是为 Web 服务数据源注册昵称。

#### 相关任务：

- 第 359 页的『注册 Web 服务包装器』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考：

- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## 为 Web 服务数据源注册昵称

### 为 Web 服务数据源注册昵称

为 Web 服务数据源注册昵称是将 Web 服务添加至联合系统这一大型任务的一部分。

为每个 Web 服务操作创建一个昵称层次结构。Web 服务操作是在 Web 服务描述语言（WSDL）文档中定义的。父昵称至少包含一个子昵称。子昵称对应于嵌套在父昵称的元素中的那些元素。

可以从 DB2 命令行或 DB2 控制中心创建昵称。

在 DB2 控制中心中，可以使用“发现”工具来快速创建昵称。“发现”工具的输入是 WSDL 文档的位置的 URL。“发现”工具通过处理 WSDL 文档来创建昵称。WSDL 文档可以包含嵌套在是在 WSDL 文件中导入的 WSDL 文件或外部 XML 模式文件中的模式定义。这些模式定义是通过使用 URL 地址导入的。

#### 先决条件：

对于用来描述您想与之进行通信的 Web 服务的有效 WSDL 文档具有存取权。

#### 限制：

- 仅支持请求 - 响应操作。
- 与 HTTP 传输方法的 SOAP 绑定是唯一受支持的绑定。
- 除了具有 SOAPACTIONCOLUMN、URLCOLUMN、PRIMARY\_KEY 或 FOREIGN\_KEY 选项的特殊列之外，必须在每一列上使用 TEMPLATE 选项或 XPATH 选项。

#### 过程：

要从 DB2 命令行为 Web 服务数据源注册昵称，发出 CREATE NICKNAME 语句。

例如，在 Windows 上要称为 GETTEMP 的 Web 服务注册昵称，发出以下语句：

```
CREATE NICKNAME GETTEMP (  
    ZIPCODE VARCHAR (48) OPTIONS(TEMPLATE '&column'),  
    RETURN VARCHAR (48) OPTIONS(XPATH './return/text()')  
)  
FOR SERVER "EHPWSSERV"
```

```

OPTIONS(URL 'http://services.xmethods.net:80/soap/servlet/rpcrouter',
        SOAPACTION ' ',
        TEMPLATE '<soapenv:Envelope>
                <soapenv:Body>
                <ns2:getTemp>
                <zipcode>&zipcode[1,1]</zipcode>
                </ns2:getTemp>
                </soapenv:Body>
                </soapenv:Envelope>',
        XPATH '/soapenv:Envelope/soapenv:Body/*',
        NAMESPACES ' ns1="http://www.xmethods.net/sd/TemperatureService.wsdl",
                   ns2="urn:xmethods-Temperature" ,
                   soapenv="http://schemas.xmlsoap.org/soap/envelope/"');

```

DB2 控制中心生成的数据定义语言 (DDL) 将所有输入元素映射至昵称层次结构中的根昵称的列。创建的昵称是从 WSDL 文档中派生出来的。

要从 DB2 控制中心为 Web 服务数据源注册昵称:

1. 展开**联合数据库对象**文件夹。
2. 展开想要为其注册昵称的**包装器**文件夹。
3. 展开**服务器定义**文件夹。
4. 展开想要为其注册昵称的**服务器**文件夹。
5. 右键单击**昵称**文件夹, 并选择**创建**。
6. 在“创建昵称”窗口中, 单击**发现**以定义可以帮助您选择数据源中的对象的搜索条件。
7. 指定 WSDL 文档, 它包含您想让 DB2 Information Integrator 用户访问的 Web 服务的定义。

WSDL 文档可以是一个本地文档, 也可以使用 URL 来指定它的位置。

8. 单击**确定**以便根据所选择的 WSDL 文档来创建昵称。

DB2 控制中心将 WSDL 文档抽取到多个创建昵称 DDL 语句中, 并提供适当的父子关系定义。创建的昵称表示在 WSDL 文档中定义的 Web 服务层次结构。

可选: 在这一系列任务中的下一个任务是为 Web 服务昵称创建联合视图。

#### 相关概念:

- 第 381 页的『Web 服务包装器安全性』

#### 相关任务:

- 第 58 页的『为非关系数据源指定昵称列』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 359 页的『注册 Web 服务包装器』

#### 相关参考:

- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 381 页的『Web 服务包装器的消息』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## 昵称级别和列级别的 TEMPLATE 选项

本主题适用于 WebSphere® Business Integration 包装器和 Web 服务包装器。

WebSphere® Business Integration 包装器和 Web 服务包装器构建 WebSphere Business Integration 适配器和 Web 服务环境所必需的 XML 文档。这些包装器在创建昵称时需要昵称级别和列级别的模板片段，也就是 CREATE NICKNAME 语句上的 TEMPLATE 选项。这些包装器在查询规划和查询执行阶段将使用此信息。

### Web 服务包装器

对于 Web 服务包装器，必需的属性和可选属性将随着 WSDL 文档中的定义以及派生列的方式不同而有所不同。可以从一个元素或者元素的属性来派生列。

- 如果列是从元素派生的，则 minOccurs 值将确定一列是否是可选的。
- 如果 minOccurs 的值等于 0，则表示该列是可选的。
- 如果 minOccurs 的值等于 1，则表示该列是必需的。
- 如果列是从元素的属性派生的，则该属性上使用的值将确定一列是否是可选的。
- 如果属性包含值 use=optional，则该列是可选的。
- 如果属性包含值 use=required，则该列是必需的。

以下示例是模式定义中与某列相关联的一种属性：

```
<xsd:complexType>
<xsd:sequence>
  <xsd:element ref="tns:ZooName"/>
  <xsd:element ref="tns:Count"/>
  <xsd:element ref="tns:LastModified"/>
  <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>
```

### WebSphere Business Integration 包装器

对于 WebSphere Business Integration 包装器，必需的列和可选列将随着应用程序和相关联的适配器不同而有所不同。需要通过为这些列指定适当的模板选项值来标识必需的输入列和可选输入列。在使用 DB2® 控制中心来创建昵称之前，必须修改 XML 模式定义文件来标记必需的输入列和可选输入列。

#### SAP BAPI

IBM® DB2 控制中心根据 XML 模式定义 (XSD) 文件中表示业务对象定义的特定标志的值来确定必需的输入列和可选输入列。

在业务对象层次结构的任何级别的元素（父业务对象或子业务对象）的 annotation 部分，appSpecificInfo 值中的 I 前缀指示业务对象定义映射至的 SAP BAPI 的导入参数。E 前缀指示 SAP BAPI 的导出参数。某些元素可以同时是 BAPI 的导入和导出参数。以下示例显示一个既是导入参数又是导出参数的元素：

```
<bx:appSpecificInfo>ICOMPANYCODE:ECOMPANYCODE</bx:appSpecificInfo>
```

前缀是由 WebSphere Business Integration 对象发现代理工具根据从 SAP 业务对象库中抽取的信息自动生成的。

如果表示导入参数（appSpecificInfo 值中的 I 前缀）的元素是使用属性 minOccurs=1 指定的，则 DB2 控制中心将在昵称定义中将该元素标识为必需的输入参数并且将元素标记为必需的输入列。对于 SAP BAPI 的必需输入参数，WebSphere Business Integration 对象发现代理工具不会自动将 minOccurs 的值设置为 1。必须参考“SAP 业务对象库”来确定您想访问的 BAPI 的所有必需输入参数。然后，必须通过将该属性手工设置为 minOccurs=1 来编辑 XML 模式文件中的相应元素。如果输入参数的 minOccurs 属性值仍然保持为缺省值 0，则 DB2 控制中心将在生成的昵称层次结构中将该列指定为可选输入列。

以下示例显示一个可选输入列：

```
<xsd:element name="Company_code" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICOMPANYCODE:</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="4" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
```

以下示例显示一个必需的输入列：

```
<xsd:element name="Company_id" minOccurs="1">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICOMPANYID:</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="true" isKey="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="4" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
```

SAP 商业应用程序的必需输入列和可选输入列是按下表中显示的语法来指定的：

表 88. SAP 输入列信息的加标志模式

SAP XSD 文件中使用的标志	必需的输入列	昵称模板中的列引用
层次结构中前缀 = 'I' 并且 minOccurs=1 的任何元素	是	&columnname[1,1]
层次结构中前缀 = 'I' 并且 minOccurs=0 的任何元素	否	&columnname[0,1]

### Siebel 和 PeopleSoft

DB2 控制中心根据元素的 annotation 的 attributeInfo 部分中是否存在 isRequired 标志以及该标志的值来确定必需的输入列和可选输入列。如果没有

| isRequired 标志，该列就不是一个输入列。WebSphere Business Integration 对象  
| 发现代理工具不会在 XSD 文件中自动生成这些标志。在使用 DB2 控制中心来  
| 生成昵称 DDL 之前，必须标识必需的输入列和可选输入列，并在 XSD 文件  
| 中适当地标记它们。

| 以下示例显示 Siebel 或 PeopleSoft 业务对象定义的 XSD 文件中的必需输入列  
| 和可选输入列的标志。  
|

---

```
<xsd:element name="sieb_ssa_Contact_Contact">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="1.0.0">
        <bx:appSpecificInfo>ON=Contact;CN=Contact</bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Id" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=Id</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false"
          isKey="true" isRequired="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
...

```

---

图 33. Siebel 业务对象定义部分 (1/2)

```

...
<xsd:element name="FirstName" minOccurs="1">
<xsd:annotation>
  <xsd:appinfo>
    <bx:boAttribute>
      <bx:appSpecificInfo>FN=First Name</bx:appSpecificInfo>
      <bx:attributeInfo isForeignKey="false" isKey="false"
        isRequired="false" />
    </bx:boAttribute>
  </xsd:appinfo>
</xsd:annotation>
<xsd:simpleType>
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="50" />
  </xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="LastName" minOccurs="1">
<xsd:annotation>
  <xsd:appinfo>
    <bx:boAttribute>
      <bx:appSpecificInfo>FN=Last Name</bx:appSpecificInfo>
      <bx:attributeInfo isForeignKey="false" isKey="false"
        isRequired="false" />
    </bx:boAttribute>
  </xsd:appinfo>
</xsd:annotation>
...

```

图 33. Siebel 业务对象定义部分 (2/2)

Siebel 和 PeopleSoft 商业应用程序的必需输入列和可选输入列是按下表中显示的语法来指定的:

表 89. Siebel 和 PeopleSoft 输入列信息的加标志模式

Siebel 和 PeopleSoft XSD 必需的输入列文件中使用的标志		昵称模板中的列引用
isRequired="true"	是	&columnname[1,1]
isRequired="false"	否	&columnname[0,1]

以下示例显示 DB2 控制中心根据标题为 *Siebel 业务对象定义部分* 的图中显示的 XSD 文件来创建的 DDL。在该图中的 XSD 文件中, isRequired 属性的值为 false。

```

CREATE NICKNAME sieb_ssa_Contact_Contact_NN(
  Id VARCHAR(15) OPTIONS(XPATH './ns1:Id/text()'),
    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50) OPTIONS(XPATH './ns1:FirstName/text()'),
    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50) OPTIONS(XPATH './ns1:LastName/text()'),
    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255) OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255) OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255) OPTIONS(XPATH './ns1:State/text()'))

```

```
FOR SERVER siebel_server
OPTIONS(XPATH '//ns1:sieb_ssa_contact_contact',
TEMPLATE '<ns1:sieb_ssa_contact_contact>
          &Id[1,1] &FirstName[0,1] &LastName[0,1]
          </ns1:sieb_ssa_contact_contact>',
BUSOBJ_NAME 'sieb_ssa_contact_contact',
NAMESPACES 'ns1="http://www.ibm.com/websphere/
            crossworlds/2002/BOSchema/sieb_ssa_contact_contact"');
```

#### 相关概念:

- 第 109 页的『WebSphere Business Integration 包装器』
- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

#### 相关任务:

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 381 页的『Web 服务包装器的消息』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## 昵称和 XPATH 表达式

本主题适用于 WebSphere® Business Integration 包装器和 Web 服务包装器。

昵称对应于 XML 文档数据的树形结构。父昵称和子昵称对应于数据树形结构的根结构和嵌套元素。这些父昵称和子昵称是通过使用 CREATE NICKNAME 语句指定的主键和外键联系起来的。

每个昵称是由表示输出值的 XPath 表达式定义的。WebSphere Business Integration 包装器和 Web 服务包装器使用 XPath 表达式来建立 XML 文档中的数据与关系表中的各行之间的对应关系。这些 XPath 表达式标识 XML 文档中的值，并确定这些值如何与每一行的各列相对应。WebSphere Business Integration 包装器和 Web 服务包装器将只是读取 XML 文档数据。这些包装器不会更新数据。XPATH 选项包含通过 SOAP 包络和 SOAP 主体标记来查找 SOAP 消息所需要的信息。getQuote 消息包含在 SOAP 包络和主体元素中。

NICKNAME 选项 XPATH 表达式指向输出元素中的重复标记。XPath 表达式确定昵称中将具有多少行或者昵称中将包含哪些行。列选项 XPATH 表达式是相对于 NICKNAME XPATH 表达式的。列选项 XPATH 标识一行中的值。子昵称中的 NICKNAME 选项 XPATH 是相对于父昵称中的 NICKNAME 选项 XPATH 表达式的。

当创建昵称时，选择用来指定该昵称与 XML 文档之间的关联的选项。为 WebSphere Business Integration 包装器创建的昵称与 XML 模式定义 (XSD) 文档相关联。为 Web 服务包装器创建的昵称与 Web 服务描述语言 (WSDL) 文档相关联。

#### 相关概念:

- 第 385 页的『什么是 XML?』
- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』
- 第 391 页的『昵称与 XML 文档之间的数据关联』

#### 相关任务:

- 第 388 页的『将 XML 添加至联合系统』
- 第 394 页的『为 XML 数据源注册昵称』
- 第 399 页的『为非根昵称 (XML 包装器) 创建联合视图』
- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 118 页的『为商业应用程序数据源注册昵称』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考:

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』
- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## 为 Web 服务昵称创建联合视图

为 Web 服务昵称创建联合视图是将 Web 服务数据源添加至联合系统这一大型任务的一部分。

可以为描述 Web 服务文档的昵称的层次结构定义联合视图。定义联合视图可以确保用来连接 Web 服务昵称层次结构的各个部分的查询可以正确运行。

#### 过程:

要为 Web 服务昵称定义联合视图:

1. 为一个或多个 Web 服务昵称定义视图。如果想连接与 Web 服务中的一种操作相关的所有昵称, 必须定义一个包括所有这些昵称的视图。
2. 在视图的 WHERE 子句中, 对通过 PRIMARY\_KEY 和 FOREIGN\_KEY 列选项相关的所有列使用连接谓词。

在以下示例中, 主键在 zooport\_getzooreport\_report\_nn 昵称中的 ooport\_getzooreport\_pk 列上。外键在 zooport\_getzooreport\_report\_report\_nn 昵称的 ooport\_getzooreport\_fkey 列上。

```
CREATE VIEW zooreport
(zoooid, zooname, number_of_zookeeper,
 lastmodified,zookeeper_id, zookeeper_name,
 fingers_left, animal_name, animal_species, animal_lot)
AS ( SELECT zoooid, report_zooname,
 report_count, report_lastmodified,
 zookeeper_id, zk.report_name, report_numberfingersleft,
```



```

a.report_name, report_species,
report_lot
FROM zooport_getzooreport_nn ,
zooport_getzooreport_report_nn as zk,
zooport_getzooreport_report_report_nn as a
WHERE zk.ooport_getzooreport_pkey=a.ooport_getzooreport_fkey
AND zooport_getzooreport_pkey=zk.ooport_getzooreport_fkey);

```

可以使用以下 SELECT 语句来获得所有昵称中的信息:

```
SELECT * FROM zooreport WHERE zooid='1';
```

在这一系列任务中没有别的任务了。

#### 相关参考:

- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

## CREATE NICKNAME 语句 - Web 服务包装器示例

当创建昵称以访问 Web 服务时, 为 Web 服务操作的输入消息中的每个值创建一个输入列, 并为 Web 服务操作的输出消息中的每个值创建一个输出列。使用昵称列选项定义来控制输入和输出列定义。

TEMPLATE 列选项指定一列是输入列。XPath 列选项指定一列是输出列。当 TEMPLATE 昵称选项包含带有括号的注释 ([1,1]) 时, 该列就是必需的输入列。当 TEMPLATE 昵称选项包含带有括号的注释 ([0,1]) 时, 该列就是可选输入列。

NAMESPACES 昵称选项是名-值对的以逗号分隔的列表, 联合系统使用它来解析用于输入和输出 XML 文档中的元素的名称空间。在消息请求中使用名称空间, 以便定义了 TEMPLATE 昵称选项中使用的前缀。NAMESPACES 昵称选项通过使用在 WSDL 或 XML 模式中定义的名称空间 URI 来解析 XPath 表达式中使用的前缀。XPath 表达式应用于从 Web 服务返回的 XML 文档。

### 示例 1: 必需的输入列

以下示例显示了一个使用名为 getQuote 的 Web 服务的昵称。该 Web 服务读取股票报价机中的股票代码, 并返回交易价格。以下 DDL 是由 DB2 控制中心中的“发现”工具创建的。

```

CREATE NICKNAME "stockquote.stockquoteport_getquote_nn" (
symbol VARCHAR (48) OPTIONS(TEMPLATE '&column'),
result VARCHAR (48) OPTIONS(XPATH './Result/text()'))
FOR SERVER "xmethods_server" OPTIONS(
URL 'http://66.28.98.121:9090/soap' ,
SOAPACTION 'urn:xmethods-delayed-quotes#getQuote' ,
TEMPLATE '<soapenv:Envelope>
<soapenv:Body>
<ns2:getQuote>
<symbol>&symbol [1,1]</symbol>
</ns2:getQuote>
</soapenv:Body>
</soapenv:Envelope>',
XPath '/soapenv:Envelope/soapenv:Body/*' ,
NAMESPACES 'ns2="urn:xmethods-delayed-quotes" ,

```

```
ns1="http://www.themindelectric.com/wsdl/  
net.xmethods.services.stockquote.StockQuote/" ,  
soapenv="http://schemas.xmlsoap.org/soap/envelope/" ');
```

昵称 `TEMPLATE` 选项将 `SYMBOL` 列指定为必需的输入列，因为它包含 `[1,1]` 标志。在昵称 `TEMPLATE` 选项中，为 Web 服务指定了完整的 SOAP 包络。 `getQuote` 输入值包含在 SOAP 包络和主体元素中。 `XPATH` 昵称选项包含通过 SOAP 包络和主体标记来查找交易价格值所需要的信息。

使用 “`stockquote.stockquoteport_getquote_nn`” 昵称来访问 Web 服务，如以下查询中所示：

```
SELECT * FROM "stockquote.stockquoteport_getquote_nn"  
WHERE symbol='IBM';
```

在此语句中必须使用谓词 `symbol='IBM'`，这是因为 `symbol` 是一个必需的输入列。相等性谓词是输入列上唯一有效的谓词。每个相等性谓词都在输入消息中设置一个值。如果输入列是可选的，则该列上不是必须具有相等性谓词。如果输入列是必需的，则发出查询时必须带有相等性谓词。在相等性表达式中可以使用文字值（例如，`IBM`）或者使用来自连接表或昵称的值。

## 示例 2: 重复元素和子昵称

以下示例使用一个名称为 `getZooReport` 的 Web 服务，该 Web 服务将生成一个有关动物园的报告。输入值为动物园标识。输出值是由以下模式描述的一个报告：

---

```

<wsdl:definitions name="Name"
    targetNamespace="http://myzoo.com"
    ...
<wsdl:types>
  <xsd:schema elementFormDefault="qualified" targetNamespace="http://myzoo.com"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element name="Animal">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:Name"/>
          <xsd:element ref="tns:Species"/>
          <xsd:element ref="tns:Lot"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="AnimalCareList">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element maxOccurs="unbounded" minOccurs="1" ref="tns:Animal"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="Count" type="xsd:string"/>
    <xsd:element name="LastModified" type="xsd:string"/>
    <xsd:element name="Lot" type="xsd:string"/>
    <xsd:element name="Name" type="xsd:string"/>
    <xsd:element name="NumberFingersLeft" type="xsd:string"/>
    <xsd:element name="Species" type="xsd:string"/>
    <xsd:element name="Zoo">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:ZooName"/>
          <xsd:element ref="tns:Count"/>
          <xsd:element ref="tns:LastModified"/>
          <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:string" use="optional"/>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ZooName" type="xsd:string"/>
    <xsd:element name="Zookeeper">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:Name"/>
          <xsd:element ref="tns:NumberFingersLeft"/>
          <xsd:element ref="tns:AnimalCareList"/>
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:string" use="optional"/>
      </xsd:complexType>
    </xsd:element>
  </xsd:schema>
</wsdl:types>
  ...

```

---

图 34. *getZooReport* Web 服务

以下是由 DB2 控制中心“发现”工具根据包含该模式的 WSDL 生成的 DDL:

|  
|

---

```

CREATE NICKNAME zooport_getzooreport_nn (
  zooid VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  zoo_id VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/@ns1:id'),
  report_zooname VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/ns1:ZooName/text()'),
  report_count VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/ns1:Count/text()'),
  report_lastmodified VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:LastModified/text()'),
  zooport_getzooreport_pkey VARCHAR (16) FOR BIT DATA NOT NULL
    OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "zooserver" OPTIONS(
  URL 'http://localhost:9080/MaelstromTest/services/ZooPort' ,
  SOAPACTION 'http://myzoo.com/getZooReport' ,
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <zooId>&zooId[1,1]</zooId>
    </soapenv:Body>
  </soapenv:Envelope>' ,
  XPATH '/soapenv:Envelope/soapenv:Body' ,
  NAMESPACE ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
    ns1="http://myzoo.com " ');

```

---

图 35. Zoo report – parent nickname – zooport\_getzooreport\_nn

---

```

CREATE NICKNAME zooport_getzooreport_report_nn (
  zooport_getzooreport_fkey VARCHAR (16)
    FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'ZOOPORT_GETZOOREPORT_NN'),
  zookeeper_id VARCHAR (48) OPTIONS(XPATH './ns1:Zookeeper/@ns1:id'),
  report_name VARCHAR (48) OPTIONS(XPATH './ns1:Zookeeper/ns1:Name/text()'),
  report_numberfingersleft VARCHAR (48)
    OPTIONS(XPATH './ns1:Zookeeper/ns1:NumberFingersLeft/text()'),
  zooport_getzooreport_pkey VARCHAR (16)
    FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "zooserver" OPTIONS(
  XPATH './ns1:Zoo' ,
  NAMESPACE ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
    ns1="http://myzoo.com" ');

```

---

图 36. Zoo 报告 – 昵称 zooport\_getzooreport\_nn 的子代

---

```

CREATE NICKNAME zooport_getzooreport_report_report_nn (
  zooport_getzooreport_fkey VARCHAR (16) FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'zooport_getzooreport_report_nn'),
  report_name VARCHAR (48)
    OPTIONS(XPATH './ns1:Animal/ns1:Name/text()'),
  report_species VARCHAR (48)
    OPTIONS(XPATH './ns1:Animal/ns1:Species/text()'),
  report_lot VARCHAR (48) OPTIONS(XPATH './ns1:Animal/ns1:Lot/text()'))
FOR SERVER "zooserver" OPTIONS(
  XPATH './ns1:Zookeeper/ns1:AnimalCareList' ,
  NAMESPACE ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
    ns1="http://myzoo.com" ');

```

---

图 37. Zoo 报告 – zooport\_getzooreport\_report\_nn 的子代

模式包括一些重复元素或者序列元素。这些重复元素成为父昵称的子昵称，如第 372 页的图 35、第 372 页的图 36 和第 372 页的图 37 中所示。例如，zooname、count、lastmodified 和 zookeeper 都是 zoo 的元素。元素 zoo 包含 0 个或多个 zookeeper 元素。根昵称 zoo 包含 zooname、count 和 lastmodified 这几列。子昵称 zookeeper 是由 DB2 控制中心“发现”工具创建的，用来描述 zookeeper 的重复元素。zookeeper 列中的第三个元素 animalcarelist 也包含 0 个或多个元素，因此，它成为子昵称 zooport\_getzooreport\_report\_report\_nn。下图显示了昵称层次结构：



图 38. 父代 → 子代 → 昵称层次结构

以下语句是一个典型查询，可以对昵称发出该查询来访问动物园报告 Web 服务。当发出此语句时，就会根据特定标识和子昵称动物园报告的主键和外键相匹配的位置来从动物园报告中检索信息。

```

SELECT * FROM zooport_getzooreport_nn ,
           zooport_getzooreport_report_nn zk ,
           zooport_getzooreport_report_report_nn a
WHERE zooid='1'AND zooport_getzooreport_pkey=zk.zooport_getzooreport_fkey
and zk.zooport_getzooreport_pkey=a.zooport_getzooreport_fkey;
  
```

### 示例 3: 延迟绑定

以下示例显示可以如何使用“延迟绑定”选项。可以从 DB2 控制中心或 DB2 命令行使用此选项。如果您定义了昵称选项 URL 和 SOAPACTION，并且在创建昵称时启用了列选项 URLCOLUMN 和 SOAPACTIONCOLUMN，则您就在使用延迟绑定功能。DB2 控制中心将创建两个列选项 URLCOLUMN 和 SOAPACTIONCOLUMN，并将这两列的值设置为 yes。

以下示例表示一个 Web 服务，该 Web 服务提供了由一个公司的所有供应商实现的各个部件的价格表。以下是一个包括 URLCOLUMN 和 SOAPACTIONCOLUMN 定义的 CREATE 语句：

```

CREATE NICKNAME GetPartQuote(
  partnumber INTEGER OPTIONS (TEMPLATE '&column'),
  price FLOAT OPTIONS (XPATH './price'),
  urlcol VARCHAR(100) OPTIONS (URLCOLUMN 'Y'),
  soapactioncol VARCHAR(100) OPTIONS (SOAPACTIONCOLUMN 'Y'),
  FOR SERVER myServer
  OPTIONS (
    ...
    SOAPACTION 'http://example.com/GetPartPrice' ,
    URL 'http://mycompany.com:9080/GetPartPrice',
    ...
  )

```

要通过单个查询来获得所有供应商的价格表，则需要每个供应商为 SOAPACTION 和 URL 列选项提供的值。该查询看起来为如下所示：

```

SELECT * FROM supplier_endpoints p,
  GetPartQuote q
  WHERE partnumber=1234 AND
        p.url=q.urlcol AND
        p.soapaction=q.soapactioncol;

```

本地表 supplier\_endpoints 包含可以用来调用 Web 服务的所有 URL 和 SOAP 地址。可以包括 ORDER BY price 子句以确定此部件的售价最低的供应商。

#### 示例 4: ESCAPE\_INPUT 列选项

可以将 XML 片段作为查询中的输入值。当注册昵称时，包括列选项 ESCAPE\_INPUT=N。此选项将维护输入值中的 XML 片段中的特殊字符，例如，< 和 >。

当模式中包含要求您将 XML 作为 SOAP 消息的一部分来发送的重复输入值时，可以使用 ESCAPE\_INPUT 列选项来通过使用正确的 XML 来构建输出消息。

例如，“动物园” Web 服务包括一种操作，该操作可以添加与该动物园管理者相关联的新的动物园管理者和一些动物。在此示例的模式中，AnimalCareList 可以具有多个动物。

```

CREATE NICKNAME add_zookeeper(
  zookeeper_id VARCHAR(48) OPTIONS(TEMPLATE '...'),
  name VARCHAR(48) OPTIONS(TEMPLATE '...'),
  numberfingersleft VARCHAR(48) OPTIONS(TEMPLATE '...'),
  animals VARCHAR(3000) OPTIONS( TEMPLATE '...' , ESCAPE_INPUT 'N')
  ...

```

要为新的动物园管理者添加两个动物，发出诸如以下示例的查询：

```

SELECT * FROM add_zookeeper
  WHERE zookeeper_ID='37' AND
        name='Amit Tsunami' AND
        numberfingersleft='3' AND
        animals='<AnimalCareList xmlns="http://myzoo.com">
          <Animal>
            <Name>Larry</Name>
            <Species>Gorilla</Species>
            <Lot>7</Lot>
          </Animal>
          <Animal>
            <Name>Bill</Name>
            <Species>Chimpanzee</Species>
            <Lot>8H</Lot>
          </Animal>
        </AnimalCareList>';

```

| add\_zookeeper 昵称是一种 Web 服务操作，该操作可以更改 Web 服务的状态或者更新  
| 信息。尽管不能更新非关系型包装器，但是，此示例中的 SELECT 语句将更新动物园  
| 信息以添加新的动物园管理者。

| 还可以将 ESCAPE\_INPUT 列选项用于使用诸如 xsd:anyType 的元素的模式。在此情况  
| 下，元素的类型是未知的。可以在输入列上将 ESCAPE\_INPUT 列选项用于该元素，以  
| 便可以为输入指定任意 XML 片段。

#### | 相关概念:

- | • 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

#### | 相关任务:

- | • 第 358 页的『将 Web 服务数据源添加至联合服务器』
- | • 第 361 页的『为 Web 服务数据源注册昵称』

#### | 相关参考:

- | • 第 381 页的『Web 服务包装器的消息』
- | • 第 378 页的『Web 服务数据源 - 示例查询』
- | • 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

---

## | 对于商业应用程序和 Web 服务的包装器的查询限制

### | Equal 谓词

| 在输入列上唯一有效的谓词是 equal 谓词。对于输出列，任何谓词都有效。

| 以下示例将返回一个错误，将发出一条消息来指示谓词在该列上不受支持。在此示例  
| 中，zipcode 列是一个输入列:

```
| SELECT return FROM gettemp WHERE zipcode<'95141'
```

| 以下示例显示了一个在输入列上使用 equal 谓词的有效查询。customers 昵称与包含客  
| 户标识的本地 DB2 UDB 表相连接。该查询在 Sales 列（这是一个“仅输出”列）上  
| 包含附加谓词。

```
| SELECT a.name, a.address  
| FROM customers a, local_table b  
| WHERE  
| a.customer_id=b.custid AND  
| a.Sales > 300000;
```

### | 必需的输入列的谓词

| 对于您引用的昵称层次结构，必须为 SQL 查询中的所有必需的输入列提供全等谓词值。  
| 对于违反此限制的所有查询，包装器将返回 SQLCODE 901。

### | IN 或 OR 谓词

| 对于 WebSphere Business Integration 包装器和 Web 服务包装器，输入列中不允许存在  
| IN 或 OR 谓词。

| 下列示例都是一些无效查询。customers 昵称具有一个必需的输入列 customer\_id:

```

| SELECT * FROM customers
|   WHERE customer_id IN (12345, 67890, 11223);
| SELECT * FROM customers
|   WHERE customer_id IN (SELECT custid FROM local_table); )

```

但是，对于 WebSphere Business Integration 包装器，如果您使用 SPECIFICATION ONLY 参数为必需的输入列定义一个唯一索引，则可以对必需的输入列使用 IN 列表谓词：

```

| CREATE UNIQUE INDEX myuindex ON customers(customer_id) SPECIFICATION ONLY;

```

### 可选输入列的连接

以下示例演示了对于连接可选输入列的限制。不能从本地表或昵称连接可选输入列。如果 WSDL 生成了一个可选的输入昵称列，而您需要在连接中使用该列，则必须通过编辑 DDL 来将该列更改为必需的输入列。

在此示例中，名为 order 的 Web 服务包装器昵称是在将 shipping\_method 作为可选输入列的情况下创建的。因为以下语句在谓词中使用了文字，因此它是一个有效查询：

```

| SELECT * FROM order
|   WHERE part="hammer" AND shipping_method="FEDEX";

```

但是，如果查询中包括一个名为 orderparts 的本地表（该表定义部件和装运方法），并且该表包含一个名为 shipping\_method 的可选列，则以下语句无效：

```

| SELECT * FROM
|   order o, orderparts op
|   WHERE
|     o.part="hammer" AND
|     o.shipping_method=op.shipping_method

```

对于 WebSphere Business Integration 包装器，昵称的可选输入列上的谓词可能会下推至 WebSphere Business Integration 适配器。DB2 UDB 可以决定在本地对从应用程序数据源访存的行应用那些谓词。为了确保始终将给定输入列的谓词下推至适配器，应将输入列声明为必需的输入列。昵称层次结构上的每个查询都必须对必需的输入列包括谓词值。

为了确保获得有效结果，互相连接的输入列必须是 Web 服务包装器的必需列。

### 外连接

使用父昵称中的主键的昵称与子昵称列中的外键之间的外连接是不受支持的。

如果 XML 文档中的父元素不包含任何子元素，而您使用了父称与子昵称之间的内连接时，将不会为该元素返回任何行。例如，对于给定的客户，如果 SAP 系统中没有他的任何银行详细信息，则对于特定客户不会为 sap\_bapi\_customer\_getdetail2\_sap\_customerbankdetail\_NN 昵称返回任何行。

下列 CREATE NICKNAME 语句将定义在示例查询中使用的列：



---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
...
NAME VARCHAR(35)
  OPTIONS(XPATH './ns3:sap_customeraddress/
            ns1:sap_customeraddress/ns1:NAME/text()'),
...
NN_PKEY VARCHAR(16) OPTIONS(PRIMARY_KEY 'YES'),
COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()'),
  TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()'),
  TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
...
FOR SERVER sap_server
OPTIONS(XPATH '//ns3:sap_bapi_customer_getdetail2',
TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
          &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
          &COMPANYCODE[0,1]
          &CUSTOMERNO[1,1]
        </ns3:sap_bapi_customer_getdetail2>',
...

```

---

图 39. *getdetail2* 昵称的一部分内容

---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()'),
  TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()'),
  TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()'),
  TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()'),
  TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()'),
  TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
NN_FKEY VARCHAR(16) OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
FOR SERVER sap_server
OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
TEMPLATE '<ns3:sap_customerbankdetail>
          <ns2:sap_customerbankdetail>
            &CUSTOMER[0,1]
            &BANK_KEY[0,1]
            &BANK_ACCT[0,1]
            &CTRL_KEY[0,1]
            &BANK_REF[0,1]
          </ns2:sap_customerbankdetail>
        </ns3:sap_customerbankdetail>',
...

```

---

图 40. 客户银行详细信息昵称的一部分内容

在以下示例中，因为两个昵称之间存在内连接条件，所以查询将不会返回任何件：

```

SELECT a.name, b.bank_key
FROM sap_bapi_customer_getdetail2_NN a,
     sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
WHERE a.customerno='1234567890'
AND a.NN_PKEY=b.NN_FKEY;

```

如果 WebSphere Business Integration 包装器或者 Web 服务包装器昵称定义中包含必需的输入列，则不支持此昵称与任何其它本地 DB2 UDB 表或其它昵称之间的左外连接。

**相关概念:**

- 第 120 页的『昵称级别和列级别的 TEMPLATE 选项』

**相关任务:**

- 第 115 页的『将商业应用程序数据源添加至联合系统』
- 第 358 页的『将 Web 服务数据源添加至联合服务器』

**相关参考:**

- 第 142 页的『商业应用程序数据源 - 示例查询』
- 第 126 页的『CREATE NICKNAME 语句 - WebSphere Business Integration 包装器示例』
- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 378 页的『Web 服务数据源 - 示例查询』

---

## Web 服务数据源 - 示例查询

### 示例 1: 使用具体查询表

使用具体查询表在本地对查询结果进行高速缓存并提高查询性能。可以使用 Web 服务数据源的昵称来创建具体查询表。对于某些查询，数据库可以自动确定具体查询表是否无须存取基本表就可以应答查询。以下过程显示如何创建和填充具体查询表:

1. 创建本地表或基本表:

```
CREATE TABLE mystocks(ticker VARCHAR(10));
```

可以使用本地表来维护您想高速缓存的所有值。

2. 将您想高速缓存的所有值插入到表中:

```
INSERT INTO mystocks VALUES('IBM');
INSERT INTO mystocks VALUES('MSFT');
...
```

3. 创建 Web 服务昵称:

```
CREATE NICKNAME stockquote_nn (
    ticker VARCHAR(40) OPTIONS (TEMPLATE '&column'),
    price VARCHAR(16) OPTIONS (XPATH './Result/text()')
)
FOR SERVER stock_server
OPTIONS (TEMPLATE '<ticker>&column</ticker>'
        XPATH './Result/text()' );
```

4. 创建由昵称和本地表组成的视图:

```
CREATE VIEW stock_quote_view (ticker, price)
AS (
    SELECT nn.ticker, nn.price
    FROM stockquote_nn nn, mystocks s
    WHERE nn.ticker=s.ticker
);
```

5. 创建具体查询表:

```

CREATE TABLE stockquote_MQT (ticker, ticker2, price)
      as (SELECT nn.ticker,s.ticker as ticker2, nn.price
      FROM stockquote_nn nn, mystocks s
      WHERE nn.ticker=s.ticker )
DATA INITIALLY DEFERRED REFRESH DEFERRED;

```

将连接谓词 (nn.ticker 和 s.ticker) 中使用的所有 VARCHAR 列包括在具体查询表输出列表中, 以便 DB2 通用数据库使用具体查询表的机会最多。

要延迟刷新具体查询表, 指定 REFRESH DEFERRED 关键字。使用 REFRESH DEFERRED 关键字指定的具体查询表不会反映对底层基本表所作的更改。使用子句 DATA INITIALLY DEFERRED, 以便不会作为 CREATE TABLE 语句的一部分来将数据插入表中。

6. 发出 REFRESH TABLE 语句来填充表。当您发出 REFRESH TABLE 语句时, 表中的数据以快照的形式来反映查询结果。以下示例将填充 stockquote\_MQT 表, 并设置当前刷新时间间隔专用寄存器的值。

```

REFRESH TABLE stockquote_MQT;

SET CURRENT REFRESH AGE any;

```

对具体查询表中的数据运行的查询比对基本表中的数据运行的查询速度更快。当您想使用具体查询表时, 参阅视图而不是昵称:

```

SELECT * FROM stock_quote_view
WHERE ticker='IBM';

```

如果您发出一个查询来选择尚未进行高速缓存的值, 则将返回 0 行。

## 示例 2: 使用主键和外键来发出连接

PRIMARY\_KEY 和 FOREIGN\_KEY 列用来定义父昵称与子昵称之间的关系。每个父昵称必须具有主键列选项。使用引用父昵称的主键列的外键列选项来定义父昵称的子昵称。一个昵称可以具有多个子昵称, 但是一个昵称只能具有一个父昵称。

因为这些列只包含二进制数据, 所以, 这些列是使用 FOR BIT DATA NOT NULL 关键字来定义的。当您创建昵称时, DB2 控制中心将生成此定义。创建昵称时, 可以将 PRIMARY\_KEY 和 FOREIGN\_KEY 列显式定义为 FOR BIT DATA NOT NULL。

以下示例显示 Web 服务包装器如何使用 PRIMARY\_KEY 和 FOREIGN\_KEY 列来使父昵称与子昵称相关联。

```

CREATE NICKNAME zooport_getzooreport_nn (
  zooid VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  zoo_id VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/@id'),
  report_zoo_zooname VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:ZooName/text()'),
  report_zoo_count VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:Count/text()'),
  report_zoo_lastmodified VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:LastModified/text()'),
  nn_pk VARCHAR (16) NOT NULL OPTIONS(PRIMARY_KEY 'YES'),
  url VARCHAR (256) OPTIONS(URLCOLUMN 'Y'),
  soapaction VARCHAR (256) OPTIONS(SOAPACTIONCOLUMN 'Y')
) FOR SERVER "mytestsvr"
OPTIONS(
  URL 'http://localhost:9080/MaelstromTest/services/ZooPort',
  SOAPACTION 'http://myzoo.com/getZooReport',
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>

```

```

                <zooId>&zooId[1,1]</zooId>
                </soapenv:Body>
            </soapenv:Envelope>',
            XPATH '/soapenv:Envelope/soapenv:Body' ,
            NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/",
                ns1="http://myzoo.com" ');

CREATE NICKNAME zooport_getzooreport_report_zookeeper_nn (
    nn_fk VARCHAR (16) NOT NULL
        OPTIONS(FOREIGN_KEY 'ZOOPORT_GETZOOREPORT_NN'),
    zookeeper_id VARCHAR (48) OPTIONS(XPATH './@id'),
    report_zookeeper_name VARCHAR (48) OPTIONS(XPATH './ns1:Name/text()'),
    zookeeper_numberfingersleft VARCHAR(48)
        OPTIONS(XPATH './ns1:NumberFingersLeft/text()'),
    nn_pk VARCHAR (16) NOT NULL OPTIONS(PRIMARY_KEY 'YES')
)
FOR SERVER "MYTESTSRVR" OPTIONS(
    XPATH './ns1:Zoo/ns1:Zookeeper' ,
    NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/",
        ns1="http://myzoo.com" ');

```

昵称 `zooport_getzooreport_report_zookeeper_nn` 中的外键 `nn_fk` 表示外键选项中的父昵称 `zooport_getzooreport_nn`。指定的主键昵称列和外键昵称列并不对应于 WSDL 文档中的数据，原因是这些昵称列包含由包装器生成的键。这些键标识仅在查询中是唯一的父昵称和子昵称之间的关系。如果子昵称包含输入列，则父昵称选项模板在昵称选项模板结构中表示该子昵称。

以下 SQL 语句将连接父昵称和子昵称：

```

SELECT *
FROM    zooport_getzooreport_nn a,
        zooport_getzooreport_report_zookeeper_nn z,

WHERE   a.nn_pk = z.nn_fk
        AND a.zooid = 100
;

```

以下描述说明 Web 服务包装器在执行查询期间如何使用 `TEMPLATE` 和 `XPATH` 昵称和列选项。并不打算将它作为特定实现的示例。

当连接主键列和外键列时，Web 服务包装器将向 Web 服务提供程序发送消息，并且将从 Web 服务提供程序返回一些列。包装器通过为列选项模板（`ZOOID VARCHAR (48) OPTIONS(TEMPLATE '&column')`）中的引用替代查询中的输入列（`a.zooid = 100`）的值，然后对昵称模板选项（`<zooId>&zooId[1,1]</zooId>`）中的所有列引用执行此操作来为父昵称生成消息。昵称模板选项可以包括列引用或子昵称引用。然后消息被发送至 Web 服务。

包装器通过对 Web 服务返回的文档应用昵称选项 `XPATH` 来生成昵称的行。如果昵称选项 `XPATH` 返回多个 XML 片段，则昵称包含多行。列 `XPATH` 选项将应用于获得的 XML 片段，这些 XML 片段表示要获得列值的那些行。如果一个昵称具有一个或多个间接父昵称，则在将昵称选项 `XPATH` 和列选项 `XPATH` 应用于此昵称之前，将按照层次结构中从上到下的顺序应用所有父昵称 `XPATH` 表达式。

#### 相关任务:

- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 359 页的『注册 Web 服务包装器』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考:

- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 381 页的『Web 服务包装器的消息』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』

---

## Web 服务包装器安全性

Web 服务包装器支持使用 HTTPS 作为 SOAP 消息的传输协议。HTTPS 是许多 Web 服务提供程序使用的一种标准加密协议。Web 服务提供程序生成的 WSDL 文档在 URL 中都包含 https://。HTTP 请求或 HTTP 响应中的 SOAP 消息都是加密的。

如果 Web 服务使用 HTTPS 作为传输协议，则 Web 服务包装器不会验证服务器为了进行标识而发送的 SSL 证书。Web 服务包装器可以使用自我签名的证书来调用 Web 服务。

Web 服务包装器支持使用 CREATE USER MAPPING 语句来进行 HTTP 认证。使用 CREATE USER MAPPING 语句来将联合服务器用户标识映射至 Web 服务用户标识和密码。包装器开发者使用用户映射语句来提供用户标识和密码，如以下示例中所示：

```
CREATE USER MAPPING
  FOR RSPALTEN SERVER S1
  OPTIONS ( REMOTE_AUTHID 'SYSTEM', REMOTE_PASSWORD 'MANAGER' )
```

当访问 S1 服务器上的 Web 服务昵称时，以 SYSTEM 作为用户标识并以 MANAGER 作为密码来发送 HTTP 请求。用户映射是可选的。如果不指定用户映射，则当 Web 服务提供程序期望获得认证信息您可能会发现错误。某些服务器可能会使用认证功能来限制对服务的访问。从 WSDL 文档中的信息并不能明显知道是否需要认证。

#### 相关概念:

- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

#### 相关任务:

- 第 358 页的『将 Web 服务数据源添加至联合服务器』
- 第 359 页的『注册 Web 服务包装器』
- 第 361 页的『为 Web 服务数据源注册昵称』

#### 相关参考:

- 第 381 页的『Web 服务包装器的消息』

---

## Web 服务包装器的消息

下表说明了当您使用 Web 服务包装器时可能会接收到的一些典型的错误消息。

表 90. 典型错误消息

错误	描述	用户响应
SQL1822N 从数据源 “wswrap.svl.ibm” 接收到意外的错误代码 “SOAP 故障”。相关联的文本和标记是 “java.lang.Exception: HTTP URI s”。SQLSTATE=560BD	Web 服务提供程序返回了 SOAP 故障。	<ul style="list-style-type: none"> <li>输入可能不正确。检查输入自变量是否正确。</li> <li>Web 服务提供程序可能存在问题。请与 Web 服务的所有者联系。</li> </ul>
SQL30081N 检测到通信错误。使用的通信协议为: “SOAP”。使用的通信 API: “HTTP”。检测到错误的位置: “localhos”。检测到错误的通信功能: “connect”。特定于协议的错误代码: “38309”、“10061”和“1”。SQLSTATE=08001	Web 服务提供程序未侦听端口或 URL。	<ul style="list-style-type: none"> <li>检查您使用的 URL 是否全部有效。</li> <li>检查端口信息是否正确。</li> <li>确保服务器正在运行。</li> </ul>
SQL30081N 检测到通信错误。使用的通信协议为: “SOAP”。使用的通信 API: “HTTP”。检测到错误的位置: “doesntexist.ibm”。检测到错误的通信功能: “*”。特定于协议的错误代码: “38308”、“*”和“0”。SQLSTATE=08001	Web 服务提供程序主机名不在域名服务器中。	确保主机名在名称服务器中。
SQL30081N 检测到通信错误。使用的通信协议为: “SOAP”。使用的通信 API: “HTTP”。检测到错误的位置: “www.ibm.com”。检测到错误的通信功能: “*”。特定于协议的错误代码: “38312”、“*”和“0”。SQLSTATE=08001	HTTP 返回码, 类似于浏览器通常会报告的 404 错误。	确定服务器是否正在将任何错误返回给应用程序。运行 DB2 UDB 跟踪来确定服务器正在返回的响应。
SQL30081N 检测到通信错误。使用的通信协议为: “SOAP”。使用的通信 API: “HTTP”。检测到错误的位置: “”。检测到错误的通信功能: “*”。特定于协议的错误代码: “38304”、“*”和“0”。SQLSTATE=08001	URL 不正确。SQLSTATE 38304 可能指示不知道协议。SQLSTATE 38305 指示 URL 中存在语法错误。	验证 WSDL 文档是否包含有效的 URL 语法和协议。
SQL1822N 从数据源 “wswrap.svl.ibm” 接收到意外的错误代码 “SAXException”。相关联的文本和标记为 “期望的标记结束 `ns0:Mi””。SQLSTATE=560BD	响应中包括 XML 输出上的语法分析错误。	验证服务器是否返回了正确的 XML 输出。运行 DB2 UDB 跟踪来确定 Web 服务提供程序返回的内容。还可以使用另一种工具来调用 Web 服务以确保 Web 服务响应有效。

**相关概念:**

- 第 353 页的『Web 服务包装器和 Web 服务描述语言文档』

**相关任务:**

- 第 358 页的『将 Web 服务数据源添加至联合服务器』

- 第 359 页的『注册 Web 服务包装器』

**相关参考:**

- 第 369 页的『CREATE NICKNAME 语句 - Web 服务包装器示例』
- 第 378 页的『Web 服务数据源 - 示例查询』
- 第 139 页的『对于商业应用程序和 Web 服务的包装器的查询限制』





---

## 第 24 章 配置对 XML 数据源的存取

本章说明如何配置联合服务器以存取存储在 XML 数据源中的数据。可以通过使用 DB2 控制中心或者通过发出 SQL 语句来配置对 XML 数据源的存取。

本章包含下列内容:

- 说明什么是 XML
- 列示您需要执行的任务
- 包含您需要的 SQL 语句的示例
- 列示与 XML 包装器相关联的错误消息

---

### 什么是 XML?

“可扩展标记语言”（XML）是结构化文档和数据的通用格式。XML 文件的文件扩展名为 xml。与 HTML 相似，XML 使用标记（用 > 和 < 括起来的词语）来构造文档中的数据。在图 41 中显示了样本 XML 文档。

---

```
<?xml version="1.0" encoding=UTF-8"?>
<doc>
  <customer id='123'>
    <name>...</name>
    <address>...</address>
    ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>
```

---

图 41. 样本 XML 文档

## XML 包装器的工作方式

XML 包装器使得能够使用 SQL 来查询下列类型的数据:

- 存储在单个文件中的外部 XML 文档
- 目录路径中的多个 XML 文件
- 使用“统一引用标识”（URI）引用的远程 XML 文件
- 存储在关系列中的 XML 文档

图 42 说明了 XML 包装器如何使用联合系统。

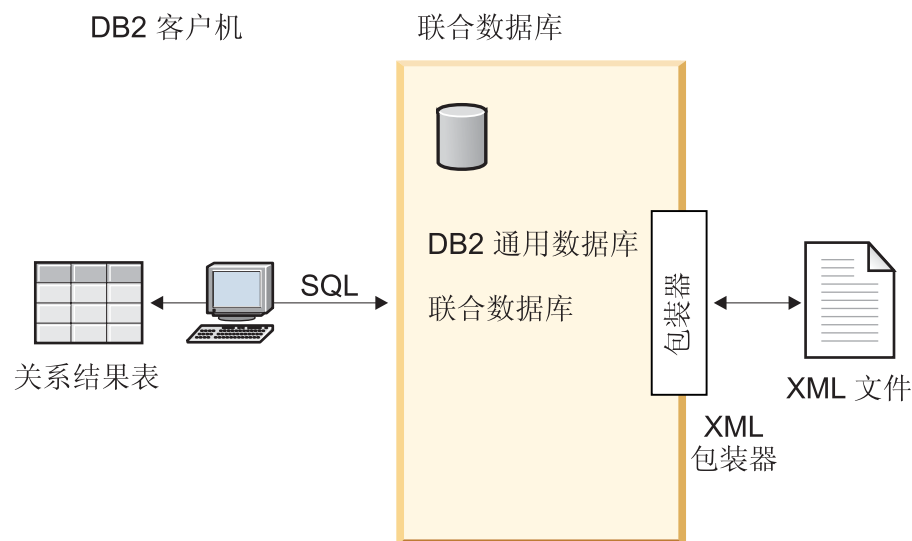


图 42. XML 包装器的工作方式

通过使用 XML 包装器，可以将 XML 数据从外部数据源映射到由一组昵称组成的关系模式中。XML 文档的结构在逻辑上等同于关系模式，在关系模式中，嵌套元素和重复元素是作为具有外键的独立表来建模的。

对应于 XML 文档的昵称组织成树形结构，在该树形结构中，子昵称映射至嵌套在与父昵称相对应的元素中的元素。

当嵌套元素发生重复或者具有复杂结构的单值标识时，可以为每个嵌套元素提供独立的昵称。

子昵称和父昵称是通过由包装器生成的主键和外键连接的。

XPath 表达式用来将 XML 文档映射到由一组昵称组成的关系模式中。XPath 是用于标识 XML 文件的各部分（例如，XML 文档树中的节点组和属性组）的寻址机制。基本 XPath 语法类似于文件系统寻址。

每个昵称是由标识用来表示个别元组的 XML 元素的 XPath 表达式以及指定如何从每个元素中抽取列值的一组 XPath 表达式来定义的。

**XML 文档映射的示例:**

以下示例说明了如何将第 385 页的图 41 中显示的样本 XML 文档映射到一组昵称中，如何使用主键和外键来建立父子关系，如何使用 XPath 表达式来定义文档的每个元素中的个别元组和列，以及在将文档注册至联合系统之后可以如何对 XML 文档运行查询。

样本 XML 文档包含一组客户（customer）元素。每个元素都包含几个订单（order）和付款（payment）元素。

订单元素包含几个商品（item）元素。

图 43 中显示了各个元素之间的关系。

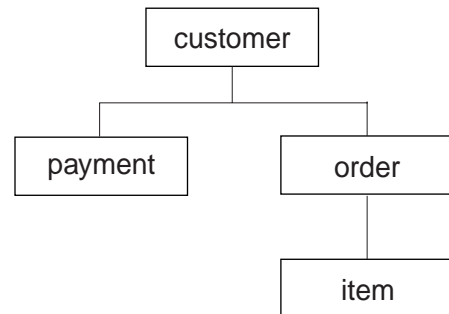


图 43. 样本 XML 文档的树形结构

从此结构中，可以使用 CREATE NICKNAME 语句来将 XML 文档映射到包括四个昵称的关系模式中：

- customers
- orders
- payments
- items

通过使用特殊的主键和外键昵称列选项来将每个昵称指定为父昵称或子昵称来定义昵称之间的关系。每个父昵称必须具有使用主键列选项指定的特殊列。使用引用父昵称的主键列的外键列选项来定义父昵称的子昵称。指定的主键昵称列和外键昵称列并不对应于 XML 文档中的数据，原因是这些昵称列将包含由包装器生成的键。一个昵称可以具有多个子昵称，但是一个昵称只能具有一个父昵称。根昵称没有父昵称。

在样本 XML 文档中，customers 昵称定义了主键，而 orders、payments 和 items 昵称定义了指向父昵称的外键。orders 和 payments 昵称的外键指向 customers 昵称，而 items 昵称的外键指向 orders 昵称。

要标识表示个别元组的 XML 元素，创建一个 XPath 表达式。在此示例中，所有客户元素都是使用 '/doc/customer' XPath 表达式来引用的，而所有订单元素是使用 './order' XPath 表达式来引用的。 './order' XPath 表达式中的句点指示每个订单元素的元组嵌套在相应的客户元素的元组中。

创建一组 XPath 表达式来指定如何从每个元素中抽取列值。在此示例中，客户元素的 id 属性（现在是在昵称中定义的一列）是使用 './@id' XPath 表达式来引用的。客户元素的姓名元素是使用 './name' XPath 表达式来引用的，而客户元素的地址元素是使用 './address/@street' XPath 表达式来引用的。

通过使用 CREATE NICKNAME 语句将 XML 文档映射到一组昵称中之后，利用定义个别元组的 XPath 表达式和文档的每个元素中的各列，通过使用主键和外键来将每个昵称定义为父昵称或子昵称。然后，可以对 XML 文档运行 SQL 查询。

**相关概念:**

- 第 391 页的『昵称与 XML 文档之间的数据关联』

**相关任务:**

- 第 388 页的『将 XML 添加至联合系统』

---

## 将 XML 添加至联合系统

### 将 XML 添加至联合系统

要配置联合服务器以存取 XML 数据源，必须为联合服务器提供有关想要存取的数据源和对象的信息。

可以通过使用 DB2 控制中心或 DB2 命令行来配置联合服务器以存取 XML 数据源。DB2 控制中心包括一个向导，可以使用该向导来指导您完成配置联合服务器所需要执行的步骤。

**先决条件:**

- DB2 Information Integrator 必须安装在将充当联合服务器的服务器上
- 联合数据库必须存在于联合服务器上

**过程:**

要将 XML 数据源添加至联合服务器:

1. 注册 XML 包装器。
2. 注册 XML 服务器定义。
3. 注册 XML 数据源的昵称。
4. 为非根昵称创建联合视图。

根昵称是位于昵称层次结构顶层的昵称。非根昵称是在昵称层次结构中具有父代昵称的昵称。可以具有不是 XML 文档中的顶级元素的根昵称。

可以从 DB2 控制中心或 DB2 命令行处理器中运行语句。将 XML 包装器添加至联合系统之后，就可以对 XML 数据源运行查询了。

**相关概念:**

- 『DB2 Information Integrator 安装过程 - 概述』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）

**相关任务:**

- 第 44 页的『创建联合数据库』
- 第 389 页的『注册 XML 包装器』
- 第 390 页的『为 XML 数据源注册服务器』
- 第 394 页的『为 XML 数据源注册昵称』

- 第 399 页的『为非根昵称 (XML 包装器) 创建联合视图』

相关参考:

- 『DB2 Information Integrator 支持的操作系统 (32 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 支持的操作系统 (64 位)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

## 注册 XML 包装器

注册 XML 包装器是将 XML 数据源添加至联合服务器这一大型任务的一部分。

必须注册包装器才能存取 HMMER 数据源。联合服务器使用包装器来与数据源通信以及从数据源中检索数据。包装器是作为一组库文件实现的。

过程:

要注册包装器, 使用包装器名称和包装器库文件的名称来发出 CREATE WRAPPER 语句。

例如, 要在使用 AIX 操作系统的联合服务器上注册一个名称为 xml\_wrapper 的包装器, 请发出以下语句:

```
CREATE WRAPPER xml_wrapper LIBRARY 'libdb2lxml.a';
```

您指定的包装器库文件的名称取决于联合服务器的操作系统。要了解在 CREATE WRAPPER 语句中应指定的正确名称, 请参阅 XML 包装器库文件的列表。

在这一系列任务中的下一个任务是为 XML 包装器注册服务器定义。

相关参考:

- 第 389 页的『XML 包装器库文件』
- 『CREATE WRAPPER statement』 (SQL Reference, Volume 2)

## XML 包装器库文件

下表列示了 XML 包装器的目录路径和库文件名。

当安装 DB2 Information Integrator 时, 将把三个库文件添加到该表中列示的目录路径中。例如, 如果联合服务器正在 AIX 上运行, 则添加到目录路径中的包装器库文件为 libdb2lxml.a、libdb2lxmlf.a 和 libdb2lxmlu.a。

当注册包装器时, 只指定该表中列示的库文件名。

表 91. XML 包装器库位置和文件名

操作系统	目录路径	包装器库文件
AIX	/usr/opt/db2_08_01/lib/	libdb2lxml.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2lxml.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2lxml.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lxml.so
Windows	%DB2PATH%\bin	db2lxml.dll

|  
| %DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows  
| 上的目录路径。缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。

| 相关任务:

- 第 389 页的『注册 XML 包装器』

## 为 XML 数据源注册服务器

为 XML 数据源注册服务器是将 XML 添加至联合系统这一大型任务的一部分。在注册包装器之后，必须注册相应的服务器。

限制:

XML 包装器不使用 TYPE 和 VERSION 关键字。如果在 CREATE SERVER 语句中使用了这些关键字，则会产生错误。

XML 包装器不支持与联合系统进行传递会话。

过程:

要向联合系统注册 XML 服务器，发出 CREATE SERVER 语句。

例如:

```
CREATE SERVER xml_server WRAPPER my_xml;
```

在这一系列任务中的下一个任务是为 XML 数据源注册昵称。

## 通过代理服务器存取 XML 文档

要通过代理服务器存取 XML 文档，必须在创建服务器定义时指定一些选项。您指定的选项取决于想要访问的代理服务器类型。

如果下列所有条件都满足，则必须在 CREATE SERVER 语句中使用代理选项:

- 您想使用 URI 来检索数据
- 使用的 URI 将通过代理从防火墙后面检索数据
- 使用的防火墙或代理是 HTTP、SOCKS4 或 SOCKS5

与网络管理员核实有关您使用的代理类型的信息以及应该在代理选项中指定的设置。

为 HTTP 代理服务器注册服务器定义的示例:

要注册服务器定义和指定 HTTP 代理服务器，使用以下语句:

```
CREATE SERVER xml_server_h  
  WRAPPER xml_wrapper  
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_h',  
          PROXY_SERVER_PORT '8080');
```

为 SOCKS4 代理服务器注册服务器定义的示例:

要注册服务器定义和指定 SOCKS4 代理服务器，使用以下语句:

```

CREATE SERVER xml_server_s4
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS4', PROXY_SERVER_NAME 'proxy_4',
          PROXY_SERVER_PORT '1080');

```

为 **SOCKS5** 代理服务器注册服务器定义而不提供认证信息的示例:

要注册服务器定义和指定 **SOCKS5** 代理服务器而不提供认证信息, 使用以下语句:

```

CREATE SERVER xml_server_s5
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081');

```

为 **SOCKS5** 代理服务器注册服务器定义并提供认证信息的示例:

要注册服务器定义和指定 **SOCKS5** 代理服务器并提供认证信息, 使用以下语句:

```

CREATE SERVER xml_server_s5a
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081', PROXY_AUTHID 'Martin',
          PROXY_PASSWORD 'aaa', );

```

当将 XML 验证功能部件与代理功能部件配合使用时, 可能对 XML 验证功能部件具有一些限制。在下列情况下您将明白此限制:

- 您正在使用代理功能部件, 在服务器级别, 您已经设置了各种代理选项。
- XML 实例文档包含对位于防火墙外部的 XML 模式的引用

如果存在这些情况的其中一种情况, 则应尝试将 XML 模式的位置更改为防火墙内的一个位置。如果更改了 XML 模式位置, 则必须使用 XML 模式的新位置来更新 XML 实例文档。

相关任务:

- 第 394 页的『为 XML 数据源注册昵称』

相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)

## 为 XML 数据源注册昵称

### 昵称与 XML 文档之间的数据关联

昵称对应于 XML 文档数据的树形结构。父昵称和子昵称对应于数据树形结构的根结构和嵌套元素。这些父昵称和子昵称是通过使用 CREATE NICKNAME 语句指定的主键和外键联系起来的。

每个昵称是由执行下列功能的 XPath 表达式定义的:

- 标识表示个别元组的 XML 元素
- 指定如何从每个元素中抽取列值

XML 包装器使用 XPath 表达式来建立 XML 文档中的数据与关系表中的各行之间的对应关系。这些 XPath 表达式标识 XML 文档中的值, 并确定这些值如何与每一行的各列相对应。XML 包装器只读取 XML 文档数据。XML 包装器不会更新此数据。

当创建昵称时，选择用来指定该昵称与 XML 文档之间的关联的选项。昵称以一种固定方式或者是通过您指定的源名与 XML 文档相关联。

通过固定关联，昵称表示特定 XML 文档中的数据。这些 XML 文档包括：

#### 一个本地文件

指定一个 XML 文件作为 XML 文档。

#### 目录路径中的多个本地文件

指定驻留了多个 XML 文件的目录路径。此目录路径中的 XML 文件为昵称提供 XML 文档数据。所有 XML 文件都必须具有相同的配置。如果目录中任何 XML 文件的配置与昵称的配置不相同，则当 XML 包装器处理 XML 数据文件时将返回空值。该目录必须相对于联合服务器来说是本地的，或者可以从共享文件系统中访问该目录。

**注：**当扫描目录时，XML 包装器将只保留扩展名为 .xml 的那些文件并对它们进行语法分析。XML 包装器将忽略其它所有文件，包括扩展名为 .txt 的文件、扩展名为 .xsd 的文件以及不带扩展名的文件。

使用 CREATE NICKNAME 语句的 FILE\_PATH 选项来指定文件中的固定数据。使用 DIRECTORY\_PATH 选项来指定目录中的固定数据。

如果在运行查询时指定了源数据，则可以使用昵称来表示其模式与昵称定义相匹配的任何 XML 文档源中的数据。这些 XML 文档包括：

#### 统一引用标识

URI 所指示的远程 XML 文件为昵称提供 XML 文档数据。（使用 DOCUMENT 'URI' 昵称列选项来指定此文档源）。

**关系列** 关系表、视图或昵称的列被用作 XML 文档的输入。（使用 DOCUMENT 'COLUMN' 昵称列选项来指定此文档源）。

**文件** 在运行查询时，提供包含 XML 数据的单个文件作为输入。（使用 DOCUMENT 'FILE' 昵称列选项来指定此文档源）。

**目录** 运行查询时，位于指定目录路径下的多个 XML 文件将提供数据。（使用 DOCUMENT 'DIRECTORY' 昵称列选项来指定此文档源）。

指定 DOCUMENT 列选项来指示源数据是在查询时提供的。使用 DOCUMENT 列来指定 URI、COLUMN、FILE 或 DIRECTORY，以指示 XML 文档源的类型。

不能用 DOCUMENT 列选项指定 FILE\_PATH 选项或 DIRECTORY\_PATH 选项。

无论您正在使用哪种类型的数据（采用固定格式的数据，或者是来自在查询时指定的源名中的数据），都可以指定 STREAMING 选项，以便于 XML 包装器将 XML 文档数据分成很多段。XML 包装器处理获得的 XML 数据流，并抽取由查询代码段请求的信息。XML 包装器一次只对一个代码段进行语法分析。由于一次只对一个代码段进行语法分析，因此，使用的内存总量会降低，但是，运行整个查询所需要的处理时间将增加，这取决于服务器的内存容量。因此，只使用 STREAMING 选项来对大型的 XML 文档（50 兆字节或更大的文档）进行语法分析。

还可以选择这样一些昵称选项值：它们可以帮助您优化将检索大量 XML 数据或者包含多个嵌套元素的数据的查询。这些选项包括：

- INSTANCE\_PARSE\_TIME



- XPATH\_EVAL\_TIME
- NEXT\_TIME

可以设置这些选项的值来测试和优化 XML 查询。这些选项值可以控制查找元素和对 XML 文档各行中的数据进行语法分析所需要的处理时间。

#### 相关概念:

- 第 385 页的『什么是 XML?』
- 第 393 页的『XML 包装器的成本模型设施』
- 第 399 页的『XML 成本模型设施的优化技巧』

#### 相关任务:

- 第 394 页的『为 XML 数据源注册昵称』

#### 相关参考:

- 第 515 页的『CREATE NICKNAME 语句语法 - XML 包装器』
- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## XML 包装器的成本模型设施

XML 包装器提供成本模型设施来优化对与 XML 源文档相对应的昵称的查询。

使用 CREATE NICKNAME 语句创建昵称时，可指定以下参数作为昵称选项值以支持成本模型设施。

- INSTANCE\_PARSE\_TIME
- XPATH\_EVAL\_TIME

可对这些参数使用缺省值。或对这些参数设置值以优化对所创建的根昵称或非根昵称的查询。

INSTANCE\_PARSE\_TIME 参数表示读取和解析一个根昵称（例如，customers）的生成行根元素所需的总时间（以毫秒计），包括所有被包含的生成行的非根元素（例如，与每个客户的订单、付款和商品相对应的所有元素）。XML 包装器将在内存中构建一种结构来表示这些生成行的根元素和非根元素。

XPATH\_EVAL\_TIME 参数表示对用来查找与昵称的一行相对应的数据的 XPath 表达式进行求值所需要的时间量（以毫秒计）。对其进行求值的 XPath 表达式包括：用来查找实际行的 XPath 表达式，以及用来查找这些行中的列值的 XPath 表达式。

#### 相关概念:

- 第 385 页的『什么是 XML?』
- 第 391 页的『昵称与 XML 文档之间的数据关联』
- 第 399 页的『XML 成本模型设施的优化技巧』

#### 相关参考:

- 第 515 页的『CREATE NICKNAME 语句语法 - XML 包装器』
- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## 为 XML 数据源注册昵称

为 XML 数据源注册昵称是将 XML 添加至联合系统这一大型任务的一部分。必须创建与 XML 数据源的树形结构相对应的昵称。父昵称对应于树的根结构。子昵称对应于嵌套在父昵称的元素中的那些元素。

### 先决条件:

数据库代码页必须与 XML 源文件的字符集相匹配。

### 限制:

不支持名称空间。

### 过程:

要为 XML 数据源注册昵称，发出 CREATE NICKNAME 语句。

在这一系列任务中的下一个任务是为非根昵称（XML 包装器）创建联合视图。

### 相关概念:

- 第 391 页的『昵称与 XML 文档之间的数据关联』
- 第 393 页的『XML 包装器的成本模型设施』

### 相关任务:

- 第 388 页的『将 XML 添加至联合系统』
- 第 399 页的『为非根昵称（XML 包装器）创建联合视图』
- 第 58 页的『为非关系数据源指定昵称列』

### 相关参考:

- 第 515 页的『CREATE NICKNAME 语句语法 - XML 包装器』
- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## CREATE NICKNAME 语句 - XML 包装器的示例

此主题提供了几个示例来说明如何使用 CREATE NICKNAME 语句来为 XML 包装器注册昵称。此主题包括一个完整的示例（该示例说明如何创建父昵称和子昵称）、特定列选项的示例以及一些说明如何使用视图的示例。

**建议:** 在查询中指定 XPATH 列和昵称选项时，不要使用 self 或 descendant 运算符 //。self 或 descendant 运算符是一个 XPath 运算符，使用它会降低联合服务器的性能。

### 完整示例:

以下示例说明了如何使用第 395 页的图 44 中显示的样本 XML 文件来为 XML 数据源创建昵称。

```

<?xml version="1.0" encoding="UTF-8"?>
  <doc>
    <customer id='123'>
      <name>...</name>
      <address>...</address>
      ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>

```

图 44. 样本 XML 文件

### 父昵称:

第一步是创建父昵称 customers。要创建该昵称，发出以下语句:

```

| CREATE NICKNAME customers
| (
|   id          VARCHAR(5)  OPTIONS(XPATH './@id')
|   name        VARCHAR(16) OPTIONS(XPATH './name'),
|   address     VARCHAR(30) OPTIONS(XPATH './address/@street'),
|   cid         VARCHAR(16) FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
| FOR SERVER xml_server
| OPTIONS(DIRECTORY_PATH '/home/db2user',
|         XPATH '/doc/customer', STREAMING 'YES');

```

此语句对位于指定目录路径 /home/db2user 中的多个 XML 文件创建 customers 昵称。STREAMING 昵称选项指示 XML 源数据是按节点来进行分隔和处理的（在此示例中，按客户记录来进行分隔和处理）。当使用 STREAMING 昵称选项时，包装器不会将整个 XML 文档存储到内存中。相反，XML 包装器将该文档分成多个部分，然后按顺序对各部分进行语法分析。只应对大型的 XML 文档使用 STREAMING 昵称选项。使用此选项时查询性能将受到影响。

### 子昵称:

下一步是为 orders、payments 和 items 元素创建子昵称。

发出以下语句来创建 orders 子昵称。

```

| CREATE NICKNAME orders
| (
|   amount     INTEGER      OPTIONS(XPATH './amount'),
|   date       VARCHAR(10)  OPTIONS(XPATH './date'),

```

```

|         oid      VARCHAR(16)  OPTIONS(PRIMARY_KEY 'YES'),
|         cid      VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
|         FOR SERVER xml_server
|         OPTIONS( XPATH './order');

```

发出以下语句来创建 payments 子昵称。

```

CREATE NICKNAME payments
(
  number  INTEGER      OPTIONS(XPATH './number'),
  date    VARCHAR(10)  OPTIONS(XPATH './date'),
  cid     VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './payment');

```

发出以下语句来创建 items 子昵称。

```

CREATE NICKNAME items
(
  name      VARCHAR(20)  OPTIONS(XPATH './name'),
  quantity  INTEGER      OPTIONS(XPATH './@quant'),
  oid       VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'ORDERS'))
FOR SERVER xml_server
OPTIONS( XPATH './item');

```

### 列选项示例:

下列示例显示当您创建昵称时如何包括 DOCUMENT 列选项。这些示例还显示如何在查询中使用那些选项。

以下 CREATE NICKNAME 示例说明了如何使用 DOCUMENT 'FILE' 列选项:

```

CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');

```

然后, 可以对昵称 customers 运行以下查询, 并在 WHERE 子句中指定 XML 文档的位置:

```

SELECT * FROM customers WHERE doc = '/home/db2user/Customers.xml';

```

以下 CREATE NICKNAME 示例说明了如何使用 DOCUMENT 'DIRECTORY' 列选项:

```

CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'DIRECTORY'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');

```

然后, 可以对昵称 customers 运行以下查询:

```

SELECT name FROM customers WHERE doc = '/home/data/xml';

```

此查询将检索位于目录路径 /home/data/xml 中的 XML 文档, 此目录路径是在 WHERE 子句中指定的。

以下 CREATE NICKNAME 示例说明了如何使用 DOCUMENT 'URI' 昵称列选项:

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'URI'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

然后, 可以对昵称 customers 运行以下查询来从远程位置检索 XML 数据:

```
SELECT * FROM customers WHERE doc = 'http://www.lg-mv.org/foo.xml';
```

以下 CREATE NICKNAME 示例说明了如何使用 DOCUMENT 'COLUMN' 昵称列选项:

```
CREATE NICKNAME emp
(
  doc      VARCHAR(500)  OPTIONS(DOCUMENT 'COLUMN')
  fname   VARCHAR(16)   OPTIONS(XPATH '@first'),
  lname   VARCHAR(16)   OPTIONS(XPATH '@last'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/name');
```

然后, 可以对昵称 emp 运行下列查询之一来检索 XML 数据:

```
SELECT * FROM emp WHERE doc = '<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <title> employees </title>
  <name first="David" last="Marston"/>
  <name first="Donald" last="Leslie"/>
  <name first="Emily" last="Farmer"/>
  <name first="Myriam" last="Midy"/>
  <name first="Lee" last="Tran"/>
  <name first="Lili" last="Farmer"/>
  <name first="Sanjay" last="Kumar"/>
</doc>';
```

或者

```
SELECT * FROM emp WHERE doc = (SELECT * FROM xml_tab);
```

xml\_tab 表中包含填充了 XML 数据的一列。

**视图示例:**

下列示例显示如何为非根昵称创建视图来描述 XML 源文档。在这些示例中, 假定先前已经创建了第 398 页的图 45 中显示的样本文件的昵称, 即 customers、orders、payments 和 items。

---

```

<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <customer id='123'>
    <name>...</name>
    <address>...</address>
    ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>

```

---

图 45. 样本 XML 文件

以下示例说明了如何为非根昵称 order 创建视图:

```

CREATE VIEW order_view AS
  SELECT o.amount, o.date, o.oid, c.cid
  FROM customers c, orders o
  WHERE c.cid = o.cid;

```

以下示例说明了如何为非根昵称 payment 创建视图:

```

CREATE VIEW payment_view AS
  SELECT p.number, p.date, c.cid
  FROM customers c, payments p
  WHERE c.cid = p.cid;

```

以下示例说明了如何为非根昵称 item 创建视图:

```

CREATE VIEW item_view AS
  SELECT i.quantity, i.name, o.oid
  FROM customers c, orders o, items i
  WHERE c.cid = o.cid AND o.oid = i.oid;

```

因为指向根目录的连接路径是存在的, 所以将正确处理提交给这些视图的查询。

例如, 以下查询将把各个客户从同一日期以来的订单总量和付款总额进行配对:

```

SELECT o.amount, p.amount
FROM order_view o, payment_view p
WHERE p.date = o.date AND
      p.cid = o.cid;

```

**相关任务:**

- 第 394 页的『为 XML 数据源注册昵称』

**相关参考:**

- 第 551 页的附录 G, 『联合系统的昵称列选项』
- 第 543 页的附录 F, 『联合系统的昵称选项』
- 第 515 页的『CREATE NICKNAME 语句语法 - XML 包装器』

## 为非根昵称 (XML 包装器) 创建联合视图

为非根昵称 (XML 包装器) 创建联合视图是将 XML 添加至联合系统这一大型任务的一部分。

可以对描述 XML 文档的昵称的层次结构定义联合视图。通过定义联合视图可以确保将 XML 昵称层次结构的各个部分 (不包括根昵称以及用来连接除了特殊 PRIMARY\_KEY 和 FOREIGN\_KEY 列之外的那些列的查询) 连接起来的查询正确运行。

过程:

要定义包括所有必需谓词以及至根目录的全路径的联合视图, 遵循下列步骤:

1. 将每个非根昵称的视图定义为在至根目录的路径上的所有昵称的连接。
2. 在 WHERE 子句中, 使连接谓词基于 PRIMARY\_KEY 和 FOREIGN\_KEY 列。
3. 在 SELECT 列表中, 包括非根昵称的所有列 (但使用 FOREIGN\_KEY 昵称列选项指定的列除外)。
4. 在 SELECT 列表中, 包括使用 PRIMARY\_KEY 选项指定的父昵称的列。

相关参考:

- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## XML 成本模型设施的优化技巧

XML 包装器的成本模型设施可以帮助您优化对所创建的昵称的查询。

成本模型设施使用 CREATE NICKNAME 语句的下列参数:

- INSTANCE\_PARSE\_TIME
- XPATH\_EVAL\_TIME

当发出 CREATE NICKNAME 语句来为 XML 数据源注册昵称时, 就可以指定这些参数的值。

当确定对 XML 源文档每一行中的数据进行语法分析和对昵称的 XPath 表达式进行求值所需要的时间量时, 成本模型设施就会使用这些参数值。

可以使用这些参数的缺省值。但是, 如果您想为所创建的昵称优化对大型的或复杂的 XML 源结构的查询, 可以使用以下示例作为指南。

优化大型查询的示例:

假定 XML 文档与下面四个昵称具有关系模式:

- customers
- orders
- payments
- items

另外，假定昵称 `customers` 是根昵称。

对每个昵称运行查询。对于在您的环境中属于典型情况的样本 XML 数据运行每个查询。

例如：

```
SELECT * from customers;
SELECT * from orders;
SELECT * from payments;
SELECT * from items;
```

记下使用 **db2batch** 命令或等价的命令或实用程序来运行每个查询所需要的时间量（以毫秒计）。（可以使用 **db2batch** 命令来获取一个包含运行查询所需时间的输出文件）。另外，记下所返回的元组数。

对于每个昵称，使用下列公式来确定 `INSTANCE_PARSE_TIME` 和 `XPATH_EVAL_TIME` 参数的最佳值：

$$\text{INSTANCE\_PARSE\_TIME} = (75\% \times \text{SELECT * 查询的运行时间}) \div \text{返回的元组数}$$
$$\text{XPATH\_EVAL\_TIME} = (25\% \times \text{SELECT * 查询的运行时间}) \div \text{返回的元组数}$$

对于根昵称（在本示例中就是 `customers`），使用 `INSTANCE_PARSE_TIME` 和 `XPATH_EVAL_TIME` 参数的计算值。

对于非根昵称（在本示例中就是 `orders`、`payments` 和 `items`），只使用 `XPATH_EVAL_TIME` 参数的计算值。`INSTANCE_PARSE_TIME` 参数值不适用于非根昵称。

可以使用这些公式来指导您调整查询。这些参数的最佳值还取决于 XML 源文档的复杂程度和所使用的处理器的速度。

相关概念：

- 第 385 页的『什么是 XML?』
- 第 391 页的『昵称与 XML 文档之间的数据关联』
- 第 393 页的『XML 包装器的成本模型设施』

相关参考：

- 『db2batch - Benchmark Tool Command』（*Command Reference*）

---

## XML 数据源 – 示例查询

此主题提供了几个使用昵称 `customers`、`orders` 和 `items` 的样本查询。先前已经使用 `CREATE NICKNAME` 语句注册了这些昵称。

以下查询将显示所有客户名称：

```
SELECT name FROM customers;
```

以下查询将显示客户名为 `Chang` 的所有记录：

```
SELECT * FROM customers
WHERE name='Chang';
```

以下查询将显示每个客户的客户名和每个订单的数量：



```
SELECT c.name, o.amount FROM customers c, orders o
WHERE c.cid=o.cid;
```

必须指定连接 `c.cid=o.cid`，以便指示昵称 `customers` 与昵称 `orders` 之间的父子关系。

以下查询将选择客户地址、订单数量、每个订单的产品名称以及每个客户的产品：

```
SELECT c.address, o.amount, i.name FROM customers c, orders o, items i
WHERE c.cid=o.cid AND o.oid=i.oid;
```

必须指定两个连接才能维护父子关系。

下列示例说明了如何使用指定 `DOCUMENT` 列选项而不是 `FILE_PATH` 昵称选项的昵称来编写查询。以下示例显示了用来创建昵称 `customers` 的相应 `CREATE NICKNAME` 语句：

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   OPTIONS(PRIMARY_KEY 'YES')
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

以下查询将从文件路径 `/home/db2user/Customers.xml` 中的 XML 文件 `Customers.xml` 中选择所有数据：

```
SELECT * FROM customers
WHERE doc='/home/db2user/Customers.xml';
```

以下查询将从 `Customers.xml` 文件中选择订单数量超过 1000 的每个订单的客户名称以及订单的日期：

```
SELECT c.name, o.date FROM customers c, orders o
WHERE c.doc='/home/db2user/Customers.xml' AND o.amount > 1000;
```

文件路径 `/home/db2user/Customers.xml` 指定 `Customers.xml` 文件的位置。

#### 相关参考：

- 第 515 页的『`CREATE NICKNAME` 语句语法 - XML 包装器』
- 第 394 页的『`CREATE NICKNAME` 语句 - XML 包装器的示例』

---

## XML 包装器的消息

本主题描述了您在使用 XML 的包装器时可能会遇到的消息。有关这些消息的更多信息，请参阅《DB2 消息参考》。

表 92. 由 XML 的包装器发出的消息

错误代码	消息	解释
SQL0405N	数字文字 “<column_name>” 无效，原因是它的值超出了范围。	指定的数字文字不在可接受的范围内。检查 <code>CREATE NICKNAME</code> 语句中该列的数据类型。
SQL0408N	值与它的赋值目标的数据类型不兼容。目标名称为 “<column_name>”。	为该列指定的值的数据类型与赋值目标的已声明数据类型不兼容。检查 <code>CREATE NICKNAME</code> 语句中该列的数据类型。

表 92. 由 XML 的包装器发出的消息 (续)

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “创建包装器对象时出错。”)	创建新的包装器对象时出错。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “<xerces_xalan_error_message>。”)	调用 Xerces 或 Xalan 函数期间出错。检查 XML 文档。如果文档的结构良好, 则参阅 Xalan 文档以获取有关错误消息的更多信息。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “XalanDOMException: 异常代码为 <exception_code>。”)	发生了 XalanDOMException 异常。参阅 Xalan 文档以获取有关异常代码的更多信息。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “XMLException: <exception_error_message>。”)	发生了 XMLException 异常。参阅 Xalan 文档以获取有关异常代码的更多信息。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “XSLEException: <exception_error_message>。”)	发生了 XSLEException 异常。参阅 Xalan 文档以获取有关异常代码的更多信息。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “SAXParseException: <exception_error_message>。”)	发生了 SAXParseException 异常。参阅 Xalan 文档以获取有关异常代码的更多信息。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “获取节点值时出错。”)	Xalan 尝试了访问无效节点。请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “对 XML 文档进行解析时出错。”)	对 XML 文档进行解析时出错。检查 XML 文档。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “获取 XML 文档的根元素时出错。”)	在对 XML 文档进行分析之后, Xalan 试图检索根元素但是失败了。检查 XML 文档。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “在对 XPath 表达式求值时产生了未指定的异常。”)	在对 XPath 表达式进行求值时 Xalan 生成了未指定的异常。检查 XML 文档, 并参阅 Xalan 文档。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “获取节点值时产生了未指定的异常。”)	检索节点值时 Xalan 生成了未指定的异常。检查 XML 文档, 并参阅 Xalan 文档。

表 92. 由 XML 的包装器发出的消息 (续)

错误代码	消息	解释
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “对输入文档进行语法分析时产生了未指定的异常。”)	在对 XML 文档进行语法分析时 Xalan 生成了未指定的异常。检查 XML 文档, 并参阅 Xalan 文档。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “对基数进行求值时出错。”)	请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “<SOAP_error_message>。”)	SOAP 库产生了错误。如果您不能解决 SQL 语句错误, 请与 IBM 软件支持机构联系。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “URI 无效。”)	包装器不能存取指定的 URL。验证是否可以访问 URL。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “XML 文档内容无效。”)	XML 文档的内容无效。验证文档的结构是否良好。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “SOAP 包络无效。”)	SOAP 包络无效。检查它的语法和内容。
SQL0901N	SQL 语句因不太严重的系统错误而失败。可以处理后续 SQL 语句。(原因: “内存分配错误。”)	分配内存时发生了错误。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“DATE 格式不正确”。	XML 文档中的日期值的格式不正确。日期值的有效格式为 yyyy-mm-dd。检查 XML 文档。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“列数据类型不受支持”。	昵称列具有不受支持的数据类型。检查 CREATE NICKNAME 语句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“TYPE 子句不受支持”。	CREATE SERVER 语句中包含 TYPE 子句。XML 包装器不支持此子句。除去该子句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“VERSION 子句不受支持”。	CREATE SERVER 语句中包含 VERSION 子句。XML 包装器不支持此子句。除去该子句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“对 DOCUMENT 列无效使用了谓词”。	查询中包含具有不正确的操作数的谓词。查查查询中的谓词。

表 92. 由 XML 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“对 FOREIGN_KEY 列无效使用了谓词”。	查询中包含具有不正确的操作数的谓词。检查查询中的谓词。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“对 PRIMARY_KEY 列无效使用了谓词”。	查询中包含具有不正确的操作数的谓词。检查查询中的谓词。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“XPath 和 DOCUMENT 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“XPath 和 FOREIGN_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“XPath 和 PRIMARY_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“DOCUMENT 和 FOREIGN_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“DOCUMENT 和 PRIMARY_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“FOREIGN_KEY 和 PRIMARY_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“缺少列选项”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“DOCUMENT 列选项不是唯一的”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。

表 92. 由 XML 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“FOREIGN_KEY 列选项不是唯一的”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“PRIMARY_KEY 列选项不是唯一的”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“DOCUMENT 选项值无效”。	在 CREATE NICKNAME 语句中指定的 DOCUMENT 选项的值无效。该值必须为 FILE。检查 CREATE NICKNAME 语句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“PRIMARY_KEY 选项值无效”。	在 CREATE NICKNAME 语句中指定的 PRIMARY_KEY 选项的值无效。该值必须为 YES。检查 CREATE NICKNAME 语句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“FOREIGN_KEY 选项值无效”。	在 CREATE NICKNAME 语句中指定的 FOREIGN_KEY 选项的值无效。该值与任何父昵称都不匹配。检查 CREATE NICKNAME 语句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“FILE_PATH 和 DOCUMENT 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 FILE_PATH 和 DOCUMENT 选项。检查 CREATE NICKNAME 语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“FILE_PATH 和 SOAP 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 FILE_PATH 和 SOAP 选项。检查 CREATE NICKNAME 语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“DIRECTORY_PATH 和 SOAP 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 DIRECTORY_PATH 和 SOAP 选项。检查 CREATE NICKNAME 语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“FILE_PATH 和 DIRECTORY_PATH 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 FILE_PATH 和 DIRECTORY_PATH 选项。检查 CREATE NICKNAME 语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。 相关联的文本和标记为“VALIDATE 和 STREAMING 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 VALIDATE 和 STREAMING 选项。检查 CREATE NICKNAME 语句的语法。

表 92. 由 XML 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“FILE_PATH 和 FOREIGN_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 FILE_PATH 和 FOREIGN_KEY 选项。检查 CREATE NICKNAME 语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“DIRECTORY_PATH 和 FOREIGN_KEY 选项不兼容”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。不能同时指定 DIRECTORY_PATH 和 FOREIGN_KEY 选项。检查 CREATE NICKNAME 语句的语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“在启用了 STREAMING 的情况下 XPATH 选项值无效”。	当启用 STREAMING 功能部件时，昵称 XPATH 表达式无效。检查 XPATH 选项，以找出无效的值，例如，/、./ 和 //。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“未能读取 XML 文件”。	在 CREATE NICKNAME 语句或查询中指定的文件路径无效。指定的文件不存在。检查 CREATE NICKNAME 语句和查询。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“打不开目录”。	在 CREATE NICKNAME 语句或查询中指定的目录路径无效。指定的目录不存在。检查 CREATE NICKNAME 语句和查询。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“缺少对 XML 数据的引用”。	CREATE NICKNAME 语句中必须包含对 XML 数据的引用。检查 CREATE NICKNAME 语句。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“缺少带有值 'SOAP' 的 DOCUMENT 列选项”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。检查 DOCUMENT 选项的值。该值必须为 SOAP。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“缺少 SOAP 选项”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。必须指定 SOAP 选项。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“INSTANCE_PARSE_TIME 仅用于根昵称”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。只能对根昵称指定 INSTANCE_PARSE_TIME 值。检查 CREATE NICKNAME 语法。
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“VALIDATE 选项仅用于根昵称”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。仅当指定的昵称是根昵称时，才能将 VALIDATE 选项设置为 YES。检查 CREATE NICKNAME 语法。

表 92. 由 XML 的包装器发出的消息 (续)

错误代码	消息	解释
SQL1822N	从数据源“XML 包装器”中接收到意外的错误代码“<trace_point>”。相关联的文本和标记为“STEAMING 选项仅用于根昵称”。	CREATE NICKNAME 语句不正确，与指定的语句不相同。仅当指定的昵称是根昵称时，才能将 STREAMING 选项设置为 YES。检查 CREATE NICKNAME 语法。
SQL1823N	对于服务器“<server_name>”中的数据类型“<data_type_name>”不存在数据类型映射。	CREATE NICKNAME 语句不正确，与指定的语句不相同。列数据类型无效。检查 CREATE NICKNAME 语法。
SQL1881N	“<option_name>”不是“<object_name>”的有效“<option_type>”选项。	指定的选项可能不存在，或者可能对于此数据源无效。检查 CREATE NICKNAME 语句。
SQL1881N	“DIRECTORY_PATH”不是“<object_name>”的有效“NICKNAME”选项。	在 CREATE NICKNAME 语句中指定的 DIRECTORY_PATH 选项的值无效。指定的目录必须是根目录。检查 CREATE NICKNAME 语句。
SQL1882N	“昵称”选项“VALIDATE”不能设置为“<object_name>”的“<option_value>”。	在 CREATE NICKNAME 语句中指定的 VALIDATE 选项的值无效。此值必须是 YES 或 NO。检查 CREATE NICKNAME 语句。
SQL1882N	“昵称”选项“STREAMING”不能设置为“<object_name>”的“<option_value>”。	在 CREATE NICKNAME 语句中指定的 STREAMING 选项的值无效。此值必须是 YES 或 NO。检查 CREATE NICKNAME 语句。
SQL1883N	“<option_name>”是“<object_name>”的必需“<option_type>”选项。	未指定必需的 DB2 选项。检查 CREATE NICKNAME 语句。

**相关概念:**

- 『消息介绍』（《消息参考》第 1 卷）

**相关参考:**

- 『SQLSTATE 消息』（《消息参考》第 2 卷）





---

## 第 4 部分 用户定义的函数



---

## 第 25 章 生命科学用户定义的函数

本章说明什么是生命科学用户定义的函数、如何将它们添加至联合系统以及如何在查询中使用它们。

---

### 生命科学用户定义的函数 - 概述

#### 生命科学用户定义的函数 - 概述

生命科学用户定义的函数为您提供通常用来分析数据的算法。

生命科学用户定义的函数使用标准的单字母代码和 IUPAC-IUB 多义代码来表示氨基酸和核苷酸。

生命科学用户定义的函数是与非关系型包装器的“生命科学用户定义的函数”组件一起安装的。在安装生命科学用户定义的函数之后，必须注册这些函数。

为了避免与名称空间发生冲突，所有生命科学用户定义的函数都是采用 DB2LS 模式注册的。

#### 相关概念:

- 『DB2 Information Integrator 非关系型包装器』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『DB2 Information Integrator 关系型包装器』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)

#### 相关任务:

- 第 413 页的『注册生命科学用户定义的函数』
- 第 414 页的『禁用生命科学用户定义的函数』

#### 相关参考:

- 第 411 页的『生命科学用户定义的函数库文件』
- 第 412 页的『按函数类别排列的生命科学用户定义的函数』

### 生命科学用户定义的函数库文件

DB2 Information Integrator 中包括的某些用户定义的函数需要库文件。当在联合服务器上注册用户定义的函数时需要这些库文件。

当为非关系数据源安装 DB2 Information Integrator 时，下列用户定义的库文件便安装在联合服务器上。

生命科学用户定义的函数库:

表 93. 生命科学用户定义的函数库位置和文件名

函数类型	操作系统	目录路径	库文件名
生命科学用户定义的函数	AIX	/SQLLIB/function	libdb2lsudfs.a
生命科学用户定义的函数	HP-UX	/SQLLIB/function	libdb2lsudfs.sl
生命科学用户定义的函数	Linux	/SQLLIB/function	libdb2lsudfs.so
生命科学用户定义的函数	Solaris	/SQLLIB/function	libdb2lsudfs.so
生命科学用户定义的函数	Windows	%DB2PATH%\bin	db2lsudfs.dll

缺省 Windows 目录路径为 C:\Program Files\IBM\SQLLIB。%DB2PATH% 是一个环境变量，用来指定 DB2 Information Integrator 安装在 Windows 上的目录路径。

### LSGeneWise 用户定义的函数库:

LSGeneWise 用户定义的函数需要一个单独的库。

表 94. LSGeneWise 函数库位置和文件名

函数类型	操作系统	目录路径	库文件名
LSGeneWise 函数	UNIX	/SQLLIB/lib	libdb2lsSTgenewise.a

### 相关概念:

- 第 411 页的『生命科学用户定义的函数 - 概述』

## 按函数类别排列的生命科学用户定义的函数

表 95 列示了按函数类别排列的生命科学用户定义的函数。它还提供了每个类别的简要描述。

表 95. 生命科学用户定义的函数

函数类别	用户定义的函数	描述
向后转换	LS Pep 2 A m b N u c LS Pep 2 Prob Nuc	和 将氨基酸序列转换为核苷酸序列。
定义行语法分析	LS Define Parse	对定义行的元素（例如，BLAST 包装器返回的元素或存在于 FASTA 格式数据文件中的元素）进行语法分析。
一般化模式匹配	LS Pattern Match LS Prosite Pattern	和 标识给定字符串中令人感兴趣的区域，如核苷酸或缩氨酸序列。
GeneWise	LS GeneWise	将蛋白质序列与基因的序列比对。
图谱	LS Multi Match、 LS Multi Match 3 和 LS Bar Code	匹配核苷酸或氨基酸序列中的模式。
逆向	LS Rev Nuc、LS Rev Pep 和 LS Rev Comp	和 将核苷酸或氨基酸序列掉转。
转换	LS Nuc 2 Pep 和 LS Trans All Frames	和 将核苷酸序列转换为缩氨酸序列。

### 相关概念:

- 第 411 页的『生命科学用户定义的函数 - 概述』

### 相关任务:

- 第 413 页的『注册生命科学用户定义的函数』
- 第 414 页的『禁用生命科学用户定义的函数』

## 注册生命科学用户定义的函数

在可以使用生命科学用户定义的函数之前，必须注册这些函数。

### 先决条件:

非关系型包装器的生命科学用户定义的函数组件必须随 DB2 Information Integrator 一起安装。

### 过程:

要注册生命科学用户定义的函数，请使用 `enable_LSFunctions` 命令。

- 在运行 Windows NT 的联合服务器上，此命令位于 `sqllib\bin` 目录中
- 在运行 AIX 的联合服务器上，此命令位于 `sqllib/bin` 目录中

`enable_LSFunctions` 命令的语法为:

```
enable_LSFunctions -n dbName -u userID -p password [-force]
```

#### **dbName**

正在其中注册函数的联合数据库的名称。

#### **userID**

联合数据库的有效用户标识。

#### **password**

用户标识的有效密码。

**force** 这是一个可以用来除去函数并重新注册它们的标志。如果函数被意外毁坏或删除了，则使用此标志来重新注册它们。

`enable_LSFunctions` 命令将在联合数据库中注册所有生命科学用户定义的函数。这些函数是使用模式名 `DB2LS` 来注册的。

以下示例显示发出 `enable_LSFunctions` 命令时返回的输出:

```
C:> enable_LSFunctions -n federateddb -u db2admin -p db2admin
```

```
(0) Life Sciences Functions were found
    -- Create Life Sciences Functions ...
    Create Life Sciences Functions Successfully.
```

```
*** Please allow a few seconds to clean up the system .....
```

以下示例显示当使用 `force` 标志并且函数已注册时发出 `enable_LSFunctions` 命令时所返回的输出:

```
C:> enable_LSFunctions -n federateddb -u db2admin -p db2admin -force
```

```
(21) Life Sciences Functions were found
```

```
Life Sciences functions already exist ...
Reinstall Life Sciences functions ...
  -- Drop Life Sciences Functions ...
  Drop Life Sciences Functions Successfully.
  -- Create Life Sciences Functions ...
  Create Life Sciences Functions Successfully.

*** Please allow a few seconds to clean up the system .....
```

#### 相关概念:

- 第 411 页的『生命科学用户定义的函数 - 概述』

#### 相关任务:

- 『向您的 DB2 Information Integrator 系统添加关系型包装器、非关系型包装器和用户定义的函数』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 第 414 页的『禁用生命科学用户定义的函数』

#### 相关参考:

- 第 412 页的『按函数类别排列的生命科学用户定义的函数』

## 禁用生命科学用户定义的函数

如果您不再想使用生命科学用户定义的函数，则可以临时禁用这些函数或者从联合数据库中永久除去它们。

#### 过程:

要禁用生命科学用户定义的函数，请使用 **disable\_LSFunctions** 命令。

- 在运行 Windows NT 的联合服务器上，此命令位于 sqllib\bin 目录中
- 在运行 AIX 的联合服务器上，此命令位于 sqllib/bin 目录中

**disable\_LSFunctions** 命令的语法为:

```
disable_LSFunctions -n dbName -u userID -p password
```

#### **dbName**

想要在其中禁用函数的联合数据库的名称。

#### **userID**

联合数据库的有效用户标识。

#### **password**

用户标识的有效密码。

#### 禁用生命科学用户定义的函数的示例:

以下示例显示发出 **disable\_LSFunctions** 命令时返回的输出:

```
C:>disable_LSFunctions -n federateddb -u db2admin -p db2admin
```

```
(21) Life Sciences Functions were found
  -- Drop Life Sciences Functions ...
  Drop Life Sciences Functions Successfully.
```

```
*** Please allow a few seconds to clean up the system .....
```

必须卸载函数才能从联合数据库中永久除去这些函数。

#### 相关任务:

- 『除去关系型包装器、非关系型包装器和生命科学用户定义的函数 (Windows)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 『除去包装器、用户定义的函数和包装器开发工具箱 (UNIX)』 (《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》)
- 第 413 页的『注册生命科学用户定义的函数』

---

## 向后转换用户定义的函数

### 向后转换用户定义的函数 - 概述

使用向后转换用户定义的函数来将缩氨酸序列转换为核苷酸序列。向后转换就是将转换反转。

因为从氨基酸到核苷酸三基码的映射是一对多，所以向后转换将生成两个结果:

#### 最不确定

单一文本转换和查找。使用 LSPep2AmbNuc 用户定义的函数来进行最不确定转换。

#### 最有可能

需要来自基码频率表的更多信息。使用 LSPep2ProbNuc 用户定义的函数来进行最有可能转换。

#### 相关概念:

- 第 411 页的『生命科学用户定义的函数 - 概述』

#### 相关参考:

- 第 415 页的『LSPep2AmbNuc 用户定义的函数』
- 第 418 页的『LSPep2AmbNuc 用户定义的函数 - 错误消息』
- 第 418 页的『LSPep2ProbNuc 用户定义的函数』
- 第 420 页的『LSPep2ProbNuc 用户定义的函数 - 错误消息』
- 第 416 页的『LSPep2AmbNuc 用户定义的函数 - 示例』
- 第 419 页的『LSPep2ProbNuc 用户定义的函数 - 示例』

## LSPep2AmbNuc 用户定义的函数

```
DB2LS.LSPep2AmbNuc(input peptide sequence [, filepath to external translation table])
```

#### input peptide sequence

描述缩氨酸序列的有效字符串表示法。字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 10890 个字节。输入数据使用标准的氨基酸符号和多义代码。

#### filepath to external translation table

如果使用定制的转换表，则包括文件路径信息以查找转换表。此路径的字符串值一定不能超过 255 个字符。

模式名是 DB2LS。

使用 LSPep2AmbNuc 函数来根据转换表从缩氨酸序列生成最不确定的核苷酸序列。

此函数的结果是具有 VARCHAR 数据类型并且实际长度不超过 32672 个字节的字符串。此结果表示根据转换表（内置转换表或您指定的转换表）生成的最不确定的核苷酸序列。

如果不指定转换表，则此函数在缺省情况下使用表 96。

表 96. 缺省转换表

氨基酸符号	缩写	基码
A	Ala	GCX
B	Asx	RAY
C	Cys	TGY
D	Asp	GAY
E	Glu	GAR
F	Phe	TTY
G	Gly	GGX
H	His	CAY
I	Ile	ATH
K	Lys	AAR
L	Leu	YTX
M	Met	ATG
N	Asn	AAY
P	Pro	CCX
Q	Gln	CAR
R	Arg	MGX
S	Ser	WSX
T	Thr	ACX
V	Val	GTX
W	Trp	TGG
X	Xxx	XXX
Y	Tyr	TAY
Z	Glx	SAR
*	End	TRR

**相关参考:**

- 第 418 页的『LSPep2AmbNuc 用户定义的函数 - 错误消息』
- 第 418 页的『LSPep2ProbNuc 用户定义的函数』
- 第 416 页的『LSPep2AmbNuc 用户定义的函数 - 示例』

## LSPep2AmbNuc 用户定义的函数 - 示例

可以使用 values 语句来调用此函数。唯一的输入是缩氨酸序列，如以下示例所示:



```
values db21s.LSPep2AmbNuc('HR');
```

以上示例使用不确定转换和内置转换表来将缩氨酸转换为核苷酸。以上语句的结果是从标准氨基酸符号创建的核苷酸序列:

```
CAYMGX
```

以下示例使用不确定转换和内置表来将缩氨酸转换为核苷酸:

```
values db21s.LSPep2AmbNuc('SRGFGFITYSHSSMIDEAQKSRPHKIDGRVVEPKRA');
```

这个 `values` 语句的结果是以下核苷酸序列。(为了能够印刷在页面上, 已对此序列作了折行处理。)

```
WSXMGXGGXTTYGGXTTYATHACXTAYWSXCAYWSXWSXATGATHGAYGARGCXCARA  
ARWSXMGXCCXCAYAARATHGAYGGXMGXGTGTXGARCCXAARMGXGCX
```

下一个示例显示将此函数应用于一组从表或昵称中抽取的值:

```
SELECT DB2LS.LsPep2AmbNuc(peptide_seq) FROM table protein_table;
```

`protein_table` 表的 `peptide_seq` 列中的数据看起来是这样的:

表 97. `peptide_seq` 列中的数据

peptide_seq
GIKEDTEEHHLRDYFE
QKYHTVNGHNCEVRKA
.....

该 `Select` 语句的结果是:

```
GGXATHAARGARGAYACXGARGARCAYCAYYTXMGXGAYTAYTTYGAR  
CARAARTAYCAYACXGTAAAYGGXCAYAAYTYGARGTXMGXAARGCX  
...
```

以下示例使用不确定转换和用户定义的表来将缩氨酸转换为核苷酸。通常, 转换表之间的差别很小。可能只有一两个符号是独特的。这可能是由于一些物种具有较多的基码或一些物种具有较少的基码而导致的。例如, 基码 `AGG` 在果蝇中是不存在的。

```
values db21s.LSPep2AmbNuc('RGNMGGGNYGNQNGGGNWNNG',  
                          '\data\transl_table_06.txt')
```

假设输入转换表是用于果蝇的, 则 `values` 语句的结果如以下示例所示:

```
MGRGGXAAYATGGGXXGGXAAAYTAYGGXAAYTARAAYGGXGGXAAAYTGAAYAAAYGGX
```

#### 相关参考:

- 第 415 页的『`LSPep2AmbNuc` 用户定义的函数』
- 第 441 页的『`LSNuc2Pep` 用户定义的函数 - 示例』

## LSPep2AmbNuc 用户定义的函数 - 错误消息

表 98. 由 LSPep2AmbNuc 用户定义的函数发出的消息

错误代码	消息	解释
SQL0443N	例程 “DB2LS.LSPEP2AMBNUC” (特定名称 “LSPEP2AMBNUC”) 已经返回带有诊断文本 “序列无效” 的错误 SQLSTATE。SQLSTATE=38608	所给序列无效。
SQL0443N	例程 “DB2LS.LSPEP2AMBNUC” (特定名称 “LSPEP2AMBNUCUT”) 已经返回带有诊断文本 “找不到转换” 的错误 SQLSTATE。SQLSTATE=38610	转换表文件是空的。
SQL0443N	例程 “LSPEP2AMBNUC” (特定名称 “LSPEP2AMBNUCUT”) 已经返回带有诊断文本 “无法打开转换表文件” 的错误 SQLSTATE。SQLSTATE=38612	指定的转换表文件不存在。
SQL0443N	例程 “DB2LS.LSPEP2AMBNUC” (特定名称 “LSPEP2AMBNUCUT”) 已经返回带有诊断文本 “从文件读取的行太长” 的错误 SQLSTATE。SQLSTATE=38614	文件包含超过所允许的长度 的行。
SQL0443N	例程 “DB2LS.LSPEP2AMBNUC” (特定名称 “LSPEP2AMBNUCUT”) 已经返回带有诊断文本 “数据文件无效” 的错误 SQLSTATE。SQLSTATE=38615	文件格式无效。
SQL0443N	例程 “LSPEP2AMBNUC” (特定名称 “LSPEP2AMBNUCUT”) 已经返回带有诊断文本 “无法构造转换表” 的错误 SQLSTATE。SQLSTATE=38611	在文件中找到无效的符号。

### 相关参考:

- 第 415 页的『LSPep2AmbNuc 用户定义的函数』

## LSPep2ProbNuc 用户定义的函数

→DB2LS.LSPep2ProbNuc(*input peptide sequence*, *filepath to codon frequency table*)→

### input peptide sequence

描述缩氨酸序列的有效字符串表示法。此字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 10890 个字节。输入数据使用标准的氨基酸符号。

### filepath to codon frequency table

这是基码频率表。包含用于查找频率表的文件路径信息。此路径的字符串值一定不能超过 255 个字符。

模式名是 DB2LS。

使用 LSPep2ProbNuc 函数来根据第二个自变量中指定的基码频率表从缩氨酸序列生成最有可能的核苷酸序列。

此函数的结果是具有 VARCHAR 数据类型并且实际长度不超过 32672 个字节的字符串，它表示最有可能的使用基码频率表的核苷酸序列。

**相关参考:**

- 第 415 页的『LSPep2AmbNuc 用户定义的函数』
- 第 420 页的『LSPep2ProbNuc 用户定义的函数 - 错误消息』
- 第 419 页的『LSPep2ProbNuc 用户定义的函数 - 示例』

## LSPep2ProbNuc 用户定义的函数 - 示例

以下示例显示可以如何使用 yeast\_high.cod 频率表中定义的最有可能的转换来将缩氨酸序列转换为核苷酸序列。

```
values db21s.LSPep2ProbNuc('RDNNDDN', '\data\yeast_high.cod')
```

以上 values 语句的结果为:

```
AGAGACAATAACGACGATGATAAC
```

第二次执行同一个语句将生成以下字符串:

```
AGAGATAATAACGACGATGACAAC
```

第三次执行同一个语句将生成以下具有随机值的字符串:

```
AGAGATAACAACGACGCGATAAT
```

粗体的基码突出显示当前转换和上一转换的区别。

来自单个 values 语句的结果显示函数 LSPep2ProbNuc 根据先前统计信息选择了其中一个可能的符号。这与 LSPep2AmbNuc 函数不同，后者使用存在更多可能转换的不确定符号。

函数 LSPep2ProbNuc 为每个符号挑选最有可能的转换，然后用先前挑选的集合中的随机转换来替换每个符号。假定频率表包含以下数据:

表 99. 样本频率表数据

氨基酸	基码	频率
Ala	GCG	0.17
Ala	GCA	0.13
Ala	GCT	0.17
Ala	GCC	0.53

假定缩氨酸序列包含 4 个“A”符号 (Ala)。函数将 A 转换为 GCC 两次; 转换为 GCG 一次, 转换为 GCT 一次。但是, 函数生成转换的顺序是随机的。查询可以将第一个 A 转换为 {GCC, GCC, GCG, GCT} 集合中的每个转换。结果始终是这样的: 在输出 DNA 序列中, GCC 出现两次, GCG 出现一次, GCT 出现一次。对同一个序列多次执行此函数可能会返回交换了值的 DNA 序列。

**相关参考:**

- 第 418 页的『LSPep2ProbNuc 用户定义的函数』
- 第 420 页的『LSPep2ProbNuc 用户定义的函数 - 错误消息』
- 第 416 页的『LSPep2AmbNuc 用户定义的函数 - 示例』

## LSPep2ProbNuc 用户定义的函数 - 错误消息

表 100. 由 LSPep2ProbNuc 用户定义的函数发出的消息

错误代码	消息	解释
SQL0443N	例程 “DB2LS.LSPEP2PROBNUC” (特定名称 “LSPEP2PROBNUC”) 已经返回带有诊断文本 “序列无效” 的错误 SQLSTATE。SQLSTATE=38608	输入序列无效。
SQL0443N	例程 “DB2LS.LSPEP2PROBNUC” (特定名称 “LSPEP2PROBNUC”) 已经返回带有诊断文本 “找不到转换” 的错误 SQLSTATE。SQLSTATE=38610	基码频率表文件是空的。
SQL0443N	例程 “LSPEP2PROBNUC” (特定名称 “LSPEP2PROBNUC”) 已经返回带有诊断文本 “无法打开转换表文件” 的错误 SQLSTATE。SQLSTATE=38612	该文件不存在。
SQL0443N	例程 “DB2LS.LSPEP2PROBNUC” (特定名称 “LSPEP2PROBNUC”) 已经返回带有诊断文本 “从文件读取的行太长” 的错误 SQLSTATE。SQLSTATE=38614	文件包含的行超出所允许的长度。
SQL0443N	例程 “DB2LS.LSPEP2PROBNUC” (特定名称 “LSPEP2PROBNUC”) 已经返回带有诊断文本 “数据文件无效” 的错误 SQLSTATE。SQLSTATE=38615	文件格式无效。
SQL0443N	例程 “LSPEP2PROBNUC” (特定名称 “LSPEP2PROBNUC”) 已经返回带有诊断文本 “无法构造转换表” 的错误 SQLSTATE。SQLSTATE=38611	文件包含无效的符号。

### 相关参考:

- 第 418 页的『LSPep2ProbNuc 用户定义的函数』
- 第 419 页的『LSPep2ProbNuc 用户定义的函数 - 示例』

## 定义行语法分析用户定义的函数

### 定义行语法分析用户定义的函数 - 概述

定义行语法分析用户定义的函数对定义行的元素进行语法分析。例如，定义行语法分析用户定义的函数:

- 能够与从定义行语法分析出来的序列标识中的其它数据源进行连接
- 对定义行各部分的谓词 (例如, 'species = "human"') 求值

定义行语法分析函数涵盖了大多数常见的定义行格式。示例包括 BLAST 包装器返回的定义行元素或存在于 FASTA 格式数据文件中的定义行元素。

### 相关概念:

- 第 99 页的『定义行分析』

### 相关参考:

- 第 423 页的『LSDefineParse 用户定义的函数 - 示例』
- 第 421 页的『LSDefineParse 用户定义的函数』

## LSDefineParse 用户定义的函数

每个 LSDefineParse 函数都对 NCBI 标准 FASTA 序列标识 (NSID) 的字段进行语法分析并将描述存放到表列中。作为复合定义的定义行输出到多个行中, 每一行包含一个组件定义。

DB2LS 是与定义行语法分析用户定义的函数配合使用的模式名。

定义行语法分析用户定义的函数为:

```
▶▶—DB2LS.LSDefineParse2—(definition line)————▶▶
▶▶—DB2LS.LSDefineParse3—(definition line)————▶▶
▶▶—DB2LS.LSDefineParse2_2—(definition line)————▶▶
▶▶—DB2LS.LSDefineParse2_3—(definition line)————▶▶
▶▶—DB2LS.LSDefineParse3_3—(definition line)————▶▶
```

### definition line

具有 FASTA 格式的定义行的有效字符串表示法。此字符串必须具有 VARCHAR 数据类型, 并且实际长度不超过 1024 个字节。

### LSDefineParse2 用户定义的函数:

LSDefineParse2 对带有双字段 NSID 的定义行进行语法分析。此函数的结果是包含 4 个列的表:

表 101. LSDefineParse2 用户定义的函数结果表的列描述

列名	描述
ROWID	一个整数, 它对函数返回的行进行编号。
TAG	最多包含 3 个字符的 VARCHAR, 它表示 NSID 标记。
IDENTIFIER	最多包含 20 个字符的 VARCHAR, 它表示 NSID 中的第二个标识字段。
DESCRIPTION	最多包含 1019 个字符的 VARCHAR。

### LSDefineParse3 用户定义的函数:

LSDefineParse3 对带有三字段 NSID 的定义行进行语法分析。此函数的结果是包含 5 个列的表:

表 102. LSDefineParse3 用户定义的函数结果表的列描述

列名	描述
ROWID	一个整数, 它对函数返回的行进行编号。
TAG	最多包含 3 个字符的 VARCHAR, 它表示 NSID 标记。

表 102. *LSDefineParse3* 用户定义的函数结果表的列描述 (续)

列名	描述
ACCESSION	最多包含 20 个字符的 VARCHAR, 它表示 NSID 中的第二个标识字段。
LOCUS	最多包含 20 个字符的 VARCHAR, 它表示 NSID 中的第三个标识字段。
DESCRIPTION	最多包含 1017 个字符的 VARCHAR。

### **LSDefineParse2\_2** 用户定义的函数:

*LSDefineParse2\_2* 对带有由一对并置的双字段 NSID 组成的混合标识的定义行进行语法分析。此函数的结果是包含 6 个列的表:

表 103. *LSDefineParse2\_2* 用户定义的函数结果表的列描述

列名	描述
ROWID	一个整数, 它对函数返回的行进行编号。
TAG1	最多包含 3 个字符的 VARCHAR, 它表示第一个标识的 NSID 标记。
IDENTIFIER1	最多包含 20 个字符的 VARCHAR, 它表示第一个 NSID 的第二个标识字段。
TAG2	最多包含 3 个字符的 VARCHAR, 它表示第一个标识的 NSID 标记。
IDENTIFIER2	最多包含 20 个字符的 VARCHAR, 它表示第二个 NSID 的第二个标识字段。
DESCRIPTION	最多包含 1015 个字符的 VARCHAR。

### **LSDefineParse2\_3** 用户定义的函数:

*LSDefineParse2\_3* 对带有由双字段 NSID 与三字段 NSID 的并置组成的复合标识的定义行进行语法分析。输入定义行中的并置顺序 (无论是双字段 NSID 位于三字段 NSID 之前还是相反) 并不重要。此函数的结果是包含 7 个列的表:

表 104. *LSDefineParse2\_3* 用户定义的函数结果表的列描述

列名	描述
ROWID	一个整数, 它对函数返回的行进行编号。
TAG1	最多包含 3 个字符的 VARCHAR, 它表示双字段标识的 NSID 标记。
IDENTIFIER	最多包含 20 个字符的 VARCHAR, 它表示双字段 NSID 的第二个标识字段。
TAG2	最多包含 3 个字符的 VARCHAR, 它表示三字段标识的 NSID 标记。
ACCESSION	最多包含 20 个字符的 VARCHAR, 它表示三字段 NSID 的第二个标识字段。
LOCUS	最多包含 20 个字符的 VARCHAR, 它表示三字段 NSID 的第三个标识字段。
DESCRIPTION	最多包含 1013 个字符的 VARCHAR。

### **LSDefineParse3\_3** 用户定义的函数:

LSDeflineParse3\_3 对带有由一对三字段 NSID 组成的复合标识的定义行进行语法分析。此函数的结果是包含 8 个列的表:

表 105. LSDeflineParse3\_3 用户定义的函数结果表的列描述

列名	描述
ROWID	一个整数, 它对函数返回的行进行编号。
TAG1	最多包含 3 个字符的 VARCHAR, 它表示第一个标识的 NSID 标记。
ACCESSION1	最多包含 20 个字符的 VARCHAR, 它表示第一个 NSID 的第二个标识字段。
LOCUS1	最多包含 20 个字符的 VARCHAR, 它表示第一个 NSID 的第三个标识字段。
TAG2	最多包含 3 个字符的 VARCHAR, 它表示第二个标识的 NSID 标记。
ACCESSION2	最多包含 20 个字符的 VARCHAR, 它表示第二个 NSID 的第二个标识字段。
LOCUS2	最多包含 20 个字符的 VARCHAR, 它表示第二个 NSID 的第三个标识字段。
DESCRIPTION	最多包含 1014 个字符的 VARCHAR。

**相关参考:**

- 第 423 页的『LSDeflineParse 用户定义的函数 - 示例』

## LSDeflineParse 用户定义的函数 - 示例

本主题包含 7 个示例, 这些示例显示 LSDeflineParse 用户定义的函数如何对定义行进行语法分析以生成结果表。

以下示例查询和结果表显示 LSDeflineParse2 用户定义的函数如何对包含双字段 NSID 的定义行进行语法分析:

```
select *
from table(DB2LS.LSDeflineParse2(
           '>gi|12346 hypothetical protein 185 -wheat chloroplast')) as t
```

结果表包含下列数据:

表 106. LSDeflineParse2 用户定义的函数结果数据

列名	数据
ROWID	1
TAG	gi
IDENTIFIER	12346
DESCRIPTION	hypothetical protein 185 - wheat chloroplast

以下示例查询和结果表显示 LSDeflineParse3 用户定义的函数如何对包含三字段 NSID 的定义行进行语法分析:

```
select *
from table(DB2LS.LSDeflineParse3('
           >gb|U37104|APU37104 Aethia pusilla cytochrome b gene')) as t
```

结果表包含下列数据:

表 107. *LSDefineParse3* 用户定义的函数结果数据

列名	数据
ROWID	1
TAG	gb
ACCESSION	U37104
LOCUS	APU37104
DESCRIPTION	Aethia pusilla cytochrome b gene

以下示例查询和结果表显示 *LSDefineParse2\_2* 用户定义的函数如何对包含由一对双字段 *NSID* 组成的复合标识的定义行进行语法分析:

```
select *
from table(DB2LS.LSDefineParse2_2(
    '>gb|U37104|gim|73401A Aethia pusilla cytochrome b gene')) as t
```

结果表包含下列数据:

表 108. *LSDefineParse2\_2* 用户定义的函数结果数据

列名	数据
ROWID	1
TAG1	gb
IDENTIFIER1	U37104
TAG2	gim
IDENTIFIER2	73401A
DESCRIPTION	Aethia pusilla cytochrome b gene

以下示例查询包含带有复合标识的定义行, 该复合标识由双字段 *NSID* 和三字段 *NSID* 的并置组成。此示例显示 *LSDefineParse2\_3* 函数如何对该定义行进行语法分析。

```
select *
from table(DB2LS.LSDefineParse2_3('
    >gi|12346|gp|CAA44030.1|CHTAHSRA_4
    hypothetical protein 185 - wheat chloroplast')) as t
```

结果表包含下列数据:

表 109. *LSDefineParse2\_3* 用户定义的函数结果数据

列名	数据
ROWID	1
TAG1	gi
IDENTIFIER	12346
TAG2	gp
ACCESSION	CAA44030.1
LOCUS	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast



以下示例查询包含带有复合标识的定义行，该复合标识由三字段 NSID 和双字段 NSID 的并置组成。此示例显示 LSDeflineParse2\_3 函数如何对该定义行进行语法分析。

```
select *
from table(DB2LS.LSDeflineParse2_3('
    >gp|CAA44030.1|CHTAHSRA_4|gi|12346
    hypothetical protein 185 - wheat chloroplast')) as t
```

结果表包含下列数据:

表 110. LSDeflineParse2\_3 用户定义的函数结果数据

列名	数据
ROWID	1
TAG1	gi
IDENTIFIER	12346
TAG2	gp
ACCESSION	CAA44030.1
LOCUS	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast

以下示例查询和结果表显示 LSDeflineParse3\_3 用户定义的函数如何对包含带有一对三字段 NSID 的复合标识的定义行进行语法分析:

```
select * from table(DB2LS.LSDeflineParse3_3('
    >dbj|AAD55586.1|AF055084_1|gp|CAA44030.1|CHTAHSRA_4
    hypothetical protein 185 - wheat chloroplast')) as t
```

结果表包含下列数据:

表 111. LSDeflineParse3\_3 用户定义的函数结果数据

列名	数据
ROWID	1
TAG1	dbj
ACCESSION1	AAD55586.1
LOCUS1	AF055084_1
TAG2	gp
ACCESSION2	CAA44030.1
LOCUS2	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast

可以使用任何定义行用户定义的函数来对复合定义行进行语法分析。以下示例查询包含带有由 Control-A 字符分隔的多个定义的复合定义行。可以在 NCBI 的非冗余蛋白质数据库 nr 中找到此类型的定义行。此示例显示 LSDeflineParse2\_3 函数如何对该定义行进行语法分析。

```
select *
from table(DB2LS.LSDeflineParse2_3('
    >gi|12346|gp|CAA44030.1|CHTAHSRA_4
    hypothetical protein 185 - wheat chloroplast
    ^Agp|CAA44030.1|CHTAHSRA_4|gi|12346
    hypothetical protein 185 - wheat chloroplast')) as t
```

结果表包含下列数据:

表 112. *LSDeflineParse2\_3* 用户定义的函数结果数据

列名	数据	数据
ROWID	1	2
TAG1	gi	gi
IDENTIFIER	12346	12346
TAG2	gp	gp
ACCESSION	CAA44030.1	CAA44030.1
LOCUS	CHTAHSRA_4	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast	hypothetical protein 185 - wheat chloroplast

相关参考:

- 第 421 页的『*LSDeflineParse* 用户定义的函数』

## 一般化模式匹配用户定义的函数

一般化模式匹配用户定义的函数标识给定字符串中的令人感兴趣的区域，如核苷酸或缩氨酸序列。

### **LSPatternMatch** 用户定义的函数

►—DB2LS.LSPatternMatch—(*input character sequence, pattern*)—►

#### **input character sequence**

此字符串表示法必须具有 `VARCHAR` 数据类型，并且实际长度不超过 32672 个字节。

#### **pattern**

在任何有效的 Perl 正则表达式中指定的模式。此字符串表示法必须具有 `VARCHAR` 数据类型，并且实际长度不超过 32672 个字节。

模式名是 `DB2LS`。

可以使用 `LSPatternMatch` 用户定义的函数来搜索指定的模式的输入核苷酸或缩氨酸序列。

此函数的结果是一个整数，它表示模式的第一个匹配在序列中的位置。如果没有匹配，则此函数返回零值。

如果模式是使用 `PROSITE` 语法书写的，则可以通过 `LSPrositePattern` 用户定义的函数将它们转换为 Perl 语法。然后，可以将经过转换的语法与 `LSPatternMatch` 用户定义的函数配合使用。

相关参考:

- 第 427 页的『*LSPatternMatch* 用户定义的函数 - 示例』
- 第 428 页的『*LSPrositePattern* 用户定义的函数』

## LSPatternMatch 用户定义的函数 - 示例

在以下示例中，查找与“coward”、“cowage”、“cowboy”或“cowl”相匹配的字符串的开始位置。

```
values DB2LS.LSPatternMatch('joe the cowboy is next', 'cow(ard|age|boy|l)')
```

此函数按字符进行搜索，在此示例中，将返回值 9。字符串“cowboy”从位置 9 开始（假定第一个位置是 1）。

在下一个示例中，查找与“not”或“non”相匹配的字符串的开始位置：

```
values DB2LS.LSPatternMatch('match not and non but  
no match for no or none', 'no[tn] ')
```

此函数按字符进行搜索，在此示例中，将返回值 7。字符串“not”从位置 7 开始（假定第一个位置是 1）。

LSPatternMatch 在 Select 语句中非常有用，它可以用来通过使用 PERL 语法来对结果进行过滤，与 SQL LIKE 语句相比，PERL 语法是功能更为强大的语法。在以下示例中，对 Blast 输出使用 LSPatternMatch 来过滤与特定模式相匹配的基因：

```
SELECT BlastOutput.*  
FROM BlastOutput  
WHERE db21s.LSPatternMatch(HSP_H_Seq, 'F[GSTV]PRL') > 0;
```

如果您更熟悉 PROSITE 语法，则可以将 LSPrositePattern 函数与上述查询配合使用。将查询更改为：

```
SELECT BlastOutput.*  
FROM BlastOutput  
WHERE db21s.LSPatternMatch(HSP_H_Seq,  
db21s.LSPrositePattern('F-[GSTV]-P-R-L.')) > 0;
```

模式匹配函数对于搜索其它类型的文本以及核苷酸或缩氨酸序列而言非常有用。当性能是关心的主要问题时，请考虑使用 SQL LIKE 语句。

以下示例显示了一个查询，该查询根据在比对的主题或目标行中找到的蛋白质图谱来过滤 BLAST hsp 比对。此示例摘自 Zhang,Z., Schaffer,A.A., Miller,W., Madden,T.L., Lipman,D.J., Koonin,E.V. and Altschul,S.F. (1998) Protein sequence similarity searches using patterns as seeds. *Nucl. Acids Res.*, **26** (3896-3990)。

以下查询只返回主题序列包含 P 循环 ATPase 域 [GA]xxxxGK[ST] 的比对。此查询对 NCBI 的非冗余蛋白质序列数据库使用 CED4（即细胞死亡 *Caenorhabditis elegans* 调节器）作为查询序列。数据库通过转换 GenBank 条目 X69016 的 CDS 特征来检索 Blast 查询序列。

```
SELECT HSP_Q_Seq, HSP_Midline, HSP_H_Seq  
FROM BlastP b, GBseq gs, gbfeat gf, gbqual gq  
WHERE gs.PRIMARYACCESSION = 'X69016' and  
gs.sequencekey = gf.sequencekey and  
gf.featurejoinkey = gq.featurejoinkey and  
gf.FeatureKey = 'CDS' and  
gq.QualifierName = 'translation' and  
gq.QualifierValue = b.BlastSeq and  
db21s.LSPatternMatch(HSP_H_Seq,  
db21s.LSPrositePattern('[GA]-x(4)-G-K-[ST].')) > 0;
```

可以使用下一个示例查询来在基因的序列中查找包含与规范查询序列相关的公认单一核苷酸多态性 (SNP) 的 HSP。此查询摘自 *Extending traditional query-based integration approaches for functional characterization of post-genomic data*. (2001) Barbara A Eckman, Anthony S Kosky, and Leonardo A Laroco Jr. *Bioinformatics* 17(7), 587-601。

此查询使用对 Blast HSP 中间行的模式匹配来查找以下模式: 大于或等于 20 个完全匹配, 后跟单一不匹配, 后跟大于或等于 20 个完全匹配。即, 比对的中间行包含 20 个 “I” 字符, 接着是一个空格, 接着又是 20 个 “I” 字符。

此示例还显示了对不是核苷酸或缩氨酸序列的字符串使用 LSPatternMatch 用户定义的函数。

```
SELECT HSP_Info, HSP_Midline, HSP_H_Seq
FROM BlastOutput
WHERE db2ls.LSPatternMatch(HSP_Midline, '\\|{20} \\|{20}') > 0;
```

可以将上一个查询重新编写为:

```
SELECT HSP_Info, HSP_Midline, HSP_H_Seq, func.Position, func.Match
FROM BlastOutput,
TABLE(SELECT * AS c FROM TABLE(
LSMultiMatch(HSP_Midline, '\\|{20} \\|{20}') )
AS f) AS func
```

第二个查询将返回包含匹配以及匹配的字符串及其在序列中的位置的 Blast 行。

BlastOutput 是 BlastN 昵称的视图。

**相关参考:**

- 第 429 页的『LSPrositePattern 用户定义的函数 - 示例』
- 第 426 页的『LSPatternMatch 用户定义的函数』
- 第 428 页的『LSPrositePattern 用户定义的函数』

## LSPrositePattern 用户定义的函数

►—DB2LS.LSPrositePattern—(pattern)—◄

### pattern

Prosite 语法指定的模式匹配语法。此字符串表示法必须具有 VARCHAR 数据类型, 并且实际长度不超过 32672 个字节。

模式名是 DB2LS。

使用 LSPrositePattern 用户定义的函数来从 PROSITE 语法转换为 PERL 语法。然后, 可以将经过转换的语法与 LSPatternMatch、LSMultiMatch 和 LSMultiMatch3 用户定义的函数配合使用。

此函数的结果是一个字符串, 它表示具有 Perl 语法的正则表达式。此字符串表示法必须具有 VARCHAR 数据类型, 并且实际长度不超过 32672 个字节。

**相关参考:**

- 第 429 页的『LSPrositePattern 用户定义的函数 - 示例』

- 第 426 页的『LSPatternMatch 用户定义的函数』

## LSPrositePattern 用户定义的函数 - 示例

在以下示例中，将模式从 PROSITE 语法转换为 PERL 语法。

```
values db21s.LSPrositePattern('[AC]-x-V-x(4)-{ED}.');
```

此函数将具有 PROSITE 语法的输入模式转换为具有 Perl 语法的等价模式，如以下示例所示：

```
[AC].V.{4}[^ED]
```

下一个示例将另一个语法模式从 PROSITE 转换为 PERL 语法：

```
values db21s.LSPrositePattern('<A-x-[ST](2)-x(0,1)-V.');
```

此函数根据输入模式从 PROSITE 语法转换字符串并返回以下结果：

```
\AA.[ST]{2}.{0,1}V
```

下一个示例将与标识号为 PS01205 的 PROSITE 数据库条目相应的模式转换为作为模式匹配函数的输入使用的 PERL 模式。

```
values db21s.LSPrositePattern('R-P-L-[IV]-x-[NS]-F-G-S-[CA]-T-C-P-x-F.');
```

此查询的结果为：

```
RPL[IV].[NS]FGS[CA]TCP.F
```

下一个示例显示可以如何在查询中使用此函数。此查询只打印出与指定的 PROSITE 模式相匹配的序列。

```
SELECT H_Accession, HSP_Info, HSP_H_Seq
FROM BlastOutput
WHERE db21s.LSPatternMatch( HSP_H_Seq,
    db21s.LSPrositePattern('R-P-L-[IV]-x-[NS]-F-G-S-[CA]-T-C-P-x-F.')) > 0;
```

下一个示例转换与标识为 PS00261 的 PROSITE 条目相应的模式：

```
values db21s.LSPrositePattern('C-[STAGM]-G-[HFYL]-C-x-[ST].');
```

此查询的结果为：

```
C[STAGM]G[HFYL]C.[ST]
```

**相关参考：**

- 第 427 页的『LSPatternMatch 用户定义的函数 - 示例』
- 第 428 页的『LSPrositePattern 用户定义的函数』

## 正则表达式支持

正则表达式支持是由 PCRE 库程序包提供的，这是由 Philip Hazel 编写的开放式源代码软件，版权归英国剑桥大学所有。

可以在 <ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/> 找到源代码。

**相关概念：**

- 第 411 页的『生命科学用户定义的函数 - 概述』

## GeneWise 用户定义的函数

GeneWise 用户定义的函数将蛋白质序列与基因的序列比对。

GeneWise 是常常用来将蛋白质序列与基因的 DNA 序列比对的组件，从而允许基因内区和移码错误。

### 链接到 GeneWise

本主题描述链接到 GeneWise 库的过程。

过程:

要链接至 GeneWise 库:

1. 从 [www.ebi.ac.uk/Wise2](http://www.ebi.ac.uk/Wise2) 下载 Wise2 程序包版本 2.1.20c。
2. 将归档展开到您首选的文件夹中。
3. 借助 pthread 支持编译程序包。有关此步骤的更多信息，请参阅 Wise2 文档。

在 HP-UX 联合服务器上，您需要在编译源代码之前将选项 `+z` 添加到编译器选项中。要添加此选项，请从 Wise2 程序包的根目录打开文件 `makefile`，并更改以下示例中所显示的行:

```
CFLAGS = -c -O -DPTHREAD to: CFLAGS = -c -O -DPTHREAD +z
```

4. 在 `make api` 的根目录中运行它。
5. 将 `WISE2_HOME` 环境变量设置为指向 Wise2 程序包根目录。
6. 在 `sqllib/cfg/db2dj.ini` 文件中设置 `WISECONFIGDIR` 变量以指向 `wisecfg` 子目录。例如，如果在 `/usr/wise2.1.20c/` 中安装了 Wise2 程序包，则将 `WISECONFIGDIR=/usr/wise2.1.20c/wisecfg/` 添加到 `db2dj.ini` 文件中。
7. 运行 `djxlinkLSGeneWise` 脚本，该脚本位于 `sqllib/bin` 目录中。
8. 检查 `djxlinkLSGeneWise` 脚本的输出。此输出文件 `djxlinkLSGeneWise.out` 位于 `sqllib/function` 目录中。
9. 如果没有报告错误的话，则表示库已构建成功。

相关参考:

- 第 430 页的『LSGeneWise 用户定义的函数』

### LSGeneWise 用户定义的函数

►►—DB2LS.LSGeneWise—(*protein sequence*, *DNA\_sequence*)—►►

#### protein sequence

描述缩氨酸序列的有效字符串表示法。此字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。

#### DNA\_sequence

描述核苷酸序列的有效字符串表示法。此字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。

表 113 显示了 LSGeneWise 函数返回的一行输出表。

表 113. LSGeneWise 函数返回的输出表的列名、类型和描述

列名	类型	描述
PROTEIN_OFFSET	INTEGER	表示在输入蛋白质序列中找到比对的位置的开始偏移。
DNA_OFFSET	INTEGER	表示在输入 DNA 序列中找到比对的位置的开始偏移。
PROTEIN	VARCHAR(32672)	输入序列中的片段，它表示比对的序列。
SIMILARITY	VARCHAR(32672)	显示蛋白质与 DNA 序列之间的匹配。完全匹配由相应的符号字母作标记。不完全但具有肯定评分的匹配由“+”号指示，不匹配由空格指示。
TRANSLATED_DNA	VARCHAR(32672)	经过转换的 DNA 序列。此序列可能包含短横线和特殊符号，如删除和基因内区。
DNA	VARCHAR(32672)	带有特殊标记（例如，移码和基因内区）的 DNA 序列。

GeneWise 程序的输出与 LSGeneWise UDF 的输出之间的相应关系如下：

- GeneWise 程序打印的蛋白质和 DNA 偏移与 PROTEIN\_OFFSET 和 DNA\_OFFSET 列相匹配。
- GeneWise 在第一行上打印的蛋白质序列与 PROTEIN 列相匹配。
- 相似行（即 GeneWise 输出的第二行）与 SIMILARITY 列相匹配。
- GeneWise 输出的第三行与 TRANSLATED\_DNA 列相匹配。
- 通过垂直地阅读 GeneWise 输出的第四、第五和第六行，这些行组合到 DNA 列中。

使用 LSGeneWise 用户定义的函数来将蛋白质序列与基因的 DNA 序列比对，从而允许基因内区和移码错误。

有关 LSGeneWise 用户定义的函数输出的更多信息，请参阅 <http://www.ebi.ac.uk/Wise2>。

#### 相关任务：

- 第 430 页的『链接到 GeneWise』

#### 相关参考：

- 第 431 页的『LSGeneWise 用户定义的函数 - 示例』

## LSGeneWise 用户定义的函数 - 示例

以下示例显示了使用 LSGeneWise 用户定义的函数的查询以及结果数据。

```
select protein_offset, dna_offset, protein, similarity, translated_dna, dna
from table( db21s.LSGeneWise( '
    VEPKRAVPRQIDSPNAGATVKKLVFGALKDDHDEQSIRDYFQHFIGNIVDINIVIDKETGK
    KRGFADFVEFDDYDPVDKVVVLQKQHLNGMKMVDVKKALPKQNDQQGGGGRRGGPGRAGGNR
    GNMGGGNYGNQGGGNWNGGNNWGNR',
    'CACTTAAGTGAAGATATTTGTTGGTGGCATTAAAGAAGACACTGAAGAACATCACCTAAG
    AGATTATTTGAACAGTATGGAAAATTGAAGTGATTGAAATCATGACTGACCGAGGCAGTGG
    CAAGAAAAGGGCTTTGCCTTRGTAACCTTTGACGACCATGACTCCGTGGATAAGATTGTCAT
    TCAGAAATACCATACTGTGAATGGCCACAACCTGTGAAGTTAGAAAAGCCCTGTCAAAGCAAGA
```

```
GATGGCTAGTGCCTTCATCCAGCCAAAGAGGTGCGAAGTGGTTCTGGAACTTTGGTGGTGGTGG
TGGAGGTGGTTTCGGTGGGAATGACAACCTTCGGTCGTGGAGGAACTTCAGTGGTGGTGGTYG
CTTTGGTGGCAGCCGTGGTGGTGGTGGATATGGTGGC' ) ) as f;
```

表 114. 结果表

列	数据
PROTEIN_OFFSET	23
DNA_OFFSET	14
PROTEIN	KLFV G A L K D D H D E Q S I R D Y F Q H F G N I V D I N I V I D K E T G K K R G F A F V E F D D Y D P V D K V V L Q K Q H Q L N G K M V D V K K A L P K Q N D Q Q G G G G R G G P G G R A G G N R G N M G G G N Y G N Q N G G N W N N G G N
SIMILARITY	K+FVG +K+D +E +RDYF+ +G I I I+ D+ +GKKRGFA+V FDD+D VDK+V+QK H +NG +V+KAL KQ RG G GN+GGG G G N+ GGN
TRANSLATED_DNA	KIFVGGIKEDTEEHHLRDYFEQYQKIEVIEIMTDRGSGK KRGFAxVTFDDHDSVDKIVIQKYHTVNGHNCCEVRKAL S K Q E M A S A S S S Q R G R S G S - - - - - GNFGGGRGGGFGGNDNFRGNGN
DNA	aagatattgttggtgcattaaagaagacactgaagaacatcacctaagagat...

**相关任务:**

- 第 430 页的『链接到 GeneWise』

**相关参考:**

- 第 430 页的『LSGeneWise 用户定义的函数』

## 图谱用户定义的函数

图谱用户定义的函数匹配核苷酸或氨基酸序列中的模式。

### LSBarCode 用户定义的函数

►►—DB2LS.LSBarCode—(input string sequence)————►►

**input string sequence**

表示两个序列段之间的 HSP 比对的有效字符串。此字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。

模式名是 DB2LS。

使用 LSBarCode 用户定义的函数来将某个序列用作输入并通过将空格和加号之外的每个字符替换为垂直竖线 (|) 生成另一个序列。

此函数的结果是表示条形码序列的变量字符序列。

**相关参考:**

- 第 433 页的『LSBarCode 用户定义的函数 - 示例』
- 第 434 页的『LSMultiMatch 用户定义的函数』
- 第 435 页的『LSMultiMatch3 用户定义的函数』



## LSBarCode 用户定义的函数 - 示例

此示例从字符串序列创建条形码:

```
values db21s.LSBarCode(
    'MDY +G++L GN ++ +PASLTK+MT YVV +A+ + +I D+VTVG+DAWA NP ')
```

以上 values 语句的结果为:

```
||| +|++| || ++ +|||||+|| ||| +|+ + +| |+||||+|||| ||
```

下一个示例显示此函数的更实际的用法。假定运行 BLAST 搜索的研究员只希望返回某些 HSP 比对: 这些 HSP 比对所包含的脯氨酸少于它们的完全匹配的 25%。此示例使用此函数来计算脯氨酸 (符号 'P') 在 BLAST 返回的比对所包含的完全匹配中所占的百分比。注意, 此示例还调用了 LSMultiMatch3 用户定义的函数。此查询使用匹配函数来查找完全匹配。由于 Blast 并不总是在比对中返回竖线 ( “|” ) 序列, 所以在此查询中将此函数与 LSBarCode 函数配合使用。以下示例显示了这一点:

```
查询: MDYTTGQILTAGNEHQQRNPASLTKLMTGYVVDRAIDSHRITPDDIVTVGRDAWAKDNPV
比对: MDY +G++L GN ++ +PASLTK+MT YVV +A+ + +I D+VTVG+DAWA NP
目标: MDYASGKVLAEAGNADEKLDPASLTKIMTSYVVGQALKADKIKLDMVTVGKDAWATGNPA
```

为了确保输出与正确的竖线序列比对, 请使用 LSBarCode 函数。该函数将使用竖线来替换除空格和加号以外的所有字符。

```
SELECT BlastOutput.* , float( p ) / float( m ) AS percent_prolines
FROM
  BlastOutput b,
  table(SELECT COUNT(*) AS p FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, 'P',
      db21s.LSBarCode(b.HSP_Midline), '\\|',
      b.HSP_H_Seq, 'P')
    ) AS f
  ) AS y,
  table(SELECT COUNT(*) AS m FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, '.',
      db21s.LSBarCode(b.HSP_Midline), '\\|',
      b.HSP_H_Seq, '.')
    ) AS f
  ) AS z
WHERE float(p) / float(m) < 0.25;
```

在此查询中, BlastOutput 实际上是 Blast 昵称的视图。此查询使用 LSMultiMatch3 函数来返回比对的完全匹配。第一处使用返回符号 “P” 的完全匹配, 第二处使用返回所有完全匹配。表 115 显示了结果表中的行。

表 115. 样本结果行

HSP_Q_SEQ	HSP_H_SEQ	HSP_INFO	PERCENT_PROLINES
NIWDFMQGN...	NIWDFMQGN...	Identities = 80/80 (100%), Positives = 80/80 (100%), Gaps = 0/80 (0%)	+2.5000000000000000E-002

上一个查询摘自 Extending traditional query-based integration approaches for functional characterization of post-genomic data. (2001) Barbara A Eckman, Anthony S Kosky and Leonardo A Laroco Jr. *Bioinformatics* 17(7), 587-601.

#### 相关参考:

- 第 435 页的『LSMultiMatch3 用户定义的函数 - 示例』
- 第 432 页的『LSBarcode 用户定义的函数』

## LSMultiMatch 用户定义的函数

►—DB2LS.LSMultiMatch—(*input nucleotide or peptide sequence, pattern*)—►

### input nucleotide or peptide sequence

描述核苷酸或缩氨酸序列的有效字符串表示法。此字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。

### pattern

Perl 语言指定的模式匹配语法。此字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。

模式名是 DB2LS。

使用 LSMultiMatch 用户定义的函数来为输入序列中的每个不重叠的匹配返回一个表。每个表都由开始位置和匹配序列片段组成。

此函数的结果是带有 2 个列的表。第一个列是一个整数，它表示模式的匹配在序列中的开始位置。第二个列是匹配序列片段。

#### 相关参考:

- 第 434 页的『LSMultiMatch 用户定义的函数 - 示例』
- 第 432 页的『LSBarcode 用户定义的函数』
- 第 435 页的『LSMultiMatch3 用户定义的函数』

## LSMultiMatch 用户定义的函数 - 示例

此示例查找所有从输入获取的非重叠匹配的位置和匹配片段。

```
SELECT position, match FROM table
  (LSMultiMatch('match not and non but no match for no or none',
                'no[tn] ')) as f
```

此查询返回基于这个 Select 语句的表，该表显示了匹配结果:

表 116. 返回多行的 LSMultiMatch 的结果

POSITION	MATCH
7	not
15	non

LSMultiMatch 返回所有匹配的位置和匹配字符串。以下示例在 Entrez 核苷酸中搜索包含特定图谱的序列条目。此查询打印序列标识和匹配的序列。位于开头和结尾处的子模式 “.{0,9}” 必须与序列之前和之后的最多 9 个字符相匹配。此查询还打印这些字符。

```
select SequenceKey, Position, Match from GBSeq,
  table(db2ls.LSMultiMatch(Sequence, '.{0,9}(ATG|CGC)ACGGGC.{0,9}') )
  as fmatch
WHERE entrez.contains(KeywordList,
  'Na/K/2Cl cotransporter AND nkcc1 gene') = 1;
```

此查询的结果如下:

表 117. 搜索 Entrez 数据

SEQUENCEKEY	POSITION	MATCH
N02B59AE0.04DD4E84	1	TGCTTGGTGATGACGGGCTACCCAAC
N02B59AE0.04DD4E84	91	GGCCATGTTTCGCACGGGCTCCAGAAGG
N02B59AE0.04DC5EF4	1	TGCTTGGTGATGACGGGCTACCCAAC
N02B59AE0.04DC5EF4	91	GGCCATGTTTCGCACGGGCTCCAGAAGG

相关参考:

- 第 434 页的『LSMultiMatch 用户定义的函数』
- 第 432 页的『LSBarCode 用户定义的函数』
- 第 435 页的『LSMultiMatch3 用户定义的函数』

## LSMultiMatch3 用户定义的函数

►DB2LS.LSMultiMatch3—(*input string1, pattern1, input string2, pattern2, input string3, pattern3*)—►

### input strings

描述核苷酸或缩氨酸序列的有效字符串表示法, 或者是来自 Blast 比对的 HSP\_Midline 字符串。此字符串表示法必须具有 VARCHAR 数据类型, 并且实际长度不超过 32672 个字节。

### pattern

Perl 语言指定的模式匹配语法。此字符串表示法必须具有 VARCHAR 数据类型, 并且实际长度不超过 32672 个字节。

模式名是 DB2LS。

使用 LSMultiMatch3 用户定义的函数来输入 3 个模式和 3 个字符串并返回全部 3 个字符串与它们各自的模式相匹配的任何位置。可以使用这个用户定义的函数来对比对执行模式匹配。

此函数的结果是带有 4 个列的表。第一个列是一个整数, 它表示模式的匹配在所有序列中的开始位置。此函数将所有字符串一起固定在第一个位置处。第二列、第三列和第四列是匹配序列片段。

相关参考:

- 第 435 页的『LSMultiMatch3 用户定义的函数 - 示例』
- 第 434 页的『LSMultiMatch 用户定义的函数』
- 第 432 页的『LSBarCode 用户定义的函数』

## LSMultiMatch3 用户定义的函数 - 示例

以下示例使用此函数来计算特定氨基酸符号在 Blast 返回的完全匹配中所占的百分比。注意, 此示例还调用了 LSBarCode 用户定义的函数。查询需要这样做的原因是 Blast 并不总是在比对中返回竖线 ( “|” ) 序列。以下示例说明了这一点:

```

查询:      MDYTTGQILTAGNEHQQRNPASLTKLMTGYVVDRAIDSHRITPDDIVTVGRDAWAKDNPV
比对:      MDY +G++L GN ++ +PASLTK+MT YVV +A+ + +I D+VTVG+DAWA NP
目标:      MDYASGKVLAEAGNADEKLDPASLTKIMTSYVVGQALKADKIKLTDMMVTGKDAWATGNPA

```

为了确保输出与正确的竖线序列比对，请使用 `LSBarCode` 函数来转换序列。此函数使用竖线来替换所有的非空格和非 “+” 字符。

```

SELECT BlastOutput.* , float( p )/ float( m ) AS percent_prolines
FROM
  BlastOutput b,
  table(SELECT COUNT(*) AS p FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, 'P',
      db21s.LSBarCode(b.HSP_Midline), '\\|',
      b.HSP_H_Seq, 'P')
    ) AS f
  ) AS y,
  table(SELECT COUNT(*) AS m FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, '.',
      db21s.LSBarCode(b.HSP_Midline), '\\|',
      b.HSP_H_Seq, '.')
    ) AS f
  ) AS z
WHERE float(p) / float(m) < 0.25;

```

在此查询中，`BlastOutput` 是 `Blast Select` 的视图。此查询使用 `LSMultiMatch3` 函数来返回比对的完全匹配。第一处使用返回符号 “P” 的完全匹配，第二处使用返回所有完全匹配。表 118 显示了结果表中的行。

表 118. 样本结果行

HSP_Q_SEQ	HSP_H_SEQ	HSP_INFO	PERCENT_PROLINES
NIWDFMQG...	NIWDFMQG...	Identities = 80/80 (100%), Positives = 80/80 (100%), Gaps = 0/80 (0%)	+2.500000000000000E-002

上一个查询摘自 *Extending traditional query-based integration approaches for functional characterization of post-genomic data.* (2001) Barbara A Eckman, Anthony S Kosky and Leonardo A Laroco Jr. *Bioinformatics* 17(7), 587-601.

以下示例在三个独立的字符串片段中查找三个独立的模式:

```

SELECT position, match_1, match_2, match_3
FROM table(db21s.LSMultiMatch3('zaza', 'a', 'abab',
  'b', 'bcbc', 'c')) as f

```

它返回所有匹配的位置和匹配字符串，如下表所示:

表 119. 使用三个输入的多重匹配的结果

POSITION	MATCH_1	MATCH_2	MATCH_3
2	a	b	c
4	a	b	c

下一个示例在三个独立的字符串片段中查找三个独立的模式:

```
SELECT position, match_1, match_2, match_3
FROM table
(LSMultiMatch3('cbccbccccbbccccbbcccc', 'c{1,3}b{1,3}c{1,3}',
'abcdefghijklmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyz',
'.', '0123456789012345678901234567890123456789', '\d')) as f
```

下表显示了结果:

表 120. 使用三个输入的多重匹配的结果

POSITION	MATCH_1	MATCH_2	MATCH_3
1	cbcc	a	0
7	cccbcccc	g	6

#### 相关参考:

- 第 433 页的『LSBarcode 用户定义的函数 - 示例』
- 第 432 页的『LSBarcode 用户定义的函数』
- 第 435 页的『LSMultiMatch3 用户定义的函数』

## 逆向用户定义的函数

逆向用户定义的函数将核苷酸或氨基酸序列反转。

### LSRevComp 用户定义的函数

►►DB2LS.LSRevComp—(*input nucleotide sequence*)◄◄

#### input nucleotide sequence

描述核苷酸序列的有效字符串表示法。此序列可以包含 IUPAC 多义代码。字符串表示法必须具有 VARCHAR 数据类型, 并且实际长度不超过 32672 个字节。

模式名是 DB2LS。

此函数的结果是具有 VARCHAR 数据类型并且实际长度不超过 32672 个字节的字符串, 它表示核苷酸序列的逆向补充。

#### 相关参考:

- 第 437 页的『LSRevComp 用户定义的函数 - 示例』
- 第 438 页的『LSRevNuc 用户定义的函数』
- 第 439 页的『LSRevPep 用户定义的函数』

### LSRevComp 用户定义的函数 - 示例

每当要使用任何接受核苷酸序列的内置函数时, 都可以在 SQL 语句中使用 LSRevComp 函数。例如:

```
SELECT DB2LS.LSRevComp(:NucSeq) FROM SYSDDUMMY1;
```

此示例使用该函数来返回来自变量的输入序列的逆向补充。

如果使用无效的字符串或无效的数据类型, 则会收到以下错误消息:

SQL0443N 例程“DB2LS.LSREVCOMP”（特定名称“LSREVCOMP”）已经返回带有诊断文本“序列无效”的错误 SQLSTATE。SQLSTATE=38608

如果输入字母不正确，则将发生异常。

以下示例显示 LSRevComp 用户定义的函数是如何在查询中工作的：

```
SELECT HSP_H_Seq, db21s.LSRevComp(HSP_H_Seq) as REV_HSP_H_Seq
FROM BlastN
WHERE BlastSeq='ccgctagtatttgcaatcttttgatatccaccgaa'
```

此查询的结果显示如下：

HSP_H_SEQ	REV_HSP_H_SEQ
AGTATTGGTCAATCTTTTGAT	ATCAAAAGATTGACCAATACT
TGGTCAATCTTTTGATA	TATCAAAAGATTGACCA
TTGGCCAATCTTTTGATATCC	GGATATCAAAAGATTGGCCAA
TCAATCTTTTGATATCC	GGATATCAAAAGATTGA
GGATATCAAAAGATTGA	TCAATCTTTTGATATCC

5 record(s) selected.

可以将这个逆向函数与其它生命科学用户定义的函数一起使用以转换核苷酸序列的逆向补充，如以下示例所示：

```
values db21s.LSNuc2Pep(
      db21s.LSRevComp('TTTTTCTTATTGTCTTCCTCATGTTTCTTATGTTGCTGATGT'))
```

此查询返回以下结果：

TSAT\*EIR\*GRQ\*EK

相关参考：

- 第 437 页的『LSRevComp 用户定义的函数』

## LSRevNuc 用户定义的函数

►—DB2LS.LSRevNuc—(input nucleotide sequence)—◄

### input nucleotide sequence

描述核苷酸序列的有效字符串表示法。字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。核苷酸序列必须是 DNA 字母表的部分或全部。

模式名是 DB2LS。

此函数的结果是具有 VARCHAR 数据类型并且实际长度不超过 32672 个字节的字符串，它表示核苷酸序列的逆序。

相关参考：

- 第 439 页的『LSRevNuc 用户定义的函数 - 示例』
- 第 437 页的『LSRevComp 用户定义的函数』

- 第 439 页的『LSRevPep 用户定义的函数』

## LSRevNuc 用户定义的函数 - 示例

每当要使用任何接受核苷酸序列的内置函数时，都可以在 SQL 语句中使用 LSRevNuc 函数。例如：

```
SELECT DB2LS.LSRevNuc(:NucSeq) FROM SYSDUMMY1;
```

此示例使用该函数来颠倒来自变量的输入数据。

如果使用无效的字符串或无效的数据类型，则会收到以下错误消息：

```
SQL0443N 例程“DB2LS.LSREVNUC”（特定名称“LSREVNUC”）已经返回带有诊断文本“序列无效”的错误 SQLSTATE。SQLSTATE=38608
```

以下示例显示 LSRevNuc 用户定义的函数在查询中的使用。

```
SELECT HSP_H_Seq, db21s.LSRevNuc(HSP_H_Seq) as REV_HSP_H_Seq
FROM BlastN
WHERE BlastSeq='gtaatacgtaggggctagcgcgggcaactgaagataaac'
```

以下结果表显示了该查询返回的逆向核苷酸序列：

HSP_H_SEQ	REV_HSP_H_SEQ
CGCGGGCAAACCTGAAGATAAAGC	CGAAATAGAAGTCAAACGGGCGC
GCGCTAGCCCCCTACGTATTAC	CATTATGCATCCCCGATCGCG
GTAATACGTAGGGGCTAGCG	GCGATCGGGGATGCATAATG
GTAATACGTAGGGGCTAGCG	GCGATCGGGGATGCATAATG
GTAATACGTAGGGGCTAGCG	GCGATCGGGGATGCATAATG

5 record(s) selected.

相关参考：

- 第 438 页的『LSRevNuc 用户定义的函数』

## LSRevPep 用户定义的函数

►►—DB2LS.LSRevPep—(*input peptide sequence*)—◄◄

### input peptide sequence

描述缩氨酸序列的有效字符串表示法。字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。输入序列必须是蛋白质字母表的一部分。

模式名是 DB2LS。

此函数的结果是具有 VARCHAR 数据类型并且实际长度不超过 32672 个字节的字符串，它表示缩氨酸序列的逆序。

相关参考：

- 第 440 页的『LSRevPep 用户定义的函数 - 示例』

- 第 437 页的『LSRevComp 用户定义的函数』
- 第 438 页的『LSRevNuc 用户定义的函数』

## LSRevPep 用户定义的函数 - 示例

每当要使用任何接受缩氨酸序列的内置函数时，都可以在 SQL 语句中使用 LSRevPep 函数。例如：

```
SELECT DB2LS.LSRevPep(:NucSeq) FROM SYSDUMMY1;
```

此示例使用该函数来颠倒来自变量的输入数据。

如果使用无效的字符串或无效的数据类型，则会收到以下错误消息：

```
SQL0443N 例程“DB2LS.LSREVPEP”（特定名称“LSREVPEP”）已经返回
带有诊断文本“序列无效”的错误 SQLSTATE。SQLSTATE=38608
```

以下示例显示如何在查询中使用 LSRevPep 用户定义的函数：

```
SELECT HSP_H_Seq, db21s.LSRevPep(HSP_H_Seq) as REV_HSP_H_Seq
FROM BlastP
WHERE BlastSeq='MLCEIECRALSTAHTRLIHDFEPRDALTYLEGKNIFTEDH'
```

下表显示查询返回的逆向缩氨酸序列。

HSP_H_SEQ	REV_HSP_H_SEQ
MLCEIECRALSTAHTRLIHDFEPRDALT...	HDETFINKGELYTLADRPEFDHILRTHATS...
RVVSTEHTRLVTDAYPEFSISFTATKN	NKTATFSISFEPYADTVLRTHETSVVR
STAHIRVLRDMVPGDEITCFYGSEFF	FFESGYFCTIEDGPVMDRLVRIHATS
AHTRRCPDHEPRGVITYL	LYTIVGRPEHDPCCRTHA

4 record(s) selected.

相关参考：

- 第 439 页的『LSRevPep 用户定义的函数』

## 转换

转换用户定义的函数将核苷酸序列转换为缩氨酸序列。

### LSNuc2Pep 用户定义的函数

```
DB2LS.LSNuc2Pep(input nucleotide sequence, filepath to external translation table)
```

#### input nucleotide sequence

描述核苷酸序列的有效字符串表示法。字符串表示法必须具有 VARCHAR 数据类型，并且实际长度不超过 32672 个字节。

#### filepath to external translation table

如果使用定制的转换表，则包括文件路径信息以查找转换表。此路径的字符串值一定不能超过 255 个字符。

模式名是 DB2LS。



此函数的结果是具有 VARCHAR 数据类型并且实际长度不超过 10890 个字节的字符串，它表示缩氨酸序列。

输入是使用 IUB 字符集的核苷酸序列。此函数假定第一个基码从核苷酸序列的第一个字符处开始。如果第一个基码不是从核苷酸序列的第一个字符开始的，则对输入序列使用 SUBSTR 函数。

此函数的结果是使用标准氨基酸符号的缩氨酸序列。

此函数:

- 删除输入序列中的空格。
- 忽略读码框架之外的外来核苷酸。
- 如果输入空的核苷酸序列，则返回空输出。

相关参考:

- 第 441 页的『LSNuc2Pep 用户定义的函数 - 示例』
- 第 442 页的『LSTransAllFrames 用户定义的函数』

## LSNuc2Pep 用户定义的函数 - 示例

假定要將核苷酸序列数据转换为缩氨酸序列。此示例假定第一个基码以核苷酸序列的首字符开始。

可以使用 values 语句来调用此函数。唯一的输入是核苷酸序列，如以下示例所示:

```
values db2ls.LSNuc2Pep('TTTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT')
```

以上语句的结果是使用标准氨基酸符号的缩氨酸序列:

```
FFLLSSSSYFLCC*C
```

如果要使用 +2 读码框架来进行转换，则使用以下示例:

```
values LSNuc2Pep(SUBSTR('TTTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT',2))
```

语句中的整数指示开始搜索基码的位置。

这里是在查询中使用此函数作为谓词的一个示例。

```
SELECT *
FROM proteindata
WHERE peptideseq=DB2LS.LSNuc2Pep('TTTTTCTTATTGTCTTCCTCATCG
TATTTCTTATGTTGCTGATGT');
```

表 121 显示了结果。

表 121. 使用 LSNuc2Pep 函数作为谓词的结果

ID	PROTEINNAME	PEPTIDSEQ
1	proteinA	FSYCLPHRISYVAD

以下示例使用外部转换表来將核苷酸序列转换为缩氨酸序列。第一个参数是核苷酸序列，第二个参数是外部转换表的路径。

```
values db2ls.LSNuc2Pep('TTTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT',
'C:\translation.txt')
```

以上使用这个特定转换表的语句的结果是以下字符串:

```
FSYCLPHRISYVAD
```

以下示例将两个用户定义的函数组合起来使用以演示这些函数的其它用途:

```
values DB2LS.LSNuc2Pep(DB2LS.LSRevCompNuc('TTT..'))
```

注意, 上一个示例返回与以下查询相同的结果:

```
select * from table (DB2LS.LSTransAllFrames ('TTT..')) as t
where t.readframe = -1
```

**相关参考:**

- 第 439 页的『LSRevNuc 用户定义的函数 - 示例』
- 第 443 页的『LSTransAllFrames 用户定义的函数 - 示例』
- 第 440 页的『LSNuc2Pep 用户定义的函数』

## LSTransAllFrames 用户定义的函数

→ DB2LS.LSTransAllFrames(*input nucleotide sequence*, *filepath to external translation table*) →

### input nucleotide sequence

描述核苷酸序列的有效字符串表示法。输入序列可以包含 IUPAC 多义代码。字符串表示法必须具有 VARCHAR 数据类型, 并且实际长度不超过 32672 个字节。

### filepath to external translation table

如果使用定制的转换表, 则包括文件路径信息以查找转换表。此路径的字符串值一定不能超过 255 个字符。

模式名是 DB2LS。

使用 LSTransAllFrames 用户定义的函数来从给定的核苷酸序列生成一组缩氨酸序列。这些缩氨酸序列表示输入核苷酸序列的可能转换 (6 个框架为一组)。当输入包含错误或未读码框架时, 此函数非常有用。

此函数的结果是带有 2 个列的表。第一列的标号是 READFRAME, 它表示用于转换的框架。此列包含表示转换的起始位置的整数值。负整数指示反向的转换。第二列 (称为 PEPTIDE) 是具有 VARCHAR 数据类型并且实际长度不超过 10890 个字节的字符串, 它表示缩氨酸序列。

此函数:

- 删除输入序列中的空格。
- 忽略读码框架之外的外来核苷酸。
- 如果输入空的核苷酸序列, 则返回空输出。

**相关参考:**

- 第 443 页的『LSTransAllFrames 用户定义的函数 - 示例』
- 第 440 页的『LSNuc2Pep 用户定义的函数』

## LSTransAllFrames 用户定义的函数 - 示例

假定您想要使用内置转换表来转换核苷酸序列，并且全都以 6 个读码框架为一组。以下示例显示了如何做到这一点：

```
SELECT * FROM table(DB2LS.LSTransAllFrames('TTTTCTTATTGTCTTCCTCATCG
                                           TATTCTTATGTTGCTGATGT')) as t;
```

此查询在一个表中返回缩氨酸，如以下示例所示：

表 122. 转换核苷酸序列的结果

READFRAME	PEPTIDE
1	FLLSSSSYFLCC*C
2	FSYCLPHRISYVAD
3	FLIVFLIVFLMLLM
-1	TSAT*EIR*GRQ*EK
-2	HQQHKKYDEEDNKK
-3	ISNIRNTMRKTIRK

下一个示例使用定制的转换表来转换核苷酸序列，并且全都以 6 个读码框架为一组。

```
SELECT * FROM table
(DB2LS.LSTransAllFrames
 ('TTTTCTTATTGTCTTCCTCATCGTATTCTTATGTTGCTGATGT',
  'C:\msvs6\MyProjects\alin_udf\test\files\translation.txt')) as t;
```

因为输入序列是相同的，并且转换表与构建到函数中的转换表相同，所以结果表与上一个示例的结果表相同。

以下示例将两个用户定义的函数组合起来使用以演示这些函数的其它用途：

```
values DB2LS.LSNuc2Pep(DB2LS.LSRevCompNuc('TTT..'))
```

注意，上一个示例返回与以下查询相同的结果：

```
select * from table (DB2LS.LSTransAllFrames ('TTT..')) as t
where t.readframe = -1
```

以下示例从 LSTransAllFrames 函数生成的输出中选择特定的读码框架。

```
SELECT * FROM
TABLE(db2ls.LSTransAllFrames('TTTTCTTATTGTCTTCCTCATCG
                              TATTCTTATGTTGCTGATGT')) AS t
WHERE t.readframe=-2
```

此查询的结果为：

表 123. Readframe 函数的使用

READFRAME	PEPTIDE
-2	HQQHKKYDEEDNKK

相关参考：

- 第 441 页的『LSNuc2Pep 用户定义的函数 - 示例』
- 第 439 页的『LSRevNuc 用户定义的函数 - 示例』
- 第 442 页的『LSTransAllFrames 用户定义的函数』

---

## 基码频率表格式

基码频率表显示氨基酸向后转换为特定基码时采用的频率。LS Pep2ProbNuc 用户定义的函数使用基码频率表来根据给定的缩氨酸序列确定核苷酸序列。

以下列表描述了基码频率表文件的格式:

- 两个相邻的句点标记表的开头。之前的任何文本都是注释。两个相邻的句点是必需的，即使它们前面没有注释亦如此。
- 此表包含下列各列:
  1. Am-Acid: 氨基酸符号的三字母代码。
  2. Codon: 该氨基酸符号的基码。
  3. Number: 该基码在编译生成该表的基因中的出现次数。
  4. x/1000: 氨基酸的预期出现次数，基因中的每 1000 个转换的基码对。
  5. Fraction: 基码在其同义基码系列中的出现的分数。

产品在 `sqllib/samples/lifesci/ls_udfs` 子目录中提供了样本基码频率表。

**相关参考:**

- 第 418 页的『LS Pep2ProbNuc 用户定义的函数』
- 第 445 页的『基码频率表 - 示例』

---

## 基码频率表 - 示例

图 46 显示了样本基码频率表的格式。

---

Am-Acid	Codon	Number	x/1000	Fraction	..
Gly	GGG	198.00	18.34	0.23	
Gly	GGA	71.00	6.58	0.08	
Gly	GGT	66.00	6.11	0.08	
Gly	GGC	527.00	48.81	0.61	
Glu	GAG	534.00	49.46	0.88	
Glu	GAA	71.00	6.58	0.12	
Asp	GAT	31.00	2.87	0.06	
Asp	GAC	481.00	44.55	0.94	
Val	GTG	396.00	36.68	0.47	
Val	GTA	22.00	2.04	0.03	
Val	GTT	44.00	4.08	0.05	
Val	GTC	384.00	35.57	0.45	
Ala	GCG	446.00	41.31	0.39	
Ala	GCA	71.00	6.58	0.06	
Ala	GCT	116.00	10.74	0.10	
Ala	GCC	503.00	46.59	0.44	
... (截断)					

---

图 46. 样本基码频率表

### 相关参考:

- 第 418 页的『LSPep2ProbNuc 用户定义的函数』
- 第 444 页的『基码频率表格式』

---

## 转换表格式

本主题描述 LSPep2AmbNuc、LSTransAllFrames 和 LSNuc2Pep 生命科学用户定义的函数使用的转换表的格式。

以下列表描述了基码频率表文件的格式:

- 两个相邻的句点标记表的开头。在此之前的任何文本都是注释。
- 表的每一行都由单字母氨基酸符号、三字母氨基酸名称、确定的基码、感叹号和不确定的基码组成。用空格将行中的每个字分隔开。
- 每个基码和氨基酸符号在文件中都只能出现一次。
- 停止基码转换为符号 “\*”。
- 由小写字母组成的基码是起始基码。
- 所有其它基码都是大写的。
- 将无法转换为相应氨基酸符号的基码转换为符号 “X”。

产品在 `sqllib/samples/lifesci/ls_udfs` 子目录中提供了样本转换表。

### 相关概念:

- 第 411 页的『生命科学用户定义的函数 - 概述』

相关参考:

- 第 446 页的『转换表 - 示例』

---

## 转换表 - 示例

图 47 显示了样本转换表的格式。

---

Standard Translation Table				
Symbol	3-letter	Codons	! IUPAC	..
A	Ala	GCT GCC GCA GCG	! GCX	
B	Asx		! RAY	
C	Cys	TGT TGC	! TGY	
D	Asp	GAT GAC	! GAY	
E	Glu	GAA GAG	! GAR	
F	Phe	TTT TTC	! TTY	
G	Gly	GGT GGC GGA GGG	! GGX	
H	His	CAT CAC	! CAY	
I	Ile	ATT ATC ATA	! ATH	
K	Lys	AAA AAG	! AAR	
L	Leu	TTG TTA CTT CTC CTA CTG	! TTR CTX YTR	; YTX
M	Met	atg	! ATG	
N	Asn	AAT AAC	! AAY	
P	Pro	CCT CCC CCA CCG	! CCX	
Q	Gln	CAA CAG	! CAR	
R	Arg	CGT CGC CGA CGG AGA AGG	! CGX AGR MGR	; MGX
S	Ser	TCT TCC TCA TCG AGT AGC	! TCX AGY	; WSX
T	Thr	ACT ACC ACA ACG	! ACX	
V	Val	GTT GTC GTA GTG	! GTX	
W	Trp	TGG	! TGG	
X	Xxx		! XXX	
Y	Tyr	TAT TAC	! TAY	
Z	Glx		! SAR	
*	End	TAA TAG TGA	! TAR TRA	; TRR

---

图 47. 样本转换表

相关概念:

- 第 411 页的『生命科学用户定义的函数 - 概述』

相关参考:

- 第 445 页的『转换表格式』

---

## 第 26 章 KEGG 用户定义的函数

本章说明什么是 KEGG 用户定义的函数，如何将它们添加至联合系统以及如何在查询中使用它们。

---

### KEGG 用户定义的函数 - 概述

基因和基因组京都百科全书 (KEGG) 是包含基因组信息的一系列数据库。KEGG 用户定义的函数是随 DB2<sup>®</sup> Information Integrator 一起提供的一组函数，可以用来存取 KEGG 数据库中的基因组信息。

DB2 Information Integrator 通过 KEGG Web 服务界面可以存取 KEGG 套件中的唯一两个数据库是“途径” (Pathway) 数据库和“序列相似性数据库” (SSDB)。“途径”数据库提供了有关生物过程中的分子相互作用网络的一组数据，包括代谢途径、调节途径和分子。SSDB 是有关完整基因组组合中的蛋白质编码基因的数据集合。

KEGG 用户定义的函数使用 KEGG API 来存取这些数据库。

许多 KEGG 方法将返回一些值的列表，例如，基因或途径。这些方法中的某些方法还需要将这些值的列表作为输入。为了便于组合多种方法中的复合操作，大多数 KEGG 用户定义的函数同时提供了表格式和标量格式。表函数将返回许多单个值的表。标量函数将采用以空格定界的列表形式返回值。

KEGG 用户定义的函数是与非关系型包装器的生命科学用户定义的函数组件一起安装的。在安装 KEGG 用户定义的函数之后，必须注册这些函数。

为了避免与名称空间发生冲突，所有 KEGG 用户定义的函数都是采用 DB2LS 模式注册的。

#### 相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』
- 第 474 页的『禁用 KEGG 用户定义的函数』

#### 相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

---

### 按函数类别排列的 KEGG 用户定义的函数

DB2 Information Integrator 包括一些 KEGG 用户定义的函数，用来存取“途径”数据库和“序列相似性数据库” (SSDB) 中的数据。

下表列示了可以用来检索“途径”数据库中的数据的用户定义的函数。

表 124. “途径” 用户定义的函数

描述	函数类型	函数名
按途径分类的化合物	标量	CompoundsByPathwyS
	表	CompoundsByPathwyT
按途径分类的酶	标量	EnzymesByPathwyS
	表	EnzymesByPathwyT
按途径分类的基因	标量	GenesByPathwyS
	表	GenesByPathwyT
按化合物分类的途径	标量	PathwysByCompndsS
	表	PathwysByCompndsT
按酶分类的途径	标量	PathwysByEnzymesS
	表	PathwysByEnzymesT
按基因分类的途径	标量	PathwysByGenesS
	表	PathwysByGenesT

下表列示了可以用来检索 SSDB 数据库中的数据的用户定义的函数。

表 125. SSDB 用户定义的函数

描述	函数类型	函数名
按基因分类的所有相邻关系	标量	AllNbrsByGeneS
	表	AllNbrsByGeneT
按基因分类的最佳相邻关系	标量	BestNbrsByGeneS
	表	BestNbrsByGeneT
按基因分类的最佳 - 最佳相邻关系	标量	BstBstNbrsByGeneS
	表	BstBstNbrsByGeneT
按基因分类的反向 - 最佳相邻关系	标量	RevBestNbrsByGeneS
	表	RevBestNbrsByGeneT
按基因分类的同源染色体	标量	BestHmlgsByGenesS
	表	BestHmlgsByGenesT
按基因分类的最佳 - 最佳同源染色体	标量	BstBstHmlgByGenesS
	表	BstBstHmlgByGenesT
按基因分类的横向同源物	标量	ParalogsByGeneS
	表	ParalogsByGeneT
按基因分类的定义	标量	DefinitionsByGeneS
按图谱分类的基因	表	GenesByMotifsT
基因之间的 Smith-Waterman 分数	标量	ScoreBetweenGenesS

除了 GetGenesByMotifsT 函数之外，SSDB 数据库的所有表用户定义的函数都将返回一组固定的输出列。GetGenesByMotifsT 函数将返回 *keggid* VARCHAR(100) 和每个基因的 *definition* VARCHAR(1000)。

#### 相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列（表函数）』

#### 相关任务:



## KEGG 用户定义的函数的函数自变量

KEGG 用户定义的函数使用一组常用自变量。下表描述了 KEGG 用户定义的函数的自变量。

### **cpdlist**

化合物的列表。该列表必须具有 VARCHAR 数据类型，并且实际长度不超过 1630 个字节。该列表中每个条目的格式必须为：

`cpd:compound`

`compound` 是化合物标识。该列表中的每个条目必须以空格、逗号或分号定界。

### **enzymelist**

酶的列表。该列表必须具有 VARCHAR 数据类型，并且实际长度不超过 1630 个字节。该列表中的每个条目必须以空格、逗号或分号定界。

### **keggid**

每个生物体的唯一 KEGG 标识，以字符串表示。在 KEGG API 中，此标识称为 `gene_ID`。

该标识必须具有 VARCHAR 数据类型，并且实际长度不超过 100 个字节。`keggid` 自变量的格式为 `keggorg:gene_name`。`keggorg` 是 3 个字母的 KEGG 生物体代码，是以字符串表示的。`gene_name` 是基因的名称。

为 KEGG 数据库中的每个生物体都指定了一个标识。生物体的列表将频繁地发生更改。检查基因组的当前列表（网址为：

<http://www.genome.ad.jp/kegg/kegg2.html#genes>）以获得正确的标识。

### **keggidlist**

KEGG 标识的列表。该列表是定界值的一个字符串。该列表必须具有 VARCHAR 数据类型，并且实际长度不超过 1630 个字节。该列表中的每个条目必须以空格、逗号或分号定界。

### **orglist**

KEGG 生物体名称的列表。该列表是定界值的一个字符串。该列表必须具有 VARCHAR 数据类型，并且实际长度不超过 1630 个字节。该列表中的每个条目必须以空格、逗号或分号定界。

如果不指定此参数，则将搜索整个生物体列表。

### **midlist**

图谱标识的列表。`midlist` 是定界值的一个字符串。该列表必须具有 VARCHAR 数据类型，并且实际长度不超过 1630 个字节。`midlist` 中的每个图谱标识必须以空格、逗号或分号定界。

每个图谱标识的格式为 `database:entry`。`database` 是图谱数据库的标识，`entry` 是该数据库中的一个图谱条目。例如，`pfam` 数据库中的 `DnaJ` 条目的图谱标识为 `pf:DnaJ`。

### **pathwayid**

这是一个由三部分组成的名称，用来标识特定途径。该名称是一个字符串。该名称必须具有 VARCHAR 数据类型，并且实际长度不超过 100 个字节。

`pathwayid` 的格式必须为：

`path:org map`

`org` 是 KEGG 生物体标识, `map` 是途径图标识。

#### **threshold**

Smith-Waterman 分数的值。该值必须大于或等于 100。该值的数据类型必须为 INTEGER。

#### 相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

#### 相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

#### 相关参考:

- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

---

## 注册 KEGG 用户定义的函数

在可以使用 KEGG 用户定义的函数之前, 必须注册这些函数。

`enable_KEGGFunctions` 命令将在联合数据库中注册所有 KEGG 用户定义的函数。这些函数是使用模式名 `DB2LS` 来注册的。

#### 先决条件:

非关系型包装器的生命科学用户定义的函数组件必须随 `DB2 Information Integrator` 一起安装。

#### 过程:

要注册 KEGG 用户定义的函数, 运行 `enable_KEGGFunctions` 命令。

- 在运行 Windows 的联合服务器上, 此命令位于 `sqllib\bin` 目录中
- 在运行 UNIX 的联合服务器上, 此命令位于 `sqllib/bin` 目录中

#### 语法:

```
enable_KEGGFunctions -n dbName -u userID -p password [-force]
```

#### **dbName**

正在其中注册函数的联合数据库的名称。

#### **userID**

联合数据库的有效用户标识。

#### **password**

用户标识的有效密码。

**force** 这是一个可以用来除去函数并重新注册它们的可选标志。如果函数被意外毁坏或删除了, 则使用此标志。

#### 注册用户定义的函数的示例:

以下示例显示发出 `enable_KEGGFunctions` 命令时返回的输出:

```
C:> enable_KEGGFunctions -n federateddb -u db2admin -p db2admin
(0) KEGG Functions were found
-- Create KEGG Functions ...
Create KEGG Functions Successfully.

*** Please allow a few seconds to clean up the system .....
```

使用 **force** 标志来删除和注册用户定义的函数的示例:

以下示例显示在已经注册了用户定义的函数的情况下发出带有 **force** 标志的 `enable_KEGGFunctions` 命令时所返回的输出:

```
C:> enable_KEGGFunctions -n federateddb -u db2admin -p db2admin -force

(37) KEGG Functions were found

KEGG functions already exist ...
Reinstall KEGG functions ...
-- Drop KEGG Functions ...
Drop KEGG Functions Successfully.
-- Create KEGG Functions ...
Create KEGG Functions Successfully.

*** Please allow a few seconds to clean up the system .....
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 『向您的 DB2 Information Integrator 系统添加关系型包装器、非关系型包装器和用户定义的函数』（《IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版》）
- 第 474 页的『禁用 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

---

## 途径数据库函数

下列各节描述“途径”数据库的用户定义的函数。这些节中包含每种函数的语法和示例。

### GenesByPathwyS 用户定义的函数

使用 `GenesByPathwyS` 函数来搜索途径上的所有基因。

`GenesByPathwyS` 函数是一个标量函数，它将返回以空格定界的基因列表。该列表是具有 `VARCHAR` 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 `SELECT` 语句中使用此函数。

语法:

►►—DB2LS.GenesByPathwys—(pathwayid)——►►

此用户定义的函数的模式名为 DB2LS。

搜索途径上的所有基因的示例:

要搜索一个途径上的所有基因, 应在该函数中指定 *pathwayid* 自变量。

例如, 假定您想搜索途径图 00020 上的所有大肠杆菌基因。

使用的子句应为:

```
VALUES CAST(DB2LS.GenesByPathwys
            ('path:eco00020')
            AS VARCHAR(1000));
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## GenesByPathwyT 用户定义的函数

使用 GenesByPathwyT 函数来搜索途径上的所有基因。

GenesByPathwyT 函数是一个表函数, 它将返回具有途径上的基因的名称的 VARCHAR(100) 列。返回的列的名称为 *gene*。

可以在 SELECT 语句中使用此函数。

语法:

►►—DB2LS.GenesByPathwyT—(pathwayid)——►►

此用户定义的函数的模式名为 DB2LS。

搜索途径上的所有基因的示例:

要搜索特定途径上的所有基因, 应在该函数中指定 *pathwayid* 自变量。

例如, 假定您想搜索途径图 00020 上的所有大肠杆菌基因。

SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.GenesByPathwyT
                      ('path:eco00020')) AS t;
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

#### 相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## CompoundsByPathwyS 用户定义的函数

使用 CompoundsByPathwyS 函数来搜索途径上的所有化合物。

CompoundsByPathwyS 函数是一个标量函数，它将返回以空格定界的化合物列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

#### 语法:

```
►—DB2LS.CompoundsByPathwyS—(pathwyid)—►
```

此用户定义的函数的模式名为 DB2LS。

#### 搜索途径上的所有化合物的示例:

要搜索一个途径上的所有化合物，应在该函数中指定 *pathwyid* 自变量。

例如，假定您想搜索途径图 00020 上的所有化合物。

使用的子句应为:

```
VALUES CAST(DB2LS.CompoundsByPathwyS  
           ('path:00020')  
           AS VARCHAR(1000));
```

#### 相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

#### 相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

#### 相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## CompoundsByPathwyT 用户定义的函数

使用 CompoundsByPathwyT 函数来搜索途径上的所有化合物。

CompoundsByPathwyT 函数是一个表函数，它将返回具有途径上的化合物的名称的 VARCHAR(100) 列。返回的列的名称为 *compound*。

可以在 SELECT 语句中使用此函数。

#### 语法:

►—DB2LS.CompoundsByPathwyT—(*pathwayid*)—◄

此用户定义的函数的模式名为 DB2LS。

搜索途径上的所有化合物的示例:

要搜索一个途径上的所有化合物, 应在该函数中指定 *pathwayid* 自变量。

例如, 假定您想搜索途径图 00020 上的所有化合物。

SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.CompoundsByPathwyT
    ('path:00020')) AS t;
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## EnzymesByPathwyS 用户定义的函数

使用 EnzymesByPathwyS 函数来搜索途径上的所有酶。

EnzymesByPathwyS 函数是一个标量函数, 它将返回以空格定界的酶列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

语法:

►—DB2LS.EnzymesByPathwyS—(*pathwayid*)—◄

此用户定义的函数的模式名为 DB2LS。

搜索途径上的所有酶的示例:

要搜索一个途径上的所有酶, 应在该函数中指定 *pathwayid* 自变量。

例如, 假定您想搜索途径图 00020 上的所有酶。

使用的子句应为:

```
VALUES CAST(DB2LS.EnzymesByPathwyS
    ('path:00020')
    AS VARCHAR(1000));
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

| 相关参考:

- | • 第 449 页的『KEGG 用户定义的函数的函数自变量』  
| • 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

| **EnzymesByPathwyT 用户定义的函数**

| 使用 EnzymesByPathwyT 函数来搜索途径上的所有酶。

| EnzymesByPathwyT 函数是一个表函数，它将返回具有途径上的酶的名称的  
| VARCHAR(100) 列。返回的列的名称为 enzyme。

| 可以在 SELECT 语句中使用此函数。

| 语法:

| ►►—DB2LS.EnzymesByPathwyT—(*pathwyid*)—◄◄

| 此用户定义的函数的模式名为 DB2LS。

| 搜索途径上的所有酶的示例:

| 要搜索一个途径上的所有酶，应在该函数中指定 *pathwyid* 自变量。

| 例如，假定您想搜索途径图 00020 上的所有化合物。

| SELECT 语句应为:

| SELECT \* FROM TABLE(DB2LS.EnzymesByPathwyT  
| ('*path:00020*')) AS t;

| 相关概念:

- | • 第 447 页的『KEGG 用户定义的函数 - 概述』

| 相关任务:

- | • 第 450 页的『注册 KEGG 用户定义的函数』

| 相关参考:

- | • 第 449 页的『KEGG 用户定义的函数的函数自变量』  
| • 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

| **PathwysByGenesS 用户定义的函数**

| 使用 PathwysByGenesS 函数来搜索包含您指定的基因的所有途径。

| PathwysByGenesS 函数是一个标量函数，它将返回以空格定界的途径列表。该列表是具  
| 有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

| 可以在 SELECT 语句中使用此函数。

| 语法:

| ►►—DB2LS.PathwysByGenesS—(*keggidlist*)—◄◄

| 此用户定义的函数的模式名为 DB2LS。

搜索包含特定基因的所有途径的示例:

要搜索包含特定基因的所有途径, 应在该函数中指定 *keggidlist* 自变量。

例如, 假定您想搜索包含大肠杆菌基因 b0077 和 b0078 的所有途径。

使用的子句应为:

```
VALUES CAST(DB2LS.PathwysByGenesS
            ('eco:b0077 eco:0078')
            AS VARCHAR(1000));
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## PathwysByGenesT 用户定义的函数

使用 PathwysByGenesT 函数来搜索包含您指定的基因的所有途径。

PathwysByGenesT 函数是一个表函数, 它将返回具有包含基因的途径的名称的 VARCHAR(100) 列。返回的列的名称为 *pathway*。

可以在 SELECT 语句中使用此函数。

语法:

```
►►—DB2LS.PathwysByGenesT—(keggidlist)—————▶▶
```

此用户定义的函数的模式名为 DB2LS。

搜索包含特定基因的所有途径的示例:

要搜索包含特定基因的所有途径, 应在该函数中指定 *keggidlist* 自变量。

例如, 假定您想搜索包含大肠杆菌基因 b0077 和 b0078 的所有途径。

SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.PathwysByGenesT
                      ('eco:b0077 eco:0078')) AS t;
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』



- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## PathwysByCompndsS 用户定义的函数

使用 PathwysByCompndsS 函数来搜索包含您指定的所有化合物的所有途径。

PathwysByCompndsS 函数是一个标量函数，它将返回以空格定界的化合物列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

语法:

►►—DB2LS.PathwysByCompndsS—(*cpdlist*)—►►

此用户定义的函数的模式名为 DB2LS。

搜索包含指定的所有化合物的途径的示例:

要搜索包含列表中的所有化合物的所有途径，应在该函数中指定 *cpdlist* 自变量。

例如，假定您想搜索包含化合物 C00033 和 C00158 的所有途径。

使用的子句应为:

```
VALUES CAST(DB2LS.PathwysByCompndsS
            ('cpd:C00033 cpd:C00158')
            AS VARCHAR(1000));
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## PathwysByCompndsT 用户定义的函数

使用 PathwysByCompndsT 函数来搜索包含您指定的所有化合物的所有途径。

PathwysByCompndsT 函数是一个表函数，它将返回具有途径的名称的 VARCHAR(100) 列。返回的列的名称为 *pathway*。

可以在 SELECT 语句中使用此函数。

语法:

►►—DB2LS.PathwysByCompndsT—(*cpdlist*)—►►

此用户定义的函数的模式名为 DB2LS。

搜索包含指定的所有化合物的途径的示例:

要搜索包含列表中的所有化合物的所有途径，应在该函数中指定 *cpdlist* 自变量。

例如，假定您想搜索包含化合物 C00033 和 C00158 的所有途径。

SELECT 语句应为：

```
SELECT * FROM TABLE(DB2LS.PathwysByCompndsT
('cpd:C00033 cpd:C00158')) AS t;
```

相关概念：

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务：

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考：

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## PathwysByEnzymesS 用户定义的函数

使用 PathwysByEnzymesS 函数来搜索包含您指定的所有酶的所有途径。

PathwysByEnzymesS 函数是一个标量函数，它将返回以空格定界的途径列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

语法：

```
▶▶—DB2LS.PathwysByEnzymesS—(enzymelist)—————▶▶
```

此用户定义的函数的模式名为 DB2LS。

搜索包含指定的所有酶的所有途径的示例：

要搜索包含列表中的所有酶的所有途径，应在该函数中指定 *enzymelist* 自变量。

例如，假定您想搜索包含酶 1.3.99.1 的所有途径。

使用的子句应为：

```
VALUES CAST(DB2LS.PathwysByEnzymesS
('ec:1.3.99.1')
AS VARCHAR(1000));
```

相关概念：

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务：

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考：

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## PathwysByEnzymesT 用户定义的函数

使用 PathwysByEnzymesT 函数来搜索包含您指定的所有酶的所有途径。

PathwysByEnzymesT 函数是一个表函数，它将返回具有途径的名称的 VARCHAR(100) 列。返回的列的名称为 pathway。

可以在 SELECT 语句中使用此函数。

语法:

```
►►—DB2LS.PathwysByEnzymesT—(enzymelist)——►►
```

此用户定义的函数的模式名为 DB2LS。

搜索包含指定的所有酶的所有途径的示例:

要搜索包含列表中的所有酶的所有途径，应在该函数中指定 *enzymelist* 自变量。

例如，假定您想搜索包含酶 1.3.99.1 的所有途径。

SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.PathwysByEnzymesT
    ('ec:1.3.99.1')) AS t;
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

---

## 序列相似性数据库函数

下列主题描述“序列相似性数据库”（SSDB）的用户定义的函数。这些节中包含每种函数的语法和示例。

### 从 SSDB 数据库查询返回的列（表函数）

“序列相似性数据库”（SSDB）的许多用户定义的表函数都将返回一组固定输出列。当使用 SSDB 表函数时，可以指定只返回一部分列。下表列示了返回的列的名称和数据类型。

表 126. SSDB 数据库中的列

列名	列数据类型	描述
keggid1	VARCHAR (100)	在查询中指定的基因的标识。
keggid2	VARCHAR (100)	从查询中返回的基因的标识。
swscore	DOUBLE	keggid1 与 keggid2 之间的 Smith-Waterman 分数。

表 126. SSDB 数据库中的列 (续)

列名	列数据类型	描述
identity	DOUBLE	keggid1 与 keggid2 之间的全等百分比。
overlap	INTEGER	keggid1 与 keggid2 之间的重叠长度。
s1_start	INTEGER	keggid1 中的比对的起始位置。
s1_end	INTEGER	keggid1 中的比对的结束位置。
s2_start	INTEGER	keggid2 中的比对的起始位置。
s2_end	INTEGER	keggid2 中的比对的结束位置。
best1	INTEGER	用来指示从 keggid1 到 keggid2 的最佳符合项的标志。如果值为 1, 则指示 keggid1 与 keggid2 之间存在最佳符合项关系。如果值为 0, 则指示 keggid1 与 keggid2 之间没有最佳符合项关系。
best2	INTEGER	用来指示从 keggid2 到 keggid1 的最佳符合项的标志。如果值为 1, 则指示 keggid2 与 keggid1 之间存在最佳符合项关系。如果值为 0, 则指示 keggid2 与 keggid1 之间没有最佳符合项关系。
def1	VARCHAR (1000)	keggid1 的定义。
def2	VARCHAR (1000)	keggid2 的定义。
length1	INTEGER	keggid1 中的氨基酸的长度。
length2	INTEGER	keggid2 中的氨基酸的长度。

#### 相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

#### 相关参考:

- 第 452 页的『GenesByPathwyT 用户定义的函数』
- 第 461 页的『AllNbrsByGeneT 用户定义的函数』
- 第 463 页的『BstBstNbrsByGeneT 用户定义的函数』
- 第 465 页的『BestNbrsByGeneT 用户定义的函数』
- 第 467 页的『RevBestNbrsByGeneT 用户定义的函数』
- 第 469 页的『ParalogsByGeneT 用户定义的函数』
- 第 470 页的『BestHmlgsByGenesT 用户定义的函数』
- 第 471 页的『BstBstHmlgByGenesT 用户定义的函数』

## AllNbrsByGeneS 用户定义的函数

使用 AllNbrsByGeneS 函数来搜索是您指定的 KEGG 标识的同源相邻关系的所有生物体。

无须搜索所有生物体, 可以通过指定生物体的列表来缩小搜索范围。

AllNbrsByGeneS 函数是一个标量函数, 它将返回以空格定界的目标标识列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

语法:

▶▶ DB2LS.AllNbrsByGeneS—(*keggid*, *threshold*—, *orglist*)▶▶

此用户定义的函数的模式名为 DB2LS。

搜索整个生物体列表的示例:

要搜索整个生物体列表, 应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如, 要搜索 *threshold* 值大于 200 的大肠杆菌基因的所有同源基因, 使用的子句应为:

```
VALUES CAST(DB2LS.AllNbrsByGeneS
            ('eco:b0002', 200)
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例:

要指定要搜索的生物体的列表, 必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如, 要搜索 *threshold* 值大于 500 的所有同源基因, 并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株, 使用的子句应为:

```
VALUES CAST(DB2LS.AllNbrsByGeneS
            ('eco:b0002', 500, 'ece hin')
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。大肠杆菌基因的 0157 菌株的 KEGG 名称为 *ece*。流行性感胃杆菌基因的 KEGG 名称为 *hin*。

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## AllNbrsByGeneT 用户定义的函数

使用 AllNbrsByGeneT 函数来搜索是您指定的 KEGG 标识的同源相邻关系的所有生物体。

无须搜索所有生物体, 可以通过指定生物体的列表来缩小搜索范围。

AllNbrsByGeneT 函数是一个表函数, 它将返回固定的输出列集合。

可以在 SELECT 语句中使用此函数。

语法:

►►DB2LS.A11NbrsByGeneT(*keggid*, *threshold* [ ,*orglist* ] )◀◀

此用户定义的函数的模式名为 DB2LS。

搜索整个生物体列表的示例:

要搜索整个生物体列表, 应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如, 要搜索 *threshold* 值大于 200 的大肠杆菌基因的所有同源基因, SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.A11NbrsByGeneT
  ('eco:b0002', 200)) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例:

要指定要搜索的生物体的列表, 必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如, 要搜索 *threshold* 值大于 500 的所有同源基因, 并且只返回大肠杆菌基因的 0157 菌株和流行性感冒杆菌基因的所有菌株, SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.A11NbrsByGeneT
  ('eco:b0002', 500, 'ece hin')) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。大肠杆菌基因的 0157 菌株的 KEGG 名称为 *ece*。流行性感冒杆菌基因的 KEGG 名称为 *hin*。

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列(表函数)』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』

## BstBstNbrsByGeneS 用户定义的函数

使用 `BstBstNbrsByGeneS` 函数来搜索每个生物体中的基因的最佳 - 最佳相邻关系。

`BstBstNbrsByGeneS` 函数是一个标量函数, 它将返回以空格定界的目标标识列表。该列表是具有 `VARCHAR` 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 `SELECT` 语句中使用此函数。

语法:

►►DB2LS.BstBstNbrsByGeneS(*keggid*, *threshold* [ ,*orglist* ] )◀◀

此用户定义的函数的模式名为 DB2LS。

搜索整个生物体列表的示例:

要搜索整个生物体列表, 应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如, 要搜索 *threshold* 值大于 200 的大肠杆菌基因的最佳 - 最佳相邻关系, 使用的子句应为:

```
VALUES CAST(DB2LS.BstBstNbrsByGeneS  
            ('eco:b0002', 200)  
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例:

要指定要搜索的生物体的列表, 必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如, 要搜索 *threshold* 值大于 500 的最佳 - 最佳相邻关系, 并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株, 使用的子句应为:

```
VALUES CAST(DB2LS.BstBstNbrsByGeneS  
            ('eco:b0002', 500, 'ece hin')  
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。大肠杆菌基因的 0157 菌株的 KEGG 名称为 *ece*。流行性感胃杆菌基因的 KEGG 名称为 *hin*。

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BstBstNbrsByGeneT 用户定义的函数

使用 *BstBstNbrsByGeneT* 函数来搜索每个生物体中的基因的最佳 - 最佳相邻关系。

*BstBstNbrsByGeneT* 函数是一个表函数, 它将返回固定的输出列集合。

可以在 *SELECT* 语句中使用此函数。

语法:

```
►► DB2LS.BstBstNbrsByGeneT(keggid, threshold [,orglist])
```

此用户定义的函数的模式名为 DB2LS。

搜索整个生物体列表的示例:

要搜索整个生物体列表，应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如，要搜索 *threshold* 值大于 200 的大肠杆菌基因的最佳 - 最佳相邻关系，SELECT 语句应为：

```
SELECT * FROM TABLE(DB2LS.BstBstNbrsByGeneT
    ('eco:b0002', 200)) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例：

要指定要搜索的生物体的列表，必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如，要搜索 *threshold* 值大于 500 的最佳 - 最佳相邻关系，并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株，SELECT 语句应为：

```
SELECT * FROM TABLE(DB2LS.BstBstNbrsByGeneT
    ('eco:b0002', 500, 'ece hin')) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。大肠杆菌基因的 0157 菌株的 KEGG 名称为 *ece*。流行性感胃杆菌基因的 KEGG 名称为 *hin*。

相关概念：

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列（表函数）』

相关任务：

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考：

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BestNbrsByGeneS 用户定义的函数

使用 *BestNbrsByGeneS* 函数来搜索每个生物体中的基因的最佳相邻关系。

*BestNbrsByGeneS* 函数是一个标量函数，它将返回以空格定界的目标标识列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

语法：

```
►► DB2LS.BestNbrsByGeneS(keggid, threshold [ , orglist ] )
```

此用户定义的函数的模式名为 DB2LS。

搜索整个生物体列表的示例：

要搜索整个生物体列表，应在该函数中指定 *keggid* 和 *threshold* 自变量。



例如，要搜索 `threshold` 值大于 200 的大肠杆菌基因的最佳相邻关系，使用的子句应为：

```
VALUES CAST(DB2LS.BestNbrsByGeneS
            ('eco:b0002', 200)
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 `eco`。基因名称为 `b0002`。

搜索特定的一组生物体的示例：

要指定要搜索的生物体的列表，必须在该函数中指定 `keggid`、`threshold` 和 `orglist` 自变量。

例如，要搜索 `threshold` 值大于 500 的最佳相邻关系，并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株，使用的子句应为：

```
VALUES CAST(DB2LS.BestNbrsByGeneS
            ('eco:b0002', 500, 'ece hin')
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 `eco`。基因名称为 `b0002`。大肠杆菌基因的 0157 菌株的 KEGG 名称为 `ece`。流行性感胃杆菌基因的 KEGG 名称为 `hin`。

相关概念：

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务：

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考：

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BestNbrsByGeneT 用户定义的函数

使用 `BestNbrsByGeneT` 函数来搜索每个生物体中的基因的最佳相邻关系。

`BestNbrsByGeneT` 函数是一个表函数，它将返回固定的输出列集合。

可以在 `SELECT` 语句中使用此函数。

语法：

```
►► DB2LS.BestNbrsByGeneT(keggid, threshold [ , orglist ])
```

此用户定义的函数的模式名为 `DB2LS`。

搜索整个生物体列表的示例：

要搜索整个生物体列表，应在该函数中指定 `keggid` 和 `threshold` 自变量。

例如，要搜索 `threshold` 值大于 200 的大肠杆菌基因的最佳相邻关系，`SELECT` 语句应为：

```
SELECT * FROM TABLE(DB2LS.BestNbrsByGeneT
                     ('eco:b0002', 200)) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例:

要指定要搜索的生物体的列表, 必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如, 要搜索 *threshold* 值大于 500 的最佳相邻关系, 并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株, `SELECT` 语句应为:

```
SELECT * FROM TABLE(DB2LS.BestNbrsByGeneT
    ('eco:b0002', 500, 'ece hin')) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。大肠杆菌基因的 0157 菌株的 KEGG 名称为 *ece*。流行性感胃杆菌基因的 KEGG 名称为 *hin*。

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列 (表函数)』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## RevBestNbrsByGeneS 用户定义的函数

使用 `RevBestNbrsByGeneS` 函数来搜索每个生物体中的基因的反向 - 最佳相邻关系。

`RevBestNbrsByGeneS` 函数是一个标量函数, 它将返回以空格定界的目标标识列表。该列表是具有 `VARCHAR` 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 `SELECT` 语句中使用此函数。

语法:

```
DB2LS.RevBestNbrsByGeneS(keggid, threshold, [orglist])
```

此用户定义的函数的模式名为 `DB2LS`。

搜索整个生物体列表的示例:

要搜索整个生物体列表, 应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如, 要搜索 *threshold* 值大于 200 的大肠杆菌基因的反向 - 最佳相邻关系, 使用的子句应为:

```
VALUES CAST(DB2LS.RevBestNbrsByGeneS
    ('eco:b0002', 200)
    AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例:

要指定要搜索的生物体的列表, 必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如, 要搜索 *threshold* 值大于 500 的反向 - 最佳相邻关系, 并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株, 使用的子句应为:

```
VALUES CAST(DB2LS.RevBestNbrsByGeneS  
            ('eco:b0002', 500, 'ece hin')  
            AS VARCHAR(1000));
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。大肠杆菌基因的 0157 菌株的 KEGG 名称为 *ece*。流行性感胃杆菌基因的 KEGG 名称为 *hin*。

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## RevBestNbrsByGeneT 用户定义的函数

使用 *RevBestNbrsByGeneT* 函数来搜索每个生物体中的基因的反向 - 最佳相邻关系。

*RevBestNbrsByGeneT* 函数是一个表函数, 它将返回固定的输出列集合。

可以在 *SELECT* 语句中使用此函数。

语法:

```
►► DB2LS.RevBestNbrsByGeneT(keggid, threshold  
                             [, orglist])
```

此用户定义的函数的模式名为 *DB2LS*。

搜索整个生物体列表的示例:

要搜索整个生物体列表, 应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如, 要搜索 *threshold* 值大于 200 的大肠杆菌基因的反向 - 最佳相邻关系, *SELECT* 语句应为:

```
SELECT * FROM TABLE(DB2LS.RevBestNbrsByGeneT  
                      ('eco:b0002', 200)) AS t;
```

大肠杆菌基因的 KEGG 代码为 *eco*。基因名称为 *b0002*。

搜索特定的一组生物体的示例:

要指定要搜索的生物体的列表, 必须在该函数中指定 *keggid*、*threshold* 和 *orglist* 自变量。

例如，要搜索 `threshold` 值大于 500 的反向 - 最佳相邻关系，并且只返回大肠杆菌基因的 0157 菌株和流行性感胃杆菌基因的所有菌株，`SELECT` 语句应为：

```
SELECT * FROM TABLE(DB2LS.RevBestNbrsByGeneT
    ('eco:b0002', 500, 'ece hin')) AS t;
```

大肠杆菌基因的 KEGG 代码为 `eco`。基因名称为 `b0002`。大肠杆菌基因的 0157 菌株的 KEGG 名称为 `ece`。流行性感胃杆菌基因的 KEGG 名称为 `hin`。

**相关概念：**

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列（表函数）』

**相关任务：**

- 第 450 页的『注册 KEGG 用户定义的函数』

**相关参考：**

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## ParalogsByGeneS 用户定义的函数

使用 `ParalogsByGeneS` 函数来搜索生物体中的横向同源基因。

`ParalogsByGeneS` 函数是一个标量函数，它将返回以空格定界的目标标识列表。该列表是具有 `VARCHAR` 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 `SELECT` 语句中使用此函数。

**语法：**

```
►►—DB2LS.ParalogsByGeneS—(keggid, threshold)—————►►
```

此用户定义的函数的模式名为 `DB2LS`。

**搜索横向同源基因的示例：**

要搜索一个生物体中的横向同源基因，在该函数中仅指定 `keggid` 和 `threshold` 自变量。

例如，要搜索 `threshold` 值大于 5000 的大肠杆菌生物体的横向同源基因，使用的子句应为：

```
VALUES CAST(DB2LS.ParalogsByGeneS
    ('eco:b0002', 5000)
    AS VARCHAR(1000));
```

大肠杆菌生物体的 KEGG 数据库名称为 `eco`。生物体名称为 `b0002`。

**相关概念：**

- 第 447 页的『KEGG 用户定义的函数 - 概述』

**相关任务：**

- 第 450 页的『注册 KEGG 用户定义的函数』

**相关参考：**

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## ParalogsByGeneT 用户定义的函数

使用 ParalogsByGeneT 函数来搜索生物体中的横向同源基因。

ParalogsByGeneT 函数是一个表函数，它将返回固定的输出列集合。

可以在 SELECT 语句中使用此函数。

语法:

```
►►—DB2LS.ParalogsByGeneT—(keggid, threshold)—————►►
```

此用户定义的函数的模式名为 DB2LS。

搜索横向同源基因的示例:

要搜索一个生物体中的横向同源基因，应在该函数中指定 *keggid* 和 *threshold* 自变量。

例如，要搜索 *threshold* 值大于 5000 的大肠杆菌生物体的横向同源基因，SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.ParalogsByGeneT
    ('eco:b0002', 5000)) AS t;
```

大肠杆菌生物体的 KEGG 数据库名称为 *eco*。生物体名称为 *b0002*。

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列（表函数）』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BestHmlgsByGenesS 用户定义的函数

使用 BestHmlgsByGenesS 函数来从您指定的基因列表中搜索某个生物体的最佳同源相邻关系。

BestHmlgsByGenesS 函数是一个标量函数，它将返回以空格定界的目标标识列表。该列表是具有 VARCHAR 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 SELECT 语句中使用此函数。

语法:

►►—DB2LS.BestHmlgsByGenesS—(keggorg—, keggidlist)—————►►

此用户定义的函数的模式名为 DB2LS。

为了获得更好的性能，可使用 BstBstHmlgByGenesS 函数。

**搜索最佳同源相邻关系的示例:**

要从基因列表中搜索某个生物体的最佳同源相邻关系，应在该函数中指定 *keggorg* 和 *keggidlist* 自变量。

例如，要在大肠杆菌基因 b0002、b0003、b0004 和 b0005 的列表中搜索流行性感冒杆菌生物体，使用的子句应为:

```
VALUES CAST(DB2LS.BestHmlgsByGenesS
             ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')
             VARCHAR(1000));
```

**相关概念:**

- 第 447 页的『KEGG 用户定义的函数 - 概述』

**相关任务:**

- 第 450 页的『注册 KEGG 用户定义的函数』

**相关参考:**

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BestHmlgsByGenesT 用户定义的函数

使用 BestHmlgsByGenesT 函数来从您指定的基因列表中搜索某个生物体的最佳同源相邻关系。

BestHmlgsByGenesT 函数是一个表函数，它将返回固定的输出列集合。

可以在 SELECT 语句中使用此函数。

**语法:**

►►—DB2LS.BestHmlgsByGenesT—(keggorg, keggidlist)—————►►

此用户定义的函数的模式名为 DB2LS。

为了获得更好的性能，可使用 BstBstHmlgByGenesT 函数。

**搜索最佳同源相邻关系的示例:**

要从基因列表中搜索某个生物体的最佳同源相邻关系，应在该函数中指定 *keggorg* 和 *keggidlist* 自变量。

例如，要在大肠杆菌基因 b0002、b0003、b0004 和 b0005 的列表中搜索流行性感冒杆菌生物体，SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.BestHmlgsByGenesT
                     ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')) AS t;
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列 (表函数)』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BstBstHmlgByGenesS 用户定义的函数

使用 `BstBstHmlgByGenesS` 函数来从基因列表中搜索某个生物体的最佳 - 最佳同源相邻关系。

`BstBstHmlgByGenesS` 函数是一个标量函数，它将返回以空格定界的目标标识列表。该列表是具有 `VARCHAR` 数据类型并且实际长度不超过 32767 个字节的字符串。

可以在 `SELECT` 语句中使用此函数。

语法:

►►—`DB2LS.BstBstHmlgByGenesS`—(*keggorg*, *keggidlist*)—►►

此用户定义的函数的模式名为 `DB2LS`。

只搜索最佳 - 最佳相邻关系的示例:

要从基因列表中只搜索某个生物体的最佳 - 最佳相邻关系，应在该函数中指定 `keggorg` 和 `keggidlist` 自变量。

例如，要在大肠杆菌基因 `b0002`、`b0003`、`b0004` 和 `b0005` 的列表中搜索流行性感冒杆菌生物体，使用的子句应为:

```
VALUES CAST(DB2LS.BstBstHmlgByGenesS
             ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')
             AS VARCHAR(1000));
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## BstBstHmlgByGenesT 用户定义的函数

使用 `BstBstHmlgByGenesT` 函数来从基因列表中搜索某个生物体的最佳 - 最佳同源相邻关系。

BstBstHmlgByGenesT 函数是一个表函数，它将返回固定的输出列集合。

可以在 SELECT 语句中使用此函数。

语法:

►►—DB2LS.BstBstHmlgByGenesT—(keggorg, keggidlist)——►►

此用户定义的函数的模式名为 DB2LS。

只搜索最佳 - 最佳相邻关系的示例:

要从基因列表中只搜索某个生物体的最佳 - 最佳相邻关系，应在该函数中指定 *keggorg* 和 *keggidlist* 自变量。

例如，要在大肠杆菌基因 b0002、b0003、b0004 和 b0005 的列表中搜索流行性感冒杆菌生物体，SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.BstBstHmlgByGenesT
    ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')) AS t;
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』
- 第 459 页的『从 SSDB 数据库查询返回的列 (表函数)』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## ScoreBetweenGenesS 用户定义的函数

使用 ScoreBetweenGenesS 函数来确定两个基因之间的 Smith-Waterman 分数。

ScoreBetweenGenesS 函数是一个标量函数，它将返回数据类型为 DOUBLE 的单个值。

可以在 SELECT 语句中使用此函数。

语法:

►►—DB2LS.ScoreBetweenGenesS—(keggid1, keggid2)——►►

此用户定义的函数的模式名为 DB2LS。

确定两个基因之间的 **Smith-Waterman** 分数的示例:

要确定两个基因之间的 Smith-Waterman 分数，应指定每个基因的 *keggid*。

例如，要确定大肠杆菌基因 b0002 与 b3940 之间的 Smith-Waterman 分数，使用的子句应为:



```
VALUES CAST(DB2LS.ScoreBetweenGenesS
           ('eco:b0002', 'eco:b3940')
           AS DOUBLE);
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## DefinitionsByGeneS 用户定义的函数

使用 DefinitionsByGeneS 函数来搜索基因的定义。

DefinitionsByGeneS 函数是一个标量函数，它将返回数据类型为 VARCHAR(1000) 的单个值。

可以在 SELECT 语句中使用此函数。

语法:

```
►►—DB2LS.DefinitionsByGeneS—(keggid)—————►
```

此用户定义的函数的模式名为 DB2LS。

搜索基因的定义的示例:

要搜索基因的定义，应指定 *keggid*。

例如，要返回大肠杆菌基因 b0002 的定义，使用的子句应为:

```
VALUES CAST(DB2LS.DefinitionsByGeneS
           ('eco:b0002')) AS t;
           AS VARCHAR(1000));
```

相关概念:

- 第 447 页的『KEGG 用户定义的函数 - 概述』

相关任务:

- 第 450 页的『注册 KEGG 用户定义的函数』

相关参考:

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

## GenesByMotifsT 用户定义的函数

使用 GenesByMotifsT 函数来搜索包含您指定的图谱列表中的所有图谱的基因。

GenesByMotifsT 函数是一个表函数，它将返回一个表。该表中包含每个基因的 *keggid* VARCHAR(100) 和 *definition* VARCHAR(1000)。

可以在 SELECT 语句中使用此函数。

**语法:**

►—DB2LS.GenesByMotifsT—(*midlist*)—►

此用户定义的函数的模式名为 DB2LS。

搜索包含列表中的所有图谱的基因的示例:

要搜索包含列表中的所有图谱的基因，应指定 *midlist* 自变量。

例如，要查找同时包含 *Pfam 'DnaJ'* 和 *Prosite 'DNAJ\_2'* 图谱的基因，SELECT 语句应为:

```
SELECT * FROM TABLE(DB2LS.GenesByMotifsT
('pf:DnaJ ps:DNAJ_2')) AS t;
```

**相关概念:**

- 第 447 页的『KEGG 用户定义的函数 - 概述』

**相关任务:**

- 第 450 页的『注册 KEGG 用户定义的函数』

**相关参考:**

- 第 449 页的『KEGG 用户定义的函数的函数自变量』
- 第 447 页的『按函数类别排列的 KEGG 用户定义的函数』

---

## 禁用 KEGG 用户定义的函数

如果您不再想使用 KEGG 用户定义的函数，则可以临时禁用这些函数或者从联合数据库中永久除去它们。

如果禁用了 KEGG 用户定义的函数，可以通过在联合数据库中注册这些函数来再次启用它们。

必须卸载函数才能从联合数据库中永久除去这些函数。

**过程:**

要禁用 KEGG 用户定义的函数，运行 `disable_KEGGFunctions` 命令。

- 在运行 Windows 的联合服务器上，此命令位于 `sqllib\bin` 目录中
- 在运行 UNIX 的联合服务器上，此命令位于 `sqllib/bin` 目录中

**语法:**

```
disable_KEGGFunctions -n dbName -u userID -p password
```

**dbName**

想要在其中禁用函数的联合数据库的名称。

| **userID**

| 联合数据库的有效用户标识。

| **password**

| 用户标识的有效密码。

| **禁用 KEGG 用户定义的函数的示例:**

| 以下示例显示发出 `disable_KEGGFunctions` 命令时返回的输出:

| `C:>disable_KEGGFunctions -n federateddb -u db2admin -p db2admin`

| (37) KEGG Functions were found  
| -- Drop KEGG Functions ...  
| Drop KEGG Functions Successfully.

| \*\*\* Please allow a few seconds to clean up the system .....

| **相关任务:**

- 『除去关系型包装器、非关系型包装器和生命科学用户定义的函数 (Windows) 』 (《*IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版*》)
- 『除去包装器、用户定义的函数和包装器开发工具箱 (UNIX) 』 (《*IBM DB2 Information Integrator 安装指南 Linux 版、UNIX 版和 Windows 版*》)
- 第 450 页的『注册 KEGG 用户定义的函数』



---

## 第 5 部分 参考



---

## 第 27 章 非关系数据源支持的数据类型

下列各节列示了非关系型包装器支持的数据类型。

---

### 非关系数据源支持的数据类型

对于大多数非关系数据源，当创建昵称以存取数据源时必须指定列信息（包括数据类型）。

某些非关系型包装器会创建存取数据源所需的所有列。这些列被称为**固定列**。其它包装器让您指定 CREATE NICKNAME 语句中的列的部分或所有数据类型。

下面部分列出了可为其指定数据类型的包装器，以及此包装器支持的数据类型。

#### BioRS 包装器支持的数据类型

下表列出了 BioRS 包装器支持的 DB2 数据类型。

表 127. 映射到 DB2 数据类型的 BioRS 数据类型

BioRS 数据类型	DB2 数据类型
AUTHOR	CHARACTER、CLOB 和 VARCHAR
DATE	CHARACTER、CLOB 和 VARCHAR
NUMBER	CHARACTER、CLOB 和 VARCHAR
REFERENCE	CHARACTER、CLOB 和 VARCHAR
TEXT	CHARACTER、CLOB 和 VARCHAR

CLOB 数据类型允许的最大长度为 5 兆字节。

#### BLAST 包装器支持的数据类型

某些数据类型会自动为 BLAST 包装器创建的固定列设置。

对于定义行字段，可以在创建昵称时分配。如果定义行列中的数据跟本地列数据类型不兼容，将会产生错误。例如，如果定义 INTEGER 类型的定义行列并且列中具有非数字的值，则会返回错误。

下表列出了 BLAST 包装器支持的 DB2 数据类型。

表 128. 映射到 DB2 数据类型的 BLAST 数据类型

BLAST 数据类型	DB2 数据类型
定义行	CLOB
	CLOB 数据类型允许的最大长度为 5 兆字节。
定义行	DOUBLE
定义行	FLOAT
定义行	INTEGER

表 128. 映射到 DB2 数据类型的 BLAST 数据类型 (续)

BLAST 数据类型	DB2 数据类型
定义行	VARCHAR

## Documentum 包装器支持的数据类型

下表列出了 Documentum 包装器支持的 DB2 数据类型。

表 129. 映射到 DB2 数据类型的 Documentum 数据类型

Documentum 数据类型	DB2 数据类型
DOUBLE	DOUBLE、FLOAT、INTEGER 和 SMALLINT
标识	CHARACTER (16)
INTEGER	DOUBLE、FLOAT、INTEGER 和 SMALLINT
STRING (最多 255 个字符)	CHAR 和 VARCHAR
TIME	CHAR、DATE、TIMESTAMP 和 VARCHAR

## Entrez 包装器支持的数据类型

下表列出了 Entrez 包装器支持的 DB2 数据类型。

表 130. 映射到 DB2 数据类型的 Entrez 数据类型

Entrez 数据类型	DB2 数据类型
character	CHARACTER
character	CLOB
	CLOB 数据类型允许的最大长度为 5 兆字节。
date	DATE
number	DECIMAL
number	DOUBLE
integer	INTEGER
number	REAL
integer	SMALLINT
时间	TIMESTAMP
character	VARCHAR

## Excel 包装器支持的数据类型

下表列出了 Excel 包装器支持的 DB2 数据类型。

表 131. 映射到 DB2 数据类型的 Excel 数据类型

Excel 数据类型	DB2 数据类型
date	DATE
number	DOUBLE
number	FLOAT (n), 其中 n >= 25 而 <= 53
integer	INTEGER
character	VARCHAR



## Extended Search 包装器支持的数据类型

下表列出了 Extended Search 包装器支持的 DB2 数据类型。

表 132. 映射至 DB2 数据类型的 Extended Search 数据类型

Extended Search 数据类型	DB2 数据类型
Date	DATE
Double	DOUBLE
Integer	INTEGER
String	VARCHAR

## HMMER 包装器支持的数据类型

下表列出了 HMMER 包装器支持的 DB2 数据类型。

表 133. 映射到 DB2 数据类型的 HMMERT 数据类型

HMMER 数据类型	DB2 数据类型
character	CLOB
	CLOB 数据类型允许的最大长度为 5 兆字节。
character	DOUBLE
character	FLOAT
character	INTEGER
character	VARCHAR

## 表结构文件包装器支持的数据类型

下表列出了表结构文件包装器支持的 DB2 数据类型。

表 134. 映射到 DB2 数据类型的表结构文件数据类型

表结构文件数据类型	DB2 数据类型
character	CHARACTER
character	CLOB
	CLOB 数据类型允许的最大长度为 5 兆字节。
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR

## Web 服务包装器支持的数据类型

下表列出了 Web 服务包装器支持的 DB2 数据类型。Web 服务包装器使用 XML 数据类型。

表 135. 映射到 Web 服务包装器的 DB2 数据类型的 XML 数据类型

XML 数据类型	DB2 数据类型
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

## WebSphere Business Integration 包装器支持的数据类型

下表列出了 WebSphere Business Integration 包装器支持的 DB2 数据类型。WebSphere Business Integration 包装器使用 XML 数据类型。

表 136. 映射到 WebSphere Business Integration 包装器的 DB2 数据类型的 XML 数据类型

XML 数据类型	DB2 数据类型
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

## XML 包装器支持的数据类型

下表列出了 XML 包装器支持的 DB2 数据类型

表 137. 映射到 XML 包装器的 DB2 数据类型的 XML 数据类型

XML 数据类型	DB2 数据类型
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB
	CLOB 数据类型允许的最大长度为 5 兆字节。
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

### 相关概念:

- 『联合系统中的数据类型映射』（《联合系统指南》）
- 『数据类型映射和联合数据库全局目录』（《联合系统指南》）
- 『非关系数据源的数据类型映射』（《联合系统指南》）



---

## 第 28 章 改变昵称

本章说明如何改变先前注册的昵称。

---

### 改变昵称

昵称是用于引用要在数据源存取的对象标识。

您可能想要改变昵称以达到以下目的:

- 改变数据源对象的列的本地列名
- 改变数据源对象的列的本地数据类型
- 添加、设置或删除昵称和列选项
- 添加或删除主键
- 添加或删除一个或多个唯一约束、引用约束或检查约束
- 改变一个或多个引用约束、检查约束或功能相关性约束属性

#### 先决条件:

由语句的授权标识所拥有的特权必须至少包括下列其中一个权限:

- SYSADM 或 DBADM 权限
- 对在语句中指定的昵称的 ALTER 特权
- 对在语句中指定的昵称的 CONTROL 特权
- 对模式的 ALTERIN 特权 (如果昵称的模式名存在)
- 在昵称的目录视图 DEFINER 列中记录的昵称的定义者

#### 限制:

请参阅关于对改变昵称的限制的主题。

#### 过程:

可以通过 DB2 控制中心或 DB2 命令行改变昵称。

要通过 DB2 控制中心执行此任务:

1. 选择**昵称**文件夹。
2. 右键单击要更改的昵称, 然后单击**改变**。此时会打开“改变昵称”笔记本。
3. 在“昵称”页上, 更改存储在全局目录中的列的本地列名、本地数据类型或列选项。
4. 在“键”页上, 为昵称设置引用完整性约束。可以设置主键、唯一键或外键约束。
5. 在“检查约束”页上, 为昵称设置检查约束或功能相关性约束。
6. 在“设置”页上, 为昵称设置昵称选项。
7. 单击**确定**以改变昵称并关闭笔记本。

某些昵称选项是必需的且不能删除。如果已设置特定的昵称选项, 则不能添加其它昵称选项。有关对每个选项的一组描述, 请参阅联合系统的昵称选项和联合系统的昵称列选项。

要通过 DB2 命令行执行此任务，请发出设置了适当参数的 ALTER NICKNAME 语句。

当数据源对象结构或内容有显著更改时，应该更新昵称统计信息。显著更改包括添加或删除多个行。

#### 相关概念:

- 『对昵称的参考约束』（《联合系统指南》）
- 『昵称统计信息更新设施 - 概述』（《联合系统指南》）

#### 相关任务:

- 第 489 页的『改变昵称选项』
- 第 491 页的『改变数据源对象的本地类型』
- 第 488 页的『改变昵称列名』
- 第 490 页的『改变昵称列选项』

#### 相关参考:

- 第 486 页的『对改变昵称的限制』
- 第 543 页的附录 F, 『联合系统的昵称选项』
- 第 551 页的附录 G, 『联合系统的昵称列选项』
- 『ALTER NICKNAME statement』（*SQL Reference, Volume 2*）

---

## 对改变昵称的限制

当改变昵称时，应用以下限制。

**列名** ALTER NICKNAME 语句不能用来改变以下数据源的列名。必须删除昵称并用正确的列名再次创建昵称。

- BLAST
- Documentum
- HMMER

**列选项** 如果以下其中一个选项是在列上设置的，则不能对该列添加任何其它选项:

- SOAPACTIONCOLUMN
- URLCOLUMN
- PRIMARY\_KEY
- FOREIGN\_KEY

对于 BioRS

- 如果通过使用 ELEMENT\_NAME 选项来更改某列的元素名称，则不检查新名称以确保其是正确的。不正确的选项可能会导致在查询中引用列时出错。
- 如果对 IS\_INDEXED 列选项进行更改，则不通过 BioRS 服务器对更改进行验证。不正确的选项可能会导致在查询中引用列时出错。

#### 数据类型

- 如果更改某列的数据类型，则新的数据类型必须与相应的数据源列或元素的数据类型兼容。将本地数据类型更改为与远程数据类型不兼容的数据类型可能会导致不可预测的错误。

- *local\_data\_type* 不能为 long VARCHAR、LONG VARGRAPHIC、DATALINK 或用户定义的数据类型。
- *data\_source\_data\_type* 不能为用户定义的类型。
- 不能对某些非关系数据源覆盖现有的本地类型或创建新的本地类型。关于此限制的更多信息，请参阅特定数据源包装器的文档。
- 当更改了某列的数据类型的本地规范时，联合数据库管理器可将为该列收集的任何统计信息（例如，HIGH2KEY 和 LOW2KEY）无效。
- 当使用该昵称存取特定的数据源对象时，会为该对象设置本地类型。相同的数据源对象可以具有使用缺省数据类型映射的不同昵称。

**索引** ALTER NICKNAME 语句不能用来在联合数据库中注册新的数据源索引。将 CREATE INDEX 语句与 SPECIFICATION ONLY 子句配合使用来创建索引规范。

### LOCAL NAME 和 LOCAL TYPE 参数

- 如果满足以下条件，则 ALTER NICKNAME 语句不能用于更改昵称中的列的本地名称或数据类型：
  - 昵称用于视图、SQL 方法或 SQL 函数中
  - 对昵称定义了信息约束
- 如果您还需要在 ALTER NICKNAME 语句中指定 LOCAL NAME 参数和 / 或 LOCAL TYPE 参数，则必须在最后指定 federated\_column\_options 子句。

**昵称** ALTER NICKNAME 语句不能用于更改由 BioRS 昵称引用或在 BioRS 昵称中使用的 BioRS 数据银行的名称。如果更改了 BioRS 数据银行的名称，则必须删除昵称并再次创建昵称。

### 工作单元

在以下任意一种条件下，联合服务器不能处理给定工作单元内的 ALTER NICKNAME 语句：

- 如果在 ALTER NICKNAME 语句中引用的昵称在相同的工作单元内打开了游标。
- 如果为在 ALTER NICKNAME 语句中引用的昵称在相同的工作单元中发出了插入、删除或更新操作。
- 对于非关系数据源，如果 ALTER NICKNAME 语句在相同的工作单元内引用由 SELECT 语句引用的昵称。

### 相关任务：

- 第 489 页的『改变昵称选项』
- 第 491 页的『改变数据源对象的本地类型』
- 第 485 页的『改变昵称』
- 第 488 页的『改变昵称列名』
- 第 490 页的『改变昵称列选项』

## 改变昵称列名

创建昵称时，与数据源对象关联的列名存储在联合数据库中。对于某些数据源，包装器会指定列名。对于其它数据源，必须在创建昵称时指定列名。

可以改变昵称以更改列名。

### 先决条件:

发出语句的授权标识必须至少包括下列一种特权:

- SYSADM 或 DBADM 权限
- 对在语句中指定的昵称的 ALTER 特权
- 对在语句中指定的昵称的 CONTROL 特权
- 对模式的 ALTERIN 特权（如果昵称的模式名存在）
- 记录在昵称的目录视图 DEFINER 列中的昵称的定义者

### 限制:

请参阅关于对改变昵称的限制的主题。

### 过程:

可以通过 DB2 控制中心或 DB2 命令行更改列名。

要通过 DB2 控制中心执行此任务:

1. 选择**昵称**文件夹。
2. 右键单击要更改的昵称，然后单击**改变**。此时会打开“改变昵称”笔记本。
3. 在“昵称”页上，选择要更改的列然后单击**更改**。此时会打开“更改列”窗口。
4. 键入列名。
5. 单击**确定**以更改列名并关闭窗口。
6. 单击**确定**以改变昵称并关闭笔记本。

要通过 DB2 命令行执行此任务，请发出 ALTER NICKNAME 语句:

```
ALTER NICKNAME nickname  
ALTER COLUMN current_name  
LOCAL NAME new_name
```

### 示例: 更改昵称列的局部名:

例如，DB2 UDB z/OS 版表的昵称 Z\_EMPLOYEES 包含名为 EMPNO 的列。要改变昵称以使用户使用的本地列名为 *Employee\_Number* 而不是 *EMPNO*，请发出以下语句:

```
ALTER NICKNAME Z_EMPLOYEES ALTER COLUMN EMPNO  
LOCAL NAME "Employee_Number"
```

### 相关任务:

- 第 485 页的『改变昵称』

### 相关参考:

- 第 486 页的『对改变昵称的限制』
- 『ALTER NICKNAME statement』 (*SQL Reference, Volume 2*)



## 改变昵称选项

昵称选项是您发出 CREATE NICKNAME 和 ALTER NICKNAME 语句时对昵称指定的参数。

可以通过使用 ALTER NICKNAME 语句来添加、设置或删除昵称选项。

### 先决条件:

发出语句的授权标识必须至少包括下列一种特权:

- SYSADM 或 DBADM 权限
- 对在语句中指定的昵称的 ALTER 特权
- 对在语句中指定的昵称的 CONTROL 特权
- 对模式的 ALTERIN 特权 (如果昵称的模式名存在)
- 在昵称的目录视图的 DEFINER 列中记录的昵称的定义器

### 限制:

请参阅关于对改变昵称的限制的主题。

### 过程:

可以通过 DB2 控制中心或 DB2 命令行更改列名。

要通过 DB2 控制中心执行此任务:

1. 选择**昵称**文件夹。
2. 右键单击要更改的昵称, 然后单击**改变**。此时会打开“改变昵称”笔记本。
3. 在“设置”页上, 选择要添加或删除的任何选项旁边的复选框。不能除去必需的选项。
4. 要指定或更改某个选项的值, 请单击该选项的**值**字段。取决于选项, 可以从列表中选择一个值, 单击以选择多个值或者输入新值。
5. 单击**确定**以改变昵称并关闭笔记本。

要通过命令行提示执行此任务, 请使用 ALTER NICKNAME 语句。例如:

```
ALTER NICKNAME nickname
      OPTIONS (SET option_name 'option_string_value')
```

例如, 为表结构文件 drugdata1.txt 创建了昵称 DRUGDATA1。原来在 CREATE NICKNAME 语句中定义的全限定路径为 /user/pat/drugdata1.txt。

要更改 FILE\_PATH 昵称选项, 请发出以下语句:

```
ALTER NICKNAME DRUGDATA1 OPTIONS (SET FILE_PATH '/usr/kelly/data/drugdata1.txt')
```

### 相关任务:

- 第 485 页的『改变昵称』

### 相关参考:

- 第 486 页的『对改变昵称的限制』
- 『ALTER NICKNAME statement』 (SQL Reference, Volume 2)
- 第 543 页的附录 F, 『联合系统的昵称选项』

## 改变昵称列选项

您在 CREATE NICKNAME 和 ALTER NICKNAME 语句使用称为昵称列选项的参数指定列信息。可以以大写或小写字符指定任何这些值。

可以使用 ALTER NICKNAME 语句添加、设置或删除昵称列选项。

### 先决条件:

发出语句的授权标识必须至少包括下列一种特权:

- SYSADM 或 DBADM 权限
- 对在语句中指定的昵称的 ALTER 特权
- 对在语句中指定的昵称的 CONTROL 特权
- 对模式的 ALTERIN 特权 (如果昵称的模式名存在)
- 记录在昵称的目录视图 DEFINER 列中的昵称的定义者

### 限制:

请参阅关于对改变昵称的限制的主题。

### 过程:

可以通过 DB2 控制中心或 DB2 命令行更改列名。

要通过 DB2 控制中心执行此任务:

1. 选择**昵称**文件夹。
2. 右键单击要更改的昵称, 然后单击**改变**。此时会打开“改变昵称”笔记本。
3. 在“昵称”页上, 选择要更改的列, 然后单击**更改**。此时会打开“更改列”窗口。
4. 选择要添加或去除的列选项。
5. 对于要添加或更改的选项, 指定选项的值。
6. 单击**确定**以更改列选项并关闭窗口。
7. 单击**确定**以改变昵称并关闭笔记本。

要通过命令行提示执行此任务, 请使用 ALTER NICKNAME 语句。

### 示例 1: 指定具有关系数据源的 NUMERIC\_STRING 列选项:

NUMERIC\_STRING 列选项适用于字符类型列 (CHAR 和 VARCHAR)。假定数据源具有不同于联合数据库整理顺序的整理顺序。联合服务器通常将不对包含数据源的字符数据的任何列排序。它将把数据返回到联合数据库和以本地方式执行排序。但是, 假定该列是字符数据类型和只包含数字字符 (“0”、“1”、... 和 “9”)。您可以通过将值 'Y' 指定给 NUMERIC\_STRING 列选项来指示此假定。这为 DB2 UDB 查询优化器提供了在数据源上执行排序的选项。如果排序以远程方式执行, 则您可以避免在联合服务器上排序数据的开销。

昵称 ORA\_INDSALES 用于名为 INDONESIA\_SALES 的 Oracle 表。该表包含数据类型为 VARCHAR 的列 POSTAL\_CODE。最初, 该列只包含数字字符, 并且 NUMERIC\_STRING 列选项设置为 'Y'。但是, 该列现在包含数字和非数字字符的混合字符。要将 NUMERIC\_STRING 列选项更改为 'N', 使用下列语句:

```
ALTER NICKNAME ORA_INDSALES ALTER COLUMN POSTAL_CODE
  OPTIONS (SET NUMERIC_STRING 'N')
```

**示例 2: 指定具有关系数据源的 VARCHAR\_NO\_TRAILING\_BLANKS 列选项:**

VARCHAR\_NO\_TRAILING\_BLANKS 列选项可用于标识不包含尾部空格的特定列。SQL 编译器检查对列执行的所有操作（例如，比较操作）时，SQL 编译器将此设置作为考虑因素。

昵称 ORA\_INDSALES 用于名为 INDONESIA\_SALES 的 Oracle 表。该表包含数据类型为 VARCHAR 的列 NAME。NAME 列没有结尾空白。要将 VARCHAR\_NO\_TRAILING\_BLANKS 选项添加到昵称，请使用下列语句:

```
ALTER NICKNAME ORA_INDSALES ALTER COLUMN NAME
  OPTIONS (ADD VARCHAR_NO_TRAILING_BLANKS 'Y')
```

**示例 3: 指定具有非关系数据源的 XPATH 列选项:**

昵称 EMPLOYEE 用于 XML 数据源。为 fname 列指定 XPATH。要将 XPATH 列选项设置为不同的路径，请使用下列语句:

```
ALTER NICKNAME EMPLOYEE ALTER COLUMN fname
  OPTIONS (SET XPATH './@first')
```

**相关任务:**

- 第 485 页的『改变昵称』

**相关参考:**

- 第 486 页的『对改变昵称的限制』
- 『ALTER NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 551 页的附录 G, 『联合系统的昵称列选项』

---

## 改变数据源对象的本地类型

创建昵称时，与数据源对象关联的数据类型存储在联合数据库中。对于一些数据源，包装器为您指定数据类型。对于其它数据源，必须在创建昵称时指定数据类型。

可以为特定数据源对象的列指定本地类型。您应使用 ALTER NICKNAME 语句而不是 CREATE TYPE MAPPING 语句。

**注意:** 如果将列的本地数据类型更改为与其远程类型差异较大的类型，则更改本地数据类型可能导致信息错误和丢失。

**先决条件:**

发出此语句的授权标识必须至少具有下列其中一种特权:

- SYSADM 或 DBADM 权限
- 对在语句中指定的昵称的 ALTER 特权
- 对在语句中指定的昵称的 CONTROL 特权
- 对模式的 ALTERIN 特权（如果昵称的模式名存在）

与语句关联的授权标识必须是在昵称的目录视图的 `DEFINER` 列中所记录昵称的定義者。

#### 限制:

请参阅关于对改变昵称的限制的主题。

#### 过程:

可以通过 DB2 控制中心或 DB2 命令行更改数据类型。

要通过 DB2 控制中心执行此任务:

1. 选择**昵称**文件夹。
2. 右键单击要更改的昵称，然后单击**改变**。此时会打开“改变昵称”笔记本。
3. 在“昵称”页上，选择要更改的列，然后单击**更改**。此时会打开“更改列”窗口。
4. 选择数据类型。
5. 单击**确定**以更改数据类型并关闭窗口。
6. 单击**确定**以改变昵称并关闭笔记本。

要通过命令行提示执行此任务，请使用 `ALTER NICKNAME` 语句。例如:

```
ALTER NICKNAME nickname ALTER COLUMN column_name
LOCAL TYPE data_type
```

要将具有字符数据类型的本地列的内容视为位（二进制）数据，需在 `ALTER NICKNAME` 语句中使用 `FOR BIT DATA` 子句。若使用此子句来更改某列的本地数据类型，则当与其它系统交换数据时，就不会执行代码页转换。比较是以二进制形式完成的，而不论远程数据库整理顺序如何。

#### 相关任务:

- 第 485 页的『改变昵称』

#### 相关参考:

- 第 486 页的『对改变昵称的限制』
- 『ALTER NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 557 页的附录 H，『缺省正向数据类型映射』
- 第 492 页的『为数据源对象改变本地类型 - 示例』
- 第 479 页的『非关系数据源支持的数据类型』

---

## 为数据源对象改变本地类型 - 示例

以下示例说明如何为数据源对象更改数据类型。

#### 示例: 数字数据类型映射:

在职员信息的 Oracle 表中，`BONUS` 列是用 `NUMBER(32,3)` 数据类型定义的。缺省情况下，Oracle 数据类型 `NUMBER(32,3)` 被映射至 DB2 数据类型 `DOUBLE`（双精度浮点数字数据类型）。包含 `BONUS` 列的查询所返回的值可能类似以下所示:

```
5.0000000000000E+002
1.0000000000000E+003
```

科学表示法指示小数位的数目和小数点应该移动的方向。在此示例中，+002 表示小数点应该右移两位，而 +003 表示小数点应该右移三位。

包含 BONUS 列的查询可以返回类似于美元金额的值。将表中 BONUS 列的本地定义从 DOUBLE 数据类型更改为 DECIMAL 数据类型。使用反映实际奖金格式的精度和小数位。例如，如果奖金的美元部分不能超过 6 位数，请将 NUMBER(32,3) 映射至 DECIMAL(8,2)。在此新本地类型的约束下，包含 BONUS 列的查询所返回的值如下所示：

```
500.00
1000.00
```

Oracle 表的昵称为 ORASALES。要将此 ORASALES 表中的 BONUS 列映射至 DB2 DECIMAL (8,2) 数据类型，请发出以下 ALTER NICKNAME 语句：

```
ALTER NICKNAME ORASALES ALTER COLUMN BONUS
LOCAL TYPE DECIMAL(8,2)
```

**ORASALES**

您为 Oracle 表定义的昵称。

**ALTER COLUMN BONUS**

以本地方式在联合数据库 SYSCAT.COLUMNS 目录视图中定义的列的名称。

**LOCAL TYPE DECIMAL(8,2)**

标识列的新本地类型。

| 此映射只适用于由昵称 ORASALES 标识的 Oracle 表中的 BONUS 列。包含 BONUS  
| 列的所有其它 Oracle 数据源对象对 Oracle NUMBER 数据类型使用缺省的数据类型映  
| 射。

示例：日期数据类型映射：

| 名为 SALES 的 Oracle 表的昵称为 ORASALES。SALES 表包含数据类型为 Oracle  
| DATE 的列。Oracle DATE 数据类型的缺省类型映射为至 DB2 TIMESTAMP 数据类  
| 型的映射。但是，当从此列中检索数据时，您只希望显示日期值。可以改变 SALES 表  
| 的昵称以将本地类型更改为 DB2 DATE 数据类型。

```
ALTER NICKNAME ORASALES ALTER COLUMN ORDER_DATE
LOCAL TYPE DATE
```

示例：非关系数据源的数据类型映射：

| 名为 drugdata1.txt 的表结构文件的昵称为 DRUGDATA1。drugdata1.txt 文件包  
| 含列示药物名称的列。列名为 DRUG。DRUG 列初始定义为 CHAR(20)。列长度必须更  
| 改为 CHAR(30)。可以改变 drugdata1.txt 文件的昵称以将映射更改为正确的长度：

```
ALTER NICKNAME DRUGDATA1 ALTER COLUMN DRUG
LOCAL TYPE CHAR(30)
```

相关任务：

- 『创建数据类型映射』（《联合系统指南》）
- 第 491 页的『改变数据源对象的本地类型』

相关参考：

- 『ALTER NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 486 页的『对改变昵称的限制』



---

## 第 29 章 DDL 命令参考

本章提供本书中讨论的包装器 DDL 命令的语法语句、自变量和选项的详细信息。这些语句是按包装器进行排序的。

---

### BioRS DDL 参考信息

#### CREATE SERVER 语句选项 - BioRS 包装器

BioRS 的 CREATE SERVER 语句的选项为:

**TYPE** 指定服务器类型。缺省值为 BioRS。缺省值是唯一受 BioRS 包装器支持的值。不需要指定此选项。

**VERSION**

指定服务器版本。缺省值为 1.0。缺省值是唯一受 BioRS 包装器支持的值。不需要指定此选项。

**NODE** 指定可以使用 BioRS 查询工具的系统的主机名。缺省值为 *localhost*。

**PORT** 指定要用来连接至 BioRS 服务器的端口号。缺省值为 5014。

**TIMEOUT**

指定 BioRS 包装器应该等待来自 BioRS 服务器的响应时间（以分钟计）。缺省值为 10。必须指定此选项。

**CASE\_SENSITIVE**

指定 BioRS 服务器是否以区分大小写的方式对待名称。有效值为 'Y' 或 'N'。缺省值为 'Y'。

在 BioRS 产品中，一个配置参数控制着存储在 BioRS 服务器上的数据的区分大小写性。CASE\_SENSITIVE 选项是该 BioRS 系统配置参数的 DB2 Information Integrator 对应物。必须使 BioRS 系统和 DB2 Information Integrator 中的 BioRS 服务器区分大小写配置设置同步。如果不使 BioRS 和 DB2 Information Integrator 之间的区分大小写配置设置保持同步，则当您尝试通过 DB2 Information Integrator 存取 BioRS 数据时将会出错。

**要点:** 在 DB2 Information Integrator 中创建新的 BioRS 服务器之后，就不能更改或删除 CASE\_SENSITIVE 选项了。如果需要更改 CASE\_SENSITIVE 选项，则必须删除然后再次创建整个服务器。如果删除 BioRS 服务器，还必须再次创建所有相应的 BioRS 昵称。DB2 Information Integrator 会自动删除与已删除的服务器对应的所有昵称。

**相关任务:**

- 第 66 页的『为 BioRS 数据源注册服务器定义』
- 第 67 页的『为 BioRS 数据源注册昵称』

**相关参考:**

- 『CREATE SERVER statement』（*SQL Reference, Volume 2*）
- 第 496 页的『CREATE NICKNAME 语句语法 - BioRS 包装器』

## CREATE USER MAPPING 语句选项 - BioRS 包装器

### GUEST

指定是否将在 BioRS 服务器上的 BioRS 来宾认证机制下执行操作。有效值为 'Y' 或 'N'。缺省值为 'Y'。

如果将此选项设置为 'Y'，则对此 DB2 Information Integrator 用户使用来宾认证来访问 BioRS 服务器。

如果将此选项设置为 'N'，则必须提供 BioRS 授权标识和密码才允许此 DB2 Information Integrator 用户访问 BioRS 服务器。

如果没有创建用户映射或者创建用户映射时没有指定选项，则对 DB2 Information Integrator 用户使用来宾认证来访问 BioRS 服务器。

### REMOTE\_AUTHID

指定允许此 DB2 用户存取 BioRS 数据源的用户标识。此远程标识必须具有 BioRS 应用程序期望的格式。如果将 GUEST 选项设置为 'N'，则此选项是必需的。

### REMOTE\_PASSWORD

指定此远程标识的密码。如果将 GUEST 选项设置为 'N'，则此选项是必需的。

### 示例:

以下 CREATE USER MAPPING 语句将用户 Charlie 映射至 Biors\_Server1 服务器上的用户 Charlene。

```
CREATE USER MAPPING FOR Charlie SERVER Biors_Server1
  OPTIONS(GUEST 'N', REMOTE_AUTHID 'Charlene', REMOTE_PASSWORD 'Charlene_pw');
```

### 相关任务:

- 第 66 页的『为 BioRS 数据源注册用户映射』

### 相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)

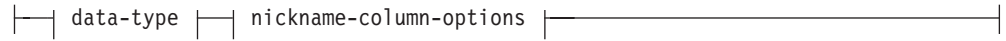
## CREATE NICKNAME 语句语法 - BioRS 包装器

用于 CREATE NICKNAME 语句的语法为:

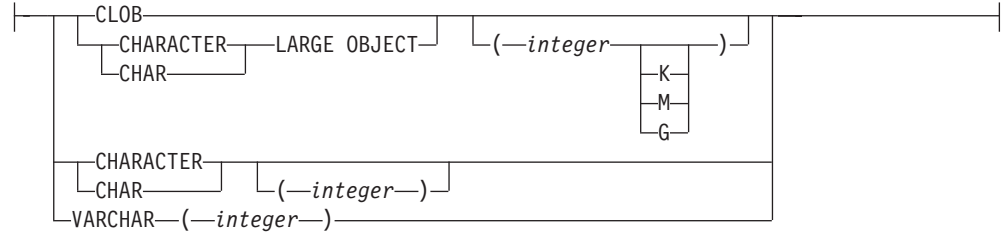
```
▶▶ CREATE NICKNAME—nickname—(—column-name—| column-information |—)▶▶
▶▶ FOR SERVER—server-name—OPTIONS—(—▶▶
▶▶ [REMOTE_OBJECT—'BioRS_databank_name' ]▶▶)
```

### column-information:

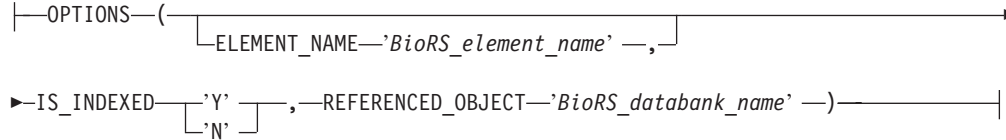




**data-type:**



**nickname-column-options:**



**昵称列选项**

昵称列选项值必须用单引号引起来。

**ELEMENT\_NAME**

指定 BioRS 元素名称。此名称是否区分大小写取决于 BioRS 服务器是否区分大小写以及 CASE\_SENSITIVE 服务器选项的值。仅当 BioRS 元素名称与列名不相同时才需要指定 BioRS 元素名称。

**IS\_INDEXED**

指示是否对相应的列建立索引（是否可以在谓词中引用该列）。有效值为 'Y' 和 'N'。只能对这样的列指定 'Y' 值：BioRS 服务器为该列的相应元素建立了索引。

在创建昵称时，将把带有 'Y' 值的此选项自动添加至任何与 BioRS 的已建立索引的元素相应的列。

**REFERENCED\_OBJECT**

此选项仅对其 BioRS 数据类型为“引用”的列有效。此选项指定当前列引用的 BioRS 数据银行的名称。此名称的区分大小写性取决于 BioRS 服务器的区分大小写性以及 CASE\_SENSITIVE 服务器选项的值。

**昵称选项**

昵称选项值必须用单引号引起来。

**REMOTE\_OBJECT**

指定与昵称相关联的 BioRS 数据银行的名称。此名称确定模式以及昵称的 BioRS 数据银行。此名称还指定此昵称与其它昵称之间的关系。此名称是否区分大小写取决于 BioRS 服务器是否区分大小写以及 CASE\_SENSITIVE 服务器选项的值。

**要点:** 不能用 ALTER NICKNAME 语句更改或删除此名称。如果此选项中使用的 BioRS 数据银行的名称进行了更改，则必须删除然后再次创建整个昵称。

**相关任务:**

- 第 67 页的『为 BioRS 数据源注册昵称』

相关参考:

- 『CREATE NICKNAME statement』 (*SQL Reference, Volume 2*)
- 第 69 页的『CREATE NICKNAME 语句 - BioRS 包装器的示例』

## BLAST DDL 参考信息

### CREATE SERVER 语句自变量 - BLAST 包装器

**BLAST 包装器的 CREATE SERVER 自变量:**

**TYPE** 确定使用给定的服务器执行的 BLAST 搜索的类型。此自变量是必需的。必须将它设置为下列值之一: blastn、blastp、blastx、tblastn 和 tblastx。

**VERSION**

指定正在使用的服务器的版本。应当将它设置为正在运行的 blastall 的版本。此自变量是必需的。

**WRAPPER**

指定您使用 CREATE WRAPPER 语句注册的包装器的名称。此自变量是必需的。

**BLAST 包装器的服务器选项:**

可以在 CREATE SERVER 语句中指定的 BLAST 的选项为:

- DAEMON\_PORT
- NODE
- USE\_CLOB\_SEQUENCE

相关任务:

- 第 98 页的『为 BLAST 数据源注册服务器』

相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)
- 第 529 页的附录 D, 『联合系统的服务器选项』

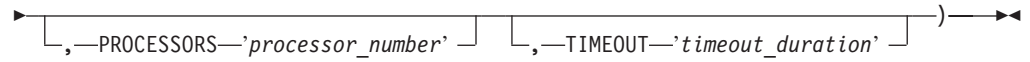
### CREATE NICKNAME 语句语法 - BLAST 包装器

用于 CREATE NICKNAME 语句的语法为:

```

▶▶ CREATE NICKNAME—nickname—(—column-name—| column-information |—)——▶
▶▶ FOR SERVER—server-name—OPTIONS—(—DATASOURCE—'data_source_name'——▶

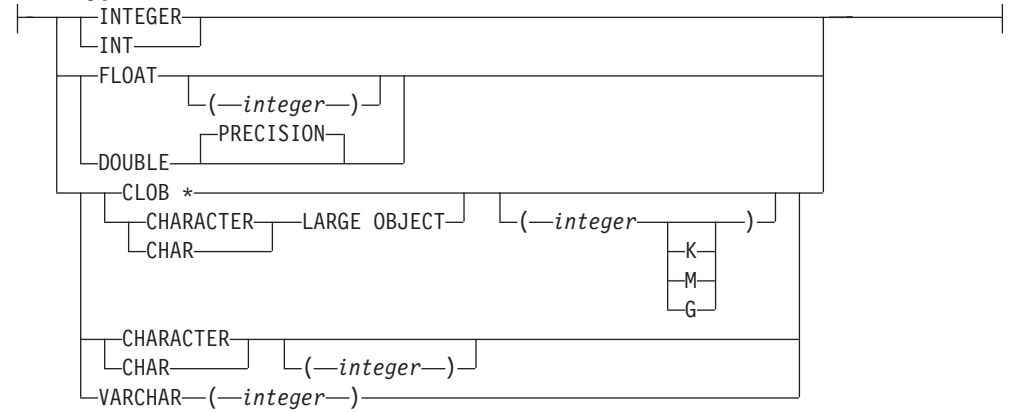
```



**column-information:**



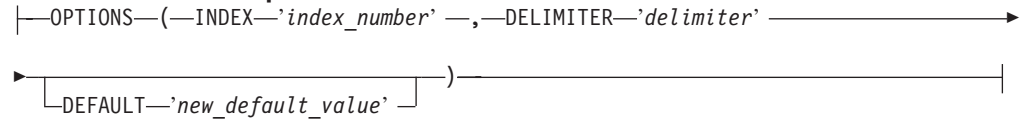
**data-type:**



**column-option:**



**nickname-column-options:**



限制: 对于 BLAST 包装器, CLOB 的长度仅限于 5 兆字节 (5MB)。

在 CREATE NICKNAME 语句中可以为 BLAST 数据源指定两种类型的选项:

- 昵称列选项
- 昵称选项

**相关任务:**

- 第 98 页的『为 BLAST 数据源注册昵称』

**相关参考:**

- 『CREATE NICKNAME statement』 (SQL Reference, Volume 2)
- 第 551 页的附录 G, 『联合系统的昵称列选项』
- 第 543 页的附录 F, 『联合系统的昵称选项』
- 第 103 页的『CREATE NICKNAME 语句 - BLAST 包装器的示例』

### CREATE SERVER 语句自变量和选项 - Documentum 包装器

与用于 Documentum 的 CREATE SERVER 语句相关联的自变量为:

**TYPE** 指定数据源的类型。对于 Documentum, 类型为 DCTM。此自变量是必需的。

**VERSION**

指定数据源的版本。对于 EDMS98, 该值为 “3”。对于 4i, 该值为 “4”。此自变量是必需的。

**WRAPPER**

指定与此服务器相关联的包装器的名称。此自变量是必需的。

与用于 Documentum 的 CREATE SERVER 语句相关联的选项为:

**CONTENT\_DIR**

指定用于存储内容文件的可本地存取的根本目录名称, 这些内容文件是由 GET\_FILE、GET\_FILE\_DEL、GET\_RENDITION 和 GET\_RENDITION\_DEL 伪列检索到的。可以使用这些伪列的所有用户必须都能够写入该目录。其缺省值为 /tmp。此选项是可选的。

**NODE** 指定 Documentum Docbase 的实际名称。此选项是必需的。

**OS\_TYPE**

指定 Docbase 服务器的操作系统。有效值为 AIX、SOLARIS 和 WINDOWS。此选项是必需的。

**RDBMS\_TYPE**

指定 Docbase 所使用的 RDBMS。有效值为 DB2、INFORMIX、ORACLE、SQLSERVER 或 SYBASE。此选项是必需的。

**TRANSACTIONS**

指定服务器事务方式。有效值为:

- NONE - 没有启用任何事务。
- QUERY - 仅对 Dctm\_Query 方法启用了事务。
- ALL - 对 Dctm\_Query 方法启用了事务。在此发行版中, ALL 与 QUERY 具有相同的功能。

缺省值为 QUERY。此选项是可选的。

相关任务:

- 第 161 页的『为 Documentum 数据源注册服务器』

相关参考:

- 『CREATE SERVER statement』 (*SQL Reference, Volume 2*)

### CREATE USER MAPPING 语句选项 - Documentum 包装器

选项定义:

**REMOTE\_AUTHID**

您在远程服务器中的授权标识。

## REMOTE\_PASSWORD

您在远程服务器中的密码。

## REMOTE\_DOMAIN

您在远程服务器中的 Windows 联网域。仅对于 Windows 平台有效。

相关任务:

- 第 161 页的『为 Documentum 数据源注册用户映射』

相关参考:

- 『CREATE USER MAPPING statement』 (*SQL Reference, Volume 2*)

## CREATE NICKNAME 语句语法 - Documentum 包装器

用于 Documentum 的 CREATE NICKNAME 语句的语法是:

```
▶▶ CREATE NICKNAME nickname ( column-name | column-information )
▶▶ FOR SERVER server-name OPTIONS (
    [ ALL_VERSIONS [ 'Y' ] | [ 'N' ] ],
    [ FOLDERS 'folder_string' ],
    [ IS_REG_TABLE [ 'Y' ] | [ 'N' ] ],
    REMOTE_OBJECT 'remote_object_type' )
```

### column-information:

```
| data-type | column-option | nickname-column-options |
```

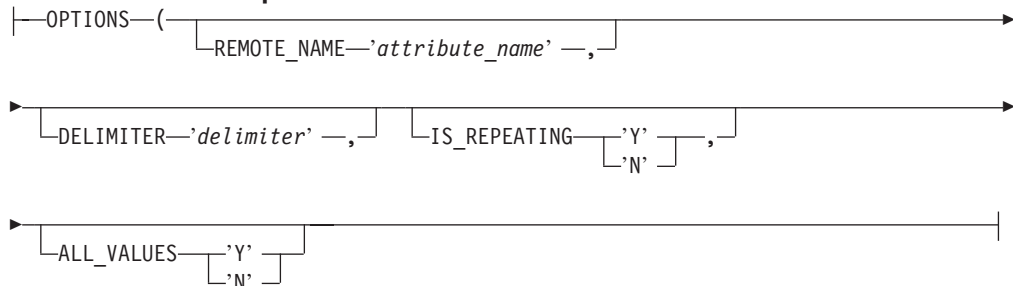
### data-type:

```
| SMALLINT |
| INTEGER |
| INT |
| DOUBLE |
| PRECISION |
| CHARACTER |
| CHAR | (integer) |
| VARCHAR | (integer) |
| DATE |
| TIMESTAMP |
```

### column-option:

```
| NOT NULL |
```

### nickname-column-options:



与用于 Documentum 的 CREATE NICKNAME 语句相关联的列选项为:

#### NOT NULL

除了定义为 `TIMESTAMP` 和 `DATE` 的那些列之外, 所有单值列都必须定义为 `NOT NULL`。在昵称中, 一定不能将重复的属性定义为 `NOT NULL`。

与用于 Documentum 的 CREATE NICKNAME 语句相关联的昵称列选项为:

昵称列选项值必须用单引号引起来。

#### ALL\_VALUES

指定将返回重复属性的所有值, 各个值之间用指定的定界符分隔。如果缺少此选项或者此选项为 `'N'`, 则将只返回重复属性的最后一个值。如 `DELIMITER` 中所述, 只能对 `IS_REPEATING` 选项为 `'Y'` (并且当 `IS_REG_TABLE = 'Y'` 时无效) 的 `VARCHAR` 列指定 `ALL_VALUES`。

#### DELIMITER

指定在并置重复属性的多个值时要使用的定界符字符串。定界符可以是一个或多个字符。缺省定界符为逗号。此选项仅对于 `IS_REPEATING` 选项设置为 `'Y'` 且数据类型为 `VARCHAR` 的对象的属性有效。此选项是可选的。

#### IS\_REPEATING

指示一列是否是多值的。有效值为 `'Y'` 和 `'N'`。缺省值为 `'N'`。此选项是可选的。

对于下列情况, 将只返回最后一个值

- 非 `VARCHAR` 重复属性
- 当指定了 `ALL_VALUES 'N'` 时的 `VARCHAR` 列

要克服这种限制, 可以为重复属性列创建双重定义。

#### REMOTE\_NAME

指定相应的 Documentum 属性或列的名称。此选项将远程属性或列名映射至本地 `DB2` 列名。它的缺省值为 `DB2` 列名。此选项是可选的。

与用于 Documentum 的 CREATE NICKNAME 语句相关联的昵称列选项为:

昵称选项值必须用单引号引起来。

#### ALL\_VERSIONS

指定是否将搜索所有对象版本。有效值为 `'y'`、`'Y'`、`'n'` 和 `'N'`。缺省值 `'N'` 意味着只有当前对象版本才包括在查询处理中。当 `IS_REG_TABLE = 'Y'` 时, 此选项无效。此选项是可选的。

## FOLDERS

指定包含一个或多个按逻辑组合起来的并且语法正确的 Documentum FOLDER 谓词的一个字符串。指定 FOLDER 谓词将把由此昵称表示的一组文档限制为在指定文件夹中的那些文档。

当指定此选项时，用单引号将 FOLDERS 选项的整个值引起来，并在字符串中使用双引号来替代单引号。

例如，如果想要插入：

```
FOLDER('/Tools',DESCEND) OR FOLDER('/Cars')
```

则指定以下 FOLDERS 选项：

```
FOLDERS 'FOLDER("/Tools",DESCEND) OR FOLDER("/Cars")'
```

当 IS\_REG\_TABLE = 'Y' 时，此选项无效。此选项是可选的。

## IS\_REG\_TABLE

指定由 REMOTE\_OBJECT 选项指定的对象是否是由 Documentum 注册的表。有效值为 'y'、'Y'、'n' 和 'N'。缺省值为 'N'。此选项是可选的。

不能通过使用 ALTER NICKNAME 语句更改此选项来将昵称从 Documentum 对象更改为已注册的表（反之亦然）。但是，必须先删除（DROP）该昵称，然后重新创建（CREATE）它。

## REMOTE\_OBJECT

指定与昵称相关联的 Documentum 对象类型的名称。该名称可以是任何 Documentum 对象类型或已注册的表。对于已注册的表，该名称应当添加表所有者的名称作为前缀。如果已注册的表属于 Docbase 所有者，则 dm\_dbo 可以用于所有者名。此选项是必需的。

通过使用 ALTER NICKNAME 来更改 REMOTE\_OBJECT 选项的值将导致错误（如果新对象的结构与原始对象的结构不相似的话）。

### 相关任务：

- 第 162 页的『为 Documentum 数据源注册昵称』

### 相关参考：

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 165 页的『CREATE NICKNAME 语句 - Documentum 包装器的示例』

---

## Excel DDL 参考信息

### CREATE SERVER 语句自变量 - Excel 包装器

与用于 Excel 的 CREATE SERVER 语句相关联的自变量为：

#### WRAPPER

指定您在相关联的 CREATE WRAPPER 语句中注册的包装器的名称。此自变量是必需的。

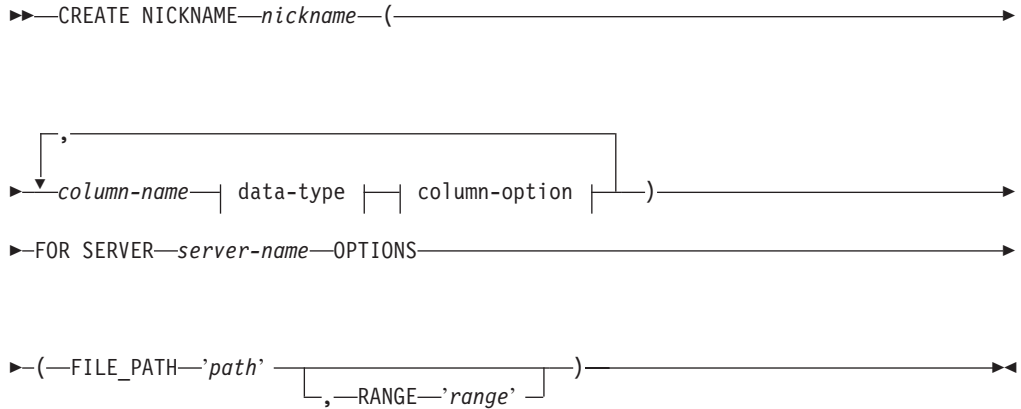
### 相关任务：

- 第 204 页的『为 Excel 数据源注册服务器』

相关参考:

- 『CREATE SERVER statement』 (SQL Reference, Volume 2)

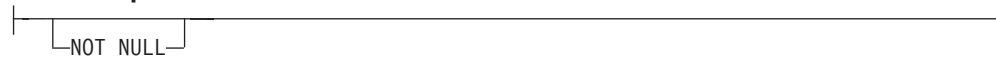
## CREATE NICKNAME 语句语法 - Excel 包装器



### data-type:



### column-option:



其中:

### FOR SERVER

标识您在相关联的 CREATE SERVER 语句中注册的服务器。此服务器用来存取 Excel 电子表格。指定服务器名。

下表描述了 Excel 的 CREATE NICKNAME 选项:

### FILE\_PATH

指定想要存取的 Excel 电子表格的全限定目录路径和文件名。

在每一列中数据类型必须一致，并且在注册昵称过程中必须正确描述列数据类型。

Excel 包装器只能存取 Excel 工作簿中的主要电子表格。

电子表格中的空白单元格将被解释为 NULL。

电子表格中最多可以存在 10 个连续的空白行，并且数据集中最多可以包括 10 个连续的空白行。超过 10 个连续的空白行就会被认为是数据集的末尾。

电子表格中可以存在空白列。但是，即使不会使用这些列，也必须注册这些列并将它们描述为有效字段。



数据库代码页必须与文件的字符集相匹配；否则，可能会获得意外的结果。

## RANGE

指定在数据源中要使用的单元的范围。此选项不是必需的。

范围选项值中的任何语法或语义错误都将导致产生 SQL1882E 消息。错误可能包括：

- 左上部和右下部指示符的定向不正确。不正确的定向就是左上部单元指示符在右下部单元指示符的下方或右边。
- 由范围值指定的列数并不对应于在 CREATE NICKNAME 语句中指定的列数。
- 找到了无效字符或其它语法错误。

以下是昵称选项 RANGE 的一个示例：

```
CREATE NICKNAME excel2
(c1 VARCHAR (10),
c2 VARCHAR (10),
c3 VARCHAR (10),
c4 VARCHAR (10)
) FOR SERVER excel_server
OPTIONS (FILE_PATH 'C:\My Documents\test2.xls',
RANGE 'B2:E5');
```

在此示例中，**B2** 表示单元范围的左上部，**E5** 表示单元范围的右下部。B2 表示法中的字母 *B* 是列标识。B2 表示法中的数字 2 是行号。

右下部表示法可以从范围中省略。在此情况下，将使用右下部的有效行。如果省略了左上部的值，则将值看作是 *A1*。如果由范围指定的行数超过了电子表格中实际存在的行数，则将使用实际的行数。

### 相关任务：

- 第 205 页的『为 Excel 数据源注册昵称』

### 相关参考：

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）

---

## Extended Search DDL 参考信息

### CREATE WRAPPER 语句语法 - Extended Search 包装器

►►—CREATE WRAPPER—*wrapper-name*—LIBRARY—'*library-name*'—◄◄

#### WRAPPER

指定此 Extended Search 包装器的唯一名称。

#### LIBRARY

指定与平台相关的下列库名之一：

- Windows: db2uies.dll
- AIX: libdb2uies.a

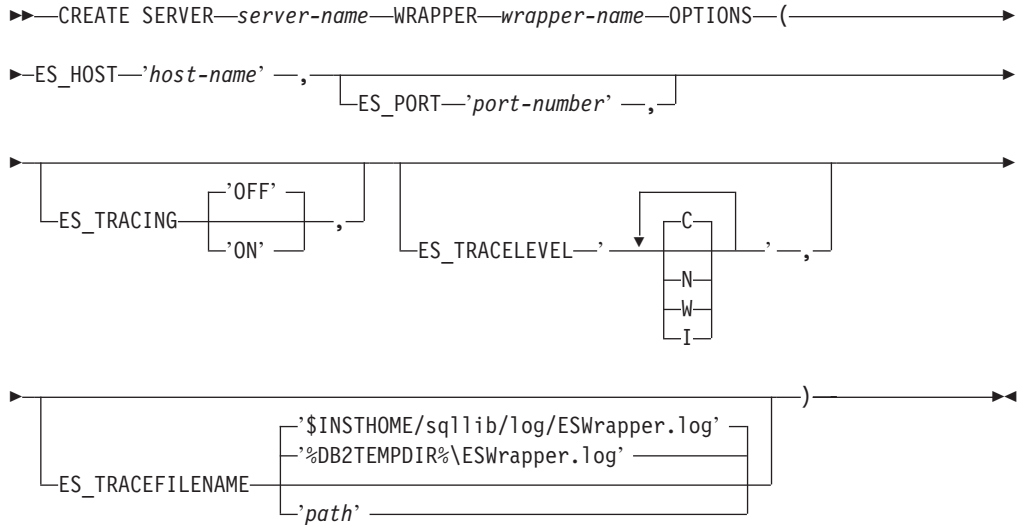
### 相关任务：

- 第 219 页的『注册 Extended Search 包装器』

相关参考:

- 『CREATE WRAPPER statement』 (SQL Reference, Volume 2)

## CREATE SERVER 语句语法 - Extended Search 包装器



### SERVER

指定此服务器定义的唯一名称。此参数是必需的。

### WRAPPER

指定想要与此服务器定义配合使用的先前已注册的 Extended Search 包装器的名称。此参数是必需的。

### ES\_HOST

指定想要搜索的 Extended Search 服务器的全限定主机名或 IP 地址。此选项是必需的。

### ES\_PORT

指定此 Extended Search 服务器侦听请求的端口号。如果省略了此选项，则缺省值为 6001。

### ES\_TRACING

指定是否应当对远程 Extended Search 服务器产生的错误消息、警告消息和参考消息启用跟踪。缺省值 OFF 意味着将不会记录任何跟踪消息。

### ES\_TRACELEVEL

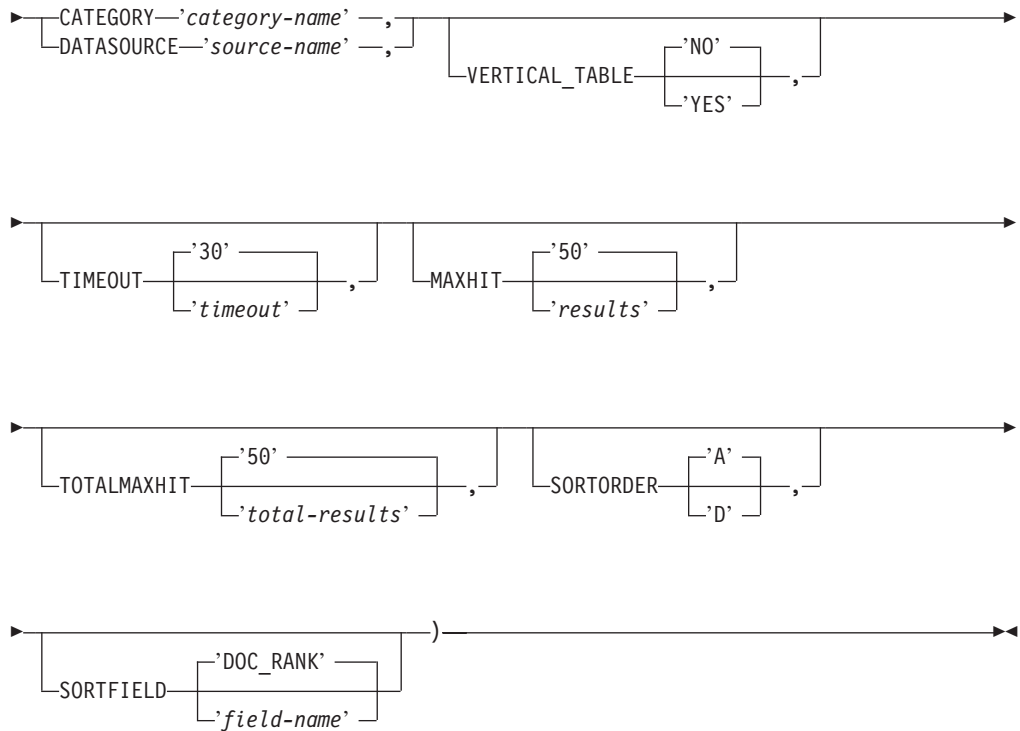
如果启用了跟踪，则此选项指定将写入日志文件中的消息类型。缺省值 C 意味着将只记录关键消息。可以单独启用和禁用下列跟踪级别:

- C - 关键错误消息
- N - 非关键消息
- W - 警告消息
- I - 参考消息

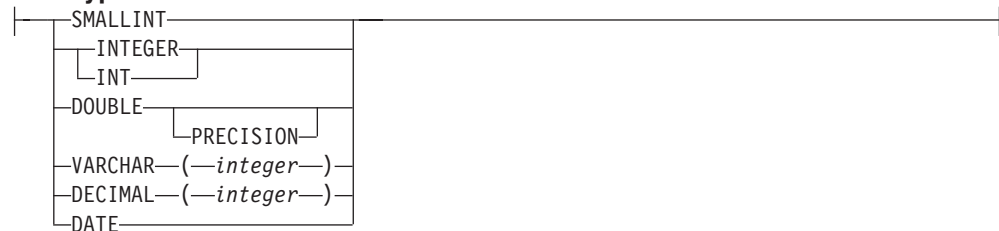
例如:

```
ES_TRACELEVEL 'W'
ES_TRACELEVEL 'CN'
ES_TRACELEVEL 'CNWI'
```





**data-type:**



**NICKNAME**

指定此 Extended Search 昵称表的唯一名称。此名称必须与正在为其定义该名称的模式中的所有其它昵称都不相同。此参数是必需的。

*column-name*

指定一个或多个用户定义的列名。列名必须与在 Extended Search 配置数据库中定义的本地字段或已映射的字段名称相匹配。此参数是可选的。

*data-type*

指定所指定的列的 SQL 数据类型。此数据类型必须对应于在 Extended Search 配置数据库中为此字段定义的数据类型。例如，要搜索 Extended Search 数据源中具有 String 数据类型的一个字段，在昵称表中为此字段定义一个 VARCHAR 列。如果指定了 *column-name*，则此参数是必需的。

**FOR SERVER**

指定先前已注册的服务器定义的名称，该服务器定义是为想要搜索的 Extended Search 服务器创建的。此参数是必需的。

#### **APPLICATIONID**

指定想要搜索的 Extended Search 应用程序的名称。Extended Search 配置数据库中必须存在此名称。此参数是必需的。

#### **CATEGORY**

指定想要搜索的一个或多个 Extended Search 类别。如果省略了此选项，则必须至少指定一个数据源名称。要指定多个类别，使用分号来隔开类别名称。例如：

```
CATEGORY 'LotusNotes;MSAccess;LDAP'
```

#### **DATASOURCE**

指定想要搜索的一个或多个 Extended Search 数据源。如果省略了此选项，则必须至少指定一个类别名称。要指定多个数据源，使用分号来为数据源名称定界。例如：

```
DATASOURCE 'AltaVista;Google!;CNN'
```

#### **VERTICAL\_TABLE**

指定搜索结果的表示格式。如果指定了 YES，则 Extended Search 将返回配置为可返回的所有字段，而不只是用户定义的列。包装器将结果以列名的垂直列表的形式存储在昵称表中。缺省值为 NO。

#### **TIMEOUT**

一个 INTEGER，它指定在请求超时之前要等待来自服务器的响应的秒数。此选项是可选的。缺省值为 30。

#### **MAXHIT**

一个 INTEGER，它指定可以从正在搜索的每个源中返回的最大结果数。此选项是可选的。缺省值为 50。

#### **TOTALMAXHIT**

一个 INTEGER，它指定可以从正在搜索的所有源中返回的最大结果数。包装器将这些结果组合成单个结果集。此选项是可选的。缺省值为 50。

#### **SORTORDER**

指定搜索结果返回时的排序顺序，要么为升序 (A)，要么为降序 (D)。缺省值为 A。

#### **SORTFIELD**

指定应当对其搜索结果进行排序的字的名称。缺省值 DOC\_RANK 是 Extended Search 用来衡量结果文档的相关性的一个字段。如果指定了另一个字段名，则应确保该名称存在于搜索的源中。

#### **相关概念:**

- 第 215 页的『Extended Search 昵称』
- 第 216 页的『Extended Search 垂直表』

#### **相关任务:**

- 第 221 页的『为 Extended Search 数据源注册昵称』

#### **相关参考:**

- 『CREATE NICKNAME statement』 ( *SQL Reference, Volume 2* )
- 第 225 页的『Extended Search 包装器 - 示例查询』

## CREATE FUNCTION 语句语法 - Extended Search 包装器

CREATE FUNCTION 语句的语法为:

```
►—CREATE FUNCTION ESWRAPPER.ES_SEARCH—(—INTEGER—,—VARCHAR(1024)—)—————►
```

```
►—RETURNS INTEGER AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION—————►
```

### INTEGER

定义查询引用参数。在查询中，此参数必须指定在昵称表中定义的 INTEGER 列的名称，由该列名称调用此定制函数。值必须为昵称的绑定列，而不是一个常量（例如，DOC\_RANK）。

引用参数标识应当将 ES\_SEARCH 函数应用于的昵称。不会对该参数本身求值。

如果 SELECT 语句的 FROM 子句中包含多个表，而 WHERE 子句中包含 ES\_SEARCH 语句，则引用参数允许您告诉 DB2 特定的搜索语句属于哪个表。例如：

```
SELECT *
FROM ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE ESWRAPPER.ES_SEARCH(N1.DOC_RANK, 'IBM')=1 AND
      ESWRAPPER.ES_SEARCH(N2.DOC_RANK, 'LOTUS')=1
```

### VARCHAR(1024)

定义查询表达式。在查询中，此参数必须指定一个使用 Extended Search 通用查询语言的字符串。

#### 相关任务:

- 第 222 页的『为 Extended Search 包装器注册定制函数』

#### 相关参考:

- 『CREATE FUNCTION (Sourced or Template) statement』 (*SQL Reference, Volume 2*)
- 第 225 页的『Extended Search 包装器 - 示例查询』
- 第 227 页的『Extended Search 包装器 - 通用查询语言』

---

## Entrez DDL 参考信息

### CREATE SERVER 语句自变量 - Entrez 包装器

用于 Entrez 的 CREATE SERVER 语句的自变量为:

**TYPE** 指定数据源的类型。服务器类型可接受的值为 PubMed 和 Nucleotide。这些值不区分大小写。

#### VERSION

指定正在使用的 NCBI XML 模式的版本。此自变量是可选的。如果未指定服务器的版本，则缺省值为 1.0。

#### WRAPPER

指定您使用 CREATE WRAPPER 语句注册的包装器的名称。

#### 相关任务:

- 第 183 页的『为 Entrez 数据源注册服务器』

相关参考:

- 『CREATE SERVER statement』 (SQL Reference, Volume 2)
- 第 529 页的附录 D, 『联合系统的服务器选项』

## CREATE NICKNAME 语句选项 - Entrez 包装器

下表描述了 Entrez 的 CREATE NICKNAME 选项:

### REMOTE\_OBJECT

指定与昵称相关联的 Entrez 对象类型的名称。此名称将确定昵称的模式和 NCBI 数据库以及该名称与其它昵称的关系。此名称不区分大小写。

### PARENT

仅对已经通过 REMOTE\_OBJECT 选项对其父昵称进行了重命名的子昵称指定此选项。当在 DB2 模式中定义了多个昵称系列时, PARENT 选项将把子昵称与父昵称关联起来。此名称是区分大小写的。

相关任务:

- 第 185 页的『为 Entrez 数据源注册昵称』

相关参考:

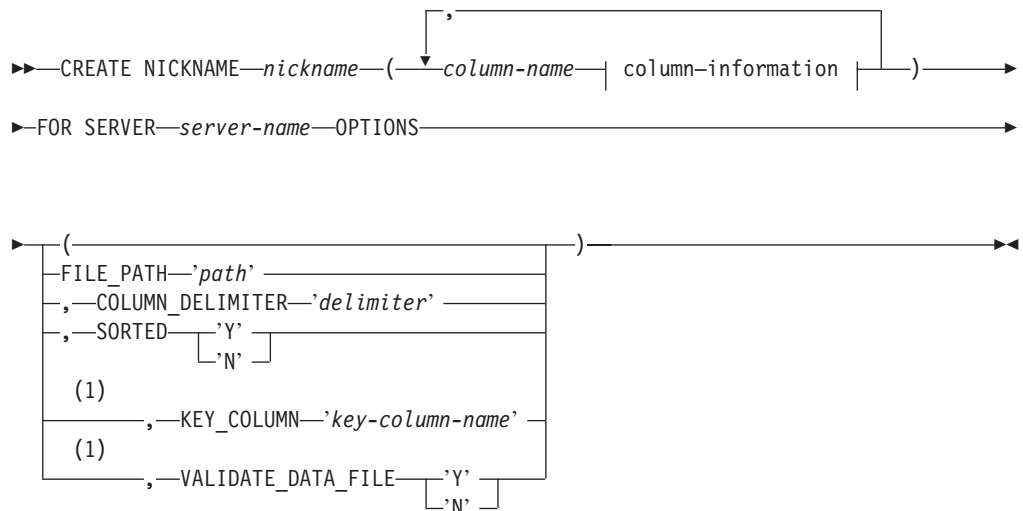
- 『CREATE NICKNAME statement』 (SQL Reference, Volume 2)

---

## 表结构文件 DDL 参考信息

### CREATE NICKNAME 语句语法 - 表结构文件包装器

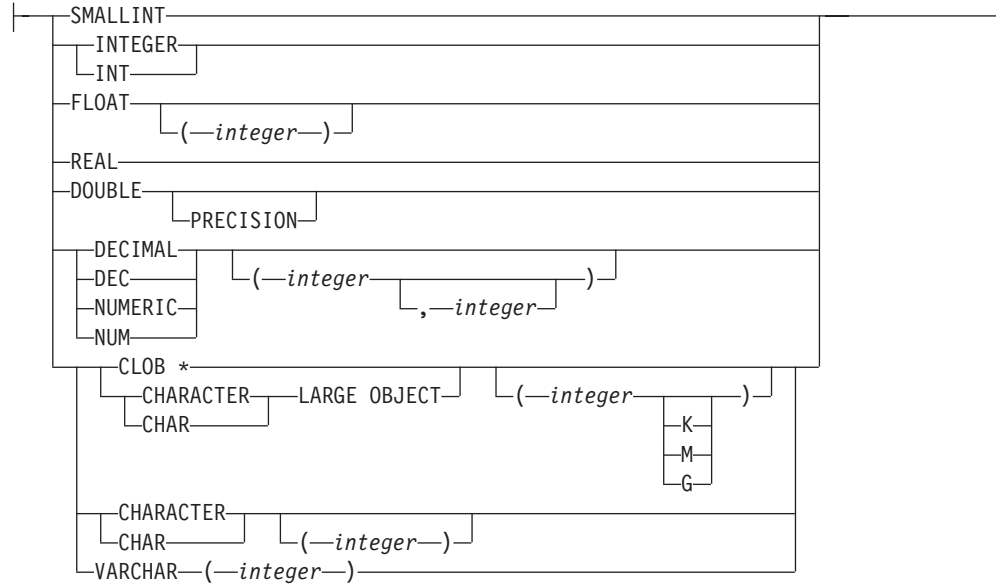
用于 CREATE NICKNAME 语句的语法为:



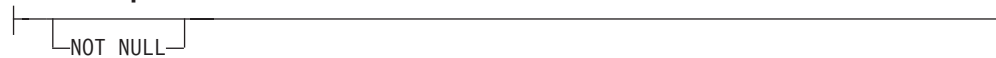
**column-information:**

| data-type | column-option | nickname-column-options |

**data-type:**



**column-option:**



**nickname-column-options:**



**注:**

1 不允许用于未排序的文件。对于已排序的文件是可选的。

**限制:** 对于表结构文件包装器，CLOB 的长度仅限于 5 兆字节 (5MB)。

*nickname*

要存取的表结构文件的唯一昵称。该昵称必须与注册该昵称的模式中的其它所有昵称、表和视图都不相同。

*column-name*

为表结构文件中的每个字段指定的唯一名称。在每个列名后面紧接着它的数据类型。仅支持类型为 CHAR、CLOB、DECIMAL、DOUBLE、FLOAT、INTEGER、REAL、SMALLINT 和 VARCHAR。

**CHARACTER(integer) 或 CHAR(integer) 或者 CHARACTER 或 CHAR**

表示长度为 *integer* 的定长字符串，其范围可以是 1 到 254。如果省略了长度规范，则假定其长度为一个字符。

**CLOB(integer)**

对于长度为 *integer* 的字符大对象，其最大长度为 5 兆字节。如果省略了长度规范，则假定其长度为 1 兆字节。

**DECIMAL(precision-integer, scale-integer) 或 DEC(precision-integer, scale-integer)**

表示十进制数。

第一个整数是该数的精度；即总位数。此值的范围可以是 1 到 31。



第二个整数是该数的小数位；即小数点右边的位数。此值的范围可以是 0 到该数的精度。

如果未指定精度和小数位，则使用缺省值 5,0。

可以将词语 **NUMERIC** 和 **NUM** 用作 **DECIMAL** 和 **DEC** 的同义词。

### **DOUBLE 或 DOUBLE PRECISION**

表示双精度浮点。

### **FLOAT(*integer*)**

表示单精度或双精度浮点数，这取决于 *integer* 的值。*integer* 的值必须在范围 1 到 53 之间。值 1 到 24 表示单精度，值 25 到 53 表示双精度。

### **INTEGER 或 INT**

表示大整数。

**REAL** 表示单精度浮点。

### **SMALLINT**

表示小整数。

### **VARCHAR(*integer*)**

表示最大长度为 *integer* 的变长字符串，其范围可以是 1 到 32672。

### **NOT NULL**

这可以防止列中包含空值。

包装器并不强制 NOT NULL 约束，但是 DB2 会强制 NOT NULL 约束。如果您创建了一个昵称，并且对一列附加了 NOT NULL 约束，然后却为该列选择了包含空值的行，则 DB2 将发出 SQL0407N 错误，指出不能为 NOT NULL 列指定 NULL 值。

对于已排序的昵称，此规则不适用。已排序的昵称的键列不能为 NULL。如果发现已排序的昵称具有 NULL 键列，则会发出 SQL1822N 错误，指出缺少该键列。

### **FOR SERVER**

标识您使用 CREATE SERVER 语句注册的服务器。此服务器将用来存取表结构文件。

### **FILE\_PATH**

要存取的表结构文件的全限定路径，用单引号引起来。数据文件必须是标准文件或符号链接，而不是管道或另一种非标准文件类型。应当指定 FILE\_PATH 或 DOCUMENT 昵称列选项。如果指定了 FILE\_PATH 昵称选项，则不能指定 DOCUMENT 昵称列选项。

### **SORTED**

指定数据源文件是已排序的还是未排序的。此选项接受 'Y'、'y'、'n' 或 'N'。缺省值为 'N'。

如 LC\_COLLATE “本地语言支持”类别中的设置所定义，排序的数据源必须根据当前语言环境的整理序列按升序排序。

如果您指定数据源已排序，则建议将 VALIDATE\_DATA\_FILE 设置为 'Y'。

### **COLUMN\_DELIMITER**

用来将表结构文件的各列分隔开的定界符，用单引号将它引起来。定界符可以是单个字符或多个字符。如果没有定义列定界符，则将把列定界符缺省设置为

逗号。不能将单引号用作定界符。在整个文件中，列定界符必须保持一致。如果行尾为 NULL 字段，空值可以由两个相邻的定界符或定界符后跟行终结符来表示。列定界符不能作为列的有效数据存在。例如，如果其中一列包含嵌入了逗号的数据，则不能使用逗号作为列定界符。

### KEY\_COLUMN

文件中组成键（文件已对该键进行排序）的列的名称，用单引号引起来。仅对已排序的文件使用此选项。一定不能将使用 DOCUMENT 昵称列选项指定的列指定为键列。

只支持单列键。不允许多列键。值必须是在 CREATE NICKNAME 语句中定义的列名。列必须按升序排序。如果没有为已排序的昵称指定值，则将它缺省为已创建昵称的文件中的第一列。建议通过在昵称语句中将 NOT NULL 选项添加到键列的定义中来指定键列不可为空。

此选项是区分大小写的。但是，DB2 将把列名转换为大写（除非该列是用双引号定义的）。

### VALIDATE\_DATA\_FILE

对于已排序的文件，此选项指定包装器是否将验证键列是按升序顺序排序的，并且将检查 NULL 键。此选项的唯一有效值是 'Y' 或 'N'，用单引号引起来。在注册时就完成了一次检查。如果未指定此选项，则不会进行验证。如果 DOCUMENT 昵称列选项用于文件路径，则不允许使用此选项。

### DOCUMENT

指定表结构文件的类型。当前，对于此选项，此包装器只支持 FILE。通过使用 DOCUMENT 选项，只能对每个昵称指定一列。与 DOCUMENT 选项相关联的列的数据类型必须是 VARCHAR 或 CHAR。

使用 DOCUMENT 昵称列选项，而不使用 FILE\_PATH 昵称选项，这意味着将在执行查询期间提供与此昵称相对应的文件。如果 DOCUMENT 选项的值为“FILE”，则意味着在执行查询期间所提供的内容是其模式与此昵称的昵称定义相匹配的文件的全路径。以下 CREATE NICKNAME 示例说明了如何使用 DOCUMENT 昵称列选项。

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name VARCHAR(16),
  address VARCHAR(30),
  ID VARCHAR(16)
)
FOR SERVER file_server
```

现在，可以对 customers 昵称运行以下查询，在该查询的 WHERE 子句中指定了表结构文件的位置：

```
SELECT name, address, id FROM customers
WHERE doc='/home/db2user/Customers.txt'
```

#### 相关任务:

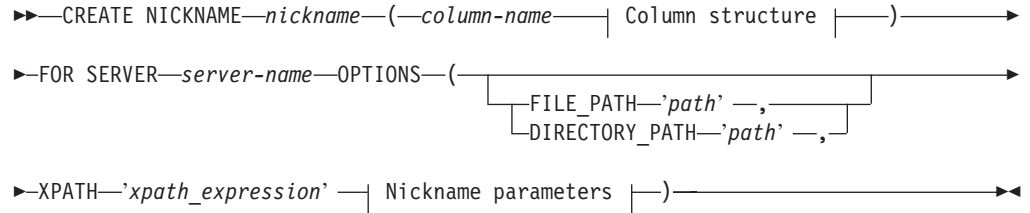
- 第 333 页的『为表结构文件注册服务器』

#### 相关参考:

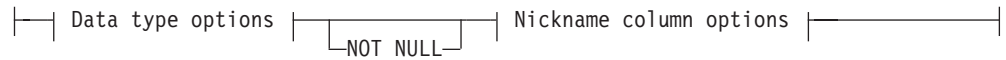
- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 334 页的『CREATE NICKNAME 语句 - 表结构文件包装器的示例』

## CREATE NICKNAME 语句语法 - XML 包装器

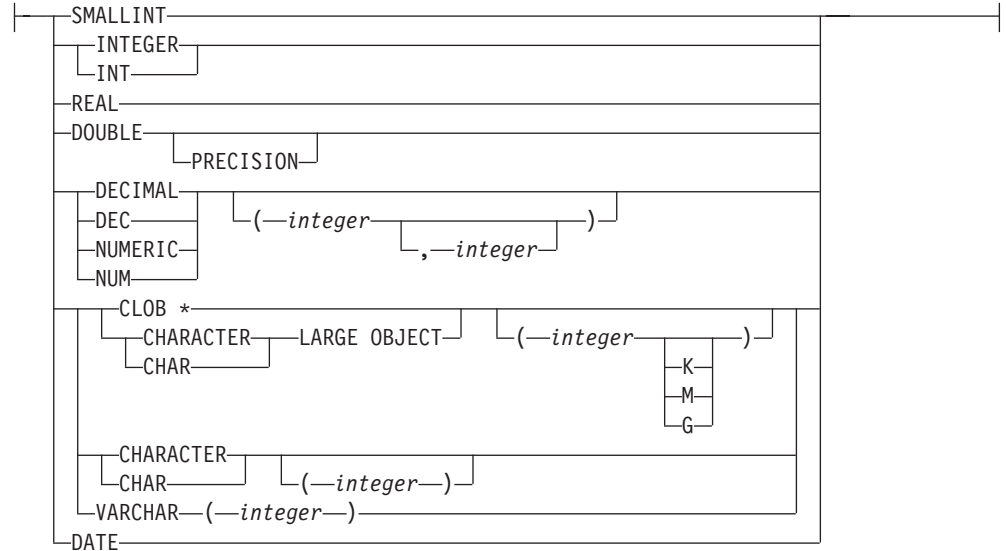
用于 CREATE NICKNAME 语句的语法为:



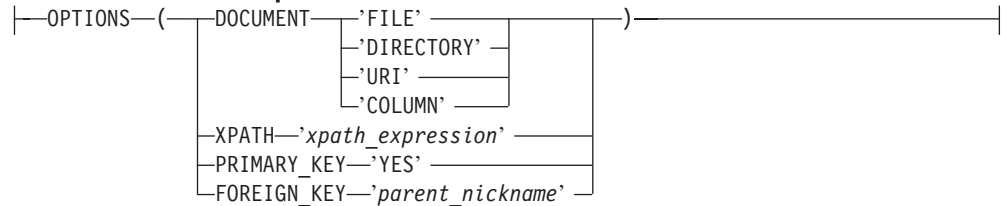
### Column structure:



### Data type options:

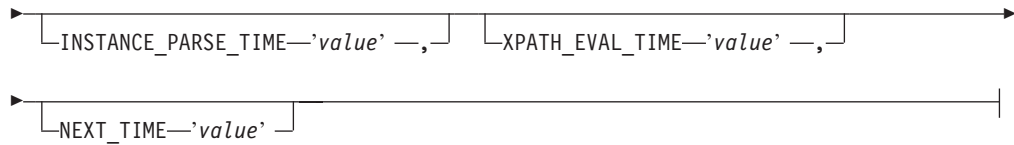


### Nickname column options:



### Nickname parameters:





限制: 对于 XML 包装器, CLOB 的长度仅限于 5 兆字节 (5MB)。

昵称参数和选项:

### FILE\_PATH

指定 XML 文档的文件路径。如果指定此昵称选项, 则不要指定 DOCUMENT 列。只有根昵称 (标识位于 XML 文档顶层的元素的昵称) 才接受此 FILE\_PATH 选项。

### DIRECTORY\_PATH

指定包含一个或多个 XML 文件的目录的路径名。使用此选项来对多个 XML 源文件创建单个昵称。XML 包装器只使用位于您指定的目录中并且扩展名为 .xml 的那些文件。XML 包装器将忽略此目录中其它所有文件。如果指定此昵称选项, 则不要指定 DOCUMENT 列。只有根昵称 (标识位于 XML 文档顶层的元素的昵称) 才接受此 DIRECTORY\_PATH 选项。

### XPATH

指定用来标识表示个别元组的 XML 元素的 XPath 表达式。子昵称的 XPATH 昵称选项是在由其父昵称的 XPATH 昵称选项所指定的路径的上下文中求值的。此 XPath 表达式被用作对由 XPATH 昵称列选项标识的列值进行求值的上下文。

在 XPath 表达式中不要指定名称空间前缀。XML 包装器不支持名称空间。

**Nickname column options:**

### DOCUMENT

指定此列是 DOCUMENT 列。DOCUMENT 列的值指示运行查询时为昵称提供的 XML 源数据的类型。只有根昵称 (标识位于 XML 文档顶层的元素的昵称) 的各列才接受此选项。通过使用 DOCUMENT 选项, 只能对每个昵称指定一列。与 DOCUMENT 选项相关联的列的数据类型必须是 VARCHAR。

如果使用 DOCUMENT 列选项, 而不使用 FILE\_PATH 或 DIRECTORY\_PATH 昵称选项, 则运行查询时将提供与此昵称相对应的文档。

DOCUMENT 选项的有效值为:

**FILE** 指定昵称列的值仅限于包含 XML 文档的文件的名称。运行查询时将提供此文件的数据。

### DIRECTORY

指定昵称列的值仅限于包含多个 XML 数据文件的目录的路径名。多个文件中的 XML 数据将在运行查询时提供。这些数据位于驻留在指定的目录路径中的 XML 文件中。XML 包装器只使用位于您指定的目录中并且扩展名为 .xml 的那些文件。XML 包装器将忽略此目录中其它所有文件。

**URI** 指定昵称列的值仅限于 URI 所指示的远程 XML 文件的名称。URI 地址指示此 XML 文件在 Web 上的远程位置。

## COLUMN

指定 XML 文档存储在关系列中。

## XPATH

指定 XML 文档中包含与此列相对应的数据的 XPath 表达式。在 CREATE NICKNAME 语句从此 XPATH 昵称选项中应用此 XPath 表达式之后，XML 包装器才对 XPath 表达式求值。

如果对具有不正确配置的 XPATH 标记引用（例如，不正确的大小写）的列名运行查询，则查询将在返回的所有行的此列中都返回空值。

在 XPath 表达式中不要指定名称空间前缀。XML 包装器不支持名称空间。

## PRIMARY\_KEY

指示此昵称是一个父昵称。列数据类型必须是 VARCHAR(16)。一个昵称最多只能具有一个 PRIMARY\_KEY 列选项。'YES' 是唯一的有效值。使用此选项指定的列将保留由包装器生成的键。在 SELECT 查询中将不能检索该列的值，并且一定不能为此列指定 XPATH 选项。该列只能用来连接父昵称和子昵称。

## FOREIGN\_KEY

指示此昵称是一个子昵称，并指定相应父昵称的名称。一个昵称最多只能具有一个 FOREIGN\_KEY 列选项。此选项的值是区分大小写的。使用此选项指定的列将保留由包装器生成的键。在 SELECT 查询中将不能检索该列的值，并且一定不能为此列指定 XPATH 选项。该列只能用来连接父昵称和子昵称。

如果父昵称具有不同的模式名，则带有 FOREIGN\_KEY 选项的 CREATE NICKNAME 语句将失败。

除非将 FOREIGN\_KEY 子句中引用的昵称明确定义为小写或混合大小写（通过在相应的 CREATE NICKNAME 语句中用引号将它引起来），否则，当您在 FOREIGN\_KEY 子句中引用此昵称时，必须用大写来指定该昵称。

## Nickname parameters:

### STREAMING

指定是否将 XML 源文档分成许多逻辑段，这些逻辑段对应于与昵称的 XPath 表达式相匹配的节点。然后，XML 包装器按段对 XML 源数据进行语法分析和处理，以减少使用的内存总量。可以对任何 XML 源文档（FILE、DIRECTORY、URI 或 COLUMN）指定 STREAMING。只有根昵称（标识位于 XML 文档顶层的元素的昵称）的各列才接受此选项。STREAMING 的缺省值为 NO。

如果将 VALIDATE 参数设置为 YES，则不要将 STREAMING 参数设置为 YES。如果将它们都设置为 YES，将接收到错误消息。

### VALIDATE

指定在抽取 XML 数据之前是否验证 XML 源文档。如果将此选项设置为 YES，则昵称选项将验证源文档的结构是否符合 XML 模式或文档类型定义（DTD）。只有根昵称（标识位于 XML 文档顶层的元素的昵称）的各列才接受此选项。缺省值为 NO。

如果 XML 包装器找不到 XML 模式文件或 DTD 文件（.xsd 或 .dtd），则不会验证 XML 源文档。如果不进行验证，则 DB2 不会发出错误消息。因此，应确保 XML 模式文件或 DTD 文件存在于在 XML 源文档中指定的位置。

如果将 STREAMING 参数设置为 YES，则不要将 VALIDATE 参数设置为 YES。如果将它们都设置为 YES，将接收到错误消息。

### **INSTANCE\_PARSE\_TIME**

指定对 XML 源文档中的一行数据进行语法分析所需要的时间（以毫秒计）。可以修改 INSTANCE\_PARSE\_TIME、XPath\_EVAL\_TIME 和 NEXT\_TIME 选项以优化大型的或复杂的 XML 源结构的查询。只有根昵称（标识位于 XML 文档顶层的元素的昵称）的各列才接受此选项。指定的数可以是整数或小数值。缺省值为 7 毫秒。

### **XPath\_EVAL\_TIME**

指定对昵称的 XPath 表达式进行求值并找到第一个元素所需要的时间（以毫秒计）。可以修改 XPath\_EVAL\_TIME、INSTANCE\_PARSE\_TIME 和 NEXT\_TIME 选项以优化大型的或复杂的 XML 源结构的查询。根昵称和非根昵称都接受此选项。指定的数可以是整数或小数值。缺省值为 1 毫秒。

### **NEXT\_TIME**

指定从 XPath 表达式中找到后续的源元素所需要的时间（以毫秒计）。可以修改 NEXT\_TIME、XPath\_EVAL\_TIME 和 INSTANCE\_PARSE\_TIME 选项以优化大型的或复杂的 XML 源结构的查询。根昵称和非根昵称都接受此选项。缺省值为 1 毫秒。

#### **用法注释:**

如果使用 DATE 数据类型选项，则 XML 源文档中的日期必须采用以下格式：CCYY-MM-DD。例如：如果日期是 2002 年 11 月 17 日，则在 XML 源文档中必须将该日期指定为：2002-11-17。如果日期采用任何其它格式，将接收到错误消息。

不要将 STREAMING 参数和 VALIDATE 参数都设置为 YES。XML 包装器将验证整个 XML 源文档，而不会验证源文档分段。如果将它们都设置为 YES，将接收到错误消息。

#### **相关任务:**

- 第 394 页的『为 XML 数据源注册昵称』

#### **相关参考:**

- 『CREATE NICKNAME statement』（*SQL Reference, Volume 2*）
- 第 394 页的『CREATE NICKNAME 语句 - XML 包装器的示例』

## 附录 A. 全局目录表中包含联合信息的视图

联合数据库中的大多数目录视图与任何其它 DB2 Linux 版、UNIX 版和 Windows 版数据库中的目录视图相同。有几个唯一视图（例如，SYSCAT.WRAPPERS 视图）包含与联合系统相关的信息。

如 DB2 Linux 版、UNIX 版以及 Windows 版版本 6 和版本 7 SQL 参考手册中所述，DB2 版本 8 SYSCAT 视图现在是只读的。如果对 SYSCAT 模式中的视图发出 UPDATE 或 INSERT 操作，则该操作将失败。使用 SYSSTAT 视图是更新系统目录的建议方法。将引用 SYSCAT 视图的应用程序更改为引用可更新的 SYSSTAT 视图。

下表列示包含联合信息的 SYSCAT 视图。这些视图是只读视图。

表 138. 通常与联合系统配合使用的目录视图

目录视图	描述
SYSCAT.CHECKS	包含您定义的检查约束信息。
SYSCAT.COLCHECKS	包含检查约束所引用的列。
SYSCAT.COLUMNS	包含有关已为其创建昵称的数据源对象（表和视图）的列信息。
SYSCAT.COLOPTIONS	包含有关为昵称设置的列选项值的信息。
SYSCAT.CONSTDEP	包含您定义的信息约束的相关性。
SYSCAT.DATATYPES	包含有关本地内置和用户定义的 DB2 数据类型的数据类型信息。
SYSCAT.DBAUTH	包含个别用户和组所拥有的数据库权限。
SYSCAT.FUNCMAPOPTIONS	包含有关您已为函数映射设置的选项值的信息。
SYSCAT.FUNCMAPPINGS	包含联合数据库与数据源对象之间的函数映射。
SYSCAT.INDEXCOLUSE	包含参与索引的列。
SYSCAT.INDEXES	包含数据源对象的索引规范。
SYSCAT.KEYCOLUSE	包含参与由唯一键、主键或外键约束所定义的键的列。
SYSCAT.REFERENCES	包含关于您定义的引用约束的信息。
SYSCAT.ROUTINES	包含本地 DB2 用户定义的函数或函数模板。函数模板用来映射至数据源函数。
SYSCAT.REVTYPE MAPPINGS	不使用此视图。所有数据类型映射在 SYSCAT.TYPE MAPPINGS 视图中记录。
SYSCAT.SERVEROPTIONS	包含有关使用服务器定义设置的服务器选项值的信息。
SYSCAT.SERVERS	包含为数据源服务器创建的服务器定义。
SYSCAT.TABCONST	每一行表示 CHECK、UNIQUE、PRIMARY KEY 或 FOREIGN KEY 类型的表和昵称约束。
SYSCAT.TABLES	包含有关您创建的每个本地 DB2 表、联合视图和昵称的信息。

表 138. 通常与联合系统配合使用的目录视图 (续)

目录视图	描述
SYSCAT.TYPEMAPPINGS	包含正向数据类型映射和反向数据类型映射。该映射是从数据源数据类型至本地 DB2 数据类型的映射。对数据源对象创建昵称时，使用这些映射。
SYSCAT.USEROPTIONS	包含您在创建联合数据库与数据源服务器之间的用户映射时设置的用户权限信息。
SYSCAT.VIEWS	包含有关您创建的本地联合视图的信息。
SYSCAT.WRAPOPTIONS	包含有关您为包装器设置的选项值的信息。
SYSCAT.WRAPPERS	包含您为其创建包装器的每个数据源的包装器和库文件的名称。

下表列示包含联合信息的 SYSSTAT 视图。这些视图是包含您可以更新的统计信息的读 / 写视图。

表 139. 联合可更新全局目录视图

目录视图	描述
SYSSTAT.COLUMNS	包含有关已为其创建昵称的数据源对象（表和视图）中的每一列的统计信息。不记录类型表的继承列的统计信息。
SYSSTAT.INDEXES	包含有关数据源对象的每个索引规范的统计信息。
SYSSTAT.ROUTINES	包含有关每个用户定义的函数的统计信息。不包括内置函数。不记录类型表的继承列的统计信息。
SYSSTAT.TABLES	包含有关每个基本表的信息。视图、同义词和别名信息不包括在此视图中。对于类型表，只有表层次结构的根表才包括在视图中。不记录类型表的继承列的统计信息。



## 附录 B. 联合系统的包装器选项

包装器选项用于配置包装器或定义联合服务器使用包装器的方式。可以在创建或改变包装器时设置包装器选项。

所有关系和非关系数据源都使用 DB2\_FENCED 包装器选项。ODBC 数据源使用 MODULE 包装器选项。Entrez 数据源使用 EMAIL 包装器选项。

表 140. 包装器选项及其设置

选项	有效设置	缺省设置
DB2_FENCED	指定包装器是以设防方式还是以可信方式运行。 Y 包装器以设防方式运行。 N 包装器以可信方式运行。	关系型包装器: N。 IBM 的非关系型包装器: N。 第三方的非关系型包装器: Y。
EMAIL	在注册 Entrez 包装器时指定一个电子邮件地址。所有查询都包含此电子邮件地址, 且允许 NCBI 在有问(例如, 过多的查询使 NCBI 服务器重载)时联系您。此选项是必需的。	
MODULE	指定包含 ODBC 驱动程序管理器实现或 SQL/CLI 实现的库的完整路径。对 UNIX 联合服务器上的 ODBC 包装器是必需的。	在 Windows 上, 缺省值为 odbc32.dll。

### 相关概念:

- 『引用昵称的查询的并行性』(《联合系统指南》)

### 相关任务:

- 『可信的和受防护的方式处理环境』(《IBM DB2 Information Integrator 包装器开发者指南》)
- 『改变包装器』(《联合系统指南》)
- 第 54 页的『为数据源注册包装器』



---

## 附录 C. SQL 语句中的有效服务器类型

服务器类型指示服务器定义将表示哪种数据源。服务器类型视供应商、用途和操作系统的不同而有所不同。受支持的值取决于正被使用的包装器。

对于大多数数据源，必须在 CREATE SERVER 语句中指定有效的服务器类型。

---

### BioRS 包装器

BioRS 数据源。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	BioRS

---

### BLAST 包装器

受 BLAST 守护进程支持的 BLAST 数据源。

服务器类型	数据源
BLASTN	BLAST 搜索，核苷酸序列在该搜索中与核苷酸序列数据库的内容进行比较，以查找具有与原始序列区域同源的区域的序列。
BLASTP	BLAST 搜索，氨基酸序列在该搜索中与氨基酸序列数据库的内容进行比较，以查找具有与原始序列区域同源的区域的序列。
BLASTX	BLAST 搜索，核苷酸序列在该搜索中与氨基酸序列数据库的内容进行比较，以查找具有与原始序列区域同源的区域的序列。
TBLASTN	BLAST 搜索，氨基酸序列在该搜索中与核苷酸序列数据库的内容进行比较，以查找具有与原始序列区域同源的区域的序列。
TBLASTX	BLAST 搜索，核苷酸序列在该搜索中与核苷酸序列数据库的内容进行比较，以查找具有与原始序列区域同源的区域的序列。

---

### CTLIB 包装器

受 CTLIB 客户机软件支持的 Sybase 数据源。

服务器类型	数据源
SYBASE	Sybase

## Documentum 包装器

受 Documentum 客户机 API / 库支持的 Documentum 数据源。

服务器类型	数据源
DCTM	Documentum

## DRDA 包装器

### DB2 系列数据源

表 141. DB2 Linux 版、UNIX 版和 Windows 版

服务器类型	数据源
DB2/UDB	IBM DB2 通用数据库
DB2/6000	IBM DB2 AIX 版
DB2/AIX	IBM DB2 AIX 版
DB2/HPUX	IBM DB2 HP-UX 版
DB2/HP	IBM DB2 HP-UX 版
DB2/NT	IBM DB2 Windows NT 版
DB2/EEE	IBM DB2 企业扩展版
DB2/SUN	IBM DB2 Solaris 版
DB2/PE	IBM DB2 个人版
DB2/2	IBM DB2 OS/2 版
DB2/LINUX	IBM DB2 Linux 版
DB2/PTX	IBM DB2 NUMA-Q 版
DB2/SCO	IBM DB2 SCO Unixware 版

表 142. DB2 iSeries 版 (和 AS/400 版)

服务器类型	数据源
DB2/400	IBM DB2 iSeries 和 AS/400 版

表 143. DB2 z/OS 和 OS/390 版

服务器类型	数据源
DB2/ZOS	IBM DB2 z/OS 版
DB2/390	IBM DB2 OS/390 版
DB2/MVS	IBM DB2 MVS 版

表 144. DB2 服务器 VM 和 VSE 版

服务器类型	数据源
DB2/VM	IBM DB2 VM 版
DB2/VSE	IBM DB2 VSE 版

表 144. DB2 服务器 VM 和 VSE 版 (续)

服务器类型	数据源
SQL/DS	IBM SQL/DS

## Entrez 包装器

Entrez 数据源。

服务器类型	数据源
NUCLEOTIDE	Entrez
PUBMED	Entrez

## Excel 包装器

受 Microsoft Excel 97、2000 和 2002 支持的 Excel 数据源。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	Microsoft Excel

## Extended Search 包装器

受“Extended Search 客户机库”支持的 Extended Search 数据源。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	IBM Lotus Extended Search

## HMMER 包装器

受 HMMER 守护进程支持的 HMMER 数据源。

服务器类型	数据源
PFAM	HMMER
SEARCH	HMMER

## Informix 包装器

受 Informix Client SDK 软件支持的 Informix 数据源。

服务器类型	数据源
INFORMIX	Informix

---

## MSSQLODBC3 包装器

受 DataDirect Connect ODBC 3.6 驱动程序或 ODBC 3.0（或更高版本）驱动程序支持的 Microsoft SQL Server 数据源。

服务器类型	数据源
MSSQLSERVER	Microsoft SQL Server

---

## NET8 包装器

受 Oracle NET8 客户机软件支持的 Oracle 数据源。

服务器类型	数据源
ORACLE	Oracle V8.0 或更新版本

---

## ODBC 包装器

受 ODBC 3.x 驱动程序支持的 ODBC 数据源。

服务器类型	数据源
ODBC	ODBC

---

## OLE DB 包装器

与 Microsoft OLE DB 2.0 或更新版本兼容的 OLE DB 提供程序。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	任何 OLE DB 提供程序

---

## 表结构文件包装器

表结构文件数据源。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	表结构文件

---

## Teradata 包装器

受 Teradata V2R3、V2R4 和 V2R5 客户机软件支持的 Teradata 数据源。

服务器类型	数据源
TERADATA	Teradata

---

## Web 服务包装器

Web 服务数据源。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	任何 Web 服务数据源。

---

## WebSphere Business Integration 包装器

受 WeSphere Business Integration 包装器支持的商业应用程序数据源。

服务器类型	数据源
WBI	WebSphere Business Integration 2.2 或 2.3

---

## XML 包装器

XML 数据源。

服务器类型	数据源
在 CREATE SERVER 语句中不需要。	XML





## 附录 D. 联合系统的服务器选项

服务器选项用于描述数据源服务器。服务器选项指定数据完整性、位置、安全性和性能信息。一些服务器选项可用于所有数据源，而另一些服务器选项则是特定于数据源的。

关系数据源的公共联合服务器选项为：

- 兼容性选项 - COLLATING\_SEQUENCE 和 IGNORE\_UDT
- 数据完整性选项 - IUD\_APP\_SVPT\_ENFORCE
- 日期和时间选项 - DATEFORMAT、TIMEFORMAT 和 TIMESTAMPFORMAT
- 位置选项 - CONNECTSTRING、DBNAME 和 IFILE
- 安全性选项 - FOLD\_ID、FOLD\_PW 和 INFORMIX\_LOCK\_MODE
- 性能选项 - COMM\_RATE、CPU\_RATIO、DB2\_MAXIMAL\_PUSHDOWN、IO\_RATIO、LOGIN\_TIMEOUT、PACKET\_SIZE、PLAN\_HINTS、PUSHDOWN、TIMEOUT 和 VARCHAR\_NO\_TRAILING\_BLANKS

下表列出了适用于每个关系数据源的服务器定义服务器选项。

表 145. 关系数据源的服务器选项

数据源	CODEPAGE	COLLATING_SEQUENCE	COMM_RATE	CONNECTSTRING	CPU_RATIO	DATEFORMAT	DB2_MAXIMAL_PUSHDOWN	DBNAME	FOLD_ID	FOLD_PW	IFILE	INFORMIX_LOCK_MODE	IO_RATIO	IUD_APP_SVPT_ENFORCE	LOGIN_TIMEOUT	NODE	PACKET_SIZE	PASSWORD	PLAN_HINTS	PUSHDOWN	TIMEOUT	TIMEFORMAT	TIMESTAMPFORMAT	VARCHAR_NO_TRAILING_BLANKS
DB2 UDB iSeries 版		X	X		X		X	X	X	X			X	X				X		X				X
DB2 UDB z/OS 和 OS/390 版		X	X		X		X	X	X	X			X	X				X		X				X
DB2 VM 和 VSE 版		X	X		X		X	X	X	X			X	X				X		X				X
DB2 UDB Linux 版、 UNIX 版和 Windows 版		X	X		X		X	X	X	X			X	X				X		X				X
Informix		X	X		X		X	X	X	X		X	X	X		X		X		X				
Microsoft SQL Server	X	X	X		X		X	X	X	X			X	X		X		X		X				

表 145. 关系数据源的服务器选项 (续)

数据源	CODEPAGE	COLLATING_SEQUENCE	COMM_RATE	CONNECTSTRING	CPU_RATIO	DATEFORMAT	DB2_MAXIMAL_PUSHDOWN	DBNAME	FOLD_ID	FOLD_PW	IFILE	INFORMIX_LOCK_MODE	IO_RATIO	IUD_APP_SVPT_ENFORCE	LOGIN_TIMEOUT	NODE	PACKET_SIZE	PASSWORD	PLAN_HINTS	PUSHDOWN	TIMEOUT	TIMEFORMAT	TIMESTAMPFORMAT	VARCHAR_NO_TRAILING_BLANKS
ODBC	X	X	X		X	X	X	X	X	X			X	X		X		X		X		X	X	X
OLE DB		X		X																				
Oracle		X	X		X		X	X	X			X			X		X	X	X					X
Sybase		X	X		X		X	X	X	X		X		X	X	X	X	X	X	X	X			
Teradata		X	X		X		X					X	X		X					X				

下表列示了适用于每个非关系数据源（除 WebSphere Business Integration 之外）的服务器定义服务器选项。WebSphere Business Integration 的服务器定义服务器选项列示于第 531 页的表 147 中。

表 146. 非关系数据源的服务器选项。

数据源	CASE_SENSITIVE	CONTENT_DIR	DAEMON_PORT	ES_HOST	ES_PORT	ES_TRACING	ES_TRACELEVEL	ES_TRACEFILENAME	HMMFAM_OPTIONS	HMMSEARCH_OPTIONS	MAX_ROWS	NODE	OS_TYPE	PORT	PROCESSORS	PROXU_AUTHID	PROXY_PASSWORD	PROXY_SERVER_NAME	PROXY_SERVER_PORT	PROXY_TYPE	RDBMS_TYPE	SOCKET_TIMEOUT	TIMEOUT	TRANSACTIONS	USE_CLOB_SEQUENCE
BioRS	X											X		X									X		
BLAST			X									X													X
Documentum		X										X	X								X			X	
Entrez											X					X	X	X	X	X		X			
Excel																									
Extended Search				X	X	X	X	X																	
HMMER			X						X	X		X			X										X
表结构文件																									
Web 服务																									
XML																X	X	X	X	X		X			

下表列示了适用于 WebSphere Business Integration 数据源的服务器定义服务器选项。

表 147. WebSphere Business Integration 数据源的服务器选项。

数据源	APP_TYPE	FAULT_QUEUE	MQ_CONN_NAME	MQ_MANAGER	MQ_RESPONSE_TIMEOUT	MQ_SVRCONN_CHANNELNAME	REQUEST_QUEUE	RESPONSE_QUEUE
WebSphere Business Integration	X	X	X	X	X	X	X	X

下表描述了每个服务器选项并列示了有效设置和缺省设置。

表 148. 服务器选项及其设置

选项	描述和有效设置	缺省设置
APP_TYPE	远程应用程序的类型。有效值为“PSOFT”、“SAP”和“SIEBEL”。此选项是必需的。	无。
CASE_SENSITIVE	指定 BioRS 服务器是否以区分大小写的方式对待名称。有效值为 Y 或 N。  ‘Y’ BioRS 服务器以区分大小写的方式对待名称。  ‘N’ BioRS 服务器不以区分大小写的方式对待名称。	Y

在 BioRS 产品中，一个配置参数控制存储在 BioRS 服务器上的数据是否区分大小写。CASE\_SENSITIVE 选项是该 BioRS 系统配置参数的 DB2 Information Integrator 对应物。必须将 BioRS 系统和 DB2 Information Integrator 中的 BioRS 服务器区分大小写配置设置同步。如果不使 BioRS 和 DB2 Information Integrator 之间的区分大小写配置设置保持同步，则当您尝试通过 DB2 Information Integrator 存取 BioRS 数据时将会出错。

当在 DB2 Information Integrator 中创建了新的 BioRS 服务器之后，不能更改或删除 CASE\_SENSITIVE 选项。如果需要更改 CASE\_SENSITIVE 选项，则必须删除然后再次创建整个服务器。如果删除 BioRS 服务器，还必须再次创建所有相应的 BioRS 昵称。DB2 Information Integrator 会自动删除与已删除的服务器对应的所有昵称。

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
CODEPAGE	指定与数据源客户机配置的编码字符集相对应的 DB2 代码页标识。如果客户机的代码页和联合数据库代码页不匹配，则必须指定客户机的代码页。  对于支持 Unicode 的数据源，CODEPAGE 选项可以设置为与数据源客户机的受支持 Unicode 编码相对应的 DB2 代码页标识。	在具有非 Unicode 联合数据库的 UNIX 或 Windows 系统上：为联合数据库代码页。  在具有 Unicode 联合数据库的 UNIX 系统上：为 1208  在具有 Unicode 联合数据库的 Windows 系统上：为 1202
COLLATING_SEQUENCE	指定数据源是否根据 NLS 代码集和国家 / 地区信息与联合数据库使用相同的缺省整理顺序。  'Y' 数据源与 DB2 联合数据库使用相同的整理顺序。  'N' 数据源的整理顺序与 DB2 联合数据库整理顺序不同。  'I' 数据源的整理顺序与 DB2 联合数据库整理顺序不同，并且数据源整理顺序不区分大小写（例如，“STEWART”与“StewART”被认为是相同的）。	'N'
COMM_RATE	指定联合服务器与数据源服务器之间的通信速率。以“兆字节 / 秒”表达该速率。  有效值为大于 0 而小于 $1 \times 10^{23}$ 。值可以用任何有效的 REAL 表示法来表示。	'2'
CONTENT_DIR	指定可本地存取的根目录的名称，以存储由 GET_FILE、GET_FILE_DEL、GET_RENDITION 和 GET_RENDITION_DEL 伪列检索的内容文件。它必须对可以使用这些伪列的所有用户为可写的。	在 UNIX 系统上：为“/tmp”  在 Windows 系统上：为“C:\temp”
CONNECTSTRING	指定连接至 OLE DB 提供程序所需的初始化属性。	无。
CPU_RATIO	表示数据源 CPU 运行速度比联合服务器 CPU 运行速度快或慢多少。  有效值为大于 0 而小于 $1 \times 10^{23}$ 。值可以用任何有效的 REAL 表示法来表示。  设置为 1 指示 DB2 联合 CPU 速度与数据源 CPU 速度相同，比率为 1:1。设置为 .5 指示 DB2 联合 CPU 速度比数据源 CPU 速度慢 50%。设置为 2 指示 DB2 联合 CPU 速度是数据源 CPU 速度的 2 倍。	'1.0'
DATEFORMAT	数据源使用的日期格式。使用“DD”、“MM”和“YY”或“YYYY”输入格式来表示日期的数字格式。还应指定定界符（例如，空格或逗号）。例如，要表示“2003-01-01”的日期格式，使用“YYYY-MM-DD”。此字段可空。	无。
DAEMON_PORT	指定守护进程要在其上侦听 BLAST 或 HMMER 作业请求的端口号。此端口号必须与在守护进程配置文件的 DAEMON_PORT 选项中指定的端口号相同。	BLAST: “4007” ; HMMER: “4098”

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
DB2_MAXIMAL_PUSHDOWN	指定选择存取方案时查询优化器使用的主要条件。查询优化器可根据成本或根据用户需求（由远程数据源尽可能多地执行查询处理）来选择存取方案。	'N'
	'Y' 查询优化器选择与其它存取方案相比将更多查询操作下推到数据源的存取方案。当几个存取方案提供的下推量相同时，查询优化器会选择成本最低的方案。	
	如果联合服务器上的具体查询表（MQT）可以处理部分或全部查询，则可使用包括具体查询表的存取方案。联合数据库不会下推将导致笛卡儿积的查询。	
	'N' 查询优化器根据成本选择存取方案。	
DBNAME	您想要联合服务器存取的数据源数据库的名称。对于 DB2 数据库，此值与初始远程 DB2 数据库连接的特定数据库相对应。此特定的数据库是使用 CATALOG DATABASE 命令或 DB2 配置助手在联合服务器进行编目的远程 DB2 数据库的数据库别名。此值不适用于 Oracle 数据源，原因是 Oracle 实例只包含一个数据库。	无。
ES_HOST	指定要搜索的 Extended Search 服务器的标准主机名或 IP 地址。此选项是必需的。	无。
ES_PORT	指定端口号，此 Extended Search 服务器要在该端口号侦听请求。此选项是可选的。	"6001"
ES_TRACING	指定是否对由远程 Extended Search 服务器产生的错误消息、警告消息和参考消息启用跟踪。有效值为：	"OFF"
	"OFF" 不记录跟踪消息。	
	"ON" 记录跟踪消息。	
	此选项是可选的。	
ES_TRACELEVEL	如果启用了跟踪，此选项指定将写入日志文件的消息类型。可以分别启用和禁用下列跟踪级别：	'C'
	'C' 重要错误消息。	
	'N' 非重要消息。	
	'W' 警告消息。	
	'I' 参考消息。	
	例如：	
	ES_TRACELEVEL 'W'	
	ES_TRACELEVEL 'CN'	
	此选项是可选的。	

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
ES_TRACEFILENAME	如果启用了跟踪, 此选项指定消息将写入到的目录和文件的名称。此选项是可选的。	对于 UNIX 操作系统: 为 \$INSTHOME/sqllib/log/ESWrapper.log。  对于 Windows 操作系统: 为 %DB2TEMPDIR%\ESWrapper.log。
FAULT_QUEUE	从适配器到包装器传递错误消息的故障队列的名称。此名称必须符合 WebSphere MQ 的队列名称的规范。这是必需的选项。	无。
FOLD_ID (请参阅此表末的注 1 和 4。)	适用于联合服务器发送至数据源服务器以进行认证的用户标识。有效值为:  'U' 联合服务器在将用户标识发送至数据源之前将该标识转变为大写形式。对于 DB2 系列和 Oracle 数据源, 这是一个逻辑选项。(请参阅此表末的注 2)  'N' 联合服务器在将用户标识发送至数据源之前不对该标识执行任何操作。(请参阅此表末的注 2。)  'L' 联合服务器在将用户标识发送至数据源之前将该标识转变为小写形式。  如果未使用任何这些设置, 联合服务器会尝试以大写形式将用户标识发送至数据源。如果该用户标识无效, 则服务器尝试以小写形式发送它。	无。
FOLD_PW (请参阅此表末的注 1、3 和 4。)	适用于联合服务器发送至数据源以进行认证的密码。有效值为:  'U' 联合服务器在将密码发送至数据源之前将该密码转变为大写形式。对于 DB2 系列和 Oracle 数据源, 这是一个逻辑选项。  'N' 联合服务器在将密码发送至数据源之前不对该密码执行任何操作。  'L' 联合服务器在将密码发送至数据源之前将该密码转变为小写形式。  如果未使用任何这些设置, 联合服务器会尝试以大写形式将密码发送至数据源。如果该密码无效, 则服务器尝试以小写形式发送它。	无。

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
HMPFAM_OPTIONS	<p>指定 hmmpfam 选项, 例如 --null2、--pvm 和 --xnu, 它们在将选项映射到列名的引用表中没有相应的列名。</p> <p>例如:</p> <pre>HMPFAM_OPTIONS '--xnu --pvm'</pre> <p>在此示例中, 守护进程通过查询的 WHERE 子句中的选项以及其它选项 --xnu --pvm 来运行 HMPFAM 程序。</p>	
HMMSEARCH_OPTIONS	<p>允许用户向 hmmsearch 命令提供更多命令行选项。仅对类型 SEARCH 有效。有关更多信息, 请参阅《HMMER 用户指南》。</p>	无。
IFILE	<p>指定 Sybase Open Client interfaces 文件的路径和名称。在 Windows NT 联合服务器上, 缺省值为 %DB2PATH%\interfaces。在 UNIX 联合服务器上, 缺省的路径和名称值为 \$DB2INSTANCE/sql/lib/interfaces。</p>	无。
INFORMIX_LOCK_MODE	<p>指定要为 Informix 数据源设置的锁定方式。Informix 包装器在建立与 Informix 数据源的连接之后立即发出“SET LOCK MODE”命令。有效值为:</p> <p><b>'W'</b> 将 Informix 锁定方式设置为 WAIT。如果包装器尝试存取锁定的表或行, 则 Informix 将一直等到锁定释放为止。</p> <p><b>'N'</b> 将 Informix 锁定方式设置为 NOWAIT。如果包装器尝试存取锁定的表或行, 则 Informix 返回错误。</p> <p><b>'n'</b> 将 Informix 锁定方式设置为 WAIT <i>n</i> 秒。如果包装器尝试存取锁定的表或行而锁定未在指定秒数内释放, 则 Informix 返回错误。</p>	'W'
IO_RATIO	<p>表示数据源 I/O 系统的运行速度比联合服务器 I/O 系统的运行速度快或慢多少。</p> <p>有效值为大于 0 而小于 <math>1 \times 10^{23}</math>。值可以用任何有效的 REAL 表示法来表示。</p> <p>设置为 1 指示 DB2 联合 I/O 速度与数据源 I/O 速度相同, 比率为 1:1。设置为 .5 指示 DB2 联合 I/O 速度比数据源 I/O 速度慢 50%。设置为 2 指示 DB2 联合 I/O 速度是数据源 I/O 速度的 2 倍。</p>	'1.0'

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
IUD_APP_SVPT_ENFORCE	指定 DB2 联合系统是否应该强制检测或构建应用程序保存点语句。当使用 SET SERVER OPTION 语句进行设置时，此服务器选项对静态 SQL 语句没有任何影响。  'Y' 如果在插入、更新或删除操作中发生错误且数据源没有强制执行应用程序保存点语句，则联合服务器将回滚插入、更新或删除事务。返回 SQL 错误代码 SQL1476N。  'N' 遇到错误时，联合服务器将不回滚事务。应用程序必须处理错误恢复。	'Y'
LOGIN_TIMEOUT	指定 DB2 联合服务器等待来自 Sybase Open Client 的对登录请求的响应的秒数。缺省值与 TIMEOUT 的缺省值相同。	'0'
MAX_ROWS	指定联合服务器为使用 Entrez 包装器的查询返回的行数。  只能指定正数和零。当将此选项设置为零时，则使查询可从 NCBI Web 站点检索的行数不受限制。然而，将 MAX_ROWS 服务器选项设置为零或很高的数值可能会影响您的查询性能。  MAX_ROWS 服务器选项不是必需的。	Microsoft Windows 操作系统: 2000 行。  基于 UNIX 的操作系统: 5000 行。
MQ_CONN_NAME	Websphere MQ 服务器在其上运行的计算机的主机名或网络地址。连接名称的一个示例为: 9.30.76.151(1420)，其中 1420 为端口号。如果不包括端口号，则使用缺省值 1414。此选项是可选的。如果省略此选项，则 MQSERVER 环境变量（如果已在 db2dj.ini 文件中指定）用于选择通道定义。如果未设置 MQSERVER，则使用客户机通道表。	包装器使用 MQSERVER 环境变量（如果已在 db2dj.ini 文件中指定）来选择通道定义。如果未设置 MQSERVER 环境变量，则包装器使用客户机通道表。
MQ_MANAGER	WebSphere MQ 管理器名称。任何有效的 WebSphere MQ 管理器名称。此选项是必需的。	无。
MQ_RESPONSE_TIMEOUT	包装器等待响应队列的响应消息的时间量。此值以毫秒为单位。可以指定特殊值 -1，表明没有超时时间段。此选项是可选的。	10000
MQ_SVRCONN_CHANNELNAME	Websphere MQ Manager 上的服务器连接通道的名称，包装器将尝试连接到此通道。仅当指定了 MQ_CONN_NAME 服务器选项时才能指定此参数。如果省略此选项，则使用缺省的服务器连接通道 SYSTEM.DEF.SVRCONN。	SYSTEM.DEF.SVRCONN



表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
NODE	<p>关系数据源: 数据源被定义为其 RDBMS 的实例时所使用的名称。</p> <p>Documentum: 指定 Documentum Docbase 的实际名称。此选项是必需的。</p> <p>BLAST: 指定 BLAST 守护进程运行于的系统的主机名。此选项是必需的。</p> <p>HMMER: 指定 HMMER 守护进程运行于的服务器的主机名。此选项是必需的。</p> <p>BioRS: 指定系统的主机名, BioRS 查询工具在此系统上可用。此选项是可选的。</p>	BioRS: <i>localhost</i>
OS_TYPE	指定 Docbase 服务器的操作系统。有效值为 无、AIX、SOLARIS 和 WINDOWS。此选项是必需的。	
PACKET_SIZE	指定 Sybase interfaces 文件的信息包大小 (以字节计)。如果数据源不支持指定信息包大小, 则连接将失败。当每个记录都很大时 (例如, 当将行插入到大表中时) 增大信息包会显著提高性能。字节大小为一个数字值。	
PASSWORD	<p>指定是否将密码发送到数据源。</p> <p>'Y' 将密码发送到数据源并进行验证。</p> <p>'N' 密码不发送到数据源且不进行验证。</p>	'Y'
PLAN_HINTS	<p>指定是否启用方案提示。方案提示是为数据源优化器提供额外信息的语句段。对于某些查询类型, 此信息可以提高查询性能。方案提示可帮助数据源优化器决定是否使用索引、要使用哪个索引以及要使用哪个表连接顺序。</p> <p>'Y' 如果数据源支持方案提示, 则将在该数据源上启用方案提示。</p> <p>'N' 不会在数据源上启用方案提示。</p> <p>此选项只对于 Oracle 和 Sybase 数据源才可用。</p>	'N'
PORT	指定包装器用来连接到 BioRS 服务器的端口号。此选项是可选的。	"5014"
PROCESSORS	指定 HMMER 程序使用的处理器数。此选项等效于 无。hmpfam 命令的 --cpu 选项。	
PROXY_AUTHID	指定当 PROXY_TYPE 值为 "SOCKS5" 时使用的用户名。如果 PROXY_TYPE 值为 "SOCKS5", 则此字段为可选的。请联系网络管理员以获取要使用的用户名。如果 PROXY_TYPE 不为 "SOCKS5", 则此选项无效。	无。

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
PROXY_PASSWORD	指定当 PROXY_TYPE 值为 “SOCKS5” 时使用的密码。如果 PROXY_TYPE 值为 “SOCKS5”，则此字段为可选的。请联系网络管理员以获取要使用的密码。如果 PROXY_TYPE 不为 “SOCKS5”，则此选项无效。	无。
PROXY_SERVER_NAME	指定代理服务器名称或 IP 地址。如果 PROXY_TYPE 值为 “HTTP”、“SOCKS4” 或 “SOCKS5”，则此字段是必需的。请联系网络管理员以获取代理服务器名称或 IP 地址。	无。
PROXY_SERVER_PORT	指定代理服务器端口号。如果 PROXY_TYPE 值为 “HTTP”、“SOCKS4” 或 “SOCKS5”，则此字段是必需的。请联系网络管理员以获取要使用的代理服务器端口号。	无。
PROXY_TYPE	指定当处于防火墙之后时用来访问因特网的代理类型。有效值为 “NONE”、“HTTP”、“SOCKS4” 或 “SOCKS5”。缺省值为 “NONE”。请联系网络管理员以获取使用的代理类型。	“NONE”
PUSHDOWN	<p>‘Y’ DB2 UDB 将考虑让数据源对运算求值。</p> <p>‘N’ DB2 UDB 将发送仅包含带有列名的 SELECT 的数据源 SQL 语句。谓词（例如，WHERE=）列和标量函数（例如，MAX 和 MIN）、排序（例如，ORDER BY 或 GROUP BY）以及连接将不包括在发送至数据源的任何 SQL 中。</p>	‘Y’
RDBMS_TYPE	指定由 Docbase 使用的 RDBMS。有效值为 DB2、INFORMIX、ORACLE、SQLSERVER 或 SYBASE。此选项是必需的。	无。
RESPONSE_QUEUE	从适配器到包装器传递查询结果的响应队列的名称。此名称必须符合 WebSphere MQ 的队列名称的规范。此选项是必需的。	无。
REQUEST_QUEUE	从包装器到适配器传递查询请求的请求队列的名称。此名称必须符合 WebSphere MQ 的队列名称的规范。此选项是必需的。	无。
SOCKET_TIMEOUT	指定 DB2 联合服务器从代理服务器等待结果的最大时间（以分钟为单位）。有效值为大于或等于零的任何数。缺省值为零 “0”。零值表示等待的时间不受限制。	0
TIMEFORMAT	数据源使用的时间格式。使用 “hh12”、“hh24”、“mm”、“ss”、“AM” 或 “A.M” 输入格式。例如，要表示 “16:00:00” 的时间格式，使用 “hh24:mm:ss”。要表示 “8:00:00 AM” 的时间格式，使用 “hh12:mm:ss AM”。此字段可空。	无。

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
TIMESTAMPFORMAT	数据源使用的时间戳记格式。该格式沿用表示日期和时间的格式，但加上“n”表示秒的十分之几、“nn”表示秒的百分之几、“nnn”表示毫秒，依此类推，一直到“nnnnnn”表示微秒。例如，要表示“2003-01-01-24:00:00.000000”的时间戳记格式，使用“YYYY-MM-DD-hh24:mm:ss.nnnnnn”。此字段可空。	无。
TIMEOUT	<p>Sybase: 指定 DB2 联合服务器将等待来自 Sybase Open Client 的针对任何 SQL 语句的响应的秒数。秒数的值是一个在 DB2 通用数据库的整数范围内的正整数。您指定的超时值取决于您正在使用哪个包装器。Sybase 包装器的 TIMEOUT 选项的缺省行为是 0，这导致 DB2 UDB 无限等待响应。</p> <p>BioRS: 指定 BioRS 包装器将等待来自 BioRS 服务器的响应的的时间（以分钟为单位）。缺省值为 10。此选项是可选的。</p>	“0”；BioRS: “10”
TRANSACTIONS	<p>指定服务器事务模式。有效值为:</p> <p>“NONE” 未启用任何事务。</p> <p>“QUERY” 仅对 Dctm_Query 方法启用事务。</p> <p>“ALL” 对 Dctm_Query 方法启用事务。ALL 与此发行版中的 QUERY 具有相同的功能。</p>	“QUERY”
USE_CLOB_SEQUENCE	此选项指定联合服务器用于 BlastSeq 或 HmmQSeq 列的数据类型。值可以为 'Y' 或 'N'。可以使用 CREATE NICKNAME 或 ALTER NICKNAME 语句。覆盖 BlastSeq 或 HmmQSeq 列的缺省数据类型。	

表 148. 服务器选项及其设置 (续)

选项	描述和有效设置	缺省设置
VARCHAR_NO_ TRAILING_BLANKS	<p>此选项适用于具有变量字符数据类型的数据源，该数据类型在比较期间不会用尾随空格来填充长度。</p> <p>一些数据源（例如，Oracle）不具有空白填充字符比较语义，该语义与 DB2 Linux 版、UNIX 版和 Windows 版比较语义返回的结果相同。当想要此选项适用于将从指定服务器存取的数据源对象中的所有 VARCHAR 和 VARCHAR2 列时，就设置此选项。这包括视图。</p> <p>Y 这些 VARCHAR 列不存在尾随空格，或者数据源具有类似于联合服务器上的语义的空白填充字符比较语义。</p> <p>联合服务器将字符比较操作下推到数据源以进行处理。</p> <p>N 尾随空格在这些 VARCHAR 列中存在，并且数据源具有不同于联合服务器的空白填充字符比较语义。</p> <p>如果不能补偿对等语义，联合服务器将处理字符比较操作。例如，重写谓词。</p>	对于受影响的数据源为 N。

有关此表的注释:

1. 不论为认证指定的值如何，都应用此字段。
2. 因为 DB2 UDB 以大写形式存储用户标识，所以值 ‘N’ 和 ‘U’ 在逻辑上互相同。
3. 当密码的设置为 ‘N’ 时，FOLD\_PW 的设置不起任何作用。因为没有发送任何密码，所以也不用考虑大小写的情况。
4. 避免对这三个选项的任何一个进行空设置。空设置可能看起来很有吸引力，原因是 DB2 UDB 将进行多次尝试来解析用户标识和密码；但是，性能可能会受到损害（在成功传递数据源认证之前，DB2 UDB 有可能会发送用户标识和密码四次）。

相关概念:

- 『影响下推机会的服务器特征』（《联合系统指南》）
- 『影响全局优化的服务器特征』（《联合系统指南》）

相关任务:

- 第 55 页的『为数据源注册服务器定义』

相关参考:

- 『DROP statement』（*SQL Reference, Volume 2*）
- 『ALTER SERVER statement』（*SQL Reference, Volume 2*）
- 『CREATE SERVER statement』（*SQL Reference, Volume 2*）

## 附录 E. 联合系统的用户映射选项

这些选项对所有关系数据源有效。对于非关系数据源，REMOTE\_AUTHID 和 REMOTE\_PASSWORD 选项对以下数据源有效：BioRS、Documentum、Extended Search 和 Web 服务。GUEST 选项对 BioRS 数据源有效。

这些选项与 CREATE USER MAPPING 和 ALTER USER MAPPING 语句配合使用。

表 149. 用户映射选项及其设置

选项	有效设置	缺省设置
ACCOUNTING	DRDA: 用来指定 DRDA 记帐字符串。有效设置包括长度不超过 255 的任何字符串。仅当需要传递记帐信息时，此选项才是必需的。请参阅《DB2 Connect 用户指南》以了解更多信息。	无
GUEST	指定包装器是否对 BioRS 服务器使用来宾访问方式。  <b>Y</b> 包装器对 BioRS 服务器使用来宾访问方式。 <b>N</b> 包装器不对 BioRS 服务器使用来宾访问方式。  当设置为值 Y 时，此选项与 REMOTE_AUTHID 选项和 REMOTE_PASSWORD 选项是互斥的。	N
REMOTE_AUTHID	指示在数据源上使用的授权标识。有效设置包括长度不超过 255 的任何字符串。	用来连接到 DB2 通用数据库的授权标识。
REMOTE_DOMAIN	Documentum: 指示用来认证连接至 Documentum 数据源的用户 Windows NT 域。有效设置包括任何有效的 Windows NT 域名。	Documentum 数据库的缺省认证域。
REMOTE_PASSWORD	指示在数据源上使用的授权密码。有效设置包括长度不超过 32 的任何字符串。  如果满足以下条件，则不需要设置此选项： <ul style="list-style-type: none"><li>数据库管理器配置参数 AUTHENTICATON 设置为 SERVER。</li><li>连接至 DB2 数据库时，指定了授权标识和密码。</li></ul> 如果服务器要求密码且您未设置此选项，则必须确保上述两个条件，否则连接将失败。	如果满足列示于有效设置列中的两个条件，用来连接到 DB2 通用数据库的密码。

### 相关概念:

- 『DB2 Connect 和 DRDA』（《DB2 Connect 用户指南》）
- 『DRDA 和数据存取』（《DB2 Connect 用户指南》）

### 相关任务:

- 第 56 页的『为数据源注册用户映射』



## 附录 F. 联合系统的昵称选项

表 150 和表 151 列示了每个数据源的昵称选项。第 544 页的表 152 描述了每个昵称选项并列示了有效设置和缺省设置。

表 150. 可用的昵称选项 - 从 A 到 P

数据源	ALL_VERSIONS	APPLICATIONID	BUSOBJ_NAME	CATEGORY	COLUMN_DELIMITER	DATASOURCE	DIRECTORY_PATH	FILE_PATH	FOLDERS	HMMTYPE	INSTANCE_PARSE_TIME	IS_REG_TABLE	KEY_COLUMN	MAXHIT	NAMESPACES	NEXT_TIME	PARENT	PROCESSORS
BioRS																		
BLAST						X												X
Documentum	X								X			X						
Entrez																	X	
Excel								X										
Extended Search		X		X		X							X					
HMMER						X				X								
表结构文件					X			X					X					
Web 服务															X			
WebSphere Business Integration			X												X			
XML							X	X			X					X		

表 151 列示了每个数据源的昵称选项（从 R 到 X）。

表 151. 可用的昵称选项 - 从 R 到 X

数据源	RANGE	REMOTE_OBJECT	SOAPACTION	SORTED	SORTFIELD	SORTORDER	STREAMING	TEMPLATE	TOTALMAXHIT	TIMEOUT	URL	VALIDATE	VALIDATE_DATA_FILE	VERTICAL_TABLE	XPATH	XPATH_EVAL_TIME
BioRS		X														
BLAST										X						
Documentum		X														
Entrez		X														
Excel	X															
Extended Search					X	X			X	X				X		

表 151. 可用的昵称选项 - 从 R 到 X (续)

数据源	RANGE	REMOTE_OBJECT	SOAPACTION	SORTED	SORTFIELD	SORTORDER	STREAMING	TEMPLATE	TOTALMAXHIT	TIMEOUT	URL	VALIDATE	VALIDATE_DATA_FILE	VERTICAL_TABLE	XPATH	XPATH_EVAL_TIME
HMMER										X						
表结构文件				X									X			
Web 服务			X				X	X			X				X	
WebSphere Business Integration								X							X	
XML							X					X			X	X

表 152 描述了每个昵称选项并列示了有效设置和缺省设置。

表 152. 昵称选项及其设置

选项	描述和有效设置	缺省设置
ALL_VERSIONS	指定是否搜索所有对象版本。有效值为 y、Y、n 和 N。缺省值 N 是指查询处理中只包含当前对象版本。当 IS_REG_TABLE = 'Y' 时，此选项无效。	N
APPLICATIONID	指定要搜索的 Extended Search 应用程序的名称。该名称必须存在于 Extended Search 配置数据库中。此选项是必需的。	
BUSOBJ_NAME	代表业务对象的 XML 模式定义文件 (.xsd) 的名称。例如 sap_bapi_customer_get_detail2。必须在父昵称中指定此选项。	
CATEGORY	指定要搜索的一个或多个 Extended Search 类别。如果省略此选项，则必须至少指定一个数据源名称。要指定多个类别，需用分号分隔类别名称。	
COLUMN_DELIMITER	用于分隔表结构文件的列的定界符，括在单引号中。列定界符的长度可以大于一个字符。如果未定义列定界符，则缺省定界符为逗号。单引号不能用作定界符。在整个文件中，列定界符必须要保持一致。如果行尾为 NULL 字段，空值可以由两个相邻的定界符或定界符后跟行终结符来表示。列定界符不能作为列的有效数据存在。	缺省的定界符为逗号。



表 152. 昵称选项及其设置 (续)

选项	描述和有效设置	缺省设置
DATASOURCE	对于 Extended Search: 指定要搜索的一个或多个 Extended Search 数据源。如果省略此选项, 则必须指定至少一个类别名称。要指定多个数据源, 需要用分号分隔数据源名称。	
	对于 BLAST: 指要在其上运行 BLAST 搜索的数据源名称。此处使用的相同字符串必须存在于 BLAST 守护进程的配置文件中。此选项是必需的。	
	对于 HMMER (类型 PFAM): 指将由 HMMPFAM 搜索的 HMM 概要文件数据库的名称。此处使用的相同字符串必须存在于 HMMER 守护进程的配置文件中。此选项是必需的。	
	对于 HMMER (类型 SEARCH): 指将由 HMMSEARCH 搜索的序列文件的名称。此处使用的相同字符串必须存在于 HMMER 守护进程的配置文件中。此选项是必需的。	
DIRECTORY_PATH	指定包含一个或多个 XML 文件的目录的路径名。使用该选项以对多个 XML 源文件创建单个昵称。XML 包装器仅使用位于指定目录中的扩展名为 .xml 的文件。XML 包装器将忽略此目录中其它所有文件。如果指定此昵称选项, 则不要指定 DOCUMENT 列。仅对根昵称 (用来识别 XML 文档的顶层元素的昵称) 接受该选项。	
FILE_PATH	对于 Microsoft Excel: 指定要访问的全限定目录路径和 Excel 电子表格的文件名。此选项是必需的。	
	对于表结构文件: 指要访问的表结构文件的全限定路径, 括在单引号中。数据文件必须是标准文件或符号链接, 而不是分隔符或其它非标准文件类型。必须指定 FILE_PATH 或 DOCUMENT 昵称列选项。如果指定了 FILE_PATH 昵称选项, 则不能指定 DOCUMENT 昵称列选项。	
	对于 XML: 指定 XML 文档的文件路径。如果指定此昵称选项, 则不要指定 DOCUMENT 列。仅对根昵称 (用来识别 XML 文档的顶层元素的昵称) 接受此选项。	
FOLDERS	指定包含一个或多个逻辑上组合且语法上正确的 Documentum FOLDER 谓词的字符串。指定 FOLDER 谓词会将此昵称所代表的文档集合限制为指定文件夹中的文档。	
	指定此选项时, 将 FOLDERS 选项的整个值括在单引号中, 并在字符串内用双引号代替单引号。	
	当 IS_REG_TABLE = 'Y' 时, 此选项无效。	
HMMTYPE	可选: 用于模型和基因序列中的字母。值可以为 NUCLEIC 或 PROTEIN, 并且不区分大小写。	PROTEIN

表 152. 昵称选项及其设置 (续)

选项	描述和有效设置	缺省设置
INSTANCE_PARSE_TIME	指定对 XML 源文档中某行内的数据进行解析的时间 (以毫秒为单位)。可以修改 INSTANCE_PARSE_TIME、XPATH_EVAL_TIME 和 NEXT_TIME 选项, 以便对大的或复杂的 XML 源结构的查询进行优化。仅对根昵称 (用来识别 XML 文档的顶层元素的昵称) 的列接受此选项。所指定的数字可以是整数或十进制值。	7
IS_REG_TABLE	指定 REMOTE_OBJECT 选项指定的对象是否是 Documentum 注册表。有效值为 'y'、'Y'、'n' 和 'N'。  不能通过使用 ALTER NICKNAME 语句更改此选项, 从而将昵称从 Documentum 对象更改为注册表 (或相反)。相反, 必须删除并重新创建昵称。	N
KEY_COLUMN	文件中构成关键字 (文件将根据此关键字进行排序) 的列名称, 括在单引号中。仅对排序文件使用此选项。通过 DOCUMENT 昵称列选项指定的列不得指定为键列。  仅支持单列关键字。不允许多列关键字。值必须在 CREATE NICKNAME 语句中定义的列的名称。列必须按升序排序。通过向 nickname 语句中的定义添加 NOT NULL 选项, 键列必须指定为不可空。  此选项区分大小写。然而, DB2 UDB 会将列名称更改为大写, 除非该列是通过双引号定义的。	如果值不是为已排序的昵称指定, 则该值是已指定昵称的文件中第一列的名称。
MAXHIT	一个 INTEGER 值, 用于指定可从正在搜索的每一个源中返回的最大结果数。	50
NAMESPACES	与用于每一列的 XPATH 和 TEMPLATE 选项中的名称前缀相关联的名称空间。语法为:  NAMESPACES 'prefix1="actual_namespace1", prefix2="actual_namespace2" '  用逗号分隔各名称空间。例如:  NAMESPACES ' c="http://www.myweb.com/cust", i="http://www.myweb.com/cust/id", n="http://www.myweb.com/cust/name"'	
NEXT_TIME	指定从 XPath 表达式查找后续的源元素所需的时间 (以毫秒为单位)。可以修改 NEXT_TIME、XPATH_EVAL_TIME 和 INSTANCE_PARSE_TIME 选项, 以便对大的或复杂的 XML 源结构的查询进行优化。根昵称和非根昵称都接受该选项。	1
PARENT	仅对父昵称已通过 REMOTE_OBJECT 选项重命名的子昵称指定此选项。当在 DB2 模式中定义多昵称系列时, PARENT 选项将子昵称与父昵称相关联。此名称区分大小写。	

表 152. 昵称选项及其设置 (续)

选项	描述和有效设置	缺省设置
PROCESSORS	指定将在对 BLAST 查询求值时使用的处理器数。 此选项对应于 blastall-a 选项。	1
RANGE	指定将在数据源中使用的单元格范围。	
REMOTE_OBJECT	<p>对于 BioRS: 指定与昵称相关联的 BioRS 数据银行名称。此名称为昵称确定模式和 BioRS 数据银行。此名称还指定此昵称与其它昵称之间的关系。此名称是否区分大小写取决于 BioRS 服务器是否区分大小写以及 CASE_SENSITIVE 服务器选项的值。不能使用 ALTER NICKNAME 语句来更改或删除此名称。如果此选项中使用的 BioRS 数据银行的名称进行了更改, 则必须删除然后再次创建整个昵称。</p> <p>对于 Documentum: 指定与昵称相关联的 Documentum 对象类型的名称。此名称可以为任何 Documentum 对象类型或注册表。注册表名称必须以表所有者的名称为前缀。如果注册表属于 Docbase 所有者, 则值 dm_dbo 可用于所有者名。此选项是必需的。如果新对象的结构与原始对象的结构不相似, 则使用 ALTER NICKNAME 语句来更改 REMOTE_OBJECT 选项的值会导致出错。</p> <p>对于 Entrez: 指定与昵称相关联的 Entrez 对象类型的名称。此名称确定昵称的模式和 NCBI 数据库, 以及它与其它昵称的关系。此名称区分大小写。</p>	
SOAPACTION	Web 服务描述语言 (WSDL) 格式的 URI SOAPACTION 属性。对于根昵称, 此选项是必需的。对于非根昵称, 不允许此选项。	
SORTED	指定数据源文件是否进行排序。此选项接受 Y、y、n 或 N。  如 LC_COLLATE “本地语言支持” 类别中的设置所定义, 排序的数据源必须根据当前语言环境的整理序列按升序排序。  如果指定数据源进行排序, 请将 VALIDATE_DATA_FILE 选项设置为 Y。	N
SORTFIELD	指定要作为搜索结果的排序依据的字段名。缺省值 DOC_RANK 是 Extended Search 用来确定结果文档的相关性的字段。如果指定了其它的字段名, 则该名称必须存在于您所搜索的源中。	DOC_RANK
SORTORDER	为返回的搜索结果指定排序顺序, 可以为升序 (A) 或降序 (D)。	A

表 152. 昵称选项及其设置 (续)

选项	描述和有效设置	缺省设置
STREAMING	<p>指定是否应该把 XML 源文档划分为逻辑片段以进行处理。片段对应于与昵称的 XPath 表达式匹配的节点。然后包装器逐片段地对 XML 元数据进行解析和处理。这种解析会使内存使用降到最低。仅对根昵称指定此选项。</p> <p>可以对任何 XML 源文档 (FILE、DIRECTORY、URI 或 COLUMN) 指定流式方法。仅对根昵称 (用来识别 XML 文档的顶层元素的昵称) 的列接受该选项。</p> <p>有效值为:</p> <p>Y XML 文档经过解析的。</p> <p>N XML 文档是未经解析。</p> <p>如果将 VALIDATE 参数设置为 YES, 则不要将 STREAMING 参数设置为 YES。如果将这两个参数均设置为 YES, 您将收到一条错误消息。</p>	N
TEMPLATE	<p>对于 WebSphere Business Integration: 用于构造 XML 输入文档的昵称模板片段。片段必须符合指定的模板语法。</p> <p>对于 Web 服务: 用于构造 SOAP 请求的昵称模板片段。片段必须符合指定的模板语法。</p>	
TOTALMAXHIT	<p>一个 INTEGER 值, 用于指定可从正在搜索的所有源中返回的最大结果数。包装器将这些结果组合为一个结果集。</p>	50
TIMEOUT	<p>对于 Extended Search: 是一个 INTEGER 值, 用于指定在请求超时之前等待服务器响应的秒数。</p> <p>对于 BLAST 和 HMMER: 是指包装器等待守护进程的结果的最长时间 (以分钟为单位)。</p>	<p>对于 Extended Search: 此值为 30。</p> <p>对于 BLAST 和 HMMER: 此值为 60。</p>
URL	<p>Web 服务端点的 URL。对于根昵称, 此选项是必需的。对于非根昵称, 不允许此选项。支持的协议为 HTTP 和 HTTPS。</p>	

表 152. 昵称选项及其设置 (续)

选项	描述和有效设置	缺省设置
VALIDATE	<p>指定是否在抽取 XML 数据之前验证 XML 源文档。如果将该选项设置为 YES, 则昵称选项会验证源文档的结构是否符合 XML 模式或文档类型定义 (DTD)。仅对根昵称 (用来识别 XML 文档的顶层元素的昵称) 的列接受此选项。</p> <p>如果 XML 包装器找不到 XML 模式文件或 DTD 文件 (.xsd 或 .dtd), 则不验证 XML 源文档。如果不进行验证, 则 DB2 UDB 不会发出错误消息。因此, 请确保 XML 模式文件或 DTD 文件存在于 XML 源文档中指定的位置。</p> <p>如果将 STREAMING 参数设置为 YES, 则不要将 VALIDATE 参数设置为 YES。如果将两个参数均设置为 YES, 您将收到一条错误消息。</p>	NO
VALIDATE_DATA_FILE	<p>对于排序文件, 此选项指定包装器是否验证键列以升序排序, 并检查空键。此选项的有效值只有 Y 或 N。在注册时执行一次检查。如果 DOCUMENT 昵称列选项用于文件路径, 则不允许此选项。</p>	N
VERTICAL_TABLE	<p>指定搜索结果的表达格式。如果指定 YES, 则除了用户定义的列之外, Extended Search 还会返回配置为可返回的所有字段。包装器将结果作为列名的垂直列表存储于昵称表中。</p>	NO
XPATH	<p>指定 XPATH 表达式, 该表达式用来标识表示独立元组的元素。子昵称的 XPATH 昵称选项在由其父昵称的 XPATH 昵称选项指定的路径上下文中进行求值。此 XPATH 表达式被用作上下文, 以便对由 XPATH 昵称列选项标识的列值求值。</p> <p>对于 XML: 不要在 XPATH 表达式中指定名称空间前缀。XML 包装器不支持名称空间。</p>	
XPATH_EVAL_TIME	<p>指定对昵称的 XPath 表达式求值以及查找第一个元素所需的时间 (以毫秒为单位)。可以修改 XPATH_EVAL_TIME、INSTANCE_PARSE_TIME 和 NEXT_TIME 选项, 以便对大的或复杂的 XML 源结构的查询进行优化。根昵称和非根昵称均接受此选项。指定的数字可以是整数或十进制值。</p>	1



## 附录 G. 联合系统的昵称列选项

可以使用称为昵称列选项的参数在 CREATE NICKNAME 或 ALTER NICKNAME 语句中指定列信息。

下表列出了各数据源的昵称列选项。

表 153. 可用的昵称列选项

数据源	ALL_VALUES	DEFAULT	DELIMITER	DOCUMENT	ESCAPE_INPUT	FOREIGN_KEY	INDEX	IS_REPEATING	NUMERIC_STRING	PRIMARY_KEY	REMOTE_NAME	SOAPACTIONCOLUMN	TEMPLATE	URLCOLUMN	VARCHAR_NO_TRAILING_BLANKS	XPATH
BLAST		X	X				X									
DB2 通用数据库 iSeries 版									X							
DB2 通用数据库 z/OS 和 OS/390 版									X							
DB2 通用数据库 VM 和 VSE 版									X							
DB2 通用数据库 Linux 版、UNIX 版和 Windows 版									X							
Documentum	X		X					X			X					
Informix									X							
Microsoft SQL Server									X							
ODBC									X							
OLE DB																
Oracle									X						X	
Sybase									X							
表结构文件				X												
Teradata									X							
WebSphere Business Integration					X	X				X			X			X
Web 服务					X	X				X		X	X	X		X
XML				X		X				X						X

表 154. 列选项及其设置

选项	描述和有效设置	缺省设置
ALL_VALUES	指定将要返回重复属性的全部值，由指定的定界符隔开。如果缺少此选项或者此选项为 N，则仅返回重复属性的最近使用值。仅能对 IS_REPEATING 选项为 'Y' 的 VARCHAR 列指定 ALL_VALUES 选项（且当 IS_REG_TABLE = 'Y' 时无效）。	
DEFAULT	<p>为以下输入固定列指定新的缺省值：</p> <ul style="list-style-type: none"> <li>• E_value</li> <li>• QueryStrands</li> <li>• GapAlign</li> <li>• NMisMatchPenalty</li> <li>• NMatchReward</li> <li>• Matrix</li> <li>• FilterSequence</li> <li>• NumberOfAlignments</li> <li>• GapCost</li> <li>• ExtendedGapCost</li> <li>• WordSize</li> <li>• ThresholdEx</li> </ul> <p>该新值会覆盖预设置的缺省值。新的缺省值必须与给定列的指示值是相同的类型。</p>	
DELIMITER	<p>对于 Documentum: 当并置重复属性的多个值时，指定要使用的定界符字符串。定界符可以是一个或多个字符。此选项仅对数据类型为 VARCHAR (IS_REPEATING 选项设置为 Y) 的对象的属性有效。</p> <p>对于 BLAST: 是指用来为列（此选项在该列上出现）确定定义行信息的终点的定界字符。如果此选项值中出现多个字符，则其中任何一个字符首次出现意味着字段信息的结束。缺省值为行末尾。此选项是必需的，除非希望最近指定的列包含定义行的剩余部分。</p>	<p>对于 Documentum: 缺省的定界符为逗号。</p> <p>对于 BLAST: 缺省的定界符为行末尾。</p>



表 154. 列选项及其设置 (续)

选项	描述和有效设置	缺省设置
DOCUMENT	<p>对于表结构文件: 指定表结构文件的类型。此包装器仅支持此选项的值 <b>FILE</b>。通过使用 <b>DOCUMENT</b> 选项, 仅能对每个昵称指定一列。与 <b>DOCUMENT</b> 选项相关联的列必须为 <b>VARCHAR</b> 或 <b>CHAR</b> 数据类型。</p> <p>使用 <b>DOCUMENT</b> 昵称列选项 (而不是 <b>FILE_PATH</b> 昵称选项) 表明与昵称相对应的文件将在查询运行时提供。如果 <b>DOCUMENT</b> 选项具有 <b>FILE</b> 值, 则在查询运行时提供的值是模式与此昵称的昵称定义相匹配的文件的完整路径。</p> <p>对于 XML: 指定此列为 <b>DOCUMENT</b> 列。 <b>DOCUMENT</b> 列的值指示在查询运行时提供给昵称的 XML 源数据的类型。仅对根昵称 (标识 XML 文档的顶层元素的昵称) 的列接受此选项。通过使用 <b>DOCUMENT</b> 选项, 仅能对每个昵称指定一列。与 <b>DOCUMENT</b> 选项相关联的列必须为 <b>VARCHAR</b> 数据类型。</p> <p>如果使用 <b>DOCUMENT</b> 列选项 (而不是 <b>FILE_PATH</b> 或 <b>DIRECTORY_PATH</b> 昵称选项), 则与此昵称对应的文档将在查询运行时提供。</p> <p><b>DOCUMENT</b> 选项的有效值为:</p> <p><b>FILE</b> 指定将昵称列的值绑定到文件的路径名。此文件的数据将在查询运行时提供。</p> <p><b>DIRECTORY</b> 指定将昵称列的值绑定到包含多个 XML 数据文件的目录的路径名。多个文件的 XML 数据将在查询运行时提供。此数据位于指定目录路径内的 XML 文件中。XML 包装器仅使用指定的目录中扩展名为 <b>.xml</b> 的文件。XML 包装器将忽略此目录中的所有其它文件。</p> <p><b>URI</b> 指定将昵称列的值绑定到 <b>URI</b> 所引用的远程 XML 文件的路径名。 <b>URI</b> 地址指示此 XML 文件在 Web 上的远程位置。</p> <p><b>COLUMN</b> 指定 XML relational 文档存储在关系列。</p>	
ELEMENT_NAME	<p>指定 <b>BioRS</b> 元素名称。此名称是否区分大小写取决于 <b>BioRS</b> 服务器是否区分大小写以及 <b>CASE_SENSITIVE</b> 服务器选项的值。仅在 <b>BioRS</b> 元素名称不同于列名称的情况下, 才必须指定 <b>BioRS</b> 元素名称。</p>	

表 154. 列选项及其设置 (续)

选项	描述和有效设置	缺省设置
ESCAPE_INPUT	<p>指定是否在 XML 输入值中替换 XML 特殊字符。使用此选项以包括 XML 片段作为输入，例如带有重复元素的 XML 片段。TEMPLATE 列选项必须在使用 ESCAPE_INPUT 列选项的列上定义。列数据类型必须为 VARCHAR 或 CHAR。</p> <p>有效值为:</p> <p>Y 如果 XML 输入包含特殊字符，则由 XML 用来表示输入字符的对应字符替换。</p> <p>N 输入字符将保持显示时的状态。</p>	Y
FOREIGN_KEY	<p>指示此昵称为子昵称并指定相应的父昵称的名称。一个昵称至多只能有一个 FOREIGN_KEY 列选项。此选项的值区分大小写。通过此选项指定的表拥有包装器生成的关键字。不得为此列指定 XPATH 选项。此列仅能用来连接父昵称和子昵称。</p> <p>如果父昵称具有不同的模式名称，则带有 FOREIGN_KEY 选项的 CREATE NICKNAME 语句将失败。</p> <p>除非通过在相应的 CREATE NICKNAME 语句中用引号把昵称括起来，从而将在 FOREIGN_KEY 子句中引用的昵称明确地定义为小写或混合大小写，否则，在 FOREIGN_KEY 子句中引用此昵称时，必须指定大写昵称。</p> <p>在列上设置此选项时，不能在此列上设置其它选项。</p>	
INDEX	<p>列（此选项在该列上出现）在定义行列组中的序数。此选项是必需的。</p>	
IS_INDEXED	<p>指示是否对相应的列建立索引（是否可以用谓词引用列）。有效值为 Y 和 N。仅可对相应元素已由 BioRS 服务器进行索引的列指定值 Y。</p>	<p>创建了昵称后，此选项（具有值 Y）会自动添加至与 BioRS 索引元素对应的任何列。</p>
IS_REPEATING	<p>指示此列是否具有多值。有效值为 Y 和 N。</p> <p>对于以下项，仅返回最后一个值:</p> <ul style="list-style-type: none"> <li>• 非 VARCHAR 重复属性</li> <li>• VARCHAR 列（指定了 ALL_VALUES 'N'）</li> </ul> <p>要克服此局限性，可以对重复属性列创建双定义。</p>	N

表 154. 列选项及其设置 (续)

选项	描述和有效设置	缺省设置
NUMERIC_STRING	<p>指定列是否包含数字字符的字符串。</p> <p>Y 此列包含数字字符的字符串“0”、“1”、“2”、...和“9”。但不包含空格。如果此列仅包含后跟尾随空格的数字字符串，则不要指定 Y。</p> <p>当对某列将 NUMERIC_STRING 设置为 Y 时，意味着您已通知优化器：此列不包含可能会影响对该列数据进行排序的任何空格。当数据源的整理顺序不同于联合服务器所使用的整理顺序时，使用此选项。由于整理顺序不同，所以不会从远程求值中排除使用此选项的列。</p> <p>N 此列不是数字字符串列，也不是包含空格的数字字符串列。</p>	N
PRIMARY_KEY	<p>指定此昵称是否为父昵称。列数据类型必须为 VARCHAR(16)。一个昵称至多只能有一个 PRIMARY_KEY 列选项。YES 是唯一的有效值。使用此选项指定的列拥有由包装器生成的关键字。不得为此列指定 XPATH 选项。此列仅能用来连接父昵称和子昵称。</p> <p>在列上设置此选项时，不能在此列上设置其它选项。</p>	
REFERENCED_OBJECT	<p>此选项仅对 BioRS 数据类型为 Reference 的列有效。此选项指定当前列所引用的 BioRS 数据银行的名称。此名称是否区分大小写取决于 BioRS 服务器是否区分大小写以及 CASE_SENSITIVE 服务器选项的值。</p>	
REMOTE_NAME	<p>指定相应 Documentum 属性或列的名称。此选项将远程属性或列名称映射到本地 DB2 UDB 列名称。</p>	DB2 UDB 列名称。
SOAPACTIONCOLUMN	<p>用于以“Web 服务描述语言”(WSDL)格式动态地指定 URI SOAPACTION 属性的列。仅在根昵称上指定此选项。</p> <p>在列上设置此选项时，不能在此列上设置其它选项。</p>	
TEMPLATE	<p>用于构造 XML 输入文档的列模板片段。片段必须符合指定的模板语法。</p>	
URLCOLUMN	<p>在运行查询时，用于为 Web 服务端点动态地指定 URL 的列。仅在根昵称上指定此选项。</p> <p>在列上设置此选项时，不能在此列上设置其它选项。</p>	

表 154. 列选项及其设置 (续)

选项	描述和有效设置	缺省设置
VARCHAR_NO_TRAILING_BLANKS	<p>此选项适用于具有变量字符数据类型的数据源，该数据类型在比较期间不会用尾随空格来填充长度。</p> <p>一些数据源（例如，Oracle）不具有空白填充字符比较语义，该语义与 DB2 UDB Linux 版、UNIX 版和 Windows 版比较语义返回的结果相同。当想要此选项仅适用于数据源对象中的特定 VARCHAR 或 VARCHAR2 列时，就设置此选项。</p> <p>Y 这些 VARCHAR 列中不存在尾随空格，或者数据源具有类似于联合服务器上的语义的空白填充字符比较语义。</p> <p>联合服务器将字符比较操作发送到数据源以进行处理。</p> <p>N 尾随空格在这些 VARCHAR 列中存在，并且数据源具有不同于联合服务器的空白填充字符比较语义。</p> <p>如果不能补偿对等语义，联合服务器将处理字符比较操作。例如，重写谓词。</p>	对于受影响的数据源为 N
XPATH	<p>在包含与此列相对应的数据的 XML 文档中指定 XPath 表达式。包装器在 CREATE NICKNAME 语句从此 XPath 选项应用该 XPATH 表达式之后对 XPath 表达式求值。</p>	

**相关概念:**

- 『下推分析』（《联合系统指南》）

**相关任务:**

- 『全局优化』（《联合系统指南》）

## 附录 H. 缺省正向数据类型映射

数据源数据类型与联合数据库数据类型之间的两种映射是正向类型映射和反向类型映射。在正向类型映射中，映射是从可比的本地类型至远程类型的映射。

您可以覆盖缺省类型映射或者使用 CREATE TYPE MAPPING 语句创建新的类型映射。

除非另有说明，否则这些映射对于所有受支持的版本都有效。

对于从数据源至 DB2 Linux 版、UNIX 版和 Windows 版的所有缺省正向数据类型映射，DB2 联合模式为 SYSIBM。

以下各表显示 DB2 Linux 版、UNIX 版和 Windows 版数据类型与数据源数据类型之间的缺省正向映射。

### DB2 z/OS 和 OS/390 版数据源

表 155. DB2 z/OS 和 OS/390 版正向缺省数据类型映射（并未显示所有列）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	255	32672	-	-	-	-	VARCHAR	-	0	N
CHAR	1	254	-	-	Y	-	CHAR	-	0	Y
CHAR	255	32672	-	-	Y	-	VARCHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
ROWID	-	-	-	-	Y	-	VARCHAR	40	-	Y
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-

表 155. DB2 z/OS 和 OS/390 版正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	Y	-	VARCHAR	-	0	Y
VARG	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N
VARGGRAPHIC	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N

## DB2 iSeries 版数据源

表 156. DB2 iSeries 版正向缺省数据类型映射（并未显示所有列）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	255	32672	-	-	-	-	VARCHAR	-	0	N
CHAR	1	254	-	-	Y	-	CHAR	-	0	Y
CHAR	255	32672	-	-	Y	-	VARCHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
GRAPHIC	128	16336	-	-	-	-	VARGGRAPHIC	-	0	N

表 156. DB2 iSeries 版正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
NUMERIC	-	-	-	-	-	-	DECIMAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	Y	-	VARCHAR	-	0	Y
VARG	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N
VARGGRAPHIC	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N

## DB2 服务器 VM 和 VSE 版数据源

表 157. DB2 服务器 VM 和 VSE 版正向缺省数据类型映射（并未显示所有列）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	1	254	-	-	Y	-	CHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBAHW	-	-	-	-	-	-	SMALLINT	-	0	-
DBAINT	-	-	-	-	-	-	INTEGER	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-

表 157. DB2 服务器 VM 和 VSE 版正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	Y	-	VARCHAR	-	0	Y
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPH	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

## DB2 Linux 版、UNIX 版和 Windows 版数据源

表 158. DB2 Linux 版、UNIX 版和 Windows 版正向缺省数据类型映射（并未显示所有列）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BIGINT	-	-	-	-	-	-	BIGINT	-	0	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	-	-	-	-	-	-	CHAR	-	0	N
CHAR	-	-	-	-	Y	-	CHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-



表 158. DB2 Linux 版、UNIX 版和 Windows 版正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
I LONGVAR	-	-	-	-	N	-	CLOB	-	-	-
I LONGVAR	-	-	-	-	Y	-	BLOB	-	-	-
I LONGVARG	-	-	-	-	-	-	DBCLOB	-	-	-
REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	0	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	0	Y
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	0	N

## Informix 数据源

表 159. Informix 正向缺省数据类型映射（并未显示所有列）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	2147483647	-	-
I BOOLEAN	-	-	-	-	-	-	CHARACTER	1	-	-

表 159. Informix 正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BYTE	-	-	-	-	-	-	BLOB	2147483647	-	-
CHAR	1	254	-	-	-	-	CHARACTER	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CLOB	-	-	-	-	-	-	CLOB	2147483647	-	-
DATE	-	-	-	-	-	-	DATE	4	-	-
DATETIME	0	4	0	4	-	-	DATE	4	-	-
DATETIME	6	10	6	10	-	-	TIME	3	-	-
DATETIME	0	4	6	15	-	-	TIMESTAMP	10	-	-
DATETIME	6	10	11	15	-	-	TIMESTAMP	10	-	-
DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
DECIMAL	32	130	-	-	-	-	DOUBLE	8	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
INTERVAL	-	-	-	-	-	-	VARCHAR	25	-	-
INT8	-	-	-	-	-	-	BIGINT	19	0	-
LVARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
MONEY	1	31	0	31	-	-	DECIMAL	-	-	-
MONEY	32	32	-	-	-	-	DOUBLE	8	-	-
NCHAR	1	254	-	-	-	-	CHARACTER	-	-	-
NCHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
NVARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
REAL	-	-	-	-	-	-	REAL	4	-	-
SERIAL	-	-	-	-	-	-	INTEGER	4	-	-
SERIAL8	-	-	-	-	-	-	BIGINT	-	-	-
SMALLFLOAT	-	-	-	-	-	-	REAL	4	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
TEXT	-	-	-	-	-	-	CLOB	2147483647	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-

注:

- 对于 Informix DATETIME 数据类型，DB2 UNIX 和 Windows 联合服务器将 Informix 高级别限定符用作 REMOTE\_LENGTH 并将 Informix 低级别限定符用作 REMOTE\_SCALE。
- Informix 限定符是在 Informix Client SDK datatype.h 文件中定义的“TU\_”常量。这些常量为:

表 159. Informix 正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
0 = YEAR		8 = MINUTE					13 = FRACTION(3)			
2 = MONTH		10 = SECOND					14 = FRACTION(4)			
4 = DAY		11 = FRACTION(1)					15 = FRACTION(5)			
6 = HOUR		12 = FRACTION(2)								

## Microsoft SQL Server 数据源

表 160. Microsoft SQL Server 正向缺省数据类型映射

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
bigint <sup>4</sup>	-	-	-	-	-	-	BIGINT	-	-	-
binary	1	254	-	-	-	-	CHARACTER	-	-	Y
binary	255	8000	-	-	-	-	VARCHAR	-	-	Y
bit	-	-	-	-	-	-	SMALLINT	2	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	8000	-	-	-	-	VARCHAR	-	-	N
datetime	-	-	-	-	-	-	TIMESTAMP	10	-	-
datetimen	-	-	-	-	-	-	TIMESTAMP	10	-	-
decimal	1	31	0	31	-	-	DECIMAL	-	-	-
decimal	32	38	0	38	-	-	DOUBLE	-	-	-
decimaln	1	31	0	31	-	-	DECIMAL	-	-	-
decimaln	32	38	0	38	-	-	DOUBLE	-	-	-
DUMMY65 <sup>1</sup>	1	38	-84	127	-	-	DOUBLE	-	-	-
DUMMY2000 <sup>3</sup>	1	38	-84	127	-	-	DOUBLE	-	-	-
float	-	8	-	-	-	-	DOUBLE	8	-	-

表 160. Microsoft SQL Server 正向缺省数据类型映射 (续)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
floatn	-	8	-	-	-	-	DOUBLE	8	-	-
float	-	4	-	-	-	-	REAL	4	-	-
floatn	-	4	-	-	-	-	REAL	4	-	-
image	-	-	-	-	-	-	BLOB	2147483647	-	Y
int	-	-	-	-	-	-	INTEGER	4	-	-
intn	-	-	-	-	-	-	INTEGER	4	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
moneyn	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	127	-	-	-	-	CHAR	-	-	N
nchar	128	4000	-	-	-	-	VARCHAR	-	-	N
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	8	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
ntext <sup>2</sup>	-	-	-	-	-	-	CLOB	2147483647	-	Y
nvarchar	1	4000	-	-	-	-	VARCHAR	-	-	N
real	-	-	-	-	-	-	REAL	4	-	-
smallint	-	-	-	-	-	-	SMALLINT	2	-	-
smalldatetime	-	-	-	-	-	-	TIMESTAMP	10	-	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
smallmoneyn	-	-	-	-	-	-	DECIMAL	10	4	-
SQL_BIGINT	-	-	-	-	-	-	DECIMAL	-	-	-
SQL_BIGINT <sup>4</sup>	-	-	-	-	-	-	BIGINT	-	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	Y
SQL_BINARY	255	8000	-	-	-	-	VARCHAR	-	-	Y
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N
SQL_CHAR	255	8000	-	-	-	-	VARCHAR	-	-	N
SQL_DATE	-	-	-	-	-	-	DATE	4	-	-
SQL_DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_DECIMAL	32	32	0	31	-	-	DOUBLE	8	-	-

表 160. Microsoft SQL Server 正向缺省数据类型映射 (续)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
SQL_DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_GUID <sup>2</sup>	1	4000	-	-	Y	-	VARCHAR	16	-	Y
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	-	-	Y
SQL_NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_REAL	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_TIME	-	-	-	-	-	-	TIME	3	-	-
SQL_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	10	-	-
SQL_TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_VARBINARY	1	8000	-	-	-	-	VARCHAR	-	-	Y
SQL_VARCHAR	1	8000	-	-	-	-	VARCHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	N
timestamp	-	-	-	-	-	-	VARCHAR	8	-	Y
tinyint	-	-	-	-	-	-	SMALLINT	2	-	-
uniqueidentifier <sup>2</sup>	1	4000	-	-	Y	-	VARCHAR	16	-	Y
varbinary	1	8000	-	-	-	-	VARCHAR	-	-	Y
varchar	1	8000	-	-	-	-	VARCHAR	-	-	N

注:

- | 1. 此类型映射仅对 Microsoft SQL Server V6.5 有效。
- | 2. 此类型映射仅对 Microsoft SQL Server V7 和 V2000 有效。
- | 3. 此类型映射仅对 Windows 2000 操作系统有效。
- | 4. 此类型映射仅对 Microsoft SQL Server V2000 有效。

## ODBC 数据源

表 161. ODBC 正向缺省数据类型映射（并未显示所有列）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
SQL_BIGINT	-	-	-	-	-	-	BIGINT	8	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	Y
SQL_BINARY	255	32672	-	-	-	-	VARCHAR	-	-	Y
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N
SQL_CHAR	255	32672	-	-	-	-	VARCHAR	-	-	N
SQL_DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	-	-	Y
SQL_NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_NUMERIC	32	32	0	31	-	-	DOUBLE	8	-	-
SQL_REAL	-	-	-	-	-	-	REAL	4	-	-
SQL_SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_TYPE_DATE	-	-	-	-	-	-	DATE	4	-	-
SQL_TYPE_TIME	-	-	-	-	-	-	TIME	3	-	-
SQL_TYPE_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	10	-	-
SQL_TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_VARBINARY	1	32672	-	-	-	-	VARCHAR	-	-	Y
SQL_VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	N
SQL_WCHAR	1	127	-	-	-	-	CHAR	-	-	N
SQL_WCHAR	128	16336	-	-	-	-	VARCHAR	-	-	N
SQL_WVARCHAR	1	16336	-	-	-	-	VARCHAR	-	-	N
SQL_WLONGVARCHAR	-	1073741823	-	-	-	-	CLOB	2147483647	-	N

## Oracle NET8 数据源

表 162. Oracle NET8 正向缺省数据类型映射

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	0	0	0	0	-	\0	BLOB	2147483647	0	Y
CHAR	1	254	0	0	-	\0	CHAR	0	0	N
CHAR	255	2000	0	0	-	\0	VARCHAR	0	0	N
CLOB	0	0	0	0	-	\0	CLOB	2147483647	0	N
DATE	0	0	0	0	-	\0	TIMESTAMP	0	0	N
FLOAT	1	126	0	0	-	\0	DOUBLE	0	0	N
LONG	0	0	0	0	-	\0	CLOB	2147483647	0	N
LONG RAW	0	0	0	0	-	\0	BLOB	2147483647	0	Y
MLSLABEL	0	0	0	0	-	\0	VARCHAR	255	0	N
NUMBER	1	38	-84	127	-	\0	DOUBLE	0	0	N
NUMBER	1	31	0	31	-	>=	DECIMAL	0	0	N
NUMBER	1	4	0	0	-	\0	SMALLINT	0	0	N
NUMBER	5	9	0	0	-	\0	INTEGER	0	0	N
NUMBER	-	10	0	0	-	\0	DECIMAL	0	0	N
RAW	1	2000	0	0	-	\0	VARCHAR	0	0	Y
ROWID	0	0	0	NULL	-	\0	CHAR	18	0	N
TIMESTAMP <sup>1</sup>	-	-	-	-	-	-	TIMESTAMP	10	-	-
VARCHAR2	1	4000	0	0	-	\0	VARCHAR	0	0	N

注:

1. 此类型映射仅对 Oracle 9i (或更新版本) 客户机和服务器配置有效。

## Sybase 数据源

表 163. Sybase CTLIB 正向缺省数据类型映射

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
binary	1	254	-	-	-	-	CHAR	-	-	Y
binary	255	16384	-	-	-	-	VARCHAR	-	-	Y
bit	-	-	-	-	-	-	SMALLINT	-	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	16384	-	-	-	-	VARCHAR	-	-	N
char null (请参阅 varchar)										
datetime	-	-	-	-	-	-	TIMESTAMP	-	-	-
datetimn	-	-	-	-	-	-	TIMESTAMP	-	-	-
decimal	1	31	0	31	-	-	DECIMAL	-	-	-
decimal	32	38	0	38	-	-	DOUBLE	-	-	-
decimaln	1	31	0	31	-	-	DECIMAL	-	-	-
decimaln	32	38	0	38	-	-	DOUBLE	-	-	-
float	-	4	-	-	-	-	REAL	-	-	-
float	-	8	-	-	-	-	DOUBLE	-	-	-
floatn	-	4	-	-	-	-	REAL	-	-	-
floatn	-	8	-	-	-	-	DOUBLE	-	-	-
image	-	-	-	-	-	-	BLOB	-	-	-
int	-	-	-	-	-	-	INTEGER	-	-	-
intn	-	-	-	-	-	-	INTEGER	-	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
moneyn	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	254	-	-	-	-	CHAR	-	-	N
nchar	255	16384	-	-	-	-	VARCHAR	-	-	N
nchar null (请参阅 nvarchar)										
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
nvarchar	1	16384	-	-	-	-	VARCHAR	-	-	N



表 163. Sybase CTLIB 正向缺省数据类型映射 (续)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
real	-	-	-	-	-	-	REAL	-	-	-
smalldatetime	-	-	-	-	-	-	TIMESTAMP	-	-	-
smallint	-	-	-	-	-	-	SMALLINT	-	-	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
sysname	1	254	-	-	-	-	CHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	-
timestamp	-	-	-	-	-	-	VARCHAR	8	-	Y
tinyint	-	-	-	-	-	-	SMALLINT	-	-	-
unichar <sup>1</sup>	1	254	-	-	-	-	CHAR	-	-	N
unichar <sup>1</sup>	255	16384	-	-	-	-	VARCHAR	-	-	N
unichar null (请参阅 univarchar)										
univarchar <sup>1</sup>	1	16384	-	-	-	-	VARCHAR	-	-	N
varbinary	1	16384	-	-	-	-	VARCHAR	-	-	Y
varchar	1	16384	-	-	-	-	VARCHAR	-	-	N

注:

1. 对非 Unicode 联合数据库有效。

## Teradata 数据源

表 164. Teradata 正向缺省数据类型映射 (并未显示所有列)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BYTE	1	254	-	-	-	-	CHAR	-	-	Y
BYTE	255	32672	-	-	-	-	VARCHAR	-	-	Y

表 164. Teradata 正向缺省数据类型映射（并未显示所有列）（续）

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BYTE	32673	64000	-	-	-	-	BLOB	-	-	-
BYTEINT	-	-	-	-	-	-	SMALLINT	-	-	-
CHAR	1	254	-	-	-	-	CHARACTER	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CHAR	32673	64000	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DECIMAL	1	18	0	18	-	-	DECIMAL	-	-	-
DOUBLE PRECISION	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	-	-
GRAPHIC	128	16336	-	-	-	-	VARGRAPHIC	-	-	-
GRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
INTERVAL	-	-	-	-	-	-	CHAR	-	-	-
NUMERIC	1	18	0	18	-	-	DECIMAL	-	-	-
REAL	-	-	-	-	-	-	DOUBLE	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	-	-
VARBYTE	1	32762	-	-	-	-	VARCHAR	-	-	Y
VARBYTE	32763	64000	-	-	-	-	BLOB	-	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
VARCHAR	32673	64000	-	-	-	-	CLOB	-	-	-
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	-	-
VARGRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-

**相关概念:**

- 『正向和逆向数据类型映射』（《联合系统指南》）

**相关参考:**

- 『将 long 数据类型改变为 varchar 数据类型』（《联合系统指南》）
- 『Unicode 缺省正向数据类型映射 - NET8 包装器』（《联合系统指南》）
- 『Unicode 缺省正向数据类型映射 - Sybase 包装器』（《联合系统指南》）
- 『Unicode 缺省正向数据类型映射 - ODBC 包装器』（《联合系统指南》）

- 『Unicode 缺省正向数据类型映射 - Microsoft SQL Server 包装器』（《联合系统指南》）



## 附录 I. 缺省反向数据类型映射

数据源数据类型与联合数据库数据类型之间的两种映射是正向类型映射和反向类型映射。在正向类型映射中，映射是从可比的本地类型至远程类型的映射。另一种映射类型为反向类型映射，它与透明 DDL 配合使用来创建或修改远程表。

对于大多数数据源，缺省类型映射在包装器中。DB2 系列数据源的缺省类型映射在 DRDA 包装器中。Informix 的缺省类型映射在 INFORMIX 包装器中，依此类推。

当对 DB2 联合数据库定义远程表或视图时，定义包括反向类型映射。该映射来自每列的本地 DB2 Linux 版、UNIX 版和 Windows 版数据类型以及相应的远程数据类型。例如，有一个缺省反向类型映射，在该映射中，本地类型 REAL 指向 Informix 类型 SMALLFLOAT。

DB2 Linux 版、UNIX 版和 Windows 版联合服务器不支持 LONG VARCHAR、LONG VARGRAPHIC、DATALINK 和用户定义的类型映射。

当使用 CREATE TABLE 语句创建远程表时，您将指定想要包括在远程表中的本地数据类型。这些缺省反向类型映射将把相应的远程类型分配给这些列。例如，假定您使用 CREATE TABLE 语句来定义一个具有列 C2 的 Informix 表。在该语句中，您将 BIGINT 指定为 C2 的数据类型。BIGINT 的缺省反向类型映射取决于您正在对其创建表的 Informix 的版本。Informix 表中的 C2 的映射将指向 Informix V8 中的 DECIMAL 和 Informix V9 中的 INT8。

您可以覆盖缺省反向类型映射，或使用 CREATE TYPE MAPPING 语句创建新的反向类型映射。

以下各表显示 DB2 Linux 版、UNIX 版和 Windows 版本地数据类型与远程数据源数据类型之间的缺省反向映射。

除非另有说明，否则这些映射对于所有受支持的版本都有效。

### DB2 z/OS 和 OS/390 版数据源

表 165. DB2 z/OS 和 OS/390 版反向缺省数据类型映射（并未显示所有列）

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-

表 165. DB2 z/OS 和 OS/390 版反向缺省数据类型映射（并未显示所有列）（续）

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
CHARACTER	-	-	-	-	-	-	CHAR	-	-	N
CHARACTER	-	-	-	-	Y	-	CHAR	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
I DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
I REAL	-	4	-	-	-	-	REAL	-	-	-
I SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	N

## DB2 iSeries 版数据源

表 166. DB2 iSeries 版反向缺省数据类型映射 (并未显示所有列)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	N
CHARACTER	-	-	-	-	Y	-	CHARACTER	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	NUMERIC	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	FLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPHIC	-	-	-	-	-	-	VARG	-	-	N

## DB2 VM 和 VSE 版数据源

表 167. DB2 VM 和 VSE 版反向缺省数据类型映射 (并未显示所有列)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	-
CHARACTER	-	-	-	-	Y	-	CHAR	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	-
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPH	-	-	-	-	-	-	VARGRAPH	-	-	N



## DB2 Linux 版、UNIX 版和 Windows 版数据源

表 168. DB2 Linux 版、UNIX 版和 Windows 版反向缺省数据类型映射（并未显示所有列）

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	FEDERATED_BIT_DATA
BIGINT	-	8	-	-	-	-	BIGINT	-	-	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	N
CHARACTER	-	-	-	-	Y	-	CHAR	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
I REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	-	N
I VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

## Informix 数据源

表 169. Informix 反向缺省数据类型映射

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BIGINT <sup>1</sup>	-	-	-	-	-	-	DECIMAL	19	-	-
BIGINT <sup>2</sup>	-	-	-	-	-	-	INT8	-	-	-
BLOB	1	2147483647	-	-	-	-	BYTE	-	-	-
CHARACTER	-	-	-	-	N	-	CHAR	-	-	-
CHARACTER	-	-	-	-	Y	-	BYTE	-	-	-
CLOB	1	2147483647	-	-	-	-	TEXT	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	SMALLFLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	DATETIME	6	10	-
TIMESTAMP	-	10	-	-	-	-	DATETIME	0	15	-
VARCHAR	1	254	-	-	N	-	VARCHAR	-	-	-
VARCHAR	255	32672	-	-	N	-	TEXT	-	-	-
VARCHAR	-	-	-	-	Y	-	BYTE	-	-	-
VARCHAR <sup>2</sup>	255	2048	-	-	N	-	LVARCHAR	-	-	-
VARCHAR <sup>2</sup>	2049	32672	-	-	N	-	TEXT	-	-	-

注:

1. 此类型映射仅对 Informix Server V8 (或更低版本) 有效。
2. 此类型映射仅对 Informix Server V9 有效。

对于 Informix DATETIME 数据类型, DB2 UNIX 和 Windows 联合服务器将 Informix 高级别限定符用作 REMOTE\_LENGTH 并将 Informix 低级别限定符用作 REMOTE\_SCALE。

Informix 限定符是在 Informix Client SDK datatype.h 文件中定义的 “TU\_” 常量。这些常量为:

0 = YEAR	8 = MINUTE	13 = FRACTION(3)
2 = MONTH	10 = SECOND	14 = FRACTION(4)
4 = DAY	11 = FRACTION(1)	15 = FRACTION(5)
6 = HOUR	12 = FRACTION(2)	

## Microsoft SQL Server 数据源

表 170. Microsoft SQL Server 反向缺省数据类型映射 (并未显示所有列)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BIGINT <sup>1</sup>	-	-	-	-	-	-	bigint	-	-	-
BLOB	-	-	-	-	-	-	image	-	-	-
CHARACTER	-	-	-	-	Y	-	binary	-	-	-
CHARACTER	-	-	-	-	N	-	char	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	4	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	8	-	-	-	-	float	-	-	-
INTEGER	-	-	-	-	-	-	int	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-
REAL	-	4	-	-	-	-	real	-	-	-
TIME	-	3	-	-	-	-	datetime	-	-	-
TIMESTAMP	-	10	-	-	-	-	datetime	-	-	-
VARCHAR	1	8000	-	-	N	-	varchar	-	-	-
VARCHAR	8001	32672	-	-	N	-	text	-	-	-
VARCHAR	1	8000	-	-	Y	-	varbinary	-	-	-
VARCHAR	8001	32672	-	-	Y	-	image	-	-	-

注:

1. 此类型映射仅对 Microsoft SQL Server V2000 有效。

## Oracle NET8 数据源

表 171. Oracle NET8 反向缺省数据类型映射

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	0	2147483647	0	0	Y	\0	BLOB	0	0	Y
CHARACTER	1	254	0	0	N	\0	CHAR	0	0	N
CHARACTER	1	254	0	0	Y	\0	RAW	0	0	Y
CLOB	0	2147483647	0	0	N	\0	CLOB	0	0	N
DATE	0	4	0	0	N	\0	DATE	0	0	N
DECIMAL	0	0	0	0	N	\0	NUMBER	0	0	N
DOUBLE	0	8	0	0	N	\0	FLOAT	126	0	N
FLOAT	0	8	0	0	N	\0	FLOAT	126	0	N
INTEGER	0	4	0	0	N	\0	NUMBER	9	0	N
REAL	0	4	0	0	N	\0	FLOAT	63	0	N
SMALLINT	0	2	0	0	N	\0	NUMBER	4	0	N
TIME	0	3	0	0	N	\0	DATE	0	0	N
TIMESTAMP	0	10	0	0	N	\0	DATE	0	0	N
VARCHAR	1	4000	0	0	N	\0	VARCHAR2	0	0	N
VARCHAR	1	2000	0	0	Y	\0	RAW	0	0	Y

注: DB2 通用数据库 Linux 版、UNIX 版和 Windows 版 BIGINT 数据类型不可用于透明 DDL。创建远程 Oracle 表时,不能在 CREATE TABLE 语句中指定 BIGINT 数据类型。

## Sybase 数据源

表 172. Sybase CTLIB 缺省反向数据类型映射

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BIGINT	-	-	-	-	-	-	decimal	19	0	-
BLOB	-	-	-	-	-	-	image	-	-	-
CHARACTER	-	-	-	-	N	-	char	-	-	-
CHARACTER	-	-	-	-	Y	-	binary	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	-	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	-	-	-	-	-	float	-	-	-
GRAPHIC	-	-	-	-	-	-	unichar	-	-	-
VARGRAPHIC	-	-	-	-	-	-	univarchar	-	-	-
INTEGER	-	-	-	-	-	-	integer	-	-	-
REAL	-	-	-	-	-	-	real	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-
TIME	-	-	-	-	-	-	datetime	-	-	-
TIMESTAMP	-	-	-	-	-	-	datetime	-	-	-
VARCHAR <sup>1</sup>	1	255	-	-	N	-	varchar	-	-	-
VARCHAR <sup>1</sup>	256	32672	-	-	N	-	text	-	-	-
VARCHAR <sup>2</sup>	1	16384	-	-	N	-	varchar	-	-	-
VARCHAR <sup>2</sup>	16385	32672	-	-	N	-	text	-	-	-
VARCHAR <sup>1</sup>	1	255	-	-	Y	-	varbinary	-	-	-
VARCHAR <sup>1</sup>	256	32672	-	-	Y	-	image	-	-	-
VARCHAR <sup>2</sup>	1	16384	-	-	Y	-	varbinary	-	-	-
VARCHAR <sup>2</sup>	16385	32672	-	-	Y	-	image	-	-	-

注:

1. 此类型映射仅对具有 Sybase server V12.0 (或更低版本) 的 CTLIB 有效。
2. 此类型映射仅对具有 Sybase server V12.5 (或更高版本) 的 CTLIB 有效。

## Teradata 数据源

表 173. Teradata 反向缺省数据类型映射（并未显示所有列）

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	FEDERATED_BIT_DATA
BLOB <sup>1</sup>	1	64000	-	-	-	-	VARBYTE	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	-
CHARACTER	-	-	-	-	Y	-	BYTE	-	-	-
CLOB <sup>2</sup>	1	64000	-	-	-	-	VARCHAR	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DBCLOB <sup>3</sup>	1	32000	-	-	-	-	VARGRAPHIC	-	-	-
DECIMAL	1	18	0	18	-	-	DECIMAL	-	-	-
DECIMAL	19	31	0	31	-	-	FLOAT	-	-	-
DOUBLE	-	-	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
REAL	-	-	-	-	-	-	FLOAT	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	-	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	-
VARCHAR	-	-	-	-	Y	-	VARBYTE	-	-	-
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

注:

1. Teradata VARBYTE 数据类型可以只包含指定长度（1 至 64000）的 DB2 BLOB 数据类型。
2. Teradata VARCHAR 数据类型可以只包含指定长度（1 至 64000）的 DB2 CLOB 数据类型。
3. Teradata VARGRAPHIC 数据类型可以只包含指定长度（1 至 32000）的 DB2 DBCLOB 数据类型。

相关概念:

- 『正向和逆向数据类型映射』（《联合系统指南》）

## 附录 J. 联合系统的函数映射选项

DB2 Information Integrator 提供了现有内置数据源函数与内置 DB2 函数之间的缺省映射。对于大多数数据源，缺省函数映射在包装器中。要使用联合服务器不识别的数据源函数，必须创建数据源函数与联合数据库的相应函数之间的函数映射。

函数映射选项的主要用途是提供有关执行数据源的数据源函数潜在成本的信息。下推分析确定数据源的函数是否能够在查询中执行函数。查询优化器决定是否将函数处理下推至数据源就是最低成本选择。

在函数映射定义中提供的统计信息帮助查询优化器将执行数据源函数的估计成本与执行 DB2 函数的估计成本进行比较。

表 174. 函数映射选项及其设置

选项	有效设置	缺省设置
DISABLE	禁用缺省函数映射。有效值为 'Y' 和 'N'。	'N'
INITIAL_INSTS	第一次和最后一次调用数据源函数时处理的指令的估计数目。	'0'
INITIAL_IOS	第一次和最后一次调用数据源函数时执行的 I/O 的估计次数。	'0'
IOS_PER_ARGBYTE	传递到数据源函数的自变量集的每个字节消耗的 I/O 的估计次数。	'0'
IOS_PER_INVOC	数据源函数的每次调用所进行的 I/O 的估计次数	'0'
INSTS_PER_ARGBYTE	传递到数据源函数的自变量集的每个字节处理的指令的估计数目。	'0'
INSTS_PER_INVOC	数据源函数的每次调用所处理的指令的估计数目。	'450'
PERCENT_ARGBYTES	数据源函数实际将读取的输入自变量字节数的估计平均百分比。	'100'
REMOTE_NAME	数据源函数的名称。	本地名称





---

## DB2 Information Integrator 文档

该主题提供了关于 DB2 Information Integrator 可用的文档的信息。该主题中的表提供了正式的文档标题、书号以及每本 PDF 书籍的位置。要订购印刷书籍，您必须知道正式的书名或文档书号。该主题还提供了 DB2 Information Integrator 发行说明和安装需求的标题、文件名以及位置。

该主题包含以下部分:

- 访问 DB2 Information Integrator 文档
- 关于 z/OS 上的复制功能的文档
- 关于 z/OS 版 DB2 通用数据库的事件发布功能的文档
- 关于 z/OS 上的 IMS 和 VSAM 的事件发布功能的文档
- 关于 Linux、UNIX 和 Windows 上的事件发布功能和复制功能的文档
- 关于 z/OS 上的联合功能的文档
- 关于 Linux、UNIX 和 Windows 上的联合功能的文档
- 关于 Linux、UNIX 和 Windows 上的企业搜索的文档
- 发行说明和安装需求

---

## 访问 DB2 Information Integrator 文档

所有 DB2 Information Integrator 书籍和发行说明都提供了 PDF 文件，在 DB2 Information Integrator Support Web 站点提供，网址为：  
[www.ibm.com/software/data/integration/db2ii/support.html](http://www.ibm.com/software/data/integration/db2ii/support.html)。

要访问最新的 DB2 Information Integrator 产品文档，可从 DB2 Information Integrator Support Web 站点单击 Product Information 链接，如第 586 页的图 48 所示。

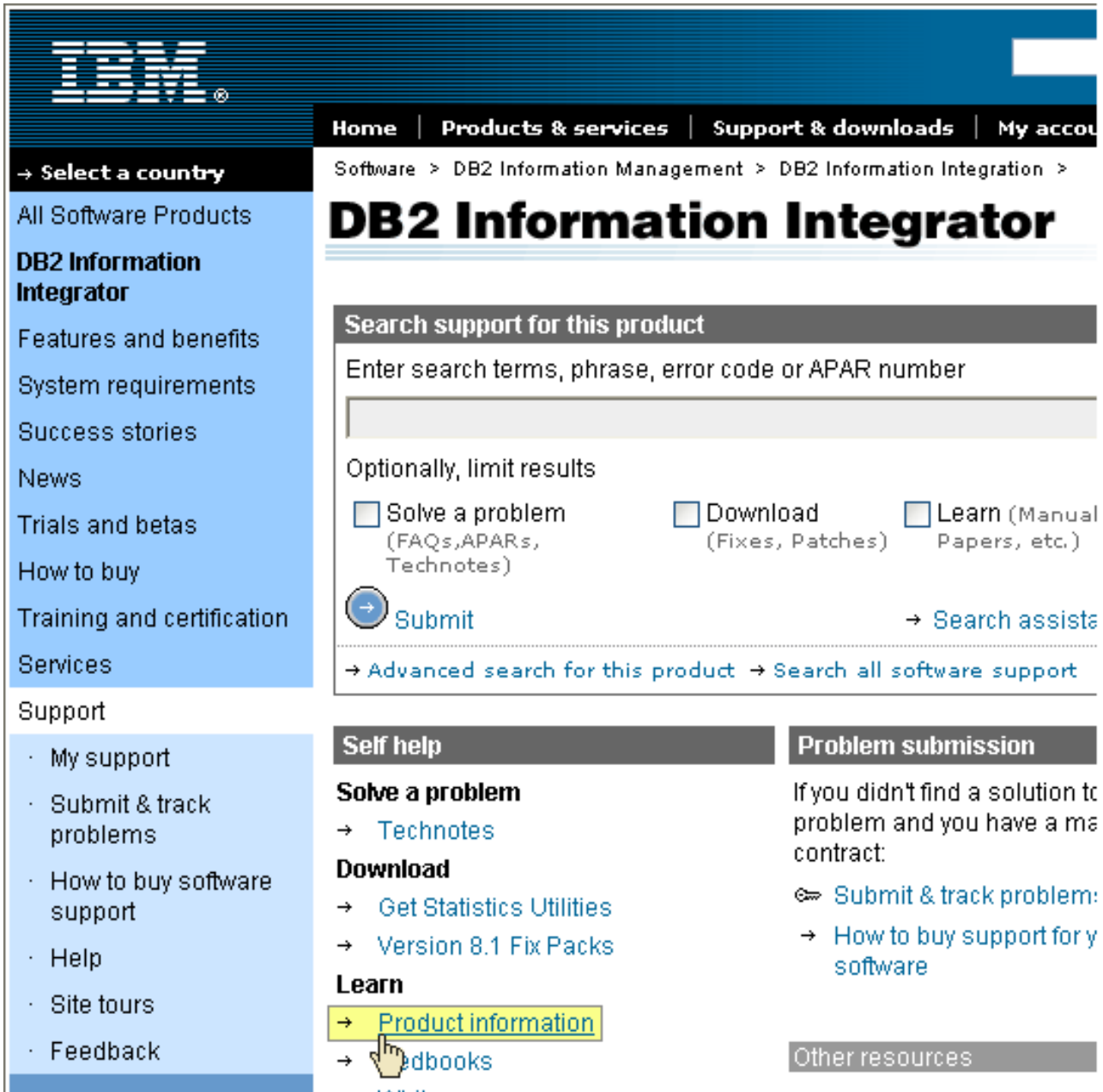


图 48. 从 DB2 Information Integrator Support Web 站点访问 Product Information 链接

从 Product Information 链接可访问所有受支持语言的最新 DB2 Information Integrator 文档:

- PDF 文件格式的 DB2 Information Integrator 产品文档
- 修订包产品文档, 包括发行说明
- 下载和安装用于 Linux、UNIX 和 Windows 的 DB2 信息中心的指示信息
- 在线 DB2 信息中心的链接

在列表中滚动以查找您正在使用的 DB2 Information Integrator 版本的产品文档。

DB2 Information Integrator Support Web 站点也提供了支持文档、IBM 红皮书、白皮书、产品下载、对用户组的链接以及关于 DB2 Information Integrator 的新闻。

也可以从 *DB2 PDF* 文档 CD 查看并打印 DB2 Information Integrator PDF 书籍。

要查看或打印 PDF 文档:

1. 从 *DB2 PDF* 文档 CD 的根目录打开 index.htm 文件。
2. 单击要使用的语言。
3. 单击要查看的文档的链接。

---

## 关于 z/OS 上的复制功能的文档

表 175. 关于 z/OS 上的复制功能的 DB2 Information Integrator 文档

书名	书号	位置
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Introduction to Replication and Event Publishing</i>	GC18-7567	DB2 Information Integrator Support Web 站点
<i>Migrating to SQL Replication</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"><li>• <i>DB2 PDF</i> 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>
<i>Replication Installation and Customization Guide for z/OS</i>	SC18-9127	DB2 Information Integrator Support Web 站点
《SQL 复制指南和参考》	S152-0734	<ul style="list-style-type: none"><li>• <i>DB2 PDF</i> 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>
<i>Tuning for Replication and Event Publishing Performance</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Tuning for SQL Replication Performance</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	N/A	<ul style="list-style-type: none"><li>• 在 DB2 信息中心中, 产品概述 &gt; 信息集成 &gt; <b>DB2 Information Integrator</b> 概述 &gt; 问题、解决办法和文档更新</li><li>• DB2 Information Integrator 安装启动板</li><li>• DB2 Information Integrator Support Web 站点</li><li>• The <i>DB2 Information Integrator</i> 产品 CD</li></ul>

## 关于 z/OS 版 DB2 通用数据库的事件发布功能的文档

表 176. 关于 z/OS 版 DB2 通用数据库的事件发布功能的 DB2 Information Integrator 文档

书名	书号	位置
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Introduction to Replication and Event Publishing</i>	GC18-7567	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
<i>Tuning for Replication and Event Publishing Performance</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	N/A	<ul style="list-style-type: none"> <li>• 在 DB2 信息中心中, 产品概述 &gt; 信息集成 &gt; <b>DB2 Information Integrator 概述</b> &gt; 问题、解决办法和文档更新</li> <li>• DB2 Information Integrator 安装启动板</li> <li>• DB2 Information Integrator Support Web 站点</li> <li>• The <i>DB2 Information Integrator</i> 产品 CD</li> </ul>

## 关于 z/OS 上的 IMS 和 VSAM 的事件发布功能的文档

表 177. 关于 z/OS 上的 IMS 和 VSAM 的事件发布功能的 DB2 Information Integrator 文档

书名	书号	位置
<i>Client Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9160	DB2 Information Integrator Support Web 站点
<i>Data Mapper Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9163	DB2 Information Integrator Support Web 站点
<i>Getting Started with Event Publisher for z/OS</i>	GC18-9186	DB2 Information Integrator Support Web 站点
<i>Installation Guide for Classic Federation and Event Publisher for z/OS</i>	GC18-9301	DB2 Information Integrator Support Web 站点
<i>Operations Guide for Event Publisher for z/OS</i>	SC18-9157	DB2 Information Integrator Support Web 站点
<i>Planning Guide for Event Publisher for z/OS</i>	SC18-9158	DB2 Information Integrator Support Web 站点
<i>Reference for Classic Federation and Event Publisher for z/OS</i>	SC18-9156	DB2 Information Integrator Support Web 站点

表 177. 关于 z/OS 上的 IMS 和 VSAM 的事件发布功能的 DB2 Information Integrator 文档 (续)

书名	书号	位置
<i>System Messages for Classic Federation and Event Publisher for z/OS</i>	SC18-9162	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS</i>	N/A	DB2 Information Integrator Support Web 站点

## 关于 Linux、UNIX 和 Windows 上的事件发布功能和复制功能的文档

表 178. 关于 Linux、UNIX 和 Windows 上的事件发布功能和复制功能的 DB2 Information Integrator 文档

书名	书号	位置
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	N/A	DB2 Information Integrator Support Web 站点
《Linux、UNIX 和 Windows 上的安装指南》	G152-0550	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
<i>Introduction to Replication and Event Publishing</i>	GC18-7567	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
<i>Migrating to SQL Replication</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
《SQL 复制指南和参考》	S152-0734	DB2 Information Integrator Support Web 站点
<i>Tuning for Replication and Event Publishing Performance</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Tuning for SQL Replication Performance</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	N/A	<ul style="list-style-type: none"> <li>• 在 DB2 信息中心中, 产品概述 &gt; 信息集成 &gt; <b>DB2 Information Integrator</b> 概述 &gt; 问题、解决办法和文档更新</li> <li>• DB2 Information Integrator 安装启动板</li> <li>• DB2 Information Integrator Support Web 站点</li> <li>• The <i>DB2 Information Integrator</i> 产品 CD</li> </ul>

## 关于 z/OS 上的联合功能的文档

表 179. 关于 z/OS 上的联合功能的 DB2 Information Integrator 文档

书名	书号	位置
<i>Client Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9160	DB2 Information Integrator Support Web 站点
<i>Data Mapper Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9163	DB2 Information Integrator Support Web 站点
<i>Getting Started with Classic Federation for z/OS</i>	GC18-9155	DB2 Information Integrator Support Web 站点
<i>Installation Guide for Classic Federation and Event Publisher for z/OS</i>	GC18-9301	DB2 Information Integrator Support Web 站点
<i>Reference for Classic Federation and Event Publisher for z/OS</i>	SC18-9156	DB2 Information Integrator Support Web 站点
<i>System Messages for Classic Federation and Event Publisher for z/OS</i>	SC18-9162	DB2 Information Integrator Support Web 站点
<i>Transaction Services Guide for Classic Federation for z/OS</i>	SC18-9161	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS</i>	N/A	DB2 Information Integrator Support Web 站点

## 关于 Linux、UNIX 和 Windows 上的联合功能的文档

表 180. 关于 Linux、UNIX 和 Windows 上的联合功能的 DB2 Information Integrator 文档

书名	书号	位置
《应用程序开发者指南》	S152-0601	<ul style="list-style-type: none"><li>• DB2 PDF 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>
《开发包装器的 C++ API 参考》	S152-0844	<ul style="list-style-type: none"><li>• DB2 PDF 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>
《数据源配置指南》	N/A	<ul style="list-style-type: none"><li>• DB2 PDF 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>
《联合系统指南》	S152-0600	<ul style="list-style-type: none"><li>• DB2 PDF 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>
<i>Guide to Configuring the Content Connector for VeniceBridge</i>	N/A	DB2 Information Integrator Support Web 站点
《Linux、UNIX 和 Windows 上的安装指南》	G152-0550	<ul style="list-style-type: none"><li>• DB2 PDF 文档 CD</li><li>• DB2 Information Integrator Support Web 站点</li></ul>

表 180. 关于 Linux、UNIX 和 Windows 上的联合功能的 DB2 Information Integrator 文档 (续)

书名	书号	位置
<i>Java API Reference for Developing Wrappers</i>	SC18-9173	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
《迁移指南》	S152-0603	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
《包装器开发者指南》	S152-0845	<ul style="list-style-type: none"> <li>• DB2 PDF 文档 CD</li> <li>• DB2 Information Integrator Support Web 站点</li> </ul>
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	N/A	<ul style="list-style-type: none"> <li>• 在 DB2 信息中心中, 产品概述 &gt; 信息集成 &gt; <b>DB2 Information Integrator</b> 概述 &gt; 问题、解决办法和文档更新</li> <li>• DB2 Information Integrator 安装启动板</li> <li>• DB2 Information Integrator Support Web 站点</li> <li>• The <i>DB2 Information Integrator</i> 产品 CD</li> </ul>

## 关于 Linux、UNIX 和 Windows 上的企业搜索功能的文档

表 181. 关于 Linux、UNIX 和 Windows 上的企业搜索功能的 DB2 Information Integrator 文档

书名	书号	位置
<i>Administering Enterprise Search</i>	SC18-9283	DB2 Information Integrator Support Web 站点
<i>Installation Guide for Enterprise Search</i>	GC18-9282	DB2 Information Integrator Support Web 站点
<i>Programming Guide and API Reference for Enterprise Search</i>	SC18-9284	DB2 Information Integrator Support Web 站点
<i>Release Notes for Enterprise Search</i>	N/A	DB2 Information Integrator Support Web 站点

## 发行说明和安装需求

发行说明提供了针对您所用产品的发行版和修订包级别的信息, 还包括了对每个发行版的文档的最新修订。

安装需求提供了针对您所用产品的发行版的信息。

表 182. DB2 Information Integrator 发行说明和安装需求

书名	文件名	位置
<i>Installation Requirements for IBM DB2 Information Integrator Event Publishing Edition, Replication Edition, Standard Edition, Advanced Edition, Advanced Edition Unlimited, Developer Edition, and Replication for z/OS</i>	Prereqs	<ul style="list-style-type: none"> <li>• The <i>DB2 Information Integrator</i> 产品 CD</li> <li>• DB2 Information Integrator 安装启动板</li> </ul>
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	ReleaseNotes	<ul style="list-style-type: none"> <li>• 在 DB2 信息中心中，产品概述 &gt; 信息集成 &gt; <b>DB2 Information Integrator</b> 概述 &gt; 问题、解决办法和文档更新</li> <li>• DB2 Information Integrator 安装启动板</li> <li>• DB2 Information Integrator Support Web 站点</li> <li>• The <i>DB2 Information Integrator</i> 产品 CD</li> </ul>
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS</i>	N/A	DB2 Information Integrator Support Web 站点
<i>Release Notes for Enterprise Search</i>	N/A	DB2 Information Integrator Support Web 站点

要查看产品 CD 上的安装需求和发行说明：

- 在 Windows 操作系统上输入：

```
x:\doc\%L
```

*x* 是 Windows CD 盘符，*%L* 是要使用的文档的语言环境，例如，en\_US。

- 在 UNIX 操作系统上输入：

```
/cdrom/doc/%L/
```

*cdrom* 表示 UNIX 上的 CD 安装点，*%L* 是要使用的文档的语言环境，例如，en\_US。



---

## 辅助功能

辅助功能部件可帮助那些身体有某些缺陷（如活动不方便或视力不太好）的用户成功地使用软件产品。以下列表指定 DB2<sup>®</sup> V8 产品中的主要辅助功能部件：

- 所有 DB2 功能可使用键盘（而不是鼠标）导航来实现。有关更多信息，请参阅『键盘输入和导航』。
- 可定制 DB2 用户界面上的字体大小和颜色。有关更多信息，请参阅『界面显示的辅助功能』。
- DB2 产品支持使用 Java™ Accessibility API 的辅助功能应用程序。有关更多信息，请参阅第 594 页的『与辅助技术的兼容性』。
- DB2 文档是以易使用格式提供的。有关更多信息，请参阅第 594 页的『文档的辅助功能』。

---

## 键盘输入和导航

### 键盘输入

只使用键盘就可以操作 DB2 工具。使用键或键组合就可以执行使用鼠标所能完成的操作。标准操作系统击键用于标准操作系统操作。

有关使用键或键组合执行操作的更多信息，请参阅 键盘快捷方式和加速键：公共 GUI 帮助。

### 键盘导航

可使用键或键组合来导航 DB2 工具用户界面。

有关使用键或键组合来导航 DB2 工具的更多信息，请参阅 键盘快捷方式和加速键：公共 GUI 帮助。

### 键盘焦点

在 UNIX<sup>®</sup> 操作系统中，击键操作起作用的活动窗口的区域将突出显示。

---

## 界面显示的辅助功能

DB2 工具所具有的功能部件使视力不太好的用户更易使用。这些辅助功能方面的增强包括了对可定制字体属性的支持。

### 字体设置

可使用“工具设置”笔记本来选择菜单和对话框窗口中文本的颜色、大小和字体。

有关指定字体设置的更多信息，请参阅 更改菜单和文本的字体：公共 GUI 帮助。

### 不依赖于颜色

不需要分辨颜色就可以使用此产品中的任何功能。

---

## 与辅助技术的兼容性

DB2 工具界面支持 Java Accessibility API，它使您能够将屏幕阅读器和其它辅助技术与 DB2 产品配合使用。

---

## 文档的辅助功能

DB2 的相关文档是以 XHTML 1.0 格式提供的，它在大部分 Web 浏览器中是可查看的。XHTML 允许您根据浏览器中设置的显示首选项来查看文档。还允许您使用屏幕阅读器和其它辅助技术。

语法图是以点分十进制格式提供的。仅当使用屏幕阅读器访问联机文档时，此格式才可用。

### 相关概念:

- 『点分十进制语法图』（基础结构主题（DB2 公共文件））

### 相关任务:

- 『键盘快捷方式和加速键：公共 GUI 帮助』
- 『更改菜单和文本的字体：公共 GUI 帮助』

---

## 声明

此信息是为在美国提供的产品和服务编写的。IBM 可能在所有国家或地区不提供本文件中讨论的产品、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，请向您当地的 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（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：

IBM World Trade Asia Corporation Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan

**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 以“按现状”的基础提供本出版物，不附有任何形式的（无论是明示的还是默示的）保证，包括（但不限于）对非侵权性、适销性和适用于某特定用途的默示保证。某些国家或地区在某些交易中不允许免除明示或默示的保证。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 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。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：（i）允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及（ii）允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Corporation  
J46A/G4  
555 Bailey Avenue

San Jose, CA 95141-1003  
U.S.A.

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

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 国际程序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。

本资料中包含用于日常业务运作的数据和报表的示例。为了尽可能完整地说明问题，这些示例可能包含个人、公司、品牌和产品的名称。所有这些名称都是虚构的，如与实际商业企业所使用的名称和地址有雷同，纯属巧合。

版权许可证:

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。用户如果是为按照 IBM 的应用程序编程接口开发、使用、经销或分发应用程序，则可以任何形式复制、修改和分发这些样本程序，而无须向 IBM 付费。

凡这些样本程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明:

©（贵公司的名称）（年）。此部分代码是根据 IBM 公司的样本程序衍生出来的。© Copyright IBM Corp.（输入年份）。All rights reserved.

---

## 商标

下列各项是国际商业机器公司在美国和 / 或其他国家或地区的商标:

IBM  
DB2  
DB2 Universal Database  
Domino  
Domino.Doc

Informix  
Lotus  
Lotus Discovery Server  
Lotus Notes  
QuickPlace  
Sametime  
SecureWay  
WebSphere

下列各项是其他公司的商标或注册商标:

Java 和所有基于 Java 的商标和徽标是 Sun Microsystems, Inc. 在美国和 / 或其他国家或地区的商标或注册商标。

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

Intel、Intel Inside (徽标)、MMX 和 Pentium 是 Intel Corporation 在美国和 / 或其他国家或地区的商标。

UNIX 是 The 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。



# 索引

## [ A ]

安全性

Web 服务包装器 381

## [ B ]

包装器

表结构文件, 注册 332

规划 22

库 34

库文件 35

联合数据源的核对表 27

描述 10

缺省名 11

商业应用程序 109, 116

注册概述 54

BioRS, 注册 65

BLAST, 注册 97

DB2 系列数据源, 注册 147

Documentum, 注册 160

Entrez, 注册 182

Excel 文件, 注册 203

Extended Search, 注册 219

HMMER, 注册 243

Informix, 注册 260

Microsoft SQL Server, 注册 273

ODBC, 注册 285

OLE DB, 注册 298

Oracle, 注册 308

Sybase, 注册 321

Teradata, 注册 344

Web 服务数据源 358, 359

WebSphere Business Integration 116

XML, 注册 389

包装器库 360

表结构文件名 332

手工链接 36

BioRS 文件名 65

BLAST 文件名 97

DB2 系列文件名 148

Documentum 文件名 160

Entrez 文件名 183

Excel 文件名 204

Extended Search 文件名 220

HMMER 文件名 244

Informix 文件名 261

Microsoft SQL Server 文件名 274

ODBC 文件名 286

OLE DB 文件名 298

Oracle 文件名 309

包装器库 (续)

Sybase 文件名 322

Teradata 文件名 344

WebSphere Business Integration 包装器

文件名 116

XML 文件名 389

包装器选项

有效设置 521

本地目录

请参阅全局目录 7

表结构文件

包装器库文件 332

概述 329

类型 329

昵称, 有效对象, 用于 14

配置存取 331

使用 DB2 Information Integrator 存取

330

示例 329

数据类型, 支持的 479

添加至联合系统 331

文件存取控制模型 335

消息 335

样本查询 330

优化 335

支持的版本 5

注册包装器 332

注册服务器定义 333

注册昵称 333

CREATE NICKNAME 语法 511

CREATE NICKNAME 语句示例 334

补偿, 描述 9

## [ C ]

参数

FEDERATED 38

残障人士 593

操作输出实用程序

配置联合服务器 51

查询

存取数据源

商业应用程序 142

BioRS 72, 74

BLAST 105

Documentum 172, 173

Entrez 188, 189

Excel 205

Extended Search 225

HMMER 252

Web 服务 378

查询 (续)

存取数据源 (续)

XML 400

分段 8

示例

商业应用程序 139, 375

Web 服务包装器 139, 375

用于 XML 文件的成本模型 393

查询优化

描述 8

BioRS 查询指南 79

XML 399

重复列属性

Documentum 167

传递 (Pass-Through)

描述 9

限制 9

创建包装器

表结构文件 332

BioRS 65

BLAST 97

DB2 系列数据源 147

Documentum 160

Entrez 182

Excel 文件 203

Extended Search 219

HMMER 243

Informix 260

ODBC 285

OLE DB 298

Oracle 308

Sybase 321

Teradata 344

XML 389

存取方案

描述 8

错误检测

Web 服务包装器 381

## [ D ]

代码页 42

联合系统 39

调整

数据源配置

Informix 267

Microsoft SQL Server 281

ODBC 296

Oracle 314

Sybase 328

Teradata 351

- 定义行语法分析
  - 描述 99
- 定义行语法分析 UDF
  - 概述 420
  - 描述 99
  - LSDefineParse 421
- 定制函数
  - BioRS 63, 64, 70
  - Documentum 167
  - Entrez 181, 186
  - Extended Search 222
- 对象
  - 命名 19
- 对象发现代理
  - 商业应用程序 111
  - 商业应用程序包装器 112
- 多分区配置 53

## [ F ]

- 方案
  - 存取 Excel 文件 206
  - 注册昵称 56
- 非关系数据源
  - 昵称列 58
  - 支持的数据类型 479
  - 指定数据类型映射 15
- 分布式数据库管理系统 3
- 分解业务对象
  - 昵称 109
- 服务器定义
  - 表结构文件 333
  - 描述 12
  - 商业应用程序 117
  - 注册概述 55
  - BioRS 66
  - BLAST 98
  - DB2 系列数据源 149
  - Documentum 161
  - Entrez 183
  - Excel 文件 204
  - Extended Search 220
  - HMMER 244
  - Informix 261
  - Microsoft SQL Server 274
  - ODBC 287
  - OLE DB 299
  - Oracle 309
  - Sybase 322
  - Teradata 345
  - Web 服务 360
  - WebSphere Business Integration 包装器 117
  - XML 390
- 服务器类型
  - 有效联合类型 523

- 服务器选项
  - 临时 12
  - 描述 12
  - 有效设置 529
- 辅助功能
  - 功能部件 593

## [ G ]

- 改变
  - 昵称
    - 本地数据类型 491
    - 概述 485
    - 昵称选项 489
- 高分值片段对 87
- 更新
  - 统计信息
    - 规划 21
- 固定列
  - BLAST 昵称 99
  - HMMER 昵称 247
- 故障诊断
  - 测试 DB2 系列服务器连接 152
  - 测试 Informix 服务器连接 265
  - 测试 Microsoft SQL Server 服务器连接 278
  - 测试 ODBC 服务器连接 290
  - 测试 Oracle 服务器连接 312
  - 测试 Sybase 服务器连接 326
  - 测试 Teradata 服务器连接 340, 348
  - Microsoft SQL Server 配置 281
  - ODBC 配置 296
  - Oracle 配置 314
  - Sybase 配置 328
  - Teradata 配置 351
  - Web 服务包装器 381
- 规划
  - 存取 Excel 数据 22
  - 非关系昵称列 58
  - 函数映射 27
  - 检查联合服务器设置 33
  - 联合服务器
    - Microsoft SQL Server 270
    - ODBC 284
  - 联合系统配置的核对表 27
  - 数据类型映射 26
  - 数据源环境变量 52
  - 选择正确的包装器 22
  - 用户映射 25
  - 注册包装器 54
  - 注册服务器定义 55
  - 注册用户映射 56

## [ H ]

- 函数模板
  - Extended Search 222
- 函数映射
  - 规划 27
  - 描述 16
  - 选项
    - 有效设置 583
- 核对表
  - 规划联合系统配置 27
- 环境变量
  - 概述, 关于联合 52
  - 在联合多分区实例中 53
  - Documentum 158
  - Informix 257
  - Microsoft SQL Server 271
  - Oracle 304
  - Sybase 318
  - Teradata 341

## [ J ]

- 基码频率表 444, 445
- 基因组数据库 179
- 键盘快捷方式
  - 支持 593
- 节点
  - 编目, 为 DB2 系列数据源 146

## [ K ]

- 客户机库
  - Documentum 158
- 控制中心
  - 联合系统接口 17
  - 配置数据源 49, 50
- 库
  - 包装器 34, 35
  - 手工链接至客户机库 36
  - 用户定义的函数 411
  - Informix 包装器 261
  - Microsoft SQL Server 包装器 274
  - Oracle 包装器 309
  - Sybase 包装器 322
  - Teradata 包装器 344

## [ L ]

- 联合服务器 4
  - 包装器 10
  - 包装器模块 10
  - 检查设置 33
  - 描述 4
  - 配置多个服务器 51



- 联合视图
  - 商业应用程序 368
  - 为非根昵称 399
- 联合数据库
  - 创建 44
  - 描述 7
  - 系统目录 7
  - 整理顺序 39, 41
- 联合系统
  - 概述 3
- 连接器配置程序
  - 业务对象应用程序配置 112
- 链接编辑
  - 请参阅 djxlink 33
- 列名
  - 更改 488
- 列选项
  - 描述 15
  - 为昵称指定 490
  - 有效设置 551

## [ M ]

- 密码
  - 区分大小写的值 20
- 命令行处理器 (CLP)
  - 联合函数 17
- 命令中心
  - 配置数据源 49
  - 用于联合 17
- 命名规则
  - 联合数据库对象 19
  - 联合数据源的核对表 27
- 模板
  - 业务对象 111
- 模式匹配 UDF
  - LSPatternMatch 426
  - LSPrositePattern 428
- 目录
  - 请参阅全局目录 519

## [ N ]

- 内置函数 16
- 昵称
  - 创建
    - 表结构文件 333
    - 商业应用程序数据源 118
    - 在昵称上 58
    - BioRS 67
    - Entrez 185
    - Extended Search 215
    - HMMER 246
    - Informix 266
    - Microsoft SQL Server 279

- 昵称 (续)
  - 创建 (续)
    - ODBC 291
    - Oracle 313
    - Sybase 327
    - Teradata 349, 350
    - Web 服务 361, 369
    - WebSphere Business Integration 数据源 118
    - XML 394
  - 更改
    - 本地列名 488
    - 本地数据类型 491
    - 本地数据类型, 示例 492
    - 概述 485
    - 列选项 490
    - 昵称选项 489
    - 限制 486
  - 描述 13, 125, 367
  - 商业应用程序 109
  - 数据库技术 139, 375
  - 有效数据源对象 14
  - 指定
    - 非关系数据源的列 58
  - 注册
    - 概述 56
    - DB2 系列数据源 153
    - Web 服务描述语言 (WSDL) 353
  - 昵称层次结构, 示例 126
  - 昵称和列选项
    - Web 服务包装器 120, 353, 363
    - WebSphere business integration 包装器 120, 363
    - 昵称层次结构, 示例 126
  - 昵称列选项
    - 描述 15
    - 示例 490
  - 昵称统计信息
    - BioRS 81, 83
  - 昵称选项
    - 指定 489
  - 逆向类型映射
    - 缺省映射 573
  - 逆向 UDF
    - LSRevComp 437
    - LSRevNuc 438
    - LSRevPep 439

## [ P ]

- 配置
  - 联合数据源
    - 概述 59
  - 配置数据源
    - 昵称选项 543
    - 使用 DB2 控制中心 50

- 平面文件
  - 另请参阅表结构文件 5

## [ Q ]

- 区分大小写
  - 保留区分大小写的值 20
  - 联合数据源的核对表 27
- 全局目录
  - 包含联合信息的视图 519
  - 描述 7
- 权限
  - 表结构文件的文件存取 335
  - Excel 文件的文件存取 207
- 缺省类型映射
  - 缺省映射 557

## [ S ]

- 商业应用程序
  - 包装器 116
  - 包装器库文件 116
  - 服务器定义 117
  - 联合视图 126
  - 配置适配器 112
  - 使用包装器来存取 111
  - 数据类型, 支持的 479
  - 样本查询 142
- 商业应用程序数据源
  - 联合视图 368
  - 配置存取 115
  - 添加至联合系统 115
  - 注册昵称 118
- 生命科学用户定义的函数
  - 除去 414
  - 概述 411
  - 概述, 定义行语法分析 UDF 420
  - 概述, 向后转换函数 415
  - 格式, 转换表 445
  - 函数列表 412
  - 示例, 转换表 446
  - 正则表达式支持 429
  - 注册 413
  - GeneWise 430
- 示例
  - 基码频率表 445
  - 商业应用程序包装器 139, 375
  - db2blast\_daemon 命令 96
  - db2hmmmer\_daemon 命令 242
  - Excel 文件 22, 201
  - Web 服务包装器 139, 369, 375
  - WebSphere Business Integration 包装器 126
  - XML 文档映射 385

示例, 查询

- 表结构文件 330
- 商业应用程序数据源 142
- BioRS 72, 74
- BLAST 105
- Documentum 172
- Entrez 189
- Excel 205
- Extended Search 225
- HMMER 250
- Web 服务 378
- XML 400

示例, 守护进程

- BLAST 配置文件 94
- HMMER 配置文件 240

示例, CREATE USER MAPPING

- Oracle 312
- Teradata 348

示例, CREATE USER MAPPING 语句

- DB2 系列数据源 151
- Informix 264
- Microsoft SQL Server 278
- ODBC 290
- OLE DB 301
- Sybase 325

示例, CREATE WRAPPER 语句

- ODBC 286

示例, UDF

- 生命科学的转换表 446
- LSBarCode 用户定义的函数 433
- LSDeflineParse 用户定义的函数 426
- LSMultiMatch 用户定义的函数 434
- LSMultiMatch3 用户定义的函数 435
- LSNuc2Pep 用户定义的函数 441
- LSPatternMatch 用户定义的函数 427
- LSPep2AmbNuc 用户定义的函数 416
- LSPep2ProbNuc 用户定义的函数 419
- LSPrositePattern 用户定义的函数 429
- LSRevComp 用户定义的函数 437
- LSRevNuc 用户定义的函数 439
- LSRevPep 用户定义的函数 440
- LSTransAllFrames 443

视图

- 联合, 为非根昵称 399
- 商业应用程序 126, 368

守护进程

- 配置文件
  - BLAST 示例 94
  - HMMER 示例 240
- 为 BLAST 配置 91
- 为 BLAST 启动 95
- 为 HMMER 配置 237
- 为 HMMER 启动 241

数据类型

- 不受支持 15
- 用于非关系型数据源 479

数据类型映射

- 规划 26
- 联合数据源的核对表 27
- 描述 15
- 逆向 573
- 为特定数据源对象 491, 492
- 正向 557

数据银行

- 描述 61

数据源 7, 8

- 可选的配置步骤 59
- 描述 4
- 配置 49
- 缺省包装器名称 11
- 商业应用程序 115
- 有效服务器类型 523

数据源对象

- 描述 13
- 有效对象类型 14

数据元素

- 为 Entrez 转换 188

索引规范

- 描述 16

## [ T ]

通用查询语言

- Extended Search 227

统计信息

- 对于 BioRS 80
- 更新数据源 21
- 联合数据源的核对表 27

途径数据库

- 函数列表 447

途径数据库 UDF

- CompoundsByPathwyS 453
- CompoundsByPathwyT 453
- EnzymesByPathwyS 454
- EnzymesByPathwyT 455
- GenesByPathwyS 451
- GenesByPathwyT 452
- PathwysByCompndsS 457
- PathwysByCompndsT 用户定义的函数 457
- PathwysByEnzymesS 458
- PathwysByEnzymesT 459
- PathwysByGenesS 455
- PathwysByGenesT 456

图谱 UDF

- LSBarCode 432
- LSMultiMatch 434
- LSMultiMatch3 435

## [ W ]

外键

- 与昵称配合使用 126

伪列

- Documentum 162

谓词

- Entrez 188

文件类型

- 表结构文件 329

## [ X ]

下推分析

- 描述 8

限制

- 改变昵称 486
- db2dj.ini 文件 53

向后转换 UDF

- 概述 415
- LSPep2AmbNuc 415
- LSPep2ProbNuc 418

消息

- 表结构文件包装器 335
- BioRS 包装器 83
- BLAST 包装器 107
- djxlink 35
- Documentum 包装器 173
- Entrez 包装器 197
- Excel 包装器 207
- Extended Search 包装器 228
- HMMER 包装器 253
- LSPep2AmbNuc 用户定义的函数 418
- LSPep2ProbNuc 用户定义的函数 420
- Web 服务包装器 381
- XML 包装器 401

序列相似性数据库

- 返回的列 459

函数列表 447

序列相似性数据库 UDF

- AllNbrsByGeneS 460
- AllNbrsByGeneT 461
- BestHmlgsByGeneS 469
- BestHmlgsByGeneT 470
- BestNbrsByGeneS 464
- BestNbrsByGeneT 465
- BstBstHmlgByGeneS 471
- BstBstHmlgByGeneT 471
- BstBstNbrsByGeneS 462
- BstBstNbrsByGeneT 463
- DefinitionsByGeneS 473
- GenesByMotifsS 473
- ParalogsByGeneS 468
- ParalogsByGeneT 469
- RevBestNbrsByGeneS 466
- RevBestNbrsByGeneT 467

序列相似性数据库 UDF (续)

ScoreBetweenGenesS 472

选项

昵称 543

## [ Y ]

样本

查询

Extended Search 223

业务对象 111

添加至联合系统 115

WebSphere Business Integration 包装器  
109

业务对象应用程序

WebSphere Business Integration 包装器  
109

业务对象应用程序包装器

示例 139, 375

应用程序数据实体

业务对象定义 111

映射

XML 文档 125, 367

映射字段

Extended Search 216

用户标识

区分大小写的值 20

用户定义的函数

KEGG 447

用户定义的函数 (UDF) 16

除去, 生命科学 414

除去, KEGG 474

生命科学 411, 412

注册, 生命科学 413

注册, KEGG 450

KEGG 447

用户定义的类型 (UDT)

不受支持的数据类型 15

用户映射

规划 25

联合数据源的核对表 27

描述 12

选项 12

有效设置 541

注册概述 56

Web 服务包装器 381

优化

表结构文件 335

BLAST 107

优化器

描述 8

语法

KEGG 用户定义的函数 449

远程目录信息 7

## [ Z ]

正则表达式支持

生命科学用户定义的函数 429

整理顺序

联合系统 39, 41

主键

与昵称配合使用 126

转换表

生命科学用户定义的函数

格式 445

示例 446

转换 UDF

LSNuc2Pep 440

LSTransAllFrames 442

纵向表

Extended Search 216

## A

ACCOUNTING\_STRING 用户选项

有效设置 541

AllNbrsByGeneS 用户定义的函数

语法 460

AllNbrsByGeneT 用户定义的函数

语法 461

AllText 元素

BioRS 74

ALTER NICKNAME 语句

示例

更改本地列名 488

更改列选项 490

限制 486

ALTER NICKNAME 语句

示例

本地数据类型 492

## B

BestHmlgsByGenesS 用户定义的函数

语法 469

BestHmlgsByGenesT 用户定义的函数

语法 470

BestNbrsByGeneS 用户定义的函数

语法 464

BestNbrsByGeneT 用户定义的函数

语法 465

Biomax 信息学 61

BioRS

包装器库文件 65

定制函数 63, 70

描述 61

配置存取 62

使用连接谓词 72

数据类型, 支持的 479

添加至联合系统 62

BioRS (续)

统计信息

列基数 81, 83

昵称基数 81

数据银行基数 80

维护 80

消息 83

样本查询 72, 74

优化包装器性能 79

注册包装器 65

注册定制函数 64

注册服务器定义 66

注册昵称 67

AllText 元素 74

CREATE NICKNAME 语法 496

CREATE NICKNAME 语句示例 69

CREATE SERVER 语句 495

CREATE USER MAPPING 语句 66,  
496

BLAST

包装器库文件 97

查询 105

定义行语法分析 99

高分值片段对 (HSP) 87

固定输入和输出列 99

昵称, 有效对象, 用于 14

配置存取 90

配置 TurboBlast 104

守护进程

配置 91

启动 95

守护进程配置文件

示例 94

数据类型, 支持的 479

添加至联合系统 90

验证是否安装了正确的矩阵文件 91

验证是否安装了正确的 blastall 可执  
行文件 91

消息 107

样本查询 105

优化配置 107

支持的版本 5

注册包装器 97

注册服务器定义 98

注册昵称 98

CREATE NICKNAME 语法 498

CREATE NICKNAME 语句 98

CREATE NICKNAME 语句示例 103

CREATE SERVER 语句 498

BLASTALL\_PATH

BLAST 守护进程 91

BLAST\_OUT\_DIR\_PATH

BLAST 守护进程 91

BstBstHmlgByGenesS 用户定义的函数

语法 471

BstBstHmlgByGenesT 用户定义的函数  
语法 471  
BstBstNbrsByGeneS 用户定义的函数  
语法 462  
BstBstNbrsByGeneT 用户定义的函数  
语法 463

## C

CATALOG DATABASE 命令  
存取 DB2 系列数据源 147  
CATALOG TCPIP NODE 命令  
存取 DB2 系列数据源 146  
CLP (命令行处理器)  
联合函数 17  
COLLATING\_SEQUENCE 服务器选项  
有效设置 529  
COMM\_RATE 服务器选项  
有效设置 529  
CompoundsByPathwyS 用户定义的函数  
语法 453  
CompoundsByPathwyT 用户定义的函数  
语法 453  
CONNECTSTRING 服务器选项  
有效设置 529  
CPU\_RATIO 服务器选项  
有效设置 529  
CREATE FUNCTION 语句  
Extended Search 222, 510  
CREATE INDEX 语句 16  
CREATE NICKNAME 语法  
表结构文件 511  
BioRS 496  
BLAST 498  
Documentum 501  
Excel 504  
Extended Search 507  
XML 515  
CREATE NICKNAME 语句 58  
表结构文件 333, 334  
商业应用程序数据源 118  
BioRS 67, 69  
BLAST 98, 103  
DB2 系列数据源 153, 154  
Documentum 162, 165  
Entrez 185, 511  
Excel 文件 205  
Extended Search 221  
HMMER 246, 250  
Informix 266  
Microsoft SQL Server 279, 280  
ODBC 291, 292  
Oracle 313, 314  
Sybase 327, 328  
Teradata 350  
Web 服务 361

CREATE NICKNAME 语句 (续)  
WebSphere Business Integration 包装器  
118  
XML 394  
CREATE SERVER 语句 4  
表结构文件 333  
BioRS 66, 495  
BLAST 98, 498  
DB2 系列数据源 149, 150  
Documentum 161, 500  
Entrez 183, 510  
Excel 503  
Excel 文件 204  
Extended Search 220, 506  
HMMER 245  
Informix 261, 262  
Microsoft SQL Server 274, 275  
ODBC 287, 288  
OLE DB 299  
Oracle 309, 310  
Sybase 322, 323  
Teradata 345, 346  
Web 服务 360  
XML 390  
CREATE USER MAPPING 语句  
BioRS 66, 496  
DB2 系列数据源 151  
Documentum 161, 500  
Extended Search 221, 507  
Informix 263, 264  
Microsoft SQL Server 277, 278  
ODBC 289, 290  
OLE DB 300, 301  
Oracle 311, 312  
Sybase 325  
Teradata 347, 348  
CREATE WRAPPER 语句  
Extended Search 505  
ODBC 286

## D

DAEMON\_LOGFILE\_DIR  
BLAST 守护进程 91  
HMMER 守护进程 237  
DAEMON\_PORT  
BLAST 守护进程 91  
HMMER 守护进程 237  
DATALINK 数据类型  
不受支持 15  
DATEFORMAT 服务器选项  
有效设置 529  
DB2 系列数据源  
包装器库文件 148  
编目节点条目 146  
编目远程数据库 147  
DB2 系列数据源 (续)  
测试服务器连接 152  
配置存取 145  
添加至联合系统 145  
注册包装器 147  
注册服务器定义 149  
注册昵称 153  
注册用户映射 151  
CREATE NICKNAME 语句示例 154  
CREATE SERVER 语句 150  
CREATE USER MAPPING 语句 151  
DB2 iSeries 版  
昵称, 有效对象, 用于 14  
缺省包装器名称 11  
缺省逆向类型映射 573  
缺省正向类型映射 557  
有效服务器类型 523  
支持的版本 5  
DB2 Linux 版、UNIX 版和 Windows 版  
昵称, 有效对象, 用于 14  
缺省包装器名称 11  
缺省逆向类型映射 573  
缺省正向类型映射 557  
有效服务器类型 523  
支持的版本 5  
DB2 VM 和 VSE 版  
昵称, 有效对象, 用于 14  
缺省包装器名称 11  
缺省逆向类型映射 573  
缺省正向类型映射 557  
有效服务器类型 523  
支持的版本 5  
DB2 z/OS 和 OS/390 版  
昵称, 有效对象, 用于 14  
缺省包装器名称 11  
缺省逆向类型映射 573  
缺省正向类型映射 557  
有效服务器类型 523  
支持的版本 5  
db2blast\_daemon 命令 96  
要发出的步骤 95  
db2dj.ini  
多分区实例配置 53  
限制 53  
db2hmmmer\_daemon 命令  
示例 242  
要发出的步骤 241  
db2nodes.cfg 文件  
联合数据库 44  
DB2\_DJ\_COMM 环境变量  
Informix, 调整 267  
Microsoft SQL Server, 调整 281  
ODBC, 调整 296  
Oracle, 调整 314  
Sybase, 调整 328  
Teradata, 调整 351

DB2\_MAXIMAL\_PUSHDOWN 服务器选项  
有效设置 529

DBNAME 服务器选项  
有效设置 529

DefinitionsByGeneS 用户定义的函数  
语法 473

DISABLE 函数映射选项  
有效设置 583

djxlink  
错误消息文件 35  
确认与客户机库的连接 33  
手工链接至客户机库 36

dmcl.ini 文件  
Documentum 158

Documentum  
包装器库文件 160  
定制函数 167  
环境变量, 设置 158  
客户机库 158  
描述 155  
昵称列 167  
昵称, 有效对象, 用于 14  
配置存取 157  
示例 155  
数据类型, 支持的 479  
双定义重复属性 167  
添加至联合系统 157  
伪列 162  
消息 173  
样本查询 172  
用户对文档的存取 173  
支持的版本 5  
注册包装器 160  
注册服务器定义 161  
注册昵称 162  
注册用户映射 161  
CREATE NICKNAME 语法 501  
CREATE NICKNAME 语句 162  
CREATE NICKNAME 语句示例 165  
CREATE SERVER 语句 500  
CREATE USER MAPPING 语句 161, 500

## E

Entrez  
包装器库文件 183  
定制函数 181, 186  
关系谓词 188  
描述 179  
昵称选项 511  
昵称, 有效对象, 用于 14  
配置存取 180  
添加至联合系统 180  
无效 WHERE 子句 188  
消息 197

Entrez (续)  
样本查询 189  
支持的版本 5  
注册包装器 182  
注册定制函数 181  
注册服务器定义 183  
注册昵称 185  
CREATE SERVER 语句 510  
Nucleotide 模式表 194  
PubMed 模式表 190  
SQL 模式中的数据元素 188

EnzymesByPathwyS 用户定义的函数  
语法 454

EnzymesByPathwyT 用户定义的函数  
语法 455

Excel 文件  
包装器库文件 204  
描述 201  
昵称, 有效对象, 用于 14  
配置存取 202  
使用 ODBC 包装器来存取 293  
数据类型, 支持的 479  
添加至联合系统 202  
注册昵称 205  
CREATE NICKNAME 语句 205  
文件存取控制模型 207  
消息 207  
样本查询 205  
样本用户方案 206  
支持的版本 5  
注册包装器 203  
注册服务器定义 204  
准备存取 22  
CREATE NICKNAME 语法 504  
CREATE SERVER 语句 503

Extended Search  
包装器库文件 220  
查询基准 223  
创建多个昵称 215  
定制查询 222  
可搜索源 213  
描述 213  
昵称, 有效对象, 用于 14  
配置存取 218  
数据类型, 支持的 479  
添加至联合系统 218  
通用查询语言 227  
消息 228  
样本查询 225  
映射字段 216  
支持的版本 5  
注册包装器 219  
注册服务器定义 220  
注册函数模板 222  
注册昵称 221  
注册用户映射 221

Extended Search (续)  
纵向表 216  
CREATE FUNCTION 语句 222, 510  
CREATE NICKNAME 语法 507  
CREATE NICKNAME 语句 221  
CREATE SERVER 语句 506  
CREATE USER MAPPING 语句 221, 507  
CREATE WRAPPER 语句 505

## F

FaultQueue  
业务对象应用程序配置 112

FEDERATED 参数  
设置 38

FOLD\_ID 服务器选项  
区分大小写的值 20  
有效设置 529  
在 Informix 数据源上设置 267

FOLD\_PW 服务器选项  
区分大小写的值 20  
有效设置 529  
在 Informix 数据源上设置 267

## G

GenesByMotifsS 用户定义的函数  
语法 473

GenesByPathwyS 用户定义的函数  
语法 451

GenesByPathwyT 用户定义的函数  
语法 452

GeneWise 430, 432

GeneWise UDF  
链接至库 430

## H

H2X\_PATH, HMMER 守护进程 237

HMMER  
包装器库文件 244  
昵称的固定列 247  
CREATE SERVER 语句示例 245

HMMER 数据源  
查询 252  
描述 233  
昵称, 有效对象, 用于 14  
配置存取 236  
实用程序列表 233  
守护进程  
配置 237  
启动 241  
守护进程配置文件示例 240  
数据类型, 支持的 479

HMMER 数据源 (续)  
添加至联合系统 236  
消息 253  
样本查询 250  
支持的版本 5  
注册包装器 243  
注册服务器定义 244  
注册昵称 246  
CREATE NICKNAME 语句示例 250  
hmmpfam 可执行文件 237  
HMMERPFAM\_PATH, HMMER 守护进程 237  
HMMER\_OUT\_DIR\_PATH  
HMMER 守护进程 237  
hmmpfam 可执行文件  
描述 233  
验证已安装的版本 237

**I**

IFILE 服务器选项  
有效设置 529  
IGNORE\_UDT 服务器选项  
有效设置 529  
Informix  
包装器库文件名 261  
测试服务器连接 265  
测试客户机配置文件 256  
调整 267  
环境变量, 设置 257  
昵称, 有效对象, 用于 14  
配置存取 255  
缺省包装器名称 11  
缺省逆向类型映射 573  
缺省正向类型映射 557  
设置客户机配置文件 256  
添加至联合系统 255  
有效服务器类型 523  
支持的版本 5  
注册包装器 260  
注册服务器定义 261  
注册昵称 266  
注册用户映射 263  
CREATE NICKNAME 语句示例 266  
CREATE SERVER 语句 262  
CREATE USER MAPPING 语句 264  
INFORMIX\_LOCK\_MODE 服务器选项  
有效设置 529  
INITIAL\_INSTS 函数映射选项  
有效设置 583  
INITIAL\_IOS 函数映射选项  
有效设置 583  
INSTS\_PER\_ARGBYTE 函数映射选项  
有效设置 583  
INSTS\_PER\_INVOC 函数映射选项  
有效设置 583

IOS\_PER\_ARGBYTE 函数映射选项  
有效设置 583  
IOS\_PER\_INVOC 函数映射选项  
有效设置 583  
IO\_RATIO 服务器选项  
有效设置 529  
IUD\_APP\_SVPT\_ENFORCE 服务器选项  
有效设置 529

## K

KEGG 用户定义的函数  
除去 474  
概述 447  
函数列表 447  
注册 450  
自变量描述 449  
SSDB 返回的列 459

## L

LOGIN\_TIMEOUT 服务器选项  
有效设置 529  
LSBarCode 用户定义的函数  
示例 433  
语法 432  
LSDeflineParse 用户定义的函数  
示例 426  
语法 421  
LSGeneWise 用户定义的函数 430, 432  
库 411  
LSMultiMatch 用户定义的函数  
示例 434  
语法 434  
LSMultiMatch3 用户定义的函数  
示例 435  
语法 435  
LSNuc2Pep 用户定义的函数  
示例 441  
语法 440  
LSPatternMatch 用户定义的函数  
示例 427  
语法 426  
LSPep2AmbNuc 用户定义的函数  
错误消息 418  
示例 416  
语法 415  
LSPep2ProbNuc 用户定义的函数  
错误消息 420  
基码频率表格式 444  
示例 419  
语法 418  
LSPrositePattern 用户定义的函数  
示例 429  
语法 428

LSRevComp 用户定义的函数  
示例 437  
语法 437  
LSRevNuc 用户定义的函数  
示例 439  
语法 438  
LSRevPep 用户定义的函数  
示例 440, 443  
语法 439, 442

## M

MAX\_PENDING\_REQUESTS  
BLAST 守护进程 91  
HMMER 守护进程 237  
Microsoft Excel  
请参阅 Excel 文件 5  
Microsoft SQL Server  
包装器库文件名 274  
测试服务器连接 278  
调整 281  
故障诊断 281  
环境变量, 设置 271  
昵称, 有效对象, 用于 14  
配置存取 269  
缺省包装器名称 11  
缺省逆向类型映射 573  
缺省正向类型映射 557  
添加至联合系统 269  
有效服务器类型 523  
支持的版本 5  
注册包装器 273  
注册服务器定义 274  
注册昵称 279  
注册用户映射 277  
准备联合服务器 270  
CREATE NICKNAME 语句示例 280  
CREATE SERVER 语句 275  
CREATE USER MAPPING 语句示例 278  
MODULE 包装器选项  
有效设置 521

**N**

NET8 包装器  
库文件名, Oracle 309  
NLS (本地语言支持) 39  
NODE 服务器选项, 有效设置 529  
Nucleotide  
模式表 194  
数据库 179  
NUMERIC\_STRING 列选项  
示例 490  
有效设置 551

# O

## ODBC

- 包装器库文件 286
- 测试服务器连接 290
- 调整 296
- 故障诊断 296
- 昵称, 有效对象, 用于 14
- 配置存取 283
- 缺省包装器名称 11
- 缺省正向类型映射 557
- 添加至联合系统 283
  - 注册昵称 291
- 有效服务器类型 523
- 支持的版本 5
- 注册包装器 285
- 注册服务器定义 287
- 注册用户映射 289
- 准备联合服务器 284
- CREATE NICKNAME 语句示例 292
- CREATE SERVER 语句 288
- CREATE USER MAPPING 语句 290
- CREATE WRAPPER 语句 286

## ODBC 包装器

- 存取 Excel 文件 293

## OLE DB

- 包装器库文件 298
- 配置存取 297
- 缺省包装器名称 11
- 添加至联合系统
  - 概述 297
- 有效服务器类型 523
- 支持的版本 5
- 注册包装器 298
- 注册服务器定义 299
- 注册用户映射 300
- CREATE SERVER 语句 299
- CREATE USER MAPPING 语句 301

## OMIM 数据库 179

## Oracle

- 包装器名称 309
- 测试服务器连接 312
- 测试客户机配置文件 307
- 调整 314
- 故障诊断 314
- 环境变量, 设置 304
- 库名 309
- 昵称, 有效对象, 用于 14
- 配置存取 303
- 缺省包装器名称 11
- 缺省逆向类型映射 573
- 缺省正向类型映射 557
- 设置客户机配置文件 307
- 示例, CREATE USER MAPPING 语句 312
- 添加至联合系统 303

## Oracle (续)

- 注册昵称 313
- 注册包装器 308
- 注册服务器定义 309
- 注册用户映射 311
- CREATE NICKNAME 语句示例 314
- CREATE SERVER 语句 310

# P

## PACKET\_SIZE 服务器选项

- 有效设置 529

## ParalogsByGeneS 用户定义的函数

- 语法 468

## ParalogsByGeneT 用户定义的函数

- 语法 469

## PASSWORD 服务器选项

- 有效设置 529

## PathwysByCompndsS 用户定义的函数

- 语法 457

## PathwysByCompndsT 用户定义的函数

- 语法 457

## PathwysByEnzymesS 用户定义的函数

- 语法 458

## PathwysByEnzymesT 用户定义的函数

- 语法 459

## PathwysByGenesS 用户定义的函数

- 语法 455

## PathwysByGenesT 用户定义的函数

- 语法 456

## PCRE 库程序包 429

## PeopleSoft

- 业务对象 109

## PERCENT\_ARGBYTES 函数映射选项

- 有效设置 583

## PLAN\_HINTS 服务器选项

- 有效设置 529

## PubMed 模式表 190

## PubMed 数据库 179

## PUSHDOWN 服务器选项

- 有效设置 529

# Q

## Q\_SEQ\_DIR\_PATH

- BLAST 守护进程 91
- HMMER 守护进程 237

# R

## REMOTE\_AUTHID 用户选项

- 有效设置 541

## REMOTE\_DOMAIN 用户选项

- 有效设置 541

## REMOTE\_NAME 函数映射选项

- 有效设置 583

## REMOTE\_PASSWORD 用户选项

- 有效设置 541

## RequestQueue

- 业务对象应用程序配置 112

## ResponseQueue

- 业务对象应用程序配置 112

## RevBestNbrsByGeneS 用户定义的函数

- 语法 466

## RevBestNbrsByGeneT 用户定义的函数

- 语法 467

## RUNPFAM\_PATH

- HMMER 守护进程 237

# S

## SAP

- 业务对象 109

## ScoreBetweenGenesS 用户定义的函数

- 语法 472

## SET SERVER OPTION 语句

- 临时设置选项 12

## Siebel

- 业务对象 109

## SQL 编译器

- 在联合系统中 8

## SQL 语言

- 描述 9

## SQLNET 包装器

- 库文件名 309

## Sybase

- 包装器库文件名 322
- 测试服务器连接 326
- 测试客户机配置文件 320
- 调整 328
- 故障诊断 328
- 环境变量, 设置 318
- 昵称, 有效对象, 用于 14
- 配置存取 317
- 缺省包装器名称 11
- 缺省逆向类型映射 573
- 缺省正向类型映射 557
- 设置客户机配置文件 320
- 示例, CREATE USER MAPPING 语句 325
- 添加至联合系统 317
  - 注册昵称 327
- 有效服务器类型 523
- 支持的版本 5
- 注册包装器 321
- 注册服务器定义 322
- 注册用户映射 325
- CREATE NICKNAME 语句示例 328
- CREATE SERVER 语句 323
- SYSCAT 目录视图 519

**T**

## Teradata

- 包装器库文件名 344
- 测试服务器连接 340, 348
- 存取记录 351
- 调整 351
- 故障诊断 351
- 环境变量, 设置 341
- 联合服务器上的昵称, 概述 349
- 昵称, 有效对象, 用于 14
- 配置存取 339
- 缺省包装器名称 11
- 缺省逆向类型映射 573
- 缺省正向类型映射 557
- 添加至联合系统 339
  - 验证运行时链接 (AIX) 341
  - 注册昵称 350
- 有效服务器类型 523
- 运行时链接, 启用 (AIX) 341, 351
- 注册包装器 344
- 注册服务器定义 345
- 注册用户映射 347
- CREATE NICKNAME 语句示例 350
- CREATE SERVER 语句 346
- CREATE USER MAPPING 语句 348

## TIMEFORMAT 服务器选项

- 有效设置 529

## TIMEOUT 服务器选项

- 有效设置 529

## TIMESTAMPFORMAT 服务器选项

- 有效设置 529

## TurboBlast 104

**U**

## Unicode 42

**V**

## VARCHAR\_NO\_TRAILING\_ BLANKS 服务器选项

- 有效设置 529

## VARCHAR\_NO\_TRAILING\_ BLANKS 列选项

- 示例 490
- 有效设置 551

**W**

## Web 服务

- 包装器库文件 360
- 故障诊断 381

## Web 服务 (续)

- 昵称 353, 369
- 配置存取 358
- 示例 139, 375
- 数据类型, 支持的 479
- 添加至联合系统 358
- 样本查询 378
- 注册服务器定义 360
- 注册昵称 361

## Web 服务包装器

- 创建 359

## Web 服务描述语言 (WSDL)

- 昵称 353
- 映射至昵称 125, 367

## Web 服务提供程序

- 故障诊断 381

## WebSphere Business Integration

- 配置 111

## WebSphere Business Integration 包装器 109

- 包装器库文件 116
- 创建 116
- 服务器定义 117
- 示例 126, 139, 375
- 数据类型, 支持的 479
- 注册昵称 118

## WebSphere Business Integration 适配器

- 安装 112

## WebSphere MQ

- 消息队列 112

**X**

## XML

- 包装器库文件 389
- 成本模型工具 393, 399
- 描述 385
- 昵称, 有效对象, 用于 14
- 配置存取 388
- 数据类型, 支持的 479
- 添加至联合系统 388
  - 注册昵称 391, 394
  - CREATE NICKNAME 语句 394
- 为非根昵称创建联合视图 399
- 消息 401
- 样本查询 400
- 源文档和昵称 391
- 支持的版本 5
- 注册包装器 389
- 注册服务器定义 390
- CREATE NICKNAME 语法 515
- CREATE NICKNAME 语句示例 394

## XML 模式

- 商业应用程序 111
- 映射至昵称 125, 367

## XPath

- 与昵称配合使用 125, 367



---

## 与 IBM 联系

在中国，请致电下列其中一个号码以与 IBM 联系：

- 800-810-1818 或 (010) 84981188 分机 5151，可获得售前客户服务；
- 800-810-1818 或 (010) 84981188 分机 5200，可获得售后客户服务；
- 800-810-1818 或 (010) 84981188 分机 5017，可获得市场营销与销售的信息；

要查找您所在国家或地区的 IBM 营业处，可在网上查看 IBM 全球联系人目录 (Directory of Worldwide Contacts)，网址为：[www.ibm.com/planetwide](http://www.ibm.com/planetwide)。

---

## 产品信息

关于 DB2 Information Integrator 的信息可通过万维网获取，网址为：<http://www-900.ibm.com/cn/software/db2/>。

此站点包含有关 DB2 产品家族、DB2 解决方案、技术前沿与趋势、DB2 服务、成功案例、市场活动、培训与认证、DB2 开发者园地、合作伙伴、下载中心、资料库、第三方分析报告、殊荣与奖项、DB2 新闻以及如何购买 DB2 的最新信息。

要查找您所在国家或地区的 IBM 营业处，可在网上查看 IBM 全球联系人目录 (Directory of Worldwide Contacts)，网址为：[www.ibm.com/planetwide](http://www.ibm.com/planetwide)。

---

## 对文档的意见

您的反馈有助于 IBM 提供高质量的信息。请发送您对本书或其它 DB2 Information Integrator 文档的任何意见。可以使用下列任何一种方法提出意见：

- 使用 [www.ibm.com/software/data/rcf](http://www.ibm.com/software/data/rcf) 上的在线读者意见表发送您的意见。
- 通过电子邮件 (e-mail) 将您的意见发送至 [ctscrcf@cn.ibm.com](mailto:ctscrcf@cn.ibm.com)。确保包括产品的名称、产品的版本号和书籍的名称及部件号 (如果适用的话)。如果您对特定文本有意见，请包括此文本的位置 (例如，标题、表号或页码)。







中国印刷

Spine information:



**IBM DB2 Information  
Integrator**

**数据源配置指南**

版本 8.2